

Berufsbildende Schule I Mainz



In Kooperation mit dem HERDT-Verlag stellen wir Ihnen eine PDF inkl. Zusatzmedien für Ihre persönliche Weiterbildung zur Verfügung. In Verbindung mit dem Programm HERDT|Campus ALL YOU CAN READ stehen diese PDFs nur Lehrkräften und Schüler*innen der oben genannten Lehranstalt zur Verfügung. Eine Nutzung oder Weitergabe für andere Zwecke ist ausdrücklich verboten und unterliegt dem Urheberrecht. Jeglicher Verstoß kann zivil- und strafrechtliche Konsequenzen nach sich ziehen.

Netzwerke Sicherheit

Siegmund Dehn

11. Ausgabe, April 2019

NWSI_2019



ISBN 978-3-86249-848-2

Bevor Sie beginnen ...	4	7 Spyware, Phishing und Browser Hijacking 71
1.1 Grundforderungen an Sicherheit	6	7.1 Geld verdienen im Internet 71
1.2 Sicherheitsziel Vertraulichkeit	6	7.2 Spyware 73
1.3 Sicherheitsziel Integrität	7	7.3 Browser Hijacking 76
1.4 Sicherheitsziel Verfügbarkeit	8	7.4 Was ist Phishing? 77
1.5 Rechtliche Aspekte	9	7.5 Anti-Spyware einsetzen 80
		7.6 Übung 84
2 Risikolage für Unternehmen	14	8 Stand-Alone-Virenschutz 85
2.1 Warum ist das Internet nicht „sicher“?	14	8.1 Einfache Virenprävention 85
2.2 Schadensmöglichkeiten	15	8.2 Gängige Antivirensoftware 91
2.3 Wie abhängig sind Firmen vom IT-Einsatz?	16	8.3 Computer scannen 95
		8.4 Viren entfernen 97
		8.5 Übung 98
3 Angriffsvorbereitung	18	9 IT-Sicherheitsstandard 99
3.1 Hacker und Cracker	18	9.1 Standards im Bereich Informationssicherheit 99
3.2 „Staatliche“ Hacker	19	9.2 IT-Grundschutz-Kompendium 99
3.3 Elektronische Kriegsführung	21	9.3 Weitere Kriterienwerke zur IT-Sicherheit 100
3.4 Netzwerkscans	21	9.4 DIN EN 50600 103
3.5 Wardriving	27	9.5 Security Policy 104
3.6 Social Engineering	27	9.6 Aufgaben eines IT-Sicherheitsbeauftragten 105
		9.7 Übung 106
4 Angriffe auf Serverdienste	31	10 Symmetrische Kryptografie 107
4.1 Exploits	31	10.1 Das Problem von Alice und Bob 107
4.2 Rootkits	37	10.2 Einfache Verschlüsselungsmethoden 109
4.3 DoS/DDoS/DRDoS	39	10.3 Symmetrische Verfahren 116
4.4 Sniffer	40	10.4 Übung 125
4.5 Replay-Attacken	42	
4.6 TCP/IP Session-Hijacking	42	
4.7 Übung	44	
5 Sicherheitsprobleme durch Mitarbeiter	45	11 Asymmetrische Kryptografie 126
5.1 Ausfall/Krankheit	45	11.1 Nachteile symmetrischer Verfahren 126
5.2 Unrechtmäßige Systemzugänge	46	11.2 Einwegfunktion 127
5.3 Spionage	47	11.3 Diffie-Hellman-Schlüsseltausch 131
5.4 Mangelnde Kompetenz	49	11.4 El-Gamal 132
5.5 Übung	51	11.5 RSA 133
		11.6 Digitale Signatur 136
		11.7 Hashfunktionen 137
		11.8 Schwachstellen in RSA 138
		11.9 Public Key Infrastructure 141
		11.10 Übung 144
6 Virenarten und ihre Verbreitung	52	12 Kryptografische Protokolle und ihre Anwendung 145
6.1 Grundkonzepte von Viren	52	12.1 SSL/TLS 145
6.2 Virenarten	54	12.2 SSH 150
6.3 Tarnmechanismen von Viren	59	12.3 IPsec 151
6.4 Würmer	66	12.4 Übung 152
6.5 Trojaner	67	
6.6 Adware und PUA	69	
6.7 Tendenzen und Ausblick	69	
6.8 Übung	70	

13 Sichere E-Mail-Verfahren	153	18 Alternative Software	204
13.1 Grundlagen der E-Mail-Verschlüsselung	153	18.1 Warum Nicht-Standard-Software sinnvoll sein kann	204
13.2 Schlüssel generieren	155	18.2 Alternative Webbrowser	206
13.3 Schlüsselexport und -import	157	18.3 Alternative E-Mail-Clients	208
13.4 Signieren von Schlüsseln	160		
13.5 E-Mail signieren und verschlüsseln	161		
13.6 Dateien signieren und verschlüsseln	162		
13.7 Übung	162		
14 Firewalls	163	19 Authentifizierungssysteme	211
14.1 Wie Firewalls arbeiten	163	19.1 Kerberos	211
14.2 Paketfilter-Firewall	165	19.2 PAP, CHAP, EAP und RADIUS	214
14.3 Stateful Inspection Firewall	167	19.3 Smartcards und Tokensysteme	217
14.4 Proxy Level/Application Level Firewall	168	19.4 Biometrie	218
14.5 NAT	169		
14.6 Personal Firewall	170		
14.7 Sicherheitskonzept Firewall	172		
14.8 Erweiterte Funktionen der Firewall	172		
14.9 Übung	173		
15 Intrusion-Detection/Prevention-Systeme	174	20 Proaktive Sicherheit	222
15.1 Notwendigkeit von Intrusion-Detection-Systemen	174	20.1 Defensive Programmierung	222
15.2 Arbeitsweise eines IDS	175	20.2 Gehärtete Betriebssysteme	223
15.3 Auf erkannte Angriffe reagieren	177	20.3 Patches	225
15.4 Intrusion-Prevention-Systeme (IPS)	178	20.4 Vulnerability Assessment	226
15.5 Snort	179		
15.6 Honeypot-Netzwerke	180		
15.7 Übung	182		
16 Virtual Private Network	183	Stichwortverzeichnis	232
16.1 Zielsetzung	183		
16.2 PPTP	184		
16.3 L2TP/IPsec	185		
16.4 OpenVPN	190		
16.5 Abgrenzung zu anderen VPN-Arten	190		
16.6 Übung	190		
17 WLAN und Sicherheit	191		
17.1 WLAN-Arbeitsweise	191		
17.2 Access-Points	195		
17.3 WEP – Wired Equivalency Protocol	196		
17.4 WPA – Wi-Fi Protected Access	197		
17.5 WPA2 – Wi-Fi Protected Access 2	198		
17.6 WPA3 – Wi-Fi Protected Access 3	198		
17.7 Weitere Authentifizierung und Verschlüsselung im WLAN	199		
17.8 Funkausleuchtung	200		
17.9 Übung	202		

Bevor Sie beginnen ...

HERDT BuchPlus – unser Konzept:

Problemlos einsteigen – Effizient lernen – Zielgerichtet nachschlagen

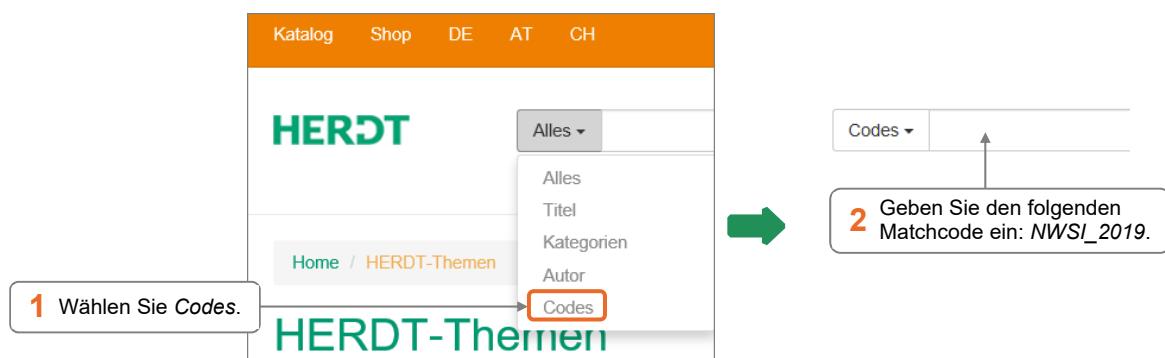
(weitere Infos unter www.herdt.com/BuchPlus)

Nutzen Sie unsere maßgeschneiderten, im Internet frei verfügbaren Medien:



So können Sie schnell auf die BuchPlus-Medien zugreifen:

- Rufen Sie im Browser die Internetadresse www.herdt.com auf.



Empfohlene Vorkenntnisse

- ✓ Grundkenntnisse im Bereich der Informationstechnologie
- ✓ Netzwerke – Grundlagen

Lernziele

Dieses Buch vermittelt Ihnen die Grundlagen zu wesentlichen Aspekten der Sicherheit in Netzwerken. Es beschreibt sowohl allgemeine Sicherheitsanforderungen als auch spezielle, die bei der Nutzung von Komponenten und Protokollen in Netzwerken entstehen.

Sie lernen die IT-Sicherheit vernetzter Systeme aus der Sicht unterschiedlicher Gruppen, wie Management, Administratoren und Benutzer, zu betrachten. Nach dem Durcharbeiten des Buches wissen Sie, dass für die Herstellung eines angemessenen Sicherheitsniveaus eine Analyse der möglichen Gefahren, des Bedarfs für Sicherheit und eine Abschätzung des Risikos vorausgehen müssen.

Sie kennen den Planungsablauf von IT-Sicherheitsmaßnahmen und sind mit den wesentlichen technischen und organisatorischen Maßnahmen vertraut, mit denen bestimmte Sicherheitsbedrohungen bekämpft werden können. Sie können selbstständig anhand der Ihnen bekannten Kriterien die optimale Sicherheitsmaßnahme für eine Problemstellung auswählen.

Hinweise zu Soft- und Hardware

Die im Buch beispielhaft vorgestellte Hard- und Software wurde nicht unter der Prämisse ausgewählt, das jeweils beste Produkt in dieser Kategorie zu sein. Für Schulungszwecke sind die vor gestellten Produkte jedoch geeignet, da sie z. B. im Falle von Free- oder Shareware für Sie relativ leicht und kostengünstig zur Verfügung stehen oder – wenn es sich bei der dargestellten Software um kommerzielle Software handelt – sich gut für eine Demonstration der zu vermittelnden Lehrinhalte eignen, aus der Sie die wichtigsten Erkenntnisse für die Arbeit mit ähnlicher Software ableiten können.

Da es sich um ein Buch handelt, das verschiedene Aspekte der IT-Sicherheit in Computersystemen und Netzwerken beleuchten soll und nicht nur einen speziellen Teil, wurden auch die Inhalte auf der Grundlage verschiedener Betriebssysteme erstellt.

Da in der Praxis Microsoft-basierte Betriebssysteme die größte Verbreitung besitzen, kommen in diesem Buch verschiedene Varianten dieser Systeme, z. B. Windows Server 2008/2012/2019 oder Windows 7/8/8.1/10 zum Einsatz.

Inhaltliche Gliederung

Das Buch erklärt zuerst die Grundlagen der IT-Sicherheit und die Notwendigkeit entsprechender Maßnahmen. Anschließend werden die häufigsten Bedrohungsszenarien beschrieben. Im letzten Teil des Buches werden Ihnen dann die unterschiedlichen Abwehrstrategien für die beschriebenen Bedrohungsszenarien erläutert.

Typografische Konventionen

Damit Sie bestimmte Elemente auf einen Blick erkennen und zuordnen können, werden diese im Text durch eine besondere Formatierung hervorgehoben. So werden beispielsweise Bezeichnungen für Programmelemente wie Register oder Schaltflächen immer *kursiv* geschrieben und wichtige Begriffe **fett** hervorgehoben.

- Kursivschrift** kennzeichnet alle vom Programm vorgegebenen Bezeichnungen für Schaltflächen, Dialogfenster, Symbolleisten etc., Menüs bzw. Menüpunkte (z. B. *Datei-Speichern*), Internetadressen und vom Benutzer angelegte Namen (z. B. Rechner-, Benutzernamen).
- Courier wird für Systembefehle sowie für Datei- und Verzeichnisnamen verwendet. In Syntaxangaben werden Parameter kursiv ausgezeichnet (z. B. `cd Verzeichnisname`). Eckige Klammern [] kennzeichnen optionale Elemente. Alternative Eingaben sind durch einen senkrechten Strich | getrennt. Benutzereingaben auf der Konsole werden **fett** hervorgehoben.

1

Was ist Sicherheit?

1.1 Grundforderungen an Sicherheit

Sicherheit und die in diesem Buch beschriebene **IT-Sicherheit** sind grundlegender Bestandteil der Unternehmenssicherheit. Sie umfasst alle Prozesse, Strategien und das Know-how eines Unternehmens, um es vor Eingriffen durch Dritte zu schützen.

Bei der IT-Sicherheit geht es grundsätzlich um:

- ✓ **Funktionssicherheit** (engl. safety) des Systems, welches als Hardware und/oder Software vorhanden ist. Dabei darf das System unter allen vorgegebenen Betriebsbedingungen keine Zustände annehmen, die unzulässig sind.
- ✓ **Datensicherheit** (engl. protection) definiert die Eigenschaft eines funktionssicheren Systems, die zu keinem unautorisierten Zugriff auf die Ressourcen des Systems und insbesondere auf die Daten führen. Dazu nutzt man Protokolle, die Vertraulichkeit, Integrität und Verfügbarkeit umsetzen.
- ✓ **Datenschutz** (engl. privacy) ist die Fähigkeit einer natürlichen Person, sein Persönlichkeitsrecht bezogen auf die eigenen Daten wahrzunehmen, um einen etwaigen Missbrauch durch Dritte zu unterbinden.

1.2 Sicherheitsziel Vertraulichkeit

Unter dem Sicherheitsziel der **Vertraulichkeit** (engl. confidentiality) wird verstanden, dass Informationen nur diejenigen erreichen, die diese Informationen auch besitzen dürfen. Bezogen auf Kommunikation in Netzwerken ist das Sicherheitsziel der Vertraulichkeit vergleichbar mit dem Briefgeheimnis. Wenn Sie eine E-Mail an einen bestimmten Empfänger absenden, erwarten Sie, dass nur der von Ihnen bestimmte Empfänger den Inhalt der E-Mail lesen kann.

Das Sicherheitsziel der Vertraulichkeit beschränkt sich nicht nur auf E-Mails. Jede auf einem Computersystem gespeicherte Information dient einem bestimmten Zweck, und in den meisten Fällen ist es nicht erforderlich oder nicht erwünscht, dass diese Informationen öffentlich zugänglich sind.

In der realen Welt sind Schutzmaßnahmen für Vertraulichkeit z. B. ein Briefumschlag, in den man seine nicht öffentliche Nachricht steckt, oder eine abgesperrte Tür, die nur den Personen Zugang zu einem Raum gewährt, die den passenden Schlüssel besitzen.

Um Vertraulichkeit zu gewährleisten, können verschiedene Maßnahmen eingesetzt werden: beispielsweise eine Verschlüsselung von Dateien oder Nachrichten zwischen den Kommunikationspartnern oder eine Zugangskontrolle, die nur bestimmten Personen einen Einblick in das geschützte Datenmaterial erlaubt.

1.3 Sicherheitsziel Integrität

Wenn mit Daten gearbeitet wird, muss ein sicheres IT-System gewährleisten können, dass die Daten **korrekt** sind (engl. integrity). Beispielsweise müssen Fehler bei der Übertragung von Daten verhindert oder wenigstens erkannt und ggf. korrigiert werden können. Es muss aber auch möglich sein, Daten und IT-Systeme gegen Manipulationen zu schützen.

Wird an die Möglichkeit, die Integrität der Daten zu gewährleisten oder bestätigen zu können, auch eine Information über den Urheber oder Verfasser der Daten gekoppelt, so entsteht eine **Authentizität** (engl. authenticity) der entsprechenden Daten – sozusagen eine **digitale Unterschrift**.

Authentifizierung stellt in gewisser Weise eine detailliertere Sicht von Integrität als Sicherheitsziel dar. In der aktuellen politischen Diskussion um digitale Signaturen wird eine weitere Stufe von Authentifizierung sichtbar:

Eine E-Mail, die eine Bestellung enthält, wird vor Gericht ohne weiteres keinen Bestand haben: Der Inhalt könnte beispielsweise manipuliert sein, oder es wurde sogar der Absender der E-Mail gefälscht, und der vermeintliche Auftraggeber weiß gar nichts von seiner Bestellung.

Selbst wenn hier Methoden zur Gewährleistung der Integrität des Inhalts (keine Manipulation mehr möglich) und zur Authentifikation (die Mail stammt wirklich vom genannten Absender) wahrgenommen wurden, reicht das im juristischen Sinne mitunter nicht aus, um eine gültige Willenserklärung zum Abschluss eines Kaufvertrages darzustellen. Es wäre immer noch relativ leicht möglich, einen Grund zu finden, warum diese E-Mail keine gültige Willenserklärung sein sollte.

Durch die eigene Unterschrift auf einem Stück Papier belegen Sie, dass Sie mit dem Inhalt des Textes einverstanden sind und seine Konsequenzen akzeptieren. Da Ihre Unterschrift durch das Papier direkt (und relativ schwer trennbar) mit dem unterschriebenen Text zusammengebracht wird, ist hier die **Verbindlichkeit** gewährleistet – aufgrund der Natur von Informationssystemen ist diese Untrennbarkeit von Inhalt und Unterschrift nicht ganz so einfach zu realisieren.

Als eine Forderung, die Authentifikation erweitert und der digitalen Signatur erst einen Sinn gibt, wird die **Verbindlichkeit** (engl. non-repudiation) einer digitalen Unterschrift definiert.

Ist in einem System die Verbindlichkeit für die Kommunikation sichergestellt, kann ein Teilnehmer nicht zu einem späteren Zeitpunkt behaupten, die Kommunikation habe nicht oder mit einem anderen Inhalt stattgefunden.

1.4 Sicherheitsziel Verfügbarkeit

Ein weiteres Hauptziel für die Sicherheit von Daten ist die Verfügbarkeit (engl. availability). Ein sicheres IT-System muss auch gewährleisten können, dass die Daten, die es verarbeitet, auch zugreifbar sind bzw. dass die Dienste, die angeboten werden, auch wirklich genutzt werden können.

Verfügbarkeit umfasst in der Regel logische Schutzmaßnahmen (zum Beispiel gegen versehentliches Löschen) genauso wie geeignete Maßnahmen, die einen Betrieb bei Störungen von Hard- und Software aufrechterhalten können. Auch äußere Einflüsse, wie zum Beispiel Stromausfälle oder gezielte Manipulationen von Saboteuren mit dem Ziel, die Dienste dieses Systems für berechtigte Nutzer zu blockieren, sind Probleme, mit denen sich ein Verfügbarkeitskonzept befasst.

Speziell für Einsatzgebiete, in denen eine Verfügbarkeit der Dienste rund um die Uhr gewährleistet sein muss, gibt es angepasste Hochverfügbarkeitslösungen, die einerseits durch spezielle Hardware und anderseits durch angepasste Algorithmen in der Software versuchen, eine möglichst hohe Ausfallsicherheit zu erreichen.

Beispiele

- ✓ Feuer-, Wasser- und EMP-feste Auslegung der Serverräume
- ✓ Redundante physikalische Server-Systeme (doppelte Netzteile, Controller, Netzwerkinterfaces, RAID, etc.)
- ✓ Clustering von Servern (active/active oder active/passive)
- ✓ Virtualisierung der Daten und deren Backup
- ✓ „Watchdog“: Hard- oder Software, die das Funktionieren eines Systems überwacht
- ✓ Redundante physikalische Topologien (Ring- bzw. Maschentopologie)
- ✓ Redundante Layer-2-Verbindungen zur Erhöhung der Bandbreite (Link Aggregation Control Protocol IEEE 802.3ad oder Port Aggregation Protocol)
- ✓ Redundante Layer-2-Verbindungen (Spanning Tree Protocol, Rapid Spanning Tree Protocol, Multiple Spanning Tree Protocol, Shortest Path Bridging, TRILL)
- ✓ Dynamische Routing-Protokolle bei vorhandenen physikalisch redundanten Wegen (z. B. Open Shortest Path First)
- ✓ Verfügbarkeitsprotokolle auf Layer 3 (z. B. Virtual Router Redundancy Protocol oder Gateway Load Balancing Protocol)
- ✓ Redundante Dienste (z. B. Primary Domain Controller und Backup Domain Controller)
- ✓ Verteilte Anwendungen

1.5 Rechtliche Aspekte

Gesetzliche Grundlagen der Informationssicherheit

Das deutsche und europäische Recht bietet eine Reihe von juristischen Möglichkeiten, um der Sicherheit im Telekommunikationsbereich Rechnung zu tragen. Das sind in Deutschland insbesondere:

- ✓ Strafgesetzbuch 15. Abschnitt – Verletzung des persönlichen Lebens- und Geheimbereichs (<http://dejure.org/gesetze/StGB/202a.html>)
- ✓ § 202a Ausspähen von Daten
- ✓ § 202b Abfangen von Daten
- ✓ § 202c Vorbereiten des Ausspähens und Abfangens von Daten
- ✓ § 206 Verletzung des Post- oder Fernmeldegeheimnisses
- ✓ Strafgesetzbuch 27. Abschnitt – Sachbeschädigung (<http://dejure.org/gesetze/StGB/303a.html>)
- ✓ § 303a Datenveränderung
- ✓ § 303b Computersabotage
- ✓ Telekommunikationsgesetz (TKG, <http://dejure.org/gesetze/TKG/88.html>)
- ✓ Teil 7 – Fernmeldegeheimnis, Datenschutz, Öffentliche Sicherheit (§§ 88 –115)
- ✓ Gesetz über das Bundesamt für Sicherheit in der Informationstechnik (BSI-Gesetz)
- ✓ Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme (IT-Sicherheitsgesetz)
- ✓ EU-Datenschutzgrundverordnung (EU-DSGVO)
- ✓ Besondere Vertragsbedingungen für die Beschaffung von DV-Leistungen (BVB)

IT-Sicherheitsgesetz

Das Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme (IT-Sicherheitsgesetz) wurde am 12. Juni 2015 vom Bundestag beschlossen, am 24. Juli 2015 im Bundesgesetzblatt verkündet (BGBl. I, Nr. 31, S. 1324) und trat am 25. Juli 2015 in Kraft.

Es regelt, dass Betreiber sogenannter **Kritischer Infrastruktur** (vgl. 1. b) ein Mindestniveau an IT-Sicherheit einhalten und dem Bundesamt für Sicherheit in der Informationstechnik (BSI) IT-Sicherheitsvorfälle melden müssen. Werden keine Maßnahmen organisatorischer und technischer Art zur Vermeidung von Störungen getroffen, droht ihnen ein Bußgeld. Gleichzeitig werden Hard- und Software-Hersteller zur Mitwirkung bei der Beseitigung von Sicherheitslücken verpflichtet.

Durch das IT-Sicherheitsgesetz werden mehrere bestehende Gesetze, darunter insbesondere das Gesetz über das Bundesamt für Sicherheit in der Informationstechnik (BSI-Gesetz), das Atomgesetz, das Energiewirtschaftsgesetz, das Telemediengesetz, das Telekommunikationsgesetz, geändert.

1. Bundesamt für Sicherheit in der Informationstechnik (Änderung im BSI-Gesetz)

a) Aufgaben des BSI

Der Zentralstelle für das Chiffrierwesen wurde 1986 neben dem Chiffrieren von Verschlusssachen des Bundes der zusätzliche Aufgabenbereich der **Computersicherheit** zugewiesen. 1989 wurde daraus die „Zentralstelle für die Sicherheit in der Informationstechnik“. Das BSI (Bundesamt für Sicherheit in der Informationstechnik) wurde 1990 mit dem BSI Errichtungsgesetz geschaffen.

Bislang war der Schutz der EDV-Anlagen der Bundesbehörden (Bundesministerien, Bundesämter) die Kernaufgabe des BSI. Die nicht zur Exekutive gehörenden Bundesorgane (Bundesrat und Bundestag, die neun Bundesgerichte) zählen aufgrund der Gewaltentrennung nicht zum Bund im Sinne des BSI-Gesetzes, (§ 2 Absatz 3 Satz 2 BSI-Gesetz). Schon seit dem BSI-Gesetz vom 14.08.2009 durfte das BSI die Öffentlichkeit oder die betroffenen Kreise in Behörden vor Sicherheitslücken in informationstechnischen Produkten und Diensten oder vor Schadprogrammen warnen (§ 7 Absatz 1 BSI-Gesetz) und im Verbund mit der Privatwirtschaft „Kommunikationsstrukturen zur Krisenfrüherkennung, Krisenreaktion und Krisenbewältigung sowie Koordinierung der Zusammenarbeit zum Schutz der kritischen Informationsinfrastrukturen“ aufbauen (§ 3 Absatz 1 Satz 2 Nr. 15 alte Fassung BSI-Gesetz). Mit dem BSI-Gesetz wurde der Begriff „Kritische Informationsinfrastrukturen“ in „Informationstechnik Kritischer Infrastrukturen“ geändert.

Mit dem IT-Sicherheitsgesetz erhielten der Schutz der Öffentlichkeit und der Kritischen Infrastrukturen innerhalb der Aufgaben des BSI eine ähnlich starke Stellung wie der Schutz der EDV-Anlagen des Bundes. Das BSI darf nun z. B. auf dem Markt angebotene informationstechnische Produkte und Systeme untersuchen (Absatz 1 des durch das IT-Sicherheitsgesetz neu eingefügten § 7a BSI-Gesetz). Nachdem es den Anbietern Gelegenheit zur Stellungnahme gegeben hat, darf das BSI seine Prüfergebnisse weitergeben und veröffentlichen, soweit dies zur Erfüllung seiner Aufgaben erforderlich ist, (§ 7a Absatz 2 BSI-Gesetz).

b) Begriff der Kritischen Infrastrukturen

Das BSI-Gesetz enthält in § 2 Absatz 10 eine Definition für Kritische Infrastrukturen. Bei Kritischen Infrastrukturen handelt es sich um „Einrichtungen, Anlagen oder Teile davon, die

1. den Sektoren Energie, Informationstechnik und Telekommunikation, Transport und Verkehr, Gesundheit, Wasser, Ernährung sowie Finanz- und Versicherungswesen angehören und
2. von hoher Bedeutung für das Funktionieren des Gemeinwesens sind, weil durch ihren Ausfall oder ihre Beeinträchtigung erhebliche Versorgungsengpässe oder Gefährdungen für die öffentliche Sicherheit eintreten würden.“

Welche Einrichtungen, Anlagen oder Teile davon im Einzelnen „von hoher Bedeutung“ sind, wird in die Hände des Bundesinnenministeriums gelegt. Dieses hat in einer Rechtsverordnung die kritischen Infrastrukturen zu benennen. Eine Zustimmung des Bundesrats zu der Verordnung ist nicht erforderlich; allerdings hat das BMI vor Erlass der Rechtsverordnung Vertreter der Wissenschaft, der betroffenen Betreiber und der betroffenen Wirtschaftsverbände anzuhören, (§ 10 Absatz 1 BSI-Gesetz). Hinsichtlich ihrer jeweiligen Fachbereiche ist die Verordnung im Einvernehmen mit anderen Bundesministerien zu erlassen, darunter die Ressorts für Finanzen, Verteidigung, Wirtschaft und Energie, Gesundheit, Verkehr und Digitale Infrastruktur sowie Umwelt und Reaktorsicherheit, (§ 10 Absatz 1 BSI-Gesetz).

c) Schutz der Kritischen Infrastrukturen

Vier neu ins BSI-Gesetz eingefügte Paragrafen dienen dem Schutz der Kritischen Infrastrukturen, die §§ 8a bis 8d. Sie enthalten Rechte und Pflichten sowohl des BSI als auch von Betreibern Kritischer Infrastrukturen.

Unternehmen, die in der Rechtsverordnung des Bundesinnenministeriums als Betreiber Kritischer Infrastrukturen bezeichnet werden, erhalten zwei Jahre Zeit, um „organisatorische und technische Vorkehrungen zur Vermeidung von Störungen“ zu treffen, (§ 8a Absatz 1 Satz 1 BSI-Gesetz).

2. Änderung Telemediengesetz und Telekommunikationsgesetz

Diensteanbieter nach dem Telemediengesetz werden verpflichtet, im Rahmen der wirtschaftlichen Zumutbarkeit und technischen Machbarkeit, unerlaubte Zugriffe auf die für die Telemedien genutzten Einrichtungen sowie Verletzungen persönlicher Daten zu verhindern, (Art. 4 IT-Sicherheitsgesetz = neuer § 13 Absatz 7 Telemediengesetz).

Diensteanbieter nach dem Telekommunikationsgesetz erhalten die Erlaubnis, Bestands- und Verkehrsdaten der Teilnehmer und Nutzer zu erheben und zu verwenden, um Störungen oder Fehler der Telekommunikationsanlagen zu erkennen, einzugrenzen oder zu beseitigen, (Art. 5 Nr. 2 IT-Sicherheitsgesetz = Neufassung § 100 Absatz 1 Satz 1 Telekommunikationsgesetz). Es wurde eine Mitteilungspflicht des Netzbetreibers oder Erbringens öffentlich zugänglicher Telekommunikationsdienste eingeführt, wenn Störungen zu beträchtlichen Sicherheitsverlusten führen oder führen können, (Art. 5 Nr. 3c IT-Sicherheitsgesetz = neuer § 109 Absatz 5 Telekommunikationsgesetz). Die Bundesnetzagentur darf die erhaltenen Informationen über Sicherheitsmängel an das BSI weitergeben, (Art. 5 Nr. 3e IT-Sicherheitsgesetz = neuer § 109 Absatz 8 Telekommunikationsgesetz).

3. Der Europarechtliche Rahmen

Die Richtlinie 2008/114/EG des Rates verpflichtet die Mitgliedstaaten, zum einen kritische Infrastrukturen im Energie- und Verkehrssektor zu ermitteln und auszuweisen, zum anderen zu bewerten, inwieweit es notwendig ist, ihren Schutz zu verbessern. Die Richtlinie verpflichtet die EU-Staaten weiter, dafür zu sorgen, dass Betreiber kritischer Infrastrukturen von grenzüberschreitender Bedeutung (EKI) Risikoanalysen durchführen und Sicherheitspläne aufstellen. Für Betreiber von Anlagen sieht die Richtlinie jedoch keine Meldepflichten bei schwerwiegenden Sicherheitsverletzungen vor.

Die Richtlinie enthält folgende Definition: Kritische Infrastrukturen sind eine „Anlage, ein System oder ein Teil davon, die von wesentlicher Bedeutung für die Aufrechterhaltung wichtiger gesellschaftlicher Funktionen, der Gesundheit, der Sicherheit und des wirtschaftlichen oder sozialen Wohlergehens der Bevölkerung sind und deren Störung oder Zerstörung erhebliche Auswirkungen auf einen Mitgliedstaat hätte, da diese Funktionen nicht aufrechterhalten werden könnten“. Europäische kritische Infrastruktur (EKI) ist demnach eine Störung oder Zerstörung, die erhebliche Auswirkungen in mindestens zwei anderen EU-Staaten hätte.

4. Kritik am IT-Sicherheitsgesetz

Verfassungsrechtlich bedenklich hinsichtlich der Normenklarheit ist, dass eine auch nur einigermaßen konkrete Bestimmung des Begriffs der Kritischen Infrastruktur und eine Bestimmbarkeit der betroffenen Betreiber im Gesetz fehlt und auf den Verordnungsweg ausgelagert wird. Unklar sind auch die datenschutzrechtlichen Weitergabepflichten und -befugnisse. Die Datenschutzbeauftragten des Bundes und der Länder sind in die Meldewege nicht mit einbezogen.

Das BSI erhält zwar einen erweiterten Aufgabenbereich; mehr Selbstständigkeit gegenüber dem Bundesinnenministerium erhält es aber nicht. Es mangelt an präventiven Ansätzen, die proaktiv zur Verbesserung der IT-Sicherheit beitragen. Vorgesehen sind Meldepflichten und Bußgelder. Es fehlen aber Anreizsysteme wie z. B. die Zertifizierung von Verfahren, ebenso wie gesetzliche Kriterien, die zu einer Qualitätssteigerung von IT-Sicherheitskonzepten und Sicherheitsprüfungen wie z. B. Penetrationstests beitragen. Die Meldepflichten für Sicherheitsvorfälle bestehen erst, wenn es schon zu spät ist, nämlich „bei erheblichen Störungen“, statt bereits zu einem Zeitpunkt, in dem noch kein Schaden eingetreten ist. Die technischen Schutzstandards begnügen sich zu sehr mit einem angenommenen Stand der Technik (§ 8a Absatz 1 Satz 2 BSI-Gesetz), statt auf einem Weiterdenken in Form von Risikoanalysen.

EU-Datenschutzgrundverordnung (EU-DSGVO)

Nach mehrjähriger Debatte hat sich der EU-Trilog (Europäischer Rat, Europäisches Parlament, Europäische Kommission) im Dezember 2015 auf einen endgültigen Inhalt einer neuen EU-Datenschutzverordnung geeinigt. Sie soll die bisher geltende EU-Datenschutzrichtlinie (Richtlinie 95/94/EG) ersetzen und in den nächsten zwei Jahren vollständig in den EU-Mitgliedstaaten umgesetzt werden.

Ziel der Verordnung ist die Vereinheitlichung und die Vereinfachung der Datenschutzrichtlinien innerhalb der EU-Mitgliedstaaten. Mit dieser Verordnung werden die **Nutzerrechte** gegenüber den Verwertern von Nutzerdaten (z. B. Facebook oder Google) nachhaltig gestärkt. Der Nutzer hat das Recht, ausführlich zu erfahren, welche Daten und zu welchem Zweck über ihn gespeichert, verarbeitet und weitergegeben werden. **Personenbezogene Informationen** gehören nun dem Nutzer und nicht den mit der Datenverarbeitung befassten Internetanbietern. Auch wird das vollständige Löschen von personenbezogenen Daten im Internet erleichtert und die rechtswirksame Einwilligung für die Verarbeitung von persönlichen Daten auf ein Mindestalter von 16 Jahren angehoben.

Diese Regelung gilt nicht nur für europäische, sondern auch für nicht in Europa ansässige Unternehmen. Bei einer Verletzung des Datenschutzes sind laut Verordnung Bußgelder bis zu 4 % des Jahresumsatzes möglich.

Die EU-Datenschutzgrundverordnung ist in folgende Abschnitte unterteilt und in seiner übersetzten Fassung unter <https://dsgvo-gesetz.de/> verfügbar:

- ✓ Kapitel 1 Allgemeine Bestimmungen
- ✓ Kapitel 2 Grundsätze
- ✓ Kapitel 3 Rechte der betroffenen Person
- ✓ Kapitel 4 Für die Verarbeitung Verantwortlicher und Auftragsverarbeiter
- ✓ Kapitel 5 Übermittlung personenbezogener Daten in Drittländer oder an internationale Organisationen
- ✓ Kapitel 6 Unabhängige Aufsichtsbehörden
- ✓ Kapitel 7 Zusammenarbeit und Kohärenz
- ✓ Kapitel 8 Rechtsbehelfe, Haftung und Sanktionen
- ✓ Kapitel 9 Vorschriften für besondere Datenverarbeitungssituationen
- ✓ Kapitel 10 Delegierte Rechtsakte und Durchführungsrechtsakte
- ✓ Kapitel 11 Schlussbestimmungen

Das europäische Parlament hat am 21.04.2016 die neue EU-Datenschutzgrundverordnung beschlossen, welche 2018 in nationales Recht umgesetzt wurde. Damit wird die als überholt geltende Richtlinie von 1995 ersetzt.

Besondere Vertragsbedingungen für die Beschaffung von DV-Leistungen (BVB)

Die Vertragsbedingungen für die Beschaffung von DV-Leistungen dienen der öffentlichen Hand für die Planung und Beschaffung einer funktions- und datensicheren Hard- und Software, entsprechend der Forderungen der rechtlichen Rahmenbedingungen. Zusätzlich gelten ergänzende Vertragsbedingungen für die Beschaffung von Informationstechnik (EVB-IT Version 2 vom 17.03.2016), welche die besonderen Vertragsbedingungen für die Beschaffung von DV-Anlagen und Geräten (BVB) teilweise ablösen bzw. ergänzen.

Die Anwendung der EVB-IT und der BVB ist für Bundesbehörden gemäß Verwaltungsvorschrift zu § 55 BHO festgeschrieben. Auch die Länder richten sich größtenteils nach diesen Vorschriften.

Die aktuellen EVB-IT- und noch gültigen BVB-Dokumente können Sie unter http://www.cio.bund.de/Web/DE/IT-Beschaffung/EVB-IT-und-BVB/Aktuelle_EVB-IT/aktuelle_evb_it_node.html;jsessionid=9E691728ADCBF026C7C35365E927456F.2_cid334 einsehen.

Das Vertrauensdienstegesetz (<http://www.gesetze-im-internet.de/vdg/>) regelt die wirksame Durchführung der Vorschriften über Vertrauensdienste in der Verordnung (EU) Nr. 910/2014 des Europäischen Parlaments und des Rates vom 23. Juli 2014 über elektronische Identifizierung und Vertrauensdienste für elektronische Transaktionen im Binnenmarkt. Das Gesetz wird von der Bundesnetzagentur und dem Bundesamt für Sicherheit in der Informationstechnik beaufsichtigt.

Das „Nationale Cyber-Abwehrzentrum“ (NCAZ) hat die Bundesrepublik mit Wirkung vom 01.04.2011 zur Abwehr von Angriffen auf die IT-Infrastruktur der Länder, des Bundes und der Wirtschaft geschaffen. Das NCAZ soll Informationen sammeln, Defizite bei IT-Lösungen aufzeigen, Angriffsanalysen und Hackerprofile erstellen und auf dieser Grundlage Empfehlungen für den Cyber-Sicherheitsrat bereitstellen. Der „Nationale Cyber-Sicherheitsrat“ (NCS) wird aufgrund der Empfehlungen die erforderlichen Schutzmaßnahmen und die notwendige Netzpolitik koordinieren. Unter der Aufsicht des Bundesamtes für Sicherheit in der Informationstechnik (BSI, http://www.gesetze-im-internet.de/bundesrecht/bsig_2009/gesamt.pdf) sind Mitarbeiter des Bundesamtes für Verfassungsschutz (BfV) und des Bundesamtes für Bevölkerungsschutz und Katastrophenhilfe (BKK) im NCS tätig.

2

Risikolage für Unternehmen

2.1 Warum ist das Internet nicht „sicher“?

Die Entstehung des Internets

Das Internet und die dazugehörigen Protokolle wurden in den 60er-Jahren, zur Zeit des Kalten Krieges zwischen den USA und der Sowjetunion, entwickelt.

Die Computersysteme der damaligen Zeit waren zentral gesteuert. Der Ausfall eines zentralen Knotenpunktes (z. B. durch einen Angriff) hätte das gesamte angeschlossene Netz außer Betrieb gesetzt. Paul Baran (Rand Corporation), wurde mit der Konzeption eines ausfallsicheren Netzwerkes beauftragt.

Barans revolutionäres Konzept sah ein Netzwerk vor, bei dem prinzipiell jeder Rechner mit jedem anderen kommunizieren konnte – ein vollständig **dezentrales Netz**. Die Datenpakete sollten sich „selbstständig“ einen Weg von der Quelle zum Ziel suchen und, wenn notwendig, einen anderen Weg einschlagen, falls ein bestimmter Netzknoten ausgefallen war.

Der Vorschlag wurde vom Pentagon ignoriert. Kurz darauf wurde die Projektgruppe Advanced Research Project Agency mit der Entwicklung eines dezentralen Netzes beauftragt. Ende der 60er-Jahre wurde das nach ihr benannte ARPANET in Betrieb genommen.

In den 70er-Jahren wurde das Übertragungsprotokoll **TCP** entwickelt. TCP war dafür konzipiert, Datenströme in Pakete aufzuteilen und diese über das Netzwerk zu versenden. Auf der Empfängerseite konnte TCP die Datenpakete wieder korrekt zu einem Datenstrom zusammensetzen. Auch E-Mail und andere Dienste wurden nach und nach entwickelt.

Bis Ende der 80er-Jahre war das ARPANET und spätere Internet in der Hand der amerikanischen Regierung und vernetzte Militär- und Forschungseinrichtungen.

Anfang der 90er-Jahre begann die amerikanische Regierung sich aus dem Internet zurückzuziehen und es für kommerzielle Firmen zu öffnen.

Der Internet-Boom

Der Boom des Internets begann mit der Entwicklung von **HTTP** (Hypertext Transfer Protocol) und dem ersten **Internet-Browser**, der es auch technisch nicht versierten Benutzern erlaubte, über eine grafische Bedienoberfläche vernetzte Inhalte abzurufen.

Seit mehreren Jahrzehnten ist das Internet ein öffentliches Netz, das von Privatpersonen, Firmen und Behörden weltweit genutzt werden kann. Für die Frage nach der Sicherheit des Internets sind folgende Fakten wichtig:

Die Internetprotokolle wurden im Hinblick darauf entwickelt, eine Datenübertragung auch nach einem Ausfall eines oder mehrerer Netzknoten zu gewährleisten. Die automatische Wegfindung im Netzwerk (auch **Routing** genannt) war hier das Hauptziel.

Für die Übertragung wurden hierfür **TCP** (Transmission Control Protocol) und **UDP** (**User Datagram Protocol**) standardisiert. TCP realisiert eine **verbindungsorientierte** Übertragung, d. h., es wird gewährleistet, dass nicht empfangene Daten-Segmente nochmals versendet werden. UDP arbeitet dagegen **verbindungslos** – d. h., es findet keine Quittierung der empfangenen Segmente statt – und wird vorwiegend für Realtime-Anwendungen genutzt.

Vernetzung: jeder mit jedem

Da die Entwickler nicht absehen konnten, dass dieses Netzwerk später nicht nur die Rechenanlagen des amerikanischen Militärs, sondern Computer weltweit vernetzen würde, wurde auch kein Mechanismus eingebaut, der die Korrektheit der Angaben in den Protokollen sicherstellt. Somit ist es möglich, Datenpakete mit gefälschten Daten oder gezielt manipulierte Pakete in das Netz zu senden.

Da die Internetprotokolle mit dem Ziel entwickelt wurden, dass alle an das Internet angeschlossener Computer miteinander kommunizieren können, haben auch Kriminelle über ihre Rechner Zugriff auf das Internet.

2.2 Schadensmöglichkeiten

Was passieren kann

Wenn ein oder mehrere Rechner eines Unternehmens an das Internet angeschlossen sind, gibt es zahlreiche Möglichkeiten, wie der Einsatz der IT vom Sollzustand abweichen kann. Aus dem Internet kann bösartige Software wie Viren in das Unternehmensnetzwerk gelangen und dort Datenverluste sowie Ausfallzeiten verursachen. Spam und E-Mail-Viren können E-Mail-Server überlasten und Netzwerk-Bandbreiten belegen.

Vertrauliche und geheime Informationen könnten unkontrolliert das Unternehmen verlassen, wenn Hacker in die Netzwerke eindringen, um Informationen auszuspähen und zu stehlen, oder wenn Mitarbeiter unvorsichtig Dokumente versenden. Erlangen unautorisierte Personen von intern als auch von extern Zugriff auf Dateien und Systeme, können Daten manipuliert und Netzwerke modifiziert werden.

Der entstandene Schaden lässt sich meist nur schwer beziffern. Bei einem Serverausfall oder einem Datenverlust können die zur Behebung des Schadens angefallenen Arbeitsstunden und der Ausfall an produktiver Arbeitszeit ermittelt werden, aber Imageschäden lassen sich nur äußerst schwer beziffern.

Eine Onlinebank, deren Webseite von einem Hacker verändert wurde, oder eine Firma, deren Kundendaten inklusive Kreditkartendaten veröffentlicht werden, hat das Vertrauen ihrer Kunden verloren. Gelangt der Schriftverkehr eines Versicherungsunternehmens in unbefugte Hände, ist dies eine unerwünschte Situation, die Forderung nach Vertraulichkeit wurde verletzt – der finanzielle Schaden ist jedoch nicht genau errechenbar.

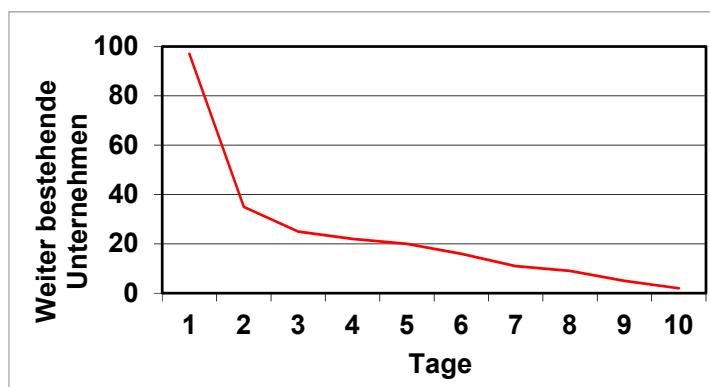
Folgen einer Datenpanne

Problematisch bei Datenpannen ist die Wiederherstellung des Betriebszustandes deswegen, weil verlorene Daten rekonstruiert werden müssen und die inzwischen angefallene Arbeit auch erledigt werden muss. Untersuchungen zeigen, dass die benötigte Zeit zur Rückkehr in den Normalzustand nach einer geplanten Betriebsunterbrechung die Ausfallzeit circa um den Faktor fünf übersteigt. Nach einem ungeplanten Zwischenfall oder einer Computerkatastrophe kann dieser Faktor zehn betragen.

2.3 Wie abhängig sind Firmen vom IT-Einsatz?

Eine Untersuchung der Universität von Minnesota schätzte die durchschnittlichen Zeiten für das Weiterbestehen von Unternehmen nach einer Katastrophe im Rechenzentrum auf folgende Werte:

- ✓ Banken 2 Tage
- ✓ Handelsunternehmen 3 Tage
- ✓ Industrie 5 Tage
- ✓ Versicherungen 6 Tage



Umfrage: Wie viele Tage könnten Sie den Ausfall Ihrer IT überleben?

Weiterhin zeigte eine Studie über amerikanische Unternehmen, die von einer Katastrophe betroffen waren, dass 25 % kurz nach der Katastrophe und 40 % innerhalb von zwei Jahren Konkurs anmelden mussten. Nach 5 Jahren waren weniger als 7 % der betroffenen Firmen noch auf dem Markt tätig.

Diese Studie definierte eine Computerkatastrophe als Totalausfall des IT-Systems in einem Unternehmen. Extremfälle dieser Art sind relativ selten. Dennoch zeigen diese Zahlen, dass bei Problemen im IT-Bereich die Achillesferse einer Firma getroffen wird – zu viele Daten von der einfachen E-Mail bis zur Auftragsverarbeitung und Produktionsplanung und Steuerung laufen über Computersysteme.

Um eine Computerkatastrophe zu verhindern, aber auch um kleinere Ausfälle zu vermeiden, sollte sich nicht nur die IT-Abteilung eines Unternehmens Gedanken machen, sondern auch das Management. Es gilt, schützenswerte Elemente der Firma zu identifizieren, deren Risiko zu analysieren und gegen denkbare Bedrohungen geeignete Maßnahmen zu ergreifen und deren Wirksamkeit regelmäßig zu prüfen.

3

Angriffsvorbereitung

3.1 Hacker und Cracker

Wer Firmennetzwerke angreift

Hacking ist die Bezeichnung für die Verfahren, in Applikationen und Netzwerke in unberechtigter Weise einzudringen. Der Begriff ist heute negativ besetzt und bezieht sich im allgemeinen Sprachgebrauch auf illegale Aktivitäten in IT-Netzwerken – vornehmlich das unbefugte Ein- dringen in fremde Systeme und Umgehen von Sicherheitsvorkehrungen.

Die Sicherheits-Gemeinde, vornehmlich auch die Hacker, nehmen eine genauere Unterscheidung dieses Begriffs vor. Ursprünglich wurde ein Hacker als Person definiert, die mit hohem technischen Wissen und großer Neugierde die Arbeitsweise von Systemen und Netzwerken untersucht – und dadurch zwangsläufig auch auf Sicherheitslücken stößt. Der idealtypische Hacker würde, wenn er eine Sicherheitslücke in einem System entdeckt, den Eigentümer des Systems umgehend über diese Lücke informieren.

Als Gegenstück zum Hacker, der einer Art Ehrenkodex folgt, gilt der **Cracker**, dessen Ziele eindeutig krimineller Natur sind: Lücken in Netzwerken werden explizit gesucht, um diese zum Diebstahl von Informationen, zur Diffamierung von unliebsamen Firmen/Personen oder aus finanziellen Interessen ausnutzen zu können.

Black-Hats, White-Hats und Grey-Hats

Eine Konvention in alten Westernfilmen war es, dass die Helden immer weiße und die Schurken immer schwarze Hüte trugen. In Anlehnung daran entstanden die Bezeichnungen für „gute“ und „böse“ Hacker.

- ✓ **White-Hats** sind Ethikhacker mit fest definierten, meist gesetzeskonformen Regeln, die das Ziel haben, Schwachstellen in Netzwerken (z. B. durch Penetrationstests oder durch erkannte Programmierfehler) aufzudecken und zu dokumentieren.
- ✓ **Black-Hats** nutzen ihr Wissen für kriminelle Handlungen, um z. B. Systeme für ihre Zwecke zu manipulieren oder Informationen zu entwenden, um finanzielle Vorteile zu erhalten. Staatliche und halbstaatliche Institutionen rekrutieren mitunter diese Cracker für wirtschaftliche, politische oder militärische Zwecke.
- ✓ **Grey-Hats** handeln nicht immer im Sinne des gesetzlichen Rahmens und der Hackerethik, z. B. legen sie Sicherheitsmängel durch illegale Mittel offen.

Je nachdem, um welchen Typ von Hacker es sich handelt, wird dieser verschiedene Strategien benutzen, um einen Angriff durchzuführen.

Während White-Hats eine Lücke in einem Netz finden, das sie gerade untersuchen, wählen Black-Hats ihr Ziel mit Bedacht. Black-Hats, die finanzielle Interessen verfolgen oder ihr Prestige unter ihresgleichen steigern wollen, wählen erst das Ziel aus und schöpfen dann sämtliche Möglichkeiten für einen Angriff aus.

Innerhalb der IT-Sicherheitsbranche werden White-Hats unter dem Berufsbild „Bedrohungsjäger“ bzw. „Threat Hunter“ geführt. Sie versetzen sich hierbei in die Lage und Denkweise eines Angreifers, um diesem in der Analyse von Schwachstellen voraus zu sein.

3.2 „Staatliche“ Hacker

Nicht nur regierungsferne Personen versuchen, Schwachstellen in Systemen aufzuzeigen, auch der Staat hat ein Interesse, an Informationen zu gelangen. Ein herausragendes Beispiel dafür waren die Enthüllungen des ehemaligen NSA-Mitarbeiters Edward Snowden.

Staatliche Institutionen bedienen sich hierbei oft des Argumentes der Terrorabwehr, des Staatschutzes oder der Sicherung von wirtschaftlichen/politischen Interessen, um ihre Tätigkeiten zu legitimieren. Dabei nutzen sie Ressourcen aus der Hackerszene, rekrutierte Forschungseinrichtungen und Firmen aus dem Soft- und Hardwarebereich.

Die hierzu notwendigen Aufwendungen im Bereich der Hardwareinfrastruktur sollen am Beispiel der TOP 500 verdeutlicht werden. Jährlich wird das Ranking der wissenschaftlichen Zwecken dienenden Supercomputer (s. a. <https://www.top500.org/lists/2018/11/>) veröffentlicht. Mit Stand 11/2018 wies das Ranking der Top 7 eine Gesamtleistungsaufnahme von 62,68 Megawatt auf.

Das Ende 2013 fertiggestellte Rechenzentrum der NSA (National Security Agency) in Bluffdale (US-Bundesstaat Utah) mit seinen Cray-Supercomputern benötigt geschätzte 65 Megawatt und dient nicht wissenschaftlichen Zwecken. Die Leistungsaufnahme gibt Ihnen einen Anhaltspunkt, welche Anstrengungen für das Sammeln, Auswerten, Verknüpfen, Bewerten und Archivieren von Informationen aufgewendet werden, um politische, wirtschaftliche und militärische Vorteile zu erlangen.

Die dabei benutzten Applikationen sind mit hohem finanziellen und technischen Aufwand erstellt worden. Exemplarisch soll dies an Projekten der NSA dargestellt werden. Dabei bedienen sich andere Geheimdienste ähnlicher Methoden bzw. kooperieren mit der NSA. Einen sehr intensiven Austausch von Spionagedaten pflegt die Partnerschaft „Five Eyes“. Dies ist ein Zusammenschluss der Geheimdienste der USA, Großbritanniens, Kanadas, Australiens und Neuseelands.

Codename Bullrun

Im Rahmen von Bullrun werden Security-Lösungen beeinflusst, um Verschlüsselungssysteme zu umgehen. Dazu werden z. B. durch das Platzieren von NSA-Mitarbeitern in Unternehmen oder durch interne Unterstützung der Firmen Hintertüren (Backdoors) in die Sicherheitslösungen integriert. Sicherheitsbugs werden verspätet gemeldet oder es wird zu spät auf sie reagiert. Dadurch können Sicherheitsmängel gezielt für Angriffe genutzt werden.

Codename PRISM

PRISM dient der umfassenden Überwachung und Auswertung der kompletten Kommunikation von Nutzer-Accounts. Dies geschieht teilweise in Echtzeit, z. B. wenn sich ein Nutzer bei einem Dienst anmeldet. Die größten Internetprovider der USA liefern auf Anforderung der Geheimdienste hierzu die Daten bzw. haben Schnittstellen in ihr System integriert. Diese Schnittstellen ermöglichen z. B. der NSA den Zugriff auf alle Informationen der Diensteanbieter. Der britische GCHQ (Government Communications Headquarters) nutzt hierbei eine noch weiterreichende Lösung mit dem Codename **Tempora**.

Codename XKeyscore

XKeyscore umfasst die Analyse und Auswertung von Informationen aus verschiedenen Datenbank-Quellen (z. B. E-Mail-Accounts, Facebook-Kontakte, Webseiten-Historie, Verbindungs-nachweise, Kreditkarteninformationen, Smartphoneden etc.). Daraus werden Metadaten (Eigenschaften eines Objektes innerhalb einer Datenbank) gewonnen. Die Verknüpfung der Metadaten erfolgt über einen Suchalgorithmus (ähnlich einer Google-Suchfunktion). Diese Software wird auch in Europa, u. a. vom BND, genutzt. Das Ergebnis wird verwendet, um z. B. Bewegungsprofile, Verhaltensmuster, Trendanalysen oder Anomalien in den zu überwachenden Zielgruppen zu erkennen.

Aus der Analyse der Metadaten kann man z. B. folgende Erkenntnisse gewinnen:

- ✓ Welche Person aus der Firma X hat ab dem Zeitpunkt Y häufig verschiedene Kontakte in der Region Z gehabt?
- ✓ Wie oft wurden Mails mit Dateianhängen zur Firma/Region geschickt und dabei Verschlüsselungsalgorithmen genutzt?
- ✓ Kommunierte eine Person zum Zeitpunkt X mit verdächtiger Firma/Person Y und nutzte dabei ein Smartphone?
- ✓ Welches Verhältnis (Freund/Verwandter) hat Person X zur Gruppe Y und wann hatte sie den letzten Kontakt zu dieser Gruppe?
- ✓ Welche Produkte hat der Personenkreis X im Zeitraum Y mit der Kreditkarte Z erworben und in welcher Höhe?

Metadaten werden manchmal auch „offiziell“ verkauft. Anfang November 2013 wurde bekannt, dass der amerikanische Geheimdienst CIA (Central Intelligence Agency) die Datenbanken des größten US-Netzbetreibers nutzt. Der Provider AT&T hatte freiwillig einer Kooperation zugesagt und erhielt im Gegenzug dafür 10 Millionen US-\$ pro Jahr.

Die vorgestellten Projekte dienen dem Vorteil eines Staates in wirtschaftlicher, politischer und militärischer Hinsicht. Eine Verletzung der Persönlichkeitsrechte wird dabei billigend in Kauf genommen.

3.3 Elektronische Kriegsführung

Schon vor dem Angriff auf die Infrastruktur des Staates Estland (2007), der Attacke auf staatliche Webseiten in Georgien (2008) und dem Einsatz des Stuxnet-Wurms (2010) auf die iranische Atomindustrie rückte das Thema Elektronische Kriegsführung in den Vordergrund. Militärisch ist dies deshalb hochinteressant, weil der Initiator eines Angriffs sehr schwer identifizierbar ist und er demzufolge auch kaum einen sofortigen Gegenschlag befürchten muss.

Auch die Bundeswehr widmet sich seit Jahren dieser Thematik. In dem vertraulichen Strategiepapier „Strategische Leitlinie Cyber-Verteidigung“ hatte die Verteidigungsministerin Ursula von der Leyen im April 2015 die geänderten Richtlinien für die Bundeswehr aufgezeigt. Darin wurde ausgeführt, dass neben den klassischen Kriegsschauplätzen Land, Luft, See und Weltraum das Internet und andere Kommunikationsplattformen als neuer Operationsraum der Bundeswehr definiert ist. Um der Leitlinie gerecht zu werden, wurde ein Projekt „Digitale Kräfte“ unter Federführung der Staatssekretärin Karin Suder initiiert. Damit wurden einerseits die bisherigen Kräfte innerhalb der Bundeswehr konsolidiert und andererseits mehr als 1400 IT-Experten für die Bundeswehr rekrutiert. Mit der Umsetzung kann die Bundeswehr neben dem Abwehren von Cyber-Angriffen auch aktiv fremde Informations- und Telekommunikationsanlagen in deren Funktion beeinträchtigen.

3.4 Netzwerkscans

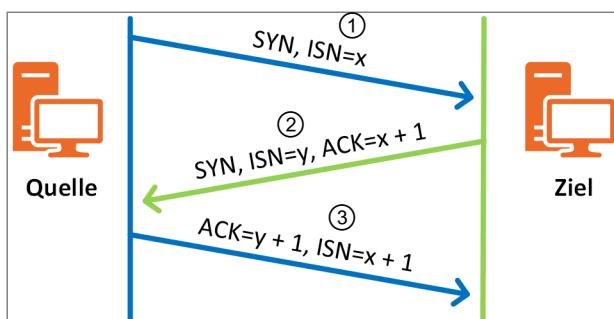
Warum Netzwerkscans benötigt werden

Bevor der eigentliche Angriff auf ein System oder Netzwerk stattfinden kann, benötigt der Hacker einen möglichst genauen Überblick über die installierten Dienste oder die Struktur des Netzwerkes. Die nachfolgenden Punkte stellen grundlegende Angriffsszenarien vor.

Portscanning

Eine Möglichkeit, etwas über die installierte Software oder verwendete Betriebssysteme in Erfahrung zu bringen, ist das Portscanning.

Um in IPv4- und IPv6-basierten Netzen eine logische Verbindung herzustellen, kommen UDP und TCP zum Einsatz. TCP stellt mit einem **3-Way Handshake** sicher, dass alle an der Datenübertragung beteiligten Komponenten und die Kommunikationspartner über das Vorhandensein einer Verbindung und deren Übertragungsparameter informiert sind.



TCP/IP – 3-Way Handshake

Während eines normalen Verbindungsaufbaus wird der anrufende Rechner dem Zielrechner seinen Verbindungswunsch zu der Ziel-IP-Adresse und dem ausgewählten Server-Port mitteilen, eine Synchronisation der Verbindung initiieren und die Sequenznummer nennen, ab der mit der Übertragung begonnen wird. Hierzu wird ein TCP-Segment mit entsprechend gesetzten Flags an den Zielrechner gesendet ①: SYN = Synchronize, ISN = Initial Sequence Number (Seq).

Ist auf dem Zielrechner der entsprechende Dienst für die Portanfrage vorhanden, so schlägt dieser eine Synchronisation und eine Sequenznummer vor und quittiert die empfangene Sequenznummer mit ACK = Acknowledgment ②. Der ACK-Wert ist die empfangene ISN + 1.

Um den Verbindungsaufbau abzuschließen, sendet der anrufende Rechner eine Bestätigung (ACK) der vom Zielrechner genutzten Sequenznummer, inkrementiert um den Wert 1 ③.

Source	Destination	Protocol	Info
192.168.1.34	192.168.1.37	TCP	idcp(2326) → ftp(21) [SYN] Seq=0 Win=24684 Len=0 MSS=1452 SACK_PERM=1
192.168.1.37	192.168.1.34	TCP	ftp(21) → idcp(2326) [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 SACK_PERM=1
192.168.1.34	192.168.1.37	TCP	idcp(2326) → ftp(21) [ACK] Seq=1 Ack=1 Win=24684 Len=0
192.168.1.37	192.168.1.34	FTP	Response: 220-FileZilla Server version 0.9.37 beta

Tracefile eines 3-Way Handshakes

Mit dem 3-Way Handshake wird erreicht, dass beide Kommunikationspartner sich über die Parameter der Verbindung einig sind und sich diese jeweils gegenseitig bestätigt haben. Ein Angreifer kann mittels dieser Methode Informationen über ein Zielsystem gewinnen. Erhält ein Hacker auf seinen ersten SYN ein ACK, weiß er, dass unter der ausgewählten IP-Adresse ein Rechner zu erreichen ist und ein Dienst auf den angewählten Port reagiert.

Einige Serverprogramme identifizieren ihre Dienste mit einem sogenannten Banner (Begrüßungstext) oder benutzen typische Portnummern, sodass beim Durchsuchen sämtlicher Ports eines Rechners schnell eine Liste der dort aktiven Netzwerkdienste erstellt werden kann.

Für Hacker hat diese Vorgehensweise jedoch einen gravierenden Nachteil: Da eine gültige Verbindung zwischen dem für den Scan benutzten Rechner und dem Dienst auf dem Zielsystem hergestellt wird, erscheint diese Verbindung in den Protokolldateien des Zielsystems – ein potenzieller Angreifer hinterlässt also Spuren.

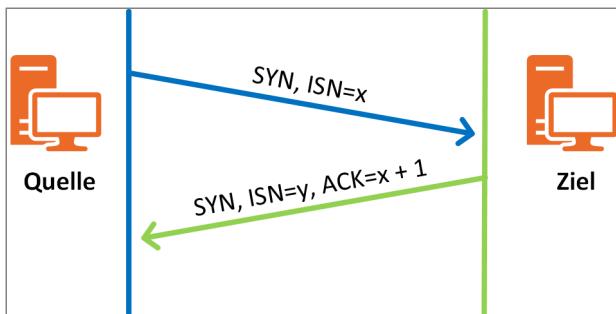
Aus diesem Grund kommen bei der Informationssammlung durch Hacker oft Tools zum Einsatz, die nicht konform zu TCP sind, aber deren Arbeitsweise dazu dient, ebenfalls Informationen über das Zielsystem zu erhalten.

Stealth Scans

Mit dem Oberbegriff „Stealth Scan“ werden Aktionen bezeichnet, die im Netzwerk, ähnlich wie ein Handshake, Pakete zu einem Zielsystem senden. Die Antworten (oder das Fehlen der Antworten) werden ausgewertet, um Informationen zu gewinnen. Der Begriff „Stealth“ (engl. heimlich) ist möglicherweise missverständlich, da diese Scanmethoden zwar darauf abzielen, einer normalen Protokollierung zu entgehen, mit geeigneten Sicherheitsmaßnahmen jedoch kann auch ein Stealth Scan nicht verborgen bleiben.

Half-Open Scan

Beim Half-Open Scan führt der Angreifer nur den ersten Teil eines 3-Way Handshakes aus. Das Zielsystem wird entweder den Verbindungsaufbau ablehnen oder das Öffnen einer Verbindung mit einem entsprechenden TCP-Segment bestätigen.



Half-Open Scan

In obigem Beispiel ist der Zielpo rt auf dem gescannten Rechner offen, deswegen erhält der linke Rechner eine positive Antwort mit der Bestätigung der zuerst gesendeten ISN und dem Vorschlag für eine ISN für den Versand von Daten vom Zielrechner zum Angreifer.

An dieser Stelle besitzt der Angreifer schon sämtliche Informationen, die er besitzen möchte: Nach seinem ersten Segment erhält er vom Zielrechner entweder eine Ablehnung (RST-Flag statt SYN-Flag) oder eine Antwort mit dem 2. Teil des 3-Way Handshakes. Der Hacker kann so, durch das Testen unterschiedlicher Portnummern, eine Liste der installierten Dienste auf dem Zielsystem erstellen.

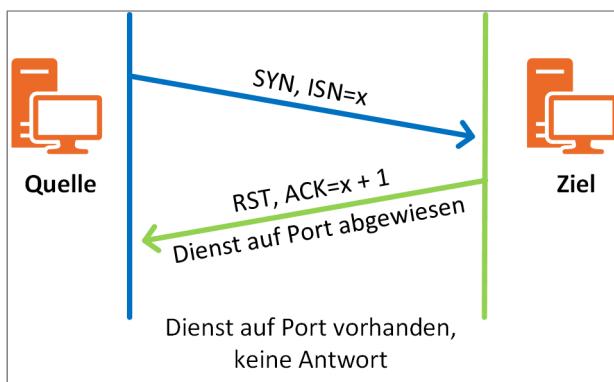
Dadurch aber, dass der Angreifer den dritten Teil des Handshakes nicht schickt, wird der Verbindungsaufbau nicht vollständig abgeschlossen. Software auf dem Zielsystem, die auf Verbindungsebene mitprotokolliert, wird in den Logfiles diesen Scanversuch nicht aufführen, da nie eine vollständige Verbindung aufgebaut wurde. Beim Ausbleiben des letzten Teils des Handshakes wird das Zielsystem bis zum Ablaufen des Timeouts warten und dann belegte Ressourcen wieder freigeben.

An dieser Stelle wird deutlich, dass diese Eigenschaft von TCP auch zur Störung des TCP/IP-Stacks und ggf. darauf aufsetzender Dienste benutzt werden kann. Ein derartiger Angriff wird als **Denial of Service (DoS)** bzw., wenn verteilte Systeme das Hacking gleichzeitig ausführen, **Distributed Denial of Service (DDoS)** genannt. Für jeden Verbindungsaufbau, der nach obigem Prinzip durchgeführt wird, belegt das Zielsystem Ressourcen, um die vermeintlich zu bearbeitenden Datenströme behandeln zu können.

Schickt nun ein Angreifer sehr schnell sehr viele dieser Segmente, so versucht das Zielsystem, für jede vermeintliche Verbindung entsprechende Ressourcen zu reservieren. Ein Angreifer, der ohnehin nicht plant, eine normale Datenverbindung aufzubauen, wird diese Ressourcen auf seinem System nicht reservieren. Da die Ressourcen auf dem Zielsystem so lange belegt bleiben, bis der entsprechende Timeout abgelaufen ist, kann ein derartiger Angriff, der innerhalb kurzer Zeit viele Verbindungen erstellt, ein System bis zum Stillstand ausbremsen.

SYN-ACK Scan

Eine weitere häufig genutzte Variante für Scavorgänge ist der sogenannte SYN-ACK Scan, der seinen Namen daher bezieht, dass in dem ursprünglich vom Angreifer gesendeten Segment nur das SYN- und ACK-Flag gesetzt wurde. Hier wird also eine Synchronisation angefordert und fälschlicherweise eine ISN bestätigt.



SYN-ACK Scan

Ein Zielsystem, welches die in der Abbildung dargestellte Anfrage erhält, reagiert auf zwei unterschiedliche Weisen:

- ✓ Ist der Port und damit der angefragte Dienst nicht vorhanden, wird als Antwort ein Segment mit dem RST-Flag zurückgesendet, es wird also die Verbindungsanfrage nicht angenommen.
- ✓ Ist der Port geöffnet, wird zunächst angenommen, dass es sich um ein Segment handelt, das einer Datenübertragung zuzuordnen ist. Da aber die vom Angreifer gesendete ACK-Nummer nicht in den Datenstrom einer gültigen Verbindung eingeordnet werden kann, wird das Zielsystem das Segment verwerfen.

Auch hier erhält der Angreifer Informationen über das Zielsystem. Um zu wissen, welche Ports geöffnet bzw. geschlossen sind, müssen nur über eine Reihe von Ports die entsprechenden Antworten bzw. deren Ausbleiben ausgewertet werden. Da auch hier keine offizielle Verbindung eingegangen wird, taucht dieser Scan in den Logfiles der jeweiligen Anwendungen nicht auf.

Scans dieser Art sind auch geeignet, einfachste Paket-Firewalls zu umgehen, die keine Verbindungsverfolgung (Connection Tracking) aufweisen. Firewalls mit Connection Tracking speichern dagegen die Statusinformationen aller ein- und ausgehenden Verbindungen (u. a. Quell- und Ziel-IP-Adresse, Portnummer, Flags, Sequenznummer und Timeouts) in einer Tabelle. Mit dieser Funktion kann man auch die Anzahl der gleichzeitigen Verbindungen (half-open/open) limitieren und somit auch DoS-Angriffen vorbeugen.

Weitere Methoden

Basierend auf dem Prinzip der SYN-ACK Scans existieren noch weitere Methoden, um TCP-Segmente so zu manipulieren, dass sie zur Informationsgewinnung vorbei an Protokolldateien und Firewalls eingesetzt werden können. Diese Methoden unterscheiden sich nur bei der Anzahl der gesetzten Flags im ausgesendeten TCP-Segment.

Beispiele sind:

- ✓ **FIN Scans**; hier wird nur das Finalize Flag gesetzt
- ✓ **X-Mas Scans**; hier werden alle TCP Flags gesetzt (Namenserklärung: X-Mas → Christmas → hell erleuchteter Weihnachtsbaum)
- ✓ **NUL Scans**; hier sind keine Flags gesetzt

Ähnliche Vorgehensweisen existieren für UDP-Segmente. Die Grundidee ist immer das Aus-senden von speziell formatierten Paketen und das Auswerten der zurückgesendeten Informa-tionen. Als Information wird teilweise auch das Ausbleiben von Antworten gewertet.

Fingerprinting (TCP/IP-Stack-Fingerprinting)

Die meisten Scanmethoden liefern nicht nur Informationen über die Existenz eines Rechners bzw. das Aktivsein eines bestimmten Dienstes auf diesem Rechner. Werden die als Antwort auf Anfragen empfangenen Pakete genau analysiert, ist es mitunter möglich, sehr genaue Rück-schlüsse über die verwendete Hardwareplattform bzw. das Betriebssystem zu ziehen.

Werden die Antworten, die ein Dienst als Bannerinformation sendet, untersucht, so kann relativ einfach in Erfahrung gebracht werden, um welches System es sich handelt. Diese Vorgehens-weise wird als **Bannergrabbing** bezeichnet. Der Befehl

```
telnet trajet.websitewelcome.com 25
```

im DOS-Fenster zeigt Ihnen folgendes Ergebnis:

```
Eingabeaufforderung
220-trajet.websitewelcome.com ESMTP Exim 4.91 #1 Sun, 13 Jan 2019 06:08:22 -0600
220-We do not authorize the use of this system to transport unsolicited,
220 and/or bulk e-mail.
help
214-Commands supported:
214 AUTH STARTTLS HELO EHLO MAIL RCPT DATA BDAT NOOP QUIT RSET HELP
ehlo
250-trajet.websitewelcome.com Hello [91.235.142.81]
250-SIZE 52428800
250-8BITMIME
250-PIPELINING
250-AUTH PLAIN LOGIN
250-STARTTLS
250 HELP
quit
221 trajet.websitewelcome.com closing connection

Verbindung zu Host verloren.

C:\Users\User>
```

Antwort eines ESMTP Servers auf Port 25

Das Beispiel zeigt die Informationen, die bei einem Verbindungsaufbau zu Port 25 (ESMTP) eines Rechners ausgegeben werden. Hieraus lassen sich sehr genau Name und Versionsnummer der verwendeten Software ablesen. Daraus jedoch den Umkehrschluss zu ziehen, ein System könnte man durch Abschaltung der Bannermeldungen einbruchssicher machen, wäre leichtsinnig. Viele Anbieter haben inzwischen die Bannerfunktionen deaktiviert.

Äußerst effektiv und deutlich schwerer abzuwehren ist Fingerprinting auf TCP/IP-Basis. TCP/IP schreibt nicht spezifisch vor, wie Sequenznummern generiert werden müssen oder Fenstergrößen etc. für eine neue Verbindung bzw. Antworten auf Pakete gewählt werden müssen. Die Hersteller eines jeden Betriebssystems, die TCP/IP-Funktionen in ihr System integrieren, haben also eine gewisse Wahlfreiheit.

Werden die Antworten eines Zielrechners hinsichtlich gewählter Sequenznummern, Fenstergrößen etc. ausgewertet, so ist es möglich, Hardware- und Betriebssystemimplementierungen zu unterscheiden, ohne dass explizite Verbindungen zwischen den Rechnern aufgebaut werden müssen. Wird diese Technik benutzt, um einen potenziellen Zielrechner zu identifizieren, ohne dass der Rechner des Angreifers selbst Pakete sendet, so handelt es sich um sogenanntes passives Fingerprinting. Die typischen Werte für TTL, Window Size, DF und TOS, die die Hersteller für ihre Betriebssysteme verwenden, erlauben eine Identifikation des Betriebssystems oder zumindest eine Eingrenzung auf wenige infrage kommende Kandidaten, die mit Analysen anderer Aspekte letztendlich identifiziert werden können.

Host-Fingerprinting-Tabelle (Auszug)

OS	VERSION	PLATTFORM	TTL	WINDOWSIZE	DF	TOS
Windows	9x/NT4	Intel	128	5000–9000	y	0
Windows	2000	Intel	128	17000–18000	y	0
Windows	XP	Intel	128	65535	y	0
Windows	7 – 10	Intel	128	8192	y	0
Windows	Server 2008	Intel	128	8192	y	16
IRIX	6.x	SGI	60	61320	y	16
FreeBSD	5.1(1)	Intel	64	65535	y	16
Linux	2.2/2.6	Intel	64	5860	y	0
OpenBSD	2.x	Intel	64	17520	n	16
SCO	R5	Compaq	64	24820	n	0
Solaris	8	Intel/Sparc	64	24820	y	0
macOS	10.x	Intel	255		n	0
Cisco	IOS 12.4	Motorola	255	4128	n	192

Verteilte Scans

Einen Zielrechner auf allen 65536 Ports zu scannen, würde für einen Angreifer möglicherweise zu lange dauern und auch zu auffällig sein. Der Nutzen hält sich zudem in Grenzen, da nicht alle Ports von entsprechender Serversoftware belegt sind und somit keinen Angriffspunkt bieten würden. Ein Hacker konzentriert sich deswegen auf wenige Ports, die von häufig installierter Software genutzt werden, zu der auch entsprechende Lücken bekannt sind.

Um das Risiko einer Entdeckung noch weiter zu senken, benutzen Hacker sogenannte verteilte Scans. Hier wird nicht von einem einzigen Rechner aus ein Netzwerk- oder Host-Scan durchgeführt, sondern eine Vielzahl von Rechnern, die unter der Kontrolle des Hackers stehen, bekommen jeweils einen Scanvorgang als Teilaufgabe zugeordnet. Somit wäre es im Extremfall möglich, dass für das Überprüfen von X Ports auf einem Zielsystem genau X kontrollierte Rechner benutzt werden, die ihrerseits jeweils nur einen einzigen Port untersuchen.

Das Ergebnis dieses verteilten Scans wird der Hacker dann in eine einzige Datenbasis zusammenführen, um so ein Gesamtbild der Situation zu erhalten. Werden die Einzel-Scans auch noch über einen größeren Zeitraum verteilt (Stunden, Tage, Wochen), so ist eine Bedrohung extrem schwer festzustellen.

Ein einzelnes Paket von einem einzelnen Rechner an einem bestimmten Wochentag kann nur sehr schwer einem Spähangriff zugeordnet werden, wenn dieses Paket in Millionen von Paketen untergeht, die pro Stunde in einem Netzwerk versendet werden können.

3.5 Wardriving

Suche nach offenen oder schlecht gesicherten WLANs

Das früher praktizierte Wardialing (die Suche nach Modemverbindungen anhand einer Telefonnummer) findet heute auf WLAN-Verbindungen Anwendung. WLANs stellen für Hacker eine verlockende Beschäftigungsmöglichkeit dar, da hier mit überschaubarem Aufwand ein Zugang zu einem Firmennetz bzw. zum Internet hergestellt werden kann.

Wardriving bezeichnet das Aufspüren von vorhandenen WLAN-Installationen, indem man mit WLAN-Equipment (vorzugsweise Notebook oder Smartphone mit WLAN-Karte) die infrage kommende Gegend abfährt. Ist eine bestimmte Firma das Ziel, so sind Straßen, die direkt an das Firmengebäude angrenzen, oder Kundenparkplätze ein idealer Aufenthaltsort für Wardriving-Aktivitäten. Regionale Listen von gehackten WLAN-Netzen findet man nicht nur in geschlossenen Internetforen. Wardriving verwendet auch jede WLAN-Software, um Systeme in seinem Umfeld zu identifizieren.

3.6 Social Engineering

Informationen im sozialen Umfeld

Mitunter ist die einfachste Möglichkeit für einen Hacker, in das Netzwerk einer Firma einzudringen, nicht ein erfolgreicher Angriff über die Firewall oder das Knacken eines Passworts. Vielerorts ist es erstaunlich einfach, sensible Informationen von den Mitarbeitern des Unternehmens selbst in Erfahrung zu bringen.

Die Grundziele des Social Engineerings sind dieselben wie die des Hackens im Allgemeinen: Der Hacker möchte unautorisierten Zugang zu Systemen oder Informationen, um Betrug, einen Netzwerkeinbruch, Industriespionage oder Identitätsdiebstahl zu begehen.

Social Engineering über das Telefon

Eine Variante des Social Engineering wird über das Telefon durchgeführt. Der Hacker gibt sich gegenüber dem angerufenen als Mitarbeiter des Helpdesks oder als ein Kollege aus, der zur Behebung eines Problems dringend Informationen des Systems benötigt.

Aber auch die Mitarbeiter des Helpdesks sind ein beliebtes Ziel für Social-Engineering-Angriffe: Es ist ihre Aufgabe, freundlich zu sein und den Hilfesuchenden Antworten zu geben. Berücksichtigen Sie nun, dass Helpdeskmitarbeiter in Sicherheitsbelangen meist schlecht bis gar nicht geschult sind, ergibt sich für den Hacker eine Informationsquelle von unschätzbarem Wert.

Ein Anrufer, der sich beim Helpdesk meldet und Probleme mit dem RAS-Zugang hat, wird vom Helpdesk minutiös erklärt bekommen, wie der RAS-Zugang konfiguriert ist und welche Einstellungen in der Firma gefordert sind. Falls der Helpdesk-Mitarbeiter nach dem Namen fragen sollte, kann der Hacker durchaus eine Antwort geben: Die öffentlichen Webseiten der Firmen informieren oft über Ansprechpartner und Mitarbeiter in den entsprechenden Abteilungen.

Dumpster Diving

Wörtlich übersetzt als „Mülleimertauchen“ ist Dumpster Diving ebenfalls eine nützliche Informationsquelle für den Hacker. Gelingt es dem Hacker, Zugang zu weggeworfenen Akten, Memos, Organisationsplänen, Backup-CDs oder Ähnlichem zu bekommen, lässt sich aus diesen Unterlagen rekonstruieren, wer welche Rolle im Unternehmen spielt, wer welche Telefonnummer hat, zu welchen Zeiten er/sie im Büro anwesend ist etc.

Beim weiteren Vorgehen des Hackers lassen sich die so gewonnenen Daten, wie Namen von Mitarbeitern, Telefonnummern, Projektdetails usw. geschickt einsetzen, sodass in zusätzlichen Schritten weitere Informationen gewonnen werden können.

Online Social Engineering

Online Social Engineering benutzt das Internet selbst, um neue Informationen zu gewinnen:

„Sie könnten gewinnen!“, heißt es möglicherweise in einer E-Mail an einen Mitarbeiter. Um an dem Gewinnspiel teilzunehmen, muss er nur schnell einen Fragebogen ausfüllen. Manche Menschen denken bei der Aussicht auf einen Gewinn nicht an den Wert der Informationen, die sie in ein Formular unbekannten Ursprungs eingeben.

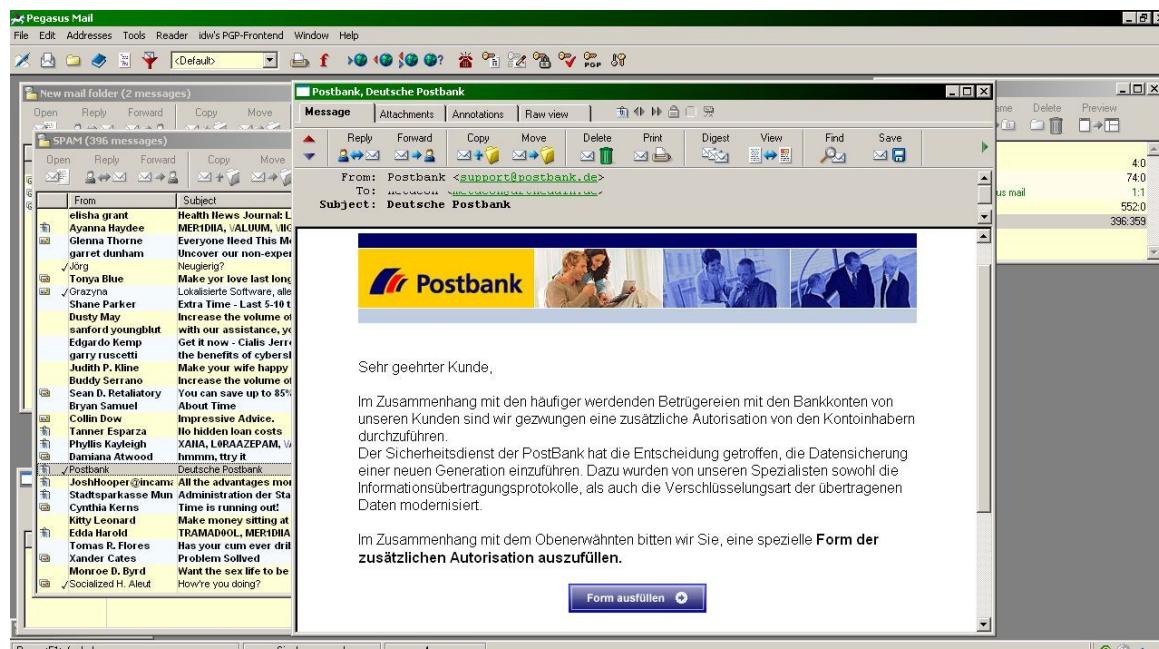
Denkbar wäre auch, den Mitarbeiter einen Account für einen kostenlosen Internetdienst anlegen zu lassen: „Sichern Sie jetzt 10 GB kostenlosen Webspace“, „gratis Antivirus-Download“ etc. Sobald der Account angelegt wurde, kann der Hacker das eingegebene Passwort versuchsweise bei anderen Accounts desselben Mitarbeiters benutzen. Da es sehr häufig vorkommt, dass Benutzer aus reiner Gewohnheit für verschiedene Accounts dasselbe Passwort verwenden, ist diese Methode, in den Besitz von Passwörtern zu kommen, relativ erfolgreich.

Phishing

Eine Variante der Social-Engineering-Angriffe ist das sogenannte Phishing (engl. Kunstwort aus „password“ und „fishing“: Passwort fischen). Bei dieser Art des Internetbetrugs werden zuerst massenhaft Mails verschickt, die vorgeben, von einer Bank, Onlinezahlungsdiensten wie PayPal oder Auktionshäusern wie eBay zu sein. Diese Mails gleichen in ihrem Erscheinungsbild den Mails der Originale.

Allen gemeinsam ist, dass unter irgendwelchen Vorwänden ein Log-in des Mail-Empfängers und potenziellen Phishing-Opfers auf der Website gefordert wird. In den Mails wird üblicherweise ein Link angegeben, der aber mit diversen Techniken verschleiert wurde und nicht auf die Original-Website, sondern auf eine Kopie führt (für Beispiele vgl. Abschnitt 7.4).

Der ahnungslose Nutzer, der auf diese Website gelangt und sich mit seinen authentischen Online-Zugangsdaten anmelden will, gibt den Phishern seine Daten preis, die so Zugang zu seinem Konto erhalten oder die gestohlenen Zugangsdaten anderweitig ausnutzen.



Eine Phishing-Mail

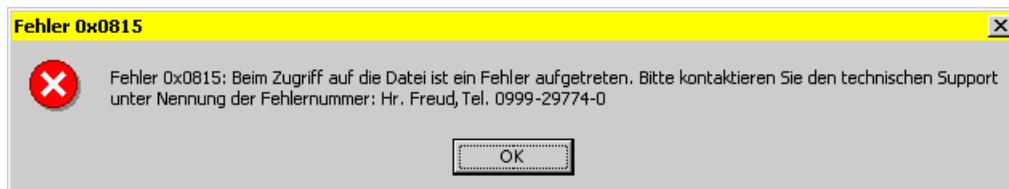
Reverse Social Engineering

Normale Social-Engineering-Versuche versetzen den Hacker immer in eine Situation, in der er sich Informationen von Mitarbeitern oder aus dem Web bzw. den Müllheimern selbst beschaffen muss. Dies kann dazu führen, dass bei den Mitarbeitern Zweifel auftreten, ob das Passwort richtig genannt werden sollen usw.

Deutlich trickreicher sind Reverse-Social-Engineering-Methoden, bei denen der Mitarbeiter dem Hacker die gewünschten Informationen freiwillig gibt. Das oben beschriebene Phishing arbeitet im Prinzip auch schon nach dieser Methode.

Zum Beispiel kann der Hacker einen Telefonanruf durchführen und sich als neuer Helpdesk-Mitarbeiter ausgeben – mit entsprechender Rufnummer. Beim Auftreten eines Problems bittet der betroffene Mitarbeiter dann den Hacker um Hilfe – dieser benötigt wiederum zur Lösung des Problems nur schnell einige Daten.

Für einen Hacker wird es keine Schwierigkeit sein, die Ursache des Problems zu beheben, da Sie davon ausgehen können, dass der Hacker für ein Problem im Netzwerk sorgen wird, um auch wirklich angerufen zu werden.



Fehlermeldung mit der Aufforderung, einen Supportmitarbeiter anzurufen

Idealerweise schafft der Hacker es, dem Benutzer beim Zugriff im Netzwerk oder in einer bestimmten Software eine entsprechende Fehlermeldung anzuzeigen. Der Rückruf erfolgt nun seitens des Benutzers, der dem vermeintlichen Supportmitarbeiter ohne Argwohn Auskunft über sensible Zugangsdaten gibt.

Der Vorteil aus der Sicht des Hackers ist beim Reverse Social Engineering, dass der Benutzer ihn kontaktiert, um die gewünschten Daten zu übermitteln, und dass diese Ereignisse weniger verdächtig sind und somit nicht so lange im Gedächtnis bleiben.

Der Nachteil für den Hacker ist, dass diese Methode eine deutlich längere Vorbereitungszeit, genauere Planung und möglicherweise auch schon einen Zugang zum Netzwerk oder zum Computer des Anwenders voraussetzt.

4

Angriffe auf Serverdienste

4.1 Exploits

Ziele eines Exploits

Ein Angriff auf einen Server mit signifikanten Folgen ist ein sogenannter Exploit. Ein Exploit, das „Ausnutzen“ einer vorhandenen Sicherheitslücke, gestattet es dem Hacker, ohne Kenntnis eines Systempasswortes oder sonstige Zugangsberechtigung die teilweise oder vollständige Kontrolle über das System zu erlangen. Als Exploits werden auch die Programme oder der Code bezeichnet, die an einen Server geschickt werden, um das vom Hacker gewünschte Verhalten zu provozieren.

Richtig eingesetzte Exploits sind deswegen so gefährlich, weil die Hacker durch sie auch administrative Rechte und damit uneingeschränkten Zugriff erlangen können. Voraussetzung hierfür ist, dass der Serverdienst Systemrechte erfordert. Auf jeden Fall erhält er dabei die Rechte des Benutzers, mit denen der gestartete Serverprozess läuft.

Die Möglichkeit für einen Exploit ergibt sich durch **Programmierungs- oder Konfigurationsfehler**. Der Hacker führt hier absichtlich eine Fehlersituation herbei, um das Verhalten des Zielsystems zu ändern. Im schlimmsten Fall ist das die Ausführung von eingeschleustem Code oder der Zugriff auf eine Benutzerumgebung mit Systemrechten.

Aufbau von Programmen

Um zu verstehen, wie ein Exploit funktioniert und welche Mechanismen es ermöglichen, Code nach Wahl des Hackers auf dem Zielsystem auszuführen, sollen zuerst die Arbeitsweise eines Programms im Normalzustand und sein Aufbau im Arbeitsspeicher des Rechners untersucht werden.

Wie Sie in der schematischen Grafik auf der folgenden Seite sehen, unterteilt sich der von einem Programm belegte Speicherplatz im RAM eines Computers in drei große Bereiche:

- ✓ Programmcode
- ✓ Heap
- ✓ Stack (auch Stapspeicher)

Bei den niedrigen Speicheradressen befindet sich der Maschinencode des Programms. Dieser enthält die Anweisungen an die CPU. Direkt darüber befindet sich der Heap-Bereich, der die Werte globaler und dynamischer Variablen speichert.

Da dynamische Variablen zur Laufzeit des Programms hinzukommen oder entfallen können, wird die obere Grenze des Heaps je nach Bedarf nach oben bzw. unten verschoben.

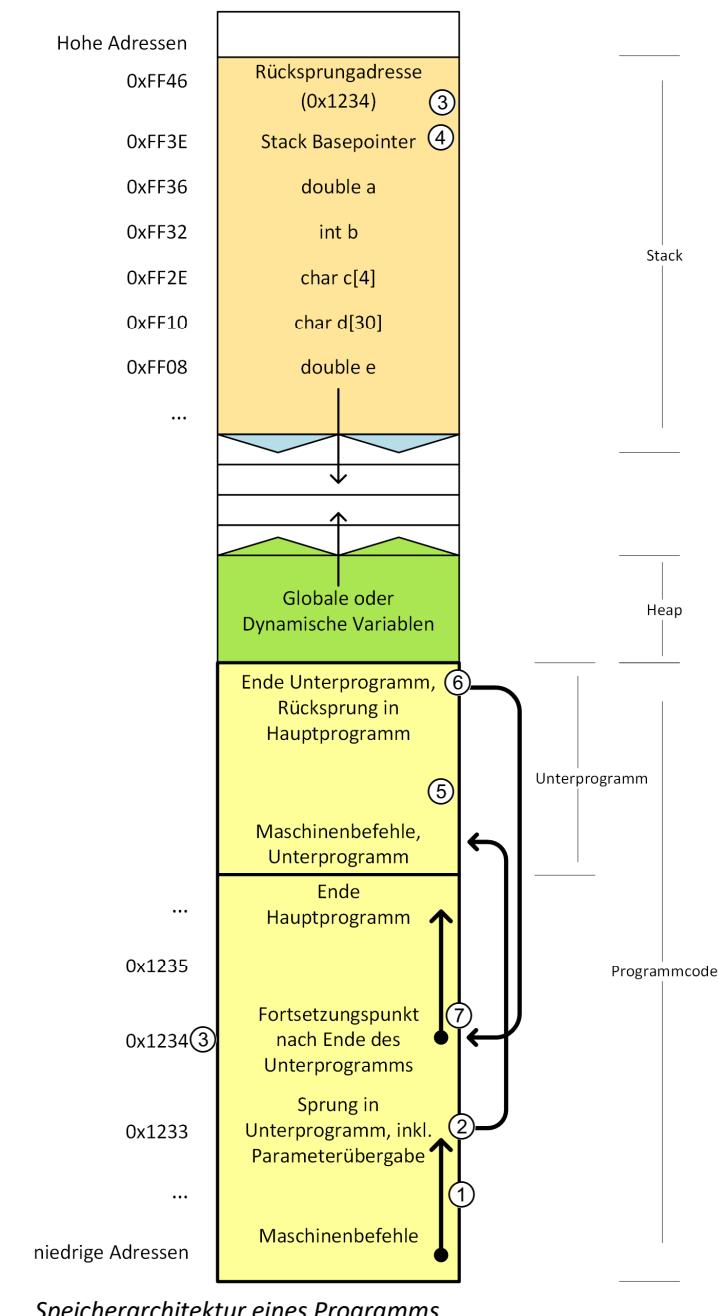
Der Stack-Speicher beginnt bei den hohen Speicheradressen und enthält automatische Variablen und Sprungadressen. Auch der Stack wächst und schrumpft während der Laufzeit eines Programms. Da der Stack im Arbeitsspeicher gewissermaßen auf dem Kopf steht, wächst er bei Bedarf in Richtung kleinerer Speicheradressen – also nach unten.

Programmablauf

Während der normalen Abarbeitung des Programms werden schrittweise die Befehle des Codeteils vom Prozessor verarbeitet, und es wird dann an der nächsthöheren Speicheradresse der nächste Befehl verarbeitet. Der Programmablauf erfolgt in der Grafik also zunächst von unten nach oben ①.

Soll vom Programm aus in ein Unterprogramm verzweigt werden, so befindet sich an entsprechender Stelle im Hauptprogramm eine entsprechende Sprunganweisung ②. Damit jedoch nach dem Ende des Unterprogramms die Bearbeitung des Hauptprogramms fortgesetzt werden kann, wird die Speicheradresse, die unmittelbar auf den Unterprogrammaufruf folgt, auf den Stack geschrieben – in diesem Beispiel ist das die Adresse 0x1234 ③.

Da auch das Unterprogramm einige Variablen zur Durchführung seiner Operationen benötigt, wird der Stack um benötigten Platz nach unten vergrößert und somit Platz für die Variablen a, b, c, d und e reserviert. Der Stack Basepointer definiert das Ende des Stacks ④. Erst nachdem diese Aktionen durchgeführt worden sind, erfolgen der eigentliche Sprung in das Unterprogramm und die Bearbeitung der dort befindlichen Kommandos ⑤.



Am Ende des Unterprogramms sind Anweisungen enthalten, die aus dem Stack die Rücksprungadresse (0x1234) auslesen, den Speicher wieder freigeben und einen Rücksprung in das Hauptprogramm veranlassen ⑥, wo dieses weiter abgearbeitet wird ⑦.

Schwachstelle Speicherverwaltung

Eine Schwachstelle von Programmiersprachen, Compilern und Programmierern, die zu den häufigsten Sicherheitslücken in Programmen führt, ist die Speicherverwaltung auf dem Stack, die sogenannte **Buffer Overflows** ermöglicht.

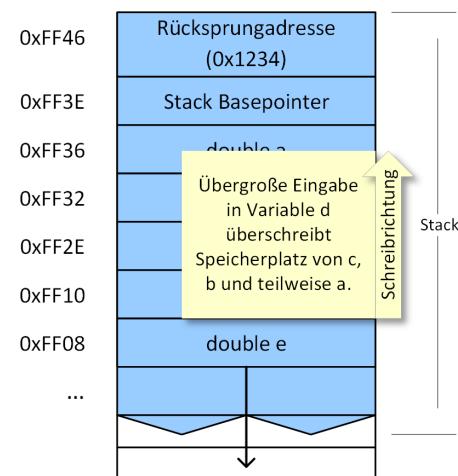
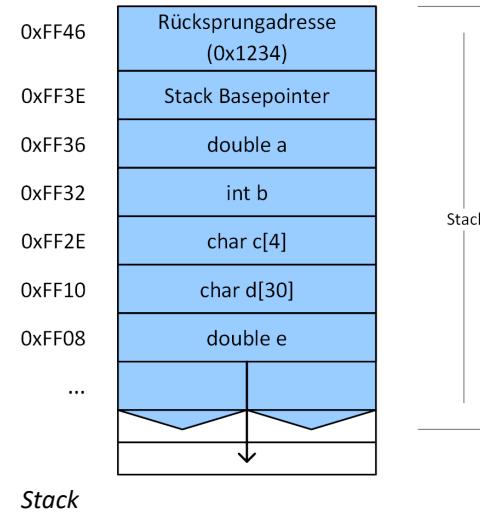
In einigen Programmiersprachen wird bei Lese- oder Schreibzugriffen auf Variablen **unzureichend geprüft**, ob diese innerhalb des dafür reservierten Speicherbereichs stattfinden.

Betrachten Sie hierfür den Stack aus dem Beispielprogramm genauer:

Nach der Rücksprungadresse und dem Basepointer werden von oben nach unten die Variablen in der Reihenfolge ihrer Definition mit dem benötigten Speicher reserviert. Für den String d wurden zum Beispiel 30 Zeichen reserviert.

Solange die Funktion in diesen Variablen nur Strings speichert, die eine Länge von bis zu 29 Zeichen haben, läuft die Ausführung des Programms normal.

(Das Ende jedes Strings muss durch eine binäre Null gekennzeichnet werden, die das 30. Byte belegt.)



Ein Buffer Overflow

Würde eine Eingabe in diesen String erfolgen, die 30 oder mehr Zeichen enthält, würden während der Eingabefunktion in aufsteigender Reihenfolge die nächsten Bytes überschrieben – sofern das Unterprogramm die Maximalgröße des Strings nicht explizit berücksichtigt.

Zuerst würden die zur Variablen c gehörenden Bytes überschrieben, dann die Bytes der Variablen b usw.

In diesem Zustand sind die Daten des Programms schon durch den Programmfehler korrumptiert worden, da die für den Programmablauf nötigen Variablen c, b und gegebenenfalls auch a mit anderen Werten überschrieben wurden – der Puffer ist übergelaufen (Buffer Overflow).

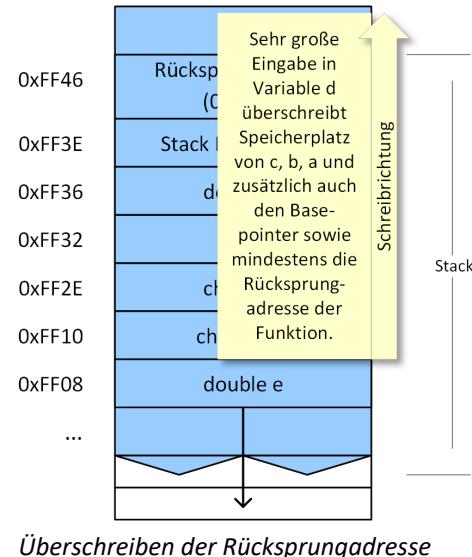
Würde in diesem Zustand ein Lesezugriff auf den eingegebenen String d erfolgen, wäre dieser korrekt gespeichert – trotz seiner Überlänge. Wird allerdings auf die Variablen a, b oder c zugegriffen, erhält man meist sinnlose Daten, da z. B. der durch Stringvariablen veränderte Bereich der Variablen a immer noch als Datentyp double interpretiert wird.

Werden so viele Daten in diese Variable geschrieben, dass der komplette Stack dieses Unterprogramms überschrieben wird, so sind danach der Basepointer, die Rücksprungadresse und schließlich die darüberliegenden Speicherzellen nicht mehr verfügbar. Endet nun das Unterprogramm, so wird für den Rücksprung in das Hauptprogramm die Speicheradresse ausgelesen, an der die Rücksprungadresse gespeichert war.

An der ausgelesenen Adresse 0xFF46 befinden sich nun aber Daten der Variablen d, die als Adresse interpretiert werden.

Angenommen, die rechts abgebildete Zeichenkette überschreibt die 2-Byte-Rücksprungadresse so, dass die Zeichenkette „Fu“ die ursprüngliche Rücksprungadresse 0x1234 ersetzt, so ergibt eine Interpretation der Zeichenkette „Fu“ als Adresse den Wert 0x4675.

Der Computer wird also, sobald das Unterprogramm beendet ist, zur Speicherstelle 0x4675 springen, um dort mit der Ausführung des Hauptprogramms fortzufahren. Da die angesprungene Speicheradresse ein vom Inhalt der Variablen d abhängiges Zufallsprodukt ist, steht an der angesprungenen Adresse nicht der gewünschte Code. Ein derart beschädigtes Programm wird meist an diesem Punkt abstürzen.



Buffer Overflows in Serverdiensten

Derartig aufgebaute Funktionen werden in Software immer benutzt, wenn Eingaben verarbeitet werden. Die Quelle der Eingabe spielt dabei eine untergeordnete Rolle.

Wurde bei der Programmierung versäumt, die Größe einer Zeichenkette in einer Funktion auf ihren maximal zulässigen Wert zu beschränken, ist es möglich, einen Pufferüberlauf gezielt herbeizuführen. Überprüft z. B. ein HTTP-Server, der Anfragen aus dem Internet erhält, nicht die maximale Länge des URL-Eingabestrings, kann dieser Dienst möglicherweise remote zum Absturz gebracht werden.

Ein Angreifer, der die Kontrolle über ein System übernehmen will, möchte nicht primär einen Dienst zum Absturz bringen und damit seinen Zugriff einschränken oder ggf. unmöglich machen. Sein Ziel ist es, Befehle seiner Wahl auf dem angegriffenen Rechner durchzuführen.

Aufbau von Exploitcodes

Bei der Konstruktion eines Exploits wird versucht, eine überlange Zeichenkette an ein fehlerbehaftetes Programm so zu übertragen, dass die durch Überschreiben manipulierte Rücksprungadresse einer Funktion zu Befehlen führt, die der Hacker auf dem Zielsystem ausführen will. Der aus Sicht des Hackers ideale Fall ist damit, eine Shell (Eingabeaufforderung) auf dem attackierten Rechner zu starten. Gelingt es dem Hacker, im Zielsystem einen Befehl anzuspringen, der zum Start einer Shell führt und diese auf den Rechner des Hackers umleitet, so hat er diesen Rechner unter seine Kontrolle gebracht.

Der Befehl, eine Shell zu starten (indem z. B. unter Windows das Programm CMD.EXE gestartet wird), muss vom Hacker ebenso zum Zielrechner übermittelt werden wie die IP-Adresse, auf die die Ein- und Ausgabe der Shell umgeleitet werden soll. Die Übermittlung der entsprechenden Befehle erfolgt innerhalb der Zeichenkette, die für den Pufferüberlauf benutzt wird.

Der Hacker hat dabei mehrere Probleme zu umgehen:

- ✓ Er weiß nicht, an welcher Stelle sich die Rücksprungadresse im Speicher befindet.
- ✓ Er kennt die absoluten Speicheradressen des Programms nicht.
- ✓ Die übertragene Zeichenkette darf keine binären Nullen enthalten.

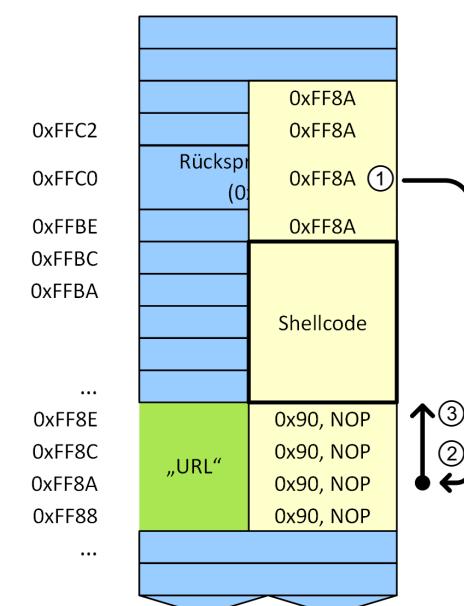
Ist dem Hacker der Sourcecode der Anwendung bekannt (den Quellcode bekommt er meist durch Nutzung eines Decompilers), so kann er durch Analyse die Größe des verwundbaren Stacks exakt berechnen. Damit lässt sich herausfinden, wie viele Bytes er in den Puffer schreiben muss, um die Stelle mit der Rücksprungadresse gezielt überschreiben zu können. Falls er diese Informationen allerdings nicht besitzt, wird sein Exploitcode am Ende die neue Rücksprungadresse mehrfach beinhalten.

Da der Hacker versucht, einen Sprung in den von ihm übermittelten Code zu provozieren, aber die absolute Adresse nicht kennt, weiß er nicht genau, wohin der Sprung führt, wenn er eine bestimmte Adresse angibt. Hier wird viel experimentiert, da ein blinder Sprung meist überhaupt nicht innerhalb des Exploitcodes endet und somit zum Programmabsturz führt. Ziel des Hackers ist es, eine Sprungadresse zu finden, die auf den eigenen, vom Hacker programmierten Code verweist.

In nebenstehend abgebildetem Stack hat der Programmierer eine Variable gefunden, deren Eingabegröße nicht überprüft wird. Nehmen Sie an, es handele sich um einen Webserver und die betroffene Variable sei diejenige, in der die URL gespeichert wird.

Wie Sie sehen, wurden anstatt der vorgesehenen Speichermenge für die URL viel mehr Daten übermittelt und der Stack wurde somit überschrieben.

Sobald die Funktion, in der sich die Eingaberoutine befindet, beendet wird, wird die Rücksprungadresse gelesen. Diese wird verwendet, um in das Hauptprogramm zu kommen ①. An der ausgelesenen Stelle befindet sich nicht mehr die ursprüngliche Adresse 0x1234, sondern die vom Hacker eingesetzte Adresse 0xFF8A. Der Hacker vermutet, dass sich 0xFF8A noch auf dem Stack befindet.



In einen Stack eingeschleuster Exploit

Da er nicht weiß, wie groß der Stack genau ist, hat er den Wert 0xFF8A mehrmals am Ende seiner übertragenen Zeichenkette angehängt.

Im abgebildeten Beispiel gehört die Adresse 0xFF8A wirklich schon zu dem vom Hacker beeinflussten Speicherbereich und wird angesprungen ②. Da der Hacker jedoch nicht wissen kann, welche Stelle genau in seinem Code angesprungen wird, hat er in seinem Exploit sozusagen eine „Landezone“ eingerichtet. Ein Teil des Exploits besteht aus der Zeichenkette 0x90, was bei Intel-basierten Systemen der Hex-Code für den Maschinenbefehl **NOP** (No Operation) ist.

Diese als **NOP-Sliding** bezeichnete Technik hat den Effekt, dass, unabhängig davon, welche Stelle genau angesprungen wird, die CPU einen Taktzyklus lang wartet und dann den nächsten Befehl bearbeitet. Auf diese Weise läuft die Ausführung des Codes schrittweise nach oben, bis der eigentliche Shellcode beginnt ③.

Aufbau eines Shellcodes

Der Shellcode ist der Teil eines Exploits, der das Zielsystem veranlasst, eine Eingabeaufforderung zu starten und diese auf den Rechner des Angreifers umzuleiten. Es handelt sich hierbei um ein kurzes Programm in Maschinensprache, das Befehle für den Start der Shell und deren Umleitung enthält.

Damit dies gelingt, benötigt dieses Programm mindestens drei Variablen: die IP- und Portadresse des Angreifers sowie Name und Ort des Shellprogramms auf dem Zielsystem (z. B. `\WINDOWS\SYSTEM32\cmd.exe`).

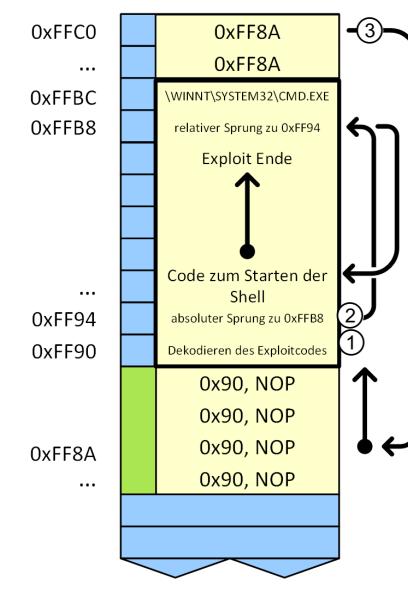
Bei der Programmierung hat der Hacker nicht unbedingt freie Hand. Der Exploitcode, den er an das Zielsystem übermitteln will, darf keine binären Nullen enthalten. Seine Daten werden bei der Übertragung an den Puffer als Zeichenkette interpretiert, die definitionsgemäß mit einer binären Null terminiert wird. Er muss also den Wert `0x00` vermeiden, damit der Exploit korrekt übertragen wird. Dieses Problem lösen Hacker unter anderem durch eine Quasi-Verschlüsselung des eigentlichen Maschinencodes, die in der Prozessausführung rückgängig gemacht wird.

Des Weiteren kann der Hacker die Zeichenketten für die Shell nicht benutzen, solange er deren direkte Adressen nicht kennt. Diese ermittelt er durch mehrfache Sprunganweisungen.

Wie nebenstehendes Beispiel verdeutlicht, führt die geratene Rücksprungadresse zu irgendeiner Stelle im NOP-Bereich des Exploits. Zuerst wird der gesamte Shellcode-Block dekodiert, um die versteckten binären Nullen wieder sichtbar zu machen ①.

Jetzt benötigt der Hacker die direkte Adresse des Strings mit dem Namen der Eingabeaufforderung. Es erfolgt ein relativer Sprung zur Speicheradresse, die direkt vor dem gewünschten String steht. Diesen Abstand kann er leicht berechnen (hier 40 Bytes). Der Maschinenbefehl lautet also "Springe 0x28 Bytes vorwärts" ②.

Der Befehl dort enthält nur die Anweisung, sofort wieder zurückzuspringen, allerdings nicht als absoluter, sondern als relativer Sprung. Dies gleicht dem Aufruf eines Unterprogramms ③.



Bei einem Unterprogrammaufruf wird auf dem Stapelspeicher die Rücksprungadresse abgelegt, um das Hauptprogramm an der entsprechenden Stelle fortsetzen zu können. In diesem Fall ist die Rücksprungadresse bei einem Sprungbefehl von `0xFFB8` die `0xFFBC`. `0xFFBC` ist aber die exakte Position der Shell-Zeichenkette.

Jetzt wird der Rest des Codes bearbeitet, und der Hacker kann auf die absolute Adresse seines Strings zugreifen, indem er die vermeintliche Rücksprungadresse auf dem Stack ausliest. Das Betriebssystem wird aufgefordert, eine Eingabeaufforderung zu öffnen und diese auf den angreifenden Rechner umzuleiten. Damit ist das Exploitprogramm beendet.

Der Sicherheitskontext (Ausführungsrechte des Exploit-Codes)

Ab diesem Zeitpunkt verfügt der Hacker über eine Eingabeaufforderung auf seinem System, mit dem er den angegriffenen Rechner fernsteuern kann. Dieser Angriff ist deswegen so verheerend, weil der vom Hacker kontrollierte Prozess aus der Sicht des Zielsystems immer noch ein Prozess des gerade attackierten Serverdienstes ist.

Wurde ein Webserver mit administrativen Rechten gestartet, so besitzt auch der Hacker diese Rechte. Wäre ein Webserver mit den eingeschränkten Rechten eines Benutzers gestartet worden, so hätte der Hacker auch nur die limitierten Rechte des Users.

Buffer-Overflow-Angriffe auf Computersysteme sind deshalb so gefährlich, weil viele Systemdienste verlangen, dass sie mit administrativen Rechten auf den jeweiligen Rechnern gestartet werden.

Das Know-how eines Hackers

Das Entdecken einer Sicherheitslücke und das Entwickeln eines passenden Exploits ist eine komplexe Aufgabe. Einen funktionsfähigen Exploit zu erstellen, erfordert exzellente Kenntnisse über die Systemdienste, die man angreifen möchte, die verwendete Hardwareplattform und fortgeschrittene Programmierkenntnisse. Neben den professionell agierenden Hackern und Crackern bilden experimentierfreudige Script-Kiddies, die auf die von Hackern entwickelten Tool-boxen zurückgreifen, eine nicht kalkulierbare Bedrohung. Man kann bei ihnen kein statistisch nachvollziehbares Angriffsmuster prognostizieren.

Mitunter ist es gar nicht notwendig, eine Sicherheitslücke zu suchen. Gelegentlich sind Sicherheitslücken im Quellcode der Software vorhanden, um „staatlichen“ Hackern eine Möglichkeit zu eröffnen, die Applikation anzugreifen.

Eine weitere Möglichkeit stellen die diversen Internet-Tauschbörsen da. Dort werden gecrackte Applikationen mit integrierten Exploits kostenfrei bereitgestellt.

4.2 Rootkits

Was passiert, wenn ein System geknackt wurde?

Hat ein Hacker per Exploit die Kontrolle über ein System übernommen, hängt die weitere Vorgehensweise nur noch von den Absichten und Fähigkeiten des Hackers ab.

Üblicherweise wird ein Hacker, der das gerade in Besitz genommene System zu einem späteren Zeitpunkt wieder benutzen will, nicht den umständlichen Weg über einen Exploit gehen. Diese Möglichkeit könnte ihm ja auch durch ein Softwareupdate zunichtegemacht werden. Der Hacker wird den praktischen Weg gehen und sich zuerst einen User-Account anlegen, der ebenfalls mit administrativen oder zumindest Benutzerrechten ausgestattet ist. Anschließend kann der Hacker seine Exploit-Shell verlassen und sich auf normalem Weg in das System einloggen.

Wozu Rootkits gebraucht werden

Um die Spuren seiner Tätigkeiten in den protokollierenden Logfiles des Betriebssystems und ggf. in den Files der Applikation zu verschleiern, gibt es mehrere Möglichkeiten:

- ✓ Manuelles Modifizieren der entsprechenden Logfiles des Betriebssystems und der Applikationen
- ✓ Automatisches, skriptgesteuertes Manipulieren aller bekannten Logfiles

Bezüglich der Verschleierung der Spuren besteht für die Hacker jedoch oftmals ein gravierender Nachteil:

Obwohl die Existenz des Hackers auf diese Weise rückwirkend aus dem System getilgt werden kann, werden zukünftige Log-ins ggf. wieder in den Protokollen aufgeführt. Auch ein Blick in die Tabelle mit laufenden Prozessen kann dem Systemadministrator die Anwesenheit eines ungeliebten Gastes offenbaren.

Aus diesem Grund sind in einschlägigen Kreisen die sogenannten **Rootkits** (engl. „Root“: Wurzel = Name des Superuser-Accounts in UNIX-Systemen) sehr beliebt. Ein Rootkit enthält wichtige Module des Betriebssystems des Zielsystems in modifizierter Form.

Der Hacker ersetzt die Originalkomponenten des Zielrechners durch die Komponenten aus seinem Rootkit und veranlasst gegebenenfalls einen Neustart. Die Betriebssystem-Funktionen des Rootkits wurden so manipuliert, dass der Rechner weiterhin seine gewohnte Arbeit ausführt. Allerdings werden sämtliche Prozesse und Aktivitäten, die auf den Hacker zurückzuführen sind, verborgen.

Professionelle Rootkits

Früher waren Rootkits aufgrund der notwendigen komplexen Programmierung eher selten anzutreffen und dann auch nur bei Betriebssystemen, deren Sourcecode frei zur Verfügung stand. Dadurch konnten kundige Programmierer leicht modifizierte Varianten der Systemmodule erstellen.

Gegen diesen vermeintlichen Schutz für die nicht quelloffenen Betriebssysteme spricht z. B., dass die Firma Sony bei einer professionellen Firma ein Rootkit für Windows entwickeln ließ, um dieses zusammen mit kopiergeschützten Audio-CDs an arglose Kunden zu verkaufen.

Der beabsichtigte Zweck dieses Rootkits war, sämtliche Module des Kopierschutzsystems für den Kunden unsichtbar zu machen, damit dieser die Kopierschutzkomponenten nicht deinstallieren konnte.

Allerdings verursachte das Rootkit auf den befallenen Windows-Rechnern Systeminstabilitäten und Abstürze und ermöglichte es, auch andere Dateien neben denen des Sony-Kopierschutzes zu tarnen.

Ein Cracker, der ein Rootkit benötigte, musste sich nur die Software von einer derart „geschützten“ Sony Audio-CD besorgen und konnte sich so die aufwendige Arbeit sparen, selbst ein Rootkit zu entwickeln.

4.3 DoS/DDoS/DRDoS

Denial-of-Service-Angriffe (DoS) und DDoS

Eine DoS-Attacke dient dazu, den TCP/IP-Stack eines Dienstes im Netzwerk zu überlasten. Sie kann z. B. als **SYN-Flooding** oder **Smurf-Attacke** ausgeführt werden. Dadurch wird die Kommunikation mit dem Dienst des Zielrechners stark eingeschränkt oder unterbrochen. Im Extremfall, z. B. durch Fehler in der Programmierung des Dienstes, kann dieser zum Absturz gebracht werden. Einen DoS-Angriff durchzuführen, ist für einen Hacker kein Problem. Er benötigt hierzu nur die Adresse des Zielsystems und den Port des anzugreifenden Dienstes. Da dieses Hacking ein simples Angriffsmuster hat, sind auch die Gegenmaßnahmen einfach. Es genügt, auf der Firewall im Zielnetz die Anzahl der gleichzeitigen Verbindungsanfragen von einer IP-Adresse zu limitieren.

Verteilte DoS-Angriffe auf einen Serverdienst, auch DDoS (Distributed Denial of Service) genannt, sind dagegen schwerer abzuwehren, weil diese über viele unterschiedliche IP-Adressen initiiert werden. Jedoch auch bei DDoS gibt es Optionen. Hierzu sollte eine Firewall die gleichzeitigen Anfragen auf Ports reglementieren.

Für verteilte DoS-Angriffe installiert der Hacker auf nicht ausreichend geschützten Systemen einen **Trojaner** (vgl. Abschnitt 6.5). Damit kann er diese infizierten Rechner (Bots) als **Botnetz** nutzen. Neben DDoS-Attacken sind das Versenden von Spam-Mails oder andere Cyber-Angriffe hierüber möglich. Die Bots können vom Hacker über einen zentralen oder verteilten Command-and-Control-Server gesteuert werden.

Botnetze können einige Tausend bis einige Hunderttausend übernommene Systeme umfassen. Man kann sie auch als BaaS (Botnet-as-a-Service) für definierte Zeiträume mieten.

Hier eine Auswahl der wichtigsten Malwaretypen, die für die Bildung von Botnetzen verantwortlich sind:

- | | | |
|---------------------|---------------------|---------------------------|
| ✓ Sality/Sality P2P | ✓ Ramdo | ✓ Conficker A/Conficker B |
| ✓ Mobile Fakeinst | ✓ Zero Access 2 P2P | ✓ APT EquationDrug |
| ✓ TinyBanker | ✓ Adware | ✓ Trojan |

Auf der Seite <https://map.lookingglasscyber.com> finden Sie in Echtzeit die existierenden Botnetze, deren Infektionsrate und die territoriale Verbreitung. Weitere Informationen zu Botnetzen stehen Ihnen unter <https://wiki.botfrei.de/Botnetze> zur Verfügung.

Smurf-Attacks

Eine Variante eines DDoS-Angriffs, die ohne zuvor installierte spezielle Software auskommt, ist die sogenannte **Smurf-Attacke** (engl. „Smurf“: Schlumpf) oder auch ICMP Smurf.

Der Angreifer schickt hierbei einen ICMP Echo Request (auch „Ping“ genannt) an die Broadcast-Adresse eines ungeschützten Netzwerkes. Hierbei wird der Angreifer bevorzugt ein Netzwerk wählen, in dem sich möglichst viele Rechner befinden.

Üblicherweise wird ein Rechner, der einen ICMP Echo Request zugesandt bekommt, ein ICMP Echo Reply an den Absender schicken, um zu signalisieren, dass die IP-Verbindung in Ordnung ist.

Durch das Senden eines Echo Requests an die Broadcast-Adresse eines Netzwerkes fühlen sich sämtliche in diesem Netzwerk aktiven Rechner angesprochen und antworten mit einem Echo Reply. Der Angriff kommt dadurch zustande, dass der Angreifer in dem ursprünglichen Echo Request nicht seine eigene IP-Adresse als Absender verwendet hat, sondern die seines Opfers. Deswegen werden sämtliche Echo Replys des angepingten Netzwerkes an das Opfer gesandt.

Auch hier führt der enorme Verbrauch an Netzwerk- und Rechenressourcen zu einem Stillstand des Zielsystems.

Distributed-Reflected-Denial-of-Service-Attacke (DRDoS)

Bei der DRDoS-Attacke kommuniziert der Hacker nicht direkt mit dem anzugreifenden System, sondern mit regulär arbeitenden Internetdiensten. Als Absenderadresse trägt er dabei die IP-Adresse des anzugreifenden Systems (**IP-Spoofing**). Die Beantwortung dieser Anfragen stellt dann den eigentlichen DoS-Angriff auf das Zielsystem dar. Deshalb ist der Ursprung des Angriffs für das Opfer nicht mehr direkt ermittelbar.

Folgende DRDoS-Typen finden hierbei Anwendung:

- ✓ SYN Reflection Attacks
- ✓ SNMP/NTP/CHARGEN Reflection Attacks
- ✓ DNS Reflection Attacks
- ✓ Gaming Server Reflection Attacks
- ✓ Memcached Amplification Attack

4.4 Sniffer

Ausspähen von Passwörtern mit Keyloggern

Durch die volle Verfügungsgewalt und das Installieren entsprechender Tools kann der Hacker in den Besitz weiterer sensibler Informationen gelangen. So ist es z. B. möglich, von einem Rechner die Passwordatenbank herunterzuladen und zu versuchen, die enthaltenen Passwörter mithilfe spezieller Software knacken zu lassen.

Effektiver für den Hacker ist es, im System ein Programm zu verankern, das alle Tastatureingaben im Anmeldedialog mitprotokolliert und dem Hacker zur Verfügung stellt (**Passwort-Sniffer**). Auf diese Weise gelangt er relativ schnell an die Anmeldeinformationen von Administratoren und Usern.

Protokollieren des Netzwerk-Datenverkehrs

Auch die Überwachung des gesamten Datenverkehrs aus der Sicht des kompromittierten Computers ist denkbar. Netzwerk-Sniffer-Software ändert die Eigenschaften der installierten Netzwerkkarte in einem Computer so, dass nicht nur für diesen Computer bestimmte Pakete gelesen werden, sondern sämtliche Pakete, die an dem Netzwerkinterface ankommen (promiscuous mode). Das wird erst dann effizient, wenn der Angreifer den kompletten Verkehr durch einen Man-in-the-middle-Angriff (vgl. Abschnitt 4.6) überwachen kann.

Um das Datenaufkommen in Grenzen zu halten, wird der Hacker meist gezielt nach bestimmten Informationen suchen lassen und nur diese protokollieren lassen. Ein lohnendes Ziel ist z. B. die Suche nach unverschlüsselten Passwörtern, die im Netzwerk übertragen werden, und ihren Nutzern. Dies wird nachfolgend exemplarisch an zwei Beispielen aufgezeigt.

```
C:\>ftp 192.168.1.37
Verbindung mit 192.168.1.37 wurde hergestellt.
220-FileZilla Server version 0.9.37 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
Benutzer (192.168.1.37:(none)): peter.pan
331 Password required for peter.pan
Kennwort:
230 Logged on
ftp> dir
200 Port command successful
150 Opening data channel for directory list.
```

Beispiel einer FTP-Sitzung mit Passworteingabe

Beim FTP-Server meldet sich der Nutzer (USER) peter.pan mit seinem Passwort (PASS) peter.pan an.

Source	Destination	Protocol	Info
192.168.1.37	192.168.1.34	FTP	Response: 220-FileZilla Server version 0.9.37 beta
192.168.1.37	192.168.1.34	FTP	Response: 220-written by Tim Kosse (Tim.Kosse@gmx.de)
192.168.1.37	192.168.1.34	FTP	Response: 220 Please visit http://sourceforge.net/projects/filezilla/
192.168.1.34	192.168.1.37	FTP	Request: USER peter.pan
192.168.1.37	192.168.1.34	FTP	Response: 331 Password required for peter.pan
192.168.1.34	192.168.1.37	FTP	Request: PASS peter.pan
192.168.1.37	192.168.1.34	FTP	Response: 230 Logged on
192.168.1.34	192.168.1.37	FTP	Request: PORT 192,168,1,34,9,24
192.168.1.37	192.168.1.34	FTP	Response: 200 Port command successful
192.168.1.34	192.168.1.37	FTP	Request: LIST
192.168.1.37	192.168.1.34	FTP	Response: 150 Opening data channel for directory list.

Tracefile der FTP-Sitzung

Viele Nutzer sind sich bei der Nutzung eines E-Mail-Clients auch nicht bewusst, dass die Verbindung zum Postausgangsserver (SMTP) und dem Posteingangsserver (POP3 bzw. IMAP) im Normalfall unverschlüsselt erfolgt. Die Nutzung des User-Accounts (USER) mit einem unverschlüsselten Passwort (PASS) ermöglicht es einem Hacker, Einsicht in fremde E-Mails zu bekommen bzw. E-Mails unter falschem Namen zu versenden.

Source	Destination	Protocol	Info
193.24.239.34	192.168.2.112	SMTP	S: 250-mail.netz-werker.net 250-PIPELINING 250-SIZE 102456789
192.168.2.112	193.24.239.34	SMTP	C: AUTH LOGIN
193.24.239.34	192.168.2.112	SMTP	S: 334 VXNlcm5hbWU6
192.168.2.112	193.24.239.34	SMTP	C: User: bWIwMDkw
193.24.239.34	192.168.2.112	SMTP	S: 334 UGFzc3dvcmQ6
192.168.2.112	193.24.239.34	SMTP	C: Pass: M3Fyb2J4dkI=
193.24.239.34	192.168.2.112	SMTP	S: 235 2.0.0 Authentication successful

Verbindungsauflaufbau eines E-Mail-Clients mit seinem Server

Allerdings bieten die E-Mail-Provider auch eine gesicherte Kommunikation an. Hierfür muss man in den Einstellungen des E-Mail-Accounts für die Protokolle SMTP, POP3 oder IMAP die SSL/TLS-Verschlüsselung aktivieren und die Portnummern der Protokolle ändern.

4.5 Replay-Attacken

Verwendung aufgezeichneter Kommunikation

Ist der Hacker in der Lage, beispielsweise per Sniffer-Software, den Datenverkehr in einem Netzwerk aufzuzeichnen, muss er nicht unbedingt den Inhalt der Datenpakete verstehen, um unautorisiert Zugang zu einem System erlangen zu können.

Kann der Hacker davon ausgehen, dass er eine verschlüsselte Kommunikation zur Autorisierung eines Benutzers aufgezeichnet hat und dass der Verschlüsselungsschlüssel immer derselbe ist, wäre es möglich, die aufgezeichneten Pakete zu einem späteren Zeitpunkt wieder ins Netz einzuspielen und sich dadurch als der betreffende Benutzer zu identifizieren.

Aus diesem Grund wäre es leichtsinnig, immer denselben Schlüssel zu benutzen. In Protokollen, bei denen die Bedrohung durch Replay-Attacken extra berücksichtigt wurde, werden daher zusätzliche Informationen zusammen mit den verschlüsselten Daten transportiert, die es ermöglichen, aufgezeichnete Datenpakete von Originalen zu unterscheiden. Das Einbringen eines Zeitstempels in die Datenpakete hat sich in diesem Fall am besten bewährt. Pakete mit ungültiger oder veralteter Zeitinformation werden demnach in sicheren Protokollen als ungültig verworfen.

4.6 TCP/IP Session-Hijacking

Designschwächen im TCP-Protokoll

Wie Sie bereits erfahren haben, war das Hauptziel bei der Entwicklung der zur TCP/IP-Familie gehörenden Protokolle das Herstellen einer zuverlässigen Verbindung zwischen Rechner-systemen und nicht die optimale Sicherheit im Datenverkehr.

Die beim Handshake zwischen zwei Parteien ausgetauschten Sequenznummern sollen später die Wiederherstellung der korrekten Reihenfolge von Datenströmen garantieren. Dabei ist im Standard nicht eindeutig festgelegt, wie die ISN (Initial Sequence Number) festzulegen sind. Je nach Implementierung können hier unterschiedliche Regeln zum Einsatz kommen.

Die unterschiedliche Sequenznummerierung ist jedoch bei jedem Betriebssystem typisch. Ist dem Angreifer bekannt, wie ein Betriebssystem X eine neue ISN für die nächste Verbindung bildet, kann er dieses Wissen gezielt ausnutzen. Zum Beispiel, wenn einem Hacker bekannt ist, dass das ver-wendete Betriebssystem für eine neue ISN die ISN der vorausgegangenen Verbindung + 1 verwen-det, kann er zuerst eine normale Verbindung zum Zielsystem aufbauen und sofort wieder beenden. Damit kann er die nächste ISN des Betriebssystems berechnen.

Will das Zielsystem nun eine neue Verbindung mit einem Dritten herstellen, kann der Hacker an der Stelle des Dritten mithilfe der berechneten ISN den Verbindungsaufbau in seinem ACK (Acknowledgement) bestätigen. Als Absender benutzt der Angreifer die IP-Adresse des Dritten, sodass das Zielsystem keine Möglichkeit hat, zu erkennen, dass der Angreifer statt des Dritten in der Kommunikation antwortet.

Eine TCP-Sitzung entführen (Man-in-the-middle-Angriff)

Ein weiterer, sehr technischer Angriff benutzt ebenfalls eine Schwachstelle der Sequenznummern im TCP: das sogenannte **Session Hijacking**. Hier schafft es der Angreifer, sich in eine bestehende TCP-Verbindung zwischen zwei Kommunikationspartner zu drängen. Bei der Kommunikation eines Clients mit seinem Server übernimmt der Angreifer aus der Sicht des Clients die Rolle des Servers und aus der Sicht des Servers diejenige des Clients. Durch Analyse der empfangenen TCP-Segmente und geschickte Manipulation der Sequenznummern und Absenderadressen aller versendeten Pakete bemerken weder Client noch Server, dass die Kommunikation nicht mehr direkt, sondern über einen Dritten läuft. Auf diese Weise kann der Hacker den gesamten Datenverkehr zwischen Client und Server beobachten. Handelt es sich um den Aufbau einer geschützten Verbindung, braucht der Hacker nur abzuwarten, bis sich der Client beim Server authentifiziert hat und kann dann den so authentifizierten Datenstrom missbrauchen, um eigene Befehle einzuschleusen. Dies kann beispielsweise so erfolgen:

- ✓ Der Client will eine verschlüsselte Verbindung zu einem Server A herstellen.
- ✓ Der Hacker suggeriert dem Client, dass er der Server A sei und baut seinerseits eine verschlüsselte Verbindung zum Server A auf.
- ✓ Der Client authentifiziert sich mit Namen und Passwort am Hackersystem. Dieser leitet die Daten zum Server A weiter.
- ✓ Der Client sendet seine Kommandos zum Server A über das Hackersystem.
- ✓ Der Hacker modifiziert oder filtert seinerseits die Kommandos und leitet diese an den Server A weiter.
- ✓ Auf dem Server A werden die Kommandos ausgeführt und das Ergebnis an das Hackersystem übermittelt.
- ✓ Der Hacker modifiziert oder filtert die Antwort des Servers A so, dass eine Manipulation unerkannt bleibt.
- ✓ Der Client erhält vom Hackersystem eine manipulierte Antwort auf seine gesendeten Kommandos.

Dies ist eine sehr gefährliche Art des Angriffes, da sie dem Hacker erlaubt, die Autorisierung des Clients abzuwarten und dann Befehle in seinem Namen einzuschleusen.

Session Hijacking ist zwar ein technisch hoch komplexer Vorgang, es sind jedoch entsprechende Softwaretools im Umlauf, die es auch Amateuren ermöglichen, derartige Angriffe durchzuführen.

ARP-Spoofing

Diese Methode nutzt, im Gegensatz zu den oben beschrieben Verfahren, Eigenschaften des Layer-2-Protokolls (Data Link Layer). Dazu nutzt der Angreifer das **Address Resolution Protocol (ARP)**. ARP-Spoofing manipuliert in einem Netzwerk dabei den ARP-Cache der anzugreifenden Systeme, indem die Zuordnung der MAC-Adresse zur IP-Adresse verändert wird. Dies gibt dem Angreifer dann die Möglichkeit, den Verkehr auf sein System umzuleiten und so die Verbindung zwischen Rechnern abzuhören bzw. zu manipulieren. Der Hacker fungiert somit als Proxy zwischen diesen Systemen.

4.7 Übung

Fragen zu Angriffe auf Serverdienste

Übungsdatei: --

Ergebnisdatei: uebung04.pdf

1. Was ist ein Exploit?

a	Ein Virenschutzprogramm
b	Eine Software zum Ausnutzen von Fehlfunktionen bei Systemdiensten
c	Eine Systemfunktion des Kernels
d	Ein Script, das gezielt Sicherheitslücken in einer Applikation ausnutzt
e	Eine Rescue-Funktion von Anti-Malware

2. Welche Möglichkeiten sind Ihnen bekannt, um Angriffe auf Client-/Serversysteme zu realisieren?
3. Was ist ein Botnet?

5

Sicherheitsprobleme durch Mitarbeiter

5.1 Ausfall/Krankheit

Wer hat das Passwort und das Know-how?

Die Person mit den höchsten Befugnissen in einem Netzwerk ist üblicherweise der Netz- oder Systemadministrator. Unabhängig davon, ob die Verwaltungsaufgaben in kleineren Firmen nur von einer Person ausgeübt oder in großen Netzwerken auf mehrere Administratoren mit genau definierten Aufgabenbereichen verteilt werden, besitzen alle Administratoren erweiterte Zugriffsrechte, um diesen Aufgaben nachkommen zu können. Außerdem kennen Administratoren den Aufbau und die Organisation der von ihnen verwalteten Rechnernetzwerke.

In vielen Firmen sind für geplante Abwesenheitszeiten dieser Mitarbeiter schon Regelungen in Kraft, die zum Beispiel vor dem Urlaubsantritt eines Administrators eine Übergabe von Informationen und Berechtigungen an die Vertretung vorsehen. Kommt es aber zu einem unvorhergesehenen Ausfall eines Mitarbeiters, z. B. durch einen Unfall, eine schwere Krankheit oder Tod, so kann dies gravierende Konsequenzen für die IT-Infrastruktur eines Unternehmens haben, wenn dieser Mitarbeiter als Einziger für die Netzwerkverwaltung oder einen Teilbereich hiervon (z. B. Serveradministration) zuständig war.

Auch die unerwartete Kündigung eines Mitarbeiters kann schwerwiegende Konsequenzen nach sich ziehen. Möglicherweise wird zwar das Administratorpasswort übergeben, aber während eines kurzen Übergabegesprächs wird es kaum möglich sein, alle denkbaren Besonderheiten der Organisations- und Verwaltungsaspekte des Netzwerks zu besprechen, über die nur der scheidende Administrator Bescheid wusste. Dies kann auch lange nach dem Ausscheiden des alten Administrators noch zu unangenehmen Überraschungen für seinen Nachfolger führen.

Den Notfall einplanen

Bei der Planung von Vertretungsregelungen sollten auch unvorhergesehene Fälle berücksichtigt werden, um die Fortführung der administrativen Tätigkeiten zu gewährleisten. Dabei sollten folgende Bedingungen erfüllt sein:

- ✓ Der Stand von Projekten, Konfigurationen und Verfahrensweisen muss jederzeit ausreichend (schriftlich) dokumentiert sein.

- ✓ Die Benennung einer Vertretung alleine reicht nicht aus, um eine Fortführung der geforderten Tätigkeiten im Notfall garantieren zu können. Es ist zu prüfen, ob und wie die Vertretung geschult werden muss, damit sie auch in der Lage ist, die gestellten Aufgaben bei Bedarf zu erfüllen. Wenn sich bei einer derartigen Überprüfung herausstellt, dass aufgrund des geforderten Spezialwissens kurzfristig keine Ersatzkraft eingesetzt werden kann, ist es besonders wichtig, diese Kräfte langfristig zu schulen.
- ✓ Es muss genau definiert sein, welche Aufgaben im Notfall auf welche Vertreter aufgeteilt werden.
- ✓ Der Vertreter darf die erforderlichen Berechtigungen nur im Notfall erhalten (also keine vollständigen administrativen Rechte).
- ✓ Wenn es nicht möglich ist, für Personen einen kompetenten Vertreter zu benennen oder zu schulen, sollte der Einsatz einer externen Vertretung eingeplant werden. Dabei ist auch die Reaktionszeit im Einsatzfall zu fixieren.

Besonders kritisch ist die Situation, wenn Sie bei der Analyse des Ist-Zustands bei Personalaufgaben an einen „Single-Point-of-Knowledge“ gelangen. Besitzt eine einzelne Person das alleinige Spezialwissen, um unternehmensrelevante Aufgaben im IT-Bereich wahrnehmen zu können, so sollten Sie mit besonderer Sorgfalt für deren Ausfall Vertreter schulen oder externe Fachkräfte auswählen.

5.2 Unrechtmäßige Systemzugänge

Administrative Zweitzugänge

Die Möglichkeit, Rechte zu erteilen und Zugangsmöglichkeiten zu schaffen, kann von administrativen Mitarbeitern auch fahrlässig bzw. böswillig eingesetzt werden. So mag es vielleicht sinnvoll sein, einen zusätzlichen Account mit erweiterten Rechten anzulegen und die Zugangsdaten für diesen Zweit-Account in einem versiegelten Briefumschlag in der Firmenzentrale sicher aufzubewahren. Dieser Briefumschlag dürfte nur geöffnet werden, wenn das Passwort des normalen Accounts vergessen wurde oder der entsprechende Mitarbeiter für ein dringendes Problem nicht zu erreichen ist. Allerdings schaffen diese Zugänge neue Angriffspunkte, wenn diese z. B. auch zur Fernwartung eingesetzt werden können und deswegen für Benutzer aus dem Internet nutzbar sind.

Unzufriedene Mitarbeiter könnten auch gezielt Möglichkeiten schaffen, die es ihnen erlauben, ohne die normalen Sicherheitsfunktionen und Protokolle Zugang zu Ressourcen des Unternehmens zu erhalten. Ein unscheinbarer Account mit unverfänglichem Namen, aber administrativen Rechten oder ein zusätzlich an einen Rechner angeschlossener WWAN-Stick (Wireless Wide Area Network auf der Basis von LTE, WiMax oder UMTS), über den mit dem Internet kommuniziert werden kann, unterwandern jede Protokollierung in einer Unternehmens-Firewall.

Systemzugänge programmieren

Auch bei der Entwicklung von Software bieten sich für den Programmierer Möglichkeiten, zusätzliche undokumentierte Funktionen zu implementieren. Diese können, je nach Komplexität der Software und den Absichten des Programmierers, von einfachen Debugging-Schnittstellen bis zur kompletten Umgehung der in der Software vorhandenen Sicherheitsmechanismen reichen. Im letzteren Fall wäre eine derart manipulierte Software quasi eine Applikation, in der vom Programmierer bei der Entwicklung schon eine Hintertür (Backdoor) implementiert hat. Mit zunehmender Größe des Softwareprojekts wird es für den Programmierer leichter, unbemerkt unübliche Funktionen im gewöhnlichen Sourcecode zu verstecken. Als Gegenmittel ist hier die stringente Einhaltung der strukturierten Programmierung und deren Validierung anzuwenden.

Ein fiktives Beispiel für eine vom Programmierer des Systems bereits eingebaute Hintertür wurde im Hollywoodfilm „Wargames“ gezeigt. Hier erhielt ein Hacker Zugriff auf einen Computer der amerikanischen Streitkräfte, da er herausfand, dass der Entwickler den Namen seines verstorbenen Sohnes als Hintertür eingebaut hat.

5.3 Spionage

Datenmitnahme

Mitarbeiter, die berechtigt sind, Informationen zu lesen, können diese Berechtigung auch nutzen, um eine Kopie dieser Daten anzufertigen. Spätestens dann, wenn die Notwendigkeit nicht mehr vorhanden ist, dass ein Benutzer auf bestimmte sensible Daten zugreifen können muss, sollte ihm die entsprechende Berechtigung wieder entzogen werden.

Da nur schwer verhindert werden kann, dass während der befugten Arbeit mit sensiblen Daten bereits Kopien angefertigt werden, sollten in Bereichen mit hohem Sicherheitsanspruch Überlegungen angestellt werden, wie der Transport von Daten aus dem geschützten Bereich heraus verhindert werden kann. Besonders die Verbreitung von USB-Anschlüssen an PCs und die hohe Verfügbarkeit von USB-Memory-Sticks stellen mangels ausreichender ins Betriebssystem integrierter Kontrollmethoden zuweilen ein Problem dar.

Folgende Überlegungen sollten hier angestellt werden:

- ✓ Gibt es eine Möglichkeit für Benutzer, Laufwerke über den USB-Port anzuschließen, um Daten zu kopieren?
- ✓ Können Daten auf externen Laufwerken (NAS, SAN, Cloud) gespeichert werden?
- ✓ Besteht vom geschützten PC aus die Möglichkeit eines Internetzugangs?
- ✓ Wenn der Internetzugang nötig ist: Welche Programme sind zugelassen bzw. unabdingbar notwendig?

Eine Lösung für den eingeschränkten USB-Zugriff auf Windows-8/10-Systeme wird nachfolgend aufgezeigt:

- ▶ Starten Sie den Registrierungs-Editor (`regedit.exe`) als Administrator.
- ▶ Erstellen Sie eine Sicherungskopie der aktuellen Windows-Registry (Menüpunkt *Datei - Exportieren*), damit Sie im Fehlerfall auf den ursprünglichen Zustand der Einstellungen zurückkehren können.
- ▶ Öffnen Sie bei Windows 8 das Verzeichnis
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control`.
- ▶ Rufen Sie den Menüpunkt *Bearbeiten - Neu - Schlüssel* auf. Geben Sie als Schlüsselnamen *StorageDevicePolicies* ein. Achten Sie auf die korrekte Schreibweise.
- ▶ Öffnen Sie bei Windows 10 das Verzeichnis
`HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows`.
- ▶ Rufen Sie den Menüpunkt *Bearbeiten - Neu - Schlüssel* auf. Geben Sie als Schlüsselnamen *RemovableStorageDevices* ein. Achten Sie auf die korrekte Schreibweise.
- ▶ Rufen Sie im Feld *StorageDevicePolicies* (Windows 8) bzw. *RemovableStorageDevices* (Windows 10) den Menüpunkt *Bearbeiten - Neu - DWORD-Wert* (32-Bit bzw. 64-Bit, abhängig vom Betriebssystem) auf und setzen Sie als Namen *WriteProtect* (Schreibweise beachten) ein.
- ▶ Doppelklicken Sie mit der linken Maustaste *WriteProtect* und geben Sie über den Punkt *Ändern* im Feld Wert eine 1 ein.
Damit wird der Schreibschutz auf USB-Devices aktiviert. Dadurch wird ein potenzielles Sicherheitsrisiko ausgeschlossen. In Ausnahmefällen ist ein Neustart des Systems erforderlich.

Erpressung/Manipulation

Mitarbeiter, die illoyal gegenüber dem Unternehmen sind, könnten erhebliche Schäden verursachen, indem sie z. B. die entwendeten Kundendaten oder Forschungsergebnisse veröffentlichten oder an Dritte veräußern. Verärgerte Mitarbeiter (zu wenig Lohn oder Gehalt, Kündigung o. Ä.) können solche Aktionen initiieren, da sie, viel leichter als Hacker, direkten Zugriff auf unternehmenskritische Daten haben, um diese manipulieren oder löschen zu können.

Statistiken über Schadensfälle im IT-Bereich weisen darauf hin, dass die Mehrheit der Fälle von den eigenen Mitarbeitern verursacht wurden und nur ein geringerer Anteil von unternehmensexternen Personen.

5.4 Mangelnde Kompetenz

Falsche Bedienung

Die Hauptursache für Sicherheitsprobleme liegt – im Gegensatz zu den vorsätzlichen Sicherheitsverletzungen – in der mangelnden Kompetenz der Mitarbeiter. Viele erhalten, wenn überhaupt, nur eine eingeschränkte Einarbeitung in die IT-Umgebung am Arbeitsplatz. Findet diese statt, so ist sie oft auf das Ziel „Erfüllung der Aufgabe“ ausgerichtet, nicht aber auf die sicherheitstechnischen Belange des IT-Betriebes.

Viele Sicherheitsprobleme entstehen deshalb durch das Fehlverhalten der Benutzer selbst. Mitunter wird dies auch durch nicht ergonomische Software mit verwirrenden Dialogen und umständlicher Bedienung gefördert.

Zu den wichtigsten Problemen gehören:

- ✓ Fehlerhafte Konfiguration des Betriebssystems bzw. der Applikationen
- ✓ Unabsichtliches Löschen von Dateien
- ✓ Versehentliches Senden von sensiblen Daten an Unberechtigte
- ✓ Inkonsistenz bei der Synchronisation von Daten

Fehlerhafte Konfiguration

Auch auf der administrativen Seite überwiegt der menschliche Anteil der Fehlerursachen und Sicherheitsrisiken. Die unzureichende Konfiguration von Software oder eine Installation Out-of-the-Box ohne weitere Wartung und Konfiguration lässt teilweise viele Sicherheitsprobleme unberücksichtigt. Durch vermehrt eingesetztes Plug & Play in Hard- sowie Software ist es für den Administrator vielfach einfacher geworden, eine lauffähige Installation zu erhalten, ohne sich konkret mit dem neuen System selbst auseinanderzusetzen und die daraus resultierenden Implikationen für die IT-Sicherheit zu bedenken.

Meist haben die Administratoren auch nicht viele Alternativen, da dem Thema Sicherheit zwar im Marketing und auf der Managementebene viel Bedeutung beigemessen wird, aber im operativen Geschäft keine bzw. nur wenig Zeit und Ressourcen für Sicherheitsprozesse aufgewandt werden. So ist der Administrator aufgrund seiner vielfältigen Aufgaben z. B. gezwungen, Installationen ohne ausreichende Sicherheit zum Laufen zu bekommen, weil er sich schon um die nächste Aufgabe kümmern muss.

Ein historisches Beispiel hierfür ist eine Sicherheitslücke im Internet Explorer, die Anfang 2010 bekannt wurde und die Betriebssysteme Windows XP, Vista und Windows 7 traf. Der für diese Sicherheitslücke entwickelte IE-Exploit ermöglichte es Hackern, über eine manipulierte HTTP-Seite den Code einzuschleusen und so den Rechner zu übernehmen. Durch diesen Exploit wurden massive Angriffe (Codename Aurora) auf die Webserver von Google und anderen Firmen durchgeführt. Besonders interessant war dabei, dass der Exploit-Code in diversen Mailinglisten auftrat, was es jedem ermöglichte, diesen Code auszuführen.

Daraus lassen sich folgende Maßnahmen ableiten:

- ✓ Regelmäßiges Update der Applikationen nach vorherigem Testen des Updates
- ✓ Sofern Sicherheitslücken durch Update nicht beseitigt werden können, Wechsel zu einer sicheren Applikation
- ✓ Regelmäßige Recherche über Sicherheitsdefizite von Applikationen und Betriebssystemen

Eine repräsentative, jedoch nicht vollständige Liste von Sicherheitsvorfällen der Anbieter finden Sie hier:

Anbieter	Sicherheitsbulletin
Adobe	https://helpx.adobe.com/security.html
Android	https://source.android.com/security/
Apple	https://support.apple.com/en-us/HT201222
Cisco	https://tools.cisco.com/security/center/publicationListing.x
Citrix	http://support.citrix.com/securitybulletins/
Debian	https://www.debian.org/security/
Google	https://developers.google.com/compute/docs/security-bulletins
Linux	http://www.linuxsecurity.com/
Microsoft	https://blogs.technet.microsoft.com/msrc/
Mozilla	https://www.mozilla.org/en-US/security/advisories/
OpenOffice	http://www.openoffice.org/security/bulletin.html
Oracle	https://www.oracle.com/technetwork/topics/security/alerts-086861.html
Red Hat	https://access.redhat.com/site/security/updates
Ubuntu	https://usn.ubuntu.com

Herstellerübergreifende Sicherheitsinformationen erhalten Sie auch vom Computer Emergency Response Team (CERT). Es agiert auf nationaler und internationaler Ebene. Hier ein Auszug der Kontaktdaten:

- ✓ CERT Deutschland, <https://www.cert-verbund.de/>
- ✓ CERT Österreich, <https://www.cert.at/>
- ✓ CERT EU, <http://cert.europa.eu/cert/filteredition/en/CERT-LatestNews.html>
- ✓ CERT USA, <https://www.us-cert.gov/>

Die publizierten Sicherheitsinformationen basieren auf dem CVE-Standard (Common Vulnerabilities and Exposures). Dieser definiert eine einheitliche zeitliche und namenstechnische Konvention für Sicherheitslücken (Vulnerabilities) und Gefährdungen (Exposures). Das jeweilige Risiko eines CVEs wird durch das Common Vulnerability Scoring System (CVSS) abgebildet. Das Bewertungskriterium erfolgt von informell bis critical.

Eine umfassende und herstellerübergreifende CVE-Liste wird von der Mitre Corporationen (<https://cve.mitre.org/>) in Zusammenarbeit mit den CVE Numbering Authority (CNA) verwaltet. Die CNAs sind autorisierte Sicherheitspartner (u. a. CERT, Hersteller von Soft- und Hardware, Sicherheitsexperten).

Sicherheitsvorschriften

Manchmal kennen Mitarbeiter die Sicherheitsvorschriften (Sicherheits-Policies) und Arbeitsvorgänge nicht, die für die IT-Sicherheit an ihrem Arbeitsplatz gelten.

Die Sicherheitsvorschriften selbst können Grund für Probleme sein, wenn z. B. in einem Unternehmen ein beliebiger Mitarbeiter Datenschutzbeauftragten oder IT-Sicherheitsbeauftragten wird, der nicht über die notwendige Fachkompetenz zur Erfüllung dieser Aufgaben verfügt und fachlich nicht die Kompetenz besitzt, eine Sicherheitsvorschrift zu erstellen.

Schulen Sie nicht nur das IT-Personal und die IT-Sicherheits-Mitarbeiter, sondern die gesamte Belegschaft. Die Wahrscheinlichkeit, dass ein erfahrener Administrator ein unverlangt von Unbekannten zugesendetes Attachment in einer Mail öffnet, ist deutlich geringer, als dass ein Mitarbeiter einer Nicht-IT-Abteilung aus Neugier ein derartiges Attachment ausführt.

In größeren Firmen existiert ggf. auch ein Problem der Personal-Hierarchie. Sofern die IT-Sicherheit nicht direkt unterhalb der oberen Firmenleitung angesiedelt ist, könnten leitende Angestellte der mittleren Führungsebene IT-Sicherheitsmaßnahmen nach eigenem Ermessen auslegen.

Dies wird bei Unternehmen mit kritischen IT-Infrastrukturen (u. a. Energieversorgung, Gesundheitswesen, Transportwesen, Telekommunikationsversorger, Banken und Versicherungen, Wasserversorgung) vom Gesetzgeber angemahnt. Die Maßnahmen zur Bewusstmachung von IT-Sicherheitsmaßnahmen (Information Security Awareness) sollten Sie stufenweise umsetzen:

Integration: Um alle Mitarbeiter des Unternehmens in den IT-Sicherheitsprozess einzubeziehen, sollten Sie die Sensibilität der Mitarbeiter zur Datensicherheit ermitteln. Dies kann auch anonymisiert erfolgen. Die hieraus gewonnenen Erkenntnisse bilden die Basis für das Umsetzen der Sicherheitsregeln.

Akzeptanz und Regeln: Es muss bei allen Beteiligten der Wille zum Umsetzen von Sicherheitsmaßnahmen geweckt werden. Dies kann z. B. durch praxisrelevante Workshops erreicht werden. Erst wenn reale Sicherheitslücken bekannt gemacht werden, wird eine Akzeptanz erreicht. Die Regeln zum Beseitigen der Sicherheitsprobleme werden dann bewusster umgesetzt.

Nachhaltigkeit: Nur fortlaufende Kommunikation über Sicherheitsvorschriften, die konsequente Einbindung der Mitarbeiter in diesen Prozess und die ständige Aktualisierung der Sicherheitsrichtlinien erzielen den Erfolg der Sicherheitsrichtlinien.

5.5 Übung

Fragen zu Sicherheitsproblemen

Übungsdatei: --

Ergebnisdatei: uebung05.pdf

1. Wie können Administratoren IT-Sicherheitsprobleme vermeiden?
2. Wie können Sicherheitsprobleme, die durch Mitarbeiter verursacht werden, minimiert werden?

6

Virenarten und ihre Verbreitung

6.1 Grundkonzepte von Viren

Geschichtlicher Rückblick

Das Konzept von Computerviren reicht so weit zurück wie die Existenz von Computern generell. Schon 1949 stellte der Informatiker John von Neumann eine Theorie von „sich selbst reproduzierenden Automaten“ auf. Neumann dachte also an Programme, die die nötigen Anweisungen enthalten, um sich selbst weiter zu kopieren. Damit hatte er schon Ende der 40er-Jahre die wesentliche Eigenschaft eines Computervirus definiert.

Fred Cohen definierte 1983 schließlich auch formal den Begriff „Computervirus“ und stellte einen selbst entwickelten UNIX-Virus vor. 1984 veröffentlichte Cohen seine Doktorarbeit mit dem Titel „Computer Viruses – Theory and Experiments“. Diese Arbeit sorgte international für Aufsehen, und ihre Veröffentlichung ist sehr umstritten, da sie neben der wissenschaftlichen Arbeit auch zahlreiche experimentelle Viren enthielt.

Robert T. Morris, Student an der Universität Cornell in den USA, schrieb 1988 den ersten **Wurm**, der als „der Internet-Wurm“ in die Geschichte einging. Der Wurm verbreitete sich selbstständig aufgrund eines Programmierfehlers in UNIX im Internet. Aufgrund eines Programmierfehlers von Morris verbreitete sich der Virus allerdings exponentiell und legte so innerhalb von wenigen Stunden fast das gesamte Internet still.

In den 90er-Jahren wuchs die Anzahl existierender Viren und Varianten weiterhin exponentiell. Viele kursierende Sourcecodes und sogenannte Virus Construction Kits sowie Mutation Engines erlaubten auch wenig versierten Programmierern das Erstellen komplexer Computerviren.

1995 tauchten die ersten **Makroviren** auf. Das bis dahin geltende Credo, Viren könnten keine Datendateien befallen, war somit überholt. Da Makrosprachen einfacher erlernbar sind als assemblerbasierte Sprachen, führte dies zu einer großen Anzahl von Virenprogrammierern.

Seit 2000 waren Würmer auf dem Vormarsch. „Nimda“ läutete die Zeit der hybriden Bedrohungen ein. Der Begriff bezeichnet die Tatsache, dass diese Viren nicht nur einen, sondern mehrere Verbreitungswege suchen und großflächig vernetzte Rechnersysteme infiltrieren.

Diese schädlichen Programme werden auch als **Malware** (lateinisch **malus**: schlecht; englisch **malicious**: bösartig) bezeichnet.

Es herrscht eine hohe Professionalisierung in der Szene der Malware-Programmierer. Viren werden mittlerweile nicht mehr geschrieben, um in der Szene zu beweisen, dass man der „beste Virenprogrammierer“ ist.

Ein Teil der heute kursierenden Malware wurde von organisierten Kriminellen zu dem Zweck programmiert, die infizierten Computer fernzusteuern, um sie in ein sogenanntes **Botnetz** zu verwandeln.

Hunderttausende derart infizierter Rechner warten so auf Befehle des Master-Computers und stehen unter der Kontrolle der Viren- und Wurmautoren. Mithilfe dieser ferngesteuerten Rechner lässt sich dann leicht ein DDoS-Angriff durchführen (vgl. Abschnitt 4.3) oder – was der kommerzielle Hauptzweck dieser Netze zu sein scheint – Spam versenden.

Da sich mit dem Versand von Spam viel Geld verdienen lässt, ist es bei den Autoren derartiger Schadprogramme Praxis, die gekaperten Botnetze ganz oder teilweise an spamwillige Firmen zu vermieten, damit diese ihre unerwünschten Werbebotschaften an Millionen versenden können, ohne dabei eigene Computerressourcen aufwenden zu müssen.

Grundbauplan eines Virus

Ein Virus besteht aus mehreren funktionalen Komponenten, von denen eine obligatorisch ist, die anderen aber nicht unbedingt vorhanden sein müssen.

- ✓ Infektion
- ✓ Payload (Nutzlast)
- ✓ Tarnung

Grundsätzlich ist ein Virus ein Programm, das von der CPU befehlsweise abgearbeitet wird. Die grundlegende Eigenschaft eines Virus ist allerdings, dass dieses spezielle Programm die nötigen Befehle enthält, die zur Erzeugung von Kopien seiner selbst führen. Ein Virus verbreitet sich, indem er sich selbst in noch nicht infizierte Dateien kopiert. Der Schadcode wird ausgeführt, wenn diese Datei geöffnet bzw. gestartet wird.

Der Teil des Computervirus, der sich mit der Anfertigung von Kopien seiner selbst beschäftigt, wird **Infektionsroutine** genannt. Im Laufe dieses Kapitels werden Sie erfahren, welche unterschiedlichen Möglichkeiten zur Infektion von Computerviren benutzt werden können.

Das aus dem militärischen Bereich stammende Wort **Payload** bezeichnet die „Nutzlast“ eines Virus. Jedoch ist in seltensten Fällen von echtem Nutzen die Rede, sondern von den im Virus verankerten Schadensfunktionen.

Ein Virus kann abhängig von seiner Programmierung die unterschiedlichsten Funktionen ausführen.

Dies reicht von der Anzeige störender Bildschirmmeldungen bis zum Löschen von Dateien oder dem Unbrauchbarmachen ganzer Datenträger. Meist koppeln die Programmierer diese Schadensfunktionen auch an bestimmte Ereignisse. Die Schadensfunktion kann an ein bestimmtes Datum (z. B. Freitag, den 13.) oder an ein anderes Ereignis (z. B. 50. Neustart des Rechners) gekoppelt sein. Es existieren allerdings auch Viren, die keine Payload mit sich führen, da sie z. B. nur als konzeptionelle Viren geschrieben wurden.

Auch Viren ohne Payload sind nicht ganz ungefährlich. Durch fehlerhaft programmierte Infektionsroutinen können auch diese zu einem Datenverlust führen.

Um auf einem System nach der Infektion nicht entdeckt zu werden, enthalten die meisten Viren spezielle **Tarnroutinen**. Dies kann sich auf die Art und Weise der Infektion oder Manipulation des befallenen Systems auswirken. Die Tarnung kann so weit gehen, dass auf einem befallenen System die Entdeckung eines Virus mit entsprechenden Scannern nicht mehr möglich ist.

6.2 Virenarten

Bootsektorviren

Zu der ältesten Art von Viren, die bis ca. 1995 weit verbreitet waren und seit 2005 aufgrund bootfähiger CDs, DVDs und USB-Sticks wieder im Kommen sind, gehören die sogenannten **Bootsektorviren**. Sie haben ihren Namen von dem „Lebensraum“ erhalten, den sie besetzen. Ein Bootsektorvirus nutzt die Tatsache, dass jeder Computer gestartet werden muss.

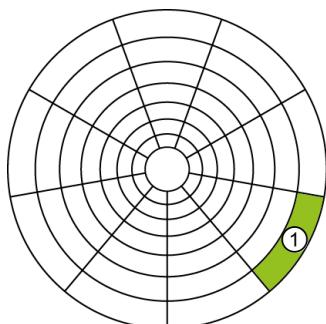
Nach dem Einschalten führt das BIOS zuerst eine Fehlererkennung durch einen Power-on-self-test (POST) und leitet dann den Start des Betriebssystems ein, in dem auf den verfügbaren Datenträgern nach einem Bootsektor gesucht wird. Der Inhalt des Bootsektors wird in den Speicher geladen und ausgeführt. Der Code im Bootsektor enthält weitere Anweisungen, wie und wo die Startdateien des Betriebssystems zu laden sind.

Bootsektorviren nutzen diese Vorgehensweise, um vor dem Start des eigentlichen Betriebssystems vom befallenen Rechner ausgeführt zu werden. Eine weitere Verbreitung dieser Viren erfolgt über die Infektion der Bootsektoren von CDs, DVDs oder USB-Sticks. Werden diese Medien weitergegeben und bei einem Neustart eines Rechners im Laufwerk vergessen, wird der Virus auf dem Datenträger ausgeführt und kann einen neuen Rechner infizieren.

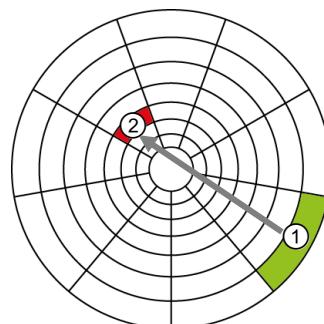
Infektionsmethode

Die Abbildung unten links zeigt schematisch die Festplatte eines virusfreien PCs. Der Bootsektor (MBR = Master Boot Record) des Systems befindet sich auf dem ersten Sektor des Datenträgers und wird grundsätzlich vom BIOS geladen, um den Startvorgang des Betriebssystems einzuleiten.

Wird dieser PC eingeschaltet, so lädt das BIOS den Bootsektorcode ① und startet diesen. Der Code enthält Anweisungen und Informationen, die es erlauben, den Rest des Betriebssystems von den jeweiligen Sektoren der Festplatte zu laden.



Virusfreier PC, Bootsektor an vordefinierter Stelle



Virus infiziert die Festplatte

Im Beispiel wird angenommen, dass der Benutzer dieses Computers ein bootfähiges Medium mit einem Bootsektorvirus erhalten und Dateien hiervon auf seinen Rechner kopiert hat. Das bootfähige Medium wurde im Laufwerk vergessen bzw. nicht abgezogen.

Nach einem Neustart des Rechners, nachdem der POST erfolgreich durchgeführt wurde, sucht das BIOS gemäß der eingestellten Bootreihenfolge nach startbaren Laufwerken. Von dem noch im Laufwerk befindlichen bzw. noch am Rechner angeschlossenen Medium wird der Bootsektor ausgelesen und ausgeführt. In diesem Fall wird der Virus aktiviert.

Die Infektionsroutine des Virus sucht andere installierte Datenträger ab und findet z. B. das C-Laufwerk. Zur Infektion des Laufwerks erstellt der Virus zuerst eine Kopie des originalen Bootsektors ①. Hierzu wird meist eine im Virus fest programmierte Sektoradresse als Ort für die Backup-Kopie gewählt.

Anschließend schreibt der Virus sich selbst an die Stelle des Bootsektors ②. Um den erfolgreichen Infektionsvorgang zu vertuschen, wäre es jetzt für den Virus möglich, eine Fehlermeldung auf dem Bildschirm auszugeben, die besagt, dass ein nicht bootfähiges Medium im Laufwerk liegt. Der Benutzer wird daraufhin den Datenträger entfernen und den Rechner nochmals starten.

Beim folgenden Bootvorgang liest das BIOS den Virus-Sektor und startet diesen. Wiederum wird nach infizierbaren Datenträgern oder Festplatten gesucht, mit dem Unterschied, dass der Virus nun weiß, dass er von einer bootfähigen Festplatte gestartet wurde. Da die Adresse für das Backup des Originalsektors bekannt ist, verzweigt der Virus nach Abarbeitung seines Codes an die Adresse des originalen Bootsektors, und der PC startet wie gewöhnlich.

Speicherresidente Viren

Speicherresident ist ein Virus dann, wenn er nicht wie ein gewöhnliches Programm nach seiner Ausführung beendet wird, sondern im Speicher verbleibt und weiterhin aktiv ist. Viren dieser Art werden manchmal auch **TSR-Viren** genannt („Terminate and Stay Resident“). Ein speicherresidenter Bootsektorvirus würde bei seiner Ausführung zuerst über eine BIOS-Funktion den Wert für den maximal verfügbaren Speicher um den Betrag seiner eigenen Größe reduzieren.

Ist der Virus beispielsweise 1 KB groß, wird ein für die DOS-Architektur geschriebener Bootsektorvirus den Wert für verfügbaren Speicher von 640 KB auf 639 KB reduzieren. Anschließend kopiert sich der Virus in den als „nicht existent“ markierten Arbeitsspeicher.

Damit der Virus aktiv bleiben kann, wird die Software-IRQ-Tabelle (IRQ = Interrupt Request) des PCs manipuliert. Die Software-IRQs stellen grundlegende PC-Funktionalitäten zur Verfügung, auf denen Programmierer auch ohne Vorhandensein eines Betriebssystems aufbauen können. In der IRQ-Tabelle ist zu einer bestimmten IRQ-Nummer (entspricht einer Funktion) die Speicheradresse abgelegt, an der die entsprechende Funktion aufgerufen werden kann.

Der Virus liest einen Wert für einen gewünschten IRQ aus (z. B. den IRQ für Datenträgerzugriffe) und schreibt seine eigene Adresse im Speicher in diese Tabelle. Die ursprüngliche IRQ-Adresse wird im Virus gespeichert.

Anschließend kann sich diese Instanz des Virus beenden; seine gerade angelegte Kopie bleibt im Speicher aktiv und wird von allen anderen Programmen unabsichtlich immer dann aufgerufen, wenn sie die entsprechende IRQ-Funktion benutzen. Der residente Virus hat dann bei seiner Ausführung Gelegenheit, weitere Ziele zu infizieren, Schadensfunktionen auszuführen oder den Funktionsaufruf an die Originalfunktion weiterzuleiten. Im letzteren Fall würde die Anwesenheit eines Virus nicht weiter auffallen, da der PC normal zu arbeiten scheint.

Dateiviren/Linkviren

Diese Virenart ist relativ häufig anzutreffen. Ihre Aufgabe ist das Einschleusen eines Schadcodes in ausführbare Dateien und Bibliotheken des Betriebssystems. Der Schadcode wird je nach Implementierung am Anfang (Infektion durch Prepper) oder am Ende (Infektion durch Appender) der Datei eingefügt und beim Aufruf mit ausgeführt.

In den folgenden Grafiken sind die Daten des Wirtsprogramms dargestellt, wie sie auf der Festplatte gespeichert sind. Bei Dateiviren sind folgende Infektionsmethoden gebräuchlich:

Die **Overwrite-Infektion** ist die am einfachsten zu realisierende Infektionsmethode und wird daher von ungeübten Virenprogrammierern verwendet oder wenn eine Programmiersprache zum Einsatz kommt, die detaillierte Dateimanipulationen nicht zulässt (wie z. B. Batch-Dateien).



Overwrite-Infektion

Der Virus überschreibt die Originaldatei komplett und übernimmt dabei den ursprünglichen Namen. Die resultierende Datei hat die Länge des Viruscodes, trägt aber den Namen der Anwendungsdatei, die bei diesem Vorgang komplett zerstört wurde.

Die in obiger Abbildung gezeigte Methode erhält Teile des Anwendungsprogramms, schreibt sich jedoch über den Anfang der Programmdatei und zerstört diesen somit unwiederbringlich. Will der Anwender das Programm starten, wird der Virus vom Betriebssystem ausgeführt. Ein Start des Anwendungsprogramms kann nicht erfolgen, da dieses zerstört wurde.

Infektion durch Dateianhang

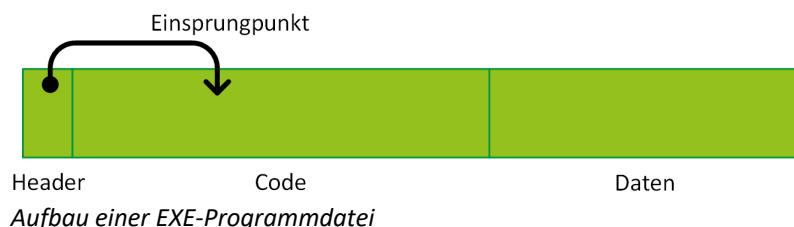
Bei der häufigsten Infektionsmethode der Dateiviren versucht der Virus, sich am Ende der Wirtsdatei anzuhängen. Würde der Viruscode einfach an das Ende des Programms platziert werden, so hätte das bis auf die Änderung der Programmlänge keinen Effekt. Das Programm wird bei einem Aufruf normal ausgeführt und beendet, der Viruscode bleibt inaktiv, da vom Programm aus kein Sprungbefehl in den Viruscode führt.

Der Virus muss durch gezielte Manipulation der Programmdatei zusätzlich dafür sorgen, dass seine Befehle angesprungen werden.

Die meisten Dateiviren infizieren COM-, EXE-, SYS- sowie DLL-Dateien. COM-Dateien (COM = command) stellen ein Abbild des Programms im Arbeitsspeicher dar, das unverändert auf die Festplatte gespeichert wurde, und müssen anders behandelt werden als EXE-Dateien (EXE = Executable), die vor dem Start erst aufbereitet werden.

Da bis auf wenige Ausnahmen EXE-Programmdateien zum Einsatz kommen, werden im Folgenden die Mechanismen anhand dieses Dateityps beschrieben.

Eine EXE-Datei stellt im gewissen Sinne ein noch unfertiges Programm dar. Es enthält zwar sämtliche für den Programmablauf benötigten Anweisungen, aber die im Programm verwendeten Adressen für Verzweigungen, Sprünge oder Zugriffe auf Variablen sind noch nicht endgültig, da die Speicheradressen, an denen das Programm geladen wird, beim Kompilieren des Programms noch nicht bekannt waren. Der Programmlader des Betriebssystems wird beim Starten des Programms die Datei laden, die korrekten Speicheradressen ermitteln und sämtliche relativen Adressangaben im Programm durch die absoluten Adressen im Arbeitsspeicher ersetzen.



Um das Programm zu starten, muss die Kontrolle an das Programm übergeben werden. Dazu ermittelt das Betriebssystem aus dem Header der Programmdatei den sogenannten **Einsprungpunkt**. Dieser zeigt auf den ersten auszuführenden Befehl des Programms.

Ein Dateivirus, der eine EXE-Datei infiziert, hängt sich zuerst an das Ende der betroffenen Datei an ①. Er liest die Einsprungadresse aus dem Header ②. Anschließend ersetzt er diese durch den Beginn seines eigenen Codes ③. Bei einem Programmstart wird nun als Einsprungadresse der Beginn des Virus ermittelt und es werden zuerst dessen Befehle bearbeitet ⑤. Wenn der Viruscode komplett abgearbeitet wurde, so erfolgt am Ende ein Rücksprungbefehl zu der ursprünglich ausgelesenen Einsprungadresse. Das ursprünglich aufgerufene Programm wird nun ausgeführt mit ⑥ und ④.



Makro-/Skriptviren

Viren waren bis dato an ausführbaren Code gebunden, also an Bootsektoren oder Programmdateien. Mit der Einführung von Makro-Programmiersprachen in Anwendungssoftware wurde dieses Paradigma allerdings gebrochen. Makros werden zusammen mit der entsprechenden Dokumentdatei gespeichert. Sie sollten Funktionen enthalten, die dem Benutzer der Anwendungssoftware die Arbeit mit dem Dokument erleichtern, z. B. eine Ausfüllhilfe für Formulare oder das Erstellen eines Seriendokuments nach Eingabe der vom Benutzer gewünschten Informationen.

Microsoft unterstützt in den Makrosprachen seiner Officeprodukte eine Autostart-Funktion, die ein Makro aktiv werden lässt, sobald der Benutzer das entsprechende Dokument öffnet. Innerhalb der Makrosprache werden zusätzlich zahlreiche Funktionen angeboten, die komplexe Operationen wie Dateimanipulationen oder sogar das automatische Versenden von E-Mails stark vereinfachen.

Zum Beispiel kann ein Word-Makrovirus beim Öffnen eines infizierten Word-Dokumentes ohne Wissen des Benutzers sofort aktiv werden, weil seine Makroanweisungen vom Word-Anwendungsprogramm sofort bearbeitet werden. Die Infektionsroutine sucht hier nach weiteren Dokumenten und fügt dort den Virus als neues Autostart-Makro ein.

Wird die E-Mail-Funktionalität von Makrosprachen genutzt, um den Virus automatisch an bekannte Empfänger aus dem Adressbuch des Benutzers zu versenden, ergibt sich ein extrem hohes Verbreitungs- und Schadenspotenzial.

Visual Basic Script Viren

Eine ähnlich starke Verbreitung wie Makroviren erlebten die Skriptviren, die ebenfalls ohne große Programmierkenntnisse erstellt werden können. Im Vergleich zu den herkömmlichen auf Assembler basierten Boot- und Dateiviren kann auch ein Ungeübter durch kurzes Studium eines VBS-Virus-Sourcecodes schnell einen eigenen Virus oder eine Variante entwerfen.

Die beiden abgebildeten Auszüge aus den Sourcecodes bekannter Viren verdeutlichen, wie stark sich die Verständnisforderungen für den Leser beider Codes unterscheiden. Trotz der Kommentare im vorliegenden Assembler Code ist immer noch eine gute Kenntnis von Assembler und Rechnerarchitektur notwendig, um die Arbeitsweise des Virus zu verstehen bzw. auf Basis dieses Codes eine eigene Variante zu entwickeln. Für das Verständnis des kommentierten VBS-Sourcecodes genügen meist nur Englisch- und grundlegende Computerkenntnisse. Die Namen der aufgerufenen Funktionen sind größtenteils selbsterklärend.

```
entervirus:
    xor    ax,ax
    mov    ds,ax
    cli
    mov    ss,ax
    mov    ax,7C00h           ; set stack to just below
    mov    sp,ax              ; virus load point
    sti
    push   ds                ; save 0:7C00h on stack for
    push   ax                ; later retf
    mov    ax,ds:[13h*4]
    mov    word ptr ds:[7C00h+offset oldint13h],ax
    mov    ax,ds:[13h*4+2]
    mov    word ptr ds:[7C00h+offset oldint13h+2],ax
    mov    ax,ds:[413h]         ; memory size in K
    dec    ax                ; 1024 K
    dec    ax
    mov    ds:[413h],ax        ; move new value in
    mov    cl,6
    shl    ax,cl              ; ax = paragraphs of memory
    mov    es,ax              ; next line sets seg of jmp
    mov    word ptr ds:[7C00h+2+offset highmemjmp],ax
    mov    ax,offset int13h
    mov    ds:[13h*4],ax
    mov    ds:[13h*4+2],es
    mov    cx,offset partitioninfo
    mov    si,7C00h
    xor    di,di
```

Auszug aus dem Michelangelo-Sourcecode (Assembler)

```

Sub Main()
    On Error Resume Next
    Dim Wscr,rr
    Set Wscr=CreateObject("WScript.Shell")
    'check the time out value for WSH
    rr=Wscr.RegRead("HKEY_CURRENT_USER\Software\Microsoft\Windows Scripting Host\Settings\Timeout")
    If (rr>=1) Then
        ' Set script time out to infinity
        Wscr.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\Windows Scripting Host\Settings\Timeout",
    End If
    'Create three copies of the script in the windows, system32 and temp folders
    Set dirwin = fso.GetSpecialFolder(0)
    Set dirsystem = fso.GetSpecialFolder(1)
    Set dirstemp = fso.GetSpecialFolder(2)
    Set c = fso.GetFile(WScript.ScriptFullName)
    c.Copy(dirsystem &"\MSKernel32.vbs")
    c.Copy(dirwin &"\Win32DLL.vbs")
    c.Copy(dirsystem &"\LOVE-LETTER-FOR-YOU.TXT.vbs")
    'Set IE default page to 1 of four locations that downloads an executable.
    'If the executable has already been downloaded set it to run at the next login
    and set IE's start page to be
    blank
    regrun()
    'create an html file that possibly runs an activex component and runs one of
    the copies of the script
    html()
    'Resend script to people in the WAB
    spreadtoemail()
    'overwrite a number of file types with the script if the files are not already scripts
    'create a script file with the same name
    with vbs extention and
    'delete the original file mirc client have a script added to send the html file
    'created earlier to a channel
    listadrv()
End Sub

```

Auszug aus dem ILOVEYOU-Sourcecode (VBS)

JavaScript-Viren

JavaScript, eine einfache Programmiersprache, die die Interaktion mit Webseiten erlaubt, kann auch für die Verbreitung von Schadsoftware missbraucht werden. Microsoft verwendet für die Interaktion mit Webseiten JScript. JScript unterstützt nur die Befehle des Internet Explorers. In JavaScript/JScript gibt es einige Kommandos, die prinzipiell Sicherheitsprobleme darstellen können, da diese auf ein ActiveX-Control im Internet Explorer zugreifen können, sofern dieses aktiviert worden ist. Das gilt entsprechend für Mozilla/Opera, wenn JavaScript ein Java-Applet startet.

6.3 Tarnmechanismen von Viren

Warum Tarnung?

Virenprogrammierer wollen meist, dass sich Viren möglichst unbemerkt verbreiten und ihre Schadwirkung erst sehr spät oder gar nicht auffällt. Bei Viren, die z. B. Code für die Bildung von Botnetzen enthalten, wäre dies fatal. Schließlich soll das Botnetz so groß wie möglich werden, damit sich somit ein größerer Gewinn bei der Vermietung oder dem Verkauf erzielen lässt.

Zu den Zeiten einfacher Dateiviren war das Erkennen einer Virusinfektion relativ leicht: Die befallenen Programme verlängerten sich exakt um die Größe des Viruscodes.

Nach Erscheinen der ersten Viren wurde spezielle Antivirussoftware entwickelt, die die Viren aufspüren und, möglichst ohne Datenverlust vom Computer entfernen sollte. Eine alte Methode war, die Längen sämtlicher Programmdateien in einer Datenbank zu speichern und die gespeicherten Werte regelmäßig mit den aktuellen Werten zu vergleichen. Wurde ein Längenzuwachs festgestellt, gab das entsprechende Programm Alarm.

Relativ schnell hat sich beim Virensuchen die sogenannte **Signaturprüfung** durchgesetzt. Hierbei werden die typischen Merkmale aller bekannten Computerviren zu einer **Signaturdatenbank** zusammengefasst. Der Virensucher überprüft nun, ob in den Dateien oder im Arbeitsspeicher ein zu einer Signatur passendes Bytemuster gefunden werden kann.

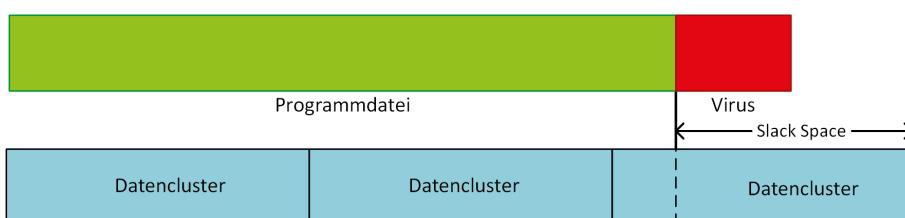
Dies war der Beginn eines Wettkampfs, in dem Virenprogrammierer neue Tarnmechanismen entwickelten, um einer Entdeckung durch Scanner zu entgehen, und die Hersteller von Antivirengeschäftssoftware versuchten, geeignete Gegenmaßnahmen zu finden.

Slackviren

Slackviren nutzen die Tatsache aus, dass ein Dateisystem auf dem Datenträger für eine Datei mehr Platz als belegt markiert, als die Größe der Datei eigentlich erfordern würde. Dies ist darauf zurückzuführen, dass auf Datenträgern die kleinste einzeln adressierbare Speichereinheit (Sektor) meist 512 Bytes ist. NTFS fasst diese Sektoren zu Clustern zusammen. Die Clustergröße (4, 8, 16, 32 ... KB) ist dabei abhängig von der Plattenkapazität und definiert somit den kleinsten Speicherbereich für die Daten.

Ist eine Festplatte mit NTFS formatiert und in 32-KB-Cluster unterteilt, so belegt eine 40.000 Bytes große Datei auf der Festplatte 2 Cluster und somit 65.536 Bytes. Die verlorenen 25.536 Bytes werden **Slackspace** genannt.

Ein Slackvirus versucht, seine Anwesenheit dadurch zu verbergen, dass er bei der Infektion nur Programmdateien sucht, deren Slackspace mindestens der Größe des Virus entspricht. Hat der Virus eine solche Datei gefunden, so infiziert er das Programm und schreibt sich in den Slackspace.



Ein Slackvirus belegt den verbleibenden Speicher im letzten Datencluster

Anschließend wird meist noch die Dateizuordnungstabelle (Journal) des Datenträgers so modifiziert, dass wieder die Originalgröße der Datei angezeigt wird. Für den Virus besteht hier keine Gefahr, vom Betriebssystem überschrieben zu werden, da der Cluster, in dem er sich befindet, noch ordnungsgemäß durch das Programm als belegt markiert ist.

Eine falsche Einschätzung der Slackgröße durch den Virus und eine Infektion kann hier zur Folge haben, dass nachfolgende Datencluster durch Überschreiben korrumptiert oder Teile des Virus überschrieben werden. Der erste Fall führt zu Datenverlust, während im zweiten Fall das infizierte Programm nicht mehr lauffähig ist.

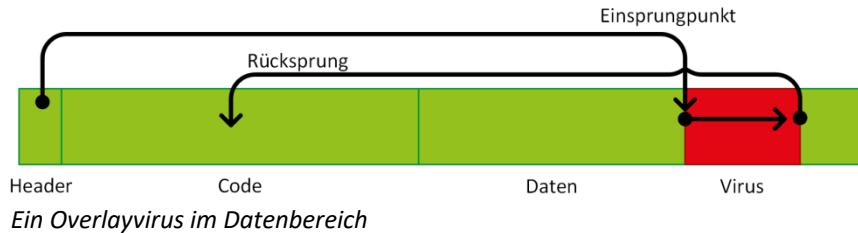
Overlayviren

Viele Virensucher nutzen eine Basiseigenschaft der Viren: Sie hängen sich an das Programmende und sorgen durch manipulierte Einsprungadressen und den Rücksprung in das Wirtsprogramm am Ende des Viruscodes für die Ausführung bei einem Programmstart.

Anstatt die komplette Programmdatei nach einem Viren-Muster zu untersuchen, prüfen einige Scanner jeweils nur das Ende der Programmdateien auf die Signatur von bekannten Viren. Diese Scanner sind beim Überprüfen eines kompletten Datenträgers deutlich schneller als Scanner, die alle Daten überprüfen.

Overlayviren versuchen, sich vor derart einfachen Suchmethoden zu verbergen, indem sie sich in die Programmdatei hineinschreiben. Vorzugsweise wird hierfür innerhalb des Programms nach unbenutztem Platz gesucht. Dies können z. B. große Datenbereiche für globale Variablen sein.

Die Länge der Programmdatei ändert sich hierbei nicht. Virensucher, die nur das Dateiende von Programmen untersuchen, können in so einem Fall den Virus nicht entdecken.



Wird ein derart infiziertes Programm gestartet, gelangt zuerst der Virus zur Ausführung und kann seine Schadensaktionen durchführen bzw. sich als speicherresidentes Programm installieren. Wird anschließend das infizierte Programm fortgeführt, ist es für den Virus bedeutungslos, wenn es den in seinem Datenbereich versteckten Virus mit Daten überschreibt, da eine Kopie des Virus bereits im Arbeitsspeicher aktiv ist.

Problematisch für den Virus ist jedoch, einen genügend großen zusammenhängenden Bereich innerhalb des Programms zu finden. Fortgeschrittene Viren teilen sich selbst in mehrere Programmteile auf und schreiben diese an verschiedene Stellen im Datenbereich. Damit der Virus dennoch korrekt ausgeführt wird, werden am Ende eines jeden Teilssegments Sprunganweisungen eingefügt.

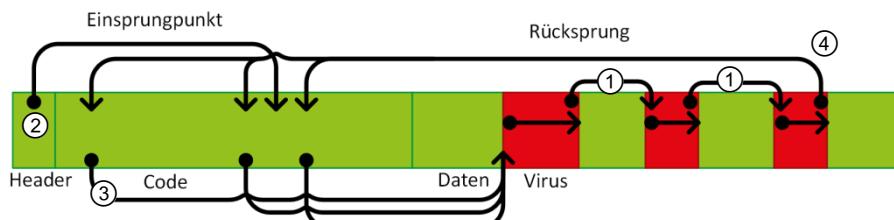
Virensucher wurden für das Auffinden von Overlayviren so modifiziert, dass sie den Einsprungpunkt des Programms ermitteln und den dort befindlichen Code mit den vorhandenen Signaturen überprüfen. Ein besonders trickreicher Virus vermeidet die Manipulation des Einsprungpunktes und manipuliert Funktionsaufrufe innerhalb des infizierten Programms.

Ein Virus, der in der Lage ist, sich in mehrere Segmente aufzuteilen und diese in kleine, verfügbare Speicherbereiche innerhalb des Programms schreibt, wird mit einer höheren Wahrscheinlichkeit eine Programmdatei finden, die erfolgreich infiziert werden kann.

Die folgende Abbildung zeigt einen derartigen Virus, der zusätzlich auf die einfache Manipulation des Einsprungpunktes verzichtet, um schwerer entdeckt werden zu können. Bei der Infektion hat der Virus drei Bereiche identifiziert, die jeweils groß genug sind, um seine Segmente darin platzieren zu können.

Damit der Virus als ganzes Programm zusammenhängend ausgeführt wird, werden die Segmente durch Sprungbefehle verbunden ①.

Der im Header der EXE-Datei angegebene Einsprungpunkt wurde nicht verändert und zeigt auf den Start des Wirtsprogramms ②. Der Virus sucht bei seiner Infektion nach Funktionsaufrufen innerhalb des Wirtsprogramms und ändert diese so ab, dass sie nicht die Unterfunktion, sondern den Virus aufrufen ③. Im Beispiel hat der Virus drei Funktionsaufrufe manipuliert.



Overlayvirus mit mehreren Segmenten und modifizierten Funktionsaufrufen

Gelangt der Virus zum Ende seiner Ausführung, wird die vom Programm eigentlich beabsichtigte Funktion ausgeführt. Nach deren Ende erfolgt der Rücksprung zum jeweiligen Herkunftsplatz ④.

Ein Virensucher, der nur den durch die Einsprungadresse angegebenen Code untersucht, wird dort nur die Befehle des Hauptprogramms vorfinden und darum den Virus übersehen. Diese Art der Infektion bedingt zusätzlich, dass der Virus nicht zwangsläufig vor dem Start des befallenen Programms ausgeführt wird, sondern nur dann, wenn im Programm eine entsprechend manipulierte Unterfunktion gestartet werden soll.

Manipuliert der Virus nur selten benötigte Funktionen im Programm, kann es durchaus sein, dass der Virus sehr lange Zeit im Programm schläft, nur um dann plötzlich aktiv zu werden, wenn die manipulierte Funktion doch einmal benutzt wird.

Selbstverschlüsselnde Viren

Um der Entdeckung durch Signaturprüfungen zu entgehen, schreiben **selbstverschlüsselnde Viren** (manchmal auch nur „**verschlüsselnde Viren**“ genannt) ihren Code nicht im Klartext in das Wirtsprogramm, sondern verschlüsseln diesen vorher. Der Schlüssel wird bei jeder Infektion als Zufallszahl neu erzeugt.

Damit der Virus lauffähig ist, wird bei der Infektion eine Entschlüsselungsroutine am Anfang eingebracht, in der der für diese Infektion benutzte Schlüssel hinterlegt worden ist.

Wird der Viruscode in einem Programm aufgerufen, so dechiffriert die Entschlüsselungsroutine den Virus zuerst und fährt dann mit den gerade entschlüsselten Anweisungen fort.

0000	Original „Verschlüsselnder Virus“	Urvirus Klartextvariante liegt nur dem Autor vor.
6b42	2978q pō t ouzfgalg l34 hky	
77a5	85tb oiluhluzfdf 0+980z IU€	Jede neue Infektion wird mit einem neuen Zufallsschlüssel verschlüsselt. Die Entschlüsselungsroutine am Anfang erhält diesen Schlüssel.
56bd	p764tvhxlukgzdflo7duih	

Verschlüsselnder Virus, keine Version ist identisch mit einer anderen

Den unverschlüsselten Virus in Dateiform besitzt auf diese Weise nur der Autor des Virus. Je nachdem, wie viele verschiedene Schlüssel der benutzte kryptografische Algorithmus unterstützt, kann es von ein und demselben Virus eine unüberschaubare Anzahl an Erscheinungsformen in einer Datei geben.

„Gepanzerte“ Viren, Armored Code

Eine Erweiterung des Verschlüsselungskonzepts stellen **Armored Viren** dar, was wörtlich „gepanzert“ bedeutet. Da auch Viren nur Programme sind, können sie wie gewöhnliche Programme analysiert werden; d. h., ein Antivirenexperte ist in der Lage, mit entsprechenden Editoren, Disassemblern und Debuggern den Assemblercode und somit die Arbeitsweise des Virus zu analysieren.

Ein einfaches Disassemblieren würde im Falle eines verschlüsselnden Virus schon nicht mehr genügen, da die Virus-Anweisungen nicht im Klartext in der Datei stehen. Mit einem Debugger kann man jedoch den auszuführenden Code sichtbar machen, sobald er im Arbeitsspeicher entschlüsselt wurde.

Armored Viren sind speziell programmiert worden, um auch diese Analyse zu erschweren. Beispielsweise erkennt ein Virus die Existenz eines Debugger-Programms im Speicher und beendet sich gegebenenfalls sofort selbst. Im Falle von verschlüsselnden Viren bedeutet das, dass der Dekoder nicht am Beginn des Viruscodes steht und den gesamten Virus entschlüsselt. Ein gegen Debugging geschützter Virus entschlüsselt im Arbeitsspeicher immer nur die Codeteile, die als Nächstes ausgeführt werden sollen. Er führt diese aus und verschlüsselt sie danach sofort wieder. Dies erschwert die Arbeit für Analysten deutlich.

Herkömmliche Signaturanalyse ist bei verschlüsselnden Viren nutzlos, da der Virenkörper sich bei jeder Neuinfektion ändert. Eine Schwachstelle ist jedoch die Entschlüsselungsroutine des Virus. Er muss unverschlüsselt im Programm stehen, da sonst der Virus nicht entschlüsselt werden kann. Signaturen für verschlüsselnde Viren werden also über den Dekodierteil gebildet und nicht über den gesamten Virus.

Polymorphe Viren

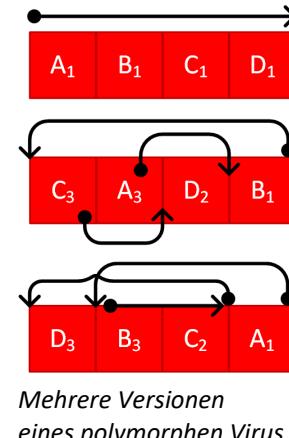
Polymorphe (vielgestaltige) Viren gehen noch einen Schritt weiter als verschlüsselnde. Wo ein verschlüsselnder Virus sein äußeres Erscheinungsbild dadurch ändert, dass er bei einer Infektion jeweils einen neuen Schlüssel zur Verschlüsselung benutzt, ändert ein polymorpher Virus seinen Code von Infektion zu Infektion, behält aber dieselbe Funktionalität bei.

Ein polymorpher Virus enthält gewissermaßen eine Datenbank mit alternativen Befehlsfolgen für seinen Code. Bei einer Neuinfektion wird der Virus nicht 1:1 in das neue Wirtsprogramm geschrieben, sondern mit einer Zufallsfunktion wird für jedes Segment eine neue Alternative gewählt. Daraus ergibt sich eine schier unüberschaubare Anzahl von alternativen Erscheinungsformen.

Kombiniert mit der Möglichkeit, die verschiedenen Segmente in unterschiedlicher Reihenfolge anzurichten und die logische Reihenfolge durch entsprechende Sprunganweisungen wiederherzustellen, steht die Virensuche mit reiner Signaturanalyse vor einem Problem.

In der rechten Abbildung wurde angenommen, dass ein Virus aus 4 Segmenten besteht, für die jeweils drei funktionsgleiche Alternativen existieren. Durch diese Kombinationen hat der Virus schon 81 Erscheinungsformen und somit 81 verschiedene Signaturen. Die Grafik berücksichtigt zusätzlich noch die Möglichkeit, die einzelnen Segmente im Wirtsprogramm in vertauschter Reihenfolge zu speichern und den korrekten Programmablauf durch Sprunganweisungen sicherzustellen. Dies steigert die Anzahl der Erscheinungsformen noch weiter.

Das Erstellen von polymorphen Viren ist extrem komplex, da der Viruscode nicht nur einmal geschrieben werden muss, sondern der Autor auch mehrere alternative Versionen für die Codeabschnitte erstellen muss. Anschließend muss der gesamte Virus mit allen Alternativen in eine Form gebracht werden, die bei der Infektion die Neuzusammensetzung sinnvoll erstellt.



Neue Suchmethoden der VirensScanner

Um Viren trotz der bisher vorgestellten Tarnmethoden noch erkennen zu können, wurden neue Ansätze zur Suche entwickelt. Herkömmliche Signaturen wurden um Signaturen mit Wildcards erweitert. Ähnlich, wie man in einem Textverarbeitungsprogramm mit der Suchmaske *Dampf*gesellschaft* alle Wörter findet, die mit *Dampf* beginnen und mit *-gesellschaft* enden, kann mit modernen Scanner-Engines nach Bytemustern gesucht werden, die innerhalb definierbarer Bereiche mit beliebigen Bytes zu finden sind.

Heuristische Methoden (von griechisch „heuriskein“: finden, Erkenntnis gewinnen) sind von Signaturen unabhängig und analysieren, ob ein Programm virentypische Aktionen ausführen würde, wenn es gestartet werden würde. Einsprungpunkte am Dateiende und Rücksprünge in die Dateimitte oder Schreibzugriffe auf andere EXE-Dateien wären Hinweise auf einen Virenbefall. Heuristiken haben aber auch eine höhere Wahrscheinlichkeit für Fehlalarme.

Stealth-Viren

Stealth- oder auch Tarnkappenviren besitzen die umfassendsten und effektivsten Methoden, eine Entdeckung zu verhindern. Dabei nutzen sie die Methoden **Simple Stealthing** (einfache Tarnung), **Read Stealthing** (Tarnung bei Lesezugriff) und **Size Stealthing** (Tarnung bei Dateigrößenänderungen).

Für eine Installation als speicherresidentes Programm lädt sich der Virus in den Arbeitsspeicher und manipuliert den IRQ-Vektor einer gewünschten Funktion derart, dass der Virus statt der Funktion aufgerufen wird. Nachdem der Virus seine Aktionen durchgeführt hat, ruft er die ursprüngliche Funktion auf, deren Adresse er gespeichert hat.

Ein Stealth-Virus leitet bei der Infektion beispielsweise die IRQ-Vektoren für Datenträgerzugriffe und Speicherzugriffe auf sich um. Deshalb wird er bei derartigen Aufrufen aktiv. Seine Stealth-Eigenschaften bezieht der Virus daher, dass er nach Ende seiner Aktionen nicht einfach die ursprünglich angefragte Funktion aufruft, sondern vorher prüft, **was** mit dem Funktionsaufruf bewirkt werden soll.

Handelt es sich um einen normalen Datenträgerzugriff, werden die Aktionen unverändert ausgeführt. Fatal ist jedoch, wenn der Benutzer einen VirensScanner startet und dieser den Computer prüft:

Der VirensScanner durchsucht zuerst den Arbeitsspeicher nach Viren. Sämtliche Leseanfragen, die der Scanner an die Hardware stellt, werden aber zuerst vom Virus untersucht. Handelt es sich um einen normalen Speicherbereich, reicht der Virus diesen Lesezugriff weiter und der Scanner „sieht“ den normalen Arbeitsspeicher.

Will der VirensScanner den Speicherbereich untersuchen, in dem der Virus sich selbst befindet, gaukelt der Virus dem Scanner leeren Speicher vor oder leitet die Suchanfrage ohne Wissen des Scanners auf andere Speicherbereiche um.

Prüft der Scanner anschließend die Festplatte, verfährt der Virus ähnlich. Solange Sektoren oder Dateien gelesen werden sollen, die vom Virus nicht beeinflusst wurden, werden die Antworten nicht manipuliert. Soll aber z. B. der infizierte Bootsektor gelesen werden, leitet der Virus die Leseanfrage auf die Originalkopie des Bootsektors um. Wird eine EXE-Datei gelesen, die vom Virus infiziert ist, gibt der Virus den Inhalt der EXE-Datei an den VirussScanner weiter – allerdings nur nachdem er vorher seinen eigenen Code aus dem Dateiinhalt entfernt hat.

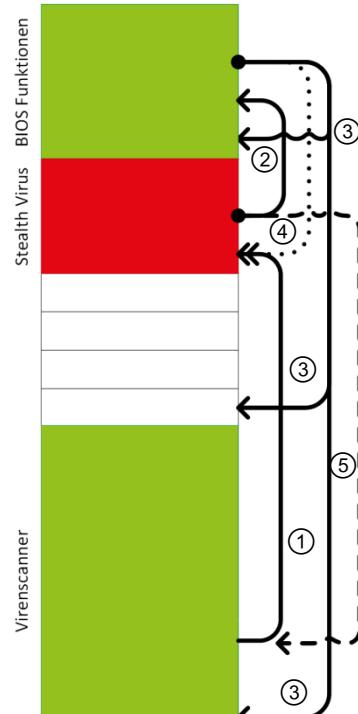
Diese Vorgehensweise macht es bei geschickter Programmierung des Virus im Prinzip unmöglich, den Virus zu entdecken. Wurde ein Stealth-Virus gestartet, bevor ein entsprechendes Suchprogramm zum Einsatz kommt, ist das Suchprogramm auf die gegebenenfalls manipulierten Input/Output-Funktionen des bereits infizierten Rechners angewiesen und erhält im Zweifelsfall falsche Daten.

In der Grafik rechts befindet sich ein Stealthvirus im Arbeitsspeicher des Rechners, und ein VirensScanner wurde geladen. Dieser muss eine Funktion im BIOS aufrufen, um die Inhalte von verschiedenen Speicheradressen untersuchen zu können. Der Virus hat allerdings die Adresse für diese Funktionsaufrufe so geändert, dass der Scanner jedes Mal den Viruscode anspringt ①.

Der Virus untersucht den Funktionsaufruf und leitet diesen an das BIOS weiter, wenn Speicherbereiche untersucht werden sollen, in denen der Virus selbst nicht anzutreffen ist ②.

Die Funktion im BIOS liest die indirekt vom VirensScanner angeforderten Speicheradressen und liefert das Ergebnis an den Scanner zurück. Auf diese Weise werden unverdächtige Speicheradressen überprüft ③.

Die Anfrage des VirensScanners, den Speicherinhalt an der Adresse des Viruscodes auszugeben, wird nicht an das BIOS weitergeleitet ④. Diese Anfrage beantwortet der Virus in diesem Beispiel selbst, indem er beispielsweise eine Anzahl Nullen an den Scanner zurückgibt und so leeren Speicher vortäuscht ⑤.



Stealth-Virus im Speicher
modifiziert Funktionsaufrufe



Aus diesem Grund ist es extrem wichtig für jedes Computersystem, einen schreibgeschützten und garantiert virenfreien Datenträger zu besitzen, von dem im Zweifelsfall das System gestartet und untersucht werden kann.

Wird der Computer von einem sauberen Datenträger gestartet und befindet sich auf dem Datenträger ebenfalls eine saubere Kopie des Virenscanners, so kann von dort aus das System untersucht und auch ein Stealth-Virus wie jeder andere Virus aufgespürt werden.

Moderne Stealth-Viren müssen nicht mehr unbedingt auf Manipulationen der Software-IRQs zurückgreifen. Eine Installation des Virus-Prozesses als Systemdienst oder Gerätetreiber an entsprechenden Stellen im Betriebssystem kann denselben Effekt haben. Erlangt der Virus Kontrolle über den Datenfluss im Rechner, so kann er ihn gezielt ändern oder umleiten, um seine Entdeckung zu verhindern.

6.4 Würmer

Core Wars

Im Gegensatz zu Viren, die zu ihrer Verbreitung eine Interaktion des Benutzers benötigen (z. B. Starten eines Programms), sind Würmer nicht auf einen Benutzer angewiesen. Würmer führen ein „Eigenleben“ in Rechnernetzen und **vermehren sich selbstständig**.

Die Grundidee für dieses Konzept geht auf das Computerspiel Core Wars zurück, was so viel wie „Krieg der Kerne“ bedeutet. Das aus den 70er-Jahren stammende Spiel beruht darauf, ein Programm zu schreiben, das gegen andere Programme in einer „Arena“ (dem Computer) antritt. Ziel ist es, dem gegnerischen Programm möglichst viel Rechenzeit zu entziehen oder es zu zerstören.

Robert T. Morris, ein Wissenschaftler des National Computer Security Center, der bis 1986 bei AT&T beschäftigt und an der Entwicklung des Betriebssystems UNIX beteiligt war, schrieb 1982 das Siegerprogramm für den Core-Wars-Vorgänger „Darwin“. Morris schrieb damals schon mehrere Artikel über die mangelnde Sicherheit von UNIX.

Der Morris-Wurm

1988 entwickelte Robert T. Morris ein Programm, das mehrere Lücken in UNIX-Betriebssystemen derart ausnutzen konnte, dass es selbstständig seinen Sourcecode auf das Zielsystem übertragen, dort kompilieren und starten konnte. Der erste Internet-Wurm war entstanden.

Der von Morris in Umlauf gebrachte Wurm legte innerhalb kurzer Zeit fast das gesamte bis dahin bekannte Internet lahm. Ein Programmierfehler sorgte dafür, dass der Wurm auf einem befallenen Rechner nicht in 1 von 15 Fällen einen neuen Prozess startete, sondern in 14 von 15 Fällen.

Dies führte zu einer sehr schnellen, anfangs quasi exponentiellen Ausbreitung des Wurms auf andere Systeme und zur Überlastung der betroffenen Rechner und Netzwerke.

6.5 Trojaner

Versteckte Funktionen

Trojaner sind keine Viren im eigentlichen Sinn. Auf den ersten Blick kann es sich bei Trojanern um ganz normale Software handeln. Abgesehen von der Hauptfunktion enthält ein Trojaner (in Anlehnung an trojanisches Pferd aus der Sage) allerdings auch eine dem Benutzer nicht bekannte, unerwünschte Zusatzfunktion.

Welcher Zusatzfunktion das ist, hängt von den Absichten und dem Ideenreichtum des Programmierers ab. So könnte ein als Texteditor getarnter Trojaner an einem bestimmten Datum wichtige Systemdateien durch Überschreiben vernichten oder die Festplatte nach interessanten Informationen durchsuchen und diese versteckt an den Autor des Programms zurücksenden.

Trojaner kommen in vielen Formen vor. Mitunter werden sie als Free- oder Sharewareprogramme zum kostenlosen Download angeboten. Der Autor will ja, dass sich sein Trojaner möglichst großflächig verbreitet. Aber auch kommerzielle Software, deren Lizenzcode z. B. durch einen Hacker gecrackt wurde, kann Trojaner enthalten. Oft werden über Trojaner zusätzlich **Backdoors** (Hintertüren) geöffnet, über die ein Angreifer Zugriff auf das System erlangen und somit z. B. ein **Botnetz** kreieren kann.

Eine spezielle Form von Trojaner ist **Ransomware**. Dieser Krypto- bzw. Erpressungstrojaner verhindert den Zugriff auf Daten und Systeme, indem er Bereiche oder das ganze System verschlüsselt. Eine Freigabe der Daten erfolgt nur gegen Zahlung eines Lösegeldes (engl. ransom). Der Sicherheitsspezialist SonicWall stellte fest, dass die Zahl der Ransomware-Angriffe in den ersten sechs Monaten des Jahres 2018 weltweit verstärkt angestiegen sind. Seit Jahresbeginn zeichnete das Unternehmen 181,5 Millionen Ransomware-Angriffe auf. Dies entspricht einer Steigerung von 229 Prozent im Vergleich zum gleichen Zeitraum des Jahres 2017.

Ransomware wird hauptsächlich über Spam-Mails, Exploit-Kits, Drive-by-Exploits, Würmer bzw. über ungesicherte Fernwartungszugänge verteilt. Nachfolgend ein Auszug der gängigsten Ransomwaretypen (Stand 11/2018):

- | | | |
|---------------------------|------------|--------------|
| ✓ NotPetya/Petra | ✓ GandCrab | ✓ CTB-Locker |
| ✓ Locky | ✓ WannaCry | ✓ Jaff |
| ✓ CryptoWall/CryptoLocker | ✓ Crysis | ✓ Bad Rabbit |
| ✓ HydraCrypt | ✓ Cerber | ✓ GoldenEye |

Spyware

Zu den Untergattungen der Trojaner zählt **Spyware**. Diese sammelt unautorisiert Daten über den Computer oder das Benutzerverhalten des Users und sendet diese an den Autor zurück. Ein deutscher Fall von Spyware war ein kostenloses Tool, das den Umgang mit T-Online erleichtern sollte. Bei der kostenlosen elektronischen Registrierung schickte das Tool allerdings auch die Zugangsdaten des Benutzers an die Autoren des Programms.

Auch werbefinanzierte kostenlose Programme sind in Mode, die innerhalb ihres Anwendungsfensters Werbebanner einblenden. Einige Hersteller dieser speziellen Banner-Komponenten beschränken sich nicht nur darauf, regelmäßig neue Banner von einem entsprechenden Internetserver herunterzuladen, sondern vielmehr auch komplette Benutzerstatistiken an den Werbetreibenden zurückzusenden. Diese Statistiken enthalten dann detailliert das Surfverhalten des Benutzers, eine komplette Liste aller jeweils besuchten Webseiten usw. (vgl. Kapitel 7 für eine detaillierte Darstellung von Spyware).

Dialer

Als Trojaner kann auch das speziell in Deutschland vorhandene Problem der 0900-Dialer gesehen werden. Diese Programme versprechen meist Zugang zu interessanten Informationen oder geben vor, Internetbeschleuniger zu sein.

Nach ihrem Start ändern solche Programme allerdings den jeweiligen Internetzugang auf einen Zugang über eine teure Servicenummer. Seitdem Nummern mit 0900-Vorwahlen zunehmend gesperrt werden oder von vorneherein verdächtig erscheinen, weichen die Hersteller von Dialern auch auf Satellitentelefonnummern oder internationale Nummern aus kleinen Inselstaaten aus. Nach einer Boomphase vor einigen Jahren sind Versuche, mit illegalen Dialern Geld zu verdienen, zurückgegangen.

Diese Dialer funktionieren nicht über DSL-Verbindungen, sondern nur über einen Internetzugang via Satellit mit einem Telefon-Rückkanal oder einer zusätzlichen analogen/digitalen Faxverbindung am Computer.

Fernzugriff

Zahlreiche Trojaner öffnen eine sogenannte **Backdoor** und dienen damit als Fernsteuerungssoftware. Einmal installiert, geben sie dem Autor des Programms die Kontrolle über den befallenen Rechner und erlauben den Zugriff auf Dateien, Registry und eingeschränkt auch auf die Hardware des Computers – als ob der Autor des Trojaners oder derjenige, der die „Rechte“ an der Fernsteuerung des Rechners erworben hat, selbst den infizierten PC nutzen würde.

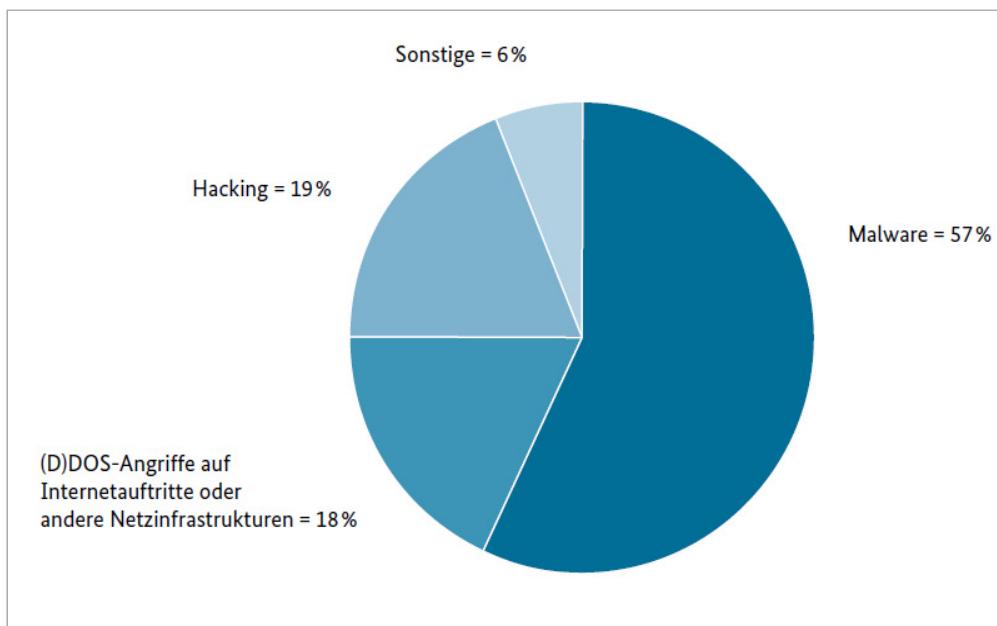
In letzter Zeit ist der Zweig der Computerkriminalität stark im Wachsen begriffen, bei dem großflächig PCs mithilfe von Spyware, Viren, Trojanern und ähnlichen Programmen unter Kontrolle gebracht werden, um sie wie oben beschrieben für die kommerzielle Ausnutzung innerhalb eines Botnetzes zu missbrauchen.

6.6 Adware und PUA

Anders als Viren, Trojaner und Würmer stellen potenziell unerwünschte Anwendungen, sogenannte PUA (Potentially Unwanted Applications), primär keine direkte Gefahr für das System dar. Sie werden auch als PUP (Potentially Unwanted Programs) deklariert. Unübersichtliche Standardinstallationsprozesse verleiten den Anwender mitunter zur Aktivierung von Programmfunctionen, die er überhaupt nicht benötigt. PUA werden bei der Installation von Freeware häufig als Toolbars, Browser-Plug-ins oder Werbebanner aktiviert. PUA, die Produktwerbung anzeigen, werden auch als **Adware** bezeichnet. Download-Manager auf diversen Webseiten können berechtigterweise auch als PUA angesehen werden, da diese mitunter werbefinanziert sind und u. U. andere „nützliche“ Applikationen nachladen wollen. Deshalb sollten Sie Applikationen immer von den Webseiten der Hersteller herunterladen.

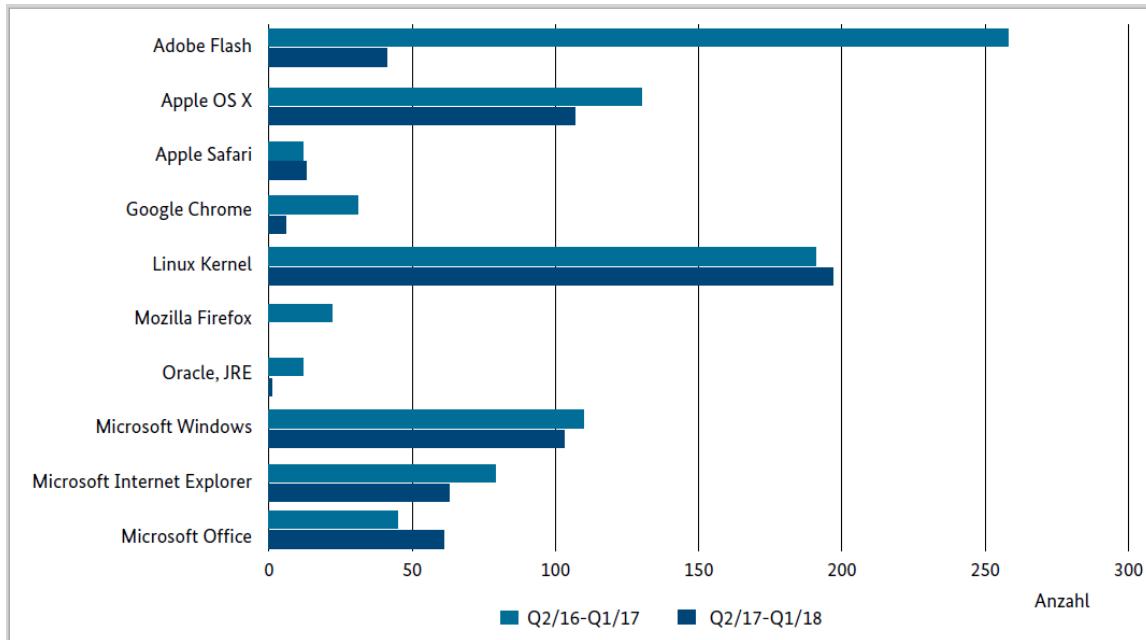
6.7 Tendenzen und Ausblick

Die für das Jahr 2018 veröffentlichten Security Bulletins von Sicherheitsfirmen und dem BSI zeigen eine weitere Fortschreibung der Bedrohungslage auf. Bei den bekannten 800 Millionen Schadprogrammen kommen täglich etwa 390 Tausend neue Modifizierungen hinzu. Auch die Angriffsvektoren für die Einschleusung von Malware werden umfangreicher. Früher waren Betriebssysteme, Browser und JavaScript das vorrangige Einfallstor für Schadprogramme. Aktuell und in Zukunft sind es vorrangig die Bereiche SmartHome, Industrial Control System (ICS), Internet of Things (IoT), SmartCards, Überwachungssysteme und E-Mail-Verschlüsselung. Die meisten Cyberangriffe wurden 2018 hauptsächlich durch die Einschleusung von Malware initiiert. Dieser Trend wird sich auch in den folgenden Jahren kontinuierlich fortsetzen.



Arten von Cyber-Angriffen (Quelle: BSI, Die Lage der IT-Sicherheit in Deutschland 2018)

Ein nachgeordneter Trend sind Hacking- und (D)DOS-Attacken. Diese basieren u. a. auf technischen Schwachstellen in der Softwareimplementierung (Common Vulnerabilities and Exposures, CVE). Diese werden auch 2019 ihre Wichtigkeit bei Microsoft Windows, macOS und beim Linux-Kernel behalten. Die nachfolgende Tabelle skizziert dies.



Kritische Softwareschwachstellen (CVE) (Quelle: BSI, Die Lage der IT-Sicherheit in Deutschland 2018)

Zu Beginn des Jahres 2018 wurde von Sicherheitsforschern vom Google Project Zero und der Technischen Universität Graz eine neue Angriffsklasse auf Central-Processing-Unit (CPU)-Architekturen aufgedeckt. Diese Sicherheitslücken firmieren unter dem Namen **Meltdown** und **Spectre**. Die entdeckten Schwachstellen ermöglichen es Angreifern, geschützte Speicherbereiche eines Systems zu kompromittieren. So können Informationen wie Passwörter, RSA-Schlüssel oder E-Mails mittels des Spectre-Angriffs aus abgeschotteten Speicherbereichen anderer Programme und mittels des Meltdown-Angriffs aus dem abgeschotteten Speicherbereich des Betriebssystems ausgelesen werden. Die Ursache hierfür sind CPU-Designfehler, u. a. der Hersteller Intel, AMD und ARM. Diese Schwachstellen können durch Firmwareupdates nicht vollständig beseitigt werden und bleiben für einen nicht eingrenzbaren Zeitraum ein Risiko. Abhilfe schafft nur die Umrüstung der CPUs (Produktionsreihe ab 2019), welches jedoch nicht als realistisch anzusehen ist.

6.8 Übung

Fragen zu Viren

Übungsdatei: --

Ergebnisdatei: uebung06.pdf

1. Welche grundlegenden Virenarten sind Ihnen bekannt?
2. Was wird als Malware bezeichnet?

7

Spyware, Phishing und Browser Hijacking

7.1 Geld verdienen im Internet

Internetwerbung

Seitdem das Internet für das Marketing entdeckt wurde besteht der Wunsch, Werbung möglichst zielgruppenorientiert zustellen zu können. Der Wunsch ist verständlich, da zielgruppenorientierte Werbung höhere Preise für diese Werbeschaltungen erwirtschaftet.

Der massive Einsatz der Werbung, beispielsweise in Form von Banner- und Suchmaschinenwerbung führt dazu, dass viele Benutzer aus Ärger darüber Werbung auf Internetseiten ablehnen ignorieren, indem Sie reflexartig auf den *Schließen*-Knopf eines Fensters drücken, sobald es auf dem Bildschirm aufgebaut wird.

Internetwerbung hat sich von simplen Bannergrafiken zu einer breiten Palette von Werbeformen entwickelt, wie zum Beispiel:

- ✓ Pop-ups (sich in einem neuen Fenster in den Vordergrund schaltende Werbung),
- ✓ Pop-unders (sich in einem neuen Fenster in den Hintergrund schaltende Werbung),
- ✓ Macromedia-Flash und andere multimediale Formate, die nur schlecht automatisch gefiltert werden können und den Benutzer noch aufdringlicher davon abhalten, den eigentlichen Seiteninhalt wahrzunehmen, bevor er seine Aufmerksamkeit nicht der Werbung gewidmet hat.

Cookies

Um nun nicht nur Inhalte an den Benutzer abliefern zu können, sondern gleichzeitig herauszufinden, wofür sich ein Benutzer interessiert, benutzen sehr viele Bannerwerbungsvermarkter sogenannte Cookies.

Cookies (englisch für „Kekse“ oder „Plätzchen“) sind kleine Informationshäppchen, die ein Webserver einem Rechner, der eine bestimmte Webseite besucht, zuteilen kann bzw. die durch den Browser mittels JavaSkripts erzeugt werden. Grundsätzlich kann ein Cookie auch nur von dem Server ausgelesen werden, der einem bestimmten Benutzer dieses Cookie gegeben hat.

Auf diese Weise wird es z. B. ermöglicht, dass Sie auch bei einer Unterbrechung Ihrer Einkaufssitzung den Inhalt Ihres Onlinewarenkorbs beim Fortsetzen dieser Shoppingtour wiederfinden oder dass ein Webserver registrierte Benutzer erkennt, ohne dass diese ihr Passwort eingeben mussten. Dabei handelt es sich jedoch nicht unbedingt um eine empfehlenswerte Vorgehensweise zur Benutzerauthentifikation.

Einige Serverbetreiber (unter denen sich üblicherweise viele Werbefirmen befinden) missbrauchen Cookies zur Erstellung von Bewegungsprofilen der beim Besuch von Webseiten zwangsbeworbenen Nutzer, indem sie die Tatsache ausnutzen, dass sie Werbeeinblendungen an viele verschiedene Webseitenbetreiber vermitteln.

Das übliche Geschäftsmodell sieht vor, dass ein Besitzer einer Homepage, der bereit ist, gegen Entgelt Bannerwerbung auf seinen Seiten einzublenden, sich beim Werbevermittler registriert und von diesem ein Stück HTML-Code erhält, den er an allen Stellen auf seinen Webseiten einbaut, an denen Werbung erscheinen soll.

Lädt ein Besucher diese Website, werden die normalen Inhalte vom Webserver des Betreibers der Homepage heruntergeladen. Der Code des Werbevermittlers sorgt allerdings dafür, dass die Werbeinhalte vom Server des Vermittlers nachgeladen werden. Dieser Server kann dem Besucher ein Cookie geben, an dem der Besucher später wiedererkannt werden kann.

Surft der Besucher nun zu einer völlig anderen Webseite, deren Besitzer ebenfalls beim selben Werbevermarkter registriert ist, lädt der Browser des Benutzers die Werbeinhalte wiederum vom ursprünglichen Werbeserver herunter. Dieser ist in der Lage, das zuvor gesetzte Cookie auszulesen.

Die beiden Betreiber der jeweiligen Websites wissen nicht mehr über die Besucher ihrer Seiten, als sie aus ihren eigenen Statistiken ersehen können. Der eigentliche Gewinner ist der Werbevermarkter, da er anhand seiner Logdateien und der Informationen aus den Cookies genau ableiten kann, wann ein bestimmter Benutzer in welcher Reihenfolge welche Websites besucht hat. Da ein Besitzer von Webseiten beim Registrierungsprozess für Werbevermittlung auch den Typ und die Inhalte seiner Website angeben muss, sind Werbevermittler durchaus in der Lage, statistisch sehr genau auszuwerten, welche Seiten ein Benutzer typischerweise besucht, welche Informationen oder Inhalte er bevorzugt bzw. aus welcher Region er kommt. Ein Cookie, das so eingesetzt wird, wird auch als **Tracking Cookie** (verfolgendes Cookie) bezeichnet.

Die nutzerseitig installierten Browser bieten die Möglichkeit, Cookies zu selektieren, z. B. die Annahme zu verbieten, sie temporär anzunehmen oder dauerhaft zu akzeptieren. Neuere Entwicklungen, wie Flash-Cookies, entziehen sich jedoch weitestgehend der Selektion innerhalb des Browsers.

Die Informationen über Benutzer und deren Verhaltensprofile sind auf Werbemarkten äußerst lukrativ. Deshalb besteht hier ein reges Interesse seitens der Werbevermittler, derartige Daten zu sammeln oder zu erhalten. Viele davon treiben das Ausmaß dessen, was statistisch erfasst und ausgewertet wird, immer weiter.

Die Methode, Verhaltensprofile über Internetbesucher zu erstellen und damit Geld zu verdienen, ist für betroffene Besucher von Webseiten aus Datenschutzgründen problematisch. In diesem Zusammenhang entstand schon früh der Begriff **Spyware**. Spyware gehört zu Kategorie der potentiell unerwünschten Anwendungen (**PUA**, engl. Potentially Unwanted Applications).

7.2 Spyware

Den Computer für Geld ausspionieren

Der Begriff Spyware wurde erst gefestigt, als Werbefirmen begonnen haben, spezielle Software zur Auswertung von Benutzerverhalten zu programmieren. Einmal installiert, erlauben diese Programme quasi die Rundumüberwachung des betroffenen Computers, auch ohne dass Webseiten besucht werden. Je nach Fantasie des Programmautors ist derartige Software in der Lage, jegliche Benutzereingaben und Aktionen statistisch auszuwerten und an den Server des Werbetreibenden zurückzusenden.

Im Gegenzug werden dann Werbeinhalte vom Server der Vermittlerfirma auf Vorrat heruntergeladen, sodass auch Werbung auf dem PC angezeigt werden kann (und wird), wenn keine Internetverbindung besteht.

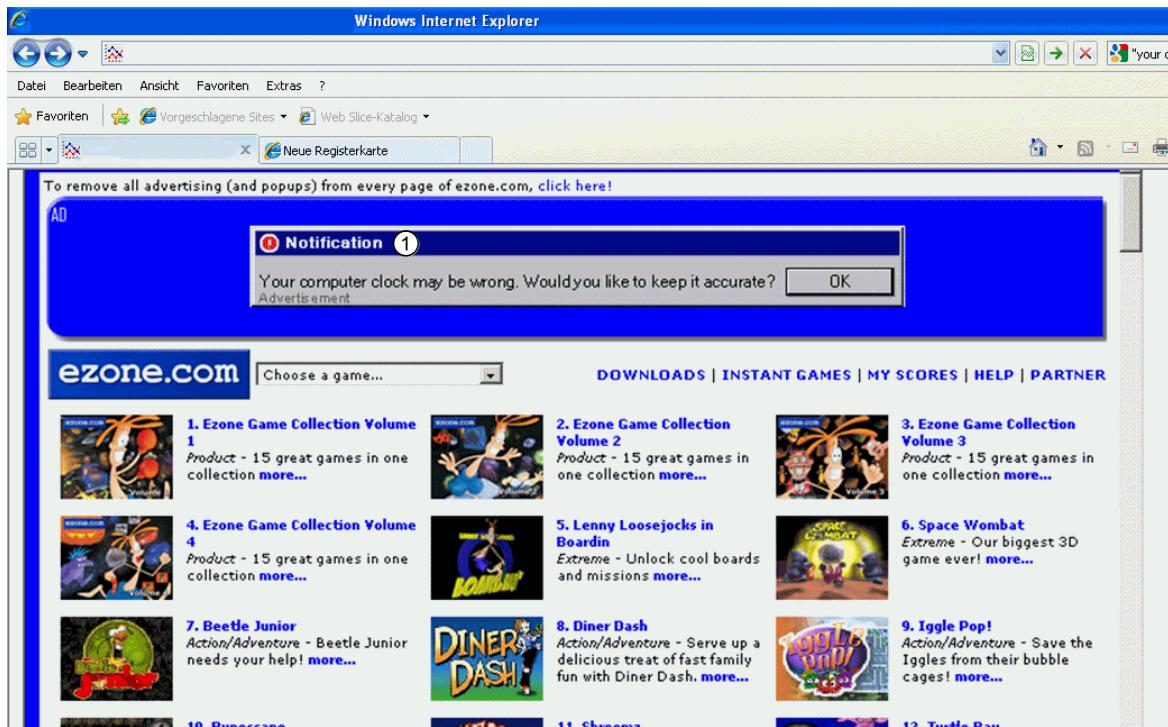
Derartige Software wird sich in der Regel nicht freiwillig als Spyware oder nicht einmal als Werbesoftware zu erkennen geben.

Die Betreiber unternehmen vielmehr alles Mögliche, um z. B. Dritten per Gerichtsbeschluss die Bezeichnung der eigenen Software als Spyware zu verbieten, oder man versteckt den eigentlichen Zweck der Software in blumig formulierten, seitenlangen Lizenztexten, die meist niemand genau liest, oder versucht, die eigene Software als „Adware“ zu beschreiben. Die letztendliche Namensgebung dürfte jedoch sehr flexibel sein und sich wohl hauptsächlich daran orientieren, wie sehr sich der Betroffene an der Art und Weise der Informationssammlung und Werbeschaltung stört.

Kostenlose Downloads mit Tücken

Da sich Benutzer in den seltensten Fällen freiwillig derartige Software auf ihren Computern installieren, wird mit einigen Tricks gearbeitet, um die Software dennoch verteilen zu können. Die offensichtlichste Methode ist, dem ahnungslosen Benutzer die Software einfach beim Besuch einer präparierten Website automatisch unterzuschieben, mit der Begründung, dass ohne die Bestätigung der Sicherheitsabfrage, die der Browser normalerweise stellen sollte, die Inhalte der Website nicht korrekt angezeigt werden können.

Durchaus häufiger anzutreffen sind Versuche, die Software als Utility anzupreisen, die irgendeinen Nutzeffekt für den Benutzer hat. Damit wäre eigentlich die klassische Definition für ein trojanisches Pferd erfüllt, das ja eben einen Nutzen für den Benutzer vorgaukelt (und eventuell sogar wirklich haben kann), mit dem aber in Wirklichkeit der Programmierer der Software andere, verdeckte Ziele verfolgt.



Installationslink für ein Spyware-Programm, das als Windows-Fehlermeldung getarnt wurde und die Korrektur einer angeblich falsch laufenden PC-Uhr verspricht

Über das Web sind inzwischen zahlreiche Varianten bekannt, mithilfe derer versucht wird, Benutzern die Programme unterzuschieben. Einige Varianten hiervon sind:

PC-Uhr	Ein als Windows-Fehlermeldung getarnter Dialog ① fordert den Benutzer auf, seine falsch laufende PC-Uhr zu synchronisieren. Die anschließend installierte Software mag das vielleicht auch tun (egal, ob das Betriebssystem vorher schon eine korrekte Uhrzeit hatte oder nicht), die Software fungiert aber dann auch als Werbevermittler.
Webbeschleuniger	Die Software wird als Webbeschleuniger angepriesen, der auf nicht näher beschriebene Weise das Surferlebnis des Benutzers verbessern soll.
Smileys/Emoticons	Die Software verspricht, durch lustig animierte Smileys selbst erstellte Webseiten oder neu verfasste E-Mails interessanter zu gestalten.
Toolbars	Toolbars werden üblicherweise als Add-on zum Internet Explorer geschrieben und versprechen ebenfalls „schöneres“ Surfen oder beschleunigten Zugriff auf Suchmaschinen etc. Auch hier wird das Surfverhalten des Benutzers ausspioniert und entsprechend Werbung geschaltet.

Software Bundles

Ebenfalls üblich ist es, Spyware-Programme oder speziell davon abgeleitete Module an Softwareautoren auszugeben. Der Autor erhält für das Einbinden von Spyware-Modulen in sein Programm Geld vom Spyware-Ersteller. Gegenüber dem Benutzer wird begründet, dass nur so die entsprechende Software als „Freeware“ zur Verfügung gestellt werden könne, weil sie anderweitig nicht finanziert wären. Dieses Verfahren ist auch als „Sponsoring“ bekannt.



Das Spyware-Programm blendet Werbung ein, die vor Spyware warnt und angeblich eine Lösung anbietet

Teilweise werden Werbeeinblendungen dann innerhalb des gesponserten Programms angezeigt; teilweise wird während der Installation des gesponserten Programms einfach eine Installationsroutine für das Standalone-Spyware-Programm ausgeführt.

Für eine kostenlose Software erscheinen Werbefenster vertretbar. Problematisch ist jedoch, wenn die Werbesoftware im System verbleibt und weiter wirbt, wenn der Benutzer die gesponserte Software entfernt hat.

Einige Autoren oder Vertreiber von Software wollen gleich mehrere Geldquellen parallel erschließen und bündeln mit ihrer 'gratis' Software nicht nur ein, sondern mehrere Spyware-Programme diverser Hersteller. Ein PC, der durch Ausprobieren mehrerer Programme mit einem halben Dutzend oder mehr Spyware-Programmen gleichzeitig befallen ist, wird auch trotz großzügiger Hardware-Ausstattung Leistungseinbußen aufweisen. Immerhin benötigen alle parallel laufenden Spyware-Programme ständig CPU-Zeit und Speicherkapazität, und sie belegen für den Upload von Statistiken und den Download von neuer Werbung Internetbandbreite.

Hinzu kommen aber auch noch Programmierfehler und Inkompatibilitäten der Spyware-Programme, die dann dem Benutzer durch häufige Abstürze noch weitere Probleme bescheren.

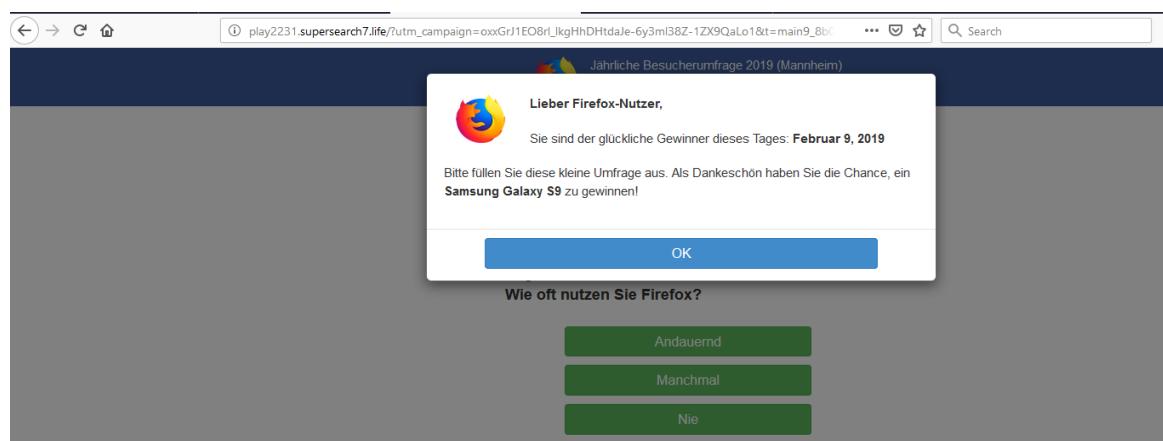
Häufig wird Spyware mit angeblicher Freeware gebündelt. Sie sollten bei der Installation von kostenloser Software aus dem Internet die Lizenzbedingungen aufmerksam lesen – auch wenn es keine Garantie gibt, dass der Hersteller die Verwendung von Werbesoftware in diesen Texten erwähnt.

7.3 Browser Hijacking

Entführte Browser

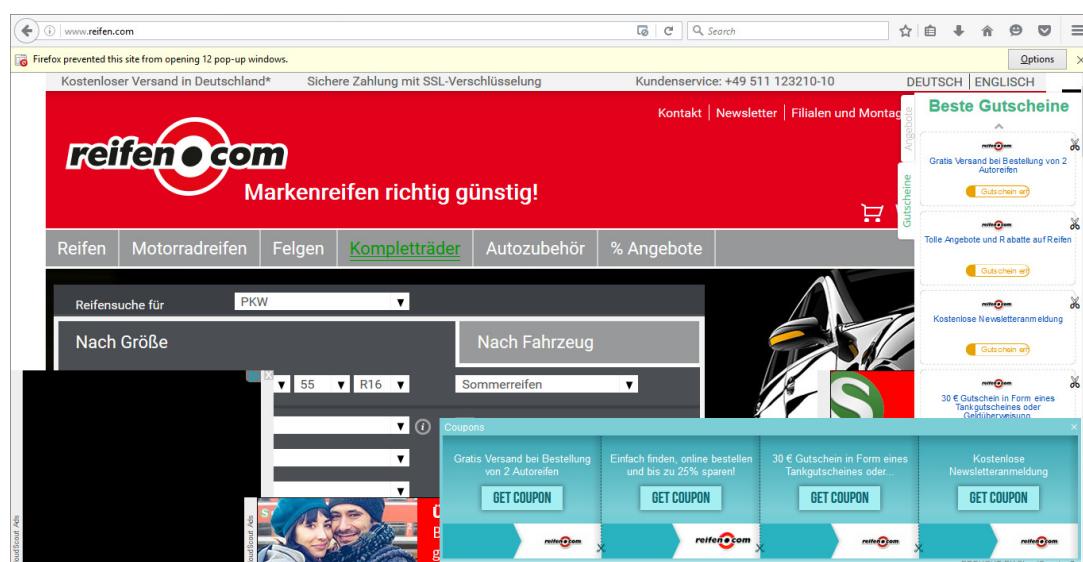
Beim **Browser Hijacking** wird durch speziell präparierte Webseiten oder Software, die dem Benutzer untergeschoben wird, der Browser des Opfers so manipuliert, dass z. B. die Startseite des Browsers nicht mehr auf die Original-Startseite, sondern auf die vermeintliche Suchmaschine des Hijackers zeigt.

Die unerwünschte Startseite, auf die der Benutzer des Computers dann bei jedem Start des Browsers oder bei der Falscheingabe einer URL gelenkt wird, enthält Werbung in Form von Bannern, Pop-ups, etc. Üblich sind in diesem Umfeld auch Verlinkungen zu Onlinecasinos und pornografischen Angeboten, wobei aber auch häufiger Pseudoangebote wie Gratis-Gewinnspiele oder vermeintlich harmlose Flirt-Dienste auftauchen.



Beispiel einer Weiterleitung zu einer Gewinnspielseite für das Abgreifen von persönlichen Daten

Browser Hijacker legen auch ungefragt zahlreiche neue Lesezeichen an oder verändern die Einstellung im Webbrowser.



Ein Firefox-Browserfenster, nachdem der Browser von mehreren Parteien gehijacked wurde

Allgemein sind alle Browser wie Internet Explorer, Firefox, Opera, Google Chrome, Apple Safari etc. von dieser Thematik mehr oder weniger betroffen. Ob Kriminelle sich die Mühe machen, für einen speziellen Browser oder eine bestimmte Version eines Browsers entsprechende Hijacking-Methoden zu entwickeln, hängt davon ab, wie stark dieser Browser verbreitet ist und wie viel finanziellen Erfolg sich dubiose Geschäftemacher davon versprechen, eine bestimmte Plattform zu infiltrieren.

Die Hersteller von Browsersoftware haben Sicherheitsfeatures in ihre Produkte implementiert, die ein Hijacking erschweren. Die Geschäftemacher suchen aber immer neue Möglichkeiten, gutgläubigen Nutzern ihre Manipulationen unterschieben zu können.

Die Kontrolle über den eigenen PC verlieren

Besonders lästig ist Browser Hijacking dann, wenn der Hijacker durch trickreiche Methoden dafür gesorgt hat, dass der Benutzer entweder seine Browser-Einstellungen gar nicht korrigieren kann (dies ist über Manipulationen an der Registry möglich) oder dass nach der Korrektur der Startseite und der Lesezeichen die unerwünschten Varianten spätestens nach einem Neustart des Systems wieder da sind.

Hier wird parallel noch weitere Software im System verankert, die ständig überprüft, ob die vom Hijacker gewünschten Einträge noch vorhanden sind, und diese gegebenenfalls wieder erstellt.

Problemen durch Spyware können Sie am besten dadurch begegnen, dass Sie zum einen sehr genau darauf achten, welche Software Sie aus welchen Quellen installieren. Grundsätzlich sollten Sie die Software vom Hersteller direkt laden. Zum anderen sollten Sie spezielle Anti-Spyware benutzen, die im Stile von Virenscannern die Festplatte nach installierten unerwünschten Programmen und Cookies durchsucht und diese auf Wunsch entfernt. Der Einsatz von Anti-Spyware wird nachfolgend in diesem Kapitel behandelt.

Auch die Verwendung alternativer Software für Ihre tägliche Internetkommunikation kann die Probleme durch Cookies und unerwünschte Softwareinstallation eventuell eindämmen, da alternative Software möglicherweise über bessere Sicherheitsmechanismen zur Abwehr derartiger Probleme verfügt (vgl. Kapitel 18).

Der **beste Schutz** bleibt allerdings nach wie vor Ihre **Vorsicht** beim Besuch von Websites und bei der Installation neuer Software bzw. beim Bestätigen von Dialogboxen.

7.4 Was ist Phishing?

Passwortdiebstahl per Internet

Eine weitere kriminelle Methode, sich auf Kosten von Computernutzern zu bereichern, ist unter dem Namen Phishing (ausgesprochen als: „fischung“) bekannt. Der Begriff kommt vom englischen „Fishing“, wobei das Ph am Anfang des Wortes wohl auf die Verknüpfung von „Password“ und „Fishing“ zurückzuführen ist.

Beim Phishing handelt es sich um eine spezielle Form des elektronischen Trickbetrugs, die als Social-Engineering-Angriff eingestuft werden kann. Hierbei versucht der Angreifer, das Opfer durch offiziell aussehende E-Mails und Webseiten zur Preisgabe von sensitiven Daten zu verleiten.

Daten, die beim Phishing begehrte sind:

- ✓ Onlinebanking-Zugangsdaten
- ✓ Anmeldeinformationen für Auktionshäuser (eBay etc.)
- ✓ Anmeldedaten für Onlinezahlungsdienste (PayPal etc.)
- ✓ Accounts für soziale Netzwerke (z. B. Facebook, Twitter oder XING)

Die Phishing-Attacken haben signifikant zugenommen. Das dürfte nicht zuletzt daran liegen, dass es bei dieser Variante des Social Engineering wesentlich leichter ist, sich Geld (oder Informationen, die einen geldwerten Vorteil bringen) zu erschwindeln, als z. B. mit früher genutzten 0900-Dialer-Betrugsmaschen oder Ähnlichem. Obwohl die Methoden im Einzelfall variieren können, laufen Phishing-Attacken immer nach einem ähnlichen Muster ab:

- ✓ Der Phisher sammelt möglichst viele E-Mail-Adressen seiner Opfer, ähnlich wie ein Spammer. Datenbanken mit E-Mail-Adressen werden auch im Darknet angeboten.
- ✓ Es wird eine E-Mail verfasst, die den Empfänger auffordert, sich aus einem fingierten Grund (gerne werden hier Sicherheitsprobleme oder Aktualisierungen angeführt) bei der Onlinebank, dem Auktionshaus oder der jeweiligen Partei, deren Zugangsdaten gestohlen werden sollen, anzumelden.
- ✓ Diese E-Mail gleicht in ihrem äußeren Erscheinungsbild einem offiziellen Schreiben der Bank oder einer anderen Institution oder Firma.
- ✓ Es wird ein Link in die Mail eingebettet, den der Benutzer aktivieren soll, um auf die Log-in-Seite der Bank zu gelangen. Dieser Link führt aber nicht auf die Seite der echten Bank, sondern auf einen Server, der unter der Kontrolle des Phishers steht. Dort wurde das optische Erscheinungsbild der Bank oder der Institution detailgetreu nachgebildet.
- ✓ Klickt das Opfer auf den Link und gibt seine Zugangsdaten auf der gefälschten Website ein, so gelangt der Phisher in den Besitz dieser Daten und kann sich im Namen des Opfers bei dessen Online-Account anmelden und z. B. Transaktionen oder andere Aktivitäten veranlassen.

Hat ein Phisher auf diese Weise die Zugangsdaten für einen fremden Online-Account erlangt, ist es ihm möglich, Überweisungen vom Konto des Opfers auf das eigene Konto vorzunehmen oder in dessen Namen betrügerische Auktionen zu veranlassen.

Damit der Phisher möglichst lange ungestört über den gestohlenen Account verfügen kann, ist es auch nicht unüblich, dass das Passwort des Accounts geändert wird, um den rechtmäßigen Besitzer von einem Zugang und somit von einer vorzeitigen Entdeckung der illegalen Aktionen auszuschließen.

Damit Links in gefälschten E-Mails nicht so leicht als Fälschungen zu erkennen sind, werden mehrere Methoden benutzt, um die wahre Herkunft einer URL zu verschleiern. Würde der offizielle Link der Sparkasse Regensburg beispielsweise auf <http://www.sparkasse-regensburg.de> lauten, so wäre eine Fälschung mit <http://217.146.123.18> zu leicht zu erkennen.

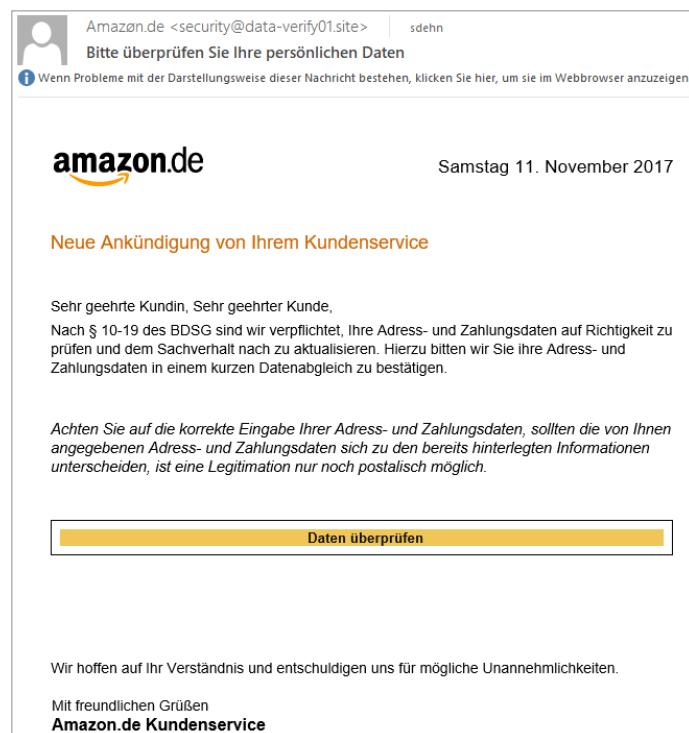
Bessere Fälschungen könnten sein:

www.security-sparkasse-regensburg.de www.meinebank-regensburg.com	Täuschend ähnliche, aber tatsächliche anders lautende Domains
www.sparkasse-regensburg.de@217.146.123.18	Der Original-Domänenname ist hier aufgrund der Konstruktion der URL nur ein Benutzername, der eigentliche Server ist 217.146.123.18.
www.pastbank.de	Absichtliches Einbringen von Rechtschreibfehlern, die nicht sofort auffallen statt www.postbank.de

Erkennungsmerkmale von Phishing-Mails

Phishing-Mails sind leicht zu erkennen, wenn Sie mehrere der folgenden Merkmale in einer Mail vorfinden:

- ✓ Sie werden aufgefordert, (meist dringend) eine Anmeldung vorzunehmen. Oft werden Sicherheitsargumente vorgebracht.
- ✓ Die Mail enthält einen Link, der angeklickt werden soll, oder hat ein Programm beigelegt, welches die Zugangsdaten von Ihnen anfordert.
- ✓ Speziell im deutschen Onlinebanking werden Sie explizit zur Eingabe von TAN-Nummern aufgefordert.
- ✓ Im Anschreiben wird Ihnen mit Accountsperreng gedroht, falls Sie die „Aktualisierung“ etc. nicht unverzüglich vornehmen.
- ✓ Sie werden im Anschreiben nicht persönlich angesprochen. Ihre echte Onlinebank würde Ihren Vor- und Nachnamen kennen.
Diese Phishing-Attacken wurden jedoch zunehmend durch personalisiertes Phishing abgelöst, bei dem das Opfer persönlich und individuell angesprochen wird (sog. „Spear-Phishing“).
- ✓ Rechtschreib- oder Grammatikfehler in der Mail
Da viele Phisher im Ausland (China, Ghana, Russland usw.) sitzen und von dort ihre Mails verteilen, ergeben sich bei der automatisierten Übersetzung ins Deutsche teilweise eine ungewöhnliche Wortwahl oder unübliche grammatischen Konstellationen.



Eine Phishing-Mail mit dem Ziel, Zugangsdaten zu erhalten

Filtern von Phishing

Auch wenn das Wissen und die Erfahrung des Benutzers beim Erkennen von Phishing-Mails noch die beste Abwehr darstellt, gibt es einige technische Methoden, um die Ausbreitung von Phishing-Mails einzudämmen bzw. zu verhindern.

- ✓ **Spam-Filter:** Da Phishing-Mails genauso unerwünscht sind wie Werbe-Mails, werden die Server, von denen aus Phishing-Mails versandt werden, früher oder später über DNS-Blacklisten gesperrt. Ein Spam-Filter, der diese Blacklisten nutzt, sortiert dann auch von vornherein diese Phishing-Mails aus. Selbstlernende Bayes-Filter haben hier einen Vorteil: Sobald im Bayes-Filter die typische Struktur und Wortwahl einer Phishing-Mail erfasst ist, kann der Bayes-Filter Phishing-Mails einfach als Spam klassifizieren und aussortieren.
- ✓ **Restriktivere E-Mail-Software:** Aktuelle E-Mail-Clients haben einen speziellen Schutz vor Phishing-Mails implementiert, der den Benutzer warnt, wenn er auf einen Link in einer E-Mail klickt, der zu einem anderen Ziel führt als derjenige, der in der E-Mail angezeigt wird, oder wenn der Link UTF8-codiert wurde (ein Format zur Darstellung von Unicode-Zeichen mit einem 8-Bit-ASCII-Zeichensatz), um den eigentlichen Inhalt für menschliche Leser zu verschleiern.
- ✓ **Restriktivere Einstellungen in den Webbrowsern oder alternative Webbrowser:** Um Verschleierungsversuche durch ähnliche Zeichen in Domainnamen zu verhindern, haben Browser einen Schutzmechanismus implementiert, falls in einem Domänenamen ähnlich dargestellte Zeichen verwendet werden. Als Beispiel soll www.paypal.com dienen, wobei das zweite a ein kyrillisches Unicode-Zeichen ist und für den Leser optisch von dem lateinischen a nicht zu unterscheiden ist. Dieser Link führt somit zu einer anderen Domäne als der, die der Besucher annimmt. Ein aktueller Webbrowser warnt Sie beim Aktivieren des Links vor dem Besuch einer Webseite, wenn in der URL ein Zeichen verwendet wird, das optisch gleichartig dargestellt wird, weil das Erscheinungsbild nicht mit dem Zeichencode des Domänenamens übereinstimmt (vgl. Kapitel 18).

7.5 Anti-Spyware einsetzen

Den PC wieder benutzbare machen

Sind auf einem PC erst einmal zweifelhafte Programme installiert worden, bleiben diese in der Regel nicht lange allein. Entweder werden aufgrund des Surfverhaltens des Benutzers noch weitere Spyware-Programme und sonstige „Tools“ installiert, oder die bereits installierte Software entwickelt ein gewisses Eigenleben und installiert selbsttätig weitere Software nach, auf die der Benutzer nie hingewiesen wurde.

Analog zu Antivirensoftware gibt es inzwischen auf dem Markt zahlreiche Produkte, die sich zur Aufgabe gemacht haben, einen befallenen PC von Spyware und verfolgenden Cookies zu befreien. Diese Funktion war anfänglich aufgrund juristischer Auseinandersetzungen mit Spyware-Herstellern nicht in der Antivirensoftware integriert worden. Inzwischen hat sich diese Situation aber grundlegend geändert und fast alle Hersteller von Antivirenprogrammen bieten integrierte Spyware-Signaturen an.

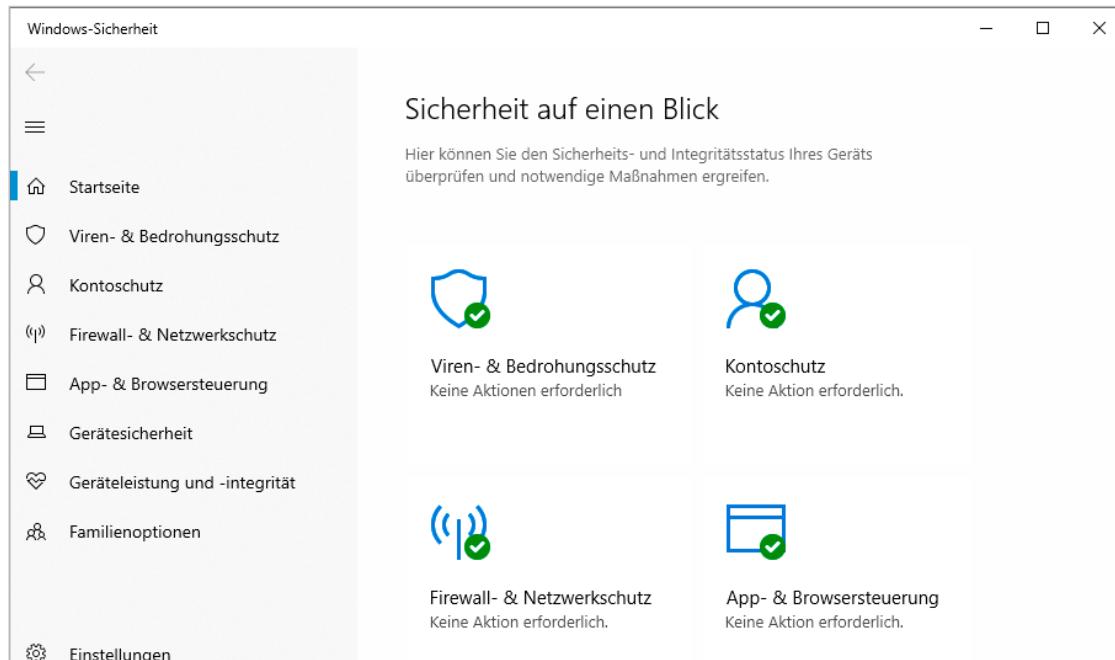
Wenn Sie planen, Ihren Computer von Spyware zu befreien, bedenken Sie, dass Sie eventuell Gratis-Software installiert haben könnten, mit der diese Spyware gekoppelt war. Rein rechtlich gesehen dürften Sie diese Software nur benutzen, solange Sie auch die Werbeeinblendungen und die Überwachung der Spyware dulden. Einige werbefinanzierte Programme sind sogar so entworfen worden, dass sie nur funktionieren, solange sich die Spyware noch auf dem PC befindet.

Auf dem Markt ist gute kostenlose Anti-Spyware/Anti-Malware erhältlich, es gibt aber auch einige Softwarefirmen, die im kommerziellen Umfeld Anti-Spyware-Lösungen vertreiben. Sollten Sie auf der Suche nach einer speziellen Anti-Spyware-Lösung sein, so können Sie hier dieselben Auswahlkriterien anlegen wie bei der Anschaffung einer bestimmten Antivirenlösung.

Microsoft Defender

In den aktuellen Betriebssystemen von Microsoft kommt als Nachfolger von Security Essentials der Defender zum Einsatz. Er stammt von der Betaversion **Microsoft Windows AntiSpyware**. Er kann auch unter den älteren Betriebssystemen (Windows XP mit Service Pack 2, Windows Server 2003 mit Service Pack 1 oder Windows Server 2008) genutzt werden. Die ursprüngliche Aufgabe bestand in der Beseitigung von Spyware. Dieses Stadium hat der Defender inzwischen verlassen und realisiert einen umfassenden Malwareschutz.

Durch die vollständige Integration in das Betriebssystem startet der Scan-Vorgänge automatisch und schützt Applikationen wie den *Internet Explorer* oder das *Outlook*-Postfach in Echtzeit. Die Signaturen für die Virendefinitionen und Heuristiken werden direkt über die Updatefunktion des Betriebssystems geladen.



Security Center des Microsoft Defenders

Sie können über das Security Center zusätzlich die Malwareerkennung offline aktivieren oder den Edge-Browser in einer geschützten Umgebung ausführen.

Der Malwareschutz im Defender sollte für die Blockierung potenziell unerwünschter Anwendungen aktiviert werden bzw. sein. Testen Sie dies, indem Sie mit Administratorrechten in der PowerShell den Befehl `Get-MpPreference | findstr PUA` absetzen. Wenn der Wert 0 beträgt, müssen Sie die PUA-Erkennung mit dem Kommando `Set-MpPreference -PUAProtection enable` aktivieren.

Bezüglich des Malwareschutzes liegt die Defender-Version 4.18 (02/2019) auf gleichem Niveau wie die kommerziellen Mitbewerber. Es gibt also keinen signifikanten Grund, mit Ausnahme von speziellen Funktionen bzw. Gewohnheiten, andere Produkte für den Einsatz zu nutzen.

Externe Antimalware greift zudem tief in das Betriebssystem ein und könnte so ein Sicherheitsproblem schaffen. Diese Meinung wird u. a. vom Ex-Mozilla-Entwickler Robert O'Callahan vertreten. Sofern Sie nicht auf den Einsatz von Antimalware von Drittanbietern verzichten wollen, kann der Defender parallel im aktiven oder passiven Modus arbeiten (siehe auch <https://docs.microsoft.com/de-de/windows/security/threat-protection/windows-defender-antivirus/windows-defender-antivirus-compatibility>).

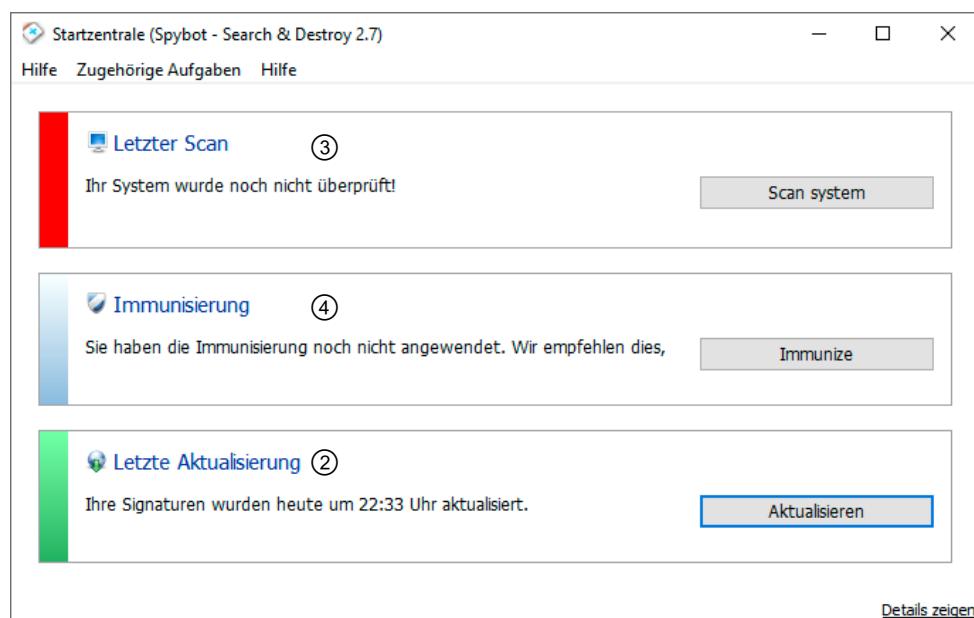
Spybot - Search & Destroy

Spybot - Search & Destroy ist ein Klassiker unter den Antivirenprogrammen und über die Website <https://www.safer-networking.org> kostenlos verfügbar. Neben der Erkennung und Vorbeugung verfügt Spybot über interessante, teilweise kostenpflichtige, Tools.

- ▶ Nach dem Download starten Sie die Installation und wählen die Sprache das Nutzerverhalten, die Komponentenausführung und die zusätzlichen Aufgaben aus.



Nach Abschluss der Installation können Sie *Fertigstellung des Spybot-Assistenten* Spybot - Search & Destroy ausführen ①.



Hauptmenü von Spybot

Nachdem Spybot seine Informationen aktualisiert hat ②, können Sie über den *System-Scan* ③ das System auf Malware überprüfen.

Über *Immunisierung* ④ bietet Ihnen Spybot eine Möglichkeit, sowohl das komplette System als auch nur die Benutzerkontensteuerung gegen unbeabsichtigte Änderungen zu schützen.

Nach Ende des System-Scans werden die Ergebnisse der erkannten Mal-Spyware mit Pfadangabe, klassifizierter Bedrohungsstärke (von Grün = Warnung bis Rot = Infizierung) und Dateityp aufgelistet ⑤.

Sofern einzelne Dateien keine Bedrohung darstellen, können Sie diese durch Entfernen des Häkchens ⑥ abwählen und die Reinigung über *Ausgewähltes beheben* ⑦ starten.

Beschreibung	Pfad	Bedrohung...	Typ	Kategorie	Regel-ID
PU.AdvancedSystemRepairPro	C:\ProgramData\TSR7Settings\	■	Directory	PUPS-C	7B908DF0
Executable	C:\ProgramData\TSR7Settings\dsutil.exe	■	File	PUPS-C	01F788D0
Macromedia.FlashPlayer.Cookies					
Text file	C:\Users\User\AppData\Roaming\Macromedia\FlashPlayer\Temporary\Global\#00000000000000000000000000000000	■	File	Tracks	6AA61750
MediaPlex					
Tracking cookie...	.mediaplex.com/ (svid)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.mediaplex.com/ (rts)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.mediaplex.com/ (mojo3)	■	Browser: Cookie	Browser	ASBRCOOK
Statcounter					
Tracking cookie...	.statcounter.com/ (_cfduid)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.statcounter.com/ (ts_unique)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.statcounter.com/ (is_visitor_unique)	■	Browser: Cookie	Browser	ASBRCOOK
DoubleClick					
Tracking cookie...	.doubleclick.net/ (permutive-session)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.doubleclick.net/ (permutive-id)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.doubleclick.net/ (IDE)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.doubleclick.net/ (DSID)	■	Browser: Cookie	Browser	ASBRCOOK
LinkSynergy					
Tracking cookie...	.linksynergy.com/ (lsn_statp)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.linksynergy.com/ (lsn_track)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.linksynergy.com/ (lclick_mid24908)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.linksynergy.com/ (iccts)	■	Browser: Cookie	Browser	ASBRCOOK
Tracking cookie...	.linksynergy.com/ (rmuid)	■	Browser: Cookie	Browser	ASBRCOOK
Tradedoubler					
Tracking cookie...	.tradedoubler.com/ (BT)	■	Browser: Cookie	Browser	ASBRCOOK

Ergebnis nach einer Malwareanalyse

Kombination von Anti-Spyware

Wie auch bei Virenscannern haben Sie bei Anti-Spyware keine Garantie, dass Sie mit einem Durchlauf sämtliche Spyware, die sich auf einem PC befindet, erkennen und entfernen können. Je nach Aktualität der Signaturen und der Spyware kann es vorkommen, dass Löschungsversuche fehl-schlagen, weil die Spyware gezielte Gegenmaßnahmen gegen Anti-Spyware implementiert hat (quasi eine Anti-Anti-Spyware-Funktion) oder die Anti-Spyware den Schädling gar nicht erst findet.

Um Ihre Chancen zu erhöhen, können Sie mehrere Anti-Spyware-Programme parallel benutzen, aber nicht gleichzeitig auf Verhindern neuer Infektionen konfigurieren, weil sich sonst die Spyware-Scanner gegenseitig blockieren könnten. Dies gilt übrigens auch für Virenscanner. Es sollte immer nur ein Scanner als aktiver auf dem Rechner installiert sein. Allerdings ist der beste Schutz nach wie vor, Spyware gar nicht erst auf Ihren Computer zu lassen. Da einige Spyware-Programme speziell auf die Lücken des Internet Explorers ausgelegt sind, wäre eine Alternative, einen anderen Webbrowswer zu benutzen, der ein besser konfigurierbares Sicherheitskonzept bietet (vgl. Kapitel 18).

7.6 Übung

Fragen zu Spionagemethoden

Übungsdatei: --

Ergebnisdatei: uebung07.pdf

1. Was ist Phishing?
2. Welche dieser Aussagen treffen auf Adware und Spyware zu?

a	Ausschließlich Spyware sammelt Daten über den Benutzer.
b	Der Begriff „Adware“ steht für Software, die oft kostenlos ist und sich über Werbeeinblendungen finanziert.
c	Adware und Spyware sammeln personenbezogene Daten.

3. Was trifft auf Browser Hijacking zu?

a	Browser Hijacking betrifft ausschließlich den Internet Explorer.
b	Die Kontrolle über den eigenen PC geht verloren.
c	Der Browser wird so manipuliert, dass die Startseite nicht mehr die eingestellte Original-Startseite darstellt.
d	Der Neustart des Browsers beendet das Browser Hijacking.

8

Stand-Alone-Virenschutz

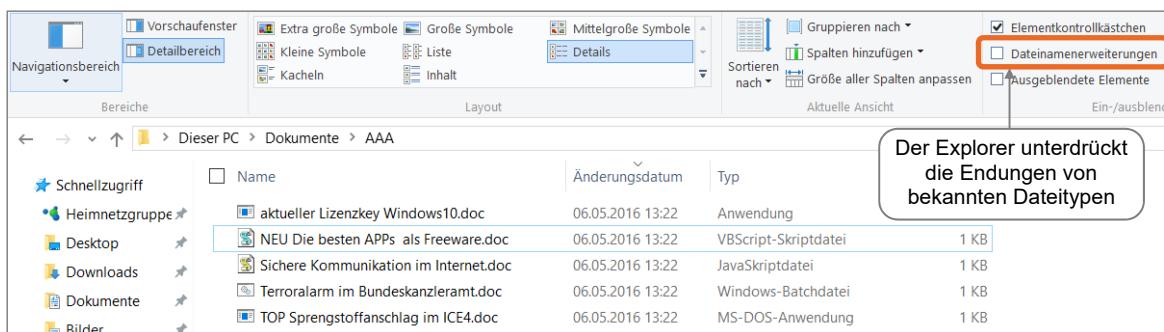
8.1 Einfache Virenprävention

Einstellungen im Betriebssystem

Die Voreinstellungen von neu installierten Betriebssystemen und Anwendungen sind primär in Hinblick auf Komfort optimiert und erst sekundär auf Betriebssicherheit. Mit wenigen Handgriffen können Sie Ihren Computer proaktiv schützen.

Dateiendungen auch bei bekannten Dateitypen anzeigen

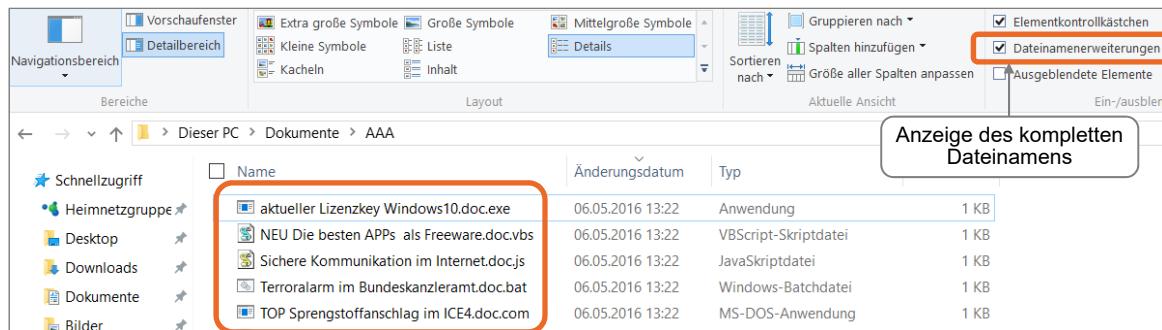
In der Voreinstellung werden im Windows-Explorer die Dateiendungen bei bekannten Dateitypen nicht angezeigt. In der nachfolgenden Abbildung erkennen Sie den Dateityp nur anhand des Icons oder bei der Detailansicht durch die Typangabe.



Diese Eigenschaft wird von Malware gerne benutzt, um über doppelte Dateiendungen den Benutzer in Sicherheit zu wiegen. Dem User wird vorgegaukelt, es handle sich um ein sicheres Dokumentformat wie Bilder, Videos oder Texte, während die vom Betriebssystem verborgene Endung zu einem ausführbaren Dateityp gehört. Ein Doppelklick auf eine so getarnte Datei öffnet dann kein Dokument, sondern startet den Viruscode.

In den meisten Fällen wird ein Dateiname gewählt, der einen für den Empfänger der Datei interessanten Inhalt vermuten lässt, wie: aktueller Lizenzkey Windows10.doc oder Sichere Kommunikation im Internet.doc.

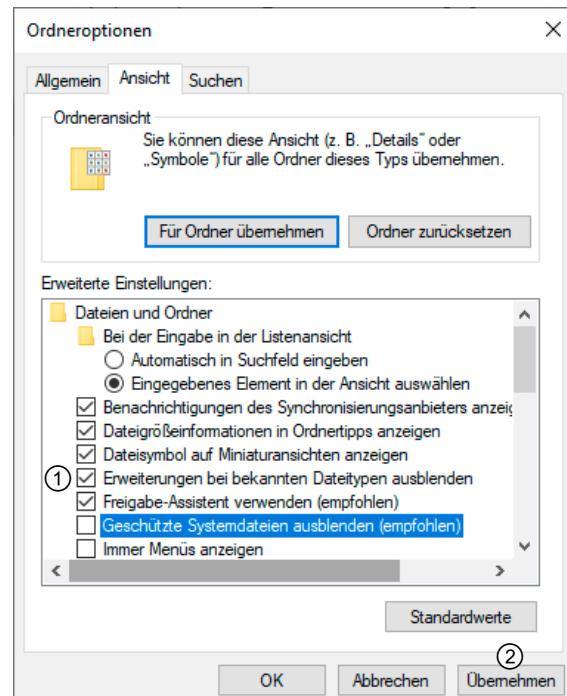
Unter Windows 8/10 können Sie die Anzeige der Dateierweiterungen im jeweiligen Ordner sehr einfach realisieren. Öffnen Sie einen Dateiordner. Aktivieren Sie im Register *Ansicht* im rechten Teil des Menübandes das Feld *Dateinamenerweiterungen*.



Um die Dateierweiterung immer anzeigen zu lassen, gehen Sie wie folgt vor:

- ▶ Klicken Sie in einem beliebigen Explorer-Fenster auf den Menüpunkt *Datei* und dann auf *Optionen* (bei Windows 10 auf *Ordner- und Suchoptionen ändern*).
- ▶ Wählen Sie das Register *Ansicht*.
- ▶ Deaktivieren Sie das Feld *Erweiterungen bei bekannten Dateitypen ausblenden* ①.
- ▶ Bestätigen Sie mit *Übernehmen* ② und *OK*.

Auf diese Weise können Sie ein versehentliches Ausführen eines entsprechenden Virus vermeiden, da jetzt die ausführbare **doppelte** Dateiendung angezeigt wird.



Anzeige der Dateinamenerweiterungen unter Windows 10 aktivieren

VBS-Default ändern

Die Dateiendung .vbs steht für Visual Basic Script und ist eine in aktuellen Windows-Betriebssystemen integrierte, sehr mächtige Skriptsprache.

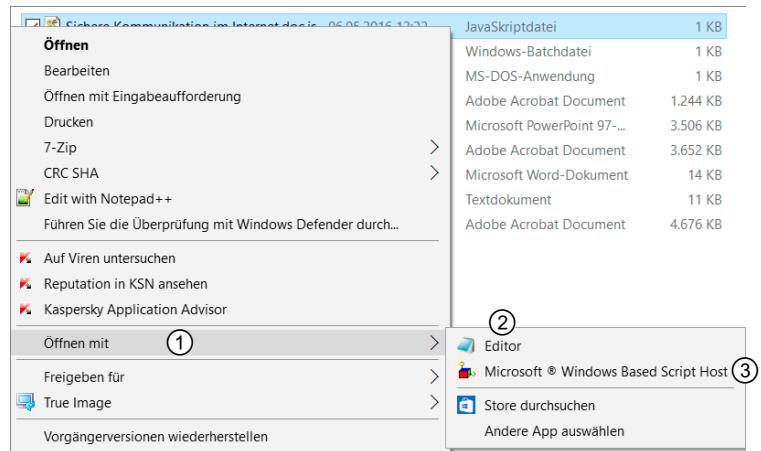
Da einige Skriptviren diese Sprache nutzen, der normale Benutzer allerdings eher selten VBS-Skripte ausführen muss, empfiehlt es sich, die Standardeinstellung für die Darstellung u. a. von VBS-Dateien zu ändern. Dies muss für jeden Benutzer (User) separat erfolgen.

VBS ab Windows 7

Um ab Windows 7 die verknüpfte Anwendung zu einer VBS (oder einem anderen Dateityp) zu ändern, gehen Sie wie folgt vor:

- ▶ Klicken Sie mit der rechten Maustaste auf eine Datei des Typs .vbs, und wählen Sie **Öffnen mit** ①.
- ▶ Wählen Sie als Anwendungsprogramm den Editor ② aus.

Jetzt kann die vbs-Datei ohne deren Ausführung geöffnet werden. Sollten Sie diese ausführen wollen, so steht Ihnen immer noch der Kontextmenüpunkt ③ zur Verfügung.



Verknüpfte Programme unter Windows 10 ändern

E-Mail-Clients sichern

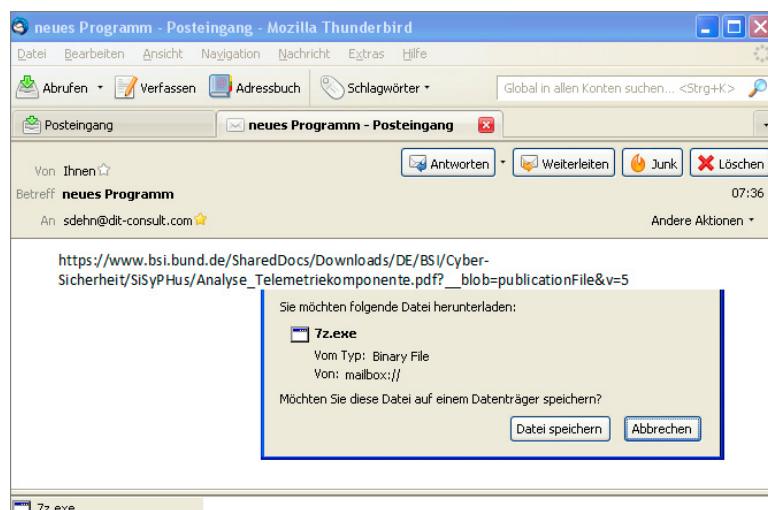
Eine häufig auftretende Verbreitungsart für Viren ist die E-Mail. Hierbei versendet sich der Virus entweder als Attachment und ist darauf angewiesen, dass Benutzer das Attachment per Doppelklick starten. Oder der Virus versendet sich an seine Opfer in Form einer speziell formatierten HTML-E-Mail, die bei unsicherer E-Mail-Software dazu führt, dass beim Lesen der E-Mail selbst der Virus schon aktiv werden kann (E-Mail-Wurm).

Den größten Einfluss auf Ihre E-Mail-Sicherheit haben Sie, wenn Sie Ihre E-Mail-Software frei wählen können. Wichtige Auswahlkriterien sind, wie das Programm mit HTML-Mails umgeht und in welchem Umfang Art und Inhalt von E-Mails dem Benutzer angezeigt werden, ohne dass eventuell schädliche Inhalte sofort aktiviert werden. Im Folgenden wird exemplarisch auf die Absicherung der E-Mail-Clients Mozilla Thunderbird Mail und dem Klassiker Microsoft Outlook eingegangen.

Mozilla Thunderbird Mail sichern

Das frei verfügbare E-Mail-Programm Mozilla Thunderbird Mail (<http://www.thunderbird-mail.de/>) behandelt Mail schon in der Grundkonfiguration relativ sicher (vgl. Kap. 18 für eine allgemeine Programmbeschreibung).

Sofern eine E-Mail ein Attachment besitzt, wird explizit eine Warnung ausgegeben. Der Nutzer kann nun entscheiden, ob er diese Datei speichern will. Nach einem Scan kann diese geöffnet werden.



Hinweis auf ein Attachment

Thunderbird ist auch in der Lage, HTML-E-Mails anzuzeigen. Die HTML-Funktionen wurden jedoch von den Programmierern mit eingeschränkter Funktionalität im Hinblick auf Sicherheit implementiert.

Webbugs

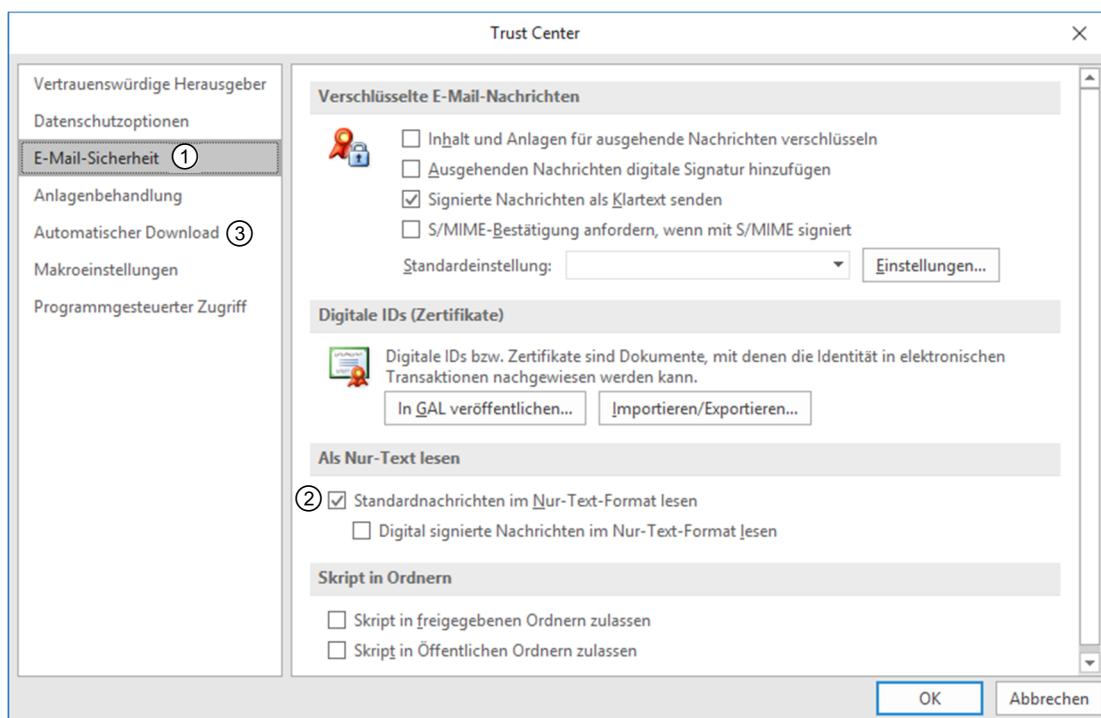
Webbugs sind kleine Grafiken (oft auch in derselben Farbe wie der Bildhintergrund, um sie quasi unsichtbar zu machen), die dem Ersteller dieser Applikationen ohne Wissen des Empfängers umfangreiche Statistiken über die Hard-/Software und das Leseverhalten des Computernutzers übermitteln.

Trotz sorgfältiger Konfiguration sind in HTML-E-Mails enthaltene Webbugs immer noch aktiv. Wenn Sie eine mit einem Webbug versehene HTML-E-Mail öffnen oder diese in der Vorschau anzeigen, erhält der Ersteller des Webbugs eine Lesebestätigung für diese E-Mail. Zusätzlich werden Ihre aktuelle IP-Adresse und eine Auflistung weiterer Daten übermittelt, die Ihren Computer, seine Konfiguration und das von Ihnen verwendete Betriebssystem genauer beschreiben.

Microsoft Outlook 2010/2013/2016 sichern

Im Gegensatz zu älteren Versionen von Outlook ist es ab der Version 2010 und höher relativ leicht, sich vor unliebsamen HTML-Mails zu schützen – auch ohne die Registry zu bearbeiten.

- ▶ Klicken Sie im Register *Datei* auf die Schaltfläche *Optionen*.
- ▶ Betätigen Sie in der Kategorie *Sicherheitscenter* oder *Trust Center* die Schaltfläche *Einstellungen für das Sicherheitscenter/Trust Center*.
- ▶ Wählen Sie im folgenden Dialogfenster links die Kategorie *E-Mail-Sicherheit* ①.
- ▶ Aktivieren Sie die Option *Standardnachrichten in Nur-Text-Format lesen* ②.



E-Mail-Empfang mit Outlook absichern

Darüber hinaus verfügt das Vertrauensstellungscenter unter der Kategorie *Automatischer Download* ③ über die Möglichkeit, in der Anzeige von HTML-E-Mails das automatische Nachladen extern verlinkter Bilder nur von vertrauenswürdigen Quellen – in der Regel eine von Ihnen erstellte Liste sicherer Absender oder Empfänger – zu gestatten. Alle übrigen Bilder werden nicht automatisch angezeigt, wie es bei älteren Outlook-Varianten ja der Fall war.

Alternative E-Mail-Clients

Neben den beschriebenen E-Mail-Clients Mozilla Thunderbird und Microsoft Outlook existieren im Umfeld des Windows-Betriebssystems weitere, frei verfügbare Clients. In Kapitel 18 finden Sie eine ausführliche Aufstellung, hier eine kleine Auswahl:

- | | | |
|---------------------|-----------------------|-------------|
| ✓ SeaMonkey | ✓ Opera Mail | ✓ Evolution |
| ✓ Hiri | ✓ Mailbird Lite | |
| ✓ Windows Live Mail | ✓ Mozilla Thunderbird | |

Diese weisen in den aktuellen Versionen umfangreiche Sicherheitsfeatures zur Abwehr von Malware auf.

Webbrowser sichern

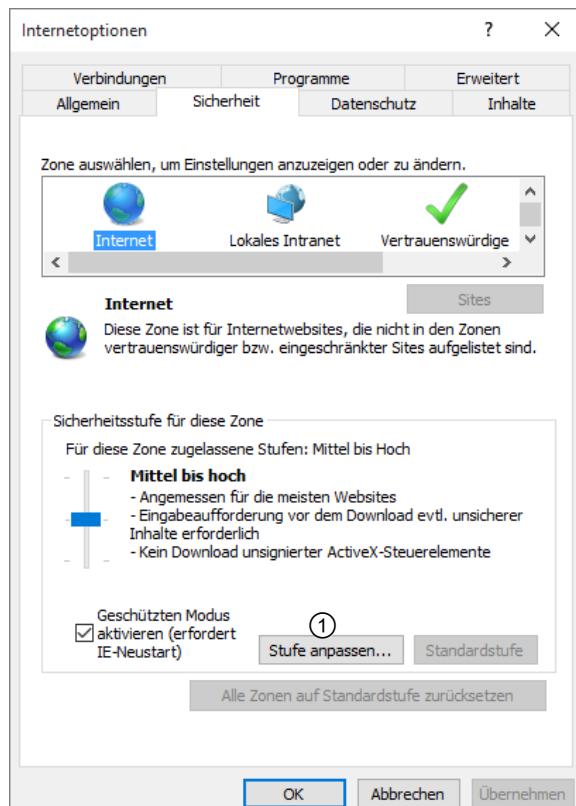
Um die Sicherheitseinstellungen Ihres Browsers zu optimieren, sollten Sie z. B. beim Internet Explorer die Einstellungen für die einzelnen Zonen überprüfen und gegebenenfalls anpassen. Standardmäßig fallen betrachtete Webseiten in die Zone „Internet“. Konfigurieren Sie die Einstellungen für die Zone eingeschränkter Sites so, dass keine aktiven Inhalte oder Interaktionen erlaubt werden.

Um die Einstellungen zu editieren, gehen Sie wie folgt vor:

- ▶ Klicken Sie im Internet Explorer auf *Extras - Internetoptionen*.
- ▶ Wählen Sie das Register *Sicherheit*.
- ▶ Markieren Sie die Zone, deren Einstellungen Sie ändern möchten und bestätigen Sie dies über *Stufe anpassen* ①.

Deaktivieren Sie alle Optionen, die Sie in der gewählten Zone nicht ermöglichen wollen.

Handelt es sich um die Zone für eingeschränkte Sites, ist es empfehlenswert, alle Optionen zu deaktivieren.



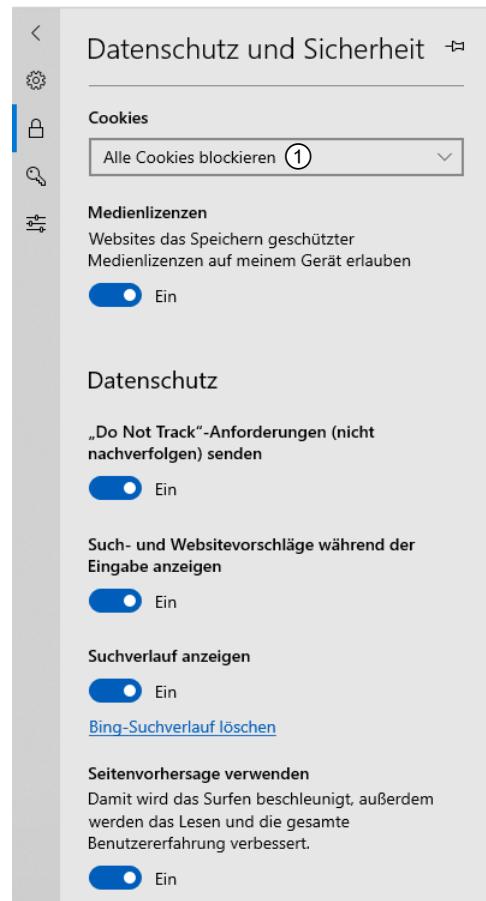
Sicherheitsoptionen des Internet Explorers

Alternativ zur Konfiguration des Internet Explorers (IE) könnten Sie in Erwägung ziehen, einen anderen Browser (wie z. B. Firefox, vgl. Kap. 18) zu verwenden, der gegenüber älteren IE-Versionen einen höheren Sicherheitsstandard aufweist.

Unter Windows 10 steht Ihnen der Browser Microsoft Edge zur Verfügung. Die Sicherheitseinstellungen können Sie wie folgt modifizieren:

- ▶ Öffnen Sie unter Explorer Edge den Menüpunkt *Einstellungen*
- ▶ Modifizieren Sie den Punkt *Datenschutz und Sicherheit* entsprechend der Abbildung rechts, um ein Optimum an Sicherheit zu erhalten.

Das Deaktivieren aller Cookies ① führt jedoch oft dazu, dass die Seitenanzeige komplett verhindert wird. Deshalb sollte die Option *Nur Cookies von Drittanbietern blockieren* gewählt werden.



Sicherheitseinstellungen Microsoft Edge
(Windows 10)

Gewährleistung der Privatsphäre

Eine Eigenschaft von Windows 10 ist es, regelmäßig Diagnosedaten an Microsoft zu senden. Der Hersteller begründet dies mit der Gewährleistung der Stabilität und der Versorgung mit Updates. Das dabei auch das Nutzerverhalten aufgezeichnet werden kann, liegt auf der Hand. Abschalten lässt sich die Telemetrie jedoch nicht vollständig. Es gibt aber Möglichkeiten, die Informationsflut der gesendeten Daten stark zu reduzieren.

Das BSI hat hierzu den Leitfaden „Analyse der Telemetrikkomponente in Windows 10, Konfigurations- und Protokollierungsempfehlung, Version: 1.1“ (https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Cyber-Sicherheit/SiSyPHus/Analyse_Telemetrikkomponente.pdf?__blob=publicationFile&v=5) herausgegeben. Sie enthält umfangreiche Empfehlungen zur Konfiguration bzw. der Abschaltung einiger Telemetriefunktionen.

8.2 Gängige Antivirensoftware

Bekannte Antivirusprogramme

Damit Sie Malware, wie Trojaner und Würmer, aufspüren und entfernen können, stehen Sie vor der Entscheidung, welches Antivirenprogramme Sie verwenden sollen.

Produktnname	Hersteller	Link
Avast Free Antivirus	AVAST Software	https://www.avast.com/de-de/index
AVG Antivirus FREE	AVG Technologies	https://www.avg.com/de-de/free-antivirus-download
Avira Free Security Suite	Avira	https://www.avira.com/de/free-security-suite
Bitdefender Antivirus Free Edition	Bitdefender	https://www.bitdefender.com/solutions/free.html
G Data Internet Security	G Data Software AG	https://www.gdata.de/
Kaspersky Internet Security	Kaspersky Lab	https://www.kaspersky.de/internet_security
McAfee Total Protection	McAfee LLC	https://www.mcafee.com/consumer/de-de/store/m0/index.html
Microsoft Defender Antivirus	Microsoft	https://www.microsoft.com/de-de/windows/comprehensive-security
Norton Internet Security	Symantec Corp.	https://de.norton.com/internet-security/
Panda Free Antivirus	Panda Security S. L.	https://www.pandasecurity.com/germany
Internet Security	Trend Micro Inc.	https://www.trendmicro.com/de_de/forHome.html
ZoneAlarm Internet Security Suite	Check Point Software Tech. Inc.	https://www.zonealarm.com/de/

Diese Tabelle erhebt keinen Anspruch auf Vollständigkeit. Auf dem Markt existiert eine Vielzahl von Antivirenprogrammen verschiedener Anbieter zu unterschiedlichen Konditionen bzw. teilweise in eingeschränktem Funktionsumfang als Freeware.

Die Entscheidung für ein bestimmtes Programm hängt von mehreren Faktoren ab, z. B. vom geplanten Einsatz. So mag ein Programm für eine Einzelrechner-Installation von Vorteil sein, während ein anderes speziell für den Einsatz im vernetzten Unternehmen entwickelt wurde. Auch der Einsatz auf verschiedenen Betriebssystemen bzw. die Nutzung zusätzlicher Systemtools können Einsatzkriterien sein.

Kriterien für die Auswahl eines Antivirusprogramms

Vor einer Kaufentscheidung für ein bestimmtes Antivirensystem sollten Sie mehrere Merkmale von infrage kommenden Produkten untersuchen und feststellen, ob diese Ihren Anforderungen entsprechen:

- ✓ Erkennungsquote
- ✓ Bereitstellung neuer Signaturen
- ✓ OnAccess/OnDemand -Funktionen
- ✓ Heuristiken
- ✓ Client-/Server-Funktionen
- ✓ Stabilität/Kompatibilität/Komfort/Performance
- ✓ Firewall-Applikation durch integrierte Firewall
- ✓ Kompatibilität zu anderen Betriebssystemen

Eine objektive Entscheidungshilfe erhalten Sie nicht durch Werbung in Fachzeitschriften, sondern nur über unabhängige Testlabore. Hier die Übersicht:

- ✓ Anti-Virus Test-LAB (Germany) (<https://www.av-test.org>)
- ✓ Anti-Virus Test-LAB (Austria) (<https://www.av-comparatives.org>)
- ✓ Anti-Virus Test-LAB SE Labs (UK) (<https://selabs.uk/>)
- ✓ Anti-Virus Test-LAB MRG (UK) (<https://www.mrg-effitas.com/>)
- ✓ Anti-Virus Test-LAB NSS (USA) (<https://www.nsslabs.com/>)
- ✓ Anti-Virus Test-LAB Bulletin (USA) (<https://www.virusbulletin.com/>)
- ✓ Anti-Virus Test-LAB (China) (<https://www.pitci.com/>)
- ✓ Anti-Virus+Application+Hardware Test-LAB (USA) (<https://www.icsalabs.com/>)

Erkennungsquote

Das wichtigste und gleichzeitig auch am schwersten zu beurteilende Merkmal eines Scanners ist die Erkennungsquote. Die Anbieter nutzen hierbei verschiedene Mechanismen zur Erkennung und Klassifizierung von Malware.

Die Hersteller bewerben Ihre Produkte gerne mit entsprechendem Zahlenmaterial – viele erhöhen jedoch die Zahl der erkannten Viren zusätzlich, indem Virenfamilien und Baukastenviren mit der Zahl ihrer erfassten Modifikationen und zum Teil auch experimentelle Viren mitgezählt werden.

Für den praktischen Einsatz ist eher relevant, dass Viren, die real in Umlauf sind (ITW, „in the wild“), zuverlässig erkannt werden. Wenn die Testresultate bei den ITW-Viren von einer 100%-Erkennungsquote abweichen, so sollten Sie einen genaueren Blick auf die nicht erkannten „Kandidaten“ werfen. Handelt es sich hierbei um die etwas seltener gewordenen Boot- oder Dateiviren, wäre ein nicht 100 %iges Abschneiden eines Scanners noch zu verschmerzen. Ergibt aber der Test bei den aktuell kursierenden Viren ein schlechtes Ergebnis, sollten Sie den Erwerb dieses Scanners überdenken.

Bereitstellung neuer Signaturen

Auch der Scanner mit der besten Erkennungsquote ist schnell nutzlos, wenn er mit dem Erscheinen neuer Malware nicht entsprechend zeitnah aktualisiert wird. Es kommt hierbei darauf an, wie schnell ein Antivirenhersteller für neu entwickelte Malware eine Signatur bereitstellen kann.

Sie sollten sich schon bei der Kaufentscheidung darüber informieren, mit welcher Häufigkeit der Hersteller entsprechende Updates herausgibt. Als Untergrenze für Computer (stand alone und vernetzt) ist heutzutage ein stündliches Update Pflicht.

Update-Abonnements

Viele Hersteller bieten dem Käufer von Antivirensoftware Signatur-Updates für 12 oder 24 Monate an. Anschließend muss entweder eine neuere Version der Software gekauft oder ein kostenpflichtiges Abonnement eingerichtet werden. Andere Hersteller (Freeware) geben Signatur-Updates ohne extra Lizenz oder Abonnement kostenfrei heraus. Hier sollten Sie jedoch bedenken, dass nicht alle Risiken standardgemäß abgedeckt werden. Das betrifft sowohl ggf. die Einschränkung beim E-Mail-Schutz, als auch beim Zugriff auf HTML-Seiten. Es kommt auch vor, dass Anbieter das automatische Update der Maleware-Signaturen nach einiger Zeit deaktivieren.

OnAccess-/OnDemand-Funktionen

Ein OnAccess-Modul, das bei Antivirenherstellern zum Standard gehört, wird während der Nutzung des Systems als Hintergrundprozess gestartet. Dabei werden sämtliche Dateizugriffe und ggf. auch eingehende E-Mails überwacht. Wird beim Versuch, eine Datei zu öffnen, Malware festgestellt, so kann das OnAccess-Modul zumindest den Zugriff auf die Datei verhindern oder im Optimalfall sogar die Malware entfernen.

Die Minimalausstattung von Antivirensoftware ist ein OnDemand-Scanner. Dabei handelt es sich um einen Scavorgang, der nur nach explizitem Aufruf durch den Benutzer gestartet wird und dann die gewünschten Datenträger auf Infektionen untersucht.

Obwohl sich diese Vorgehensweise für eine Komplettuntersuchung der geschützten PCs (z. B. nachts) eignet, können reine OnDemand-Scanner Infektionen immer nur im Nachhinein feststellen – meist ist der Schaden da schon angerichtet.

Heuristiken

Ein Scanner, der über heuristische (von griechisch „heuriskein“: finden, Erkenntnis gewinnen) Suchmethoden verfügt, versucht, eine Malware nicht anhand vorgegebener Signaturen zu identifizieren, sondern anhand von typischen virulenten Merkmalen gutartige Software von bösartiger zu unterscheiden. Es wird analysiert, was das Programm machen würde, wenn es gestartet werden würde. Ein Programm, das nach dem Start andere Programmdateien öffnet und etwas schreibt, ist ebenso verdächtig wie ein Programm, das auf den Master-Boot-Record der Festplatte zugreift. Auch Einsprungpunkte kurz vor dem Dateiende und anschließende Sprünge zurück in die Mitte einer ausführbaren Datei sind verdächtig.

Mit heuristischer Analyse ist der Scanner in der Lage, auch ohne Signatur-Update neue Viren anhand typischer Verhaltensmuster und Vorgehensweisen zu erkennen. Dies kann er jedoch nicht mit absoluter Sicherheit tun – je nach Qualität der implementierten Heuristik kann es zu Fehlalarmen kommen (z. B. das Partitionierungsprogramm FDISK.EXE von Microsoft ändert den Master-Boot-Record, muss aber nicht von einem Virus befallen sein), oder neue Viren können durch die Erkennung schlüpfen.

Zur Unterstützung der heuristischen Analyse arbeiten einige Hersteller von Antiviren auch mit künstlichen neuronalen Netzen, die aus dem Forschungsgebiet der künstlichen Intelligenz stammen. Bezuglich der Möglichkeit, unbekannte Viren zu erkennen, aber auch Fehlalarme zu erzeugen, sind neuronale Netze der herkömmlichen Heuristik gleichzustellen.

Ein idealer Scanner besitzt eine Heuristik, die Sie automatisch vor verdächtigen Dateien warnt, und eine Skalierbarkeit in der Erkennungstiefe, um Fehlalarme zu begrenzen.

Client/Server

Wenn Sie den Einsatz in einem Firmennetzwerk mit mehreren PC-Arbeitsplätzen planen, ist eine zentralisierte Verwaltung von Softwarekonfiguration, Signaturen und Virenmeldungen unabdingbar. Überprüfen Sie in solchen Fällen, ob eine Client/Server-Variante des geplanten Antivirus verfügbar ist und ob deren Kauf verwaltungsmäßig sinnvoller ist. Beim Scannen von Netzwerklauferwerken sollten Sie auch überprüfen, wie sich dies auf die Performance des Netzwerkes auswirkt.



Sofern Sie PCs im Heimbereich nutzen, sollten Sie auch darauf achten, ob der Hersteller der Antivirensoftware das Scannen von Netzwerklauferwerken (z. B. NAS) unterstützt.

Kompatibilität/Stabilität/Komfort/Performance

Vergewissern Sie sich, dass der Scanner mit Ihrem Betriebssystem vollständig kompatibel ist. Dies ist vor allem dann wichtig, wenn geänderte oder neue Betriebssystemarchitekturen auf den Markt kommen. Ein bisher verwendetes Antivirenprogramm muss nicht zwangsläufig die beste Wahl beim Wechsel der Betriebssystemplattform sein.

Falls eine kostenlose, zeitlich begrenzte Demoversion existiert, sollten Sie diese nutzen, um auch die Stabilität und den Komfort der Software zu testen. Ein nicht optimal programmiert OnAccess-Scanner, der jeden Dateizugriff signifikant verlangsamt, frustriert Benutzer nur und fordert dazu heraus, ihn abzuschalten. Dies kommt in der Praxis leider mitunter vor. Für im Hintergrund automatisch durchgeführte Scans sollte eine Zeitplanungsfunktion verfügbar und eine Begrenzung der Prozessorleistung obligatorisch sein. Auf diese Weise können Sie geplante Überprüfungen in einen Zeitraum legen, in dem ein Suchvorgang den Benutzer am wenigsten behindert (z. B. in der Mittagspause). Oder Sie können die CPU-Last des Scanners so weit reduzieren, dass der Benutzer während seiner Arbeit keine Einschränkungen feststellen kann.

8.3 Computer scannen

avast! Free Antivirus

Aufgrund der vielen unterschiedlichen Malware und der vielen Schadensfunktionen, die die Malware beinhalten kann, ist es mitunter nicht möglich, eine Programm-Fehlfunktion von den Symptomen durch Maleware zu unterscheiden.

Am Beispiel des für nichtkommerzielle Anwender kostenlos erhältlichen Scanners avast! (www.avast.com) lernen Sie im Folgenden, wie Sie einen Computer auf Virenbefall untersuchen.



Computer mit avast! überprüfen

Nach der Installation wird durch avast! Free Antivirus automatisch ein Wiederherstellungspunkt des Systems gesetzt und ein Scan durchgeführt.

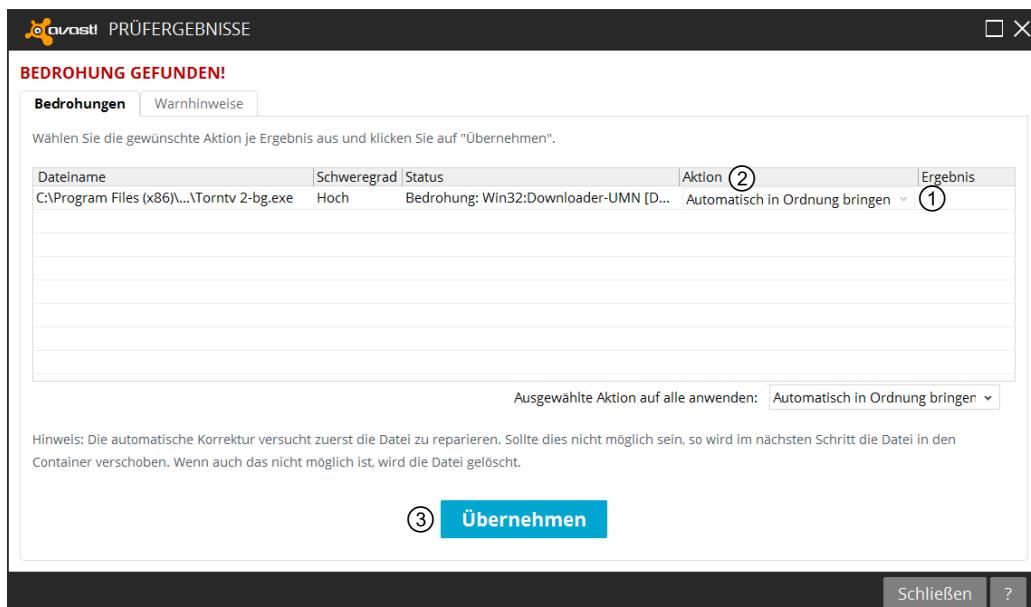
Sie können avast! jederzeit auch manuell als OnDemand-Scanner starten, indem Sie auf das Symbol des Programms klicken.

- ▶ Wählen Sie unter dem Menüpunkt *Überprüfung* aus, welche Medien bzw. Ordner überprüft werden sollen.
- ▶ Entscheiden Sie, ob eine schnelle oder vollständige Überprüfung stattfinden soll. Eine gründlichere Suche sucht zum Beispiel auch innerhalb von Archiven, beansprucht aber in der Regel auch mehr Zeit. Klicken Sie auf *Starten*, um mit dem Scannen zu beginnen.

Sie können während des Scanvorgangs sehen, welche Dateien avast! gerade untersucht. Bei gefährlichen Dateien unterbricht avast! die Suche und fordert Sie auf, eine Entscheidung zu treffen, was mit dieser Datei geschehen soll.

Am Ende des Scanvorgangs wird Ihnen eine Zusammenfassung angezeigt.

In diesem Beispiel wurde auf allen Laufwerken und in allen Ordnern eine vollständige Überprüfung durchgeführt, mit dem Ergebnis ①, dass bedrohliche Malware gefunden wurde. Über die eingestellte Aktion ② wird durch Klicken auf *Übernehmen* ③ die Aktion ausgelöst.



Scanergebnis

! Besteht der Verdacht, dass ein Computer mit einem unbekannten Virus befallen ist, sollten Sie zum Scannen eines solchen Systems einen virenfreien Datenträger zum Starten des Systems haben und eine virenfreie Kopie Ihres Scanners. Im ungünstigen Fall verhindert ein vor dem Scanner gestarteter Stealth-Virus, dass Ihre Antivirussoftware den Schädling jemals entdeckt.

Neben dem aktiven Schutz von Dateisystem, Mail-Client und Browser bietet avast! eine Deep-screen-Funktion an, welche eine bessere Echtzeitüberwachung ermöglicht. Die integrierte „stille“ Firewall und die Safezone, ein isolierter Bereich für sichere Transaktionen, sind einige der zusätzlichen Optionen.

Bootbare Antiviren-CD

Es gibt viele Freeware-Programme, die über ein Boot-Image und ein Antiviren-Tool eine Untersuchung des Systems ermöglichen. Die Hersteller haben sich dieser Entwicklung angeschlossen und bieten ein **startfähiges System samt VirensScanner und Online-Update an**. Voraussetzung hierfür ist eine bestehende Internetverbindung. Exemplarisch hier eine kleine Auswahl:

- | | |
|-------------------------|------------------------------|
| ✓ AntiVir Rescue System | ✓ ESET SysRescue Live |
| ✓ AVG Rescue CD | ✓ Kaspersky Rescue Disk 18 |
| ✓ Bitdefender Rescue CD | ✓ Windows Defender Offline |
| ✓ Desinfec't 2018/2019 | ✓ Sophos Bootable Anti-Virus |

8.4 Viren entfernen

Risiko beim Entfernen von Viren

Wenn Sie einen Virus gefunden haben, sollten Sie diesen nur durch Antivirensoftware entfernen lassen, es sei denn, Sie selbst verfügen über Detailwissen zu dem Virus und dessen Entfernung. Das einfache Löschen einer Virusdatei oder das Entfernen z. B. eines Bootsektorvirus mit dem Kommando `bootrec /fixboot` oder `bootrec /rebuildbcd` kann bei entsprechend programmierten Viren dazu führen, dass Sie das befallene System unbrauchbar machen und somit viel mehr Daten verlieren, als der Virus von sich aus möglicherweise beschädigt hätte.

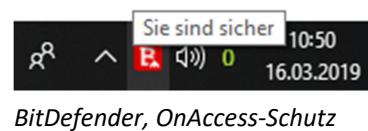
Hat ein Bootsektorvirus die Partitionsinformationen bei der Infektion verschlüsselt und an einen anderen Ort auf der Festplatte verschoben, verlieren Sie durch das Überschreiben des Virus mit der Option /MBR die Möglichkeit, diese Daten zurückzugewinnen. Unter Windows-Systemen könnte sich ein Dateivirus derart im Betriebssystem verankern, dass ein voreiliges Löschen ohne Korrektur der Registry-Einträge das Ausführen jeglicher Programme blockiert (inklusive des Registry-Editors, der zur Lösung dieses Problems nötig wäre).

Windows-Echtzeitschutz

Antivirusprogramme, die auf einer grafischen Benutzeroberfläche aufsetzen, sind auch für Laien deutlich einfacher zu bedienen und eignen sich eher für die tägliche Routineüberprüfung als Programme im Textmodus.

Die sicherste Methode, einen Windows-Scanner zu benutzen, ist die Aktivierung der Echtzeitüberprüfung des Scanners, sofern die Software diese Funktion unterstützt. Der Scanner bleibt so im Hintergrund aktiv und überprüft sämtliche vom Benutzer geöffneten, kopierten oder erstellten Dateien auf Malware.

Fast alle Scanner können bei der Installation ein entsprechendes Tray-Symbol über die Systemleiste einblenden, um die Aktivität des Echtzeit-Moduls zu signalisieren und um dem Benutzer einen schnellen Zugriff auf die Konfiguration des Scanners zu geben.



Sofern die Virensignaturen regelmäßig aktualisiert werden und der Echtzeitschutz aktiv ist, werden Viren hier schon vom Scanner abgefangen, bevor sie zur Ausführung gelangen. Je nach verwendeter Software kann eine bestimmte Aktion ausgeführt werden, sobald der Scanner einen Virus entdeckt. Übliche Optionen sind hier: „den Benutzer fragen“, „automatisch desinfizieren“, „Quarantäne“, „Umbenennen“, „Löschen“ oder „nur eine Warnung anzeigen“.

Während es ein etablierter Standard ist, dass ein Echtzeit-Virenschanner zumindest das Öffnen von infizierten Programmen verhindert, bieten viele Produkte auf dem Markt besonders komfortable Funktionen an, die alleine schon den Download von infiziertem Material verhindern.

Mit einem OnDemand-Scanner überprüfen

Soll ein Scanvorgang manuell durchgeführt werden, bieten die meisten Scanner eine Möglichkeit, die zu überprüfenden Laufwerke auszuwählen und besondere Optionen für die Suche zu verwenden.

Beispielsweise kann festgelegt werden, dass nicht nur Dateien durchsucht werden, die für Virenbefall typisch sind, sondern alle gefundenen Dateien. Diese Option würde den Suchvorgang drastisch verlängern.

Nach Abschluss der Suche sollte der Scanner ein übersichtliches Ergebnis präsentieren ① und im Idealfall auch hier eine Konfiguration der zu treffenden Aktionen erlauben.

Typ	Details
PUP.Optional.AdvancedSystemCare	1 Bedrohung erkannt
PUP.Optional.Legacy	7 Bedrohungen erkannt
PUP.Optional.SofTonicAssistant	6 Bedrohungen erkannt
PUP.Optional.SpeedItUpFree	2 Bedrohungen erkannt
PUP.Optional.TheBrightTag	4 Bedrohungen erkannt
PUP.Optional.TweakBit	4 Bedrohungen erkannt
PUP.Optional.WiseFolderLock	1 Bedrohung erkannt

OnDemand-Scan Malwarebytes

Im obigen Beispiel wurden einige PUP (potentiell unerwünschtes Programm) erkannt. Deren Schadpotential ist jedoch im Regelfall gering. Im Normalfall sind dies Toolbars, Nachrichtenbanner und das Sammeln von Nutzerinformationen für eine gezielte Bannerwerbung.

Je nach Beschaffenheit des Virus und Art der Infektion ist es in manchen Fällen nicht möglich, eine bestimmte Aktion durchzuführen.

8.5 Übung

Fragen zum Virenschutz

Übungsdatei: --

Ergebnisdatei: uebung08.pdf

1. Welche Entscheidungskriterien sollten Sie für den Einsatz einer Antivirensoftware beachten?
2. Welchen Vorteil bieten alternative Mail-Clients und Browser?
3. Wie sollten Sie beim Verdacht auf Malware reagieren?

9

IT-Sicherheitsstandard

9.1 Standards im Bereich Informationssicherheit

Der Aufwand für die Sicherung der IT-Ressourcen in einem Unternehmen kann mitunter sehr hoch sein, da traditionellerweise zuerst eine Analyse der vorhandenen schützenswerten Objekte (Assets) und eine Risiko- und Bedrohungsanalyse erfolgen muss. Danach werden die notwendigen Sicherheitsmaßnahmen ausgewählt, die zum Schutz der jeweiligen Assets für nötig erachtet werden.

Um den Zeit- und Arbeitsaufwand für die Sicherung reduzieren zu können und unternommene Sicherheitsbemühungen besser vergleichen zu können, werden in der Praxis oft Kriterienkataloge angewandt, die den Sicherheitsverantwortlichen bei seiner Arbeit unterstützen.

Die verschiedenen Kriterienwerke haben aber eine unterschiedliche Auslegung bezüglich der Anwendung, der verwendeten Methoden und der betrachteten Problemstellungen. Damit Sie entscheiden können, welche Kriterien für Ihre Aufgaben ideal sind, soll hier ein kurzer Überblick über die wichtigsten Kriterienwerke gegeben werden:

- | | |
|-----------------------------|---------------------------------------|
| ✓ IT-Grundschutz-Kompendium | ✓ Common Criteria/ITSEC/ISO/IEC 15408 |
| ✓ BSI-Standard 200 | ✓ ISO 9000 |
| ✓ ISO/IEC 13335 | ✓ COBIT |
| ✓ ISO/IEC 19790 | ✓ ITIL |
| ✓ ISO/IEC 2700X | ✓ DIN EN 50600 |

Basis für eine intensivere Auseinandersetzung mit IT-Sicherheitsstandards könnte der „Kompass-Sicherheitsstandards“ des Branchenverbandes BITCOM sein (<http://www.kompass-sicherheitsstandards.de/>).

9.2 IT-Grundschutz-Kompendium

Das IT-Grundschutz-Kompendium hat als Ziel, durch personelle, technische, organisatorische und infrastrukturelle Maßnahmen ein Standard-Sicherheitsniveau herzustellen, das auch für Bereiche mit höheren Sicherheitsansprüchen ausbaufähig ist. Das IT-Grundschutz-Kompendium löst seit Oktober 2017 als Nachfolger den IT-Grundschutz-Katalog ab. Der IT-Grundschutz besteht aus einigen Prozess-Bausteinen, Methodiken und Hilfsmitteln, die folgende Bereiche abdecken:

- ✓ Übergreifende Funktionen, Infrastruktur, IT-Systeme, Netze und Anwendungen, Sicherheitsmanagement (ISMS)
- ✓ Einfluss von höherer Gewalt, Mängel in der Organisation, menschliches Versagen, technische Mängel, Detektion und Reaktion
- ✓ Maßnahmen bezogen auf die Infrastruktur, die Organisation, das Personal, die Hard- und Software, die Kommunikation, die Vorsorge im Notfall
- ✓ Formulare, Mustervorlagen, Checklisten, IT-Grundschatzprofile und weitere Informationen

9.3 Weitere Kriterienwerke zur IT-Sicherheit

BSI-Standard 200

Diese Standards beinhalten grundlegende Methoden und Maßnahmen zur IT-Sicherheit und wurden vom Bundesamt für Sicherheit in der Informationstechnik (BSI) herausgegeben. Sie orientieren sich u. a. an den OSI-Standards ISO 2700x. Das IT-Grundschatz-Kompendium basiert auf diesen Standards. Durch die Möglichkeit, diese Standards zu zitieren, wird auch in methodischer Hinsicht eine Vereinheitlichung der IT-Sicherheitsbegriffe erzielt. Die IT-Grundschatz-Standards sind wie folgt eingeteilt:

- ✓ BSI-Standard 200-1: Managementsysteme für Informationssicherheit (ISMS)
- ✓ BSI-Standard 200-2: IT-Grundschatz-Vorgehensweise
- ✓ BSI-Standard 200-3: Risikoanalyse auf der Basis von IT-Grundschatz
- ✓ BSI-Standard 100-4: Notfallmanagement (Nachfolger wird BSI Standard 200-4)

ISO/IEC 13335

ISO/IEC 13335 besteht aus fünf technischen Berichten. Diese geben Hinweise zum IT-Sicherheitsmanagement, ohne eine bestimmte Lösung zu erzwingen. Die Normdokumente befassen sich mit folgenden Themenbereichen:

Teil 1, Konzepte und Modelle der IT-Sicherheit	<ul style="list-style-type: none"> ✓ Grundbegriffe der IT-Sicherheit ✓ Bedrohungen, Risiken, Schwachstellen ✓ Notfallvorsorge, Risikoanalyse, Sensibilisierung
Teil 2, Managen und Planen von IT-Sicherheit	<ul style="list-style-type: none"> ✓ Gestaltung von IT-Sicherheitsprozessen ✓ Integration in bestehende Unternehmensprozesse ✓ IT-Sicherheitsorganisationen
Teil 3, Techniken für das Management von IT-Sicherheit	<ul style="list-style-type: none"> ✓ Sicherheitsprozesse verfeinern ✓ Methoden und Techniken für Sicherheitsprozesse
Teil 4, Auswahl von Sicherheitsmaßnahmen	<ul style="list-style-type: none"> ✓ Schutzmaßnahmen gegen Bedrohungen
Teil 5, Management Guide für Netzwerksicherheit	<ul style="list-style-type: none"> ✓ Kommunikationssicherheit ✓ Netzwerktypen und Organisationen ✓ Business Continuity

ISO/IEC 19790 (FIPS 140-2)

Dieser Standard beschäftigt sich mit den Anforderungen an Kryptographiemodule. Der vom NIST (National Institute of Standards and Technology) herausgegebene „Federal Information Processing Standard 140“ befasst sich in der Version 2 mit der Überprüfung und Validierung von kryptografischen Modulen auf Hardware- und Softwarebasis. FIPS 140-2 ist im Standard ISO/IEC 19790 (Security requirements for cryptographic modules) aufgegangen.

ISO/IEC 2700X

ISO/IEC 2700X ist eine Familie internationaler Standards, die u. a. auf der nicht mehr gültigen ISO 17799 und dem British Standard BS 7799 aufbaut. Darin werden über 20 Normen definiert, in denen u. a. dem Anwender in einem Kriterienkatalog Best-Practice-Lösungen für die Informationssicherheit unter Berücksichtigung folgender Aspekte bereitgestellt werden:

- ✓ Regeln und Richtlinien zur Informationssicherheit
- ✓ Organisation von Sicherheitsmaßnahmen und Managementprozessen
- ✓ Personelle Sicherheit
- ✓ Asset-Management
- ✓ Physikalische Sicherheit und Zugangsdienste
- ✓ Zugriffskontrolle (Access Control)
- ✓ Umgang mit sicherheitstechnischen Vorfällen
- ✓ Systementwicklung und deren Wartung
- ✓ Planung einer Notfallvorsorge
- ✓ Einhaltung gesetzlicher Vorgaben und Überprüfung durch Audits

Common Criteria/ITSEC/ISO/IEC 15408

Die Common Criteria for Information Technology Security Evaluation (CC) fügen die in unterschiedlichen Wirtschaftszonen entstandenen Standards zueinander. Wie die ITSEC für Europa geben sie ein einheitliches Prüfverfahren vor, mit dem sicherheitsrelevante Aspekte von Hard- und Software so geprüft werden können, dass nachvollziehbare und vergleichbare Ergebnisse erzielt werden. Durch die Entwicklung des ISO/IEC 15408 sind die CC ein international anerkannter Standard. Dieser besteht aus drei Teilen:

- ✓ Teil 1: Einführung und allgemeines Modell (Introduction and general model)
- ✓ Teil 2: Funktionale Sicherheitsanforderungen (Security functional requirements)
- ✓ Teil 3: Anforderungen an die Vertrauenswürdigkeit (Security assurance requirements)

IT-Produkte und IT-Systeme können nach dem Standard, auch unter dem Namen „Common Criteria (CC)“ bekannt, zertifiziert werden. Im Rahmen der Zertifizierung wird die Sicherheit durch eine unabhängige Instanz (Prüfstellen, Zertifizierungsstellen und die nationalen Behörden) überprüft.

EN ISO 9000 Normenreihe

Die ISO-9000-Normreihe ist kein IT-Sicherheitsstandard, sondern definiert ein Prüfverfahren für Qualitätsmanagementsysteme. Da ein Qualitätsmanagementsystem gute Voraussetzungen für ein hohes Sicherheitsniveau bietet, ist die Nennung von ISO 9000 in diesem Zusammenhang durchaus angemessen.

Die Eigenschaften des Qualitätsmanagementsystems müssen dokumentiert und sowohl durch interne als auch externe Stellen nachvollziehbar sein.

In diesem Rahmen wird auch überprüft, ob die EDV-Ausstattung und -Organisation dem Unternehmenszweck angemessen sind.

COBIT

COBIT (Control Objectives for Information and Related Technology) ist ein internationales Framework, das die Aufgaben der IT in Prozesse und Kontrollziele aufgliedert. Primär geht es hierbei aber nicht darum, **wie** etwas umzusetzen ist, sondern **was** umzusetzen ist. Es werden also Ergebnisse und nicht Wege zum Ziel beschrieben.

Von einem Werkzeug, das früher nur von Auditoren eingesetzt wurde, hat es sich zu einem Werkzeug entwickelt, das nun dazu dienen kann, die Informationstechnologie einer Firma aus Unternehmenssicht zu steuern. Es wird auch benutzt, um die Einhaltung gesetzlicher Anforderungen an das Unternehmen modellieren zu können.

Prinzipiell wird hierbei ein „Top-Down“-Ansatz verfolgt, d. h. ausgehend von den Unternehmenszielen werden Ziele der IT-Infrastruktur abgeleitet und daraus wiederum alle Kriterien der IT beeinflusst.

ITIL

Auch die ITIL (IT Infrastructure Library) ist kein IT-Sicherheitsstandard. In ihr werden Regeln und Definitionen für den Betrieb einer gesamten IT-Infrastruktur (IT Service Management – ITSM) anhand von sogenannten Best Practices vorgegeben. Im Grundansatz beschreibt die ITIL darin einen ITSM-Lifecycle (Lebenszyklus). In einem Regelkreis wird die kontinuierliche Verbesserung der Prozesse, der Organisation und der entsprechenden Werkzeuge und somit der gesamten IT-Infrastruktur beschrieben.

Dabei gliedert sich die ITIL in fünf Bücher, die gleichzeitig die fünf Lebensphasen des ITSM-Lebenszyklus darstellen:

- ✓ Service Strategy (Servicestrategie)
- ✓ Service Design (Serviceentwurf)
- ✓ Service Transition (Serviceüberführung)
- ✓ Service Operation (Servicebetrieb)
- ✓ Continual Service Improvement (Kontinuierliche Serviceverbesserung)

Jegliche Tätigkeit im IT-Umfeld kann so mithilfe eines standardisierten Vorgehens angegangen werden. Das betrifft selbstverständlich auch den IT-Sicherheitsprozess.

Zielgruppen

Die vorgestellten Kriterienwerke haben eine unterschiedliche Zielsetzung und sind daher eher für den Einsatz in bestimmten Zielgruppen geeignet. Obwohl einzelne Kriterienwerke durchaus auch als Hilfsmittel für andere Anwendungsbereiche nützlich sein können, liegt die Priorität der Anwendung einzelner Werke in folgenden Bereichen:

		IT-Grundschutz-Kompendium	BSI 200	ISO TR 13335	ISO 19790	ISO 2700X	Common Criteria	ISO 9000	COBIT	ITIL
Unternehmenskategorie	Hardware-Hersteller				x			x		x
	Software-Hersteller		x		x		x	x		x
	Server-Betreiber	x	x			x		x	x	x
	Netzprovider	x	x			x		x	x	x
	Content-Provider	x	x			x		x	x	x
	Unternehmen (als Kunde)	x		x		x		x		
Personalrolle	IT-Leitung	x	x	x		x			x	x
	Administrator	x	x							x
	IT-Sicherheits-/ Datenschutzbeauftragter	x	x	x	x	x	x			x
	Management			x		x		x	x	x
	Projektmanager	x	x	x	x	x	x	x		x

x = Haupt-Zielgruppe

9.4 DIN EN 50600

Standardisierung für die Infrastruktur von Rechenzentren

Auf die Initiative der CENELEC (Comité Européen de Normalisation Electrotechnique) wurde eine europaweite Norm für alle Einrichtungen und Infrastrukturen eines Rechenzentrums geschaffen. Es wurden dabei die Aspekte wie RZ-Auslegung, RZ-Integration, Installationen und Instandhaltung von Einrichtungen und Infrastrukturen innerhalb von Rechenzentren länderübergreifend vereinheitlicht.

Die DIN EN 50600 beinhaltet auch gewisse Freiheitgrade (für unterschiedliche RZ-Konzepte) und ist als normierter Leitfaden und Baukastensystem zu verstehen. Die Norm besteht aus den folgenden Einzelbausteinen:

- ✓ DIN EN 50600-1 Informationstechnik-Einrichtungen und Infrastrukturen von Rechenzentren: Allgemeine Konzepte
- ✓ DIN EN 50600-2 Gebäudekonstruktion, Stromversorgung, Regelung der Umgebungsbedingungen, Infrastruktur der Telekommunikationsverkabelung, Sicherungssysteme

- ✓ DIN EN 50600-3 Informationen für das Management und den Betrieb
- ✓ DIN EN 50600-4 Überblick über und allgemeine Anforderungen an Leistungskennzahlen, Kennzahl zur eingesetzten Energie, Anteil erneuerbarer Energien

Die DIN EN 50600-X unterstützt u. a. Betreiber, Planer, IT-Management, Facility-Manager bei der Planung, der Realisierung und dem Betrieb von Rechenzentren. Sie ermöglicht eine auf die Bedürfnisse des Nutzers maßgeschneiderte, normkonforme Auslegung der technischen Gewerke.

9.5 Security Policy

Warum eine Security Policy gebraucht wird

Eine Security Policy oder auch Sicherheitsrichtlinie wird gebraucht, weil es sonst keinen strukturierten Plan und keine Handlungsvorschriften gibt, wie welche Systeme und Komponenten sicher zu machen sind. Ein Unternehmen, das über keine Security Policy verfügt, wird ziemlich wahrscheinlich an irgendeiner Stelle IT-Sicherheitsprobleme bekommen. In solchen Fällen wird das Problem eventuell durch eine Einzellösung behandelt, und es wird bis zum Auftreten des nächsten Sicherheitsproblems gewartet.

Beispiele rein reaktiven Verhaltens, die mitunter in der Praxis anzutreffen sind:

- ✓ Ein Computervirus verursacht Schäden an den Datenbeständen an einem vermeintlich sicheren System. Nach Beseitigung wird eine Lizenz eines Antivirusprogramms gekauft oder das vorhandene durch eine andere Version ersetzt.
- ✓ Die Computer einer Firma sind vernetzt und an das Internet angebunden. Eines Tages stellt die Telefongesellschaft eine exorbitante Telefonrechnung zu. Der verantwortliche 0900-Dialer wird erst nach dem Schadensfall gesucht und gefunden. Anschließend wird nach einem Schutz gesucht.
- ✓ Hardware im Serverraum fällt grundsätzlich nach relativ kurzer Nutzungsdauer wegen elektrischer Defekte aus. Es wird zwar neue Hardware beschafft, aber trotz nachweislich zu hoher Lufttemperatur im Serverraum (z. B. 30° C und höher) lehnt das Management die Anschaffung einer Klimaanlage ab.

Was in einer Security Policy steht

Eine sinnvolle Sicherheitsrichtlinie legt fest, was getan werden muss, um ein IT-System und die gespeicherten Informationen zu schützen. Mit ihrer Hilfe können Mitarbeiter leicht entscheiden, was und wie es zu tun ist.

Im Prinzip ist eine Sicherheitsrichtlinie also nichts anderes als eine schriftlich niedergelegte Strategie, in der beschrieben wird, wie ein Computernetzwerk und seine Ressourcen zu schützen sind.

Die schriftliche Fixierung erlaubt es, Maßnahmen und auch Notfallpläne vorab zu definieren, sodass Sicherheitsprobleme minimiert werden können und bei deren Auftreten schnell und zielsicher gehandelt werden kann.

Wie eine Security Policy entsteht

Eine Security Policy von Grund auf zu entwerfen ist ohne Hilfsmittel ein aufwendiges Unterfangen. Die von verschiedenen Gremien herausgegebenen Kriterienkataloge geben gute Hinweise und bieten (wie beim IT-Grundschutz-Kompendium) auch einen modularen Ansatz, der es ermöglicht, komponentenweise die Sicherheitsrichtlinien für das eigene Unternehmen zu definieren und gegebenenfalls die im Kriterienwerk vorgeschlagenen Empfehlungen anzupassen.

In einer kompletten Security Policy darf nicht nur der Maßnahmenkatalog zur Sicherung der Unternehmens-Assets enthalten sein, sondern es sollten auch die notwendigen Rollen und Verantwortlichkeiten den Mitarbeitern zugewiesen werden.

Zusätzlich ist es nötig, in der Policy selbst zu definieren, welche Sanktionen es nach sich zieht, wenn gegen die in der Policy festgelegten Richtlinien verstoßen wird, und die fertige Policy auch bei allen Mitarbeitern bekannt zu machen – nur so kann sie im Unternehmenseinsatz auch wirksam werden.

Eine Policy ist im Prinzip nie ganz fertig. Sicherheit ist immer ein Prozess, nie ein Produkt. Aus diesem Grund ist es nötig, die Wirksamkeit der Richtlinien ständig zu prüfen und zu überarbeiten oder an aktuelle Vorfälle anzupassen. Die ständige Bewertung von Assets und die Beurteilung von Risiken zu dem Zweck, angemessene Maßnahmen in einer Security Policy treffen zu können, wird als aktives Risikomanagement verstanden.

Auf der Webseite <https://www.sicher-im-netz.de/dsin-sicherheitscheck> finden Sie den „DsiN-Sicherheitscheck“, der für kleine Unternehmen geeignet ist, ihre Security Policies zu überprüfen.

9.6 Aufgaben eines IT-Sicherheitsbeauftragten

Der IT-Sicherheitsbeauftragte ist für die Umsetzung der Security Policies in einem Unternehmen verantwortlich. Um dieser Aufgabe gerecht zu werden, muss er der Geschäftsführung direkt unterstellt werden und darf nicht in die operative IT-Administrierung involviert sein. Die Rolle des Sicherheitsbeauftragten wird u. a. im IT-Grundschutzkatalog des BSI definiert. Diese umfasst:

- ✓ die unternehmensweite Verantwortung für die Erstellung, Entwicklung und Kontrolle der Sicherheitsrichtlinien,
- ✓ die Berichtspflicht aller Maßnahmen zur IT-Sicherheit gegenüber der Geschäftsführung und den Mitarbeitern,
- ✓ die Koordination der IT-Sicherheitsziele mit den Unternehmenszielen und Abstimmung mit den einzelnen Unternehmensbereichen,
- ✓ die Festlegung der Sicherheitsaufgaben für die nachgeordneten Unternehmensbereiche
- ✓ die Weisungsbefugnis in Fragen der IT-Sicherheit,
- ✓ die Kontrolle der IT-Sicherheitsmaßnahmen auf Korrektheit, Nachvollziehbarkeit, Fortschritt und Effektivität,
- ✓ die Koordination von unternehmensweiten Ausbildungs- und Sensibilisierungsprogrammen für die Mitarbeiter.

9.7 Übung

Fragen zu Standards im Bereich IT-Sicherheit

Übungsdatei: --

Ergebnisdatei: uebung09.pdf

1. Nennen Sie die wichtigsten IT-Sicherheitsstandards.
2. Welche Tools können für die Umsetzung des IT-Grundschutz-Kompendiums verwendet werden?

10

Symmetrische Kryptografie

10.1 Das Problem von Alice und Bob

Was ist Kryptografie?

Der Begriff **Kryptografie** und die Bezeichnung für die verwandten Disziplinen **Kryptologie** und **Kryptoanalyse** stammen aus dem Griechischen. „Kryptos“ bedeutet so viel wie „geheim“ oder „verborgen“. „Graphein“ steht für „schreiben“. Die Endung „-analyse“ stammt von „analysein“, deutsch „entziffern“. „Logos“ bedeutet „Sinn“. Somit lassen sich die drei Disziplinen wie folgt unterteilen:

- ✓ Kryptografie: die Wissenschaft der Geheimschrift
- ✓ Kryptoanalyse: die Kunst, Geheimschrift (unbefugt) entziffern zu können, den Code zu brechen
- ✓ Kryptologie: die Wissenschaft, die Kryptografie und Kryptoanalyse miteinander vereint

Ziel der Kryptografie ist es, Nachrichten in eine Art „Geheimschrift“ zu übersetzen und diese so auf einem möglicherweise unsicheren Weg zum vorgesehenen Empfänger zu schicken. Die für die Nachricht gewählten Zeichen sollen so gewählt sein, dass nur der vorbestimmte Empfänger in der Lage sein sollte, den Inhalt der Nachricht wieder lesbar zu machen.

Die Verwendung von geheimen Zeichen ist für die Kryptografie nicht notwendig und auch nicht sinnvoll. Die Zeichen des **Klartextes** sind dieselben wie die des **Chiffretextes**. Bei den historischen Verschlüsselungsverfahren wurden die Zeichen des Alphabets verwendet, bei den modernen, computergestützten Verfahren werden Gruppen binärer Zustände (meist Bytes oder Blöcke aus mehreren Bytes) eingesetzt.

Akteure in der Kryptologie

In der Kryptologie hat es sich eingebürgert, Problem- oder Protokollbeschreibungen nicht abstrakt vorzunehmen: „A möchte B eine verschlüsselte Nachricht zusenden, C versucht, diese Nachricht abzufangen und deren Inhalt zu entschlüsseln.“

Vielmehr werden in der kryptologischen Literatur Akteure mit Vornamen genannt, deren Anfangsbuchstabe Ihnen Aufschluss über die Rolle des Akteurs im jeweiligen Beispiel gibt. Aus den Hauptteilnehmern A und B werden auf diese Weise Alice und Bob.

Häufig verwendete Akteure

Alice	Hauptakteurin A, startet einen Vorgang
Bob	Hauptakteur B, ist Nutznießer des Vorgangs
Eve	Lauscherin, E = Eavesdropper
Mallory	Böswilliger, aktiver Angreifer, M = Malicious

Seltener vorkommende Akteure

Carol	Hauptakteurin C, dritte Teilnehmerin an einem kryptografischen Protokoll
Dave	Hauptakteur D, vierter Teilnehmer
Trent	Vertrauenswürdiger Vermittler, T = Trust
Walter	Ein Wächter, der Alice und Bob beschützt, W = Warden
Sara	Ein Server

Informationen und Schlüssel

Die grundlegende Herausforderung der Kryptografie ist es, eine Nachricht so zu verschlüsseln, dass deren Inhalt während des Transports vor der Kenntnisnahme durch Unbefugte geschützt ist. Der rechtmäßige Empfänger der Nachricht muss aber in der Lage sein, diese zu dechiffrieren, also die Verschlüsselung wieder rückgängig zu machen.

Damit dies möglich ist, benötigt der Empfänger eine Zusatzinformation – einen Schlüssel. Ein ideales Verschlüsselungsverfahren ist so sicher, dass es nur mithilfe des passenden Schlüssels möglich ist, an den Inhalt der Nachricht zu kommen.

Kryptografische Verfahren, bei denen Sender und Empfänger denselben Schlüssel zum Ver- wie auch zum Entschlüsseln verwenden, werden **symmetrische Verschlüsselungsverfahren** genannt.

Kerckhoffs' Forderungen

Der niederländische Philologe Kerckhoffs von Nieuwenhof (1835–1903) formulierte sechs Prinzipien für militärische Verschlüsselungsalgorithmen. Von diesen sind die wichtigsten:

- ✓ Das System selbst darf nicht geheim sein, es sollte kein Problem darstellen, wenn es in die Hände des Feindes gelangt.
- ✓ Der Schlüssel muss ausreichend klein, variabel und modifizierbar sein.

Vor allem die Implikationen der ersten Forderungen sind als das **Kerckhoffs'sche Prinzip** in der Öffentlichkeit bekannt. Ein sicheres System muss auch noch sicher sein, wenn der Algorithmus bekannt ist, mit dem gearbeitet wird. Die Sicherheit hängt also einzig und allein vom verwendeten Schlüssel ab.

Ein Verfahren, das sich an dieses Prinzip hält, genießt mehrere Vorteile:

- ✓ Zwischen Alice und Bob kann der Algorithmus offen bekannt gegeben werden, und auch Eve weiß, wie er funktioniert.
- ✓ Alice und Bob brauchen **nur einen** gemeinsamen geheimen Schlüssel, den sie niemand anderem mitteilen. Der Transport des Schlüssels ist relativ leicht, da er im Vergleich zur Nachricht oder den Instruktionen für den Algorithmus relativ klein ist.
- ✓ Dadurch, dass der Algorithmus öffentlich bekannt ist, haben viele Kryptoanalytiker die Möglichkeit, nach Schwachstellen und Angriffspunkten zu suchen. Sollte eine gefunden werden, kann über Lösungen diskutiert werden. Algorithmen, die öffentlich bekannt und anerkannt sind, gelten als relativ sicher, da viele Analytiker vergeblich versucht haben, eine Schwachstelle zu finden.
- ✓ Eve kann die Nachricht nur entziffern, wenn sie in den Besitz des Schlüssels gelangt.

Einige Institutionen und Firmen beherzigen dieses Prinzip trotz der offensichtlichen Vorteile nicht. Sie versuchen, ein höheres Sicherheitsniveau durch die Geheimhaltung der Algorithmen zu erreichen. Wenn die Algorithmen in Hardware implementiert sind, bedeutet das: Gelangt ein Angreifer in den Besitz der Verschlüsselungsmaschine, so ist das ein wesentlicher Schritt zum Knacken des Codes. Ein historisches Beispiel hierfür ist die kriegsentscheidende Erbeutung einer deutschen Enigma-Maschine durch die alliierten Streitkräfte.

Auch in der Software wird oft auf Geheimhaltung der Algorithmen gesetzt. Diese Verschleierungstaktik wird auch „Security through obscurity“ genannt und ist ebenfalls relativ wirkungslos. Jegliche Software kann disassembliert und somit auf Sourcecode-Ebene analysiert werden. Mit entsprechendem Zeit- und Arbeitsaufwand ist es also jedem Angreifer möglich, Kenntnis über die vermeintlich versteckten Algorithmen zu erlangen. Ein Beispiel hierfür ist die DVD-Verschlüsselung CSS, die gebrochen wurde, indem eine DVD-Playersoftware per Debugger analysiert wurde. Damit wurden Vendor-Key und Verschlüsselungsalgorithmus herausgefunden.

10.2 Einfache Verschlüsselungsmethoden

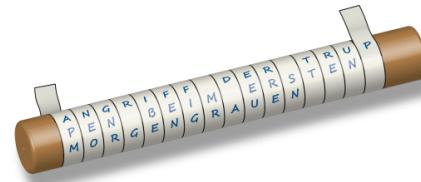
Stabchiffre

Eine der ältesten überlieferten Verschlüsselungsmethoden ist die von den Griechen eingesetzte sogenannte Stabchiffre (Skytale). Hierbei wird zur Übermittlung von Nachrichten zwischen dem Befehlshaber und seinen Truppen ein Band der Länge nach um einen Stab mit einem bestimmten, vorher festgelegten Durchmesser gewickelt. Die Nachricht wird entlang der Achse des Stabes niedergeschrieben. Zum Transport an den Empfänger wird das so beschriebene Band abgewickelt und zusammengerollt.

Würde der Feind den Boten abfangen und das Band betrachten, so sähe er nur scheinbar sinnlose Buchstabenkombinationen, solange er das Verschlüsselungsprinzip und den Durchmesser des Stabes nicht kennt. Ein autorisierter Empfänger dieser Nachricht besitzt jedoch einen Stab mit identischem Durchmesser. Wickelt der Empfänger das Band um diesen Stab, so kann er die ursprüngliche Nachricht wieder lesen. Der Durchmesser des verwendeten Stabes ist hier also in gewisser Weise der Schlüssel für dieses System.

Schreibt der Feldherr z. B. einen Befehl für seine Truppen in dreizeiliger Form auf das Band:

Angriff der Truppen beim ersten Morgengrauen



Eine Skytale

So ist auf dem abgewickelten Band Folgendes zu lesen:

ApMneognrr gibefenfig mrd aeeurre snTt re un p

Die Stabchiffre nimmt also nur eine Umverteilung der Buchstaben vor. Die Zeichen selbst und deren Häufigkeit bleibt erhalten, nur die Reihenfolge ändert sich (Transposition).

Caesar-Chiffre

Der römische Feldherr Caesar setzte zur Kommunikation mit seinen Generälen eine Methode ein, die später auch nach ihm benannt wurde. Cicero lobte Caesar in seinen Schriften dafür, dass er eine absolut sichere Methode gefunden habe, geheime Nachrichten zu verfassen.

Caesars Methode bestand darin, in einer Nachricht jeden Buchstaben durch denjenigen Buchstaben zu ersetzen, der drei Stellen später im Alphabet folgt. Wollte jemand eine derartige Nachricht lesen, so musste er im Chiffretext jeden Buchstaben durch den um drei Stellen vorgestellten Buchstaben ersetzen.

Caesar verschob also das Alphabet des Chiffretextes im Vergleich zum Klartextalphabet. Im Gegensatz zu einer Chiffre, die auf Änderung der Reihenfolge beruht (Transpositionschiffre), ist die Caesar-Chiffre also eine Rotationschiffre (ROT).

Die Zuordnung von Klartextbuchstaben zu Buchstaben im Chiffretext sieht nach Caesar so aus:

Klartext	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Chiffre	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Der Buchtitel **DE BELLO GALLICO** mit der Caesar-Chiffre verschlüsselt lautet: **GH EHOOR JDOOLFR.**

Die Caesar-Chiffre ist eigentlich ein Sonderfall. Caesar nahm zum Chiffrieren immer oben stehende Tabelle – also immer denselben Schlüssel.

Der etwas allgemeinere Fall ist die Rotationschiffre. Wenn Sie obige Tabelle betrachten, so können Sie die beiden Alphabete um 1 bis 25 Buchstaben gegeneinander verschieben. Sie haben also 25 verschiedene Schlüssel. Der Schlüssel 26 – eine Verschiebung um 0 Zeichen, würde keinen Sinn machen, da dann der Klar- und der Chiffretext identisch wären.

Eine Zahl zwischen 1 und 25 stellt also den Schlüssel dar, den Alice zu Bob übertragen muss, damit dieser eine entsprechend chiffrierte Nachricht wieder lesbar machen kann.

Kryptoanalyse von ROT-Verschlüsselung

Selbst wenn Sie sich nicht wie Caesar immer auf den Schlüssel 3 (ROT-3-Verschlüsselung) festlegen, sondern einen beliebigen Schlüssel zwischen 1 und 25 wählen, ist es relativ einfach, die Klartextnachricht aus dem Chiffretext zu ermitteln. Da es nur 25 mögliche Schlüssel gibt, reicht es, der Reihe nach sämtliche Schlüssel auf den Chiffretext anzuwenden (Brute-Force-Angriff). Nur in einer Zeile wird dann sinnvoller Text stehen.

Eve möchte wissen, welche Nachricht sich hinter der Chiffre **PUALYULADVYRPUNZPJOLYOLPA** versteckt, die mit ROT verschlüsselt wurde. Dazu erstellt sie eine Tabelle und probiert alle Schlüssel durch.

ROT-1:	OTZKXTKZCUXQOTMYOINPKXNK0Z	ROT-14:	BGMXKGXMPHKDBGZLBVAXKAXBM
ROT-2:	NSYJWSJYBTWPNSLXNHMJWMJNY	ROT-15:	AFLWJFWLOGJCAFYKAUZWJJZWAL
ROT-3:	MRXIVRIXASVOMRKWMGLIVLIMX	ROT-16:	ZEKVIEVKNFIBZEXJZTYVIYVZK
ROT-4:	LQWHUQHWZRUNLQJVLFKHUKHLW	ROT-17:	YDJUHDUJMEEHAYDWIYSXUHXUYJ
ROT-5:	KPVGTPGVYQTMKPIUKEJGTJGKV	ROT-18:	XCITGCTILDGZXCVHXRWTGWTXI
ROT-6:	JOUF5OFUXPSLJOHTJDIFSIFJU	ROT-19:	WEHSFBSHKCFYWBUGWQVSFVSWH
ROT-7:	INTERNETWORKINGSICHERHEIT	ROT-20:	VAGREARGJBEXVATFVPUREURVG
ROT-8:	HMSDQMDSVNQJHMFRHBGDQGDHS	ROT-21:	UZFQDZQFIADWUZSEUOTQDTQUF
ROT-9:	GLRCPLCRUMPIGLEQGAFCPFCGR	ROT-22:	TYEPCTYPEHZCVTYRDTNSPCSPTE
ROT-10:	FKQBOKBQTLOHFKDPFZEBOEBFQ	ROT-23:	SXD0BXODGYBUSXQCMSMROBROSD
ROT-11:	EJPANJAPSKNGEJCOEYDANDAEP	ROT-24:	RWCNAWNCFXATRWPBRLQNAQNRC
ROT-12:	DIOZMZIORJMFDIBNDXCZMCZDO	ROT-25:	QVBMZVMBEWZSQVOAQKPMZPMQB
ROT-13:	CHNYLHYNQILECHAMCWBYLCN		

Wie Sie sehen, ist die einzige Zeile, die einen Sinn ergibt, diejenige, in der von einer Verschlüsselung mit ROT-7 ausgegangen wurde. Der Klartext lautet „INTERNETWORKINGSICHERHEIT“. ROT ist wegen der geringen Anzahl möglicher Schlüssel („Größe des Schlüsselraums“) als nicht sicher einzustufen.

Dieses Beispiel zeigt, warum eine Brute-Force-Attacke für potenzielle Angreifer immer unrentabel wird, je größer der Schlüsselraum bei einer bestimmten Methode wird: Er müsste sehr viel Zeit in das Durchprobieren der Schlüssel investieren.

In der Regel gilt eine Methode für einen bestimmten Zweck dann als sicher, wenn davon ausgegangen werden kann, dass ein Brute-Force-Angriff extrem lange dauern würde. Übersteigt der Zeit- oder Kostenaufwand dann den Wert der verschlüsselten Information oder kann der Code erst zu einem Zeitpunkt geknackt werden, wenn die geschützte Nachricht bereits wertlos geworden ist, gilt das verwendete Verfahren als sicher. Beispielsweise ist die verschlüsselte Kommunikation bezüglich einer Firmenübernahme für einen Angreifer (z. B. für einen Aktienhändler, der Geschäfte mit diesen Insiderinformationen machen möchte) wertlos, wenn die Übernahme inzwischen stattgefunden hat.

Häufigkeitsanalyse

Eine andere Methode, ROT-Code zu brechen, besteht darin, die Häufigkeitsverteilung der vor kommenden Buchstaben auszuwerten. Wenn die verschlüsselte Nachricht eine natürliche Sprache ist, so treten die einzelnen Buchstaben mit unterschiedlicher Häufigkeit auf. In der deutschen Sprache ist der Buchstabe E der häufigste. Will Eve nun den Schlüssel zu einer ROT-Nachricht errechnen, ohne auf einen Brute-Force-Angriff angewiesen zu sein, zählt sie die Häufigkeit jedes Buchstabens im Chiffretext. Wenn dieser ausreichend lang ist, ergibt sich eine Verteilung der Häufigkeiten, die annähernd der Verteilung der vermuteten Sprache entspricht. Ist der am häufigsten aufgetretene Buchstabe im Chiffretext ein R, so wird sie annehmen, dass E zu R verschlüsselt worden ist. Der Abstand von E zu R im Alphabet ist 13 – somit versucht sie eine Entschlüsselung mit der Annahme, dass mit ROT-13 verschlüsselt wurde. War die Annahme richtig, so kann der gesamte Text entschlüsselt werden.

Auftretenswahrscheinlichkeiten der Buchstaben in Prozent

Zeichen:	a	b	c	d	e	f	g	h	i	j	k	l	m
Deutsch	6.47	1.93	2.68	4.83	17.48	1.65	3.06	4.23	7.73	0.27	1.46	3.49	2.58
Englisch	8.04	1.54	3.06	3.99	12.51	2.30	1.96	5.49	7.26	0.16	0.67	4.14	2.53

Zeichen:	n	o	p	q	r	s	t	u	v	w	x	y	z
Deutsch	9.84	2.98	0.96	0.02	7.54	6.83	6.13	4.17	0.94	1.48	0.04	0.08	1.14
Englisch	7.09	7.60	2.00	0.11	6.12	6.54	9.25	2.71	0.99	1.92	0.19	1.73	0.09

ROT-13

Die Rotationsverschlüsselung mit dem Schlüssel 13 hat eine besondere Stellung bei den Rotationschiffen. Da das Alphabet 26 Buchstaben hat, bedeutet eine Verschiebung um 13 Stellen eine Verschiebung um genau die Hälfte des Alphabets. Wird ein so erzeugter Chiffretext erneut ROT-13 verschlüsselt, so wird wiederum um die Hälfte verschoben – was dazu führt, dass der Chiffretext wieder dem Klartext entspricht.

Die ROT-13-Chiffre ist in vielen gängigen E-Mail- und Newsreader-Programmen integriert und erlaubt es dem Benutzer, mit einem Tastendruck den gewählten Text zu verschlüsseln. Als Einsatzgebiet ist allerdings nicht die Vertraulichkeit vorgesehen – das wäre sinnlos, da jeder Unbefugte mit ROT-13-fähiger Software leicht in der Lage wäre, den Klartext zu lesen. ROT-13 wird verwendet, um versehentliches Lesen eines Textes beim Empfänger zu verhindern.

Vigenère-Chiffre

Das Problem der ROT-Chiffre ist, dass die Zeichen zwar durch andere ersetzt werden, die Häufigkeit von einzelnen Zeichen und die Reihenfolge des Chiffrealphabets aber erhalten bleibt. In der ROT-Chiffre gibt es nur ein Chiffrealphabet. Häufige Zeichen werden also immer durch dieselben anderen Zeichen ersetzt, die dann genauso häufig sind.

Ziel einer Verbesserung wäre es also, eine Verschlüsselung in der Art vorzunehmen, dass im Ergebnis alle Buchstaben gleich häufig vorkommen. Das kann z. B. durch Verwendung von mehreren Chiffrealphabeten erfolgen.

Der Benediktiner Johannes Heidenberg (1462–1516) hat zu diesem Zweck in seiner *Tabula Recta* alle 26 ROT-Verschiebungen aneinandergereiht. Diese Tabelle wird auch Vigenère-Quadrat genannt, benannt nach dem französischen Kryptologen Blaise de Vigenère (1523–1596).

Das Vigenère-Quadrat

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Ein Text wird nun folgendermaßen verschlüsselt: Sender und Empfänger benötigen als Schlüssel ein Passwort. Dies wird dann über den Klartext geschrieben und wenn nötig so oft wiederholt, bis die Länge der Nachricht erreicht wurde.

Passwort:	KRYPTOGRAFIEKRYPTOGRAFIEKRYPTOGRAFIE
Klartext:	DIESENACHRICHTISTSTRENGGEHEIM
Chiffre:	NZCHXBGTWQGRKGHMGZIESOKOYCXF

Alice verschlüsselt nun den ersten Buchstaben ihrer Nachricht, indem sie in der Spalte mit dem ersten Buchstaben des Passworts (Spalte K) die Zeile mit dem Klartextbuchstaben am Anfang sucht (Zeile D). Am Kreuzungspunkt dieser Zeile und Spalte befindet sich der Chiffre-Buchstabe N.

Anschließend wird der zweite Buchstabe der Nachricht mit dem zweiten Buchstaben des Passworts verschlüsselt usw., bis die komplette Nachricht verschlüsselt wurde.

Die Häufigkeiten der einzelnen Buchstaben sind nun gleichmäßiger verteilt. Der Buchstabe E kommt in obigem Beispiel 5-mal vor, aber anstatt 5-mal mit demselben Chiffrezeichen verschlüsselt zu werden, wird er durch 2-mal C, X, E und O ersetzt. Eine reine Häufigkeitsanalyse würde hier also ins Leere laufen.

Kryptoanalyse von Vigenère

Allerdings hat auch die Vigenère-Verschlüsselung leicht ausnutzbare Schwachstellen. Eine davon ist das Auftreten sogenannter Parallelstellen.

Werden längere Texte verschlüsselt, so kommen zwangsläufig im Klartext identische Zeichenketten vor. Dazu gehören häufig auftretende Wörter wie „der“, „die“, „das“ usw. Treffen derartige Zeichenfolgen bei der Verschlüsselung auf unterschiedliche Zeichen des Passwortes, so werden sie auch anders verschlüsselt.

Passwort:	SECRETSECRETSECRETSECRETSECRET
Klartext:DAS.....DAS.....
Chiffre:WSW.....HCJ.....

Allerdings besteht auch eine Wahrscheinlichkeit, dass bei der Verschlüsselung dieselben Zeichenfolgen des Klartexts auf dieselben Zeichen des Passwortes treffen:

Passwort:	SECRETSECRETSECRETSECRETSECRET
Klartext:DAS.....DAS.....
Chiffre:WSW.....WSW.....

In diesem Fall werden die Klartextpassagen auch mit denselben Zeichenfolgen im Chiffretext codiert. Der preußische Infanteriemajor Friedrich Wilhelm Kasiski (1805–1881) beschreibt in seinem Buch „Die Geheimschriften und die Dechiffrierkunst“ eine Methode, in der sämtliche derartigen Parallelstellen gesucht werden. Die Idee, die mit dieser Methode verfolgt wird, geht davon aus, dass identische Textpassagen im Chiffretext dadurch zustande kommen, dass der selbe Klartext auf dieselben Zeichen des Passwortes trifft – dies kann nur in einem ganzzahligen Vielfachen der Passwortlänge passieren.

Von allen gefundenen Parallelstellen werden also die Abstände zwischen den Parallelstellen ermittelt. Es gilt nun, die Länge des Passwortes zu ermitteln, die der größte gemeinsame Teiler der Abstände aller Parallelstellen ist. Die gefundenen Abstände müssen dazu in ihre Primfaktoren zerlegt werden. Diese Ermittlung ist jedoch nicht absolut eindeutig, da es auch zufällige Parallelstellen geben kann, die somit irrelevant sind.

Hat der Angreifer nun eine vermutete Passwortlänge errechnet, so wird der Chiffretext in Gruppen eingeteilt, von denen man vermutet, dass sie mit demselben Buchstaben des Passwortes verschlüsselt wurden. Nimmt man also an, das Passwort sei 5 Zeichen lang, so bildet man 5 Gruppen. In der ersten Gruppe befindet sich der 1., 6., 11., 16. Buchstabe usw., in der zweiten Gruppe der 2., 7., 12., 17. in der dritten der 3., 8., 13., 18. usw.

Für jede dieser Gruppen kann nun getrennt eine Häufigkeitsanalyse durchgeführt und der häufigste Chiffrebuchstabe ermittelt werden – wurde Deutsch oder Englisch als Sprache der Nachricht verwendet, so kann wieder angenommen werden, dass der jeweils am häufigsten auftauchende Buchstabe ursprünglich ein E war. Führt dies allein noch nicht zum gewünschten Ergebnis, so kann für jede dieser Gruppen das komplette Häufigkeitsgebirge der vorkommenden Zeichen ermittelt werden. Dies erlaubt dann sehr zuverlässig die komplette Rekonstruktion des verwendeten Passwortes.

Chiffre: KYPQY PSWCE EMAZG FJMZI EYMLL EVV

Vermutete Länge des Passwortes: 5

Gruppen: KPEFEE, YSMJYV, PWAMMV, QCZZL, YEGIL

Neben dieser als Kasiski-Test bekannten Methode existiert eine noch effektivere Methode zur Ermittlung der Schlüssellänge: der Friedman-Test. Auf sie wird in diesem Buch aber nicht eingegangen.

Vernam

Wie Sie gesehen haben, garantiert auch die Verwendung mehrerer Chiffrealphabete (polyalphabetische Chiffre) keinen absoluten Schutz. Ist das Passwort in Relation zur versendeten Nachricht kurz, so treten Parallelstellen auf, die es Eve ermöglichen, Rückschlüsse auf die Länge des Passwortes zu ziehen. Auf diese Weise ist der Weg zu einem statistischen Angriff geebnet.

Die Konsequenz daraus wäre, eine mit Vigenère verschlüsselte Nachricht sicherer zu machen, indem das Passwort verlängert wird.

Der amerikanische Ingenieur Gilbert S. Vernam (1890–1960) erfand 1917 ein Chiffriersystem, das, basierend auf der Vigenère-Methode, perfekte Sicherheit garantieren konnte:

Ein Chiffriersystem ist dann perfekt, wenn jeder Klartext mit einem zum Chiffriersystem gehörenden Schlüssel auf jeden beliebigen Chiffretext abgebildet werden kann. Im Rahmen von Vigenère ist das möglich, wenn der Schlüssel genauso lang ist wie die Nachricht selbst.

Ein perfektes System bietet dem Angreifer keinerlei Anhaltspunkt für eine statistische Analyse, und ein Brute-Force-Angriff käme einem Ausprobieren sämtlicher möglicher Nachrichten gleich.

Wenn Eve diese Nachricht erhält: **HQAVUXGBCQMTIM**, so kann es bei einer Vernam-Chiffre sein, dass der Klartext **ANGRIFFUMFUENF** lautet, der mit dem Passwort **HDUEMSBHQLSPV** verschlüsselt wurde – genauso gut könnte es sich aber um den Klartext **FRUEHSTUECKSEI** handeln, der mit **CZGRNFNHYOCBEE** verschlüsselt wurde.

Die Vernam-Chiffre ist dann sicher, wenn die Buchstaben des Passwörtes absolut zufällig gewählt werden und einmal benutzte Passwörter nie wieder verwendet werden. Früher wurden die Passwortbuchstaben auf die Blätter eines Abreißblocks geschrieben, weshalb eine Vernam-Chiffre auch One-Time-Pad (Abreißblock) genannt wird.

Was ist der Vorteil einer Vernam-Chiffre? Sender und Empfänger müssen das gleiche Passwort besitzen – und dieses ist ebenso lang wie die Nachricht selbst. Der Vorteil hierbei liegt in der Tatsache, dass Sender und Empfänger den Austausch des Passwörtes (oder besser: lange zufällige Zeichenketten, die als Passwort auf Vorrat dienen) zu einem frei wählbaren Zeitpunkt durchführen können. Die Nachricht selbst muss aber meist zu einem bestimmten Zeitpunkt dringend abgeschickt werden.

Hier taucht das Bild eines Diplomaten auf, der sich mit einem an sein Handgelenk geketteten Koffer auf die Reise macht – er transportiert möglicherweise neue One-Time-Pad-Passwörter. Im Zweiten Weltkrieg und während der Ära des Kalten Krieges wurden für Nachrichten höchster Geheimhaltungsstufe von den beteiligten Parteien One-Time-Pads eingesetzt. Der „Heiße Draht“ zwischen Washington und Moskau wurde ebenfalls mit einem One-Time-Pad gesichert.

10.3 Symmetrische Verfahren

Blockchiffre/Stromchiffre

Im Computerzeitalter treten die klassischen Verschlüsselungsmethoden eher in den Hintergrund. Die Methoden, mit denen in der Informationstechnologie gearbeitet wird, erlauben aufgrund der Rechenleistung deutlich komplexere Operationen.

Grundsätzlich können moderne kryptografische Verfahren unterteilt werden in solche, die **symmetrisch** arbeiten (wenn Alice und Bob denselben Schlüssel benutzen), und in solche, die **asymmetrisch** arbeiten (hier besitzen Alice und Bob verschiedene Schlüssel).

Bei der Unterscheidung von Verschlüsselungsalgorithmen kann auch unterschieden werden, ob der Algorithmus die Daten als Blöcke mehrerer Bits auf einmal verschlüsselt oder jedes Bit einzeln. Erstere werden **Blockchiffren** genannt, letztere **Stromchiffren**.

DES – Data Encryption Standard

Um einen einheitlichen kryptografischen Algorithmus für die IT-Welt zu fördern, führte das NIST (National Institute of Standards and Technology, USA) im Jahre 1973 eine öffentliche Ausschreibung für einen kryptografischen Algorithmus mit folgenden Forderungen durch:

- ✓ Der Algorithmus muss einen hohen Grad an Sicherheit gewährleisten.
- ✓ Der Algorithmus muss vollständig spezifiziert und leicht nachzuvollziehen sein.

- ✓ Die Sicherheit des Algorithmus muss auf dem Schlüssel basieren, nicht auf der Geheimhaltung des Algorithmus.
- ✓ Der Algorithmus muss für alle Anwender zur Verfügung stehen.
- ✓ Der Algorithmus muss an verschiedene Anwendungen angepasst werden können.
- ✓ Der Algorithmus muss sich kostengünstig in elektronische Komponenten implementieren lassen.
- ✓ Der Algorithmus muss effizient in der Benutzung sein.
- ✓ Es muss möglich sein, den Algorithmus zu validieren.
- ✓ Der Algorithmus muss exportierbar sein.

Der aussichtsreichste Kandidat für diesen Algorithmus wurde von IBM eingereicht. Bevor er jedoch zum Standard deklariert wurde, wurde der Entwurf von IBM noch von der NSA (National Security Agency) modifiziert, die die Schlüssellänge als sichtbare Änderung von 128 Bit auf 56 Bit reduzierte. Über Gründe für Änderungen innerhalb des Algorithmus war die NSA zu keiner Stellungnahme zu bewegen, was viele Jahre für Bedenken in der Öffentlichkeit sorgte, die NSA habe eine Hintertür in den Ablauf des DES eingebaut, um diesen leichter entschlüsseln zu können.

Die am weitesten verbreitete Anwendung des DES war die Verschlüsselung des Geldautomatensystems. Die PIN einer EC-Karte wurde unter Verwendung der Bankleitzahl, Kontonummer, des Verfallsdatums und eines geheimen Schlüssels berechnet.

Aufbau des DES

Das Ziel dieses Buches ist nicht, den Leser zu befähigen, kryptografische Algorithmen selbst nachzurechnen zu können. Sie sollten jedoch ein grundlegendes Verständnis für die Arbeitsweise von computerbasierten Verschlüsselungsalgorithmen besitzen, damit Sie Ähnlichkeiten zuordnen und möglicherweise auch Sicherheit und Sicherheitsrisiken selbst besser einschätzen können.

DES ist eine Blockchiffre und verschlüsselt jeweils Blöcke aus 64 Bit. Es wird ein Schlüssel mit der Länge von 64 Bit benötigt, von dem jedoch 8 Bits als Paritätsbits zur Fehlererkennung verwendet werden. Somit ist der Schlüssel effektiv 56 Bit lang.

DES besteht aus mehreren Runden (insgesamt 16), die durchlaufen werden. Soll ein Datenblock verschlüsselt werden, so durchläuft dieser zunächst die Initial Permutation (Eingangspermutation). Hier werden die Datenbits anhand einer fest vorgegebenen Tabelle in ihren Positionen vertauscht.

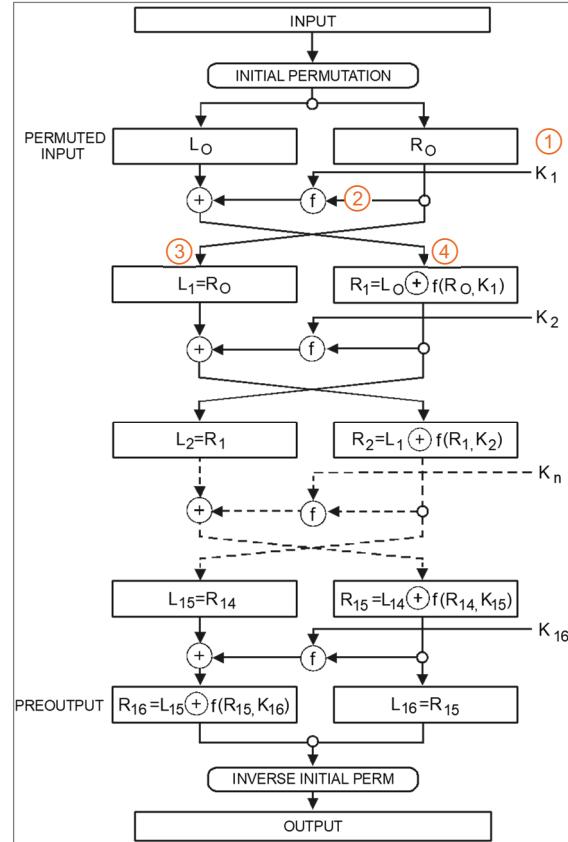
Anschließend werden die 64 Datenbits in zwei Blöcke zu jeweils 32 Bits aufgeteilt. Diese werden als L und R bezeichnet. Die nebenstehende Zahl in der Grafik zeigt jeweils an, um welche Runde es sich handelt.

Vor der ersten Runde existieren also die Blöcke L_0 und R_0 ①. Der DES berechnet, basierend auf dem 56-Bit-Schlüssel, für jede Runde einen sogenannten Round-Key (Rundenschlüssel), der in der Grafik mit K_1 bis K_{16} bezeichnet ist. Durch ein definiertes Verfahren werden in jeder Runde 48 aus den 56 vorhandenen Bits des Gesamt-schlüssels ausgewählt.

Kern des DES ist die Verschlüsselungsfunktion f ②. Die erste Runde der Verschlüsselung besteht nun darin, dass die Funktion f auf R_0 mithilfe von K_1 angewandt wird. Das Ergebnis dieser Operation wird modulo 2 zu L_0 addiert (= XOR).

Bei einer bitweisen Addition modulo 2 werden 2 Bits addiert, aber der Überlauf wird verworfen. Das Ergebnis ist dasselbe wie bei einer XOR-Operation (exklusives ODER).

Für die nächste Runde ist die neue linke Hälfte L_1 gleich dem alten R_0 ③ und die neue rechte Hälfte gleich dem Ergebnis der modulo 2 Addition ④.



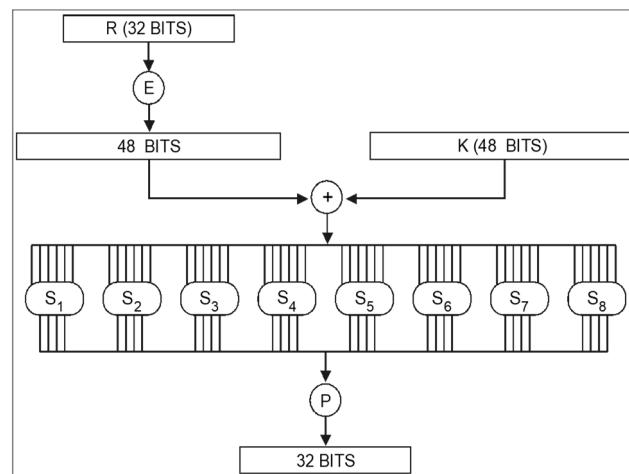
DES-Verschlüsselung

$0 + 0$	0
$0 + 1$	1
$1 + 0$	1
$1 + 1$	0

Modulo 2 Addition, XOR

Funktion f

Die Funktion f, die in jeder Runde die Hälfte R mit dem jeweiligen Rundenschlüssel verschlüsselt, erhält 32 Datenbits und führt eine Expansionspermutation (E) durch. Dieser Vorgang sorgt anhand einer festgelegten Tabelle dafür, dass bestimmte Bits von R verdoppelt werden, sodass R auf eine Länge von 48 Bit anwächst. Nur bei gleicher Länge können diese Daten nämlich mit dem ebenfalls 48 Bit langen Rundenschlüssel modulo 2 addiert werden (=XOR).



DES-Funktion f

Nach der Addition muss das Ergebnis wieder auf eine Länge von 32 Bits gebracht werden, damit es im Algorithmus weiterverwendet werden kann. Jeweils 6 Bits wandern in eine der acht sogenannten S-Boxen (S wie Substitution, Ersetzen).

Anhand der fest in den jeweiligen S-Boxen definierten Tabellen werden die 6 Eingangsbits durch 4 Ausgangsbits substituiert. Alle Ergebnisbits werden dann wieder aneinandergefügt und stellen das Endergebnis der Funktion f in der aktuellen Runde dar.

S-Box 1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Sollen 6 Bits durch eine S-Box substituiert werden, so geben das erste und das letzte Bit als Dualzahl gelesen die Zeilennummer der S-Box an. Die 4 restlichen Bits in der Mitte bilden die Spaltennummer. Das Ergebnis ist der in der entsprechenden Zeile und Spalte stehende Wert.

Oben abgebildet befindet sich die S-Box 1. Beispielsweise kommen die 6 Bits 101110 (Dezimal 46) im Laufe der Bearbeitung des DES in diese S-Box. Das erste Bit ist 1, das letzte die 0. 10 als Dualzahl hat dezimal den Wert 2. Also kommt die Zeile mit der Nummer 2 zur Anwendung.

Die restlichen Bits sind 0111. Als Dezimalzahl gelesen ergeben diese Bits den Wert 7. Das Ergebnis der S-Box 1 ist also in Zeile 2, Spalte 7 zu finden: 11. Binär liefert die S-Box also 1011 als Ergebnis.

Die S-Box-Substitution ist der wichtigste Teil des DES. Während alle anderen Operationen linear sind und sich leicht analysieren lassen, sorgen die S-Boxen mit ihrer Substitution für Nicht-Linearität. Beachten Sie, dass in der S-Box 1 (wie in den anderen 7 S-Boxen) in jeder Zeile die Werte 0 bis 15 jeweils nur einmal, aber in zufälliger Reihenfolge auftauchen.

Die Wiederholung dieser Vorgänge in jeweils 16 Runden führt dazu, dass sich der Zustand eines Bits, das an einer Stelle im Datenblock von einer Funktion geändert wurde, gleich einer Kettenreaktion auf möglichst viele andere Bits auswirken kann. Diese Kettenreaktion (Kaskadeneffekt) ist in der Kryptografie absolut erwünscht – so ist gewährleistet, dass nach Durchlaufen aller Runden jedes Bit des Schlüssels die Chance hatte, auf jedes Datenbit Einfluss zu nehmen. Das Ergebnis ist in so einem Fall ein Block aus scheinbar zufällig gesetzten Bits.

DES entschlüsseln

Soll ein mit DES verschlüsselter Datenblock wieder entschlüsselt werden, so wird derselbe Algorithmus durchlaufen. Der einzige Unterschied ist, dass die Runden in umgekehrter Reihenfolge, also von 16 bis 1, ausgeführt werden.

Dieses Rückwärts-Durchlaufen des DES-Algorithmus führt dazu, dass bei Eingabe des richtigen Schlüssels auch der Klartext wieder sichtbar wird.

Zukunft von DES

DES gilt heute als überholt. Die inzwischen aufgehobene Exportbeschränkung für kryptografische Verfahren und Produkte, die als „starke Kryptografie“ betrachtet wurden, und die Tatsache, dass DES im Rahmen eines internetbasierten Brute-Force-Angriffs geknackt wurde, machten den Bedarf für andere Verfahren deutlich.

War in den 60er-Jahren noch nicht ausreichend Rechenleistung vorhanden, um für einen Brute-Force-Angriff den Schlüsselraum von 2^{56} in ausreichender Zeit durchsuchen zu können, so ist dies heutzutage aufgrund der gestiegenen Rechenleistung und der Möglichkeit, die Brute-Force-Rechenleistung auf viele Rechner aufzuteilen, kein größerer Aufwand mehr.

Das Problem der Schlüssellänge wurde, bis ein neuer Algorithmus standardisiert wurde, zwischenzeitlich dadurch gelöst, dass die Daten mit DES dreimal hintereinander verschlüsselt wurden und dabei jedes Mal ein anderer 56-Bit-Schlüssel benutzt wurde. Nominell beträgt also die Schlüssellänge dieses als Triple DES (3DES) bekannten Verfahrens 168 Bits. Das Maß an Sicherheit ist jedoch nicht das Dreifache, sondern entspricht eher einer Schlüssellänge von 112 Bits. Aufgrund der Tatsache, dass beim 3DES der DES dreimal für jeden Datenblock aufgerufen wird, benötigt dieser auch das Dreifache an Rechenzeit.

Neben DES wurden einige weitere Algorithmen entwickelt, die das Problem des in die Jahre gekommenen DES lösen sollten. CAST, IDEA und Blowfish sind nur drei der zahlreichen Algorithmen, die neben vielen anderen auf dem Markt Beachtung gefunden haben.

AES – Advanced Encryption Standard

1997 forderte das NIST die Öffentlichkeit wiederum auf, Vorschläge für einen neuen Verschlüsselungsalgorithmus – den zukünftigen Advanced Encryption Standard – abzugeben. Der neue Standard sollte wie DES eine symmetrische Blockchiffre sein. Er musste Datenblöcke von 128 Bits verarbeiten und mit variablen Schlüssellängen von mindestens 128, 192 und 256 Bits arbeiten können. Die Implementierung sollte sowohl in Hardware als auch in Software möglich sein.

Aus vielen eingereichten Vorschlägen, unter denen sich auch der von der Deutschen Telekom entwickelte Algorithmus MAGENTA befand, wurden vom NIST fünf Finalisten ausgewählt:

- ✓ MARS (IBM)
- ✓ RC6 (RSA Laboratories)
- ✓ Rijndael (Joan Daemen und Vincent Rijmen)
- ✓ SERPENT (Ross Anderson, Eli Biham und Lars Knudsen)
- ✓ TWOFISH (Bruce Schneier und John Kelsey)

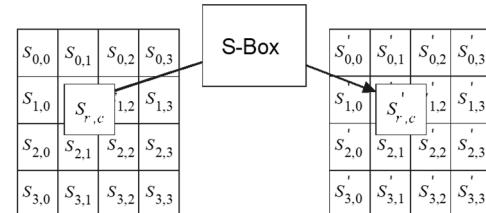
Am 2. Oktober 2000 stellte das NIST den Gewinner der Ausschreibung und somit den zukünftigen AES-Algorithmus vor: den aus Belgien stammenden Rijndael. Am 26. November 2001 wurde der AES als FIPS 197 verabschiedet (FIPS = Federal Information Processing Standards).

Wie die meisten Verschlüsselungsalgorithmen arbeitet Rijndael mit mehreren Runden, die Anzahl variiert aber mit der verwendeten Schlüssellänge zwischen 10, 12 und 14 Runden. Rijndael bearbeitet die 128 Bits als Matrix aus 16 Bytes, an denen pro Runde vorgegebene Substitutionsoperationen, Verschiebeoperationen, Spaltenoperationen und die Addition eines errechneten Rundenschlüssels vorgenommen werden.

Im Gegensatz zu DES betrachtet AES die zu verarbeitenden Daten nicht als Bitblöcke, sondern organisiert die Daten für die Verschlüsselung byteweise in einem State (Zustand). Soll ein 128-Bit-Block verschlüsselt werden, so wird dieser in 16 Bytes aufgeteilt. Diese 16 Bytes werden in eine 4-x-4-Matrix aufgeteilt, die den State darstellt.

Die Verschlüsselungsfunktion des AES besteht nun darin, in mehreren Runden durch Substitution, Verschiebung, Multiplikation und Addition von Rundenschlüsseln den State zu manipulieren.

Zu Beginn einer Runde werden die einzelnen Bytes des States durch die Funktion **ByteSub** (Byte Substitution) anhand einer S-Box durch andere Werte ersetzt.



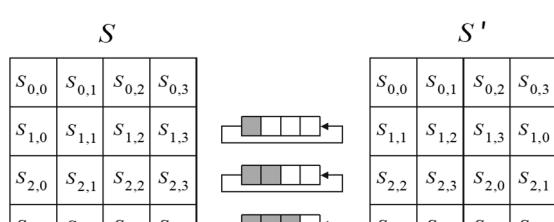
ByteSub anhand einer S-Box

Die Besonderheit beim AES im Gegensatz zum DES ist, dass die S-Box beim AES die einzige Tabelle ist, die der Algorithmus für seine Arbeit benötigt. Die Speicherung der Tabelle (und der inversen Tabelle für die Entschlüsselung) wäre sogar ganz vermeidbar, da die enthaltenen Werte auf einer Multiplikationsoperation basieren.

		y																
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	
x		0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0		
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15		
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75		
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84		
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf		
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8		
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2		
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73		
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0e	db		
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79		
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08		
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a		
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1c	9e		
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df		
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16		

AES S-Box

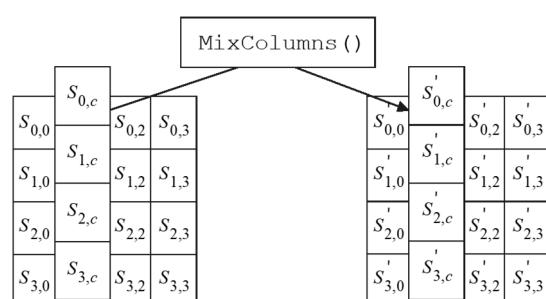
Nach dem **ByteSub** werden die Bytes durch **ShiftRow** zeilenweise um 0, 1, 2 oder 3 Stellen nach links verschoben. Die Bytes, die links aus der Matrix herausfallen, werden auf der rechten Seite wieder eingefügt.



ShiftRow

Anschließend wird spaltenweise die Funktion **MixColumns** durchgeführt. Um die Spalten zu „mischen“, d. h., um dafür zu sorgen, dass jedes Byte einer Spalte Gelegenheit hat, auf jedes andere Byte derselben Spalte einen Einfluss auszuüben, wird jede Spalte als Vektor betrachtet und mit einer fest vorgegebenen Matrix multipliziert.

$$\begin{bmatrix} S'_{0,c} \\ S'_{1,c} \\ S'_{2,c} \\ S'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} S_{0,c} \\ S_{1,c} \\ S_{2,c} \\ S_{3,c} \end{bmatrix}$$



MixColumns

Abschließend wird der gesamte State bitweise modulo 2 mit dem Rundenschlüssel addiert. Der Rundenschlüssel für jede der 10 bis 14 nötigen Runden wird basierend auf dem Gesamt-schlüssel vom AES errechnet. Sollten weitere Runden folgen, so beginnt AES wieder bei ByteSub.

Die Grafik rechts veranschaulicht den Ablauf einer AES-Runde dreidimensional.

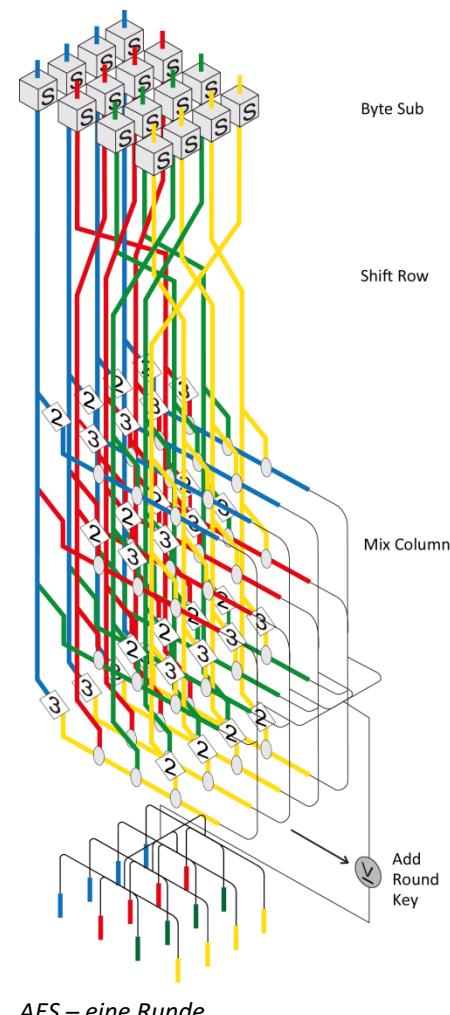
Verbreitung und Sicherheit von AES

Seit Oktober 2000 ist AES offizieller Nachfolger von DES und 3DES. Anwendungsfelder:

- ✓ WLAN mit WPA2/WPA3
- ✓ SSHv2/OpenSSH
- ✓ IPsec
- ✓ SFTP/FTPS/LFTP/OFTP
- ✓ Skype
- ✓ Dateisystemverschlüsselung (z. B. Apple macOS, ab Windows XP SP1)
- ✓ Dateikomprimierung (z. B. 7-Zip, RAR)
- ✓ PGP & GnuPG
- ✓ https usw.

Seine gründliche Analyse und die Tatsache, dass bisher keine Schwachstellen aufgedeckt wurden, fördern das Vertrauen in den neuen Algorithmus.

Während DES zwar 2^{56} mögliche Schlüssel akzeptiert, können nicht alle davon ohne Bedenken benutzt werden (sogenannte schwache Schlüssel). Bei AES sind derartige Schlüssel nicht bekannt. Zusätzlich hat der Anwender von AES die Auswahl aus mindestens 2^{128} verschiedenen Schlüsseln.



Der 1999 im Laufe eines Wettbewerbs mit verteilter Rechenleistung geknackte DES-Schlüssel wurde mit einer Brute-Force-Leistung von ca. 199.000.000.000 getesteten Schlüsseln pro Sekunde in ca. 22 Stunden gefunden.

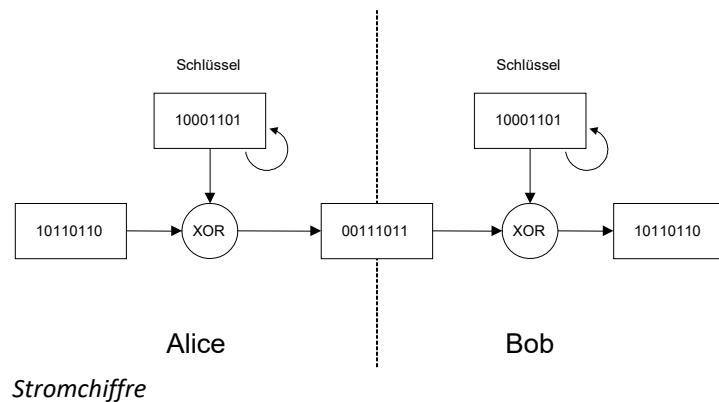
Hätte jemand dieselbe Rechenleistung zur Verfügung, um sämtliche 128-Bit-Schlüssel für AES durchprobieren zu können, so würde dies $1,709 * 10^{27}$ Sekunden dauern. Das sind $5,42 * 10^{19}$ Jahre. Das wäre also um ungefähr 387 Milliarden Mal länger als das zurzeit angenommene Alter des Universums (14 Mrd. Jahre = $14 * 10^9$).

RC4

RC4 wurde 1987 von Ronald L. Rivest entwickelt und wird heute nur noch bei PPTP eingesetzt. Im Gegensatz zu den bisher vorgestellten Algorithmen ist der RC4-Algorithmus keine Block-, sondern eine Stromchiffre. Stromchiffren eignen sich besonders gut, wenn eine blockweise Verschlüsselung der Daten (mit einhergehendem Zeitaufwand bzw. Latenzzeit) nicht geeignet ist, da die zu verschlüsselnden Daten zu einem (Echtzeit-)Datenstrom gehören.

Eine Stromchiffre betrachtet den Datenstrom bitweise und verschlüsselt diesen, indem der Datenstrom mit den Bits des Passwortstromes verknüpft wird (XOR).

Die Kernaufgabe einer Stromchiffre ist es, aus dem vorgegebenen Schlüssel, der für beide Parteien identisch ist, einen beliebig langen Bitstrom zu erzeugen, der pseudozufällig ist.



Die Verknüpfung der Schlüsselbits mit den Klartextbits macht den Chiffretext für Eve unlesbar, weil die Schlüsselbits scheinbar zufällige Modifikationen an den Datenbits vornehmen. Da Bob den korrekten Schlüssel besitzt, kann er auf seiner Seite denselben Pseudo-Zufallsprozess starten. Es wird also genau dieselbe Folge an Einsen und Nullen generiert, die mit dem Chiffretext verknüpft wird. Durch die Eigenschaften des XOR wird auf diese Weise der Klartext wieder sichtbar.

Die Qualität einer Stromchiffre hängt von der Qualität der Zufallszahlen ab, die der Generator basierend auf einem eingegebenen Schlüssel generiert. Ist die Bitfolge nicht vorhersehbar und wiederholt sie sich nicht, so ist die Chiffre sicher.

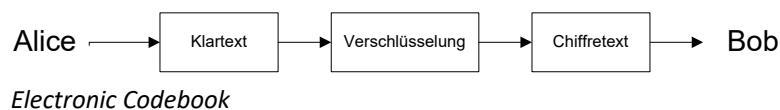
Stromchiffre weist eine hohe Ähnlichkeit zur Vernam-Chiffre auf – das ist kein Zufall. Der Unterschied zwischen Vernam- und der Stromchiffre liegt lediglich darin, dass bei Vernam das Passwort in der Länge der Nachricht vor der Übertragung ausgetauscht werden muss. Bei einer Stromchiffre muss vorher nur ein kurzes Passwort ausgetauscht werden, das dann zur Generierung eines beliebig langen binären „Vernam-Passwortes“ dient.

RC4 ist ein von Ron Rivest in den 80er-Jahren entwickelter Stromchiffre-Algorithmus, der inzwischen schon gebrochen wurde, aber dennoch sehr weit verbreitet ist. Die häufigste Anwendung ist die Verschlüsselung des Datenstromes im Webbrowser, wenn eine gesicherte Verbindung mit SSL hergestellt wird. RC4 wurde von Herstellern in Amerika gerne in Software implementiert, da RC4 zu den Zeiten der Kryptoexport-Kontrolle mit einer Schlüssellänge von 40 Bit ohne besondere Genehmigung durch die Regierung exportiert werden durfte.

Operationsmodi

Generell unterscheidet man, in welchen Operationsmodi kryptografische Algorithmen auf Daten angewandt werden. Der einfachste dieser Modi ist der ECB-Modus.

Beim **ECB**, dem **Electronic Codebook**, wird ein Datenblock gemäß dem Algorithmus mit dem gleichen Schlüssel verschlüsselt und an den Empfänger geschickt. Anschließend wird der nächste Block verschlüsselt usw.



Die Datenblöcke werden also unabhängig voneinander verschlüsselt – analog zu einem Codebuch, in dem nachgeschlagen werden muss, welche Wörter der Nachricht durch welche Chiffrewörter ersetzt werden müssen.

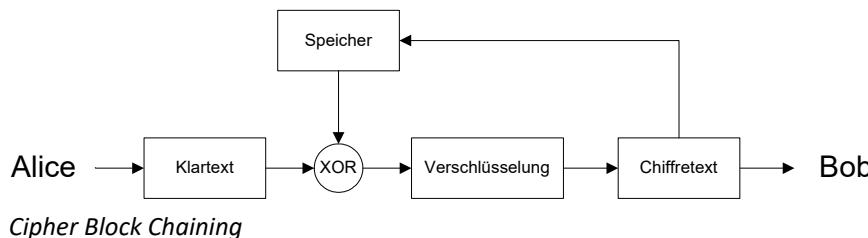
Diese Methode hat einen gravierenden Nachteil: Identische Passagen der Klartextblöcke werden mit stets den gleichen Chiffretextblöcken codiert.

Dies würde Eve wiederum erlauben, eine Häufigkeitsanalyse zu erstellen. Ist die Art der übermittelten Daten bekannt (Text, Bilddokumente, Audio etc.), so können, ähnlich wie bei der Häufigkeitsverteilung der Buchstaben in natürlicher Sprache, Rückschlüsse auf den Klartext gezogen werden.

Die anderen Betriebsmodi führen aus diesem Grund eine Rückkopplung der Art ein, dass ein verschlüsselter Datenblock die Verschlüsselung für nachfolgende Datenblöcke beeinflusst. Auf diese Weise erzeugen auch lange identische Datenblöcke im Klartext keine identischen Datenblöcke im Chiffretext.

Beim **CBC**, dem **Cipher Block Chaining**, wird ein Klartextblock vor der Verschlüsselung mit dem vorherigen Chiffretextblock XOR-verknüpft. Der nach der Verschlüsselung erhaltene Chiffretextblock wird gespeichert.

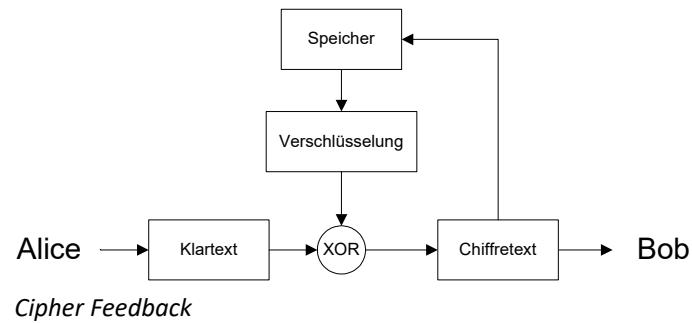
Mit diesem wird dann der nächste Klartextblock mit dem Chiffretextblock XOR-verknüpft usw.



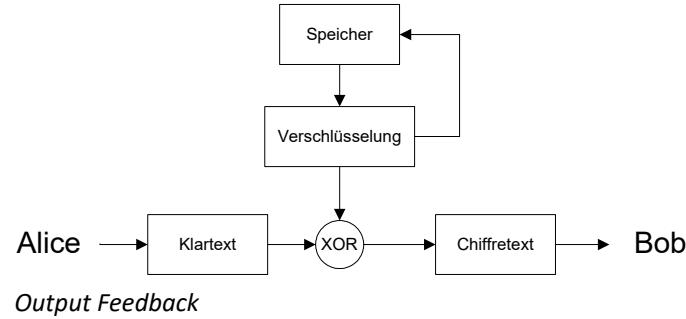
Da bei der Verschlüsselung des ersten Blockes noch kein vorhergehender Block gespeichert wurde, kommt hier ein IV, ein Initialisierungsvektor, zum Einsatz.

Im **CFB Mode**, dem **Cipher Feedback**, wird zuerst der IV verschlüsselt und dieser dann per XOR mit dem Klartext verknüpft. Das Ergebnis wird gespeichert und beim nächsten Block verschlüsselt, bevor es per XOR mit dem neuen Block verknüpft wird usw. CFB eignet sich gut für Fälle, in denen Daten verschlüsselt werden sollen, die kleiner sind als die vom Algorithmus unterstützte Blockgröße.

Soll jeweils nur ein einzelnes Byte verschlüsselt übertragen werden, die Blockgröße ist jedoch 8 Bytes (64 Bit), so wird der IV aus dem Zwischenspeicher verschlüsselt, aber die XOR-Operation nur mit dem 1. Byte des Ergebnisses am Klartextbyte durchgeführt. Das Ergebnisbyte wird dann am Ende des Zwischenspeichers angehängt, während das erste Byte gelöscht wird. Nachfolgende Bytes werden ebenfalls verschlüsselt, indem der Zwischenspeicher verschlüsselt wird, aber nur das erste Byte mit dem Klartext mit dem Chiffretextblock XOR-verknüpft wird usw.



Der **Output Feedback Mode (OFB)** ist dem CFB Mode sehr ähnlich, nur wird hier nicht das Ergebnis in die Warteschlange gestellt, sondern der gerade zuvor verschlüsselte Zwischenspeicher selbst. In diesem Modus bildet die Kryptografieeinheit einen in sich geschlossenen Kreis, der vom Schlüssel angestoßen wird. Die Verschlüsselung der Klartextdaten selbst findet über die Verknüpfung mit XOR statt. Mit diesem Betriebsmodus ist ein beliebiger kryptografischer Algorithmus als Stromchiffre einsetzbar.



Ein auf diese Weise betriebener guter kryptografischer Algorithmus ist so auch als guter Zufallszahlengenerator verwendbar.

10.4 Übung

Fragen zur symmetrischen Kryptografie

Übungsdatei: --

Ergebnisdatei: uebung10.pdf

- Welche Aussage trifft auf die symmetrische Verschlüsselung zu?

a	Wie bei der asymmetrischen Verschlüsselung existieren ein Private Key und ein Public Key.
b	Die symmetrische Verschlüsselung beruht auf dem Prinzip, dass Sender und Empfänger über den gleichen geheimen Schlüssel verfügen.
c	Der Austausch des geheimen Schlüssels muss selbst über einen verschlüsselten Kanal getätigt werden.

11

Asymmetrische Kryptografie

11.1 Nachteile symmetrischer Verfahren

Schlüsseltausch

Obwohl symmetrische Methoden in Netzwerken inzwischen bewährt und weit verbreitet sind, haben sie alle einen gravierenden Nachteil: Sender und Empfänger müssen über den gemeinsamen geheimen Schlüssel verfügen, mithilfe dessen die Nachricht ver- bzw. entschlüsselt werden kann.

Ist Alice räumlich von Bob getrennt und besteht nun der Bedarf, eine Nachricht zum Schutz vor unbefugtem Mitlesen verschlüsselt zu übertragen, so stellt sich die Frage: Wie kann Alice den Schlüssel selbst sicher übertragen?

Ein Problem der symmetrischen Algorithmen ist, dass der Schlüssel selbst für seinen Transport einen sicheren Kanal benötigt. Wenn Sie annehmen, dass Sie Verschlüsselung benutzen müssen, weil Sie wichtige Nachrichten über einen unsicheren Kanal schicken müssen und befürchten, dass diese Nachrichten kompromittiert werden könnten, stellt sich die Frage, über welchen sicheren Kanal Sie den benötigten gemeinsamen Schlüssel übertragen können.

Hätten Sie diesen sicheren Kanal für den Austausch des Schlüssels, so könnten Sie auch auf die Idee kommen, die Nachricht selbst über die sichere Verbindung zu schicken – damit würden Sie sich den Aufwand für die Verschlüsselung sparen, und die Nachricht wäre trotzdem sicher beim Empfänger.

In der Regel werden symmetrische Verfahren dennoch angewandt, weil die Schlüssel noch einen Vorteil gegenüber der Nachricht haben: Sie sind in den meisten Fällen deutlich kürzer als die Nachricht, wenn Sie einmal von Chiffren wie der Vernam-Chiffre absehen. Einen sicheren Kanal für einen kurzen Schlüssel zu finden, ist in der Regel einfacher, als einen sicheren Kanal für eine lange Nachricht.

Trotzdem stellt der sichere Austausch des Kommunikationsschlüssels mitunter ein Problem dar. Was ist, wenn sich Alice und Bob, als sie noch zusammen im selben Raum waren, nicht auf einen geheimen Schlüssel geeinigt haben? Oder Alice und Bob kennen sich im realen Leben gar nicht, sondern haben vor, jetzt das erste Mal über das Internet elektronisch in Kontakt zu treten. Wie sollen Sie in einem Online-Shop mithilfe einer verschlüsselten Verbindung sicher einkaufen können, wenn Sie noch nie vorher mit diesem Shop einen Schlüssel vereinbart haben? Die Beantwortung dieser Fragen hat einen wichtigen Einfluss auf viele Teilbereiche der Netzwerk-kommunikation.

Schlüsselmanagement

Nehmen Sie an, Alice, Bob und Carol wollen in Zukunft sicher miteinander kommunizieren. Sie möchten, dass jeder mit jedem eine verschlüsselte Verbindung aufbauen kann. Die Schlüssel sollen aber jeweils nur zwischen zwei Partnern gültig sein.

Alice vereinbart also mit Bob einen Schlüssel 1 und mit Carol einen Schlüssel 2. Damit Bob kommunizieren kann, benutzt er für Alice Schlüssel 1 und für Carol einen neuen Schlüssel 3. Carol wiederum benutzt für Alice den Schlüssel 2 und für eine Verbindung mit Bob den Schlüssel 3.

Jeder besitzt nun 2 Schlüssel und maximal 2 Personen kennen denselben Schlüssel:

- | | | | | | |
|---------|------|-------|------|---------|------|
| ✓ Alice | 1, 2 | ✓ Bob | 1, 3 | ✓ Carol | 2, 3 |
|---------|------|-------|------|---------|------|

Bei drei Personen existieren also 3 Schlüssel, jeder der Beteiligten muss sich 2 Schlüssel merken. Lassen Sie nun 10 Teilnehmer vereinbaren, ihre Kommunikation gegenseitig mit Schlüsseln zu sichern. Jeder der 10 Teilnehmer benötigt 9 Schlüssel für seine möglichen Partner. Das sind $10 * 9$ Schlüssel. Da jeweils zwei Teilnehmer einen Schlüssel gemeinsam benutzen, ist die Anzahl der existierenden Schlüssel 45. Befinden sich 100 Teilnehmer in diesem Netzwerk, so ist die Anzahl der existierenden Schlüssel 4950. Jeder einzelne Teilnehmer muss sich 99 Schlüssel merken.

Die Anzahl der Schlüssel wächst quadratisch, da für n Teilnehmer $n*(n-1)/2$ Schlüssel benötigt werden. Spätestens hier wird klar, warum ein System mit vorher ausgehandelten symmetrischen Schlüsseln (pre-shared keys) nur bei kleinen Teilnehmerzahlen sinnvoll ist. Für das Internet mit seiner unüberschaubaren Anzahl an Teilnehmern wäre so etwas undurchführbar.

11.2 Einwegfunktion

Die Anfänge der Public-Key-Verschlüsselung

1976 schlugen **Whitfield Diffie** und **Martin E. Hellman** in ihrer Arbeit „New Directions in Cryptography“ einen neuen Ansatz vor, in dem sie das Konzept der Public-Key-Cryptography entwarfen.

Ein Teilnehmer in einem **Public-Key**-Verfahren besitzt einen **public** (öffentlichen) und einen **private** (privaten) Schlüssel. Der öffentliche Schlüssel wird zum Verschlüsseln der an ihn gerichteten Nachrichten durch den Absender verwendet. Der geheime private dient zum Entschlüsseln dieser Nachrichten durch den Empfänger. Während der öffentliche Schlüssel in einem jedermann zugänglichen Verzeichnis öffentlich gemacht wird, muss der private Schlüssel geheim gehalten werden. Dies ist das Grundkonzept von **asymmetrischen Verfahren**.

Damit Public-Key-Verfahren funktionieren können, müssen bestimmte Voraussetzungen gegeben sein. Vor allem taucht hier der Begriff **Einwegfunktion** als zentrales Thema auf.

Eine Einwegfunktion $f(x)$ ist eine Funktion, die eine gegebene Eingabe x auf den Wert y so abbildet, dass es

- ✓ ein effizientes Verfahren gibt, die Werte $y = f(x)$ für alle gegebenen x zu berechnen;
- ✓ kein effizientes Umkehrverfahren gibt, für alle gegebenen y , die durch $f(x)$ berechnet wurden, das x zu bestimmen.

Es soll also bei einer Einwegfunktion nicht möglich bzw. nicht effizient möglich sein, die Umkehrfunktion zu berechnen. Die Zerlegung von Zahlen in ihre Primfaktoren ist ein (in der Kryptografie nicht ohne Grund) oft zitiertes Beispiel. Große Primzahlen zu multiplizieren und das Ergebnis festzustellen, ist relativ schnell möglich. Dahingegen beansprucht eine Zerlegung einer großen Zahl in ihre Primfaktoren mitunter sehr viel Rechenzeit. Es ist bislang kein effizienter Algorithmus hierfür bekannt.

Ein Problem ist allerdings, dass nicht bewiesen ist, ob es nicht doch eine sehr viel effizientere Methode zur Faktorisierung gibt als die bisher bekannten. Neue Erkenntnisse auf dem Gebiet der Quantencomputer könnten also die etablierten Systeme durchaus in Gefahr bringen.

Echte Einwegfunktionen machen für die Übertragung von Nachrichten keinen Sinn. Würde eine Nachricht X mit einer derartigen Funktion verschlüsselt, sodass der Empfänger Y erhält, so hat dieser ja per Definition keine Möglichkeit, aus dem Y wieder das ursprüngliche X herzuleiten.

Für die Kryptografie werden hier sogenannte Trapdoor-Einwegfunktionen benötigt (Trapdoor, hier: Geheimgang, Hintertür). Diese haben folgende Eigenschaften:

- ✓ Es gibt effiziente Verfahren, um $y = f(x)$ zu berechnen.
- ✓ Die Berechnung der Umkehrfunktion f^{-1} und somit $x = f^{-1}(y)$ ist nur effizient mithilfe einer geheimen Zusatzinformation möglich (die Hintertür).

So schwer es ist, endgültig zu beweisen, dass es effizientere Methoden zur Faktorisierung gibt, so schwer ist es, zu beweisen, dass es Trapdoor-Einwegfunktionen gibt. Diffie und Hellman haben in ihrer Arbeit ausführlich erörtert, dass die Public-Key-Kryptografie funktionieren würde, wenn es Trapdoor-Einwegfunktionen gäbe. Trotz intensiver Suche konnten sie jedoch keine finden.

Sie schlugen aber ein kryptografisches Protokoll vor, das nach ihnen benannt heute zu den wichtigsten Protokollen der Kryptografie gehört: den **Diffie-Hellman**-Schlüsselausch.

Restwertoperation, Modulo

Für das Verständnis der folgenden Abschnitte sollten Sie wissen, was eine Modulo-Operation ist. Das Ergebnis einer mathematischen Operation modulo einer Zahl n ist gleich dem ganzzahligen Restwert des Ergebnisses einer Division durch n. Bei symmetrischen Algorithmen kommt oft ein Spezialfall der Modulo-Operation zur Anwendung: Hier wird nur bitweise Modulo 2 addiert, was der Operation XOR entspricht.

Allgemeiner definiert gilt für Modulo-Operationen: Während $11 + 7 = 18$ ist, wäre $11 + 7 \bmod 8 = 2$. Die Zahl 18 ist 2x ganzzahlig durch 8 teilbar – der Rest ist 2 und somit das Ergebnis dieser Modulo-Operation. Die Zahl, die die Modulo-Operation definiert (hier 8), wird **Modul** genannt.

$5 * 3 = 15$; $5 * 3 \bmod 9 = 6$; 15 kann nur 1x durch 9 ganzzahlig geteilt werden. Der Rest ist 6.

$1 + 1 = 2$; $1 + 1 \bmod 2 = 0$. Das Ergebnis 2 ist ganzzahlig durch 2 teilbar – der Rest ist 0. Dies entspricht der XOR-Operation.

Wenn Sie also irgendeine Zahl $x \bmod n$ berechnen wollen, teilen Sie x durch n, und der ganzzahlige Rest ist das Ergebnis.

Diskrete Exponentialfunktion und diskreter Logarithmus

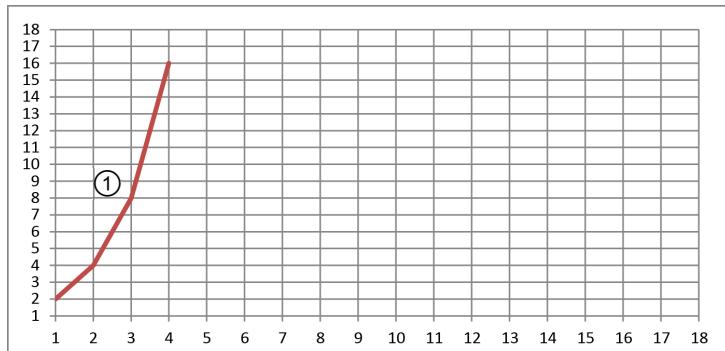
Eine gewöhnliche Exponentialfunktion a^b wird durch eine angefügte Modulo-Operation zu einer diskreten Exponentialfunktion. Diese weist in Bezug auf die Suche nach einer Einwegfunktion interessante Eigenschaften auf.

Für die Exponentialfunktion 2^x können Sie folgende Wertetabelle aufstellen, wenn Sie x ganz-zahlig von 1 bis 18 steigen lassen:

x	1	2	3	4	5	6	7	8	9
y	2	4	8	16	32	64	128	256	512
x	10	11	12	13	14	15	16	17	18
y	1024	2048	4096	8192	16384	32768	65536	131072	262144

In einer Grafik lassen sich diese Ergebnisse so veranschaulichen ①. Die Funktion steigt sehr schnell und steil an, und ihr Verlauf ist stetig.

Wenn Sie diese Funktion umkehren wollen, d. h. aus den gegebenen y -Werten die ursprünglichen x -Werte errechnen wollen, so können Sie schon in der Grafik absehen, wie die Funktion aussehen wird: Es genügt eine Spiegelung an der 45° -Achse durch den Nullpunkt.



Exponentialfunktion 2^x

Auch mathematisch gibt es bei dieser Funktion keine Probleme, eine Umkehrfunktion zu ermitteln – den Logarithmus. Im Falle von $y = 2^x$ ist die Umkehrfunktion also $x = \log_2 y$.

Falls Sie dies auf Ihrem Taschenrechner ausprobieren möchten, denken Sie daran, dass Sie den Logarithmus zur selben Basis verwenden müssen wie die Basis der Exponentialfunktion. Für dieses Beispiel benötigen Sie also den Logarithmus zur Basis 2.

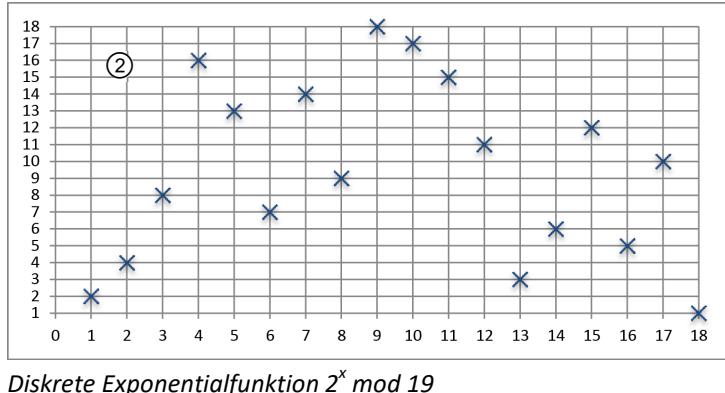
Da Sie auf Kalkulatoren mitunter den natürlichen Logarithmus (\ln) und den Logarithmus zur Basis 10 (\log) finden, müssen Sie Ihre Ergebnisse umrechnen: Sie wollen zu einer Zahl x den Logarithmus zur Basis b errechnen, können aber nur den Logarithmus mit Basis a anwenden: $\log_b x = \log_a x / \log_a b$. So können Sie relativ leicht errechnen, dass bei gegebenem y von 32768 folgt: $\log 32768 / \log 2 = 15$. Das ursprüngliche x war also gleich 15.

Wenn Sie die diskrete Exponentialfunktion $2^x \bmod 19$ bilden, so erhalten Sie folgende Wertetabelle:

x	1	2	3	4	5	6	7	8	9
y	2	4	8	16	13	7	14	9	18
x	10	11	12	13	14	15	16	17	18
y	17	15	11	3	6	12	5	10	1

In einer Grafik dargestellt, zeigt sich folgende Verteilung ②.

Abgesehen von den ersten vier Punkten ist für das menschliche Auge kein regelmäßiges Verteilungsschema zu erkennen.



Die Umkehrung mithilfe eines Rechners zu ermitteln, würde im naiven Fall für den Programmierer heißen, bei gegebenem y schrittweise alle diskreten Exponentialfunktionen zu 2^x durchzurechnen und zu prüfen, ob das Ergebnis der Berechnung gleich y ist – dann wurde die Umkehrung gefunden. Obwohl es Verfahren gibt, mit denen die Suche nach einem diskreten Logarithmus beschleunigt wird, ist diese immer noch um ein Vielfaches aufwendiger als die Bildung der entsprechenden Exponentialfunktion durch eine Reihe simpler Multiplikationen.

In diesem Sinn ist die diskrete Exponentialfunktion also eine Einwegfunktion.

Erzeugende Elemente

Ein erzeugendes Element der Addition ist die 1. Betrachten Sie den Zahlenraum Z mit Elementen von 0 bis 9 bei der Addition modulo 10, so gilt:

$$1 = 1, 1 + 1 = 2, 1 + 1 + 1 = 3$$

$$1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 = 9$$

$$1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 = 0$$

Sie können also durch wiederholte Addition von 1 alle Elemente in der Gruppe erzeugen. Deswegen wird die 1 ein **erzeugendes Element** der additiven Gruppe genannt. Betrachten Sie stattdessen die sogenannte multiplikative Gruppe Z , die durch Multiplikation modulo n entsteht (das entspricht der diskreten Exponentialfunktion), so existieren auch hier erzeugende Elemente. Obwohl hier die Verhältnisse deutlich komplizierter sind, lässt sich mathematisch nachweisen, wann ein Element a ein erzeugendes Element einer multiplikativen Gruppe modulo n ist. Voraussetzung dafür ist, dass Modul n eine Primzahl ist.

Betrachten Sie noch einmal das Beispiel oben. Sie werden feststellen, dass die 2 ein erzeugendes Element der Gruppe Z_{19} ist: Als Ergebnisse der Berechnung $2^x \bmod 19$ erhalten Sie, wenn Sie alle x von 1 bis 18 durchlaufen haben, sämtliche Zahlen von 1 bis 18 – jeweils **genau ein Mal**. Diese Eigenschaft hat äußerst interessante Nebenwirkungen für die Kryptografie. In gewisser Weise ordnet die Tabelle von $2^x \bmod 19$ den x -Werten von 1 bis 18 neue y Werte (ebenfalls von 1 bis 18) in anderer Reihenfolge zu. Dies kann auch als Permutationstabelle verwendet werden.

Ein Alphabet, das Sie mit $2^x \bmod 19$ permutieren wollen, dürfte maximal 18 Zeichen haben, da die Gruppe nicht mehr Elemente besitzt. Dieses Problem können Sie jedoch dadurch lösen, dass Sie eine Gruppe mit mehr Elementen finden.

11.3 Diffie-Hellman-Schlüsseltausch

Ein Protokoll zum geheimen Schlüsseltausch

Diffie und Hellman waren zwar bei ihrer Suche nach einer Trapdoor-Einwegfunktion nicht erfolgreich. Gleichzeitig wurden sie aber auf die interessanten Eigenschaften der diskreten Exponentialfunktion aufmerksam und entwickelten ein Protokoll, das die Eigenschaften der diskreten Exponentialfunktion nutzt, um zwischen zwei Partnern sicher einen gemeinsamen Schlüssel zu ermitteln.

Aus der symmetrischen Kryptografie wissen Sie, dass ein gemeinsamer geheimer Schlüssel für das Ver- und Entschlüsseln erforderlich ist. Allerdings stellt sich das Problem, dass meist kein sicherer Kanal für den Schlüsseltausch vorhanden ist.

Ablauf des Schlüsseltauschs

Diffie und Hellman schlagen folgendes Protokoll vor:

Alice und Bob einigen sich beide auf eine Primzahl p und eine Zahl g ①. Die Zahl g muss erzeugendes Element der Gruppe Z_p sein. Nun wählt Alice für sich eine Zahl a und Bob eine Zahl b . Diese beiden Zahlen halten Alice und Bob jedoch geheim.

Alice berechnet: $\alpha = g^a \text{ mod } p$.

Bob berechnet: $\beta = g^b \text{ mod } p$.

Beide tauschen nun die soeben errechneten α und β ②.

Alice potenziert nun das von Bob erhaltene β mit ihrer geheimen Zahl a ③.

Bob potenziert das α von Alice mit seiner geheimen Zahl b ④.

Was haben Alice und Bob gerade berechnet?

Alice berechnet: $\beta^a \text{ mod } p = (g^b)^a \text{ mod } p = g^{ba} \text{ mod } p$.

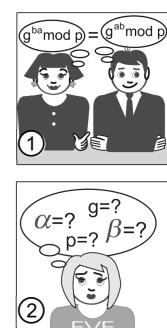
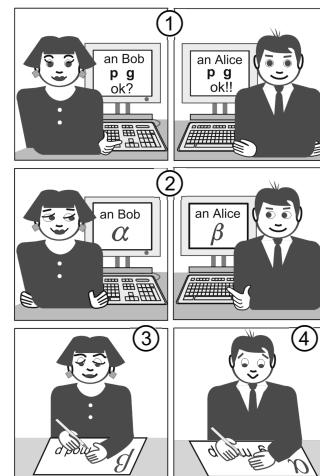
Bob berechnet: $\alpha^b \text{ mod } p = (g^a)^b \text{ mod } p = g^{ab} \text{ mod } p$.

Da für die beiden Exponenten das Kommutativgesetz gilt, ist $g^{ba} \text{ mod } p = g^{ab} \text{ mod } p$.

Diese gemeinsam ermittelte Zahl ist nun der neue Schlüssel für die weitere Kommunikation.

Alice und Bob haben also beide gerade dieselbe Zahl berechnet ①. Das Ergebnis dieser Berechnungen wurde niemals über einen öffentlichen Kanal übertragen und ist somit nur Alice und Bob bekannt. Wollte Eve den Schlüssel ebenfalls berechnen, so müsste sie diesen aus den öffentlich übertragenen Größen g , p , α und β selbst berechnen ②. Diese Berechnung allerdings läuft darauf hinaus, dass Eve entweder den diskreten Logarithmus von α oder aber β berechnen muss, um entweder a oder b zu erfahren. Wie Sie bereits gesehen haben, ist dies mitunter sehr schwierig.

Die Sicherheit des DH-Schlüsseltauschs hängt davon ab, in welcher Größenordnung die von Alice und Bob gewählte Zahl g und die Primzahl p liegen. Je größer diese beiden Zahlen gewählt werden, desto schwieriger wird es für Eve, den gemeinsamen Schlüssel zu errechnen.



Anwendungsmöglichkeiten des DH-Schlüsseltauschs

Alice und Bob können den gemeinsamen Schlüssel, der ja eine natürliche Zahl ist, als Schlüssel für ein symmetrisches Verschlüsselungsverfahren ihrer Wahl benutzen. Mithilfe des DH-Protokolls können beliebige Kommunikationspartner, ohne sich vorher getroffen zu haben, über einen unsicheren Kanal einen gemeinsamen Schlüssel aushandeln, wenn sie einen sicheren Kanal benötigen.

11.4 El-Gamal

Ein Public-Key-Verschlüsselungsverfahren

Basierend auf dem Diffie-Hellman-Protokoll schlug Taher El-Gamal eine Methode vor, wie das Diffie-Hellman-Protokoll nicht nur für den sicheren Schlüsseltausch, sondern auch als asymmetrisches Kryptosystem benutzt werden kann. Dabei steht im Mittelpunkt, dass das Generieren und Tauschen von Schlüsseln zeitlich nicht mit der Übermittlung einer Nachricht zusammenfallen muss. Das El-Gamal-Verschlüsselungsverfahren ist auch unter dem Namen **DSA** (Digital Signature Algorithmus) geläufig.

Ablauf von El-Gamal

Sämtliche Teilnehmer dieses Systems müssen sich zuerst auf eine Verschlüsselungsmethode, eine Primzahl p und eine natürliche Zahl g einigen. Anschließend wählt jeder Teilnehmer T eine natürliche Zahl t . Diese betrachtet er als seinen privaten Schlüssel und hält diesen geheim.

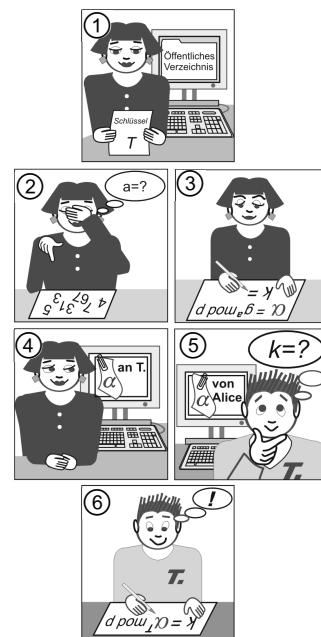
Jeder Teilnehmer berechnet mithilfe seiner Zahl t den Wert $\tau = g^t \text{ mod } p$ und stellt diesen Wert in das öffentlich zugängliche Verzeichnis. Aus den öffentlich verfügbaren Daten ist es für Eve nicht möglich, den geheimen Schlüssel t zu errechnen, da sie hierfür den diskreten Logarithmus $\log(\tau)$ kennen müsste.

Will Alice nun an den Teilnehmer T eine Nachricht senden, so schlägt sie zuerst den öffentlichen Schlüssel τ des Teilnehmers T im Verzeichnis nach ①.

Alice wählt einen zufälligen Wert a ② und berechnet selbst: $\alpha = g^a \text{ mod } p$.

Sie errechnet die Zahl k aus $\tau^a \text{ mod } p$. Diese Zahl ($k = \text{key}$) ist der Schlüssel, den sie für die symmetrische Verschlüsselung ihrer Nachricht benutzt ③.

Alice sendet die nun verschlüsselte Nachricht zusammen mit α an T ④.



Möchte der Empfänger die Nachricht entschlüsseln, so muss er zuerst den Schlüssel k errechnen, den Alice für die Nachricht verwendet hat ⑤. Dazu verwendet er das von Alice an die Nachricht angefügte α .

Er berechnet: $k = \alpha^t \bmod p$ ⑥.

Analog zu Diffie-Hellman gilt auch hier, dass $\alpha^t \bmod p = g^{at} \bmod p$ ist und $\tau^a \bmod p = g^{ta} \bmod p$ ist. Somit haben Alice und der Teilnehmer T denselben Schlüssel k errechnet.

Das System nach El-Gamal benötigt so viele öffentliche Schlüssel, wie es Teilnehmer gibt. Dies erklärt sich dadurch, dass das jeweilige τ des Teilnehmers veröffentlicht wird.

Wählen die Sender der Nachrichten für jede Nachricht eine Zufallszahl (in diesem Beispiel a), so wird jede in diesem System versandte Nachricht mit einem **anderen** Schlüssel verschlüsselt – auch wenn sie an denselben Empfänger geht.

Schwächen von El-Gamal/DSA

El-Gamal/DSA unterstützte über einen langen Zeitraum nur Schlüssel mit einer Länge von nur 1024 Bit. So verwenden nicht mehr aktuelle GnuPG-Implementierungen diese Einstellung. Des Weiteren werden für die Erzeugung **jeder** Signatur ungewöhnliche Zufallszahlen benötigt. Sofern der Generator für die Erzeugung der Zahlen auch nur punktuell kompromittiert wird, ist keine Sicherheit mehr gegeben. Dies kann z. B. durch einen Patch des Quellcodes erzeugt werden. Auch gewöhnliche (in der Normalverteilung öfter genutzte) Zufallszahlen stellen ein Sicherheitsdefizit dar.

11.5 RSA

Forschungsarbeit des RSA-Teams

Dadurch, dass Diffie und Hellman bei ihrer Suche nach einer Trapdoor-Einwegfunktion erfolglos blieben, wurde ein anderes Forscherteam auf diese Thematik aufmerksam: **Ronald Rivest, Adi Shamir** und **Len Adleman**. Laut Adi Shamir war das ursprünglich erklärte Ziel, zu beweisen, dass es keine Trapdoor-Einwegfunktionen geben kann.

Ironischerweise sind sie mit diesem Vorhaben nicht nur gescheitert – sie haben sogar bei dem Versuch, die Unmöglichkeit einer Trapdoor-Einwegfunktion nachzuweisen, genau eine solche entdeckt. 1977 entwickelten sie somit das bekannteste Public-Key-Verfahren, das nach den Initialen der drei Erfinder benannt ist: RSA.

Ablauf von RSA

Für RSA benötigen Sie das RSA-Modul, hier als n bezeichnet. Dies ist das Produkt zweier Primzahlen, denn eine große Zahl zu erzeugen, indem man zwei Primzahlen miteinander multipliziert, ist relativ einfach. Die Faktorisierung (Zerlegung einer Zahl in ihre Primfaktoren) des Ergebnisses ist dagegen ungleich aufwendiger.

Neben dem RSA-Modul n benötigen Sie einen Wert e und dessen modulares Invers d . Diese werden wie folgt gebildet:

Es werden zufällig zwei unterschiedliche Primzahlen p und q gewählt. In der Praxis werden hierzu ab einer Zahl der gewünschten Mindestgröße Primzahltests durchgeführt, bis eine Primzahl gefunden wird. Dann wird für eine zweite Mindestzahl, die deutlich größer als die erste Primzahl sein sollte, identisch vorgegangen. Die so erhaltenen Primzahlen werden hier p und q genannt.

z. B. $p = 23$ und $q = 47$

Bilden Sie das Produkt n aus den beiden Primzahlen p und q . Diese wird als RSA-Modul bezeichnet. Das RSA-Modul ist ein Teil sowohl des privaten als auch des öffentlichen Schlüssels.

z. B. $23 * 47 = 1081$.

Berechnen Sie nun die Euler'sche φ -Funktion (Phi) von n : $\varphi(n) = (p - 1) * (q - 1)$

z. B. $(23 - 1) * (47 - 1) = 1012$

Für den öffentlichen Schlüssel benötigen Sie nun einen Wert e , der teilerfremd zu $\varphi(n)$ ist. Teilerfremd bedeutet, dass er keinen gemeinsamen Teiler mit $\varphi(n)$ haben darf (relativ prim). Für diesen muss gelten $1 < e < \varphi(n)$. In der Praxis wird hier häufig $2^{16} + 1 = 65537$ verwendet.

z. B. $e = 3$

Berechnen Sie nun für den privaten Schlüssel den Wert d , der das modulare Invers von e darstellt:

$(e * d) \bmod ((p - 1) * (q - 1)) = 1$.

z. B. $(3 * 675) \bmod ((23 - 1) * (47 - 1)) = 1$

Nun sollten die Primzahlen p und q und die $\varphi(n)$ sicher gelöscht werden, um die Schlüsselkomponenten n , e (öffentliche) und n , d (privat) zu schützen.

Zum Verschlüsseln wird nun die Nachricht m mit dem öffentlichen Schlüssel e potenziert und anschließend modulo n angewandt: $m^e \bmod n = c$. Für die Klartextnachricht "5" wäre die verschlüsselte Nachricht c in unserem Beispiel:

$$5^3 \bmod 1081 = 125$$

Der Empfänger entschlüsselt nun die Nachricht c , indem er die verschlüsselte Nachricht c mit seinem geheimen Schlüssel d potenziert und anschließend modulo n anwendet: $c^d \bmod n = m$

$$125^{675} \bmod 1081 = 5$$

Wie Sie sehen, ist das RSA-Verfahren eine echte Trapdoor-Einwegfunktion. Das Potenzieren einer Nachricht modulo n wäre nur durch den diskreten Logarithmus umkehrbar, dessen Berechnung bei großem n extrem viel Zeitaufwand mit sich bringen würde. Das Trapdoor ist hier das Potenzieren mit der geheimen Zusatzinformation, also mit dem geheimen Schlüssel d des Teilnehmers. Als inverses Element zu e gewählt, macht d die Potenzierung der Nachricht mit e rückgängig.

Primzahlen und Sicherheit

Mit obigen Beispielen und Erklärungen wird klar, warum Primzahlen vor allem in der asymmetrischen Kryptografie eine wichtige Rolle spielen. Sie werden einerseits benötigt, um die notwendigen Voraussetzungen für das Erzeugen einer multiplikativen Gruppe und das Vorhandensein eines inversen Elements zu schaffen. Die Schwierigkeit für einen Angreifer, einen RSA-Schlüssel nach dem Abfangen einer verschlüsselten Nachricht zu errechnen, hängt davon ab, ob er die mit dem öffentlichen Schlüssel bekannt gegebene Zahl n (die als Modul benutzt wird) in ihre Primfaktoren p und q zerlegen kann.

Gelingt dies dem Angreifer, so kann er $\phi(n) = (p - 1) * (q - 1)$ berechnen und anschließend mit der Lösung der Gleichung $e * d = 1 \text{ mod } \phi(n)$ den geheimen Schlüssel d berechnen. Dann wäre er in der Lage, die für den Teilnehmer verschlüsselten Nachrichten selbst zu entschlüsseln.

Um die Faktorisierung schwierig zu machen, ist es also angebracht, große Primzahlen als Basis für n zu nehmen. Die oft verwendete Bitlänge bei asymmetrischen Algorithmen gibt Auskunft darüber, in welcher Größenordnung die verwendeten Werte sind. Die Bitlängen von symmetrischen und asymmetrischen Schlüsseln sind nicht direkt miteinander zu vergleichen. So kann niemand behaupten, ein RSA-Schlüssel wäre mit 2048 Bit genau 16-mal so sicher wie ein 128-Bit-AES-Schlüssel. Hier spielt die Tatsache eine Rolle, dass von den 2^{128} möglichen AES-Schlüsseln nach dem gegenwärtigen Wissensstand alle als Verschlüsselungsschlüssel infrage kommen. Bei 2^{2048} Möglichkeiten für einen asymmetrischen Schlüssel sind aber nicht alle darstellbaren Zahlen als Schlüssel bzw. Modul verwendbar, da Schlüssel und Module bestimmte Eigenschaften erfüllen müssen.

Rechenzeit und hybride Verschlüsselung

Aufgrund der verwendeten mathematischen Operationen sind asymmetrische Algorithmen bei der Ausführung etwa um Faktor 1000 langsamer als symmetrische Methoden. Letztere arbeiten mit einfachen Bitoperationen, Verschiebungen und Permutationstabellen.

Um die Vorteile beider Verfahren zu nutzen, wird in kryptografischen Systemen gerne auf hybride Lösungen zurückgegriffen. Oft wird ein Public-Key-Verfahren für das Schlüsselmanagement verwendet, zum Versenden der eigentlichen Nachrichten aber ein symmetrisches Verfahren.

El-Gamal kommt diesem Prinzip schon von vorneherein sehr nahe, da in El-Gamal der Diffie-Hellman-Schlüsseltausch benutzt wird, um einen gemeinsamen, geheimen symmetrischen Schlüssel für den Transport der Nachricht zu generieren.

Beim hybriden RSA generiert Alice eine Zufallszahl. Diese Zahl benutzt Alice als symmetrischen Schlüssel (Sitzungs-Schlüssel), um damit eine Nachricht an Bob zu verschlüsseln. Damit Bob diese Nachricht wieder entschlüsseln kann, wird der gerade benutzte Sitzungs-Schlüssel (Session-Key) mithilfe von RSA verschlüsselt und an die Nachricht angehängt. Bob empfängt die Nachricht und kann mithilfe seines privaten Schlüssels den Sitzungs-Schlüssel wieder dechiffrieren. Mit diesem Sitzungsschlüssel entschlüsselt er dann die eigentliche Nachricht.

Der Vorteil dieser Vorgehensweise liegt darin, dass die Nachricht mit einem schnellen symmetrischen Verfahren verschlüsselt werden kann. Das langsame asymmetrische Verfahren wird nur gebraucht, um den vergleichsweise kurzen Sitzungs-Schlüssel zu sichern.

Schwächen von RSA

RSA basiert auf einem Faktorisierungsalgorithmus. Untersuchungen zeigen, dass bei einer Verbesserung der Mechanismen für die Faktorisierung eine Schlüssellänge von 1024 Bit nicht als sicher angesehen werden kann. Dass u. a. die **NSA** (National Security Agency) diese Schlüssel dekodieren kann, darf als möglich angesehen werden, obwohl der explizite Beweis hierzu bisher fehlt. Eine Schlüssellänge von 2048 Bit gilt hier als sicher (Stand 2015). Auch der genutzte Standard **PKCS#1 1.5** (Public Key Cryptography Standards) ist bei nicht sicherer Implementierung eine Schwachstelle (vgl. Abschnitt 11.8).

11.6 Digitale Signatur

Eine digitale Signatur ist eine Art Unterschrift unter einem digitalen Dokument. Dabei wird mittels kryptografischer Verfahren eine genaue und eindeutige Zuordnung dieses Dokuments zu seinem Absender sichergestellt, ersetzt quasi seine eigenhändige Unterschrift.

Authentifizierung von Nachrichten

Basierend auf dem RSA-Verfahren ist es relativ leicht möglich, eine digitale Signatur zu erstellen:

Alice möchte eine Nachricht digital signieren. Dazu besitzt sie ein Schlüsselpaar d und e , das sie nach RSA erstellt hat.

Alice bildet: $c = m^d \text{ mod } n$. Sie verschlüsselt also ihre Nachricht, allerdings benutzt sie für diese Operation ihren privaten Schlüssel. Die so unterschriebene Nachricht sendet sie an Bob.

Zur Prüfung der Unterschrift benutzt Bob den von Alice erhaltenen oder bereits publizierten öffentlichen Schlüssel. Er berechnet: $m = c^e \text{ mod } n$. Er potenziert die von Alice erhaltene Nachricht mit dem öffentlichen Schlüssel und erhält wieder die Originalnachricht m . Dies ist möglich, da d und e zueinander inverse Elemente sind.

Was hat Bob damit gewonnen? Erhält Bob nach dem Potenzieren der Nachricht mit dem öffentlichen Schlüssel e von Alice die Nachricht, so weiß Bob, dass diese Nachricht nur von Alice signiert worden sein kann. Alice ist die einzige Person, die im Besitz des zu ihrem öffentlichen Schlüssel passenden geheimen Schlüssels ist.

Erhält Bob nach dem Potenzieren keine sinnvolle Nachricht, so wurde nicht der korrekte private Schlüssel benutzt oder die Signatur ist gefälscht worden.

Um die digitale Signatur prüfen zu können, benötigt Bob den öffentlichen Schlüssel von Alice. In diesem Fall ist er sogar gezwungen, die Signatur zu prüfen, da er ohne Entschlüsselung die Nachricht nicht lesen kann. Da es nicht praktikabel ist, die Nachricht in ihrer kompletten Länge so zu signieren, dass ein Lesen ohne Prüfung nicht möglich ist, wäre eine weitere Alternative, die Nachricht einmal im Klartext zu senden und daran die signierte Nachricht anzuhängen. Das würde allerdings ein Anwachsen der Nachrichtenlänge auf das Doppelte bewirken und ist somit auch ungeeignet. Eine Lösungsmöglichkeit bieten sogenannte Hashfunktionen.

11.7 Hashfunktionen

Definition einer Hashfunktion

Der Einsatz von sogenannten Hashfunktionen hat sich zur Erkennung von Fehlern im Informatik-einsatz schon länger bewährt und stellt im Zusammenhang mit digitalen Signaturen eine willkommene Alternative dar. Eine Hashfunktion ermittelt aus einer beliebigen Menge an Eingabedaten einen Hashwert. Dieser Hashwert ist quasi eine Prüfsumme und hat immer dieselbe Länge.

Hashfunktionen sind durch die Länge des berechneten Wertes gekennzeichnet. So liefert eine Funktion mit 56 Bit nur 2^{56} verschiedene Hashwerte, während eine Funktion mit 160 Bits theoretisch 2^{160} verschiedene Hashwerte liefern kann.

Sinn eines Hashwertes ist es, als Prüfsumme für eine bestimmte Eingabe (z. B. einen Dateiinhalt oder eine Nachricht) zu fungieren. Wird der Inhalt der Datei oder der entsprechenden Nachricht geändert, so sollte auch der Hashwert ein anderer sein. Die Chance, dass zwei verschiedene Nachrichten existieren, die denselben Hashwert besitzen, sollte möglichst klein sein. Eine größere Anzahl von möglichen Hashwerten wirkt sich hier positiv aus.

Eine Hashfunktion ist in gewisser Weise auch eine Einwegfunktion, da Sie leicht den Hashwert h einer Nachricht m bilden können $h = f(m)$, aber bei gegebenem Hashwert die ursprüngliche Nachricht nicht rekonstruieren können. Eine gute Hashfunktion soll wie ein Fingerabdruck der Nachricht sein – der Fingerabdruck jedes Menschen ist einzigartig, doch aus dem Fingerabdruck selbst kann man den Menschen nicht rekonstruieren.

Für die Erzeugung einer digitalen Signatur kann eine Hashfunktion so eingesetzt werden, dass die Nachricht m unverschlüsselt übertragen wird. Alice bildet den Hashwert h dieser Nachricht und chiffriert diesen mit ihrem privaten Schlüssel d ①.



Bob empfängt nun die Nachricht und kann diese lesen. Möchte er die Unterschrift von Alice prüfen, so entschlüsselt er den angehängten Hashwert mit Alice' öffentlichem Schlüssel. Bob bildet anschließend auch selbst den Hashwert der empfangenen Nachricht ②. Stimmen beide Hashwerte überein, so stammt die empfangene Nachricht wirklich von Alice ③.

Bekannte Hashfunktionen sind u. a.:

- | | | |
|-----------------|----------------|----------------|
| ✓ MD5 (1992) | ✓ SHA-1 (1995) | ✓ SHA-2 (2001) |
| ✓ RIPEMD (1992) | ✓ Tiger (1996) | ✓ SHA-3 (2012) |

MD5

Message Digest 5 ist einer der bekanntesten und immer noch genutzten Hashalgorithmen. Ron Rivest, einer der RSA-Erfinder, hat diesen Algorithmus entwickelt. MD5 berechnet 128-Bit-Werte, indem in mehreren Runden, ähnlich wie ein Verschlüsselungsverfahren, die zu hashenden Daten durch eine Reihe binärer Operationen geführt werden.

Obwohl inzwischen von der Verwendung von MD5 aufgrund möglicher Schwachstellen („Geburtstagsangriff“, vgl. Abschnitt 11.8) abgeraten wird, ist MD5 in vorhandenen Systemen aus Gründen der Kompatibilität mitunter nicht zu ersetzen.

SHA

SHA, der Secure Hash Algorithm, wurde 1993 von der NSA entwickelt. Als Grundlage diente der MD4-Algorithmus und Vorläufer von Rivests MD5. Nachdem 1995 ein Sicherheitsproblem gefunden wurde, das nicht veröffentlicht wurde, veröffentlichte die NSA den Standard SHA-1 als Abhilfe, der in dieser Form noch in vielen modernen kryptografischen Paketen zum Einsatz kommt.

SHA-1 berechnet einen 160-Bit-Hashwert. SHA-1 ist derzeit einer der am häufigsten benutzten Hashalgorithmen. Das Ende seiner Nutzungsdauer zeichnet sich jedoch schon ab, da im Februar 2005 durch erfolgreiche kryptografische Angriffe auf diesen Algorithmus die Komplexität deutlich reduziert werden konnte. Mit steigender Rechenleistung der Computer und eventuell weiteren Durchbrüchen in der Kryptologie steigt also die Wahrscheinlichkeit, dass ein Text so gestaltet werden kann, dass eine bestimmte Prüfsumme erzielt wird.

Das amerikanische NIST (National Institute of Standards an Technology) empfiehlt, wenn möglich die Hashfunktion SHA-1 durch Algorithmen der Familie **SHA-2** zu ersetzen, die Prüfsummen mit variabler Länge größer als 160 Bits berechnen. Dementsprechend heißen diese Hashfunktionen dann entsprechend der Hashlänge beispielsweise SHA-256, SHA-384 oder SHA-512.

Der neue Standard **SHA-3** beruhte auf einer Ausschreibung des NIST, die im Jahre 2012 erfolgte. Er nutzt hierbei einen neuen mathematischen Algorithmus, der sich signifikant von SHA-2 unterscheidet. Dabei sind Schlüssellängen von 224, 256, 384 und 512 Bit vorgesehen. Das NIST wollte aus Performanceerwägungen jedoch nur 128 und 256 Bit zum Standard erklären. Massive Widerstände, auch aufgrund der NSA-Aktivitäten, verhinderten dies.

11.8 Schwachstellen in RSA

Wie sicher ist RSA wirklich?

Obwohl Public-Key-Systeme relativ sicher erscheinen, gibt es einige Probleme, die in dieser Form bei symmetrischen Verfahren nicht bekannt waren.

Die Sicherheit asymmetrischer Verfahren beruht auf der Tatsache, dass Alice beim Versenden einer Nachricht an Bob auch sicher sein kann, dass der Schlüssel, den sie zum Chiffrieren an Bob verwendet, auch wirklich Bobs öffentlicher Schlüssel ist.

Haben Alice und Bob sich nicht persönlich getroffen und ihre öffentlichen Schlüssel ausgetauscht, so wird Alice den Schlüssel von Bob aus einem öffentlich zugänglichen Verzeichnis (in der Regel ein Public-Key-Server) beziehen.

Möchte Mallory in Zukunft alle Nachrichten lesen können, die Alice an Bob schreibt, so erzeugt er selbst ein Schlüsselpaar und stellt den öffentlichen Schlüssel unter dem Namen Bob in das Verzeichnis. Wenn Alice nun eine Nachricht schreibt, benutzt sie einen Schlüssel, zu dem Mallory den privaten Schlüssel besitzt und mit dem er somit die Nachrichten entschlüsseln kann.

Mallory könnte sogar noch weiter gehen und Bob ebenfalls seinen eigenen Schlüssel unterschieben. Diesmal würde er diesen Schlüssel als den von Alice deklarieren. In so einem Fall könnte Mallory dann die von Alice kommenden Nachrichten lesen und verschlüsselt an Bob weiterversenden. Prüft Bob die Unterschrift der empfangenen Dokumente, so prüft er die von Mallory erzeugte Signatur anhand von Mallorys Public-Key. Bob wäre also in dem fatalen Irrglauben, Alice hätte ihm ein digital signiertes und verschlüsseltes Dokument geschickt.

Dieser Angriff ist auch als „**Man-in-the-Middle-Attack**“ bekannt. Eine gewisse Absicherung gegen die Fälschung von öffentlichen Schlüsseln stellen die verschiedenen Infrastrukturen zur Verteilung von öffentlichen Schlüsseln (Public Key Infrastructure, PKI) dar.

Chosen-Ciphertext-Angriff

Bei einem Chosen-Ciphertext-Angriff möchte Mallory eine Nachricht lesen, die von Alice an Bob geschickt wurde. Dazu kann Mallory das Verschlüsselungssystem kompromittieren, indem er einen Chiffretext seiner Wahl erstellt.

Mallory fängt die verschlüsselte Nachricht c ab, die Alice an Bob mit dessen öffentlichem Schlüssel chiffriert sendet, und besorgt sich den öffentlichen Schlüssel e von Bob. Zusätzlich wählt Mallory eine Zufallszahl r , die kleiner als n und teilerfremd zu n ist. Da er r selbst gewählt hat, kann Bob mithilfe der bekannten Gleichung das inverse Element von r berechnen: r^{-1} .

Zuerst verschlüsselt Mallory das zufällige r :

$$x = r^e \bmod n$$

Um sein selbst gewähltes r mit der abgefangenen Nachricht verarbeiten zu können, multipliziert er die Nachricht c mit seinem errechneten x :

$$y = x * c \bmod n$$

Mallory überredet jetzt Bob dazu, die gerade errechnete Nachricht y digital mit seinem privaten Schlüssel zu signieren. Lässt Bob sich darauf ein, so berechnet Bob Folgendes:

$u = y^d \bmod n$, da Bob für eine Signatur mit RSA die Nachricht mit seinem privaten Schlüssel potenzieren muss.

Mallory ist daraufhin in der Lage, mit dem von Bob berechneten Wert weiterzuarbeiten. Er multipliziert das erhaltene u mit dem inversen Element r^{-1} . Diese Berechnung kann wie folgt aufgelöst werden:

$$\begin{aligned} r^{-1} * u \bmod n &= r^{-1} * y^d \bmod n \\ &= r^{-1} * x^d * c^d \bmod n \\ &= r^{-1} * (r^e)^d * c^d \bmod n \\ &= r^{-1} * r * c^d \bmod n \end{aligned}$$

Da sich die Exponenten e und d gegenseitig genauso aufheben wie auch r und r^{-1} , bleibt schließlich als Einziges auf der rechten Seite Folgendes übrig:

$$= c^d \bmod n$$

Die Chiffrennachricht c potenziert mit dem privaten Schlüssel d von Bob ist aber nichts anderes als die Klartextnachricht. Mallory hat es also geschafft, sich indirekt unter Mitwirkung von Bob einen Text entschlüsseln zu lassen, der nur für Bob bestimmt war. Durch Anwendung seines privaten Schlüssels auf eine ihm unbekannte Nachricht hat Bob Mallory die Arbeit erspart, Modul n in seine Primfaktoren zerlegen zu müssen.



Fazit: Unterschreiben Sie nie irgendwelche binären Daten unbekannten Inhalts mit Ihrem privaten Schlüssel – es könnte sich um einen Chosen-Ciphertext-Angriff handeln.

Der Geburtstagsangriff

Eine weitere Klasse von Angriffen basiert auf dem Phänomen, dass es zwar sehr unwahrscheinlich ist, dass eine bestimmte Person an genau demselben Tag wie Sie Geburtstag hat. Es ist aber durchaus wahrscheinlich, dass in einer Gruppe aus mehreren Personen zwei beliebige Personen am gleichen Tag Geburtstag haben.

Im ersten Fall ist von einer Wahrscheinlichkeit von 1:365 auszugehen. Im zweiten Fall ist die Wahrscheinlichkeit, dass zwei Personen in einem Raum mit 23 Leuten am selben Tag Geburtstag haben, schon ca. 50 %.

Übertragen auf die Kryptografie, Hashwerte und digitale Signaturen bedeutet das: Wenn Sie ein Dokument bzw. dessen Hashwert haben und ein anderes Dokument suchen, das genau denselben Hashwert hat, so ist die Wahrscheinlichkeit, dass Sie ein zweites finden, relativ gering.

Erstellen Sie aber eine große Menge an Dokumenten und suchen nach irgendwelchen zwei Dokumenten, die denselben Hashwert haben, so ist die Wahrscheinlichkeit, dass Sie erfolgreich sind, bedeutend größer. Weisen verschiedene Dokumente den gleichen Hashwert auf, spricht man von einer Kollision.

Dieser Sachverhalt könnte folgendermaßen ausgenutzt werden: Mallory erstellt zwei Versionen eines Dokuments: ein Überweisungsformular, in dem Alice an Bob einen kleinen Betrag überweist, und eine zweite Version, in der Alice an Mallory eine große Summe überweist. Anschließend erzeugt Mallory von jedem Dokument zahlreiche Versionen, die aber denselben Inhalt haben. Variationen könnte Mallory durch das Einfügen bzw. Weglassen von Returns, Leerzeichen und Tabulatoren produzieren, die im endgültigen Dokument nicht zu sehen sind.

Danach sucht Mallory nach zwei Dokumenten, die denselben Hashwert besitzen. Findet Mallory ein solches Paar (die Wahrscheinlichkeit dafür ist relativ groß), so legt er das eine Dokument Alice zur Unterschrift vor. Nachdem Alice die Überweisung an Bob mit dem kleinen Betrag digital unterschrieben hat, trennt Mallory die Signatur vom Dokument ab und hängt sie unter das Dokument, in dem Alice eine große Summe an Mallory überweist.

Da die digitale Unterschrift dem Verschlüsseln des Hashwerts des Dokuments mit dem privaten Schlüssel entspricht und beide Dokumente denselben Hashwert besitzen, ist Alices Unterschrift auch auf dem ruinösen Dokument gültig. Mallory hat sich also erfolgreich eine Unterschrift von Alice erschlichen.



Fazit: Unterschreiben Sie niemals ein Dokument (auch wenn es für Sie im Klartext abgefasst ist), das Sie nicht selbst verfasst haben. Es sei denn, Sie nehmen vor der Unterzeichnung einige subtile Änderungen vor. Beispielsweise könnten Sie vor dem Unterzeichnen selbst noch Leerzeichen hinzufügen oder entfernen. Damit ändern Sie wiederum den Hashwert der Ihnen vorgelegten Daten und machen Mallorys gefährliches Zweitdokument wertlos.

11.9 Public Key Infrastructure

Was ist eine PKI?

Anhand der Probleme, die sich durch eine Man-in-the-Middle-Attacke ergeben, haben Sie gesehen, dass trotz des einfachen Schlüsselmanagements und der Verschlüsselungsfunktionen eine wesentliche Schwachstelle bei asymmetrischen Verfahren besteht. Wenn Sie mit einem anderen Teilnehmer verschlüsselt kommunizieren wollen oder auch nur dessen digitale Unterschrift überprüfen, benötigen Sie dessen Public Key. Wenn Sie diesen allerdings über eine ungesicherte Quelle (z. B. einen Key-Server aus dem Internet) beziehen, haben Sie keine Garantie, dass der öffentliche Schlüssel wirklich zu Ihrem Partner gehört.

In asymmetrischen Systemen ist deswegen die Möglichkeit vorgesehen, die **Echtheit eines Schlüssels durch eine digitale Signatur einer Zertifizierungsstelle zu bestätigen**. Im Detail könnte ein solcher Vorgang so ablaufen:

Alice möchte mit Bob kommunizieren, hat aber dessen öffentlichen Schlüssel noch nicht. Beide befürchten, dass Mallory die Daten auf dem Key-Server manipulieren könnte, um Alice und Bob seinen eigenen Schlüssel unterzuschieben ①.

Deswegen zeigen Alice und Bob sich darauf, dass Bob seinen öffentlichen Schlüssel Trent aushändigt, der als Treuhänder fungiert und sich persönlich davon überzeugen kann, dass der ihm übergebene Schlüssel wirklich Bob gehört ②. Alice besitzt bereits Trevts öffentlichen Schlüssel, da sie diesen persönlich von ihm bekommen hat.

Trent beglaubigt die Echtheit von Bobs Schlüssel nun durch seine digitale Unterschrift auf dem Schlüssel ③. Er stellt ihm ein sogenanntes **Zertifikat** aus. Den so signierten öffentlichen Schlüssel stellt er nun zum Download bereit.

Alice holt Bobs signierten Schlüssel und kann anhand der Unterschrift von Trent prüfen, ob sie diesen Schlüssel unmodifiziert erhalten hat ④. Wenn das der Fall ist, so benutzt Alice den Schlüssel, um Bob eine verschlüsselte Nachricht zukommen zu lassen.

Dieses Beispiel macht deutlich, dass Alice und Bob jetzt sicher miteinander kommunizieren können, ohne dass sie persönlich die Schlüssel tauschen müssen. Sie haben diese Aufgabe an einen Dritten (den Treuhänder Trent) ausgelagert, dem beide vertrauen müssen.



Aufgaben einer PKI

Die Hauptaufgaben einer Schlüsselinfrastruktur sind die Bildung von Vertrauensbeziehungen und das Ausstellen von Echtheitszertifikaten. Zwei bekannte Ansätze sind:

- ✓ OpenPGP, ein dezentraler Standard
- ✓ X.509, das zentral und hierarchisch organisiert ist

OpenPGP

Das Verschlüsselungspaket PGP (Pretty Good Privacy), das Anfang der 90er-Jahre von Phil Zimmermann als Freeware herausgegeben wurde, war die Grundlage für den OpenPGP-Standard, der von Zimmermann stark mitbeeinflusst wurde. Bezuglich der Schlüsselinfrastruktur und der Vertrauensbeziehungen gilt hier ein Konzept, das als **Web of Trust** (Vertrauensnetzwerk) bekannt ist.

Bei OpenPGP kann jeder Teilnehmer am Verschlüsselungssystem, der ein gültiges Schlüsselpaar besitzt, selbst die Echtheit anderer Schlüssel durch seine Unterschrift bestätigen. Die Idee hierbei ist, dass Personen, die die Echtheit der Schlüssel auf ihrem elektronischen Schlüsselbund (die Sammlung aller bekannten öffentlichen Schlüssel) bestätigt haben, diese Schlüssel signieren und die signierten Schlüssel wiederum öffentlich verfügbar machen.

Bezieht nun ein neuer Teilnehmer so unterschriebene öffentliche Schlüssel, kann es sein, dass ein öffentlicher Schlüssel bereits mehrere Signaturen trägt. Vertraut der Teilnehmer mindestens einem der unterzeichnenden Schlüssel, so ist der neue öffentliche Schlüssel für ihn gültig.

Jeder Teilnehmer im Web of Trust kann selbst festlegen, welchen Schlüsseln er vertrauen möchte. Dies definiert er darüber, welche Schlüssel **Trusted Introducer** sein können und welche nicht. Zusätzlich ist es auch möglich, marginales Vertrauen in bestimmte Schlüssel anzugeben, sodass zum Beispiel 2 Unterschriften auf einem neuen Schlüssel nötig sind, damit der Teilnehmer diesen Schlüssel als gültig akzeptiert.

Die Sicherheit eines Web of Trust hängt zum einen von der Disziplin der Teilnehmer ab und zum anderen von der Gründlichkeit, mit der diese sich gegenseitig ihre öffentlichen Schlüssel signieren. Würden sie ohne weitere Prüfung einen öffentlichen Schlüssel signieren, den man ihnen vorlegt, so wäre das Vertrauensnetz damit ausgehebelt, weil sich möglicherweise Dritte auf die gerade geleistete Unterschrift verlassen.

X.509

Das X.509 sieht in seiner Infrastruktur eine streng vorgegebene Hierarchie vor. Es gibt eine Zertifizierungsinstanz an der Spitze, der alle Teilnehmer vertrauen müssen: die Certificate Authority (CA). Die CA ist in der Grundidee die Einzige, die Zertifikate ausstellen kann. Wenn das Netz allerdings sehr groß ist, kann die CA auch Zertifikate ausstellen, die Sub-CAs gestatten, wiederum selbst Zertifikate auszustellen. Dies ermöglicht eine Delegation der Arbeit und den hierarchischen Aufbau des Vertrauensnetzwerkes.

Möchte Alice eine sichere Nachricht an Bob senden und erhält dazu von Bob ein Zertifikat nach X.509, so enthält dies folgende Daten: den öffentlichen Schlüssel von Bob und die Signatur der für Bob zuständigen CA, die ihm das Zertifikat ausgestellt hat.

Die CA, die Bobs Zertifikat ausgestellt hat, kennt Alice nicht, allerdings befinden sich Alice und Bob in derselben Hierarchie. Die unterzeichnende CA besitzt ebenfalls ein Zertifikat, das von der nächsthöheren CA in der Organisation ausgestellt wurde.

In diesem Beispiel ist die nächsthöhere CA diejenige an der Spitze, die sogenannte **Root-CA**. Alice kann das Zertifikat der Root-CA überprüfen, da in X.509 die Teilnehmer eines Systems der gemeinsamen Root vertrauen müssen. Alice erkennt also das Sub-CA-Zertifikat als gültig an und weiß somit, dass sie den Zertifikaten, die die Sub-CA ausstellt, ebenfalls vertrauen kann. Sie kann also Bobs Zertifikat vertrauen.

Eine PKI nach X.509 ist in vielen kommerziellen Produkten implementiert. Die bekannteste Anwendung ist die Authentifizierung eines Webservers über ein entsprechendes Zertifikat im Browser. Der Benutzer kann somit sicher sein, dass die Website, die er gerade besucht, wirklich vom angezeigten Server stammt.

Damit der Internetbrowser die Zertifikate der Websites auch überprüfen kann, sind die Zertifikate der Root-CAs entweder im Browser oder gleich schon im Betriebssystem verankert. Bekannte Root-CAs unter den vorinstallierten Browserzertifikaten sind: Deutsche Telekom, Verisign, Thawte.

Selbst erstellte Zertifikate

Ein zunehmendes Problem stellen Zertifikate dar, die Server sich selber ausstellen. Mit diesen kann auch HTTPS-Verschlüsselung betrieben werden. Sie stammen aber von keiner vertrauenswürdigen Quelle. Benutzer tendieren jedoch dazu, zunehmend auch diese unsicheren Zertifikate relativ unbedacht zu akzeptieren, wodurch die Netzwerksicherheit unterminiert wird.

11.10 Übung

Fragen zur asymmetrischen Kryptografie

Übungsdatei: --

Ergebnisdatei: uebung11.pdf

1. Was ist ein Hashwert und welche Hashfunktionen werden hauptsächlich eingesetzt?
2. Was verstehen Sie unter PKI?
3. Welche Aussage trifft auf die asymmetrische Verschlüsselung zu?

a	Alle Teilnehmer teilen sich einen Schlüssel.
b	Ein Schlüsselpaar setzt sich aus einem Private Key und einem Public Key zusammen.
c	Für jeden Teilnehmer existiert ein Schlüsselpaar.
d	Der Private Key wird an alle Teilnehmer weitergereicht.
e	Der Public Key kann gefahrlos verteilt werden.
f	Es besteht keine Möglichkeit, aus dem Public Key den Private Key zu berechnen.

4. Was ist eine digitale Signatur?

a	eine Verschlüsselung
b	eine elektronische Unterschrift basierend auf der symmetrischen Verschlüsselung
c	ein Hashwert
d	eine elektronische Unterschrift basierend auf der asymmetrischen Verschlüsselung

5. Um von Public-Key-Infrastruktur (PKI) sprechen zu können, müssen folgende Funktionen zur Verfügung gestellt werden:

a	Es werden Zertifizierungsstellen, Tools für die Schlüssel- und Zertifikatsverwaltung und Anwendungen, die öffentliche Schüssel verwenden können, zur Verfügung gestellt.
b	Public-Key-Infrastruktur ist eine Kombination von Software, Verschlüsselungs-technologien und Diensten, die den Umgang mit öffentlichen Schlüsseln ermöglichen.
c	Zur Verfügung gestellt werden ausschließlich von Personen ausgestellte Zertifikate.

12

Kryptografische Protokolle und ihre Anwendung

12.1 SSL/TLS

Einsatz von SSL

Das ursprünglich von Netscape entwickelte **Secure-Sockets-Layer-Protokoll (SSL)** wurde 1999 als Transport Layer Security Protocol Version 1.0 in RFC 2246 (ersetzt durch RFC 4346) veröffentlicht. SSL wurde von Netscape zuerst in deren WWW-Browsern implementiert, um verschlüsselte und authentifizierte Verbindungen zwischen dem Client und dem Webserver zu gestatten.

Die Funktionsweisen beider Protokolle ähneln sich, sodass die Bezeichnungen TLS und SSL oft verwendet werden, um das Gleiche zu bezeichnen. Dienste, die das SSL/TLS-Protokoll nutzen, sind u. a. https, OpenVPN, GnuTLS, Bitlocker, Open Stack, sftp, ftps und scp. Aktuelle HTTPS-Dienste nutzen **Extended Validation TLS**, das eine erweiterte Überprüfung ermöglicht.

SSL/TLS ist im OSI-Modell oberhalb der Netzwerkschicht auf der Transportschicht angesiedelt, was einen Einsatz erlaubt, ohne dass die bestehende Netzwerkkommunikation auf den unteren Schichten beeinflusst wird. Obwohl SSL/TLS inzwischen durch seine Akzeptanz nicht mehr nur im World Wide Web, sondern auch bei anderen Anwendungen wie Secure copy verfügbar ist, liegt der Schwerpunkt seines Einsatzes bei der Sicherung von HTTP-Verbindungen.

SSL/TLS baut zwischen Client und Server eine gesicherte Verbindung auf, die für Datenübertragung genutzt werden kann. Dabei wird diese Verbindung auch durch einen Hash-Wert authentifiziert. Hier finden die Hash-Algorithmen MD5, SHA1, SHA2 und SHA3 Anwendung, wobei SHA3 die sicherste Alternative ist.

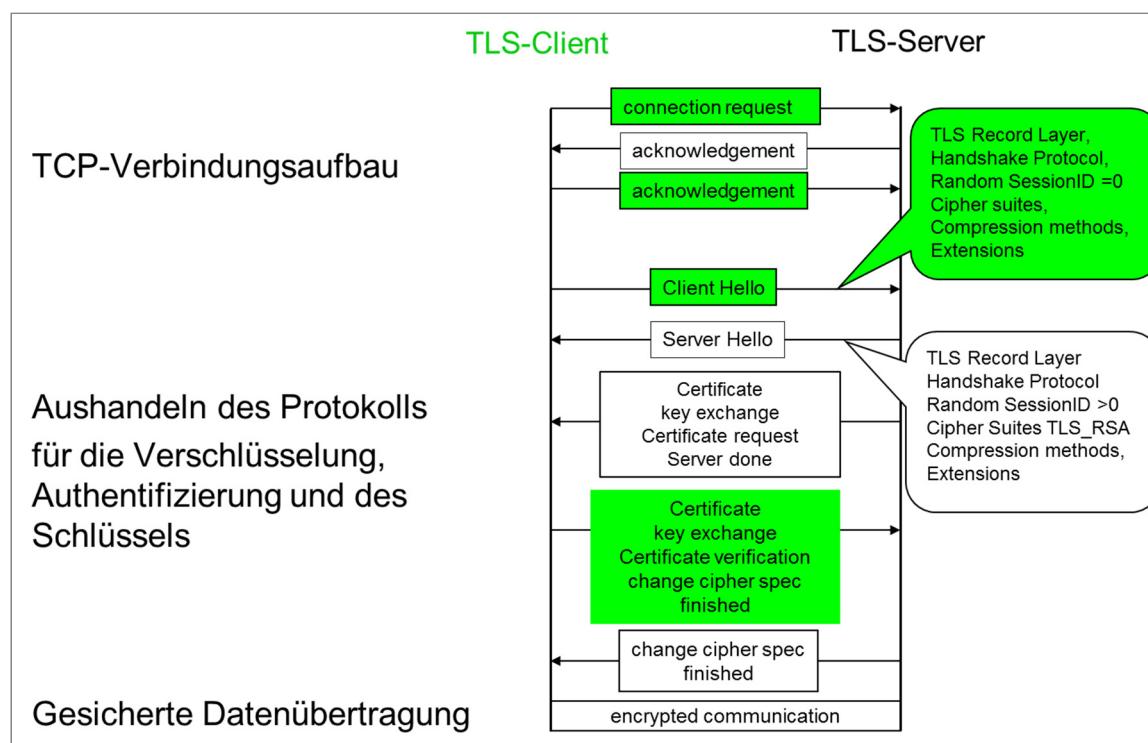
Ablauf einer TLS-Verbindung

Eine TLS-Sitzung wird mit dem Verbindungsaufbau durch den Client initiiert. Der Client teilt dem Server hier seinen TLS-Verbindungswunsch mit und gibt die höchste TLS-Versionsnummer und alle Verschlüsselungsverfahren an, die er unterstützen kann. Optional kann der Client hier auch ein Kompressionsverfahren angeben, das er unterstützt.

Der Server überprüft die empfangenen Daten und antwortet mit der höchsten gemeinsamen SSL-Versionsnummer und dem zu nutzenden Verschlüsselungsalgorithmus, der von Beiden unterstützt wird. Anschließend sendet er seinen öffentlichen Schlüssel an den Client.

Vertraut der Client dem öffentlichen Schlüssel bzw. den damit verknüpften Signaturen der CAs, so kann er einen Sitzungsschlüssel generieren und diesen mithilfe des erhaltenen Server-Public-Keys verschlüsselt an den Server senden.

Damit ist der Verbindungsaufbau vollendet. Zusätzlich besteht aber für den Server noch die Möglichkeit, auch den Client zu überprüfen. In diesem Fall muss auch der Client im Besitz eines Public/Private-Schlüsselpaares sein. Fordert der Server die Authentifizierung des Clients, so kann der Client seinen öffentlichen Schlüssel übertragen und eine vom Server zur Überprüfung gesendete zufällige Zeichenkette korrekt mit seinem privaten Schlüssel signieren (Challenge-Response-Verfahren).

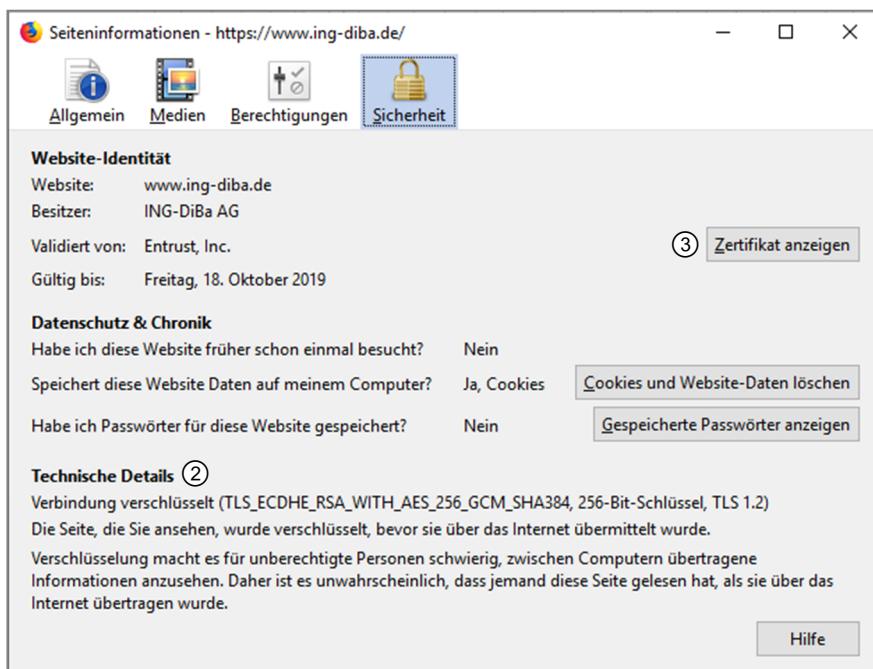
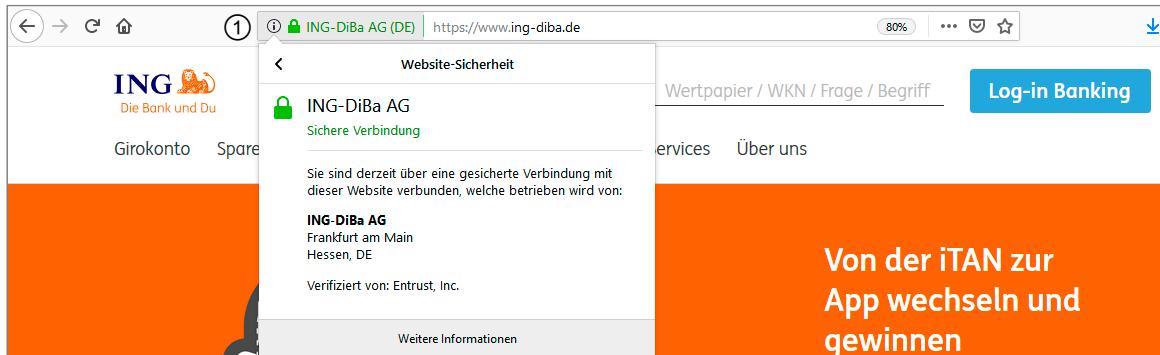


Ablauf einer TLS-Verbindung

TLS aus Client-Sicht

Früher waren die verwendeten Verschlüsselungsalgorithmen für den Datentransfer aufgrund amerikanischer Krypto-Exportregularien auf 40 Bit Länge beschränkt. Heute unterstützen alle Browser eine Schlüssellänge von mindestens 128 Bit und mehr.

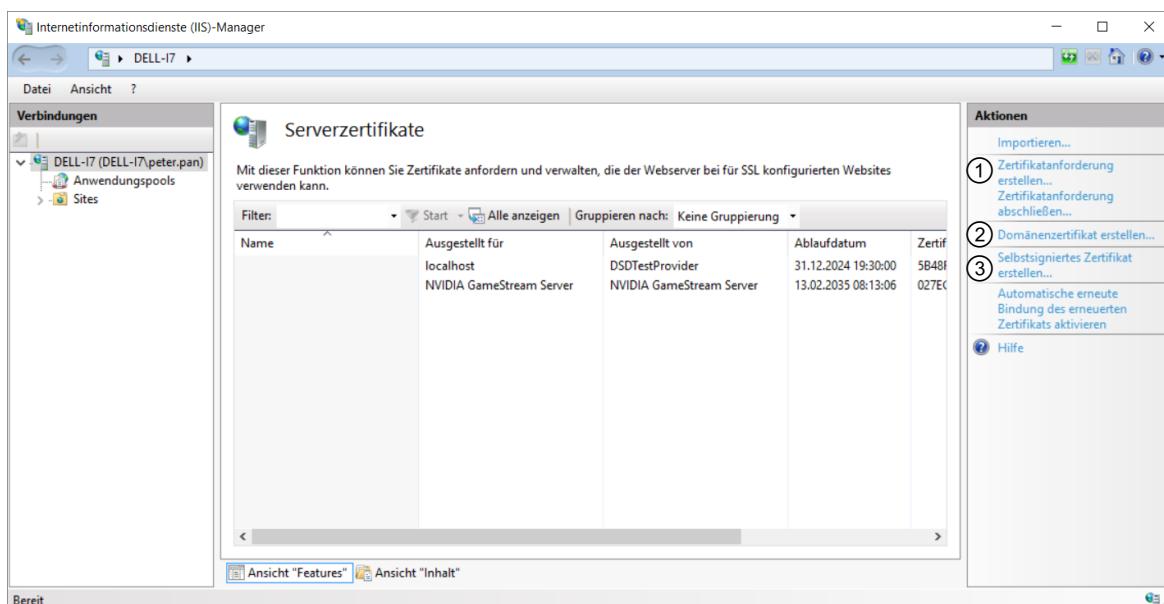
Der Browser zeigt bei aktiver TLS-Verbindung ein geschlossenes Vorhängeschloss im Browserfenster ①. Ein Klick auf dieses Schloss öffnet ein Fenster, in dem weitere Details zum aktiven Verschlüsselungsverfahren ② und zum Zertifikat des Anbieters hinterlegt sind ③.



TLS aus Server-Sicht

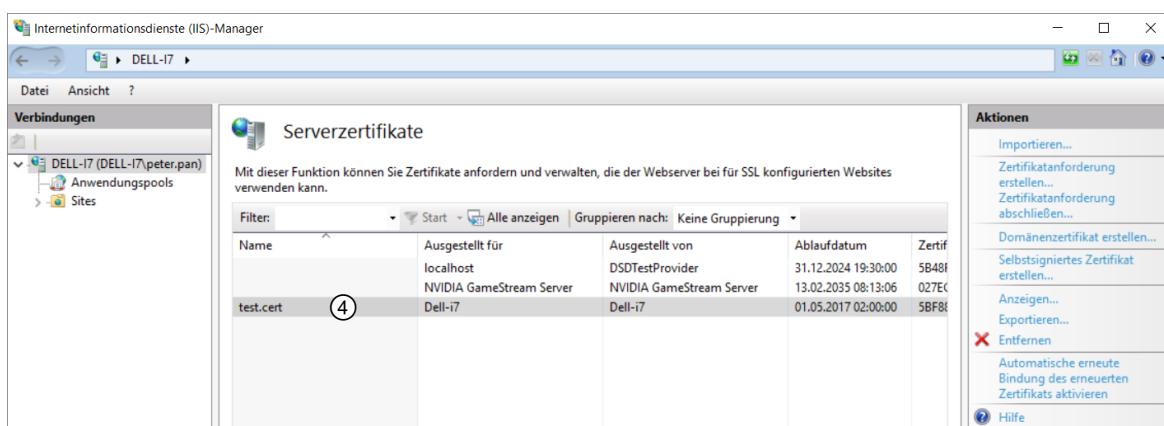
Wenn Sie Ihre Webseite dem Client über das TLS-Protokoll anbieten wollen, muss dieses Dienstmerkmal auf dem Webserver aktiviert werden. Das Prozedere von TLS wird nachfolgend an Internet Information Services Version 10 (IIS10) erläutert. SSL benötigt vor der Aktivierung ein Zertifikat, wobei Sie folgende Möglichkeiten haben:

- ✓ **Qualifiziertes Zertifikat:** ① Diese werden von einer vertrauenswürdigen Zertifizierungsstelle (Certificate Authority, kurz CA) ausgestellt. Zertifizierungsstellen (Trust Center) sind z. B. D-Trust, Telesec, Verisign oder die Bundesnetzagentur. In den Browsern sind einige Trust Center vorinstalliert, womit eigene Zertifikate dann ebenfalls vertrauenswürdig sind.
- ✓ **Domänenzertifikat:** ② Diese werden von der Domäne ausgestellt und automatisch über die Gruppenrichtlinien verteilt. Dies bildet jedoch nur das Vertrauensverhältnis innerhalb einer Domäne ab.
- ✓ **Selbst signiertes Zertifikat:** ③ Diese werden von keiner Stammzertifizierungsstelle erstellt, sondern selbst signiert und sind demzufolge nicht vertrauenswürdig. Sie sind z. B. für Testumgebungen geeignet, in denen man mit Zertifikaten arbeiten möchte, ohne eine Domänen- oder vertrauenswürdige Zertifizierungsstelle zu nutzen.



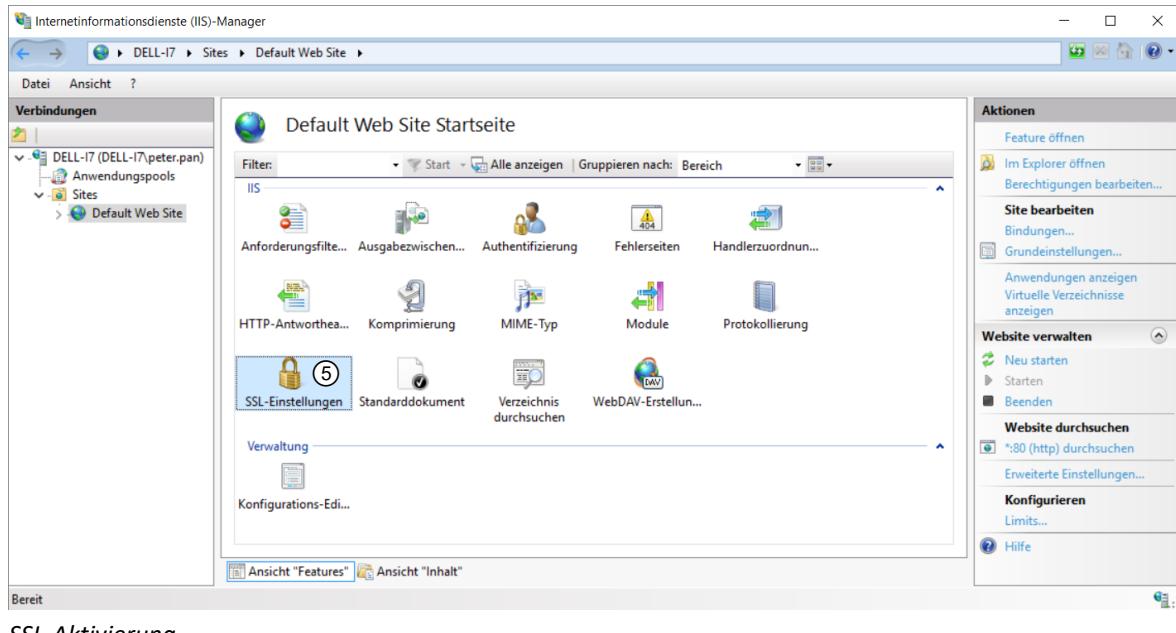
Zertifikatsaktivierung mittels IIS-Manager unter Windows 10

Nach der Wahl der Zertifizierungsart erfolgt die Zertifikaterstellung. Für Testzwecke wurde ein selbst signiertes Zertifikat mit dem Namen *test.cert* für das Webhosting generiert ④.



Zertifikaterstellung

Nach der erfolgten Sitebindung wird über *SSL-Einstellungen* ⑤ das Protokoll aktiviert und die Festlegung getroffen, in welchem Umfang Clientzertifikate akzeptiert werden.



SSL-Aktivierung

! Bedenken Sie, dass Sie mit dem Einsatz von SSL/TLS auf einem Webserver nur die Kommunikation mit den Clients schützen und signieren. SSL/TLS hat keine Schutzwirkung vor Hackerangriffen auf Ihren Webserver.

TLS 1.3 (RFC 8446)

Erkannte Schwachstellen der SSL/TLS-Protokolle der letzten Jahre haben dazu geführt, dass Sicherheitsexperten vor einiger Zeit empfohlen, alle Versionen von SSL und TLS 1.0 zu deaktivieren. Dies wurde ausgelöst von der PCI-Compliance, die von allen Kreditkartenanbietern unterstützt wird.

Ab Juni 2018 müssen alle Webseiten über die Transaktionen vollzogen werden mindestens das Protokoll TLS 1.1 oder höher unterstützen, um dem Sicherheitsstandard für Kreditkarten (PCI, Data Security Standard) zu entsprechen.

TLS 1.3 unterstützt moderne Krypto-Algorithmen für die Transportverschlüsselung. Bisherige, als problematisch oder eindeutig angreifbar geltende Algorithmen wie SHA-1, RC4 und die Export-Cipher dürfen hier nicht mehr genutzt werden. Das unterbindet Downgrade-Angriffe (u. a. Logjam oder FREAK) auf diese Algorithmen. Zusätzlich findet eine bessere Integritäts-sicherung der verschlüsselten Daten statt. Auch sind weniger im Klartext vorhandene Metadaten über den Inhalt der TLS-Sitzung einsehbar. Ein weiterer Aspekt ist die höhere Performance gegenüber TLS 1.1.

Inwieweit ein/Ihr Webserver sicher ist und die aktuellen Standards vollständig unterstützt, können Sie unter <https://globalsign.ssllabs.com/> verifizieren.

12.2 SSH

Grundlagen zu Secure Shell

Secure Shell ist ein kryptografisches Protokoll und arbeitet auf der Anwendungsebene. Es wurde für die UNIX-Welt entwickelt und dient primär für den Ersatz der Remote-Utilities *rlogin*, *rcp* und *rsh*.

SSH ermöglicht eine Verbindung (login) zu Systemen im Netzwerk (Intranet/Internet). Innerhalb der SSH-Session können Sie dann auf dem Command Line Interface (CLI) Befehle ausführen. Obwohl ursprünglich für die Unix-Welt konzipiert, haben SSH-Implementierungen in allen Betriebssystemen Einzug gehalten, da es eine entfernte Kopplung von Rechnern und deren sicheren Datentransfer garantiert. Unter Windows existiert auf der GUI (Graphical User Interface) mit der Remotedesktopverbindung eine Alternative.

Funktion von SSH

Bei SSH existieren die Protokollstandards 1.x und 2.x, wobei diese miteinander nicht kompatibel sind. Weil SSH v1.x nicht vollständig standardisiert wurde und einige sicherheitstechnische Mängel erkannt wurden, sollte der Version 2.x immer der Vorzug gegeben werden. SSH v2.x wird u. a. in den RFCs 4251–4256 und 6668 beschrieben.

Bei SSH baut der Client zuerst eine Verbindung zum Server auf. Danach tauschen beide Informationen zum eingestellten SSH-Protokoll und den Verschlüsselungsparametern aus ①. Mittels Diffie-Hellman ② wird ein zuvor auf dem Server erzeugter Schlüssel ausgetauscht.

Source	Destination	Protocol	Info
fd1::1	fd2::3c2e:1fb...	SSHv2	Server: Protocol (SSH-2.0-Mocana SSH 5.3.1) ①
fd2::3c2e:1fb1... fd1::1	fd1::1	SSHv2	Client: Protocol (SSH-2.0-PuTTY_Release_0.62)
fd2::3c2e:1fb1... fd1::1	fd1::1	SSHv2	Client: Key Exchange Init
fd1::1	fd2::3c2e:1fb...	SSHv2	Server: Key Exchange Init
fd2::3c2e:1fb1... fd1::1	fd1::1	SSHv2	Client: Diffie-Hellman Key Exchange Init ②
fd1::1	fd2::3c2e:1fb...	SSHv2	Server: Diffie-Hellman Key Exchange Reply
fd1::1	fd2::3c2e:1fb...	SSHv2	Server: New Keys
fd2::3c2e:1fb1... fd1::1	fd1::1	SSHv2	Client: New Keys
fd2::3c2e:1fb1... fd1::1	fd1::1	SSHv2	Client: Encrypted packet (len=52) ③
fd1::1	fd2::3c2e:1fb...	SSHv2	Server: Encrypted packet (len=52)
fd1::1	fd2::3c2e:1fb...	SSHv2	Server: Encrypted packet (len=244)
fd2::3c2e:1fb1... fd1::1	fd1::1	SSHv2	Client: Encrypted packet (len=68)

SSH-Verbindung

Der Schlüssel setzt sich aus einem **Public Key** und einem **Private Key** zusammen, wobei nur der Public Key zum Verschlüsseln übertragen wird. Zur Sicherheit wird der Public Key mit einem Hash-Wert zur Integritätssicherung versehen. Dieser Fingerprint muss vom Client akzeptiert werden, damit eine sichere Kommunikation stattfinden kann ③.

SSH ist relativ flexibel und lässt so die Wahl zwischen verschiedenen Verschlüsselungs- und Authentifizierungsverfahren zu. Beispielsweise hat der Benutzer die Wahl, sich per Passwort, Kerberos oder Public Key gegenüber dem Server auszuweisen.

Nach einer erfolgreichen Anmeldung kann der Benutzer auf seine Kommando-Shell zugreifen und wie bei einem Telnet-Log-in sämtliche Operationen am Remote-Rechner durchführen. Zusätzlich zur gesicherten Shell ist auch die Datenübertragung per SCP (Secure Copy Protocol) oder SFTP (SSH File Transfer Protocol) geschützt. Das gleichlautende UNIX-Programm *scp* wird quasi wie *cp* bedient und SFTP funktioniert ähnlich wie FTP, allerdings verschlüsselt.

SSH Port Forwarding

Markant bei SSH ist die Möglichkeit, eine geschützte Verbindung zu einem Remote Server aufzubauen und dann durch diese Verbindung einen Port-Forwarding-Tunnel zu bauen. Auf diese Weise kann nicht SSH-fähige Netzwerksoftware verwendet werden, indem die SSH-Umgebung als Proxy-Server verwendet wird und die Verschlüsselungs- und Sicherungsaufgaben übernimmt.

SSH ist auch in bestimmten grafisch basierten Anwendungsprogrammen unter Windows implementiert, sodass der Benutzer hier auf die übliche komfortable Benutzeroberfläche zugreifen kann.

Nachfolgend einige SSH-Clients, die Sie nutzen sollten:

- ✓ PuTTY, <https://www.putty.org/>
- ✓ KiTTY, <http://www.9bis.net/kitty/?page=Download>
- ✓ MobaXterm, <https://mobaxterm.mobatek.net/>
- ✓ SmarTTY, <http://smatty.sysprogs.com/>
- ✓ Solar-PuTTY, <https://www.solarwinds.com/free-tools/solar-putty>
- ✓ SSH-Client unter Windows 10, <https://www.windows-faq.de/2018/04/21/ssh-client-unter-windows-10-installieren/>

12.3 IPsec

Sicherheit für das „alte“ und das „neue“ Internet

Als Nachfolger des aktuellen Internet-Protokolls IPv4 ist IPv6 schon seit längerem im Einsatz. IPv6 erweitert nicht nur den Raum an verfügbaren Adressen von 32 Bit auf 128 Bit, sondern hat erstmals Sicherheitsfunktionen (IPsec) im IPv6-Protokoll selbst verankert. Da heute der überwiegende Teil der Netzwerkkomponenten aber immer noch mit IPv4 betrieben wird, wurden die Sicherheitskonzepte mit **IPsec** (IP Security) auch für die Version 4 standardisiert. Es ist für die darüber liegenden Applikationen vollständig protokolltransparent, da bei ihnen keinerlei Modifikationen bezüglich der Sicherheitsaspekte vorgenommen werden müssen.

IPsec soll eine sichere Verbindung auf der IP-Ebene ermöglichen. Hierzu sind entsprechende Verfahren zur Verschlüsselung und Integrität der Datenpakete auf Layer 3 (Network Layer) implementiert: Die IPsec-Protokollsuite nutzt hierbei zwei Mechanismen (vgl. Abschnitt 16.3).

- ✓ **ESP** (Encapsulating Security Payload) wird für die Vertraulichkeit der Daten eingesetzt. ESP nutzt einen Sitzungsschlüssel für die Verschlüsselung. Zusätzlich wird die Integrität über einen Hash-Wert signiert
- ✓ **AH** (Authentication Header) findet Einsatz, wenn nur die Datenintegrität und keine Vertraulichkeit erforderlich ist. AH signiert das Paket mit einem SHA-Hashwert. Der Empfänger erstellt auf gleiche Weise eine Signatur. Sofern beide Signaturen übereinstimmen, ist die Integrität des Paketes gewährleistet.

Das Verschlüsselungsmanagement übernimmt dabei (sofern kein manueller Schlüsselaustausch vorgenommen wird) **IKE** (Internet Key Exchange). IKE ist ein Teil von **ISAKMP** (Internet Security Association and Key Management Protocol).

In der ersten Phase (Schlüsselaustausch) werden die Verschlüsselungsparameter ① und die Authentisierung ② zwischen den VPN-Partnern ausgetauscht. Die Authentifizierung kann mit Zertifikaten oder über Pre-shared Keys (Passwörter) erfolgen.

Diese Verbindung wird genutzt, um die Parameter der zweiten Phase (Datenverschlüsselung) sicher zwischen den IPsec-Partnern aushandeln zu können. IKE Phase 2 handelt die Verschlüsselungs- und Integritätsparameter aus, mit denen die eigentlichen Daten gesichert werden. Nach Aushandlung der SAs (Security Association) in IKE Phase 2 wird das Protokoll ESP benutzt, um letztendlich die verschlüsselten Daten ③ zu transportieren.

Source	Destination	Protocol	Info	
172.16.6.73	149.154.153.237	ISAKMP	Aggressive	①
149.154.153.237	172.16.6.73	ISAKMP	Aggressive	
172.16.6.73	149.154.153.237	ISAKMP	Aggressive	
149.154.153.237	172.16.6.73	ISAKMP	Transaction (Config Mode)	
172.16.6.73	149.154.153.237	ISAKMP	Transaction (Config Mode)	
149.154.153.237	172.16.6.73	ISAKMP	Transaction (Config Mode)	②
172.16.6.73	149.154.153.237	ISAKMP	Transaction (Config Mode)	
172.16.6.73	149.154.153.237	ISAKMP	Transaction (Config Mode)	
149.154.153.237	172.16.6.73	ISAKMP	Transaction (Config Mode)	
172.16.6.73	149.154.153.237	ISAKMP	Quick Mode	
149.154.153.237	172.16.6.73	ISAKMP	Quick Mode	
172.16.6.73	149.154.153.237	ISAKMP	Quick Mode	
172.16.6.73	149.154.153.237	ESP	ESP (SPI=0xc0795a77)	
172.16.6.73	149.154.153.237	ISAKMP	Informational	
149.154.153.237	172.16.6.73	ISAKMP	Informational	
172.16.6.73	149.154.153.237	ESP	ESP (SPI=0xc0795a77)	③
172.16.6.73	149.154.153.237	ESP	ESP (SPI=0xc0795a77)	

IPsec-Verbindung

Wegen seiner flexiblen Einsetzbarkeit auf IP/IPv6-Ebene hat IPsec nicht nur zur Sicherung normaler Netzwerkkommunikation starkes Interesse gefunden, es eignet sich auch exzellent zum Bau von **Virtual Private Networks** (VPN), vgl. dazu auch Kap. 16.

12.4 Übung

Fragen zu kryptografischen Protokollen

Übungsdatei: --

Ergebnisdatei: uebung12.pdf

1. Welche Verschlüsselungsprotokolle finden im Netzwerkbereich hauptsächlich Anwendung?
2. Welche Zertifikate finden bei den Verschlüsselungsprotokollen Anwendung?

13

Sichere E-Mail-Verfahren

13.1 Grundlagen der E-Mail-Verschlüsselung

E-Mail-Korrespondenz wird im Internet im Normalfall **unverschlüsselt** übertragen. Somit kann jede Nachricht von interessierten Stellen mitgelesen, ausgewertet und mitunter manipuliert werden. Um dem vorzubeugen, sollten vertrauliche und sensible Nachrichten über eine E-Mail-Verschlüsselung gesichert zwischen Absender und Empfänger ausgetauscht werden. Diesem Problem haben sich auch die Serviceprovider angenommen und bieten umfänglich E-Mail-Verschlüsselung auf der Basis der SSL/TLS-Protokolle an. Leider ist das aber vorrangig eine **Transportverschlüsselung** (d. h. nur eine gesicherte Verbindung zwischen dem Nutzer und seinem Provider) und keine **End-to-End-Verschlüsselung** (d. h. eine gesicherte Verbindung zwischen den Nutzern). In einer Umfrage (03/2018) von WEB.DE und GMX gaben nur 14 % der E-Mail-Nutzer an, ihre Daten zu verschlüsseln.

Neben der Verschlüsselung (Vertraulichkeit) muss auch die Integrität (Authentizität) der Nachricht gewährleistet sein. Beide Verfahren werden fast immer miteinander kombiniert. Die Verschlüsselung wird durch symmetrische oder asymmetrische Verfahren realisiert. Als Verschlüsselungs- und Signierungs-Software werden Client-basierende oder Server-basierende Lösungen angewendet. Häufig kommt hierbei die PKI (Public-Key-Infrastruktur) zum Einsatz. Mit PKI wird in der Kryptologie ein System klassifiziert, das digitale Zertifikate für die Authentifizierung und Verschlüsselung ausstellen, verteilen und überprüfen kann. Dazu kommen hauptsächlich folgende Normen zur Anwendung:

- ✓ S/MIME (Secure / Multipurpose Internet Mail Extensions) – RFC 5751
- ✓ OpenPGP (Open Pretty Good Privacy) – RFC 4880

S/MIME

Bei S/MIME stehen Ihnen zwei Formate (Content Types) zur Verfügung: für die Verschlüsselung das Multipart/Encrypted-Format und für die Signierung das Multipart/Signed-Format. Dieses Protokoll wird von allen aktuellen Mailprogrammen unterstützt und basiert auf einem X.509-Zertifikat, das lokal, entfernt oder signiert bereitgestellt wird. Für die Vertrauenswürdigkeit des X.509-Zertifikates müssen Sie ein Wurzelzertifikat definieren (auch Root-CA genannt). Es dient dazu, die Gültigkeit aller untergeordneten Zertifikate zu überprüfen. Die aktuelle Version X.509v3 ist im Standard RFC 5280 beschrieben.

Das X.509-Zertifikat wird nach dem SigG (Signaturgesetz) in drei Klassen unterteilt:

- ✓ **Fortgeschrittene Signatur:** Das komplette Schlüsselmanagement liegt in der alleinigen Verantwortung und Kontrolle des Inhabers.
- ✓ **Qualifizierte Signatur:** Sie beruht auf einem nachgewiesenen Zertifikat einer sicheren Signaturstelle und ist einer handschriftlichen Unterschrift gleichgestellt.
- ✓ **Qualifizierte Signatur mit Anbieterakkreditierung:** Diese ist die **sicherste** Zertifizierungsstufe (Akkreditierung gem. § 15 SigG) für Verschlüsselung und Signierung im Internet.

OpenPGP (PGP/MIME)

Die Urversion von Pretty Good Privacy wurde 1991 von Phil Zimmermann als Freeware veröffentlicht und danach mehrfach modifiziert. Neben der kommerziellen Version, die von der Firma Symantec vertrieben wird (siehe auch <http://www.symantec.com/products-solutions/families/?fid=encryption>), existieren einige Open-Source-Varianten (siehe u. a. www.pgpi.org, www.gnupg.org, www.gpg4win.de).

S/MIME und OpenPGP sind, obwohl sie teilweise die gleichen Verschlüsselungsverfahren einsetzen, nicht miteinander kompatibel.

GnuPG

Der Gnu Privacy Guard ist eine OpenPGP-kompatible Anwendung, basierend auf einer Modifikation des originalen PGP. GnuPG ist jedoch vollständig Open Source, steht unter GPL-Lizenz und ist lizenzzfrei sowohl für private als auch für kommerzielle Anwender einsetzbar. Das Bundesamt für Sicherheit in der Informationstechnik (BSI, www.bsi.de) stellte im Rahmen des sogenannten GnuPP (Gnu-Privacy-Projekt) Fördermittel für die Entwicklung von GnuPG und einiger zugehöriger Tools zur Verfügung.

Auch das Projekt Gpg4win (www.gpg4win.de) hat es sich zur Aufgabe gemacht, mehrere Open-Source-Tools zur Integration von GnuPG-Funktionen in einer Windows-Umgebung bereitzustellen, verbunden mit einem einfachen Installationsvorgang.

GnuPG bietet mit der grafischen Benutzeroberfläche und den vorhandenen Tools eine einfache und komfortable Möglichkeit zum Verschlüsseln und Signieren unter den Betriebssystemen Microsoft Windows XP – Windows 10 an. So kann zum Beispiel in Kombination mit entsprechenden Plug-ins für die jeweils verwendete 32-Bit-E-Mail-Version (Microsoft Outlook 2007/2010/2013/2016, MS Exchange Server, Claws-Mail) und einer Desktop-Integration mithilfe von Tools wie Kleopatra ein Bedienkomfort ähnlich dem des kommerziellen PGP erreicht werden. Als Verschlüsselungsalgorithmen finden bei GnuPG ElGamal, 3DES, AES, DSA, RSA, Blowfish, Twofish, CAST5, MD5, SHA-1, RIPE-MD-160 und TIGER Anwendung.

Nachfolgend wird am Beispiel vom Gpg4win exemplarisch die Funktion der Zertifikaterstellung und -nutzung beschrieben. Im Gegensatz zu GnuPG bietet Gpg4win eine einfachere Installationsroutine, weitere nützliche Tools und Utilities zur E-Mail- und Datei-Verschlüsselung.

GnuPG for Windows (Gpg4win) installieren

Haben Sie das Windows-Installationspaket Gpg4win von www.gpg4win.de heruntergeladen, so müssen Sie bei der Installation angeben, welche Sprache bei der Installation benutzt werden soll und an welchem Ort die Dateien installiert bzw. Verknüpfungen gelegt werden sollen.

Der Gpg4win-Installer entpackt anschließend:

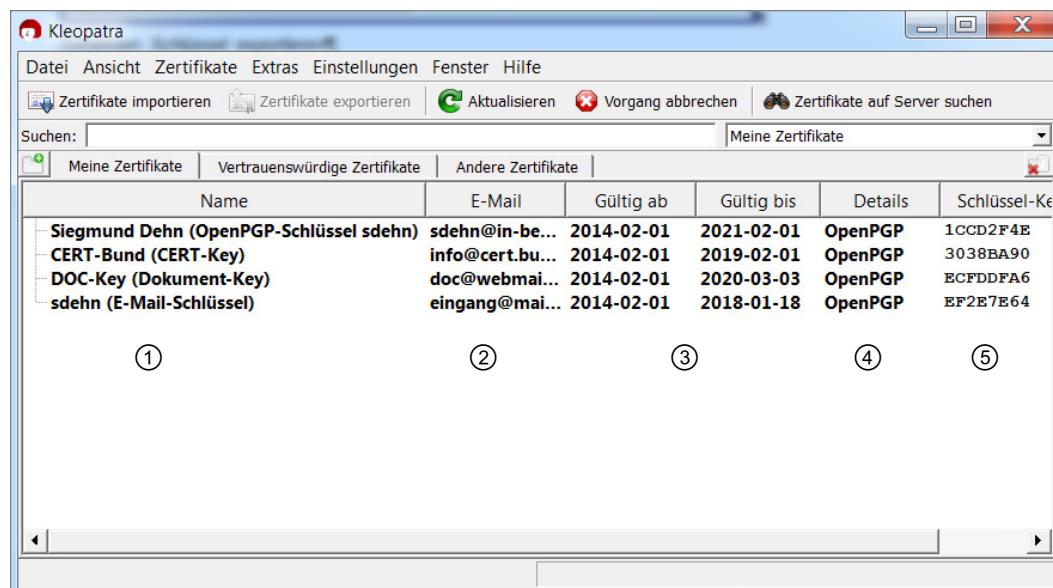
- ✓ GnuPG (GNU Privacy Guard; Kernkomponente, welche die Verwaltung der Schlüssel bereitstellt)
- ✓ Kleopatra (Zertifikatsmanager zur Erzeugung der Schlüssel)
- ✓ GPA (GNU Privacy Assistant; optional; dient der alternativen Verwaltung von Zertifikaten)
- ✓ GpgOL (GPG Outlook; Add-Ins für Microsoft Outlook ab Version 2003 zum Verschlüsseln von Nachrichten; benötigt Kleopatra oder GPA)
- ✓ GpgEX (GPG Shell Extensions; Explorer-Kontextmenüerweiterung zum Verschlüsseln; benötigt Kleopatra oder GPA)
- ✓ Claws-Mail (optional; vollständiger E-Mail-/News-Client mit kompletter Unterstützung für GnuPG)
- ✓ Gpg4win-Kompendium (Gpg4win-Dokumentation in deutscher und englischer Sprache)

Diese Tools haben die Aufgabe, das klassische kommandozeilenorientierte GPG in die grafische Benutzeroberfläche von Windows zu integrieren.

13.2 Schlüssel generieren

Schlüssel mit Kleopatra generieren

Bevor Sie GnuPG einsetzen können, müssen Sie ein Schlüsselpaar generieren. Wenn Sie das erste Mal das Dienstprogramm Kleopatra starten und noch nicht über Schlüssel verfügen, können Sie von hier aus gleich ein passendes Schlüsselpaar erzeugen lassen.



Schlüsselverwaltung in Kleopatra

Wenn Sie das mit GnuPG installierte Verwaltungsprogramm Kleopatra starten, gibt Ihnen das Hauptfenster Auskunft über alle Ihnen bekannten Schlüssel. Im Detail sehen Sie folgende Informationen:

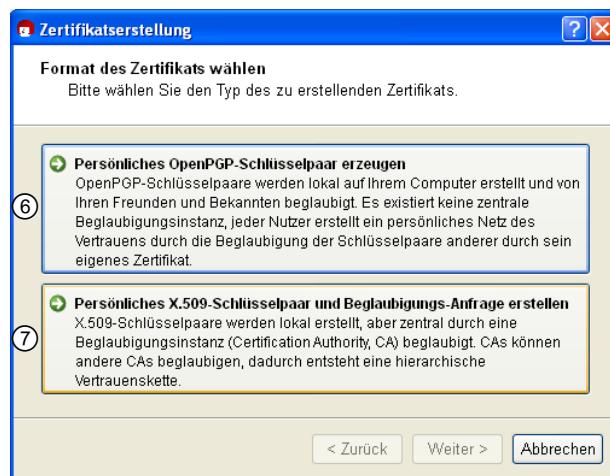
- ✓ den Namen des Schlüsselbesitzers ①,
- ✓ die E-Mail des Nutzers ②,
- ✓ den Beginn und Ablauf der Gültigkeit des Schlüssels ③,
- ✓ das Zertifikatsformat ④,
- ✓ die Kennung des Zertifikates ⑤.

Über den Menüpunkt *Datei - Neues Zertifikat* können Sie das Erzeugen eines neuen Schlüsselpaares anstoßen.

Zuerst müssen Sie festlegen, nach welchem Verschlüsselungsstandard Ihr Zertifikat generiert werden soll. Sie haben die Wahl zwischen PGP/ MIME (Open PGP) ⑥ und S/MIME (X.509) ⑦.

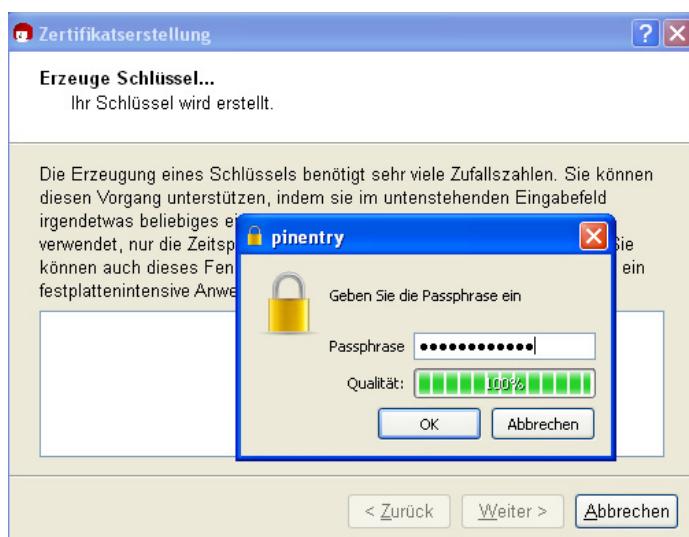
In den weiteren Schritten müssen Sie den Verschlüsselungs-Algorithmus (RSA oder DSA), die Schlüsselstärke und die Gültigkeit des Schlüssels festlegen. Neben der Gültigkeit müssen Sie die Verwendungsart des Schlüssels angeben. Miteinander kombinierbare Verwendungsarten sind:

- ✓ Signieren
- ✓ Verschlüsseln
- ✓ Beglaubigen
- ✓ Authentifizieren



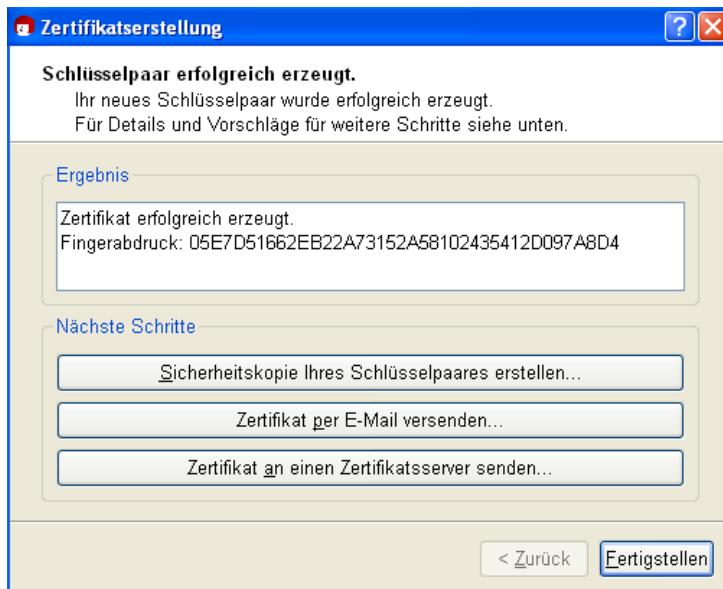
Schlüssel mit Kleopatra erzeugen

Nachdem Sie eine Passphrase für Ihren Schlüssel ausgewählt haben, wird GnuPG nun Ihre Schlüssel generieren. Je nachdem, welche Länge Sie gewählt haben, werden die Erzeugung der passenden Primzahlen und die Berechnung der nötigen Daten einige Zeit dauern.



Passphrase für den Schlüssel

Nach der Erzeugung eines Zertifikates wird Ihnen der Fingerprint des Schlüssels bereitgestellt. Diesen können Sie nun zum Zertifikatsserver senden, um ihn öffentlich zu machen. Mit der Veröffentlichung wird das Zertifikat für jeden ersichtlich. Alternativ haben Sie die Möglichkeit, das Schlüsselpaar zu sichern bzw. zur Verschlüsselung an einen Empfänger weiterzuleiten.



Erstelltes Zertifikat

13.3 Schlüsselexport und -import

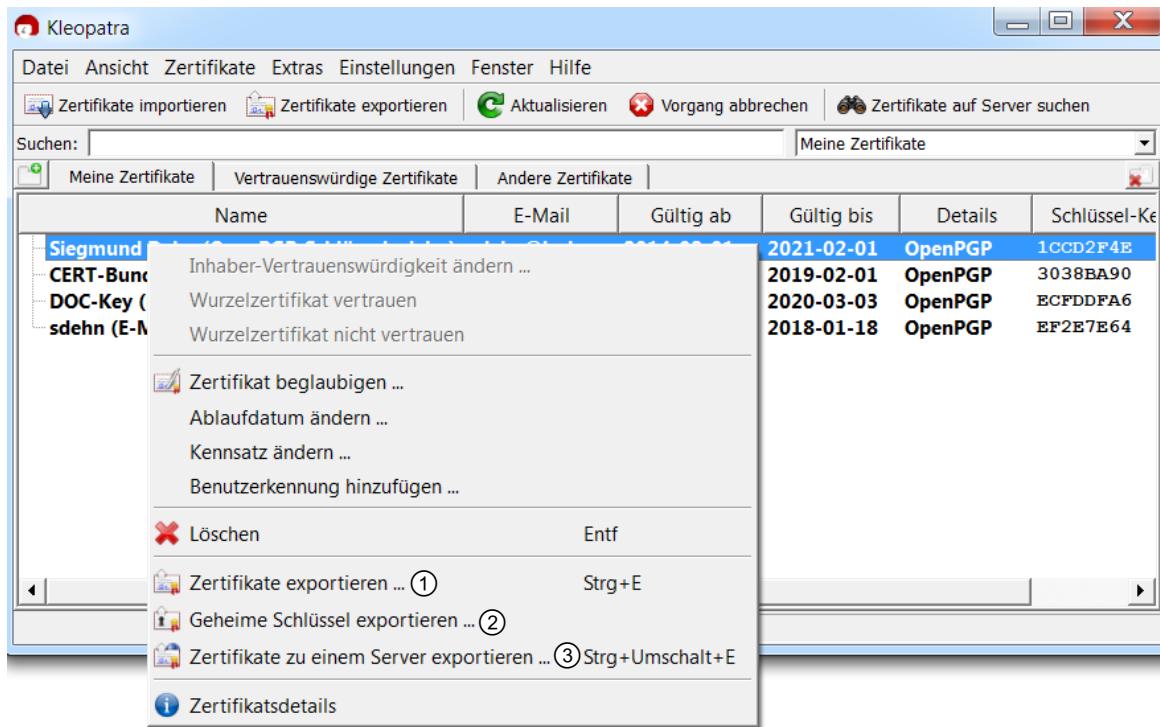
Warum exportieren?

Damit andere Teilnehmer Ihnen verschlüsselte Nachrichten oder Dateien senden können oder in der Lage sind, Ihre digitale Signatur zu überprüfen, müssen Sie Ihren öffentlichen Schlüssel zur Verfügung stellen.

Mit der Exportfunktion können Sie den öffentlichen Teil Ihres Schlüssels entweder als Datei exportieren, die Sie versenden können, oder aber den Schlüssel sofort zu einem Public-Key-Server im Internet hochladen, wo er zum Download für andere bereitsteht.

Export unter Kleopatra

- ▶ Rufen Sie Kleopatra auf und markieren Sie mit der linken Maustaste den öffentlichen Schlüssel, den Sie exportieren möchten.
- ✓ Mit der Option *Zertifikate exportieren* ① wird Ihr öffentliches Zertifikat mit der Erweiterung .asc lokal abgespeichert.
- ✓ Der private Schlüssel wird lokal über den Menüpunkt *Geheime Schlüssel exportieren* ② gesichert.
- ✓ Für die Veröffentlichung Ihres öffentlichen Schlüssels nutzen Sie *Zertifikate zu einem Server exportieren* ③.



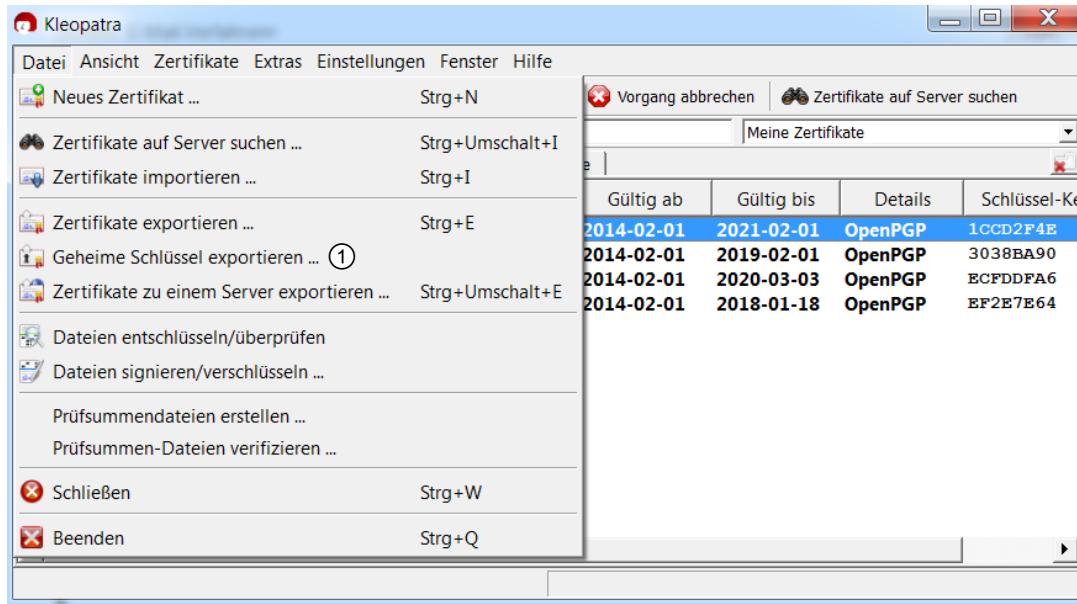
Einen Schlüssel exportieren

Die Datei können Sie per E-Mail versenden oder zum Download zur Verfügung stellen. Befindet sich der Schlüssel auf einem Key-Server, so können andere Anwender diesen über eine bequeme Such- und Downloadfunktion beziehen. Beachten Sie, dass Sie einen einmal zu einem Key-Server gesendeten Schlüssel nicht mehr löschen können. Wenn Sie einen auf einem Key-Server befindlichen Schlüssel nicht mehr benutzen wollen, so können Sie höchstens ein Rückrufzertifikat ausstellen, um den Schlüssel für ungültig zu erklären. Der Schlüssel wird dann zusammen mit den Rückrufzertifikaten auf dem Server geführt. Bedenken Sie auch, dass die mit dem Key verknüpften E-Mail-Adressen öffentlich einsehbar sind.

Export des privaten Schlüssels

Nachdem Sie Ihren Schlüssel erzeugt haben, sollten Sie von Ihrem privaten Schlüssel eine Sicherungskopie (z. B. auf einem USB-Stick) anlegen und diese an einem sicheren Ort verwahren – ansonsten können Sie bei einem Verlust dieses Schlüssels keine Nachrichten mehr entschlüsseln, die für diesen Schlüssel verschlüsselt sind.

Bei den Exportoptionen von Kleopatra können Sie hierzu über *Datei - Geheime Schlüssel exportieren* ① wählen, dass der private Schlüssel in die Exportdatei mit aufgenommen wird.

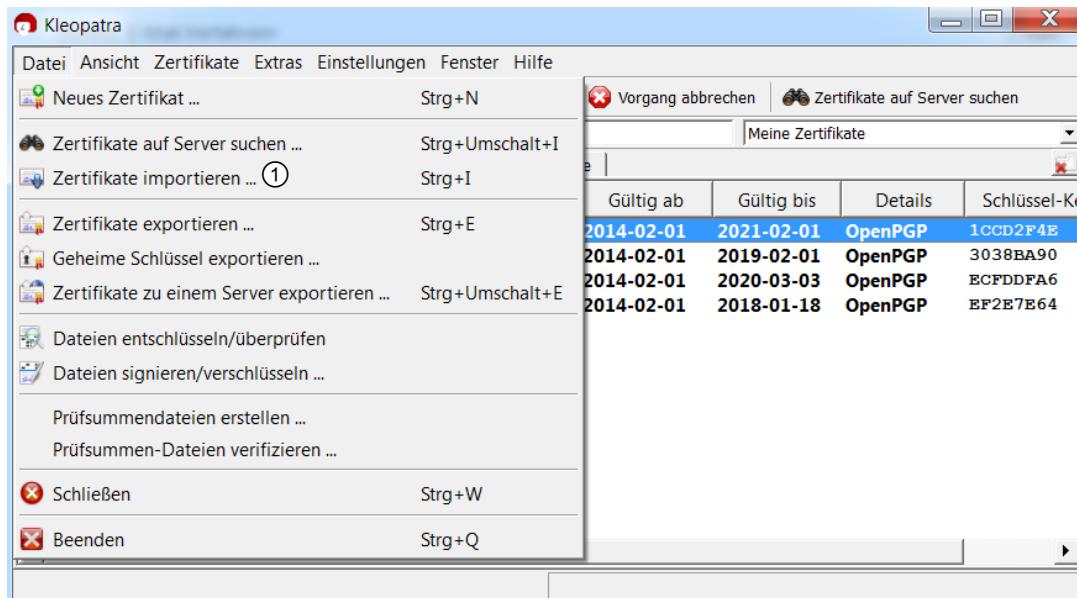


Geheimen Schlüssel exportieren

Im Falle einer Datenpanne können Sie dann Ihren privaten Schlüssel aus dieser Sicherungsdatei wieder in das System importieren.

Schlüssel in Kleopatra importieren

Um einen Schlüssel in OpenPGP zu importieren, können Sie den Menübefehl *Datei - Zertifikate importieren* ① aufrufen.



Schlüsselimport in Kleopatra

Die Zertifikate können mit den Erweiterungen

- ✓ *.asc, Textdatei, ASCII-codiert;
- ✓ *.cer, MIME-X.509-CA-Zertifikat;
- ✓ *.cert, MIME-X.509-CA-Zertifikat;
- ✓ *.crt, MIME-X.509-CA-Zertifikat;
- ✓ *.der, MIME-X.509-CA-Zertifikat;
- ✓ *.pem, Privacy-Enhanced-Mail-Zertifikat (Base64-codiert);
- ✓ *.gpg, GNU-Privacy-Guard-Zertifikat

importiert werden.

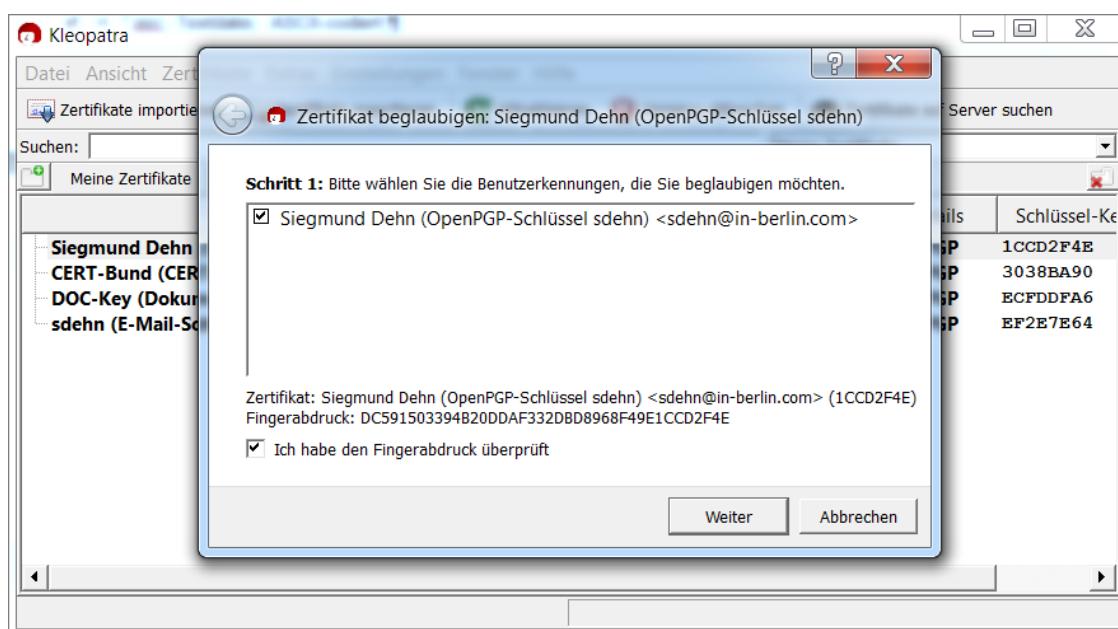
Im darauf folgenden Fenster haben Sie dann die Möglichkeit, den Ort für das Importieren des gewählten Zertifikates anzugeben.

13.4 Signieren von Schlüsseln

Warum Signaturen notwendig sind

Das Web of Trust basiert darauf, dass jeder, der sich von der Echtheit eines Schlüssels in seinem Schlüsselbund überzeugt hat, dies durch seine Signatur bestätigt und somit auch anderen die Möglichkeit gibt, anhand der geleisteten Signatur der Echtheit zu vertrauen. Damit Sie jedoch einen gerade importierten öffentlichen Schlüssel einsetzen können, müssen Sie diesen zumindest auf Ihrem eigenen Schlüsselbund signieren.

Sie können die Art der Signatur, die Sie leisten wollen, auswählen. Wenn Sie nicht wollen, dass andere sich auf Sie verlassen, wählen Sie eine nicht exportierbare Signatur. Diese verbleibt auf Ihrem Schlüsselbund. In Kleopatra können Sie dies über den Menübefehl *Zertifikate -> Zertifikate beglaubigen* für sich selbst oder für alle beglaubigen.



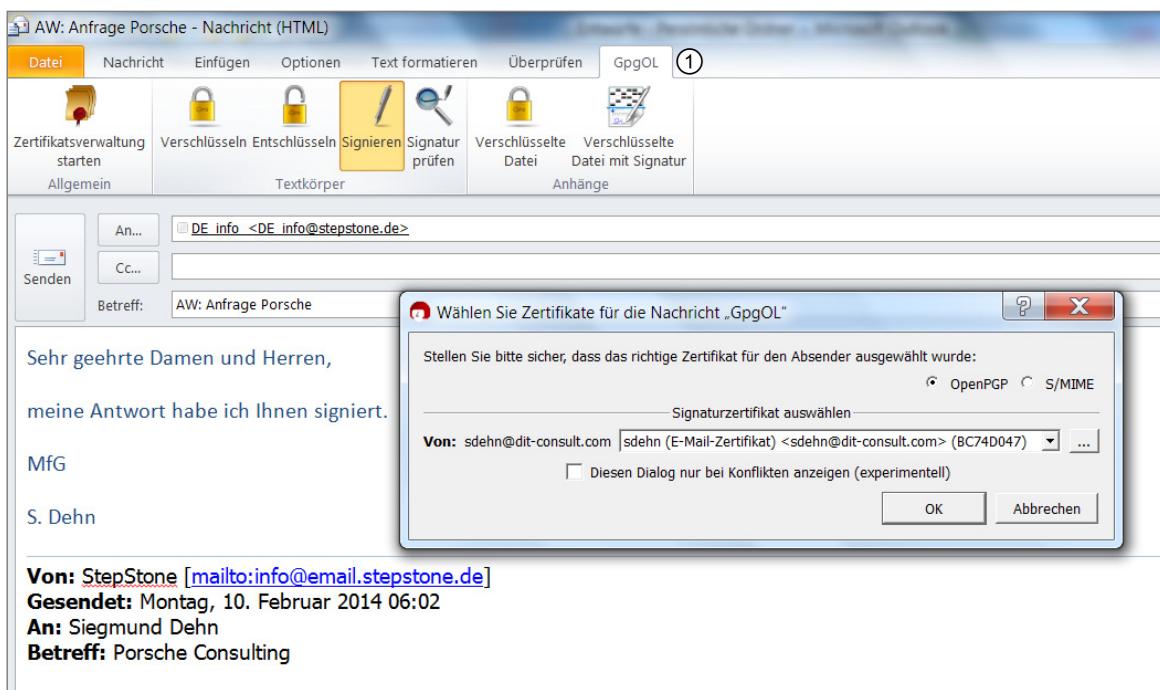
Zertifikat beglaubigen

13.5 E-Mail signieren und verschlüsseln

Arbeit mit Outlook

Nach der Installation von Gpg4win ist über die Menüleiste von Outlook das Add-In GpgOL (**GPG Outlook** für Microsoft zum Verschlüsseln/Signieren von Nachrichten) verfügbar.

Nach dem Öffnen von Outlook und dem Verfassen der Nachricht nutzen Sie über die Menüleiste den Punkt *GpgOL* ① und die Option zum *Verschlüsseln* oder *Signieren* und das erforderliche Zertifikat.



Outlook-Nachricht signieren

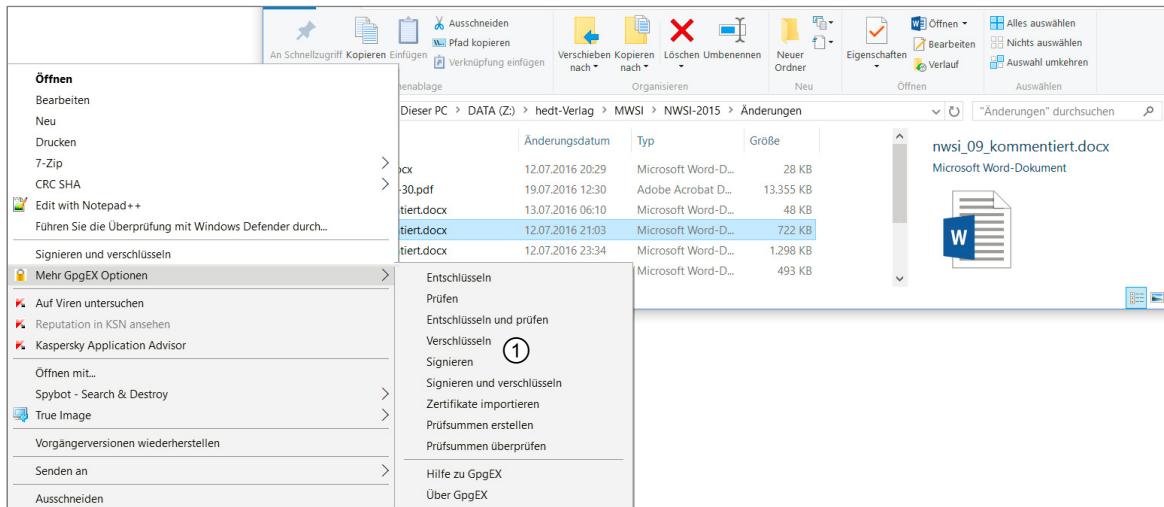
Eine empfangene verschlüsselte/signierte E-Mail können Sie mit der Passphrase Ihres privaten Schlüssels öffnen.

13.6 Dateien signieren und verschlüsseln

Weitere Optionen von Gpg4win

Nicht nur E-Mails können Sie mittels Gpg4win sicher übermitteln, auch die einfache Dateiübertragung über den Explorer ist durch das Add-In GpgEX möglich.

Über die Menüleiste des Explorers *Datei - Mehr GpgEX Optionen* stehen Ihnen auch die Optionen für Verschlüsselung/Signierung von Dateien ① zur Verfügung.



Explorer – Kontextmenü

13.7 Übung

Fragen zur E-Mail-Verschlüsselung

Übungsdatei: --

Ergebnisdatei: uebung13.pdf

1. Welche Verschlüsselungsstandards finden bei der E-Mail-Verschlüsselung Anwendung?
2. Welche Signaturarten sind beim X.509-Zertifikat nutzbar?

14

Firewalls

14.1 Wie Firewalls arbeiten

Aufgaben einer Firewall

Der englische Begriff Firewall steht für eine Wand aus unbrennbarem Material, die in Gebäuden platziert wird, um die flächendeckende Ausbreitung von Bränden zu verhindern. Als Analogie in der Informationsverarbeitung soll eine Firewall, die sich klassischerweise an der Grenze zwischen dem eigenen Netzwerk und dem Internet befindet, die Ausbreitung von Bränden vom Internet in das eigene Netz verhindern.

Eine Firewall ist ein System oder eine Gruppe von Systemen, deren Aufgabe darin besteht, die Kommunikation zu und von einem Netzwerk anhand von vorhandenen Regeln (Policies) zu erlauben oder zu verbieten. Wurde eine Firewall eingerichtet ohne eine klare Definition ihres Nutzens oder ohne dass ihre Regeln auf dem neuesten Stand sind, so dürfte sie relativ nutzlos sein.

Obwohl Firewalls gewisse Schutzmaßnahmen gegen Hacker zur Verfügung stellen können, ist ihre Existenz alleine kein Allheilmittel. Mitunter wird immer noch geglaubt, mit der Anschaffung einer Firewall wären Sicherheitsprobleme gelöst. Dieses falsche Sicherheitsgefühl kann schlimmere Folgen haben als das Bewusstsein, keinen Schutz zu besitzen.

Hard- oder Software

Firewalls sind entweder als Hard- oder Softwarelösungen verfügbar. Die Grenzen hierfür sind fließend. Die Bandbreite erstreckt sich von der Open-Source-Linux-Firewall über die Microsoft Azure Firewall bis zu Hardware Appliances, wie z. B. der SonicWall-Firewall oder der Cisco ASA-X- Firewall.



SonicWall-Firewall, Quelle: SonicWall



Cisco ASA-X Firewall, Quelle: Cisco

Bei einer Entscheidung für die eine oder andere Lösung sollten Sie berücksichtigen, dass zwar Software möglicherweise einfacher zu warten oder zu aktualisieren ist als eine Hardware-Firewall. Dagegen haben Hardwarelösungen den Vorteil, vom Betriebssystem eines normalen Rechners vollständig abgekoppelt zu sein.

In Appliances (engl. Appliance: Apparat, Gerät) kommen speziell angepasste Betriebssysteme zum Einsatz, die für die Verwendung als Firewall „gehärtet“ wurden. Das bedeutet, dass das Firewall-Betriebssystem nur über die optimierte Funktionalität der Dienste verfügt und so gegen Sicherheitslücken effizienter geschützt ist.

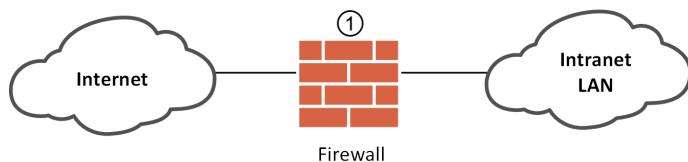
Eine Firewall ist nur so stabil wie das Betriebssystem, auf dem sie aufsetzt. Gelingt es einem Angreifer von außen, Systemrechte auf einem Firewall-System aufgrund eines Fehlers im Betriebssystem zu erlangen, so ist die Firewall nutzlos.

Firewall-Konzepte

Je nach Schutzbedarf und Topologie eines Netzwerkes können eine oder mehrere Firewalls platziert werden. Wichtig in allen Fällen ist jedoch, dass sämtliche Kommunikationswege in das geschützte Netzwerk hinein und aus dem geschützten Netzwerk heraus über die Firewall laufen.

Das beste Firewall-Konzept wird untergraben, wenn sich im internen Netz Systeme befinden, die einen Internetzugang unter Umgehung der Firewall ermöglichen, z. B. Remote-Wartungszugänge oder VPN-Clients.

Die einfachste Lösung besteht aus einer Firewall ①, die am Übergabepunkt vom Intranet zum Internet den Datenverkehr überwacht.



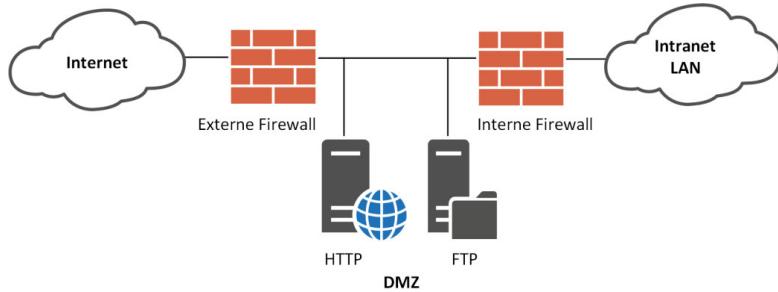
Einfaches Firewall-Konzept

Obwohl diese Lösung relativ einfach zu realisieren ist, ist die damit erzielte Sicherheit im Vergleich zu den anderen möglichen Lösungen eher bescheiden. Sollte diese Firewall selbst einem Angriff zum Opfer fallen, so steht das Intranet dem Angreifer offen bzw. ist keine Kommunikation ins Internet möglich.

Darüber hinaus ist es mit diesem Konzept problematisch, im Intranet einen Server zu betreiben, der vom Internet aus erreichbar sein soll. Konfigurieren Sie die Firewall so, dass Zugriffe von außen auf diesen benötigten Server (z. B. auf einen WWW-Server) erlaubt sein sollen, so könnte dies auch ein Angriffspunkt für Hacker werden. Gelingt es einem Angreifer in diesem Szenario, Kontrolle über den WWW-Server zu erlangen, so kann dieser als Relaystation für weitere Zugriffe auf sämtliche andere Rechner im Netzwerk benutzt werden.

Als Antwort auf die Problematik, dass das Intranet geschützt werden sollte, bestimmte Rechner aber weiterhin von außen erreichbar sein sollen, werden die von außen zugreifbaren Server in einen vorgelagerten Bereich des Intranets verlagert. Die Rechner mit diesen speziellen Serveraufgaben (Bastion-Host, da sie besonders gesichert werden sollten) befinden sich also im Niemandsland zwischen dem Intranet und dem Internet.

Als Fachausdrücke für dieses Niemandsland haben sich die Begriffe „Perimeternetzwerk“ oder „Demilitarisierte Zone“ (DMZ) durchgesetzt.



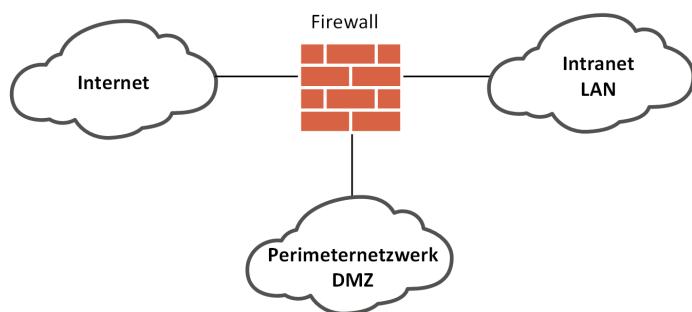
Zweistufige Firewall mit DMZ

Im Normalfall wird die externe Firewall u. a. über Zugriffsregeln (erweiterte Access-Listen) so konfiguriert, dass sie den eingehenden Datenverkehr nur erlaubt, wenn das Ziel einer der in der DMZ installierten Serverdienste ist (in diesem Beispiel HTTP und FTP). Alle anderen Verbindungsversuche werden verworfen. Die interne Firewall sollte keinen Zugriff vom Internet in das Intranet erlauben, um die Systeme im Intranet zu schützen.

Umgekehrt kann die interne Firewall so konfiguriert werden, dass es Computern aus dem Intranet nur gestattet ist, entweder den firmeneigenen WWW- oder FTP-Server zu nutzen. Eine darüber hinausgehende Nutzung des Internets kann dann an der externen Firewall unterbunden werden, die außer den Servern in der DMZ keine Verbindungen erlaubt.

Würde in so einem Fall die externe Firewall oder einer der Server in der DMZ von einem Hacker übernommen, so blieben seine Zugriffsmöglichkeiten durch das Blockieren durch die interne Firewall nur auf die DMZ beschränkt.

Eine elegante Version eines Perimeternetzwerkes kann dadurch realisiert werden, dass an einem Firewall-Rechner bzw. einer Firewall-Appliance mehrere Interfaces vorhanden sind. So wird das Perimeternetz ohne weiteren Hardwareaufwand, aber mit der Möglichkeit zur speziellen Konfiguration von unterschiedlichen demilitarisierten Zonen eingebunden.



Firewall mit frei konfigurierbaren Interfaces

14.2 Paketfilter-Firewall

Access Control Lists

Die Methoden, mit denen Firewalls zwischen erwünschtem/erlaubtem und unerwünschtem/verbotenem Datenverkehr unterscheiden, können auf unterschiedlicher Ebene im OSI-Modell ansetzen. Der einfachste Firewalltyp ist bei einer derartigen Unterscheidung die sogenannte **Paketfilter-Firewall**. Sie ist prinzipiell auf jedem Router und Multilayerswitches möglich und arbeitet statisch.

In einer Paketfilter-Firewall wird anhand von vorgegebenen Regeln verifiziert, welche Art von Datenverkehr überprüft werden soll. Das dabei verwendete Regelwerk wird Access Control List (ACL) genannt und enthält Vorschriften von Zulassen/Verbieten-Regeln zu Quelladressen, Zieladressen, Protokollen und Portnummer. Eine Paketfilter-Firewall ist auf Layer 3 und 4 des OSI-Referenzmodells angesiedelt.

```
access-list 100 permit      ip 10.0.0.0  0.255.255.255 any
access-list 100 permit      ip 172.16.0.0 0.15.255.255 any
access-list 100 permit      ip 192.168.0.0 0.0.255.255 any
access-list 100 permit      ip 192.0.2.0 0.0.0.255 any
access-list 100 permit      tcp any host mailgate eq smtp
access-list 100 permit      udp any host pdc01 eq domain
access-list 100 permit      tcp host admin1 host nas eq ftp
access-list 100 permit      tcp host admin1 host nas eq ftp-data
access-list 100 permit      tcp any gt 1024 host websrv eq www
access-list 100 deny        ip any any
```

Beispiel der ACL auf einem Cisco Router

Die ersten vier Einträge beschreiben nur Regeln auf der Netzwerkschicht (Layer 3), der zweite Teil zusätzliche Regeln auf Layer 4. Der allgemeine Aufbau einer ACL stellt sich wie folgt dar:

- ✓ Die Klassifizierung als ACL und die Nummer bzw. der Name der ACL (`access-list 100`)
- ✓ Die Aktion (Verbieten/Erlauben), die auf diesen Regeleintrag angewendet werden soll (`deny` oder `permit`)
- ✓ Das Protokoll, das diese Policy nutzen soll (Layer 3: `ip` Layer 4: `udp`, `tcp`)
- ✓ Das Source-Netz mit Wildcard (10.0.0.0 / 0.255.255.255) oder ein einzelner Host (`host admin1`) oder jeder (`any`) und ggf. der Source-Port (im Bsp. `gt 1024`, greater than 1024, größer als 1024)
- ✓ Das Destination-Netz mit Wildcard oder ein einzelner Host (`host mailgate`) oder jeder (`any`) und ggf. der Destination-Port (im Bsp. `eq snmp`, equal snmp, gleich snmp)

Die oben genannten Regeln werden nacheinander abgearbeitet. Sobald eine Übereinstimmung (Match) mit einem Paket erfolgt, wird das Regelwerk verlassen und nachfolgende Regeln kommen nicht mehr zur Anwendung. Damit es immer einen Match gibt, heißt die letzte Regel, die automatisch vom System gesetzt wird

- ✓ `access-list <Nummer oder Name> deny ip any` bzw. bei einer erweiterten ACL
- ✓ `access-list <Nummer oder Name> deny ip any any`.

Beim Konfigurieren von Access-Listen haben Sie zwei Optionen. Sie sperren jeden nicht genutzten Verkehr (IP-Adresse oder Portnummer der Applikation) oder Sie erlauben nur definierten Verkehr. Die erste Option hat den Nachteil, dass Sie ggf. nicht jeden Verkehr berücksichtigt haben. Der zweiten Option sollten Sie den Vorzug geben, auch wenn Sie nicht jeden zu erlaubenden Traffic berücksichtigt haben.

Da das Filtern von Paketen auf Grundlage dieser **statischen** Listen keinen kompletten Schutz vor komplexen Angriffen bieten kann, ist ein alleiniger Schutz durch Paketfilter nicht zu empfehlen, aber zur Basisfilterung sinnvoll.

14.3 Stateful Inspection Firewall

Pakete im Kontext betrachten

Im Gegensatz zu den statischen Paketfiltern ist das Stateful Inspection Firewalling eine **dynamiche** Filtermethode. Eine Firewall, die für die Entscheidung über die Weiterleitung eines bestimmten Datenpakets nicht nur die Merkmale dieses Pakets, sondern auch den individuellen Kontext in Betracht zieht, in dem es durchgeführt werden soll, ist eine **Stateful Packet Inspection Firewall (SPI)**.

Um die nötigen Informationen als Entscheidungsgrundlage zur Verfügung zu haben, legt die Firewall auf der Basis der Regelstabelle zusätzlich eine State-Tabelle (engl. State: Zustand) an, in der bei den ein- und ausgehenden Verbindungen alle Zustände vermerkt sind.

Der Firewall ist somit bekannt, welche Pakete zu einem Zeitpunkt legitim sind und prinzipiell passieren können und welche zu keiner gültigen Verbindung gehören. Ein Angriff, der beispielsweise Pakete mit ACK-Flags an Rechner im geschützten Netz sendet und vorgaukelt, der Angreifer würde nur auf eine Anfrage des geschützten Rechners antworten, kann somit die Firewall nicht mehr überlisten, da hier in der State-Tabelle keine ausgehende Anfrage eines geschützten Rechners vermerkt ist, zu der das empfangene Paket mit ACK-Flag passen würde – das Paket wird verworfen.

Mit anderen Worten: Die Ports einer SPI-Firewall sind grundsätzlich geschlossen. Es sei denn, eine Anfrage von innen heraus benötigt diesen Port. In so einem Fall wird der Port nur für den Zeitraum, in dem er benutzt wird, geöffnet.

Um eventuelle Verbindungsinformationen aus den Paketen auslesen zu können, muss eine SPI-Firewall den Inhalt der Pakete genauer als nur bis zu den IP- und Portnummern überprüfen. Untersuchte Daten sind bei TCP-Datenströmen z. B. die angesprochenen Flags und die Sequenznummern der Pakete. Eine spezielle Form der SPI ist die Deep Packet Inspection (einige Hersteller bezeichnen dies auch als Application Control), bei der auch die Nutzdaten der Datenpakete analysiert werden.

IP-Fragmentation

Es existieren Angriffsmethoden, die darauf abzielen, den Inhalt von Paketen vor der Inspektion durch eine Firewall zu verbergen. Bei diesen sogenannten Fragmentation-Attacken werden die versendeten Pakete bewusst in Fragmente aufgeteilt.

Fragmentierung kommt in IP-Netzwerken üblicherweise dann zum Einsatz, wenn die Paketlänge auf dem Weg vom Sender zum Empfänger die maximale Segmentgröße in einem Netzwerk übersteigt. Die aufgeteilten, kleineren Fragmente können aber problemlos übertragen werden und werden empfängerseitig wieder zusammengesetzt.

Durch geschicktes absichtliches Fragmentieren von Datenpaketen war es möglich, eine Firewall über den Inhalt der Pakete zu täuschen.

Analog dazu entgeht dem menschlichen Beobachter der Inhalt eines Wortes leichter, wenn dieses in Buchstabengruppen „fragmentiert“ wurde:

CY BE RKR I MI NA L ITÄ T.

Haben Sie nur wenig Zeit, dieses Wort zu entziffern, entgeht Ihnen möglicherweise die Bedeutung.

Ein weiterer Effekt, der durch IP-Fragmentierung genutzt wird, lässt sich als Text so umschreiben:

Ein fragmentiertes Paket wird verschickt. Die erste Hälfte enthält „LASTWAGEN“, die zweite Hälfte „~~LAST~~ERHAFT“.

Eine nicht korrekt reassemblierende Firewall würde ggf. den Inhalt des Paketes als „LASTWAGEN-~~LAST~~ERHAFT“ beurteilen und als nicht bedrohlich. „~~L~~“ ist jedoch ein Löschzeichen und löscht das zuvor gedruckte Zeichen. In diesem Textbeispiel werden also beim Zusammensetzen beider Informationshälften die letzten fünf Zeichen von LASTWAGEN gelöscht und die Zeichen ERHAFT angefügt. Das Ergebnis ist LASTERHAFT.

Auf IP-Ebene wird ein ähnlicher Effekt durch Manipulation der Datenzeiger erreicht. So werden die zuvor gesendeten „Tarn“-Daten mit den Daten der zweiten Fragmenthälfte wieder überschrieben, sodass sie auf dem Zielrechner das vom Hacker gewünschte Datenpaket ergeben.

14.4 Proxy Level/Application Level Firewall

Höhere Schichten im OSI-Modell

Werden neben der Paketinformation in den Datenströmen auch die Informationen von höheren Schichten ausgewertet, so ist es technisch möglich, einen Datenstrom bis zur Anwendungsebene auf Schicht 7 wieder zusammenzufügen. Die Firewall wertet nun den enthaltenen Datenverkehr nicht mehr nur auf Netzwerk- oder Transportebene aus, sondern überprüft die Informationen bis zur Applikationsschicht. Dadurch kann auch der Dateninhalt vollständig überprüft werden.

Logisch gesehen bedeutet das: Wird mit Datenpaketen eine E-Mail versandt, so baut die Firewall die Pakete zusammen, bis die E-Mail komplett auf dem Firewall-System vorliegt. Anschließend kann entschieden werden, ob die Weiterleitung dieser E-Mail in Abhängigkeit von Absender, Empfänger, Betreff, Inhalt oder angehängten Dateien erlaubt oder gesondert behandelt wird. Dieses Konzept ist die Basis einer **Application Level Firewall** oder **Proxy Firewall**. Je nach Protokoll ermöglicht diese Vorgehensweise durch entsprechende Regeln z. B. das Sperren von E-Mail von bestimmten Absendern oder mit definierten Inhalten sowie das Blockieren von Websites mit bestimmten Inhalten (z. B. Active Scripting). So könnten auch Werbebanner und Popup-Fenster gefiltert, der restliche Inhalt einer Website aber zugelassen werden.

Wenn Application Level Firewalls quasi als Stellvertreter für Clients den eigentlichen Server im Internet kontaktieren, werden sie als Proxy-Firewalls bezeichnet (engl. Proxy: Stellvertreter, Bevollmächtigter).

Der Nachteil dieser Methode ist der höhere Ressourcenaufwand, der nötig ist, um in der Firewall diese Protokolle nachzubilden und entsprechende Filterregeln anzuwenden. Des Weiteren ist für jede Anwendung bzw. jedes Protokoll eine entsprechend kompatible Proxy-Firewall nötig. Firewalls auf Paketebene arbeiten dagegen anwendungsunabhängig.

Zugriff auf eine Webseite im Internet

Möchte beispielsweise ein Clientrechner eine Webseite eines bestimmten Servers abrufen, so wird die Verbindung zunächst mit der Firewall aufgebaut. Diese prüft die Zulässigkeit der Verbindung und baut dann ihrerseits die Verbindung zum gewünschten Server auf. Nun wird die HTML-Seite übertragen und ihr Inhalt analysiert. Soll die Firewall z. B. Werbebanner filtern, so werden aus der empfangenen HTML-Seite alle Anweisungen entfernt, die einen Popup-Aufruf oder eine Bannereinblendung enthalten. Diese so modifizierte Seite wird dann an den Client weitergeleitet. Da der Client seinerseits die HTML-Seite auswertet und anschließend die zu deren Darstellung nötigen Grafiken nachlädt, lädt die Firewall schon vorsorglich die entsprechenden Grafiken vom Internet-Webserver und reicht diese an den Client weiter. Als Ergebnis sieht der Client eine werbefreie Webseite. Auch die Bandbreite wird geschont, da Bannergrafiken gar nicht erst heruntergeladen werden müssen.

Flexible Einsatzmöglichkeiten

Je nach Aufbau und Ausrichtung der Regelsets können so Webseiten, FTP-Dienste oder beliebige andere Internetprotokolle und -anwendungen gefiltert und überwacht werden, bevor deren Nutzung für interne Clients zugelassen wird.

Proxies bieten die umfassendsten Möglichkeiten zur Filterung von Datenverkehr, sind aber aufwendig, da die Proxyfunktionen für jede Anwendung angepasst sind. Um die Performance der Application Level Firewall zu gewährleisten, werden u. a. einige Proxyanwendungen der Applikationen zusammengefasst. Man nennt dies **generic Proxy**.

14.5 NAT

Interne IP-Adressen verbergen

Die chronische Knappheit an öffentlich verfügbaren IP-Adressen beim IPv4-Protokoll sorgt für Probleme, wenn ein Netzwerk an das Internet angebunden werden soll. Damit jeder Rechner Zugang zum Internet hat, benötigt er eine offiziell gültige IP-Adresse. Allerdings wäre es nicht sinnvoll, für jeden Rechner in einem Netzwerk auf Dauer eine Adresse zu reservieren.

Darüber hinaus wäre ein Anschluss von Netzwerkrechnern mit öffentlichen IP-Adressen problematisch, weil theoretisch eben diese IP-Adressen von außen ebenfalls zu erreichen wären, sofern man keinen konfigurierten Paketfilter betreibt.

Das Konzept hinter der **Network Address Translation (NAT)** ist nun, dass nur die Firewall oder ein Router, der den LAN-Anschluss an das Internet realisiert, eine gültige öffentliche IP-Adresse besitzt. Alle im LAN befindlichen Rechner erhalten private IP-Adressen (nach RFC 1918, z. B. aus dem Bereich 192.168.x.x), die von jedermann frei vergeben werden können.

Auf diese Weise wird vom Pool an öffentlichen Adressen nur eine einzige benötigt, und zusätzlich werden die Rechner im LAN vor dem Internet verborgen. Ein Außenstehender sieht nur die öffentliche IP-Adresse des Routers und glaubt, dieser wäre sein Kommunikationspartner. Ein Client im LAN, der eine Verbindung zum Internet aufbauen möchte, kommuniziert wie üblich über den Router als Gateway.

Verwaltung der Verbindungen mit NAT

Bei ausgehenden Verbindungen vermerkt die Firewall oder der Router in einer Tabelle, welcher interne Client zu welchem Rechner im Internet eine Verbindung aufbauen möchte. Anschließend passt sie den Header der ausgehenden IP-Pakete so an, dass die externe IP-Adresse des Routers im Absender-Feld steht. Diese Pakete werden dann versandt.

Treffen an der Firewall oder dem Router Pakete aus dem Internet ein, so wird verglichen, ob diese eine Antwort auf eine Anfrage von innen sind. Wenn ja, kann sie eine Zuordnung zur entsprechenden LAN-IP vornehmen, den Header der Datenpakete entsprechend anpassen und diese dann ins LAN weiterleiten. Kann sie keine Übereinstimmung finden, so werden von außen kommende Datenpakete verworfen.

Sicherheit von NAT

NAT ist eine sichere und effiziente Methode, die interne Netzstruktur gegenüber dem Internet zu verbergen. Zu beachten ist jedoch, dass das Betreiben von Servern, die öffentlich verfügbar sein sollen, hinter einer NAT-Firewall nicht ohne Weiteres funktioniert. Dies hängt von der Wahl des NAT-Verfahrens ab.

Da eintreffende Pakete ohne Anforderung von innen verworfen werden müssen, weil die Firewall sie keinem internen Rechner zuordnen kann, muss die Zuordnung manuell vorgenommen werden. Diese Zuordnung geschieht meist anhand einer Tabelle, in der für eingehende Pakete mit bestimmten TCP- und UDP-Portnummern die internen Ziel-Adressen definiert werden. Hierbei spielt es keine Rolle, ob ein eingehendes TCP-Paket auf Port 25 auf den Intranet-SMTP-Server ebenfalls auf Port 25 oder auf irgendeinen anderen Port umgeleitet wird. Die Verwendung von Nicht-Standard-Portnummern bringt allerdings kein erhöhtes Maß an Sicherheit.

14.6 Personal Firewall

Firewall auf dem Desktop-PC

Auf dem Softwaremarkt haben sich für Endsysteme (Desktops) Personal Firewalls etabliert. Eine Personal Firewall ist ein Softwarepaket, das den im Betriebssystem vorhandenen TCP/IP-Stack durch einen funktionserweiterten Stack der Firewall ersetzt und somit dem Anwender die Kontrolle über die Nutzung der Internet-Funktionalität durch die laufenden Programme ermöglicht. In Microsoft-Betriebssystemen ab Windows XP SP2 ist diese Firewall-Funktionalität bereits integriert.

Eine Personal Firewall erledigt also innerhalb eines PCs dieselbe Funktion wie ein Firewall-Router in einem Firmen-Netzwerk.

Der Vorteil vor allem für Privatanwender ist, dass diese ihren PC bei der Nutzung des Internets relativ kostengünstig vor Zugriffen von außen schützen können. Darüber hinaus beinhalten Personal Firewalls Funktionen, die auch den Verbindungsweg vom Computer in das Internet überwachen und so unkontrollierbares „Nach-Hause-Telefonieren“ von installierter Software unterbinden können.

Personal Firewall in der Firma

In Firmennetzwerken sind Personal Firewalls eher selten anzutreffen, da sie eingeschränkte Konfigurationsmöglichkeiten bzw. kein zentrales Managementtool aufweisen und eine Remote-Überwachung selten vorhanden ist.

Wird der Schutz eines Firmennetzwerkes nur der „Haupt-Firewall“ am Einwahlpunkt ins Internet überlassen, so könnte die geschickte Platzierung eines Trojaners oder ein unbefugt angeschlossenes Modem am PC eines Mitarbeiters jeglichen Schutz unwirksam werden lassen. Ist jedoch jeder PC einzeln wiederum durch eine „kleine“ Firewall geschützt, so erschwert das die Arbeit eines Hackers ungemein.

Bekannte Personal Firewalls

Eines der ersten verfügbaren Personal-Firewall-Produkte war **Zonealarm**, das auch in einer für Privatanwender kostenlosen funktionsreduzierten Variante angeboten wird. Zonealarm ist eindeutig auf den unerfahrenen Anwender ausgerichtet und bietet demnach nur wenige Konfigurationsmöglichkeiten. Problematisch wird die Ausrichtung auf den unerfahrenen Anwender immer dann, wenn dieser bei einem Verbindungsversuch eines Programms nach einer Erlaubnis oder einer Verweigerung gefragt wird.

Unerfahrene Benutzer können aber selten entscheiden, ob ein bestimmter Prozess für seine einwandfreie Funktion Internetzugriff benötigt oder nicht. Es besteht also durchaus die Gefahr, dass der Benutzer zwar in guter Absicht eine Personal Firewall installiert, um sein System sicherer zu machen, diese potenzielle Sicherheit aber dann durch schlechte Konfiguration wieder verliert.

Auf dem Markt existieren neben der eben kurz beschriebenen Firewall eine Reihe von frei verfügbaren Applikationen. Dazu zählen u. a.:

- ✓ <https://www.zonealarm.com/de/software/free-firewall>
- ✓ <https://www.evorim.com/de/free-firewall>
- ✓ <https://www.comodo.com/home/internet-security/firewall.php>
- ✓ <http://www.sphinx-soft.com/Vista/index.html>

Firewall-Lösungen von Drittanbietern können eigenständige Lösungen oder einen erweiterten GUI-Aufsatz für die systemeigene Windows-Defender-Firewall darstellen. Die Defender-Firewall weist als integrierter Bestandteil des Betriebssystems schon eine umfassende Schutzfunktion auf. Deshalb sollten Sie nur dann eine Alternative ins Auge fassen, wenn erweiterte Kenntnisse dieser Sicherheitstechnologie vorhanden sind.

14.7 Sicherheitskonzept Firewall

Für den Schutz Ihres Netzwerkes sollten Sie ein mehrstufiges Sicherheitskonzept anwenden. Dies sollte sowohl Aspekte der Sicherheit in Ihrem Intranet als auch Zugriffsregularien zwischen Intranet und Internet beinhalten. Aspekte des lokalen Zugriffsschutzes im Intranet finden Sie detailliert in Kapitel 20 (Proaktive Sicherheit). Einzelne Bereiche Ihres Intranets können Sie separat mit einer Stateful Packet Inspection Firewall absichern. Als Verbindung zwischen Internet und Intranet sollten mindestens eine Stateful Inspection Firewall und Proxies für die wichtigsten Dienste implementiert werden. Diese Firewall sollte auch die DMZ beinhalten. Auf dem Router, der als Verbindung zum Provider dient, sollten Sie zusätzlich Paketfilter aufsetzen.

14.8 Erweiterte Funktionen der Firewall

Eine Firewall, und an dieser Stelle wird primär auf den am häufigsten verwendeten Typ (Stateful Inspection Firewall) eingegangen, weist hauptsächlich die nachfolgende Charakteristik auf:

- ✓ Sie verfügt über mindestens drei physikalische Interfaces (Intranet, Internet, DMZ, ...). Eine Ausnahme hierbei bildet die Desktop Firewall (Personal Firewall). Da sie nur auf einem Desktop installiert ist, hat sie nur ein Interface für eingehenden und ausgehenden Traffic.
- ✓ Sie filtert/überwacht den Verkehr zwischen den Interfaces. Dies kann auf Layer 2 (z. B. Filtern vom MAC-Adressen oder L2-Protokollen), auf Layer 3 (z. B. Überprüfung der IP/IPv6-Adressierung, der Protokolle icmp, igmp, icmpv6, OSPF, GRE, IP-Optionen, ...), auf Layer 4 (z. B. die Verifizierung von TCP, UDP, SSL/TLS und der Portnummern) oder auf Layer 7 (z. B. Blockieren unerwünschter Applikationen für bestimmte Benutzer oder Gruppen, Aktivieren oder Deaktivieren bestimmter Unterfunktionen von Applikationen) erfolgen. Der letzte Punkt wird durch Deep-Inspection- oder Proxy-Funktionen umgesetzt.
- ✓ Sie verhindert DoS/DDoS-Angriffe durch die Limitierung der TCP/UDP-Zugriffe. Dadurch werden eine Überlastung des Stacks und damit eine Nacherreichbarkeit von Diensten im eigenen Netz bzw. der eigenen DMZ ausgeschlossen.
- ✓ Sie kann Netzwerkbedrohungen mittels Advanced Thread Detection proaktiv erkennen. Dazu werden die Dateien über eine kryptografische Hashdatenbank abgeglichen. Wird hierbei eine unbekannte Datei erkannt, so wird sie in einer virtuellen Sandbox emuliert, um das Verhalten der Datei zu analysieren. Dies kann in einer Sandbox des Unternehmens oder in einer Security-Cloud des Firewall-Anbieters erfolgen. Eine Security-Cloud ist eine Infrastruktur im eigenen Unternehmen oder von Drittanbietern, in der Angriffsszenarien in Echtzeit analysiert werden.
- ✓ Sie unterbindet Netzwerkbedrohungen in Echtzeit durch ein Intrusion Detection System/Intrusion Prevention System (IDS/IPS). Es ist ein eigenständiges Modul, das vollständig den Traffic am Eingang bzw. Ausgang der Firewall analysiert. Dadurch werden u. a. Bedrohungen wie willkürliche Codeausführungen, Scanningversuche und einige Malwarecodes (Backdoor, Trojaner, Rootkits, Viren, Würmer und Spyware) erkannt und beseitigt.
- ✓ Sie setzt Quality of Service (QoS) bei der Nutzung von Applikationen (z. B. Kontrolle und Drosselung von akzeptablem Datenverkehr oder Einsparung von Bandbreite und Beschleunigung geschäftskritischer Applikationen zur Sicherstellung der Geschäftskontinuität) um.
- ✓ Sie sichert den Netzwerzugang durch die Nutzung von Site-to-Site- und Remote-to-Site-VPNs. Die Kommunikation erfolgt auf der Basis von SSL/TLS- oder IPsec-Protokollen. VPNs können durch NAT oder PAT (Port Address Translation) ergänzt werden.

- ✓ Sie realisiert Verfügbarkeit. Hersteller bieten Failover-Mechanismen an, wodurch bei Systemausfall eine redundante Firewall die Sicherheitsfunktion ohne Unterbrechung der Kommunikation übernimmt. Eine weitere Funktion ist Link-Balancing, welche bei Ausfall einer physikalischen Verbindung einen alternativen Link aufbaut.
- ✓ Sie lässt sich virtualisieren. Um für jede Usergruppe eine eigene und voneinander unabhängige Sicherheitslösung anbieten zu können, kann diese virtualisiert werden. Damit erhält man eine eigenständige (mandantenfähige) Firewall. Diese kann jedoch nur eingeschränkte Features einer Sicherheitslösung umsetzen.
- ✓ Sie protokolliert umfassend und bietet transparente, belegbare und übersichtliche Angaben z. B. zur Bandbreitennutzung sowie alle anderen sicherheitsrelevanten Informationen. Zusätzlich kann ein Revisionskontrollsystem integriert sein. Es erleichtert das Audit der Infrastruktur. Darüber hinaus stellt es bei allen Änderungen sicher, dass diese die Richtlinien der Behörden und des Unternehmens erfüllen.

Die aufgezeigten Funktionen stellen die Möglichkeiten der Nutzung von Sicherheitsaspekten für eine Firewall auf.

Inwieweit eine vollständige Umsetzung aller Aspekte erfolgt, hängt vom Einsatzgebiet der Firewall ab.

14.9 Übung

Fragen zu Firewalls

Übungsdatei: --

Ergebnisdatei: uebung14.pdf

1. Was verbirgt sich hinter dem Begriff Access Control List (ACL)?
2. Welche Firewall-Typen kennen Sie? Beschreiben Sie die grundsätzlichen Unterschiede.

15

Intrusion-Detection/Prevention-Systeme

15.1 Notwendigkeit von Intrusion-Detection-Systemen

Firewall alleine ist nicht genug

Firewalls haben sich als Schutzmaßnahme gegen unbefugte Zugriffe in Unternehmensnetzwerken etabliert und gehören zur Standardausstattung eines professionell administrierten Netzwerkes. Dennoch besteht auch mit einer Firewall und einer vernünftigen Konfiguration eine gewisse Chance, dass vorhandene Schutzmaßnahmen umgangen werden können.

Ein Intrusion-Detection-System (IDS) ist, wie der englische Begriff schon vermuten lässt, ein System zur Erkennung eines Einbruchs. Während andere Maßnahmen darauf abzielen, Einbrüche in das geschützte Netzwerk zu verhindern, ist ein IDS in einem Netzwerk installiert, um einen Einbruch erkennen zu können.

Dies lässt sich vergleichen mit einem Schloss, das Sie an Ihrer Haustür anbringen. Das tun Sie, um unberechtigten Zutritt zu Ihrem Haus zu verhindern. Haben Sie nun vergessen, das Fenster auf der Rückseite des Hauses abzuschließen, oder konnte ein Einbrecher das Türschloss aufbrechen, wird er Ihr Haus trotzdem plündern. Ist er gerissen, wird er nach der Plünderung sogar Ihr Fenster wieder schließen und die Tür verriegeln. Wenn Sie abends nach Hause kommen, bemerken Sie den Einbruch womöglich erst, wenn Sie Ihre Wertgegenstände vermissen.

Alarmanlage für Netzwerke

In so einem Fall wäre es doch praktisch, eine Alarmanlage zu haben. Diese wird aktiv, wenn Sie das Schloss in der Haustür verriegeln. Zwar kann sie nicht verhindern, dass das Schloss aufgebrochen wird, aber sie wird erkennen, dass die Tür geöffnet wurde, ohne das Schloss ordnungsgemäß aufzusperren, oder dass gerade ein Fenster geöffnet wurde.

Wie Ihre Alarmanlage dann reagiert, können Sie frei bestimmen. Wollen Sie den Einbrecher in die Flucht schlagen, indem ein lauter Alarm ertönt, oder soll die Anlage einen stillen Alarm bei der nächsten Polizeistelle aktivieren, damit der Eindringling idealerweise noch in flagranti festgenommen wird?

Viele Angriffe bleiben unentdeckt

Bezogen auf Computernetzwerke zeichnet sich folgende Situation ab: Gemäß einer Studie der DISA (Defense Information Systems Agency) waren 72 % der durchgeführten Einbruchsversuche in Computer von US-Behörden erfolgreich. 82 % der erfolgreichen Einbrüche blieben von den Opfern vollkommen unbemerkt.

Um also einen eventuell schon entstandenen Schaden nicht noch weiter eskalieren zu lassen oder während eines noch laufenden Einbruchs bereits geeignete Gegenmaßnahmen ergreifen zu können, ist ergänzend zu den vorhandenen Schutzmaßnahmen dringend ein Warnsystem nötig. Schafft es ein IDS, einen laufenden Hackerangriff zu identifizieren, so können die aufgezeichneten Informationen zusätzlich zur Beweissicherung und zur Identifizierung des Angreifers benutzt werden.

15.2 Arbeitsweise eines IDS

Anomalieerkennung

Ein Einbruchsversuch in ein Computernetzwerk oder eine anderweitige Sicherheitsverletzung zeichnet sich dadurch aus, dass die stattfindende Kommunikation von den Vorgängen im Normalzustand abweicht. Ein IDS, das mit Anomalieerkennung arbeitet, geht davon aus, dass Benutzer und Komponenten sich in einem Netzwerk statistisch gesehen konstant verhalten. Dieses Verhalten lässt sich statistisch als Verhaltensmuster erfassen.

Weicht im späteren Betrieb das Verhaltensmuster des überwachten Netzwerkabschnittes vom normalen Muster ab, so deutet dies auf eine eventuelle Sicherheitsverletzung hin.

Für ein IDS, das mit Anomalieerkennung arbeitet, ist es also zuerst nötig, über einen bestimmten Zeitraum genügend Messdaten im Netzwerk zu sammeln. Hierbei werden zahlreiche Parameter des Netzwerks und Computerbetriebs berücksichtigt. Aus diesen gesammelten Daten wird ein Datenmuster für den Normalzustand ermittelt und mit Schwellenwerten versehen.

Danach kann das IDS-System „scharf“ geschaltet werden. Es überwacht die im laufenden Betrieb gemessenen Daten mit den vorgegebenen Schwellenwerten. Wenn diese überschritten werden, kann das IDS entsprechende Gegenmaßnahmen oder eine Alarmierung einleiten.

Der Vorteil der Anomalie-Erkennungsmethode ist, dass Angriffsmuster nicht explizit definiert werden müssen. Eine bis dahin unbekannte Sicherheitsverletzung wird auch als solche erkannt, wenn sie sich in einer Abweichung vom normalen Verhalten niederschlägt. IDS über Anomalieerkennung scheinen nach dem anfänglichen Erstellen des statischen Verhaltensmusters also relativ wartungsfrei zu sein.

Allerdings werden auf diese Weise Angriffe als solche nicht erkannt, deren Verhaltensmuster von der Statistik nicht signifikant abweicht. Außerdem kann es auch zu Fehlalarmen kommen, wenn beispielsweise ein Benutzer sein Verhalten legitim ändert und/oder neue Software mit neuen Verhaltensmustern auf Clients installiert wird.

Diese Anomalieerkennung kann z. B. verwendet werden, um einen Wurm-Befall des Netzwerkes festzustellen. Wenn ein PC mit einem Wurm an das Netzwerk angeschlossen wird, dann wird dieser sofort versuchen, sich weiter zu verbreiten. Dafür scannt er das Netz nach anderen PCs, die eine ausnutzbare Sicherheitslücke beinhalten. Dieser Traffic weicht stark vom regulären Netzwerkverkehr ab und kann daher sehr gut von anomaliebasierten IDS erkannt werden.

Signaturanalyse

Die Signaturanalyse arbeitet ähnlich wie ein VirensScanner mit einem Datensatz aus typischen Angriffsszenarien. Für jeden in der Datenbank erfassten Angriff existiert ein dafür typisches Angriffsmuster (die Signatur). Das IDS überwacht anhand der Signaturen den laufenden Datenverkehr und meldet bei einer Übereinstimmung eine Sicherheitsverletzung.

Eine Signaturanalyse erspart zwar das Erfassen von Messdaten im Normalzustand und die aufwendige statistische Auswertung, allerdings konzentriert sich der Aufwand auf die Aktualisierung der Signaturdatenbank. Analog zu Virensprechern kann ein signaturbasiertes IDS nur solche Angriffe erkennen, zu denen bereits eine passende Signatur bekannt ist. Neue, ungewohnte Angriffe werden vom IDS nicht erkannt.

Im Gegensatz zur Anomalieanalyse ist jedoch die Chance für einen Fehlalarm (false positive) deutlich geringer. Während bei einer Anomalieanalyse eine Abweichung vom alltäglichen Verhalten im Computernetzwerk schon zu einem Alarm führt, schlägt ein Signatur-IDS nur Alarm, wenn ein im Netzwerk gefundenes Datenmuster eindeutig zu der Signatur eines bekannten Angriffs passt.

Die meisten am Markt befindlichen Intrusion-Detection-Systeme arbeiten Signatur-basiert, haben aber oftmals eine zusätzliche Komponente der Anomalie-Erkennung.

Intrusion-Detection-Terminologie

✓ **False Positive**

Ein IDS hat einen Angriff erkannt (Positive), dies war aber eine fehlerhafte Erkennung (False). Dies wird oftmals durch zu allgemein gehaltene Signaturen hervorgerufen.

✓ **False Negative**

Ein IDS hat keinen Angriff erkannt (Negative), dies war aber ein Fehler (False). In diesem Fall ist ein Angriff durch den IDS-Sensor unentdeckt geblieben. Um False Negative zu erkennen, muss dieser Angriff durch eine andere Security-Komponente entdeckt werden.

✓ **True Positive**

Der Sensor hat auf Traffic reagiert, der wirklich ein Angriff war.

✓ **True Negative**

Der Sensor sendet auf Traffic, der legitim ist, keinen Alarm.

Host- oder netzwerkbasiertes IDS

Sollen einzelne Rechner geschützt werden, wird ein sogenanntes hostbasiertes IDS eingesetzt. Dieses überwacht direkt den zu schützenden Rechner. Zur Überwachung ganzer Netzwerkabschnitte haben sich in letzter Zeit die netzwerkbasierten IDS durchgesetzt. Netzwerkbasierte IDS müssen deutlich leistungsfähiger sein, weil sie in der Lage sein müssen, alle über ihren Netzwerkanschluss fließenden Daten zu analysieren und in Echtzeit auszuwerten. Heutige netzwerkbasierte Intrusion-Detection-Systeme können Netze mit mehreren Gbit/s Durchsatz analysieren.

IDS platzieren

Hinsichtlich der Platzierung eines IDS ergeben sich unterschiedliche Möglichkeiten, die davon abhängig sind, zu welchem Zweck ein IDS eingesetzt werden soll:

- ✓ Vor der Firewall und dem zu schützenden Netz
- ✓ Hinter der Firewall in dem zu schützenden Netz

Wenn Sie das IDS vor Ihrer Firewall platzieren, so sieht das IDS sämtliche Angriffe von außen. Egal, ob ein Angriff erfolgreich sein wird oder nicht, kann das IDS aktiv werden, wenn es selbst die Bedrohung erkannt hat.

Wird das IDS im geschützten Netz platziert, so werden nur noch die erfolgreichen Angriffe von außen registriert, die in der Lage waren, die Firewall zu passieren. Zusätzlich jedoch ist das IDS hier auch in der Lage, Sicherheitsverletzungen von internen Mitarbeitern zu entdecken. Auf diese Weise werden also potenziell auch Fehlbedienungen, Unachtsamkeiten oder absichtliche Manipulationen erkannt, die von Personen mit Zugriff auf das geschützte LAN ausgehen.

Daten zum Intrusion-Detection-System senden

Damit das IDS Angriffe erkennen kann, muss der Traffic zusätzlich zum normalen Ziel auch zum IDS gesendet werden. Dort gibt es verschiedene Möglichkeiten:

- ✓ Verwendung von Netzwerk-Taps. Dies sind Komponenten, die in die Verbindung eingeschleift werden und den Traffic an zwei Ausgabeports (einer pro Richtung bei Full-Duplex) zu dem IDS senden.
- ✓ Spiegel-Ports an manageable Switches. Hierbei werden Switches so konfiguriert, dass der Traffic eines Ports oder eines VLANs auf den Port gespiegelt wird, an dem das IDS angeschlossen ist.

15.3 Auf erkannte Angriffe reagieren

Wie ein IDS auf einen entdeckten Angriff reagiert, hängt von der Konfiguration ab und ist sehr flexibel:

- ✓ Alarmierung des für die Sicherheit zuständigen Administrators über das Netzwerk (weniger empfehlenswert) oder einen vom überwachten Netzwerk unabhängigen Kanal (besser), wie z. B. SMS
- ✓ Versuch, die Netzverbindung des Angreifers zu trennen, indem gezielt RST-Pakete versandt werden
- ✓ Automatische Rekonfiguration eines Routers oder einer Firewall, um die Verbindung zu unterbinden (inkl. späterer Wiederherstellung der Originalkonfiguration)
- ✓ Start einer detaillierten Protokollierung der Vorgänge zur Beweissicherung und späteren Analyse
- ✓ Ausführen von anderen definierbaren Programmen

Unter dem Begriff „Intrusion Reaction/Response“ werden folgende Maßnahmen zusammengefasst:

- ✓ Ermittlung der IP-Adresse des Angreifers
- ✓ Automatische Zurückverfolgung des Angriffs zu seinem Ausgangspunkt
- ✓ Automatische Sammlung von Informationen über den Angreifer aus verfügbaren Quellen (finger, whois) und Feststellung des Besitzers der IP-Adresse

! Es existieren auch IDS-Lösungen, die eine für gewöhnlich als „Fire-Back-Funktion“ beworbene Eigenschaft haben. Für den Fall, dass ein Angriff entdeckt wird, ist ein solches System in der Lage, den Angreifer selbst anzugreifen (z. B. mit einer DoS-Attacke).

Problematisch sind Fire-Back-Funktionen auch deshalb, weil es keine Garantie gibt, mit dem Gegenangriff auch den Richtigen zu treffen. Schließlich kann die vermeintliche Absender-IP mit geringem Aufwand gefälscht werden. Im Extremfall könnte Mallory die Fire-Back-Funktion sogar zu einem DoS-Angriff auf Bob nutzen, den er nicht einmal selbst ausführen muss: Mallory fingiert einen Angriff auf das Netzwerk von Alice, das mit einem Fire-Back-IDS geschützt ist. Dabei gibt er die IP-Adresse von Bob an. Das IDS von Alice bemerkt den Angriff und beginnt damit, eine DoS-Attacke auf den vermeintlichen Angreifer Bob zu fahren.

Diese Situation würde noch weiter eskalieren, wenn auch Bob ein Fire-Back-IDS hätte und nun seinerseits Alice attackieren würde.

15.4 Intrusion-Prevention-Systeme (IPS)

Die Weiterentwicklung der Intrusion-Detection-Systeme sind **Intrusion-Prevention-Systeme**. Diese arbeiten normalerweise inline, befinden sich also direkt im Kommunikationsweg.

Der Vorteil dieser Implementierung ist, dass ein erkannter Angriff zuverlässig verhindert werden kann. Dafür werden die Pakete, die zu einem Angriff gehören, von dem IPS verworfen und erreichen damit das Angriffsziel nicht. Wichtig bei einem so arbeitenden System ist, dass es zu keinen Fehlalarmen (False Positives) kommt. Denn bei diesen würde das IPS legitimen Verkehr verwerfen.

Viele Firewalls oder auch Security-Router haben heute Intrusion-Prevention-Systeme mit eingebaut. Dabei wird der Traffic, der von der Firewall durchgelassen wird, zusätzlich noch gegen die Signaturen des IPS verglichen. Dadurch kann der Schutz von Systemen weiter verbessert werden, wenn z. B. die Firewall den HTTP-Traffic per Regel zu einem Webserver erlaubt, in diesem Traffic aber ein Angriff auf den Webserver durchgeführt wird.

15.5 Snort

Was ist Snort?

Neben vielen kommerziellen Soft- und Hardware-IDS-Lösungen ist Snort ein weit verbreitetes IDS auf Open-Source-Basis. Obwohl sich Snort selbst als „Lightweight Network Intrusion Detection Tool“ bezeichnet, verfügt es über flexible Konfigurationsmöglichkeiten und zahlreiche Funktionen, die es zu einem vielseitig einsetzbaren IDS machen.

Snort ist ursprünglich für UNIX-Systeme geschrieben worden, wurde aber auch auf Windows portiert. Das Snort-Hauptprogramm wird immer noch über Kommandozeilenbefehle und Konfigurationsdateien gesteuert. Es existiert jedoch ein Windows-Verwaltungstool namens IDSCenter, das versucht, die Konfigurationsmöglichkeiten in einer grafischen Oberfläche vereinfacht darzustellen.

Um auf einem Windows-Rechner Snort zu installieren, benötigen Sie das Snort-Distributionsarchiv sowie Bibliotheksdateien, die den direkten Zugriff auf die Netzwerkhardware ermöglichen. Der TCP/IP-Stack von Windows kann vom IDS nicht genutzt werden, da das IDS im sogenannten Promiscuous Mode alle Netzwerkpakete analysieren soll, die die Netzwerkkarte passieren. Zusätzlich sollen diese Pakete auch auf Netzwerkebene auf falsch gesetzte Flags, Fragmentierung und Ähnliches untersucht werden. Eine Bibliothek, die diese Funktionen zur Verfügung stellt, ist WinPCap (Windows-Packet-Capture).

Für Snort gibt es eine Reihe von grafischen Oberflächen, die sehr unterschiedlich im Leistungsumfang sind. An dieser Stelle ist ein großer Unterschied zu kommerziellen IDS/IPS zu finden, die meist sehr leistungsfähige grafische Oberflächen haben. Snort selbst, als Kommandozeilentool, ist jedoch nach wie vor auch für Windows-Systeme erhältlich (Version 2.9.13, April 2019).

Snort für Windows ist verfügbar unter <https://snort.en.lo4d.com/>. Den Download für die aktuellen Linuxversion finden Sie unter <https://www.snort.org/downloads>.

Kernstück des IDS sind die Konfiguration der sogenannten Präprozessoren, welche die von der Netzwerkkarte empfangenen Daten vorverarbeiten, und die Konfiguration der IDS-Regeln (rules).

Mittels der Präprozessoren ist Snort in der Lage, den empfangenen Datenstrom so zu verarbeiten, dass z. B. fragmentierte Pakete reassembliert werden.

Die eigentliche Intelligenz des IDS liegt in den Regeln, welche die Signaturen für bekannte Angriffe enthalten. In Snort liegen diese Signaturen in Textdateien vor, die von der Open-Source-Community aktualisiert werden und auch vom Administrator selbst erweitert werden können. Voraussetzungen hierfür sind genaue Kenntnisse der typischen Merkmale des Sicherheitsproblems und entsprechende Programmierkenntnisse des Administrators.

Weniger Aufwand dürfte es für Sie allerdings sein, die Snort-Regeln manuell oder automatisch von der Website www.snort.org zu beziehen. Snort wurde von SourceFire entwickelt, welche 2013 von der Firma Cisco übernommen wurde, um deren Security-Portfolio zu ergänzen.

15.6 Honeypot-Netzwerke

Wozu Honeypot-Netzwerke eingesetzt werden

Ein Honeypot (Honigtopf) ist eine Netzwerkkomponente oder ein ganzes Netzwerksegment, dessen Beitrag zur Netzwerksicherheit darin besteht, von Hackern gescannt, angegriffen und übernommen zu werden. Das bedeutet, dass dieses System nur dem Zweck dient, das Ziel von Angriffen zu werden, und nicht, produktive Leistung für autorisierte Nutzer zu erbringen.

Einen Honeypot können Sie in einem Netzwerk als Ablenkung von wichtigen Produktivressourcen platzieren, sodass Sie bei einem Angriff auf den Honeypot gewarnt sind und Maßnahmen zum Schutz Ihrer Server ergreifen können. Honeypots bieten ebenfalls die Möglichkeit, Risiken für das Firmennetz genauer abzuschätzen. Alternativ können Sie Honeypots einsetzen, um zu lernen und zu verstehen, wie Hacker beim Knacken eines Systems vorgehen und welche Tools sie dazu einsetzen.

Um in beiden Anwendungsfällen darüber informiert zu werden, dass gerade ein Angriff stattfindet, und um wichtige Informationen während des Angriffs aufzeichnen zu können, benötigen Sie ein IDS und die Fähigkeit, den Datenverkehr am betreffenden System protokollieren zu können.

Honeypot-Typen

Einfache Honeypots sind beispielsweise Softwarelösungen, die nur auf eingehende Verbindungen warten und den dort auftretenden Datenverkehr protokollieren. Auf diese Weise lassen sich (z. B. TCP Port 80) Systemangriffe auffangen. Die so erlangten Informationen lassen sich über die Protokolldatei analysieren.

Komplexere Honeypots bestehen aus Software, die Serverdienste und/oder Betriebssysteme inklusive deren Schwachstellen emulieren kann.

Die meisten Interaktionsmöglichkeiten hat ein Hacker mit einem echten Computer und Betriebssystem. Da es aber selten vorkommt, dass eine gesamte Firma nur einen einzigen Rechner an das Internet anbindet, würde ein intelligenter Eindringling bei solch einer Konfiguration sofort Verdacht schöpfen. Im Realismus für den Hacker übertroffen wird der Honeypot nur noch, wenn nicht nur ein einzelner Rechner, sondern ein eigenes Netzwerk (Honeynet) mit dem Ziel eingerichtet wird, für einen Dateneinbruch zur Verfügung zu stehen und diesen genau beobachten zu können.

Ein Honeynet enthält beliebig viele Komponenten eines Netzwerks, wie es im Produktiveinsatz ebenfalls vorkommen kann. Teilweise werden auf den verwendeten Systemen auch gefälschte Nutzdaten eingespielt, um den Hacker davon zu überzeugen, dass er in einem echten Netzwerk die Kontrolle hat.

Das Ziel eines Honeypots ist es, nicht erkannt zu werden. Je länger die Täuschung eines Hackers aufrecht erhalten wird, desto mehr Informationen kann das System über seine Strategie und Methoden erfahren. Deshalb ist eines der wichtigsten Punkte der **Grad der Interaktivität (Interaction)** mit dem Hacker und seiner Ziele (Server/Client). In diesem Kontext klassifiziert man diese als Low-Interaction-Honeypots oder High-Interaction-Honeypots.

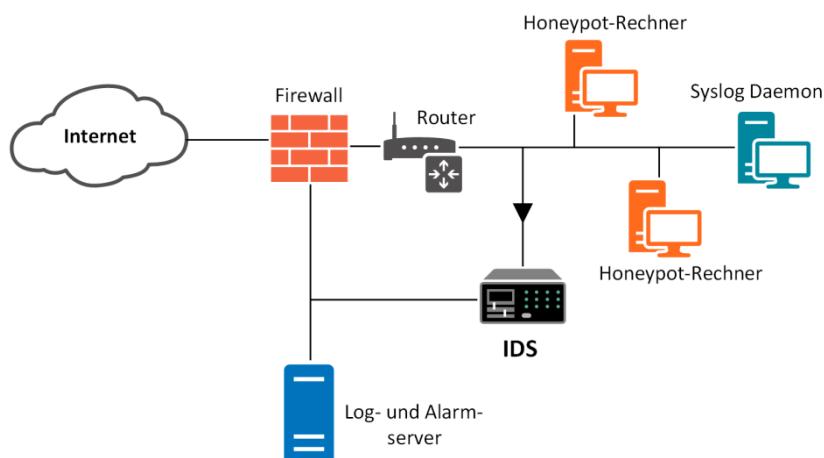
- ✓ **Low-Interaction-Honeypots (Server/Client):** Honeypots mit einem geringen Grad an Interaktivität basieren grundsätzlich auf der Abbildung realer Systeme oder/und Anwendungen. Dabei werden Betriebssysteme und Applikationen in der Regel nur in dem Maße simuliert, dass ein Angriff möglich ist.
- ✓ **High-Interaction-Honeypots (Server/Client):** Bei Honeypots mit einem hohen Grad an Interaktivität handelt es sich im Normalfall um reale Systeme, die Server-Dienste anbieten. Dabei werden alle Dienstmerkmale abgebildet, wie sie auch in der produktiven Umgebung vorkommen.

Risiken von Honeypots

Obwohl mit Honeypots nützliche Informationen über die Gegenspieler der IT-Sicherheit gewonnen werden können, sollten Sie sich darüber im Klaren sein, dass der Einsatz von Honeypots mit zunehmender Komplexität auch ein höheres Risiko mit sich bringt. Es muss bei einem Honeynet sichergestellt werden können, dass auch nach erfolgreicher Übernahme durch Hacker dieses System nicht zu weiteren Angriffen auf andere Systeme eingesetzt werden kann. Dies setzt die strenge Überwachung eines Honeynets durch den Administrator voraus, der nötigenfalls das Honeynet vom Internet abtrennt und den unkomprimierten Zustand wiederherstellt.

Beispielhafter Aufbau eines Honeynets

Ein einfaches Honeynet könnte wie in der folgenden Grafik aufgebaut sein.



Ein Honeynet

Das Netzwerk wird vom Internet zunächst durch eine Firewall getrennt, die alle eingehenden Verbindungen erlaubt, aber ausgehende Verbindungen verhindert. So kann zum Beispiel vermieden werden, dass ein kompromittiertes Honeynet für Angriffe auf andere Systeme ausgenutzt wird. Die Firewall hat eine Verbindung zum Honeynet und eine weitere, von außen nicht einsehbare Verbindung zum administrativen Netz.

Im Honeynet befindet sich direkt hinter der Firewall noch ein Router. Dieser ist dort einerseits platziert, damit ein Hacker einen Router vorfindet, wenn er sich im Netzwerk befindet und vorhandene Komponenten inspiziert. Da Router normalerweise als Internet-Gateways eingesetzt werden, lässt dies das Netzwerk für den Hacker weniger verdächtig erscheinen. Andererseits verhindert der Router, dass der Hacker die Firewall bemerkkt.

Im Honeynet selbst werden dann die Honeypot-Rechner platziert, zu denen auch ein vermeintlicher Syslog-Server gehören kann, der im vermeintlichen Netzwerk die Ereignisse protokolliert. Der eigentliche Log- und Alarmserver wird außerhalb des Honeypot-Netzwerkes geschützt hinter der Firewall in einem separaten Netzwerk platziert.

Das IDS zur Überwachung des Honeynets ist über eine Read-Only-Verbindung an das Honeynet angeschlossen und hat die Aufgabe, sämtliche über das Netz laufende Daten zu analysieren und über das Management-Interface in das administrative Netz zur Speicherung weiterzuleiten.

Mithilfe dieser Konstellation können Angriffe während ihres Ablaufs live beobachtet werden. Findige Hacker versuchen meist, den vorhandenen Syslog-Server zu deaktivieren oder dessen Datenbestand zu manipulieren. Dies ist jedoch völlig unproblematisch, da der relevante Datenbestand im administrativen Netz gespeichert wird und die Informationen auf dem Honeypot-Syslog-Rechner deswegen nicht zwingend gebraucht werden. Schafft der Hacker es allerdings, den Syslog-Server zu manipulieren, so liefern die protokollierten Daten, genau wie bei anderen Eingriffen in das Netzwerk, wertvolle Hinweise darauf, wie die Manipulation erfolgreich sein konnte.

Mit Honeynets ermittelte Statistiken

Im Rahmen des „Honeynet-Project“ wurden bzw. werden Angriffe auf verschiedene Rechner- und Netzwerkkonstellationen analysiert.

Ein Beispiel: Das Honeynet verfügte über acht IP-Adressen, die nicht veröffentlicht wurden. Es kann also davon ausgegangen werden, dass die Angreifer das Vorhandensein des Netzwerkes bzw. der Rechner durch Scanning des IP-Bereichs gefunden haben. Die wesentliche Erkenntnis dieses Projektes war, dass im Prinzip sämtliche an das Internet angeschlossenen Systeme nach relativ kurzer Zeit einem Einbruchsversuch zum Opfer gefallen sind. Bei einem frisch installierten System vergingen bestenfalls Stunden oder wenige Tage, bis dieses erfolgreich gehackt wurde.

Da heutzutage die kriminelle Szene professioneller und mit entsprechendem finanziellem Background (Geldverdienen durch Vermietung gekaperter Rechner an Spammer) arbeitet, können Sie davon ausgehen, dass die Anzahl an Scan- und Einbruchsversuchen heute wesentlich höher liegt als zum Zeitpunkt dieser Studie. Einen Rechner ohne Schutz oder aktuelle Sicherheitsupdates an ein öffentliches Netzwerk anzuschließen, ist angesichts dieses Risikos schon grob fahrlässig.

Weitere Informationen zu aktuellen Studien etc. entnehmen Sie bitte der „The Honeynet Project“-Website unter <https://www.projecthoneypot.org>.

15.7 Übung

Fragen zum Intrusion-Detection-System

Übungsdatei: --

Ergebnisdatei: uebung15.pdf

1. Worin besteht der Unterschied zwischen IDS und IPS?

16

Virtual Private Network

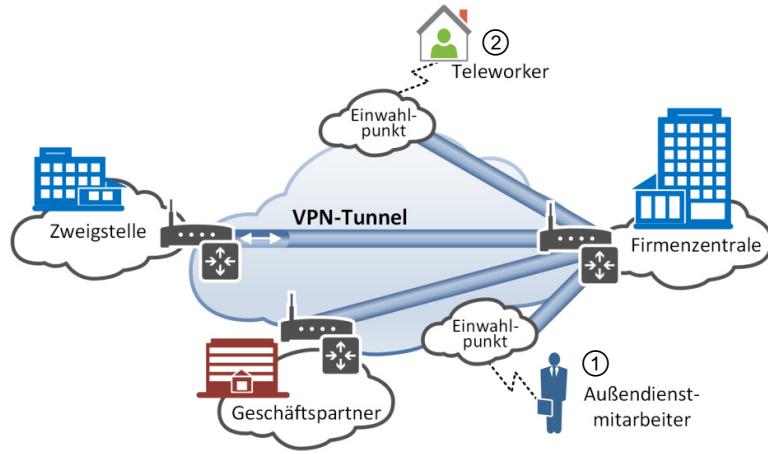
16.1 Zielsetzung

Definition und Gründe für ein Virtual Private Network

In Unternehmen, bei denen Zweigstellen, externe Firmen oder Außendienstmitarbeiter Zugang zu Ressourcen in der Firmenzentrale benötigen, mussten früher ein Einwahlserver (Remote Access Server, RAS) und eine angemessene Anzahl an Modems bzw. ISDN-Geräten installiert werden. Für eine Datenverbindung zwischen Firmensitz und Zweigstelle, die mehr Bandbreite benötigte, war unter Umständen sogar das Einrichten einer permanenten oder semipermanenten Standleitung nötig.

Der Remote-Zugang zu Firmenressourcen war auf diese Weise allerdings nicht nur in der Anschaffung der Hardware, sondern auch bei den laufenden Kosten relativ hoch. So mussten früher bei der Einwahl von Außendienstmitarbeitern die entsprechend anfallenden Telefongebühren bezahlt werden.

Die Idee eines **Virtual Private Network** (VPN) besteht nun darin, den lokal günstig zu nutzenden Internetzugang dazu zu gebrauchen, um den Kontakt zur Firmenzentrale herzustellen. Ein Außendienstmitarbeiter ① könnte also über die Einwahl bei einem Internetprovider über das Internet Zugang zu Ressourcen in der Firmenzentrale bekommen, oder ein Teleworker ② könnte per Internet-Flatrate angebunden sein.

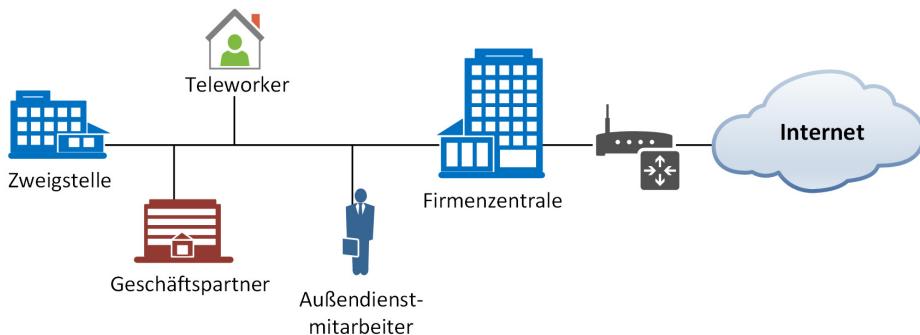


VPN, reelle Verbindungen über Internet

Da das Internet jedoch ein öffentliches Netz ist und so theoretisch jeder die übertragenen Daten an den Knotenpunkten abfangen oder sogar manipulieren könnte, mussten Möglichkeiten gefunden werden, wie dies verhindert werden kann. Hier bieten sich die Methoden der Kryptografie als Lösung an.

Mit einer **VPN-Verbindung** werden die Datenpakete des Außendienst-Rechners kryptografisch gesichert über das Internet in das Firmennetz weitergeleitet. Umgekehrt werden Datenpakete aus dem Firmennetzwerk über die gesicherte Verbindung an den Client weitergeleitet, wenn es sich um Broadcasts oder direkt an den Client adressierte Pakete handelt.

Aus der Sicht des Clients sieht es also so aus, als würde er sich im LAN der Firmenzentrale befinden.



VPN, logisches Netzwerk aus Sicht der Teilnehmer

Definition

VPNs realisieren eine Verbindung, welche in jeder Schicht eines Referenzmodells erfolgen kann. Die Kommunikation erfolgt dabei über **öffentliche** oder **private** Leitungen mittels **sicherer Protokolle**. Der Verkehr wird dabei **verschlüsselt** und der Aufbau der Verbindung **authentifiziert**. Dafür kommen u. a. die Standards RFC 2685 und RFC 2764 zur Anwendung. Man unterscheidet zwischen:

- ✓ **Intranet-VPN (site-to-site):** Dieser VPN-Typ verbindet entfernte Firmenlokationen über eine öffentliche oder private Netzwerkinfrastruktur.
- ✓ **Extranet-VPN (site-to-site oder remote access):** Es ermöglicht eine Anbindung von Partnern am Intranetzugang der Firma. Eine Verbindung erfolgt über die öffentliche Infrastruktur.
- ✓ **Remote-Access-VPN:** Außerhalb des Intranets befindliche Nutzer wählen sich über öffentliche Verbindungen (z. B. xDSL, GSM, UMTS, LTE, WiMax, 5G) in das Firmennetz ein.

Ein site-to-site-VPN kann von jedem VPN-Endsystem initiiert werden, ein remote-access-VPN wird nur vom Remote-VPN gestartet. Die Anwendung von VPNs ist sehr flexibel durch die Nutzung unterschiedlicher Sicherheitsprotokolle, wobei PPTP, L2TP/IPsec, IPsec und SSL/TLS zum Einsatz kommen.

16.2 PPTP

Aufbau von PPTP

Das Point-to-Point Tunneling Protocol ist eine hauptsächlich von Microsoft entwickelte Erweiterung des Point-to-Point Protocol (PPP). Für die Zugangskontrolle sind beim PPTP mehrere Verfahren implementiert. Dazu gehören die Optionen einer simplen, ungeschützten Passwortabfrage (PAP), eines verschlüsselten Challenge-Response-Verfahrens (CHAP) sowie einer Authentifizierung des Benutzers durch eine Smartcard (z. B. mit EAP).

Die zu übertragenden Daten selbst werden entweder per Microsoft Point-to-Point-Encryption (MPPE) mit der Stromchiffre RC4 verschlüsselt, wobei die Länge des Schlüssels zwischen 40 und 128 Bit variieren kann.

Zur Steuerung der Verbindung benutzt PPTP eine sogenannte Kontrollverbindung (PPTP Control Connection). Über diese Kontrollverbindung auf TCP Port 1723 werden der Aufbau, die Verwaltung und der Abbau der Tunnelverbindung mit den eigentlichen Daten bewerkstelligt.

Die zu tunnelnden Daten werden nach der Verschlüsselung in Pakete nach GREv2 (Generic Routing Encapsulation) gepackt und an die Zieladresse (den VPN-Server) geschickt. GRE trägt die IP-Protokollnummer 47.

Da die Userdaten bei PPTP per GRE übertragen werden und dieses Protokoll anders als TCP oder UDP keine Portnummern hat, stellen NAT/PAT-Gateways im Kommunikationspfad Problemquellen dar.

Die in PPTP verwendete Kryptografie hat bei der Kryptoanalyse so viele Mängel gezeigt, dass vom Einsatz von PPTP abgeraten wird.

16.3 L2TP/IPsec

Layer 2 Tunneling Protocol

Das L2TP ist eine Weiterentwicklung des ursprünglich von Cisco entwickelten L2F (Layer 2 Forwarding). Die Angabe einer Tunnel-ID im L2TP-Header ermöglicht es, mehrere Tunnel auf einer Verbindungsstrecke zu betreiben. NAT wird ebenfalls unterstützt. Obwohl L2TP eine Authentisierung mithilfe von CHAP oder PAP erlaubt, werden die Datenpakete der bestehenden Verbindung später nicht weiter überprüft. L2TP verschlüsselt die zu übertragenden Daten nicht. Dies erfolgt auf Layer 3 mittels IPsec (RFC 3193).

L2TP-Header

Ein L2TP-Header ist wie folgt aufgebaut:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
T	L	0	S	0	O	P	0	0	Version	Length																					
Tunnel ID																Session ID															
Sequence Number sent																Sequence Number expected															
Offset Size																Offset Pad :::															
Data																:::															

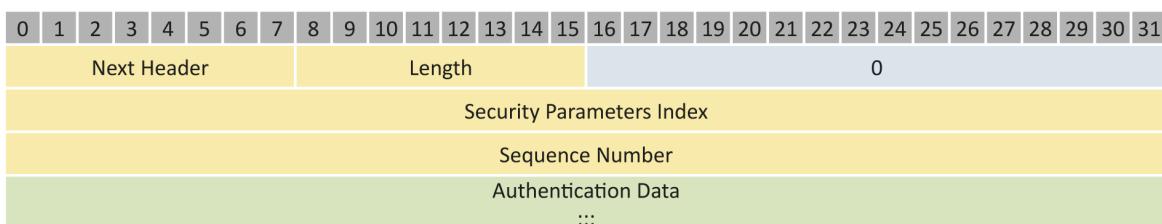
Abkürzungen: T = Message Type, L = Length present, S = Sequence present, O = Offset present, P = Priority

IP Security

Das IPsec-Protokoll war ursprünglich zur Verwendung im IPv6-Standard vorgesehen. Es wurde letztendlich jedoch auch für das bestehende IPv4 implementiert und kann im Zusammenspiel mit L2TP dieses sinnvoll um Verschlüsselung und paketweise Authentifizierung ergänzen. Weiterhin hat sich IPsec als Standard bei LAN-to-LAN-VPNs und auch bei Remote-Access-VPNs durchgesetzt (s. a. Abschnitt 12.3).

Authentication Header

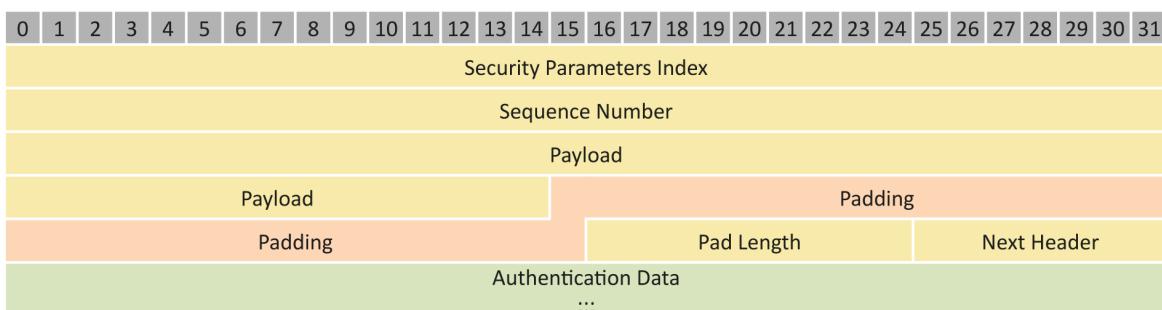
Zum Schutz der Daten vor Veränderungen ist in IPsec der sogenannte AH (Authentication Header) definiert. Dieser garantiert durch einen Hashwert (Prüfsumme) die Korrektheit der übertragenen Daten sowie des Headers. Als Hashwert-Protokoll finden MD5, SHA1, SHA2 und SHA3 Anwendung, wobei die ersten beiden Protokolle als sicherheitskritisch eingestuft sind.



ESP-Header

Die Vertraulichkeit wird bei IPsec mit ESP-Header (Encapsulating Security Payload) umgesetzt. Bei ESP war als kleinster gemeinsamer Verschlüsselungsstandard DES gefordert. Da DES schon länger keine ausreichende Sicherheit mehr liefert, bieten alle am Markt befindlichen Lösungen die Verwendung von stärkeren Algorithmen an, wie 3DES oder den aktuellen Standard AES mit seinen Schlüssellängen von 128-, 192- oder 256-Bit.

Bei der Verwendung von ESP werden die Nutzdaten sowohl verschlüsselt als auch deren Integrität gesichert. ESP ist das Standardverfahren bei IPsec.

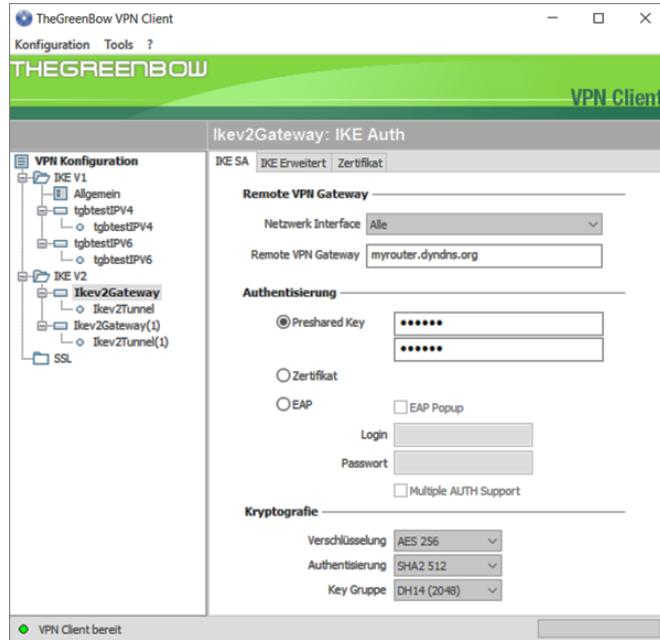


Security Parameters Index, SPI

Um den Inhalt von Paketen ordnungsgemäß authentifizieren oder entschlüsseln zu können, wird mit dem SPI angegeben, zu welcher **Security Association (SA)** dieses Paket gehört.

In einer SA wird in einem Computer für jedes Ziel (d. h. jede Ziel-IP-Adresse) festgelegt (assoziiert), welche Schlüssel und Sicherheitseinstellungen für dieses Ziel gelten sollen.

Dies kann zum Beispiel auch die erlaubten Algorithmen, die verwendeten Schlüssellängen oder die Zeitdauer bis zum Wechsel des Verschlüsselungsschlüssels beinhalten.



Auszug aus IPsec-Richtlinieneinstellungen

Schlüsselmanagement

Damit SAs gebildet werden können, müssen die Rechner in der Lage sein, sich gegenseitig zu identifizieren und einen gemeinsamen Verschlüsselungsschlüssel zu wählen. Dieser Vorgang wird unter IKE (Internet Key Exchange) zusammengefasst. Das Internet Security Association and Key Management Protocol (ISAKMP) ist dafür verantwortlich, über das unsichere Internet zunächst die Kommunikationspartner sicher zu identifizieren und dann mittels IKE eine SA inklusive kryptografischer Schlüssel zu erstellen.

Beim Erstellen der Schlüssel wird vom Diffie-Hellman-Protokoll und RSA Gebrauch gemacht. Zwei Kommunikationspartner können mittels Public-Key-Verfahren über Zertifikate gemäß X.509 authentifiziert werden. Alternativ können die Partner auch über einen Kerberos-Trust einen gemeinsamen geheimen Schlüssel erhalten. In diesem Fall ist das Kerberos-Key-Distribution-Center (KDC) der gemeinsame Dritte, dem beide Kommunikationspartner vertrauen müssen und der beide gegenseitig authentifiziert.

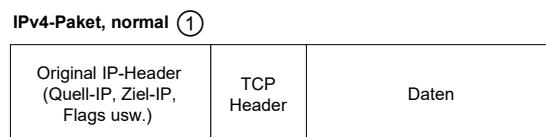
Als letzte Notfall-Möglichkeit kann eine Authentifizierung auch über sogenannte Pre-Shared-Keys vorgenommen werden. Hierzu stellt der Systemverwalter auf beiden Systemen, die miteinander über IPsec kommunizieren sollen, denselben Identifikationsschlüssel per Hand ein. Von dieser Methode ist jedoch verwaltungstechnisch abzuraten, da dies für große Netze einen immensen Aufwand bei der Verteilung und Erneuerung von Schlüsseln bedeuten würde. Für eine kleinere Anzahl von VPNs hat sich die Verwendung von Pre-Shared-Keys durchgesetzt, was bei der Verwendung von langen und komplexen Schlüsseln auch sicherheitstechnisch akzeptabel ist.

Transportmodus

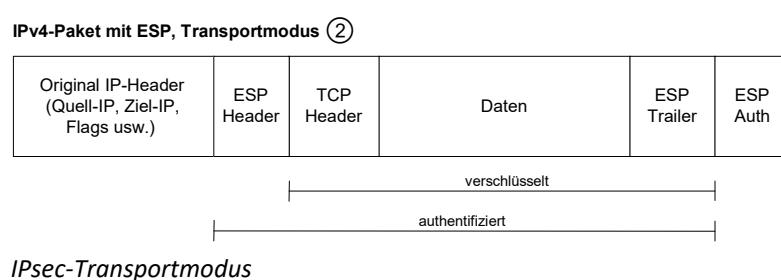
Soll ein einzelner Client eine Verbindung zu einem Server aufbauen, so kommt üblicherweise der sogenannte Transportmodus zur Anwendung. Dies ist einer der beiden Arbeitsmodi, die von IPsec unterstützt werden.

Im Transportmodus werden nur die zu übertragenden Daten durch IPsec verschlüsselt und geschützt. Der Original-IP-Header des Paketes bleibt erhalten.

In der Abbildung rechts sehen Sie oben ein IP-Paket ①, in dem die TCP-Daten einer Verbindung transportiert werden.



Im Transportmodus ② wird der originale IP-Header dieses Paketes abgetrennt und bleibt erhalten. Vor dem TCP-Header und den Daten wird der ESP-Header eingefügt. Am Ende des neuen Paketes befinden sich der ESP-Trailer und die ESP Authentication.



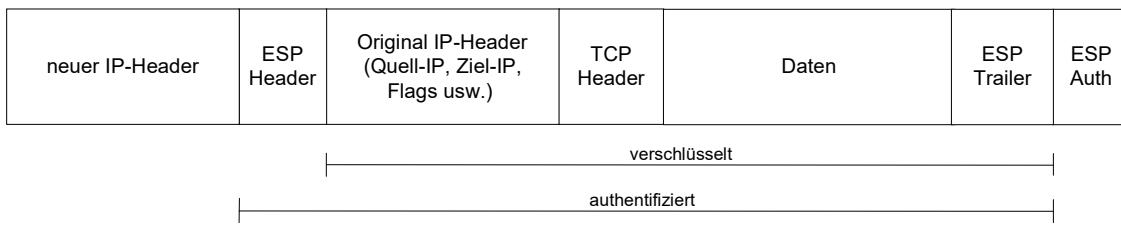
In diesem Fall ist der Bereich vom TCP-Header bis zum ESP-Trailer verschlüsselt, und der Bereich vom ESP-Header bis zum ESP-Trailer wird durch die Authentifizierung geschützt.

Obwohl für einen Lauscher die Verbindungsdaten selbst nicht mehr einsehbar sind, da sie im verschlüsselten TCP- bzw. Datenteil liegen, können durch Auslesen der Informationen im IP-Header zumindest die Absender- und die Empfänger-IP-Adresse ermittelt werden.

Tunnelmodus

Der zweite Modus von IPsec, der Tunnelmodus, eignet sich besonders, wenn zwei LANs über ein VPN gekoppelt werden sollen. Im Tunnelmodus werden an einem Gateway alle Pakete aus dem eigenen Netz, die für das jeweils andere Netz bestimmt sind, komplett verschlüsselt, signiert und neue IP-Pakete erzeugt, die an das Remote-Gateway geschickt werden. Dort werden die derart getunnelten Pakete wieder ausgepackt und den Empfängern im Remote-LAN zugestellt.

IPv4-Paket mit ESP, Tunnelmodus



Die Verschlüsselung umfasst in diesem Fall das komplette ursprüngliche IP-Paket vom Header bis zu den Daten. Die Authentifizierung reicht, ähnlich wie beim Transportmodus, wieder vom ESP-Header bis zum ESP-Trailer.

Auf diese Weise besteht ein vollständig geschützter VPN-Tunnel zwischen den beiden Gateway-Systemen. Der Datenverkehr, der über das öffentliche Netz abgewickelt wird, ist vollständig verschlüsselt. Ein Lauschangriff würde nur die IP-Adressen der beiden Gateway-Systeme offenbaren. Die Daten sowie die IP-Adressen der Clients, die innerhalb der beiden gekoppelten LANs miteinander kommunizieren, bleiben dem Angreifer wegen der Verschlüsselung verborgen.

Wenn IPsec mit Security-Gateways implementiert wird, kommt fast immer der Tunnelmodus zum Einsatz.

IPsec und Firewalls

Für den Schlüsseltausch nach IKE wird in IPsec der UDP-Port 500 benutzt. Falls Zertifikate ausgetauscht werden (beispielsweise über LDAP), so werden auch die für den Zertifikatstausch benötigten Portnummern benutzt.

Für ESP bzw. AH nutzt das Internetprotokoll die Portnummern 50 bzw. 51. Um eine IPsec-Verbindung erfolgreich durch eine Firewall betreiben zu können, muss diese in der Lage sein, die genutzten IP-Protokolle weiterzuleiten, und zusätzlich müssen die benötigten Ports konfiguriert werden.

Das Vorhandensein einer NAT-Firewall auf der Strecke zwischen den beiden VPN-Partnern unterbindet den Einsatz von AH vollständig. Würden die übertragenen IP-Headerinformationen im Zuge der Network Address Translation von intern auf extern (oder umgekehrt) von der Firewall ersetzt, so würde dies die in AH enthaltene Signatur brechen und das Paket ungültig werden lassen. Abhilfe: für den Transportmodus statt AH ESP mit NAT-T (NAT-traversal) einsetzen.

Problematisch für den verantwortlichen Administrator eines Netzwerkes ist beim Einsatz von ESP die Verschlüsselung der TCP-Informationen, die somit nicht durch die Firewall einseh- oder veränderbar sind und damit ein erhebliches Sicherheitsrisiko darstellen, beispielsweise, wenn ein Nutzer aus dem Intranet (LAN) eigenmächtig einen Tunnel zu einem Remote-VPN-Server eines anderen Netzwerkes aufbaut. Mögliche Gegenmaßnahmen wären hier, an der Firewall den Aufbau verschlüsselter Verbindungen von Clients aus dem internen LAN Richtung Internet zu blocken oder Web-Security-Appliances einzusetzen.

Standardmäßig wird ein VPN bei einem site-to-site-Tunnel auf einem Router oder auf der Firewall des Unternehmens – entweder am Router oder in der DMZ der Firewall – terminiert. Darüber können die Pakete entschlüsselt, Headerinformationen ggf. modifiziert oder optional ein weiterer Tunnel zum Empfänger initiiert werden.

16.4 OpenVPN

OpenVPN basiert auf den Sicherheitsfunktionen von OpenSSL. Die Verwendung von SSL/TLS-basierten VPNs hat sich heute als zweite Technologie neben IPsec durchgesetzt, wobei manche Hersteller SSL/TLS-basierte VPNs nur für Remote-Access-VPNs verwenden.

OpenVPN ist unter der GNU GPL lizenziert und steht unter <https://openvpn.net/> zum Download für folgende Systeme zur Verfügung:

- ✓ Client: Windows, Linux, Android, macOS
- ✓ Server: Windows, Linux (RedHat, Fedora, CentOS, Ubuntu, Debian)
- ✓ Server: Virtual Appliance (VMware ESXi 5.0 und Microsoft Hyper-V)
- ✓ Server: Clouds (Amazon Cloud, Microsoft Azure, Google Cloud)

OpenVPN ist kompatibel mit den Protokollen SSL/TLS, RSA-Zertifikaten, X509-Zertifikaten, NAT, DHCP und TUN/TAP-Interfaces. Es unterstützt jedoch nicht die Protokolle IPsec, IKE, PPTP und L2TP.

16.5 Abgrenzung zu anderen VPN-Arten

Im Gegensatz dazu gibt es auch VPNs, die keinen kryptografischen Hintergrund haben. Bei diesen geht es hauptsächlich um die Separierung von Datenströmen durch den Provider, um z. B. verschiedene Kundendaten zu trennen. Das wichtigste Beispiel dafür ist **MPLS** (Multiprotocol Label Switching), bei dem die Daten standardmäßig ungesichert übertragen werden.

Ein **MPLS-VPN**, welcher zwischen dedizierten Standorten eingesetzt wird, ermöglicht es dem Kunden, private IP-Adressen (RFC 1918) für die Kommunikation zu nutzen. Da die Daten in einem MPLS-VPN jedoch nicht verschlüsselt werden (die Verschlüsselung muss explizit beim Provider hinzugebucht oder durch eigene Maßnahmen umgesetzt werden), sind die Daten, die den MPLS-VPN passieren, für potentielle Angreifer an jedem Netzknoten des Providers und an den Übertragungsstrecken lesbar. Das „Privat“ im Begriff VPN bezieht sich hier nicht auf Verschlüsselung und Geheimhaltung, sondern nur auf die sichere Authentifizierung.

16.6 Übung

Fragen zu VPN

Übungsdatei: --

Ergebnisdatei: uebung16.pdf

1. Welche Funktion hat ein VPN?
2. Welche VPN-Typen sind Ihnen bekannt?
3. Welche Verschlüsselungsverfahren kommen bei VPNs zum Einsatz?

17

WLAN und Sicherheit

17.1 WLAN-Arbeitsweise

WLAN-Konfigurationen

Wegen ihrer einfachen Installation, die es Anwendern erlaubt, ohne das Verlegen einer Kabel-Infrastruktur Computer mobil zu vernetzen, ist Wireless LAN neben Wired LAN die zweite Möglichkeit der Vernetzung in einem LAN. Über die interne WLAN-Netzwerkkarte oder einen WLAN-USB-Adapter können vorhandene Rechner WLAN nutzen. Mobile Rechner (Netbooks, Sub-Notebooks, Notebooks, Tablet-PC etc.) haben standardmäßig WLAN-Schnittstellen integriert. Dabei werden die folgenden WLAN-Standards vorrangig genutzt:

- ✓ IEEE 802.11b (1999), Wi-Fi 1, theoretische Datenübertragungsrate 11 Mbit/s, praktisch ca. 6 Mbit/s
- ✓ IEEE 802.11a (1999), Wi-Fi 2, theoretische Datenübertragungsrate 54 Mbit/s, praktisch ca. 20 Mbit/s
- ✓ IEEE 802.11g (2003), Wi-Fi 3, theoretische Datenübertragungsrate 54 Mbit/s, praktisch ca. 11–22 Mbit/s
- ✓ IEEE 802.11n (2009), Wi-Fi 4, theoretische Datenübertragungsrate 600 Mbit/s, praktisch ca. 72–120 Mbit/s
- ✓ IEEE 802.11ac Wave 1 (2013), theoretische Datenübertragungsrate 1,3 Gbit/s, praktisch ca. 300–700 Mbit/s
- ✓ IEEE 802.11ac Wave 2 (2015), Wi-Fi 5, theoretische Datenübertragungsrate 6,9 Gbit/s, praktisch ca. 0,86–3,47 Gbit/s
- ✓ IEEE 802.11ax (2019), Wi-Fi 6, theoretische Datenübertragungsrate bis 10 Gbit/s

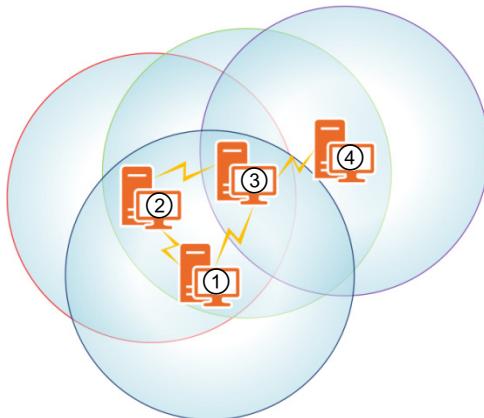
Grundsätzlich wird bei WLANs zwischen drei Konfigurationsmöglichkeiten unterschieden, mit denen der Datenaustausch organisiert wird:

- ✓ Independent (Ad-hoc-)WLAN
- ✓ Infrastructure WLAN
- ✓ Wireless Distribution System

Independent (unabhängige) WLANs arbeiten im Peer-to-Peer-Betrieb. Jeder Rechner mit einer WLAN-Karte ist in diesem Netzwerk gleichberechtigt und hat die Möglichkeit, Daten zu den jeweils anderen Rechnern zu senden.

Ein einzelner Rechner kann WLAN-Verbindungen zu den Partnern herstellen, die er „sieht“, d. h. in deren Funkreichweite er sich befindet. Verlässt ein Rechner diesen Bereich, kann er nicht mehr erreicht werden.

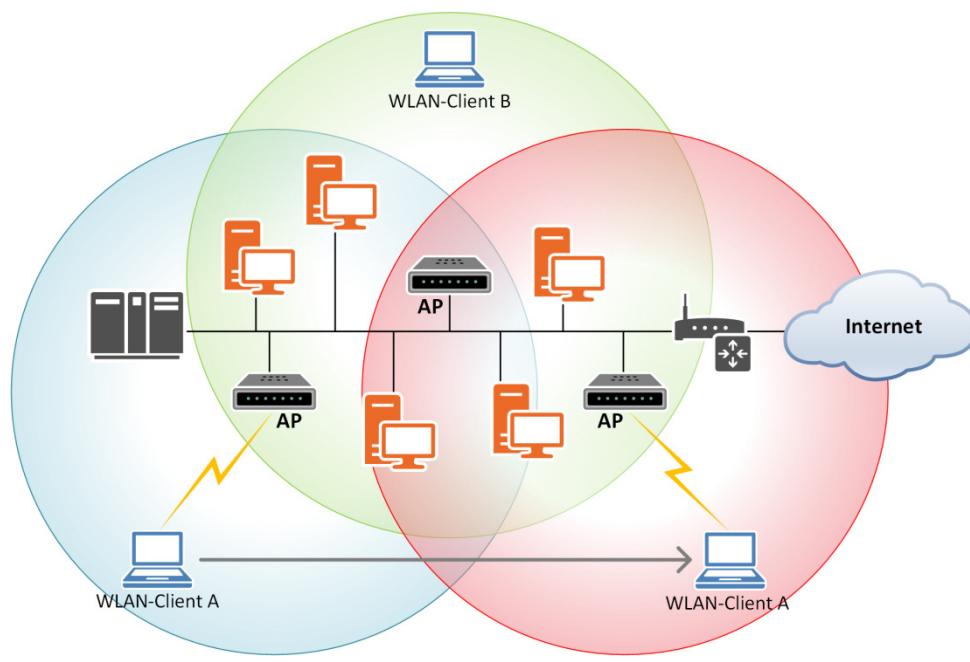
Die Abbildung rechts zeigt eine Situation, in der vier Rechner über Independent WLAN vernetzt wurden. Die Rechner ①, ② und ③ stehen nahe genug zusammen, um sich alle gegenseitig in Reichweite zu haben. Rechner ④ ist jedoch so weit entfernt, dass er nur noch zu einem der drei anderen Rechner eine Verbindung herstellen kann (Rechner ③).



Independent (Ad-hoc)-WLAN

Independent WLANs haben den Vorteil, wenig oder keine Konfiguration zu benötigen. Der Nachteil ist, dass ohne weiteren Aufwand kein Zugriff auf freigegebene Ressourcen möglich ist, die nicht auf WLAN-Rechnern liegen.

Infrastructure WLANs verfügen über eine weitere Komponente. Zusätzlich zu den WLAN-Karten in den Rechnern bilden Access-Points (AP) das Rückgrat des WLAN. Analog zu Mobilfunknetzen baut ein Access-Point eine Funkzelle um sich herum auf. Verbindungen der WLAN-Clients laufen über den Access-Point. Der Access-Point selbst besitzt zusätzlich ein kabelgebundenes Ethernet-Netzwerkinterface, mit dem er an das vorhandene LAN angeschlossen werden kann. Ein Access-Point realisiert eine Layer-2-Verbindung (Translation Bridge) zwischen den Protokollen IEEE 802.3 (Ethernet) und IEEE 802.11 (WLAN).



Infrastructure WLAN

Über den Access-Point sind die WLAN-Clients somit in der Lage, auf freigegebene Ressourcen im gesamten Netzwerk zuzugreifen.

In obiger Grafik sind in einem LAN drei Access-Points installiert, die alle zum gleichen Funknetzwerk gehören. Die in deren Reichweite befindlichen Clients A und B haben Zugriff auf alle Netzwerkressourcen. Über die Roaming-Funktion kann sich ein Client wie A auch von einer Funkzelle in die andere bewegen ① und behält die logische Verbindung zum Netzwerk bei. Die Funkverbindung besteht aber nun zum neuen Access-Point ②.

Die Installation von mehreren Access-Points bzw. WLAN-Repeatern an unterschiedlichen Standorten innerhalb eines großen Bereichs optimiert die Funkausleuchtung und stellt für die WLAN-Clients die Möglichkeit dar, sich nicht nur innerhalb der Funkzelle eines Access-Points zu bewegen, sondern sich während des mobilen Einsatzes von einer Funkzelle zur nächsten ohne Unterbrechung der Kommunikation zu bewegen (Roaming). Dabei hat der WLAN-Client aus logischer Sicht jederzeit Zugriff zum kompletten LAN.

WDS – Wireless Distribution System

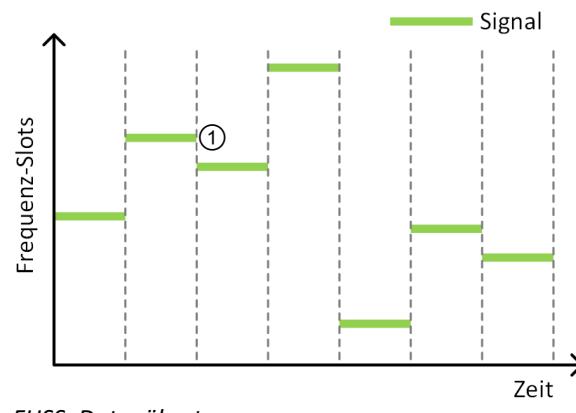
Werden WLAN-Access-Points mit WLAN-Bridges (Geräte, die zwei WLAN-Funkzellen miteinander verbinden) oder WLAN-Repeatern (Geräte, welche die Reichweite einer Funkzelle vergrößern) zu einem Funknetzwerk zusammengefasst, spricht man von WDS. Damit erreicht man eine größere Abdeckung als mit nur einem Zugangspunkt. Für die WLAN-Bridges bzw. WLAN-Repeater müssen dann nur noch entsprechende Stromanschlüsse vorhanden sein.

Werden die Access-Points nur direkt verbunden und dürfen sich an diesen keine weiteren Clients anmelden, spricht man vom **Bridging-Modus**. Ist auch die Anmeldung von Clients möglich, bezeichnet man das als **Repeating-Modus**. Alle WDS-Access-Points sollten auf dem gleichen Kanal arbeiten sowie dieselbe SSID (vgl. Abschnitt 18.2) und denselben WLAN-Schlüssel verwenden. Zudem müssen jedem Access-Point die WLAN-MAC-Adressen des anderen Access-Points bekannt sein. Sollen mehrere Access-Points zu einem WDS zusammengeschaltet werden, sollte darauf geachtet werden, dass sie nach Möglichkeit vom gleichen Hersteller stammen. Ansonsten kann es bei den neuesten WLAN-Standards (IEEE 802.11ac) ggf. zu Inkompatibilitäten führen. Gleichzeitig sollte auf den benötigten Funkstandard geachtet werden (z. B. IEEE 802.11g/h oder IEEE 802.11n).

Frequency Hopping und Direct Sequence

WLANS benutzen für den Standard IEEE 802.11 die Direct Sequence Spread Spectrum Technology (**DSSS**), um Daten über die Funkfrequenzen zu senden. Bei diesem im militärischen Bereich entwickelten Verfahren wird nicht nur eine, sondern werden mehrere Frequenzen für die Übertragung benutzt. Durch die Verteilung der Daten auf mehrere Frequenzen wird eine höhere Zuverlässigkeit und Abhörsicherheit garantiert als bei der Nutzung nur einer Frequenz.

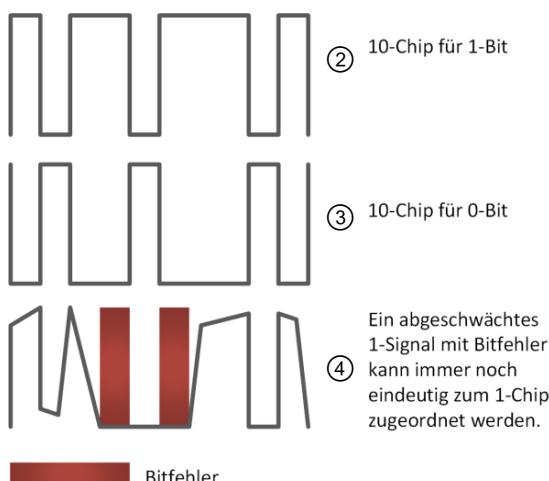
Beim Frequency Hopping Spread Spectrum (**FHSS**) wird ein Trägersignal verwendet, das seine Frequenz in einem vom Sender und Empfänger vorhersagbaren Muster ändert ①. Sind Sender und Empfänger synchron geschaltet, so verfügen beide über einen dauerhaften logischen Datenkanal. Ein zufälliger Lauscher, der sich beispielsweise mit einem Funkscanner auf eine Frequenz geschaltet hat, hört nur pulsierendes Rauschen von kurzer Dauer.



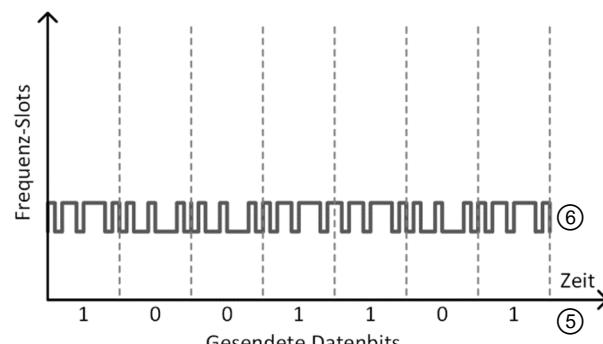
FHSS, Datenübertragung

Beim Direct Sequence Spread Spectrum wird ein mehrfach redundantes Bitmuster zur Übertragung eines 1-Bits gebildet ②. Dieses Bitmuster wird Chip genannt. Für ein 0-Bit wird der Chip des 1-Bits invertiert ③. Die Länge eines Chips entscheidet, wie viele Bitfehler bei der Übertragung vom Empfänger automatisch korrigiert ④ werden können, ohne dass der Chip erneut übertragen werden muss.

Sollen Datenbits übertragen werden, werden anstelle jedes Datenbits ⑤ die entsprechenden Chips in der korrekten Reihenfolge gesendet ⑥. Ein zufälliger Lauscher würde auf seinem Funkempfänger nur ein leises Rauschen hören, und auch dies wird von den meisten Funkscannern unterdrückt. Um die empfangenen Signale wieder korrekt in 1- und 0-Bits übersetzen zu können, muss der Empfänger wissen, wie der Sender den 1- bzw. 0-Chip gebildet hat.



DSSS



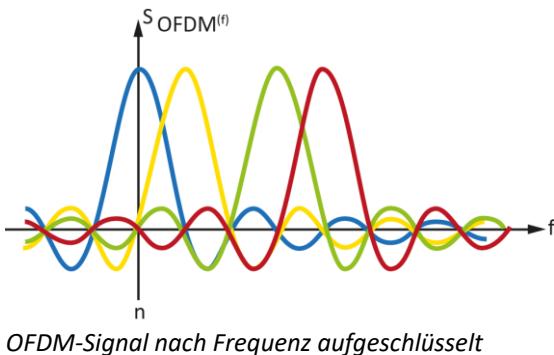
DSSS, Datenübertragung

Spread-Spectrum-Sicherheit

Die Spread-Spectrum-Verfahren dienen eher zum Schutz vor unabsichtlichem Mithören und Übertragungsfehlern als zum Schutz vor böswilligen Angreifern. Ein Hacker, der in ein WLAN einzudringen versucht, braucht sich mit FHSS- und DSSS-Problematiken nicht auseinanderzusetzen, da im Handel preiswerte WLAN-Karten verfügbar sind. Die in den Karten eingebauten Transceiver sind zwangsläufig zu den WLAN-Standards (IEEE 802.11b) kompatibel und daher in der Lage, mit anderen WLAN-Komponenten funktechnisch Kontakt aufzunehmen.

OFDM

Andere WLANs (Standard IEEE 802.11a/g/n/ac) verwenden das **OFDM-Verfahren (Orthogonal Frequency Division Multiplexing)**, um Informationen zu codieren. OFDM ist eine Technik, bei der mehrere Trägerfrequenzen gleichzeitig benutzt werden, um ein phasen- und amplitudenmoduliertes Signal mit mehreren Bits pro Symbol zu übertragen.



Die Nutzung mehrerer Trägerfrequenzen gleichzeitig hat den Vorteil, dass auf der Empfängerseite ein Signal trotz Echos und Funkreflexionen, wie sie im täglichen Betrieb aufgrund von Reflexionen und Streuung an Wänden auftreten, empfangen werden kann.

OFDM wird mit anderen Frequenzen und Trägeranzahlen auch in anderen Gebieten der Datenübertragung benutzt (z. B. DVB-T – Digital Video Broadcasting-Terrestrial, ADSL und DAB – Digital Audio Broadcasting). Für WLANs werden 52 Trägerkanäle gleichzeitig benutzt.

17.2 Access-Points

Standard-Installationen

Ein Access-Point ist ein Layer-2-Device, welches eine Verbindung zwischen einer wired (LAN) und einer wireless (WLAN) Netzwerkverbindung ermöglicht. Bei den Access-Points wird das Setup größtenteils durch Plug & Play erleichtert. So kann ein Systemadministrator ohne viel Aufwand neu zugekauftes WLAN-Equipment in sein vorhandenes Netzwerk einbinden. Problematisch wird es jedoch, wenn der herstellerseitig konfigurierte Service Set Identifier (SSID) des Access-Points nicht oder nicht ausreichend geändert wurde. Gleiches gilt auch für Zugangsdaten (Konfiguration des Access-Points über das Webinterface). Mitunter werden die Werkseinstellungen für Benutzernamen und Passwort nicht geändert, so kann keine ausreichende Sicherheit gewährleistet werden.

So können von Wardrivern auch heute noch unzureichend geschützte Firmen- und Privat-WLANs gefunden werden, die über die Hersteller-SSIDs und nicht vorhandene bzw. nicht sichere Passwörter den Zugriff auf das Firmennetzwerk ermöglichen. Auch eine Änderung der SSID auf beispielsweise den Firmen- oder Familiennamen ist nicht zu empfehlen, da dies den Hacker zusätzlich motivieren kann.

Access-Points antworten auf Broadcasts, um Clients das Auffinden von nahen APs zu ermöglichen. Dies kann jedoch auch von Hackern bzw. Wardrivern ausgenutzt werden, um nach installierten APs zu suchen. Schalten Sie die SSID-Broadcast-Funktion ab, wenn dies in Ihrer Hardware möglich ist. Damit sich ein Client mit dem gewünschten WLAN über den Access-Point verbinden kann, muss dem Client die SSID des entsprechenden WLANs bekannt sein. Um nun die SSID herauszufinden, können vorhandene Verbindungen abgehört werden. Das ist mit entsprechenden Tools auch kein Problem, damit wird allerdings gegen geltendes Recht verstößen.

Zudem kann das Abschalten der SSID das Hinzufügen neuer WLAN-Clients zum Netz behindern, sofern das Betriebssystem (z. B. macOS) keine Möglichkeit besitzt, den Beitritt zu einer nicht vorhandenen SSID zu ermöglichen. Insofern sollten Sie abwägen, ob das in Ihrem Unternehmen sinnvoll ist.

MAC-Filterung

Genauso wie Standardnetzwerkarten besitzen auch WLAN-Karten eine MAC-Adresse. Viele Access-Points unterstützen eine MAC-Filterliste und lassen nur Verbindungen mit WLAN-Karten zu, deren MAC-Adresse für den Zugriff konfiguriert ist. Benutzen Sie dieses Feature, behalten Sie aber im Auge, dass auch MAC-Adressen von entsprechend ausgerüsteten Hackern gefälscht werden können. Zudem erfordert das manuelle Eintragen und Bearbeiten einer größeren Anzahl von MAC-Adressen einen zusätzlichen Zeitaufwand und ist nicht vergleichbar dem in einem kleinen Heimnetzwerk.

Das Abschalten des SSID-Broadcast, MAC-Filterung, das Setzen einer Nicht-Default-SSID sowie eine strengere Authentifizierung der Nutzer sollten zumindest den Gelegenheits-Surfer davon abhalten, zufällig in Ihr Netzwerk zu stolpern und dort vertrauliche Daten einzusehen bzw. auf Ihre Kosten den Internetzugang Ihrer Firma zu nutzen. Einen wirklichen Sicherheitsgewinn stellen diese Maßnahmen aber nicht dar.

17.3 WEP – Wired Equivalency Protocol

WEP war das ursprünglich für WLANs verwendete Verschlüsselungsprotokoll. Wie der Name schon sagt, sollte es dafür sorgen, dass in WLANs eine Sicherheit erreicht wird, die der von konventionellen drahtgebundenen Netzen entspricht.

Bevor die amerikanische Regierung die Krypto-Export-Regulierungen lockerte, wurde WEP mit einer Schlüssellänge von 40 Bits implementiert. Seit der Liberalisierung dieser Regulierung finden sich zunehmend Produkte mit 128-Bit-Unterstützung für WEP. Ein Teil dieser 128 Bit ist der Initialisierungsvektor (IV) und fällt somit aus dem möglichen Schlüsselraum heraus. Dies führt bei der Länge des IV von 24 Bit zu einer Schlüssellänge für die Verschlüsselung von 104 Bit.

Shared Secret Keys

WEP arbeitet mit Shared Secret Keys. Das bedeutet für den Administrator, dass auf jedem WLAN-kompatiblen Gerät bzw. für jede WLAN-Karte dieselbe WEP-Schlüssel installiert werden muss. Geräte mit unterschiedlichen Schlüsseln können nicht miteinander kommunizieren.

Allein der Aufwand für diese Art der manuellen Schlüsselgenerierung (einmalig) und Verteilung (pro WLAN-Karte) wäre bei größeren Netzwerken immens.

WEP bereits geknackt

Ein Beweis, dass WEP als nicht ausreichend sicher angesehen werden muss, wurde im Sommer 2001 erbracht. WEP, das zur Verschlüsselung der Daten selbst die Stromchiffre RC4 benutzt, generiert für diese Stromchiffre Verschlüsselungsschlüssel, die auf dem installierten Shared Secret Key basieren.

Die Analyse zeigte, dass diese generierten WEP-Schlüssel einander zu ähnlich waren, sodass ein Angreifer durch rein passives Mitlauschen eine bestimmte Menge an verschlüsselten Datenpaketen speichern konnte, um aus den gesammelten Informationen Rückschlüsse auf den verwendeten Shared Secret Key zu ziehen.

In WEP kommen mehrere kryptografische Probleme zusammen: Zum einen die relativ kurzen Initialisierungsvektoren (IV) mit 24 Bits. Diese machen es möglich, dass ein Hacker statistisch gesehen schon nach ca. 5000 abgefangenen Datenpaketen eine Chance von 50 % hat, dass zwei verschiedene Pakete mit demselben IV und somit mit demselben Gesamtschlüssel verschlüsselt wurden. Werden Pakete mit demselben Schlüssel verschlüsselt, so kann man die Verschlüsselung aus den Paketen mathematisch gesehen komplett herauskürzen.

Dazu kommen noch die statischen Verschlüsselungsschlüssel und eine kryptografische Schwäche im verwendeten RC4-Verschlüsselungsalgorithmus, die WEP insgesamt sehr leicht angreifbar macht.

Im Internet existieren Tools (z. B. Airsnort, WEPCrack), die diese Berechnungen automatisieren und dem Hacker die mühsame Filterung und Berechnung des Secret Keys abnehmen.



Von der Verwendung eines nur durch WEP geschützten WLANs ist also abzuraten. Obwohl die Aktivierung sicher nicht schadet und definitiv geeignet ist, unambitionierte Angriffe abzuwenden, sollten Sie sich nicht in der Sicherheit wiegen, dass die über das Netz laufenden Daten wirklich sicher sind.

17.4 WPA – Wi-Fi Protected Access

Abhilfe für die größten Probleme

Nachdem die gravierenden Probleme von WEP bekannt wurden, wurde fieberhaft an besseren Sicherheitsprotokollen gearbeitet. Dabei wurde allerdings auch klar, dass diese neuen Standards nicht in Kürze verfügbar sein würden.

Zusätzlich war abzusehen, dass man in einem neuen Standard nicht einfach einen neuen Verschlüsselungsalgorithmus vorschreiben kann, da in der bereits verkauften und im Einsatz befindlichen Hardware der verwendete RC4-Algorithmus hardwaremäßig implementiert war.

Als Interimslösung wurde deswegen Ende 2002 der Standard WPA verabschiedet, der die wichtigsten Änderungen des endgültigen Sicherheitsstandards 802.11i vorwegnehmen sollte. Und zwar in einer Form, in der er auch auf bereits verkaufter WEP-kompatibler Hardware lief.

Wireless Protected Access führt einen auf RC4 basierten neuen Algorithmus zur Verschlüsselung ein, das Temporal Key Integrity Protocol (**TKIP**).

Als eine der wichtigsten Verbesserungen in WPA ist die Tatsache anzusehen, dass das festgelegte Passwort in WPA nicht mehr der Verschlüsselungsschlüssel selbst ist, sondern die Schlüssel mit kryptografisch gesicherten Methoden hergeleitet und regelmäßig erneuert werden. Dieser Modus wird als WPA-PSK bezeichnet. Die Schlüssel werden automatisch in einem Zeitintervall erneuert, in dem es mit üblichen Methoden nicht mehr möglich erscheint, diese Schlüssel auch zu knacken. Sollte WPA zusammen mit einem Preshared Key (PSK) betrieben werden, besteht die Gefahr, dass man durch einen schlecht gewählten oder einen einfach zu erratenen PSK den Sicherheitsgewinn von WPA wieder zunichtemacht.

Zusätzlich sieht WPA auch die Unterstützung von RADIUS-Servern (Remote Authentication Dial-In User Service) zur Authentifizierung der Funkteilnehmer vor. Dieser Modus wird WPA-Enterprise genannt.

WPA-PSK ist über Offline-Tools angreifbar. Dazu werden die Pakete während der WPA-Authentifizierungsphase aufgezeichnet. Aus diesen versucht das Tool entweder per Brute-Force-(Durchprobieren aller möglichen Kombinationen) oder mittels einer Wörterbuch-Attacke den während der Authentifizierung ausgetauschten Pairwise Master Key (PMK) zu ermitteln.

17.5 WPA2 – Wi-Fi Protected Access 2

Im Juni 2004 wurde der aktuelle WLAN-Sicherheitsstandard mit der Bezeichnung IEEE 802.11i verabschiedet, der nun von einigen Herstellern als WPA2 bezeichnet wird. Die wesentlichen Kernpunkte, die im WPA schon vorweggenommen wurden (wie z. B. regelmäßige automatische Erneuerung der verwendeten Verschlüsselungsschlüssel) blieben erhalten. Gleichzeitig wurde der moderne Standardalgorithmus AES (Advanced Encryption Standard) verbindlich vorgeschrieben.

Ein Funknetz, das mit 802.11i-Verschlüsselung betrieben wird, kann als wesentlich sicherer angesehen werden als eine WAP- oder sogar eine WEP-Verbindung. Allerdings sollten Sie dabei beachten, dass bei Absicherung Ihres Netzes mit Preshared Keys Ihre Sicherheit davon abhängt, wie leicht oder schwer die verwendeten Preshared Keys (also die Passwörter) zu erraten sind.

17.6 WPA3 – Wi-Fi Protected Access 3

Auch bei WPA2 wurden inzwischen diverser Sicherheitsmängel gefunden. Durch einen Designfehler im Protokoll ist es einem Angreifer möglich, einen Wörterbuchangriff auf schwache Passwörter (Offline-Passwort-Angriff) erfolgreich durchzuführen. Ein weiterer Aspekt ist die Unterstützung veralteter Verschlüsselungs- und Hashverfahren.

Mit WPA3 hat die Wi-Fi-Alliance 2018 einen Nachfolgestandard veröffentlicht. Dieser setzt auf eine verbesserte und sicherere Authentifizierung und Verschlüsselung. Der Schlüsselaustausch beruht nun auf dem Diffie-Hellmann-Algorithmus, wobei Perfect Forward Secret (PFS) genutzt wird. PFS ist ein Verfahren für den Schlüsselaustausch, das die nachträgliche Entschlüsselung durch Bekanntwerden des Hauptschlüssels verhindert. Es werden Teile des Schlüssels ausgetauscht, aus denen die Partner den vollständigen Schlüssel selbst berechnen. Zusätzlich hat der Sitzungsschlüssel nur eine definierte Gültigkeit. Nach Ablauf dieser Zeit startet PFS die Aushandlung eines neuen Diffie-Hellman-Prozesses. Zeitlich folgende Sitzungsschlüssel haben auch keinen Bezug untereinander und sind nicht gegenseitig ableitbar. Durch die Kenntnis eines einzelnen Sitzungsschlüssels ist kein Folgeschlüssel zu ermitteln.

Ein Software-Update auf WPA3 ist auf aktuellen Access-Points möglich – zumindest theoretisch. Ob Hersteller tatsächlich von dieser Möglichkeit Gebrauch machen oder dies nur auf neuen Geräten anbieten, liegt leider in deren Ermessensspielraum.

17.7 Weitere Authentifizierung und Verschlüsselung im WLAN

Neben den eben angesprochenen Verschlüsselungsverfahren können Sie bei WLAN alternativ auch IPsec nutzen. Dazu muss auf dem Client eine IPsec-Software installiert werden. Dies ermöglicht nun eine authentifizierte und verschlüsselte Verbindung über den Access-Point hinaus zum Router bzw. Zielserver.

IPsec ist bei fast allen Server-Betriebssystemen (bei Microsoft ab Windows 2000, Linux, Unix) schon implementiert und ermöglicht eine gesicherte End-to-End-Kommunikation.

Im WLAN können sich die Clients beim Router oder Zielsystem identifizieren, indem sie vor ihrem Einsatz ein X.509-Zertifikat erhalten. So wird über einen Public Key die Identität eines bestimmten Rechners eindeutig nachgewiesen. Ein Sitzungsschlüssel wird durch Diffie-Hellman zwischen den Endpunkten der Verbindung dynamisch ermittelt und automatisch nach dem Transfer einer bestimmten Datenmenge oder dem Ablauf einer maximalen Zeitspanne ausgetauscht. Somit wäre also auch das Problem des manuellen Schlüsseltauschs beseitigt.

Da IPsec nicht auf einen bestimmten Verschlüsselungsalgorithmus festgelegt ist, können die nach dem Schlüsseltausch ermittelten Schlüssel auf beliebig auswählbare Protokolle angewandt werden (z. B. 3DES oder AES).

Die Identifizierung über Zertifikate hat überdies auch den Vorteil, dass ein Hacker keine Man-in-the-Middle-Attacke benutzen kann, um Zugang zum Netz zu bekommen. Versucht der Hacker, sich gegenüber dem Client als Router/Zielsystem auszugeben, so kann er sich über IPsec nicht ordnungsgemäß ausweisen, da er den Private Key des Zielsystems nicht besitzt und somit dessen digitale Unterschrift nicht fälschen kann.

Radius-Authentifizierung (IEEE 802.1X)

Vor allem in Unternehmensnetzwerken, wo viele Benutzer und bei entsprechend großem Gelände auch viele Access-Points die WLAN-Umgebung darstellen, ist es nicht praktikabel, mit einer Authentifizierung auf Basis eines Preshared Keys zu arbeiten. Hier bietet es sich an, sämtliche Access-Points als sogenannte Radius-Clients (Authenticator) mit einem Radius-Server – von dem mindestens einer im Firmennetzwerk stehen sollte – zu verknüpfen. Die Access-Points werden daraufhin die Anmeldeanfragen von mobilen Benutzern an den zentralen Radius-Server weiterreichen, auf dem Sie als Administrator die Einwahlbedingungen, -zeiten oder berechtigte Benutzergruppen definieren können. Ein Radius-Server überprüft die Authentifizierung von Anmeldedaten beispielsweise anhand von Zertifikaten, die vorher ausgestellt werden müssen, oder mithilfe der Domänen-Anmeldedaten des jeweiligen Benutzers. Sie haben also nicht mehr ein einziges Passwort als „Generalschlüssel“ für das WLAN, sondern jeder WLAN-Benutzer meldet sich individuell an und kann auch individuell über die Konfiguration der Radius-Regeln Einwahlrechte erteilt oder verweigert bekommen.

Wi-Fi Protected Setup (WPS)

Sofern Ihre WLAN-Komponenten **WPS** unterstützen, kann der von der Wi-Fi Alliance initiierte Standard zur Anwendung kommen. Er ermöglicht das einfache Hinzufügen von Clients in einer gesicherten Umgebung. Die häufigsten Herstellerimplementierungen sind entweder die Eingabe einer vorgegebenen Client-PIN am Access-Point bzw. die Nutzung eines Push-Buttons. Durch das Drücken des Buttons am Client (sofern vorhanden) und am Access-Point wird der automatische Schlüsselaustausch initiiert und die verschlüsselte Verbindung hergestellt.

Beachten Sie jedoch, dass während der Authentifizierungsphase über WPS der Verkehr abgehört werden kann und der Zugriff durch Dritte über Angriffsvektoren möglich ist.

17.8 Funkausleuchtung

Reichweite von Access-Points

Oft wird bei der Installation eines Access-Points vergessen, welche Reichweite diese innerhalb und außerhalb von Gebäuden besitzen. Übliche Größenordnungen bewegen sich zwischen 20 Metern (indoor) und 250 Metern (outdoor).

Dies kann durchaus dazu führen, dass ein WLAN-Empfang auch noch jenseits der Gebäude- oder Grundstücksgrenzen möglich ist, obwohl dies vom Besitzer des WLANs nicht beabsichtigt war. Die überschüssige Reichweite erzeugt somit ein unnötiges Sicherheitsproblem.

! Unterschätzen Sie die Reichweite von WLAN-Equipment nicht. Mit handelsüblichen Antennen können Eindringlinge auch noch aus 2–3 km Entfernung versuchen, ein Netzwerk auszuspähen. Der Rekord für eine Langstrecken-WLAN-Verbindung auf der Basis des Standards 802.11b (11 Mbit, 2.4 GHz) liegt bei ca. 200 km.

Auch wenn Sie möglicherweise mit Ihrem eigenen Notebook (und der darin vermutlich verbauten 2-3dB-WLAN-Antenne) auf dem Parkplatz außerhalb des Firmengebäudes schon keinen Empfang mehr haben, bedeutet dies nicht zwangsläufig, dass ein Angreifer, der in Ihr Netzwerk eindringen will, sich keine 31dB-Parabolantenne für seinen Einbruch besorgen kann.

Richtige Platzierung eines Access-Points

Die beste Position eines Access-Points kann mitunter schwierig zu wählen sein, da der Standort mitunter einen Strom- und üblicherweise einen LAN-Anschluss erfordert. Beliebt, aber problematisch sind Aufstellorte in Ecken von Räumen und an den Wänden.

Damit ein Access-Point nicht unnötigerweise weit über die Gebäudegrenzen hinausstrahlt, sollten Sie eine Platzierung in der Mitte des Gebäudes bzw. des auszuleuchtenden Bereichs anstreben. Wenn Sie Ihren Access-Point an einer Gebäudewand befestigen müssen, gleichzeitig aber sicher sind, dass Sie hinter der Wand keinen Empfang benötigen werden, sollten Sie über eine geeignete Abschirmung des Access-Points nachdenken. Diese sollte die Signale daran hindern, direkt hinter dem Access-Point ins Freie zu gelangen.

Bei der Planung des Aufstellortes sollten Sie auf keinen Fall vergessen, dass sich die Funkwellen um einen Access-Point herum in drei und nicht nur in zwei Dimensionen ausbreiten. Achten Sie also darauf, dass Sie in mehrstöckigen Gebäuden weder die Lauscher auf dem Parkplatz vor dem Gebäude noch fremde Benutzer in den Etagen über und unter Ihnen unabsichtlich mit einem WLAN versorgen.

Insofern sind eine richtige Auswahl und Konfiguration des Access-Points (AP) wichtig:

- ✓ Achten Sie darauf, dass die **Sendeleistung** des Access-Points nahe am zulässigen Maximum von 20 dBm liegt, damit Sie eine optimale Funkausleuchtung innerhalb des von Ihnen genutzten Bereiches erhalten.
- ✓ Beachten Sie die **Empfangsempfindlichkeit** des Access-Points. Er sollte bis -97 dBm erreichen. Je höher die Empfangsempfindlichkeit, desto schwächere Signale kann der AP noch empfangen.
- ✓ Um eine Optimierung bezogen auf die Nutzung von externen Antennen zu ermöglichen, sollte der AP über Antennenanschlüsse am Gerät verfügen.
- ✓ Nutzen Sie einen Standort für den AP, der die Funkzelle nur auf den von Ihnen genutzten Bereich einschränkt.
- ✓ Wählen Sie eine sichere Verschlüsselung für Ihr WLAN (WPA2/WPA3).
- ✓ Verbergen Sie Ihre SSID, sofern Ihre WLAN-Clients dies zulassen.
- ✓ Überprüfen Sie die Funkkanäle auf weitere Sender auf der gleichen Frequenz. Nutzen Sie freie (möglichst überlappungsfreie Kanäle), um Störungen der WLANs auszuschließen.
- ✓ Checken Sie die Übertragungsstandards (z. B. keinen Mischbetrieb erlauben, wenn alle WLAN-Clients den gleichen Standard unterstützen).
- ✓ Setzen Sie Zugriffsbeschränkungen ein (MAC-Adressen-Filterung, Anzahl der möglichen DHCP-Clients beschränken etc.).
- ✓ Verwenden Sie **sichere und lange** Passwörter und Schlüssel.
- ✓ Nutzen Sie weitere Sicherheitsmechanismen, die Ihr Access-Point bietet (z. B. Firewall-Funktionalität).

17.9 Übung

Fragen zu WLAN

Übungsdatei: --

Ergebnisdatei: uebung17.pdf

1. Welche Verschlüsselungsprotokolle nutzt WLAN?
2. Welche dieser Verschlüsselungsprotokolle für drahtlose Netzwerke existieren?

a	WPA
b	SSID
c	TKIP
d	WIFI
e	MAC
f	WEP
g	WPA2

3. Das aktuell sicherste Verschlüsselungsprotokoll für drahtlose Netzwerke ist ...

a	WPA4
b	SSID
c	TKIP
d	WIFI
e	MAC
f	WEP
g	WPA3
h	PSK

4. Welche der folgenden Aussagen sind zutreffend?

a	WEP unterstützt eine maximale Schlüssellänge von 128 Bit.
b	WEP arbeitet mit Pre-Shared Keys.
c	WPA nutzt zur Verschlüsselung den Standardalgorithmus AES.
d	Unter WPA wird allgemein Wi-Fi Protected Access verstanden.
e	WPA2 arbeitet mit dem Standard IEEE 802.11i.

5. Wie wird ein drahtloses Netzwerk optimal geschützt?

a	WLANs sind prinzipiell sicher.
b	<ul style="list-style-type: none">✓ Die SSID wurde versteckt.✓ WPA wird als Verschlüsselung verwendet.✓ MAC-Adressen-Filterung✓ Der Access-Point wurde so positioniert, dass dieser nur geringfügig über die Gebäudegrenzen hinausstrahlt.
c	<ul style="list-style-type: none">✓ Die SSID wurde versteckt.✓ WEP 2 wird als Verschlüsselung verwendet.✓ Ein Mischbetrieb der Übertragungsstandards wurde unterbunden.✓ MAC-Adressen-Filterung✓ Der Access-Point wurde so positioniert, dass dieser nur geringfügig über die Gebäudegrenzen hinausstrahlt.
d	WEP wird als Verschlüsselung verwendet.
e	<ul style="list-style-type: none">✓ Die SSID wurde versteckt.✓ WPA3 wird als Verschlüsselung mit einem sicheren Passwort verwendet.✓ Ein Mischbetrieb der Übertragungsstandards wurde unterbunden.✓ Es findet eine MAC-Adressen-Filterung sowie eine Einschränkung der DHCP-Clients statt.✓ Der Access-Point ist so positioniert, dass dieser nur geringfügig über die Gebäudegrenzen hinausstrahlt.

18

Alternative Software

18.1 Warum Nicht-Standard-Software sinnvoll sein kann

Monokulturen leben gefährlich

Auch wenn dieses Kapitel keinen Anspruch auf Vollständigkeit erhebt, so soll hier doch kurz aufgezeigt werden, dass Sie durchaus die Möglichkeit haben, durch die geschickte Wahl der von Ihnen benutzten Software Ihr Sicherheitsrisiko zu minimieren oder Ihre Handlungsfähigkeit zu erweitern. Wenn es darum geht, Sicherheitsprobleme durch entsprechende Konfiguration zu vermeiden, kann es sein, dass ein anderes Programm oder sogar ein anderes Betriebssystem Ihnen mehr Optionen zur Verfügung stellt als die weitverbreitete Standardsoftware eines Marktführers.

Wie auch in anderen Bereichen unseres Alltags ersichtlich ist, z. B. bei der Massentierhaltung oder in der Landwirtschaft, werden Computernetzwerke anfälliger für einen Schädling, wenn eine starke Tendenz zur Monokultur besteht. Findet ein Schädling hier eine Lücke, die ausgenutzt werden kann, so fällt diese auf ungleich fruchtbareren Boden, wenn 90 % aller Anwender mit demselben Betriebssystem und Programm arbeiten, als wenn die gesamte IT-Landschaft in viele kleine logische Inseln aus unterschiedlicher Software und Protokollen zusammengesetzt wäre. Hier ist der Aufwand, Betriebssysteme bzw. deren Applikationen zu attackieren, ungleich höher.

Aus der Sicherheitsperspektive muss man jedoch sagen, dass „leider“ in der Informatik ein natürlicher Drang zur Entwicklung von Monokulturen bzw. Monopolen besteht. Dies ist durchaus logisch, da der Nutzen und die Integrationsfähigkeit eines Systems steigen, je mehr Teilnehmer mit denselben Standards arbeiten. Der Nachteil solcher natürlichen Monopole ist allerdings, dass Angreifer, die kriminelle Absichten verfolgen (zum Beispiel den Massenversand von SPAM) und überlegen, welches Programm und Betriebssystem sie bei dem Versuch, z. B. ein Botnetz aufzubauen, am besten attackieren sollten, das am weitesten verbreitete ins Visier nehmen.

Die weite Verbreitung von Microsoft-Betriebssystemen und den darauf installierten Standardapplikationen macht deswegen diese Plattform zum beliebten Ziel von Netzwerk-Kriminellen. Die am häufigsten benutzten Einfallstore sind installierte Software wie Outlook, der Internet Explorer und nicht installierte Patches des Betriebssystems.

Alternatives Betriebssystem

Die grundlegendste Entscheidung, die Sie treffen können, ist die Entscheidung für oder gegen ein bestimmtes Betriebssystem. Microsoft Windows hat hier den größten Marktanteil. Sollten Sie jedoch in der Lage sein, bei einer Neuinstallation das System frei wählen zu können, so sollten Sie prüfen, ob der geplante Einsatzzweck eines Computers nicht mit alternativen Betriebssystemen genauso gut oder besser erfüllt werden kann.

Mögliche Alternativen sind zum Beispiel:

- ✓ Linux-Distribution:
 - ✓ Ubuntu (<https://www.ubuntu.com/>)
 - ✓ OpenSuSE (<https://de.opensuse.org/Hauptseite>)
 - ✓ Fedora (<https://getfedora.org/>)
 - ✓ Chrome OS (<http://getchrome.eu/download.php>)
 - ✓ oder eine andere Linux-Distributionen unter <https://distrowatch.com/>
- ✓ UNIX:
 - ✓ Oracle (Sun) Solaris 11 (<https://www.oracle.com/technetwork/server-storage/solaris11/downloads/index.html>)
 - ✓ BSD-Betriebssysteme (http://de.wikipedia.org/wiki/Vergleich_der_BSD-Betriebssysteme)
 - ✓ OS X
- ✓ Sonstige:
 - ✓ Remix OS (<https://remix-os.de.uptodown.com/windows/download/440687>)
 - ✓ ReactOS (<https://www.reactos.org/de>)

Die Auswahl an Software für alternative Betriebssysteme ist ebenfalls recht groß, sodass die Chancen, dass eine Anwendung für den von Ihnen gewünschten Zweck verfügbar ist, gut stehen. Sollten Sie ein alternatives Betriebssystem in Erwägung ziehen, so informieren Sie sich gut und vergleichen Sie Kosten und Nutzen.

Alternative Anwendungen

Bedenken Sie jedoch immer: Keine Software ist vollständig fehlerfrei, auch wenn diverse Firmen oder Personen manchmal bei ihrer Lieblingssoftware versuchen, diesen Eindruck zu erwecken. Es bestehen jedoch Unterschiede in der Qualität des Sourcecodes oder der Geschwindigkeit, mit der das Entwicklerteam Patches herausgibt. Ein alternatives Betriebssystem ist vielleicht deswegen sicherer, weil nur wenige andere es verwenden und sich für Cracker die Mühe nicht lohnt, nach Lücken zu suchen, wenn damit nur wenige Computer unter Kontrolle gebracht werden können.

Sehen Sie sich auf dem Softwaremarkt nach Lösungen für Ihr Problem um, und vergleichen Sie.

Virtualisierung

Virtualisierung gehört heute zu den Schlüsseltechniken der Unternehmens-IT. Server-Virtualisierung und Storage-Virtualisierung ermöglichen eine bessere Auslastung der Hardware, Kosten können gespart und IT-Ressourcen flexibler genutzt werden. Gleichermaßen gilt für die Virtualisierung von Netzwerken, Endsystemen (Desktops) und Applikationen. Bei Endsystemen können Sie in der virtualisierten Umgebung das Betriebssystem und die notwendigen Applikationen in einer geschützten Umgebung ausführen. Mit einem Snapshot (Kopie der auf der Festplatte gespeicherten Daten einer virtuellen Maschine zu einem definierten Zeitpunkt) kann jederzeit ein Abbild der aktuellen Funktion des Systems angefertigt werden. Ein erfolgreicher Malwareangriff hat einerseits nur Auswirkung innerhalb der Virtualisierung und kann andererseits durch das Einspielen des letzten Snapshots eliminiert werden. Nachfolgend die am häufigsten genutzten Möglichkeiten der Virtualisierung:

- ✓ Citrix-Hypervisor (<https://www.citrix.de/products/citrix-hypervisor/>)
- ✓ Proxmox VE (<https://www.proxmox.com/de/proxmox-ve>)
- ✓ Microsoft Hyper-V, Unterstützung ab Windows 8 und Windows Server 2008 (<https://docs.microsoft.com/de-de/virtualization/hyper-v-on-windows/about/>)
- ✓ Oracle VM VirtualBox (<https://www.virtualbox.org>)
- ✓ Red Hat Enterprise Virtualization (<https://www.redhat.com/en/technologies/virtualization>)
- ✓ VMware (<https://my.vmware.com/de/web/vmware/downloads>)

Einige Hersteller von Virtualisierungslösungen bieten eingeschränkte Funktionen ihrer Produkte kostenfrei an. Kostenpflichtig sind dagegen Managementwerkzeuge, die in größeren Systemumgebungen zwingend benötigt werden. Mit den kostenfreien Varianten versuchen die Hersteller, potenzielle Kundschaft zu locken und zu binden.

18.2 Alternative Webbrowser

HTTP-Browser

Die am häufigsten verwendeten Browser (Stand 03/2019) sind neben Google Chrome (45 %), Safari (31 %), Firefox (14 %), der Internet Explorer (7 %) und Mozilla (3 %). Sie unterscheiden sich für den normalen Nutzer nur gering in den Punkten Stabilität, Bedienbarkeit und Sicherheit. Alternative Browser nutzen fast immer den Basiscode der Standardbrowser, wobei sie deren Funktionen optimiert bzw. um zusätzliche Add-On erweitert haben.

Für die Virtualisierung der HTML-Anzeige (Hypertext Markup Language) wird ein HTML-Renderer eingesetzt. Er wird auch als Layout-Engine oder Web-Browser-Engine bezeichnet.

Die hierbei am häufigsten angewendeten HTML-Renderer sind:

- ✓ Presto (Opera bis Version 12)
- ✓ Trident (Microsoft Internet Explorer)
- ✓ EdgeHTML (Microsoft Edge Windows 10, ab Mitte 2019 wird Edge auf der Basis von Chromium laufen)
- ✓ Webkit (Safari, Google Chrome)
- ✓ Gecko (Mozilla Firefox)
- ✓ Blink (Opera seit Version 15, Google Chrome)

Eine der möglichen Alternativen zum Internet Explorer von Microsoft ist beispielsweise Firefox. Firefox ist ein Open-Source-Projekt und kostenlos erhältlich. Er ist unter <https://www.mozilla.org/de/> verfügbar oder kann bei der Installation des Microsoft-Betriebssystems als Standardbrowser festgelegt werden.



Browser Firefox

Diese Option war auch ein Ergebnis einer Klage der EU gegen die Firma Microsoft, um eine Wettbewerbsgleichheit für alternative Browser zu erreichen.

Die Wettbewerbsgleichheit bedeutet gleichzeitig auch Gleichheit in den grundlegenden Funktionen eines Browsers. Alle der hier aufgezeigten Browser besitzen mindestens die Standardfunktion der folgenden Tabelle.

Tabbed Browsing	Der Inhalt mehrerer Webseiten wird in mehreren übersichtlichen Registerkarten innerhalb eines einzigen Browserfensters angezeigt.
Eingebauter Popup-Blocker	Popup-Werbebanner werden auf Wunsch automatisch ausgefiltert.

Weitere Browser

Neben den marktrelevanten Browsern existieren einige interessante Alternativen, die über bemerkenswerte Details verfügen. Die Auswahl spiegelt die relevanten Aspekte wieder, wobei die Wichtigkeit nicht vollständig sein kann.

Performance/Bedienung

- ✓ **Cyberfox** (Basiscode Gecko), angepasste 64-Bit-Version zur effizienten Nutzung des gesamten Speicherbereiches
- ✓ **Maxthon Cloud Browser**, verfügbar auf vielen Systemplattformen, umfassende Integration von Cloud-Funktionen
- ✓ **Midori** (Basiscode Midori), reduzierter und schneller Seitenaufbau
- ✓ **Pale Moon** (Basiscode Gecko), performanter als der Standardbrowser unter Windows
- ✓ **QtWeb**, schnell, performant, sicher auf allen Plattformen
- ✓ **Lunascape**, Multi-Engine-Webbrowser, beinhaltet die Engine von IE, Firefox, Safari und Chrome
- ✓ **CoolNovo** (Basiscode Chromium), gute Funktionserweiterung des Google Chrome
- ✓ **SeaMonkey**, plattformneutraler Browser mit zusätzlicher E-Mail-Funktion und integriertem Composer (pfiffiger HTML-Editor)

Portabilität

- ✓ Maxthon Cloud Browser
- ✓ Opera@USB (Basiscode Opera)
- ✓ QtWeb
- ✓ Midori
- ✓ Tor-Browser

Sicherheit/Anonymität

- ✓ **COMODO Dragon** (Basiscode Chromium), sehr interessante Features, u. a. der Comodo SecureDNS-Server
- ✓ **Avant Browser 2018** (Basiscode Trident, Webkit und Gecko), effizientes Sperren von Popups und der Chroniken des Surfverhaltens
- ✓ **JonDoFox** (Basiscode Gecko), Blockieren aller Zugriffe auf Java, Flash, Plugins und Cookies
- ✓ **SRWare Iron** (Basiscode Chromium), Gewährung der Privatsphäre durch Unterbindung der Weitergabe von Nutzer-ID-Angaben und des Nutzerverhaltens
- ✓ **Bitbox** (Basiscode Firefox), Ausführen aller Surfaktivitäten in einer virtuellen Maschine
- ✓ **Tor-Browser**, Eweiterungspaket mit Firefox

18.3 Alternative E-Mail-Clients

Pegasus Mail

Der frei verfügbare Mail-Client von David Harris existiert schon sehr lange auf dem Markt. Die Homepage ist unter <http://www.pmail.com> zu erreichen.

Da Pegasus Mail schon lange vor der ersten Outlook-Release bei vielen Usern als Mail-Client im Einsatz war, hat es einen anderen optischen Aufbau, der wechselwilligen Benutzern verwirrend erscheinen mag. In der letzten Version von Pegasus Mail (auch PMail abgekürzt) wurde jedoch optional auch eine Anzeigeoption integriert, die dem typischen Outlook-Aufbau stark ähnelt. Pegasus Mail war früher als E-Mail-Client sehr sicher, da es einfach HTML-Mail nicht unterstützte – was nicht gerade zu seiner Popularität beitrug – aber dieses Manko wurde behoben. Die aktuelle PMail-Version verfügt über eine eigene HTML-Rendering Engine, die für die Anzeige von HTML-Inhalten in E-Mails speziell entwickelt wurde. Aus diesem Grund ist PMail beim Anzeigen von HTML-Inhalten nicht in derselben Art und Weise angreifbar, wie viele andere Mail-Clients, die bei HTML-Inhalten einfach auf die Bibliotheksfunktionen von Internet Explorer – und damit auch auf seine Sicherheitslücken – zugreifen.

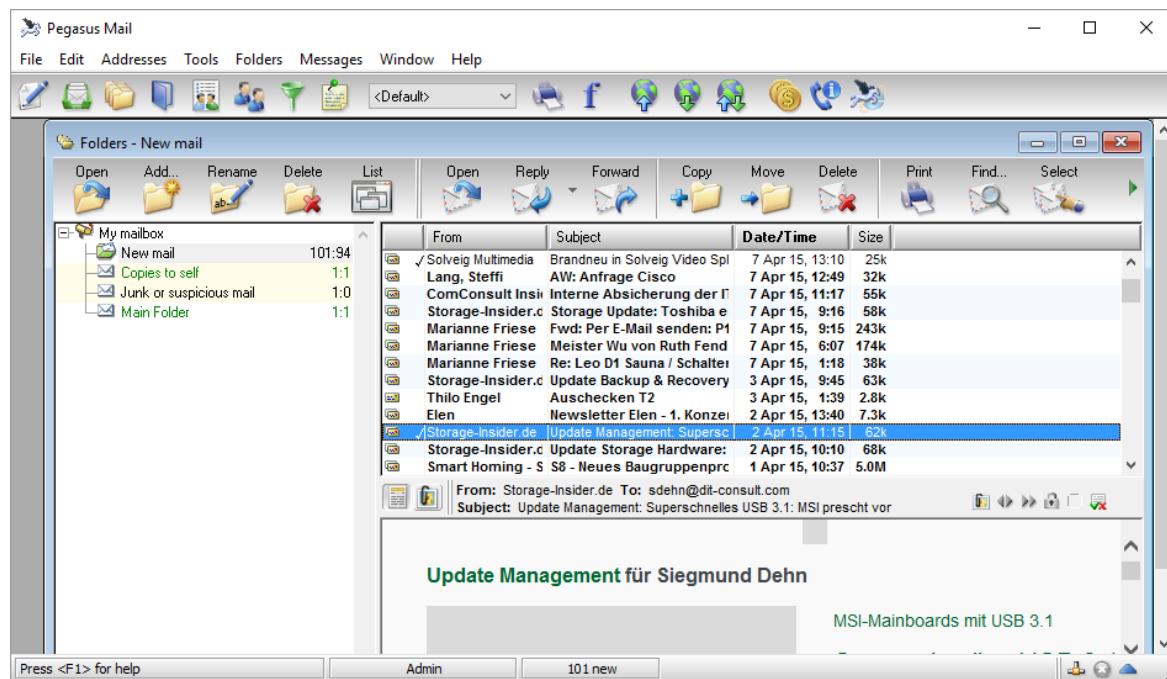
Das Ausspionieren durch Webbugs verhindert PMail, indem das Nachladen von Inhalten in E-Mails bei unbekannten Absendern standardmäßig nur auf ausdrücklichen Wunsch des Benutzers durchgeführt wird. Aktive Inhalte in HTML werden grundsätzlich nicht gerendert. Neu ist ebenfalls ein Phishing-Schutz, der Links in E-Mails bei einem Klick durch den Benutzer auf Plausibilität prüft und den Benutzer warnt, wenn übliche Phishing-Tricks benutzt werden (beispielsweise, wenn ein Link in Wirklichkeit zu einer anderen URL führt als der, die der Benutzer sehen kann).

Auch bei Dateianhängen ist PMail auf Sicherheit ausgelegt. Für einige Dateitypen bringt er interne Anzeigemöglichkeiten mit (z. B. für Bilddateien).

Während sich für diese und weitere Typen problemlos externe Programme definieren lassen, werden sicherheitskritische Dateitypen wie .EXE, .SCR, .PIF etc. von einer direkten Ausführung per Doppelklick ausgeschlossen.

Zum Schutz vor Spam-Nachrichten enthält die letzte Version von Pegasus Mail einen lernfähigen Bayes-Filter, der nach einer kurzen Trainingsphase gute Ergebnisse beim Aussortieren unerwünschter Nachrichten liefert.

Durch seine umfangreichen Konfigurationsmöglichkeiten, gute Sicherheitsfunktionen und das langjährige Bestehen hat sich PMail eine treue Nutzergemeinde geschaffen, die dieses „Arbeitspferd“ auch weiterhin rege nutzt.

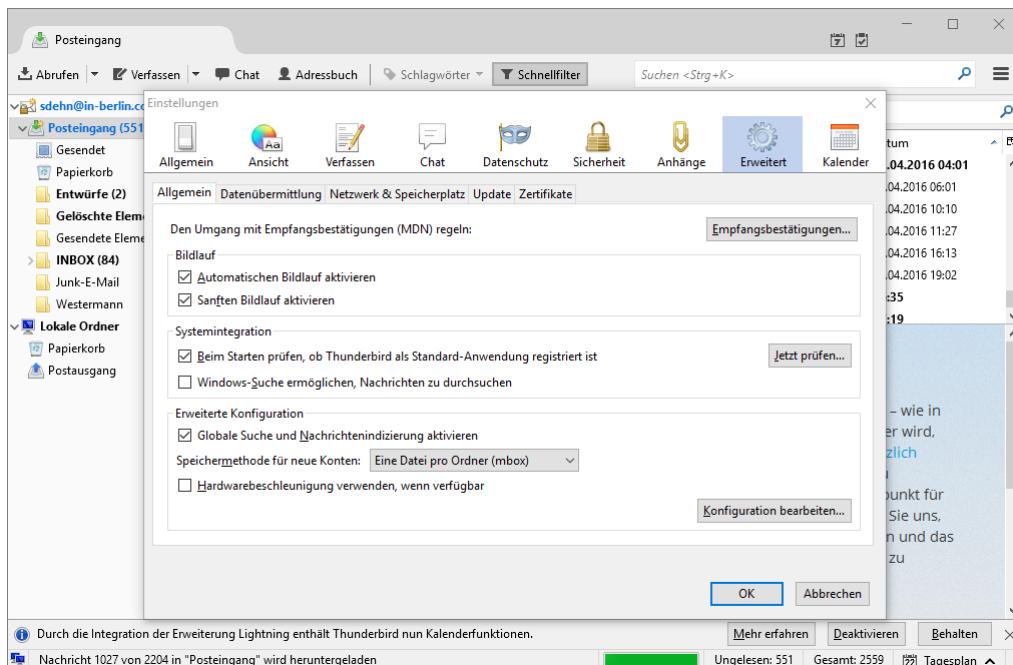


Pegasus Mail

Thunderbird

Ebenfalls wie Firefox ein Open-Source-Projekt, stellt Thunderbird einen sehr brauchbaren alternativen E-Mail-Client dar. „Look and Feel“ sind an den Aufbau von Outlook angepasst, sodass sich ein Umsteiger hier schnell zurechtfinden sollte.

Thunderbird integriert (wie PMail) die grundlegende Funktion, fehlende Bildinhalte in HTML-E-Mails nicht ungefragt von einem Server im Internet automatisch nachzuladen (sog. Lazy-HTML, vgl. Abschnitt 8.1). Der Benutzer kann allerdings auch hier Ausnahmelisten mit vertrauenswürdigen Absendern definieren. Ebenso praktisch ist die nahtlose Integration eines selbstlernenden Bayes-Spamfilters in das E-Mail-Programm.



Thunderbird mit einigen Konfigurationsoptionen

Die Projekthomepage ist unter <https://www.thunderbird.net/de/> zu erreichen.

Webmail

Alternativ zu lokalen E-Mail-Clients bietet sich auch diese Möglichkeit, die z. B. alle großen Provider ihren Kunden anbieten. Damit hat man von überall, wo ein Internetzugang verfügbar ist, per Browser Zugriff auf seine E-Mails, Kontakte etc. oder kann Unified-Messaging-Lösungen (z. B. das Versenden einer E-Mail als Fax) nutzen. Das lässt sich auch für ein einzelnes Unternehmen realisieren. Die Administration (Schutz vor Spam und jeglicher Art von Schadcode) bzw. der Support erfolgt in diesem Fall von einer zentralen Stelle aus.

Eine Zusammenstellung relevanter Webmail-Anbieter finden Sie unter <https://de.wikipedia.org/wiki/Kategorie:Webmail-Anbieter>.

19

Authentifizierungssysteme

19.1 Kerberos

Am Netzwerk anmelden

Damit im Betriebssystem der Benutzer eindeutig identifiziert werden kann, werden üblicherweise die Anmeldeinformationen in Form einer Benutzername/Passwort-Kombination auf dem System lokal abgelegt. Wenn die Anmeldung eines Benutzers nicht lokal erfolgt, sondern über einen Server im Netzwerk, so muss eine zentrale Authentifizierungsinstanz vorhanden sein.

Eine zentrale Zugriffskontrolle (Authentifizierung) hat den Vorteil des einfachen Administrierens der Userrechte. Die Übertragung von Username/Passwort über das Netzwerk erfolgt dabei verschlüsselt.

Entstehung von Kerberos

Kerberos V5 entstand als Dienst zur Berechtigungsüberprüfung im Rahmen des Projektes Athena am MIT. Ein Entwurfsziel war es, keine Kennworte im Klartext zu übertragen. Es ist in den Standards RFC 1964 (The Kerberos Version 5 GSS-API Mechanism) und RFC 4120 (The Kerberos Network Authentication Service) beschrieben.

Ab der Einführung von Windows 2000/2003 wurde Kerberos auch für Windows-Netzwerke zum Standard-Authentifizierungsprotokoll. Bei der Active Directory-Anmeldung kommt ebenfalls Kerberos zum Einsatz.

Eigenschaften von Kerberos

Gegenseitige Authentifizierung	Client und Server können eine gegenseitige Authentifizierung fordern. Dadurch besteht sowohl für den Server als auch für den Client die Sicherheit, dass der Kommunikationspartner derjenige ist, für den er sich ausgibt. Der Kerberos-Server selbst authentifiziert sich gegenüber Client und Server und überprüft deren Identität.
Effizienz	Bei der Kerberos-Authentifizierung stellt der Server die Identität direkt fest, indem er das vom Client vorgelegte Ticket überprüft.

Weiterreichen der Authentifizierung	Kerberos unterstützt Single Sign-on. Ein Benutzer muss sich nur einmal mit seinem Passwort anmelden, um auf alle für ihn vorgesehenen Funktionen zugreifen zu können.
Transitivität	Domänen brauchen bei Kerberos keine komplexen Vertrauensstellungen zu definieren. Es besteht standardmäßig ein beiderseitiges Vertrauensverhältnis. Diese Beziehungen sind transitiv. Dies bedeutet, wenn sich die Domänen A und B vertrauen und Domäne B vertraut Domäne C, vertrauen sich auch die Domänen A und C.
Interoperabilität	Kerberos ist ein Standard-Protokoll, das auf den meisten Rechnerplattformen verfügbar ist. Dadurch können beispielsweise UNIX-Rechner problemlos in eine Windows Active Directory-Domäne integriert werden.

Kerberos-Tickets

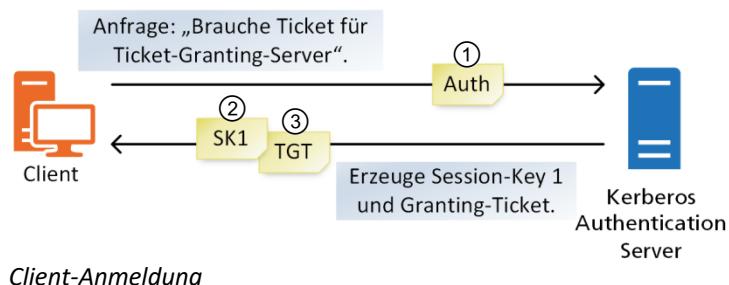
Kerberos baut auf einem Ticketsystem auf. Darüber hinaus müssen alle Teilnehmer an Kerberos dem Kerberos-Server vertrauen.

Kerberos nutzt grundsätzlich die symmetrische Kryptografie zum Schutz von Informationen, ist aber auch erweiterbar für die Funktionen der asymmetrischen Kryptografie und eine Autorisierung durch Public Keys.

Eine typische Anmeldung und der Zugriff eines Clients auf die Ressourcen eines Zielservers läuft üblicherweise in vier Schritten ab:

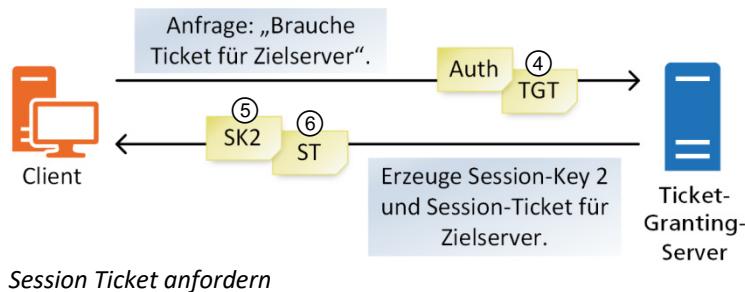
Zuerst muss der Client sich beim Kerberos Authentication Server ausweisen. Dazu bittet der Client in einer Anfrage um die Ausstellung eines TGT (Ticket Granting Ticket) ①. Das vom Client verschlüsselt gesendete Passwort kann der Server in einer Datenbank nachschlagen und so feststellen, ob die Anfrage berechtigt ist.

Der Server generiert einen Session Key 1 (SK1), der für die Kommunikation zwischen Client und Ticket Granting Server vorgesehen ist, verschlüsselt diesen mit dem geheimen Schlüssel des Clients und schickt diesen zurück ②. Zusätzlich werden der Session Key 1 und weitere Informationen (z. B. ein Timestamp) mit dem geheimen Schlüssel des Ticket Granting Servers verschlüsselt und als Datenpaket ebenfalls an den Client geschickt. Dieses Datenpaket ist das Ticket Granting Ticket (TGT) ③.



Mit dem TGT wendet sich der Client an den Ticket Granting Server (TGS). Dieser Serverdienst kann auf demselben Rechner laufen wie der Authentication Server, kann aber auch von einem anderen Rechner des Netzwerks gestellt werden.

Dies geschieht folgendermaßen: Nachdem der Client mithilfe seines geheimen Schlüssels den Session Key 1 aus der Nachricht des Authentication Servers erhalten hat, benutzt er diesen, um seine Nutzerdaten und die Anfrage an den TGS zu verschlüsseln. Er legt auch das TGT bei ④.



Session Ticket anfordern

Der TGS ist nun aufgefordert, dem Client ein Session Ticket für den Zielserver auszustellen. Der TGS kann das TGT entschlüsseln, weil es vom Authentication Server mit dem geheimen Schlüssel des TGS verschlüsselt wurde. Darin ist unter anderem der Session Key 1 enthalten, den der TGS benötigt, um die Authentifizierung des Clients entschlüsseln zu können. Kann der TGS das TGT und die Authentifizierung, die der vom Client erhalten hat, korrekt entschlüsseln, so steht Folgendes fest:

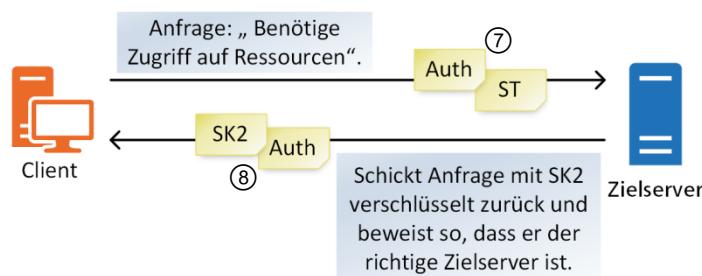
- ✓ Der Client hat sich zuvor korrekt am Authentication Server angemeldet, da er das TGT erhalten hat.
- ✓ Der Client ist wirklich der Client, da er über den Session Key 1 verfügt, der im TGT enthalten war und vom Authentication Server eingepackt wurde.

Der TGS generiert ein Session Ticket für den Zielserver, indem ein Session Key 2 erzeugt wird. Dieser wird mit Session Key 1 verschlüsselt und an den Client gesendet ⑤. Zusätzlich wird das Session Ticket erzeugt, indem der Session Key 2 zusammen mit zusätzlichen Informationen mit dem geheimen Schlüssel des Zielservers verschlüsselt und an den Client übermittelt wird ⑥.

Der Client kann die vom TGS erhaltene Nachricht entschlüsseln und erhält somit Session Key 2. Seine Anfrage an den Zielserver verschlüsselt er, indem er den Session Key 2 zur Verschlüsselung benutzt. Die Anfrage sendet er zusammen mit dem Session Ticket an den Zielserver ⑦.

Der Zielserver kann das Session Ticket entschlüsseln, weil es mit seinem geheimen Schlüssel geschützt wurde. Den darin enthaltenen Session Key 2 kann er benutzen, um die Anfrage des Clients zu entschlüsseln. Somit weiß der Zielserver, dass der Client wirklich der richtige Client und ordnungsgemäß angemeldet ist.

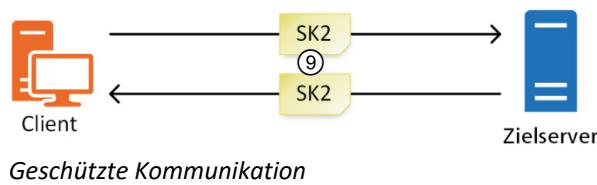
Als Antwort schickt der Zielserver die Nachricht zurück, ebenfalls mit dem verschlüsselten Session Key 2 ⑧. Die Nachricht enthält nun einen aktualisierten Timestamp.



Autorisierung zum Zielserver

Der Client, der diese Antwort erhält, weiß, dass der Zielserver in der Lage war, seine Anfrage zu entschlüsseln, und somit der richtige Server sein muss.

Für die nachfolgende Kommunikationsverbindung können sowohl der Client als auch der Zielserver darauf vertrauen, dass es sich um jeweils den echten Partner handelt. Zur Verschlüsselung benutzen beide den Session Key 2 ⑨.



Gültigkeit des TGT

Der Vorteil von Kerberos liegt nicht nur darin, dass Passwörter nur verschlüsselt über das Netzwerk laufen, sondern auch darin, dass die ursprüngliche Authentifizierung nur einmal erfolgen muss. Der Client meldet sich zuerst einmal beim Authentication Server an und kann mit dem erhaltenen TGT beliebig viele Session Tickets für weitere Dienste beim TGS beantragen.

Um Replay-Angriffe zu verhindern, bei denen ein Angreifer aufgezeichnete Diensttickets abspielt, werden in Tickets auch Informationen über den Ausstellungszeitpunkt und die Gültigkeitsdauer des Tickets sowie Namen und IP-Adressen der Kommunikationspartner gespeichert.

In Windows-Domänen beträgt die Standard-Gültigkeitsdauer eines TGT 10 Stunden. Das reicht, um sich nur einmal pro Arbeitstag authentifizieren zu müssen. Nach Ablauf muss der Client eine erneute Anmeldung durchführen.

In einem durch Kerberos geschützten Netzwerk ist es extrem wichtig, dass alle Computer eine synchrone Uhrzeit haben. Weicht die Uhrzeit eines Computers um mehr als einen gewissen Toleranzwert (Microsoft Default: 5 Minuten) von der Zeit eines anderen Computers ab, so kann der Computer, der ein solches Ticket erhält, nicht unterscheiden, ob es sich um ein gültiges Ticket oder eine Aufzeichnung handelt. Tickets mit ungültiger Timestamp werden verworfen, und der Computer mit der falschen Uhrzeit kann an Kerberos nicht teilnehmen.

19.2 PAP, CHAP, EAP und RADIUS

Authentifizierung von Einwahlverbindungen

Auch Einwahlverbindungen zu einem Remote Access Server (RAS) in der Firma oder zum Internetprovider sollten angemessen geschützt sein. Um hier den Client gegenüber dem Server ausweisen zu können, existieren ebenfalls unterschiedliche Protokolle mit sehr unterschiedlichen Sicherheitsniveaus.

PAP

Das 1992 eingeführte **Password Authentication Protocol** ist das mit Abstand einfachste und unsicherste. Hier werden Benutzernamen und Passwörter im Klartext übertragen. Vorteil von PAP ist, dass es überall implementiert ist und so zumindest als letzte Möglichkeit einer Autorisierung benutzt werden kann, wenn die Sicherheit der Benutzerpasswörter nicht relevant ist. Dieses Protokoll wird heutzutage selten verwendet.

CHAP

Das **Challenge Handshake Authentication Protocol** arbeitet mit MD5-Hashwerten. Passwörter werden hier nicht im Klartext übertragen, sondern im Challenge-Response-Verfahren:

- ✓ Einer der beiden Partner sendet dem anderen eine Anforderung zur Authentifikation. Das Anforderungs-Paket enthält einen für diese Sitzung eindeutigen Sitzungsschlüssel und eine zufällige Zeichenkette (die Challenge).
- ✓ Die Gegenseite empfängt dieses Paket und erzeugt ein Antwortpaket, in dem der Username und zusätzlich der MD5-Hashwert von Passwort, Sitzungsschlüssel und Challenge enthalten sind.
- ✓ Wenn die anfordernde Seite das Paket erhält, extrahiert sie hieraus den Usernamen. Sie erzeugt aus den in der lokalen Authentifizierungsdatenbank hinterlegten Daten mit Username und Passwort auf die gleiche Weise ein Paket und vergleicht die beiden miteinander. Sind die Hashwerte gleich, ist die Authentifizierung erfolgreich. CHAP bietet daneben noch Optionen an, um diesen Vorgang während der Sitzung in regelmäßigen Abständen zu wiederholen. Hierdurch lässt sich erkennen, wenn eine Sitzung entführt wurde und die Gegenstelle sich inzwischen geändert hat.

Neben dem im RFC 1994 beschriebenen CHAP-Standard existieren noch die Versionen MS-CHAPv1 (RFC 2433) und MS-CHAPv2 (RFC 2759) des Herstellers Microsoft.

EAP

Das **Extensible Authentication Protocol** stellt eine Programmierschnittstelle für andere Authentifikationsprotokolle zur Verfügung. Mithilfe von EAP können die Kommunikationspartner vor der eigentlichen Authentifizierung aushandeln, welche Authentifizierungsmethode verwendet werden soll. Aufgrund der Offenheit von EAP können auch in Zukunft entwickelte Protokolle auf EAP aufsetzen und dieses nutzen. EAP wird u. a. für CHAP oder TLS (Transport Layer Security, dem Nachfolger des lange verwendeten SSL).

Die freie Programmierbarkeit und die Möglichkeit, TLS zu verwenden, machen EAP besonders interessant, wenn der Benutzer per Public-Key-Verfahren über eine Smartcard authentifiziert werden soll.

RADIUS

Der **Remote Authentication Dial-In User Service** hat sich als Standard auf dem Gebiet der zentralen Authentifizierung von Einwahlverbindungen jeglicher Art etabliert. RADUIS (RFC 2865) ist eine Protokollfamilie aus den Diensten Access, Authorization und Accounting (**AAA**).

- ✓ **Access:** Zugangskontrolle durch Benutzername und Passwort
- ✓ **Authorization:** Verschiedenen Nutzern oder Gruppen können unterschiedliche Rechte oder Konfigurationen zugewiesen werden.
- ✓ **Accounting:** Abrechnung der Onlinezeit und des übertragenen Datenvolumens (Billing)

Die Zugangsdaten autorisierter Benutzer werden zentral auf dem RADIUS-Server vorgehalten und lassen sich somit relativ einfach verwalten. Bei Microsoft Windows-Server-Produkten heißt die RADIUS-Implementierung seit dem Server 2008 Network Policy Server (vorher Internet Authentication Service). Hersteller von Routern und Switches bieten teilweise eigene RADIUS-Implementierungen an. Weitere Informationen (u. a. über Open-Source-RADIUS-Projekte) finden Sie unter <http://de.wikipedia.org/wiki/RADIUS>.

Diameter

Diameter ist eine Weiterentwicklung des RADIUS-Protokolls mit einem deutlich erweiterten Funktionsumfang. Bisher hat es aber noch keine nennenswerte Verbreitung erlangt. Der Name dieses AAA-Protokolls ist ein Wortspiel aus der Geometrie, wo der Diameter der doppelte Radius ist. Damit sollte angedeutet werden, dass es sich bei Diameter um die zweite Generation von RADIUS handelt.

Weitere Informationen zu Diameter finden Sie unter <http://de.wikipedia.org/wiki/Diameter>.

IEEE 802.1X

Hierbei handelt es sich um einen alleinstehenden Standard, der eine generelle Methode für die Authentifizierung in Netzwerken definiert. Der Zugang zu einem LAN/VLAN/WLAN wird durch einen Switch/Router/Access-Point geregelt, der bei Client-Authentifizierungsanfragen einen Authentifizierungsserver (z. B. RADIUS) kontaktiert und überprüft, ob der Client berechtigt ist, Ressourcen im LAN zu nutzen. Entsprechend wird der Zugang erlaubt oder abgewiesen. Überprüfen Sie vor dem Einsatz, ob alle Ihre Netzwerkgeräte eine IEEE-802.1X-Implementierung besitzen. Heutige für den Unternehmens-Einsatz gedachte Geräte haben meist eine 802.1X-Unterstützung, die Geräten der Consumer-Klasse oftmals fehlt. Ausführliche Informationen zu verwendeten Protokollen, unterstützenden Betriebssystemen u. Ä. finden Sie unter http://de.wikipedia.org/wiki/IEEE_802.1X.

Die in einer 802.1X-Implementierung verwendeten Komponenten sind die folgenden:

- ✓ **Supplicant**
Dies ist die auf dem Endsystem laufende Komponente, die das Protokoll 802.1X implementiert und mit dem Netzwerk kommuniziert.
- ✓ **Authenticator**
Dies ist das Netzwerkgerät wie ein Switch, ein Access-Point oder ein WLAN-Controller. Dieser kommuniziert mit dem Supplicant und leitet die Anfragen an den Authentication-Server weiter. Der Authenticator ist im 802.1X die Stelle, an dem die Zugriffsberechtigungen durchgesetzt werden.
- ✓ **Authentication Server**
Dieser wird von den Netzwerk-Komponenten per RADIUS angesprochen und authentifiziert die Benutzer oder Endgeräte. Nach erfolgreicher Authentifizierung kann eine Autorisierung durchgeführt werden, bei dem das Endsystem auf dem Switch in ein bestimmtes VLAN konfiguriert wird oder aber der Authenticator vom Authentication Server eine Access-Control-List für den Benutzer oder das Endgerät empfängt.

Der ursprüngliche Standard wurde 2004 spezifiziert und war recht schwierig in einer heterogenen Umgebung zu implementieren. 2010 wurde eine neuere Revision des Standards herausgegeben, die deutlich flexibler ist und dadurch weniger Probleme in der Implementierung hervorruft. Trotzdem gehört 802.1X noch nicht zu den Baseline-Security-Technologien, die für die grundlegende Basis-Sicherheit implementiert werden.

19.3 Smartcards und Tokensysteme

Warum Smartcards vorteilhaft sind

Herkömmliche Passwörter muss sich der entsprechende Benutzer merken. Daher besteht die Wahrscheinlichkeit, dass sie vergessen werden. Falls der Benutzer dem Vergessen durch Aufschreiben vorbeugen will, besteht andererseits die Gefahr, dass sie von Unbefugten gelesen werden (z. B. das Post-it, das auf dem Monitor klebt).

Eine Smartcard und die meisten Token haben die Größe einer Kreditkarte und sind somit ein physikalischer Gegenstand, den der Benutzer braucht, um Zugang zum System zu bekommen – ähnlich wie ein Schlüssel.

Smartcards

Um Zugangskontrolle per Smartcard gewährleisten zu können, arbeitet die Smartcard mit asymmetrischer Kryptografie. Auf dem Speicherchip der Smartcard wird der Private-Key gespeichert, der mit digitaler Signatur beliebige Daten signieren kann.

Wird eine solche Smartcard in das dafür vorgesehene Lesegerät gesteckt, so kann die Smartcard durch die elektronische Signatur ihre Echtheit nachweisen. Wird der öffentliche Schlüssel dieses Smartcard-Keys mit den Benutzerinformationen des Benutzers verknüpft, an den diese Karte ausgehändigt wurde, kann sich der Benutzer in Zukunft durch Einsticken seiner Karte an einem Computer authentifizieren.

Üblicherweise wird die Signaturfunktion der Smartcard oder der Anmeldeprozess noch durch eine kurze PIN geschützt, damit ein Diebstahl der Karte dem Dieb nicht kompletten Zugang mit den Rechten des Bestohlenen ermöglicht.

Der Einsatz von Smartcards gilt wegen der verwendeten Public-Key-Algorithmen als sehr sicher, bedeutet aber bei seiner Einführung auf jeden Fall einen erheblichen Beschaffungsaufwand von Hardware (Lesegeräte, Smartcards). Zudem muss eine funktionierende Infrastruktur für die Generierung und Verteilung von Schlüssel-Zertifikaten (PKI) vorhanden sein.

Token

Token-Karten besitzen üblicherweise ein Anzeigefeld, auf dem in regelmäßigen Intervallen (meist 60 Sekunden) eine neue, scheinbar zufällige Zeichensequenz angezeigt wird. Diese Zeichensequenz wird aber nach einem deterministischen Algorithmus, der vom Systemverwalter gestartet wird, ausgegeben. Derselbe Algorithmus ist in Hardware- (PCI-Einsteckkarte) oder Softwareform auch auf dem Server installiert worden und wurde bei der Installation so synchronisiert, dass zum selben Zeitpunkt dieselben Zeichenketten produziert werden.

Ein Benutzer, der Zugang zu einem Token-geschützten System haben möchte, benötigt also das Token, um den aktuell gültigen Zugangscode eingeben zu können. Als Diebstahlschutz wird auch hier, wie bei den Smartcards, zusätzlich eine PIN vom Benutzer gefordert, die zusammen mit der aktuellen Token-Zeichenkette den Zugang autorisiert.

Tokensysteme haben gegenüber Smartcards den Vorteil, dass nicht ganz so viel neue Hardware erworben werden muss. Hier wird pro Benutzer ein Token benötigt. Der Server benötigt einen Token-Generator, der in Hardware oder als Software implementiert sein kann.

19.4 Biometrie

Der Mensch als Ausweis

Interessant als Zugangskontrolle sind auch biometrische Systeme. Das bedeutet, es werden biologische Messdaten des Benutzers erfasst und ausgewertet. Werden Eigenschaften gefunden, von denen angenommen werden kann, dass sie für ein bestimmtes Individuum eindeutig sind, so kommen sie als Grundlage für ein Authentifizierungsverfahren in Betracht.

Folgende Merkmale werden häufig bei biometrischen Verfahren genutzt:

- ✓ Fingerabdruck
- ✓ Handgeometrie
- ✓ Irismuster
- ✓ Stimmuster
- ✓ Tippverhalten eines Benutzers beim Schreiben auf einer Tastatur
- ✓ Unterschrift auf einem Grafiktablett
- ✓ Venenmuster in Hand oder Finger
- ✓ Gesichtsgeometrie

Psylock

Die meisten biometrischen Verfahren messen direkt die Eigenschaften eines Merkmals am Benutzer aus, z. B. Linienprofil der Finger, Musterung der Iris, Frequenzanteile der Stimme beim Sprechen.

Dagegen basiert die Erkennung eines Benutzers anhand des Tippverhaltens eher auf nicht so leicht erkennbaren Eigenschaften wie dem Schreibrhythmus, der verwendeten Tastenfolge (hierzu zählt z. B. die Auswahl der Shifttasten) und Tippfehlern. Genau betrachtet ist das Schreibverhalten also keine biometrische Eigenschaft, sondern eine psychometrische.

Die Forschung hat gezeigt, dass bei einigermaßen geübten Schreibern eindeutige Muster erkennbar sind, auch wenn das System den Benutzer nur relativ kurze Texte von ca. 50 bis 100 Zeichen zur Überprüfung tippen lässt. Der Benutzer kann sich also auf einem mit diesem in Regensburg entwickelten Verfahren (PSYLOCK) geschützten System durch das Tippen eines Textes anmelden, ohne sich ein Passwort merken zu müssen.

Bedarf und Lösung

Je nach Sicherheitsbedarf sind für den geplanten Einsatzzweck Lösungen in verschiedenen Preisklassen verfügbar. Vor allem Fingerabdruck-Scanner sind im PC-Sektor derzeit recht beliebt, da sie sehr leicht herzustellen bzw. zu implementieren sind. Aufwendige Lösungen, die mit zunehmendem Preis auch Möglichkeiten zur Lebenderkennung beinhalten, sind allerdings weniger für Privatanwender als für Firmen mit hohen Sicherheitsansprüchen konzipiert. Ein biometrisches System mit Lebenderkennung stellt zusätzlich zur Musterüberprüfung (Fingerabdruck) auch sicher, dass es sich um einen echten, lebenden Finger und nicht um eine Fälschung (Fingerabdruck auf Klebestreifen, Gelatine-Nachguss etc.) oder gar um den abgetrennten Originalfinger des berechtigten Benutzers handelt.



Siemens ID Mouse mit Fingerabdrucksensor

Probleme der Biometrie

Anfang 2002 ergaben unabhängige Tests, dass sich viele der billigen Systeme durch Imitate täuschen ließen. So konnte ein Fingerabdruck-Sensor mit dem von einem Wasserglas auf Tesa-film gezogenen Fingerabdruck eines autorisierten Benutzers überlistet werden, oder es wurden aus Gelatine nachgegossene Fingerimitate akzeptiert.

Auch Fälle, in denen Iris-Scanner das vorgehaltene Foto vom Auge des berechtigten Benutzers akzeptierten, zeigen deutlich, dass Hersteller nicht nur über die reine Datenerfassung, sondern auch die Absicherung gegen Umgehung der Tests implementieren müssen. Dies könnte zum Beispiel durch eine Lebenderkennung sichergestellt werden.

Der Einsatz von Biometrie sollte auch deswegen mit Vorsicht beschlossen werden, da unter Umständen der vermeintliche Sicherheitsgewinn durch die Authentifizierung ein größeres Risiko für die authentifizierten Personen bedeuten kann, wie folgendes Beispiel zeigt:

In einem authentischen Fall war in Malaysia ein Fahrzeug per Fingerabdrucksensor gesichert. Da das Fahrzeug von Dieben nicht ohne Weiteres gestohlen werden konnte, entführten sie zusätzlich den Eigentümer und zwangen ihn mit vorgehaltener Waffe, die Authentifizierung vorzunehmen. Als sie diesen loswerden wollten, hackten sie dem Eigentümer den Finger ab, warfen ihn aus dem Fahrzeug und fuhren mit dem gestohlenen Auto davon.

Die Diebstahlsicherung des Fahrzeugs hat hier also nicht, wie man erhofft hätte, Fahrzeugdiebe davon abgehalten, ein Fahrzeug zu stehlen, sondern neben dem Diebstahl des Fahrzeugs auch dazu geführt, dass der rechtmäßige Eigentümer entführt und verstümmelt wurde.

Je nach Wert der Komponenten, deren Zugang mithilfe biometrischer Maßnahmen geschützt werden soll, müssen Sie vor deren Einsatz abwägen, ob eventuell ein höheres Risiko für die zu authentifizierenden Personen entsteht und ob dies in Kauf genommen werden kann.

Ein System auswählen

Für die Auswahl eines Systems ist nicht nur die False-Positive-Rate relevant, also der Prozentsatz der unberechtigt zugelassenen Benutzer, sondern auch die False-Negative-Rate. Damit ist die Rate der Benutzer gemeint, die das System trotz Berechtigung nicht korrekt erkennt. Zu berücksichtigen sind hier Faktoren wie Alterung, Verletzungen und Krankheit, die auf bestimmte biometrische Merkmale einen Einfluss haben können. In solchen Fällen müssten die biometrischen Messdaten erneuert werden, oder der Benutzer bekommt keinen Zugang zum System.

Sind 99 % Trefferquote wirklich gut genug?

Auch wenn z. B. seitens der Industrie bei der Biometrie im Einsatz von Reisepässen mit hohen Erkennungsquoten von 99 % gerechnet wird, bedeutet das: 1 % aller Personen werden falsch klassifiziert. Die Zahlenangaben in Bezug auf Verlässlichkeit und deren Brauchbarkeit in der Praxis sollten immer mit Vorsicht genossen werden.

Hierzu ein Beispiel:

Bei der biometrischen Personenidentifikation an Flughäfen (wie sie schon öfters erprobt wurde und aktuell propagiert wird) bedeutet dies: Von 34,5 Mio. harmlosen Passagieren pro Jahr würden am Münchner Flughafen 345.000 vom System falsch klassifiziert. Das ergibt bei einem angenommenen 16-Stunden-Flugbetrieb jeden Tag eine Anzahl von fast 59 Reisenden, die **pro Stunde** vom System versehentlich als Verdächtige ausgewiesen würden. Sie könnten also beim Check-in am Münchner Flughafen beobachten, dass im Schnitt jede Minute ein Passagier womöglich fälschlich als „Terrorist“ deklariert und von Sicherheitskräften abgeführt wird – ein unzumutbarer Zustand.

Sie können auch davon ausgehen, dass die Sicherheitskräfte auf Alarmmeldungen nicht mehr reagieren würden, wenn das Computersystem sie ohnehin im Minutentakt ausgeben würde. Die Begründung wäre dann, dass die minütlichen Alarmmeldungen ohnehin immer nur harmlose Passagiere anschwärzen.

Ein False Negative wäre im Gegenzug dann eine Anzahl womöglich gefährlicher Terroristen, die vom System fälschlicherweise als harmlos eingestuft werden, obwohl dafür pro Minute ein normaler Passagier aus dem Verkehr gezogen wird. Statistisch gesehen müssten bei einer Erkennungsquote von 99 % „nur“ 100 Terroristen versuchen, an Bord eines Flugzeuges zu gelangen – einer davon wird es schaffen. Wenn Sie in die Relation nun allerdings noch mit einbeziehen, dass aufgrund der im Minutentakt auftretenden Fehlalarme das Sicherheitspersonal gar nicht mehr auf jeden einzelnen Alarm eingeht, wird die False-Negative-Quote jedoch erheblich höher als 1 % ausfallen. Die 99 Terroristen, bei denen ein Alarm auftritt, haben in diesem Szenario plötzlich eine reelle Chance, dass ihr Alarm vom Personal ignoriert wird, „weil das System ohnehin im Minutentakt bei harmlosen Passagieren klingelt ...“.

Wie Sie sehen, hört sich eine Erkennungsquote von 99 % in der politischen Diskussion um das Für und Wider eines biometrischen Zugangskontrollsystems sehr reizvoll an – für einen sinnvollen Einsatz in der Praxis sind derartige Quoten jedoch viel zu schlecht.

False Positives und False Negatives berücksichtigen

Sollten Sie sich also für ein biometrisches System zur Authentifizierung und Autorisierung entschieden haben, lassen Sie sich vom Hersteller die Wahrscheinlichkeiten für False Positives und False Negatives vorlegen und rechnen Sie anhand Ihrer geplanten Zugriffe (z. B. Anmeldevorgänge pro Tag etc.) aus, wie viele Anmeldevorgänge pro Tag statistisch gesehen fehlschlagen bzw. wie viele unautorisierte Anmeldungen dennoch genehmigt werden. Beurteilen Sie dann anhand dieser Ergebnisse die Brauchbarkeit einer solchen Lösung.

Identitätskontrolle ist keine Gedankenkontrolle

Berücksichtigen Sie bei der Suche nach einer Authentifizierungslösung, dass auch ein gutes System nur die Identität einer Person bestimmen kann – und das auch nur, weil zum Beispiel ein Fingerabdruckmuster mit einem gespeicherten Muster auf einer Chipkarte auf Übereinstimmung geprüft wird –, aber nicht deren Absichten. Zur Sabotage- und Terrorabwehr ist Biometrie also ungeeignet.

Keines der hier vorgestellten Zugangskontrollverfahren – auch nicht die modernen biometrischen Systeme – kann vorhersagen, **WAS** eine Person zu tun beabsichtigt.

20

Proaktive Sicherheit

20.1 Defensive Programmierung

Fehler vermeiden

Ein Hauptgrund für Sicherheitslücken in Software sind Funktionsaufrufe, bei denen die übergebenen Parameter nicht oder unzureichend vom Programm getestet werden. Dies ermöglicht u. a. Buffer-Overflow-Angriffe, mit denen Hacker beliebigen Code mit den Rechten des angegriffenen Programms auf dem Zielsystem ausführen können. Ausführliche Informationen hierzu wurden bereits in Kapitel 4 beschrieben.

Insbesondere, wenn in Ihrer Firma Software selbst erstellt wird, können Sie die Sicherheit Ihrer Programme positiv beeinflussen. Sorgen Sie dafür, dass bei jeder Funktion, die innerhalb eines Projekts erstellt wird, die übergebenen Parameter auf Plausibilität und zulässige Länge geprüft werden, bevor die Daten weiterverarbeitet werden. Sicheres Software-Engineering kostet zwar bei der Erstellung der Software mehr Zeit, spart diese dann aber am Ende ein, da Anwendungen weniger Ausfallzeiten haben und hinterher weniger gepatcht werden muss.

Auch wenn Sie Ihre Applikationssoftware nicht selbst erstellen, sondern outsourcen, sollten Sie den Sourcecode der neuen Software unter den Gesichtspunkten der Plausibilität überprüfen lassen. Sollte Ihnen das Softwarehaus die Sourcecodes nicht zur Verfügung stellen, so sollten Sie in Ihrem Pflichtenheft einen Qualitätssicherungs-Test jeder Funktion vereinbart haben. Wichtig ist hier, dass die Funktionen nicht nur mit zulässigen, sondern auch mit möglichst vielen unzulässigen Parametern auf ihre Laufstabilität getestet wurden. Ein Softwarehaus, das sein Qualitätsmanagement im Rahmen der ISO 9000 zertifiziert hat, wird Ihnen nach der Erstellung der Software auch die entsprechenden Prüfberichte aushändigen.

Auch können Sie auf die Ergänzenden Vertragsbedingungen für die Pflege von Standardsoftware (EVB-IT Pflege S-AGB) Bezug nehmen. Der wichtigste Punkt innerhalb der EVB-IT Pflege ist:

„Der Auftragnehmer ist zur Erbringung der vereinbarten Pflegeleistungen (Pflege von Standardsoftware) mit Hilfe von automatisierten Verfahren nur dann berechtigt, wenn er im Angebot das zu verwendende Produkt benennt und gleichzeitig den Tatsachen entsprechend gewährleistet, dass dieses Produkt keine Kommunikationsfunktionen zu Dritten und keine andere, den Interessen des Auftraggebers zuwiderlaufende Funktionalität aufweist.“

Insbesondere darf das Produkt keine Funktionalitäten zum Ausspähen von Daten enthalten, keine Informationen über die IT-Systeme, deren Daten, deren Lizenzierung oder das Benutzerverhalten an Dritte übermitteln, zu anderen Zwecken als für die Erbringung der Pflegeleistungen oder derart speichern, dass Dritte darauf Zugriff nehmen könnten. Die Auswechslung bzw. der Einsatz eines neuen Releases des Produkts bedarf der ausdrücklichen Einwilligung des Auftraggebers im Einzelfall. Der Auftraggeber wird einwilligen, wenn der Auftragnehmer in Bezug auf das neu einzusetzende Produkt die oben genannte Gewährleistung übernommen hat. Liegen zureichende tatsächliche Anhaltspunkte dafür vor, dass das Produkt den vorgenannten Anforderungen nicht entspricht und kann der Auftragnehmer diese nicht ausräumen, kann der Auftraggeber den Einsatz des Produktes untersagen.“

Skripting

Selbst, wenn Software in Ihrem Unternehmen nicht selbst erstellt wird, werden Sie als Administrator einige Verwaltungsaufgaben mit diversen Skriptsprachen lösen. Versuchen Sie, bereits beim Programmieren möglichst viele Probleme und – soweit Benutzereingaben vorgesehen sind – Falscheingaben von Benutzern vorwegzunehmen und in Ihrem Skript zu berücksichtigen.

Skripte, die auf Benutzer-Rechnern ausgeführt werden sollen, sollten möglichst nicht mit Systemrechten laufen (wie zum Beispiel bei der Ausführung als Log-in-Skript einstellbar wäre). Schafft der Benutzer es in so einem Fall, die Ausführung des Skriptes anzuhalten, so verfügt er über eine Eingabeaufforderung mit Systemrechten.

Prinzip der minimalen Rechte

Bei der Verwaltung und Konfiguration von Software sollte das Prinzip der minimalen Rechte gelten: Jede Anwendung (und jeder Benutzer) sollte nur Zugriff auf die für seine Arbeit jeweils nötigen Ressourcen erhalten. So wird das Schadenspotenzial durch einen Hackerangriff im Applikationsbereich umfassend eingeschränkt.

20.2 Gehärtete Betriebssysteme

Non-Executable Stack

Eine der besten Waffen gegen Buffer-Overflow-Angriffe ist, dass Code auf dem Stack gar nicht erst ausgeführt wird. Dies ist allerdings eine Aufgabe, die nur auf Betriebssystemebene erledigt werden kann. Allen Buffer-Overflow-Angriffen gemeinsam ist, dass der Shellcode zusammen mit den Daten übertragen wird, die den Stack zum Überlaufen bringen und die Ausführung des Shellcodes erzwingen. Prüft das Betriebssystem grundsätzlich vor dem Ausführen von Code, wo sich dieser befindet, so kann Code auf dem Stack gezielt blockiert und das entsprechende Anwendungsprogramm beendet werden.

Bei der Verwendung von **Java** anstelle C/C++ überwacht diese standardgemäß zur Laufzeit die Grenzen der Speicherbereiche und entlastet somit den Softwareentwickler. Auch C/C++ hat Optionen, diese Sicherheitslücken zu minimieren.

Microsoft-Betriebssysteme bieten diese Möglichkeit der Sicherung ab Windows XP 64 Service Pack 2. Auch sind modifizierte Linux- und UNIX-Systeme verfügbar, die die Ausführung von Code auf dem Stack verhindern. Da es auch legitime Software gibt, die Code-teile zur Ausführung auf dem Stack platziert, ist diese mit einem Non-Executable Stack nicht mehr lauffähig. Darum sollten Sie die nötige Software vor dem produktiven Einsatz erst eingehend auf Lauffähigkeit unter einem Non-Executable Stack prüfen.

Nur benötigte Services

Wollen Sie einen Rechner als Webserver einsetzen, so sollten Sie alle nicht unbedingt benötigten Services auf diesem Rechner deaktivieren. Das verringert die Wahrscheinlichkeit, dass etwaige in diesen Diensten vorhandene Sicherheitslücken von Angreifern ausgenutzt werden können.

Appliances

Der Sicherheitsvorteil von Software/Hardware-Appliances liegt darin, dass die Hersteller in diesen Geräten speziell modifizierte Versionen eines Betriebssystems implementieren. Diese sind ausschließlich auf die jeweilige Anwendung bzw. Hardware zugeschnitten und besitzen keine unnötigen Dienste. Sie sind somit das Hardware-Äquivalent zum Deaktivieren von Services auf normalen Rechnern. Vor allem bei Komponenten, die Sie zur Sicherung selbst einsetzen, sollten Sie darauf achten, dass die Appliances (z. B. Hardware-Firewall oder VPN-Gateway) über ein gehärtetes Betriebssystem und einen gehärteten TCP/IP Stack verfügen.

Beispielsweise wird hierfür SELinux (Security-Enhanced Linux) verwendet. Es besteht aus einem modifizierten Kernel-Patch und den ausschließlich notwendigen Applikationen. Die Zugriffs-kontrolle auf die Ressourcen erfolgt über definierte Regeln (Policies). Durch das Regelwerk werden Angriffe auf das System und die Applikation wirksam unterbunden. Sofern Sie weitere Applikatio-nen innerhalb der Appliance nutzen wollen, müssen Sie diese als SELinux-Policy-Pakete hinzufügen, um die definierten Sicherheitsrichtlinien weiterhin zu erhalten. Hier einige Links dazu:

- ✓ <https://www.nsa.gov/what-we-do/research/selinux/>
- ✓ <https://fedoraproject.org/wiki/SELinux>
- ✓ <https://software.opensuse.org/search?utf8=%E2%9C%93&baseproject=ALL&q=selinux>
- ✓ <https://www.ubuntuupdates.org/package/core/cosmic/universe/base/selinux>
- ✓ <https://linuxwiki.de/SELinux>

Im Bereich der Windows-Plattform können Sie als Appliance Sandboxie verwenden. Dieses Programm realisiert die entkoppelte Ausführung einer Applikation im Betriebssystem, wobei alle verändernden Zugriffe dabei in einen definierten Ordner umgelenkt werden. Dies bietet sich besonders für E-Mail-Programme, Browser bzw. neu zu testende Applikationen an. Bei einer Infektion durch Malware wird diese nur in der Sandbox aktiv und nicht in Ihrem System. Ein Löschen des Inhaltes des Ordners beseitigt auch die Malware (weitere Informationen und Down-load unter <https://www.sandboxie.com/DownloadSandboxie/>).

Um den Sicherheitsvorteil der Appliance jederzeit aufrechtzuerhalten, sollten Sie auch hier regelmäßig die entsprechenden Updates einspielen und deren Funktion überprüfen.

20.3 Patches

Fehler in Software

Auch wenn viele Hersteller von Software und Hardware es nicht gerne hören: Ihre Produkte werden von Menschen entwickelt, die bekanntlich Fehler machen. Und früher oder später werden diese Fehler gefunden. Die Vorgehensweisen der Hersteller, wenn ein Fehler in ihrem Produkt bekannt geworden ist, sind unterschiedlich. Einige beginnen damit, die Schwachstelle zu suchen und eifrig Patches zur Verfügung zu stellen. Andere Hersteller versuchen, die gefundene Lücke zu bagatellisieren, sie zu dementieren oder sogar den Entdecker der Lücke wegen Copyright-verletzung (Disassemblieren des Produktes) anzuklagen und ihm mutwillige Schädigung zu unterstellen.

Obwohl es zweifelsfrei nicht in Ordnung sein kann, direkt nach dem Entdecken einer Sicherheitslücke ihre detaillierte Beschreibung sowie ein lauffähiges Programm zur Ausnutzung dieser Lücke zu veröffentlichen, sollten Sie sich folgendes zeitliches Schema vor Augen halten:

- ✓ Eine Lücke wird entdeckt.
- ✓ Ist der Entdecker ein White-Hat, kontaktiert er den Hersteller bzw. macht den Fehler ohne Herstellereinbindung öffentlich.
- ✓ Der Hersteller behebt das Problem und verteilt einen Patch.
- ✓ Sie installieren den Patch und sind wieder auf der sicheren Seite.

Da es aber nicht nur White- und Grey-Hats gibt, ist folgendes Schema wahrscheinlicher:

- ✓ Ein Black-Hat entdeckt eine Lücke.
- ✓ Es wird ein Exploit entwickelt.
- ✓ Der Black-Hat nutzt die Lücke, um Systeme selbst zu kompromittieren.
- ✓ Der Exploit wird im kriminellen Untergrund (u. a. im Darknet) an andere Black-Hats verkauft, um so zusätzlich Gewinn zu generieren.
- ✓ Irgendwann kommt der Exploit wegen einer undichten Stelle an die Öffentlichkeit, oder ein White-Hat entdeckt diese Lücke unabhängig, oder Untersuchungen bei bereits kompromittierten Systemen offenbaren die Lücke.
- ✓ Erst jetzt hat der Hersteller Kenntnis von der Lücke und kann diese beheben oder alternativ die Lücke kleinreden bzw. ggf. den White-Hat verklagen, der sie publik gemacht hat.

Wie Sie sehen, gibt es in diesem System ein Problem. Ein Black-Hat, der eine Lücke findet, wird mit diesem Wissen nicht an die Öffentlichkeit gehen, sondern sie so lange wie möglich ausnutzen. Das heißt, dass es selbst dann, wenn ein Produkt offiziell keine bekannten Lücken hat, theoretisch möglich ist, dass bereits (unveröffentlichte) Schwachstellen ausgenutzt werden. Diese in der Öffentlichkeit noch unbekannten Lücken werden im kriminellen Untergrund als sogenannte 0-Day-Exploits (gesprochen: Zero-Day) gehandelt und erzielen auf dem Schwarzmarkt hohe Preise, da ein 0-Day-Exploit quasi ein Generalschlüssel zu jedem System ist, das dieser Exploit betrifft – auch zu denjenigen, auf denen alle verfügbaren Patches installiert wurden.

Erst ab dem Zeitpunkt, zu dem ein unabhängiger White- oder Grey-Hat dieselbe Lücke entdeckt, bzw. dann, wenn bei der Revision von sich auffällig verhaltenden Systemen die betreffende Lücke als Einstiegspunkt gefunden wird, können Sie Gegenmaßnahmen ergreifen.

Out-of-the-Box

Aus den oben genannten Gründen sollten Sie eine Netzwerkkomponente, die Sie in Ihrem Unternehmen installiert haben, nicht als Out-of-the-Box nutzen, sondern während der Installation die verfügbaren Aktualisierungen einspielen. Obwohl manche Hersteller ihre Produkte mit einer „Fire-and-Forget“-Strategie bewerben, sollten Sie dennoch während der gesamten Einsatzdauer des Produktes in regelmäßigen Abständen überprüfen, ob neuere Patches zur Verfügung stehen.

Niemand wird behaupten, dass dieser regelmäßige Aufwand keine Zeit oder kein Geld kostet. Dies sind jedoch Kosten, die im TCO (Total Cost of Ownership) berücksichtigt sein sollten. Sicherheit ist auch ein Managementproblem. Solange für IT-Sicherheit keine definierten Budgets eingeplant sind und deswegen bei Auswahl und Betrieb von Komponenten die günstigste Lösung gewählt werden muss, werden diese Systeme nicht die sichersten sein.

Würde die TCO-Kalkulation eines IT-Systems die potenziellen Schäden enthalten, die beim Ausfall einer Komponente durch unterlassene Patches, falsche Konfiguration oder unsichere Programmierung entstehen können, so würden regelmäßig gewartete Systeme deutlich günstiger abschneiden.

20.4 Vulnerability Assessment

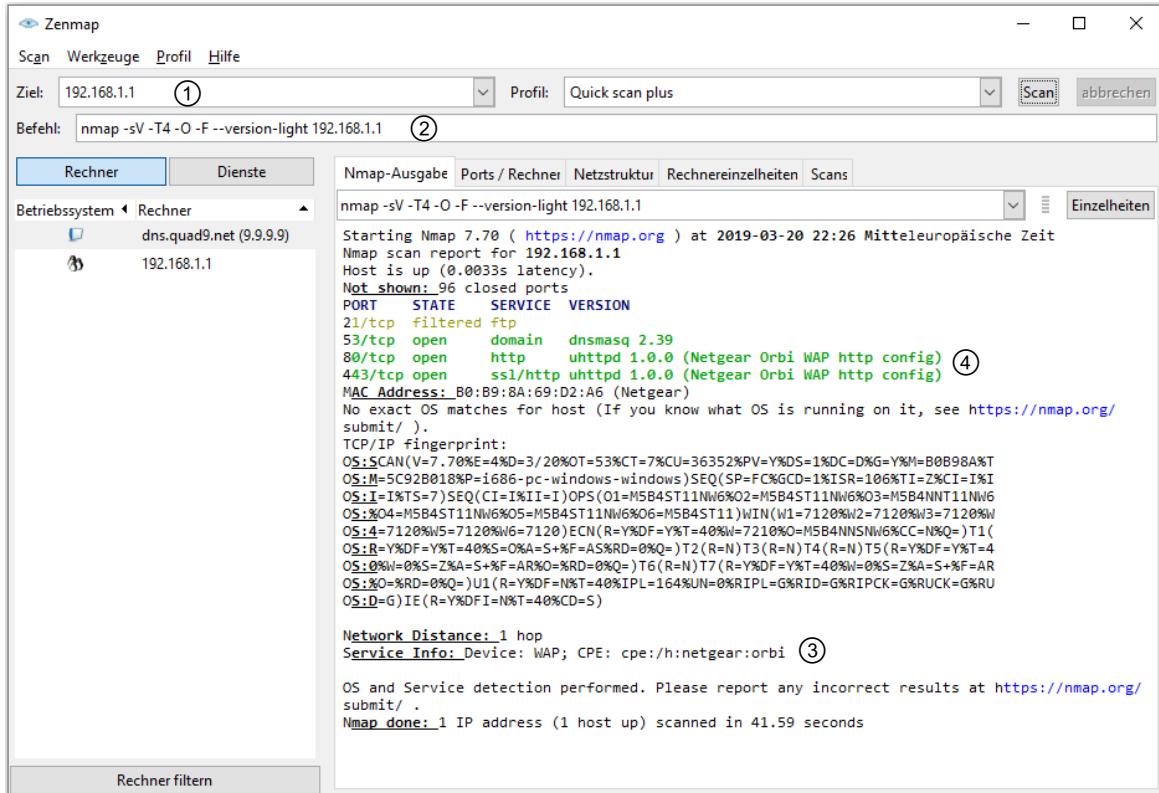
Wie sicher sind Ihre Client/Server-Systeme wirklich?

Ob sich Schwachstellen in Ihren Systemen befinden, die ein Hacker für seine Zwecke ausnutzen könnte, erfahren Sie am besten, wenn Sie Ihr Netz aus der Sicht eines Hackers betrachten. Gängige Praxis zur Überprüfung von Netzwerken ist, mit den von Hackern und Crackern verwendeten Scannern und Tools selbst eine Attacke zu starten und die so entdeckten Sicherheitslücken dann über vorhandene Patches zu schließen.

Die Möglichkeiten, ein Vulnerability Assessment oder einen sogenannten Schwachstellentest durchzuführen, sind vielfältig und hängen von den zu testenden Komponenten ab. Die hierbei verwendeten Scanner können wie folgt eingeteilt werden:

- ✓ **Data-Link-Sniffer**, sie arbeiteten auf Layer 2 des TCP/IP-Referenzmodells und ermöglichen z. B. das Erkennen von vorhandenen MAC-Adressen, das Erkennen von PPP-Passwörtern, das Finden von WLAN-SSIDs und weiterer WLAN-Parameter.
- ✓ Der **IP-Scanner** arbeitet auf Layer 3 und ermittelt aktive IP/IPv6-Adressen im Netz.
- ✓ Der **Port-Scanner** findet gestartete Applikationen (offene Ports) auf den Systemen und nutzt hierbei die Funktionalität der Schicht 4 des TCP/IP-Referenzmodells aus.
- ✓ Ein **Password-Recovery-Tool** versucht, Passwörter zu entschlüsseln und nutzt dabei die Protokolle der Applikationsschicht.
- ✓ **Applikation-Scanner** arbeiten ebenfalls auf der Applikationsschicht und überprüfen neben den Applikationsfunktionen auch den Dateninhalt.
- ✓ **Remote-Control-Tools** (z. B. TeamViewer oder AnyDesk) übernehmen die teilweise oder vollständige Steuerung eines entfernten Systems mit den Rechten, die der angemeldete Nutzer hat.

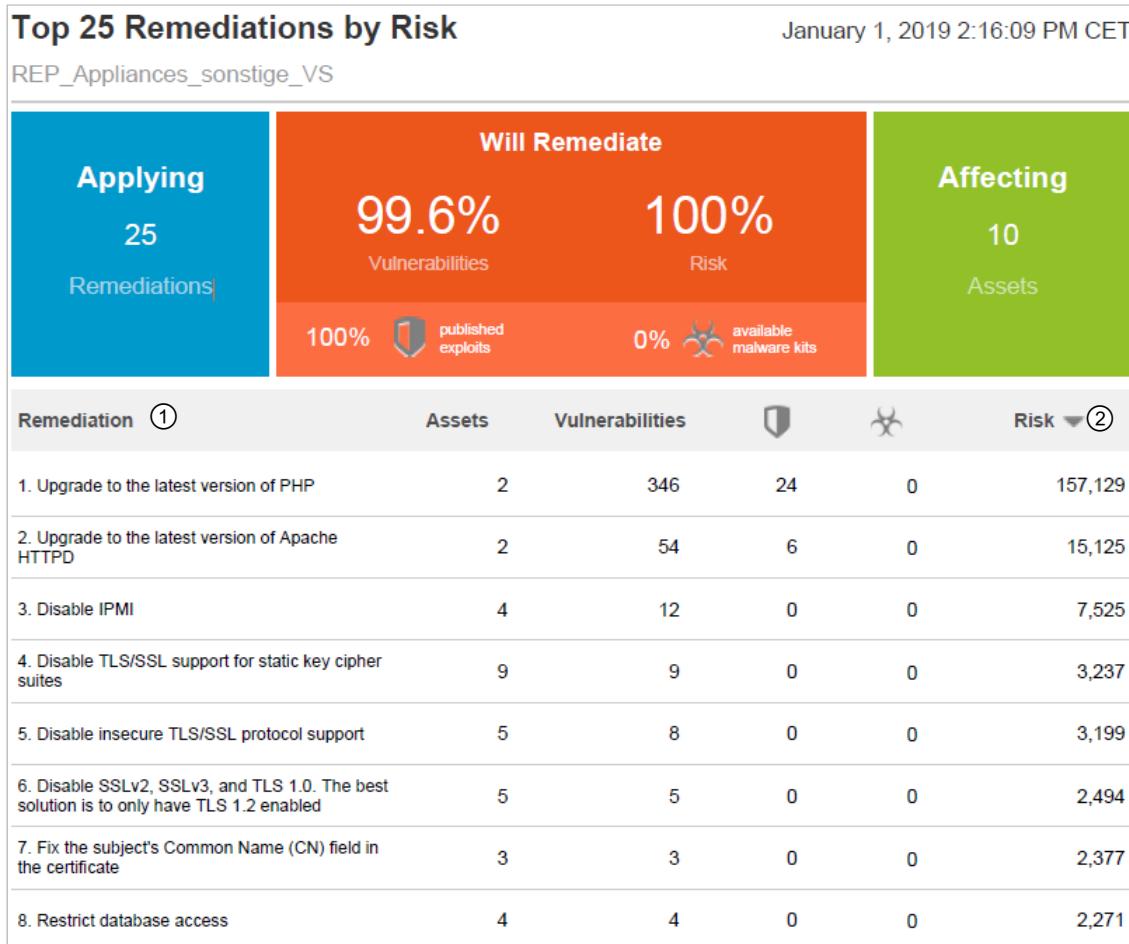
Ein einfacher und beliebter Scanner ist Nmap (**Network mapper**). Er analysiert Hosts ① oder Netzwerke. Je nach verwendeter Befehlsoptionen ② erhalten Sie Informationen u. a. über das genutzte Betriebssystem, den Hersteller ③ oder die vorhandenen Dienste ④. Die Möglichkeiten sind dabei aber noch nicht ausgeschöpft. Der Leser findet unter <https://nmap.org> den passenden Reference Guide.



Port-Scanner Nmap und GUI Zenmap im Einsatz

Der Einsatz von Vulnerability Scannern beschränkt sich jedoch nicht nur auf Netzwerke. Das Einsatzszenario erstreckt sich von der Datenbankanalyse, über Prozess- und Leitsysteme bis zu Unternehmensprozessen. Wir bleiben aber bei unseren Netzwerken.

Professionelle Vulnerability Scanner (z. B. Nessus [<http://www.nessus.org/products/nessus>] oder Rapid7) nutzen neben beschriebenen Funktionalitäten auch diverse Schwachstellen-Datenbanken (z. B. CERT, Symantec DeepSight, Mitre oder Seclists) zur Analyse und Korrelation. Auf dieser Datenbasis werden dann gezielt gestartete Prozessen verifiziert und auf Schwachstellen kontrolliert. Dazu werden auch echte Exploits auf die zu testenden Systeme angewendet oder nur Versionsinformationen oder bestimmte Reaktionen der anfälligen Prozesse ausgewertet.



Rapid7 Nexpose, Auszug aus einen Schwachstellenscan

Der Vulnerability Scan von Rapid7 zeigt auszugsweise die erkannten Schwachstellen ① in einer Systemumgebung und deren Risiko ② auf der Basis der Datenbankkorrelation.

Die für diese Zwecke bestimmte Software ist in einem breiten Spektrum erhältlich. Entweder als teure kommerzielle Lösung, lizenfreie GNU Public License Software oder sogar als „echtes“ Tool, das aus dem Hacker-Untergrund stammt – alle können sie für denselben Zweck eingesetzt werden. Hier eine Auswahl:

- ✓ Nessus
- ✓ McAfee VM
- ✓ QualysGuard
- ✓ Nmap
- ✓ Retina
- ✓ Nexpose Rapid7
- ✓ OpenVAS
- ✓ CoreImpact
- ✓ Saint
- ✓ Greenbone

So, wie Hacker und Cracker auch kommerzielle und Open-Source-Produkte einsetzen, um bei der Planung ihrer Angriffe geeignete Systeme zu finden, können Sie genau dieselbe Software verwenden, um Ihr Netz zu schützen. Solange Sie vor einem Hacker feststellen, dass Ihr Netz eine undichte Stelle hat, sind Sie im Vorteil.

Obwohl Sie vielleicht ein Hacker-Tool aus dem Untergrund kostenlos beziehen können und dieses seine Arbeit verrichtet, sollten Sie sich überlegen, ob Sie die Sicherheit Ihres Netzwerks einem Programm unbekannter Herkunft anvertrauen möchten.

Schließlich ist nicht dokumentiert, ob das Tool nicht ein trojanisches Pferd ist oder ob es Ihnen nur die Sicherheitslücken anzeigt, die der Hacker nicht selbst auszunutzen plant.

So, wie Sie mit entsprechenden Softwaretools die Sicherheit von Netzwerkkomponenten testen können, lassen sich auch die von den Benutzern verwendeten Passwörter auf ihre Sicherheit testen. Dictionary- oder Brute-Force-Programme gehen wie ein Hacker systematisch vor und liefern dem Auditor eine Auflistung aller Accounts, deren Passwort zu entschlüsseln war.

Rechtliche Absicherung

Bevor Sie ein Netzwerk auf Schwachstellen abklopfen oder z. B. beginnen, mit Programmunterstützung Passwörter zu knacken, bedenken Sie die rechtlichen Konsequenzen Ihres Handelns. Haben Sie keine Befugnis für derartige Tests bzw. nicht das Einverständnis des Netzwerkeigentümers oder eine entsprechende Regelung in der Security Policy des Unternehmens, können Sie sich durch Ihre unautorisierten Hackversuche strafbar machen. In diesem Fall hätten Sie nach einem erfolgreichen Einbruch große Schwierigkeiten, nachzuweisen, dass Sie Ihr Wissen bzw. erlangte Systemrechte nicht illegal ausgenutzt haben.

Der im Mai 2007 vom Bundestag verabschiedete Zusatzparagraf § 202c des Strafgesetzbuchs – im Volksmund als sogenannter „Hackerparagraf“ bekannt – hat leider auf dem Gebiet der Vulnerability-Analyse und der Sicherheitsaudits anstatt für mehr Sicherheit für mehr Rechtsunsicherheit gesorgt.

Dem Wortlaut des Paragrafen zufolge werden die Herstellung, der Vertrieb, die Überlassung oder sonstige Verbreitung von Programmen, die geeignet sind, Computer auszuspähen oder Zugangsdaten zu knacken, unter Strafe gestellt.

Aufgrund der vagen Formulierung betrifft dies bei buchstabengetreuer Auslegung auch Hersteller und Nutzer von Sicherheitstools, da deren Zweck ja eben das Aufspüren derartiger Sicherheitslücken ist. Deswegen steht dieser Paragraf bei IT-Branchenverbänden und Experten berechtigterweise unter heftiger Kritik. Diese Rechtsunsicherheit hat schon nachweislich dazu geführt, dass Firmen, die in der Softwareentwicklung in der Sicherheitsbranche tätig sind, ihre Tätigkeiten vorsichtshalber ins Ausland verlagert haben.

Wie jedoch Administratoren und Sicherheitsberater die Sicherheit von Netzwerken und Systemen überprüfen sollen, wenn der Besitz, die Herstellung und die Verbreitung entsprechender Software unter Strafe stehen, konnte die Politik bis jetzt noch nicht plausibel beantworten. Obwohl einige mutige Sicherheitsexperten durch eine Selbstanzeige ein Exempel statuieren wollten (wie z. B. ein iX-Chefredakteur Ende 2008) und diese Verfahren von der Staatsanwaltschaft abgelehnt oder bald wieder eingestellt wurden, sollten Sie dennoch zu Ihrer eigenen Absicherung auf jeden Fall einen entsprechenden schriftlichen Arbeitsauftrag Ihres Arbeitgebers einholen.

Regelmäßige Überprüfung

Wie bei allen Aufgaben in der IT-Sicherheit gilt auch für Vulnerability Assessments: Die Ergebnisse, die Ihnen ein solcher Test liefert, können nur dann eine konkrete Aussage liefern, wenn die angewandten Tests aktuell sind. Ebenso sollten Sie in Ihrem Netzwerk diese Sicherheitschecks regelmäßig durchführen. Auf diese Weise können Sie sich eine Datenbasis aufbauen und so überprüfen, ob zu einem früheren Zeitpunkt entdeckte Lücken ordnungsgemäß geschlossen werden konnten und ob sich eventuell neue aufgetan haben.

Wie sicher sind Ihre Netzwerkkomponenten wirklich?

Bisher wurden in diesem Buch die Sicherheitsaspekte primär aus der Sicht der Client/Server- und Firewall-Systeme betrachtet. Eine umfassende Sicherheitspolitik muss sich auch der lokalen Netzinfrastruktur (Switche, Access-Points, Multilayerswitches und Router) annehmen, die die Vernetzung der Systeme erst ermöglichen. Besonders in mittleren und größeren Netzwerken muss sie integraler Bestandteil der Sicherheitsphilosophie sein.

Dieser Sicherheitskontext beinhaltet:

- ✓ den Infrastruktur-Device-Zugriff
- ✓ die lokale Switching-Infrastruktur
- ✓ die lokale Routing-Infrastruktur
- ✓ die Komponenten-Verfügbarkeit und -Sicherheit

Infrastruktur-Device-Zugriff

Genauso, wie Sie den sicheren Zugriff auf Client/Server-Systeme mittels Passwörtern bzw. Authentifizierungsverfahren gestalten, müssen Sie dies auch für den administrativen Zugriff auf die Benutzeroberfläche der Netzwerkkomponenten handhaben. Die auf den Komponenten installierte Firmware gibt Ihnen hierzu umfassende Möglichkeiten. Folgende Funktionen stehen Ihnen, in Abhängigkeit von der implementierten Firmware, zur Verfügung:

- ✓ Zugriff auf das Device bei fehlerhafter Eingabe terminieren bzw. Sitzung automatisch bei Inaktivität unterbrechen
- ✓ Den administrativen Zugriff auf das Gerät nur von bestimmten IP-Adressen (Filtern mittels Access List) oder über das AAA-Protokoll erlauben
- ✓ Administrative Rechte (Kommandoprivilegien) für verschiedene Administratoren festlegen
- ✓ Die Kommunikationsverbindung über das SSH-Protokoll (min. Version 2) verschlüsseln
- ✓ Die Protokollierung (Logging) aller administrativen Zugriffe; das Logging sollte nicht lokal erfolgen

Lokale Switching-Infrastruktur

Die Sicherheit und Performance in Layer-2-Netzwerken hängt maßgeblich von der konsequenten Umsetzung der Sicherheitsfeatures der Data-Link-Protokolle ab. Die Sicherheitsaspekte für WLAN (WLAN-Netzwerkkarte, Access-Point) wurden bereits in Kapitel 18 beschrieben, deshalb werden im Weiteren primär Switche betrachtet.

Ein Switch flutet Multicast- und Broadcast-Frames auf allen Ports. Dieser Mechanismus wird als Broadcastdomäne bezeichnet. Hierdurch kann ein Performanceproblem entstehen. Sie können die Broadcastdomäne und damit den Verkehr in diesem Segment durch die Bildung von VLANs (**Virtual Local Area Network**) einschränken. Dies erfordert:

- ✓ Aufteilung des Userverkehrs in mehrere VLANs,
- ✓ Deaktivieren der Nutzung des Default-VLANS (VLAN 1)
- ✓ Nutzung von privaten VLANs, um die Kommunikation der User untereinander zu unterbinden
- ✓ Begrenzung des VLANs auf einen Switch oder Switch-Stack,
- ✓ Reduzierung des Multicast-Verkehrs durch Nutzung von IGMP (**Internet Group Management Protocol**).

Bei der Nutzung des Spanning Tree Protocols (STP), das Loops im Layer-2-Segment vermeidet, müssen Sie darauf achten, dass keine STP-Informationen über Access-Ports der User verbreitet und empfangen werden können. Damit eliminieren Sie das Risiko eines Angriffs auf Spanning Tree und Inkonsistenzen im lokalen Netzwerk.

Damit Unbefugte nicht auf Ihr Netzwerk zugreifen können, sollten Sie alle nicht genutzten Switch-Ports deaktivieren und für die Autorisierung der vorhandenen User Port-Security oder IEEE 802.1X aktivieren. Diese Protokolle ermöglichen eine Zugriffskontrolle anhand der User-MAC-Adresse bzw. die Authentifizierung über einen RADIUS-Server und somit z. B. nur die ausschließliche Nutzung von firmeneigenen Systemen im Netzwerk.

Die Lokale Routing-Infrastruktur

Die Absicherung des Internet-Protokolls (IP/IPv6) erfolgt über klassische Router oder Multilayer-schalter. Dabei können Sie durch das Setzen von IP-Filtern (Access-Listen oder Paket-Filter) den Verkehr zwischen Netzwerken reglementieren. Entsprechende Beispiele finden Sie in Abschnitt 14.2.

Bei der Nutzung von dynamischen Routingprotokollen (z. B. RIP, OSPF, BGP, IS-IS) sollten Sie die sicherheitstechnischen Protokollparameter anwenden, wie:

- ✓ Nutzung eines Routingprotokolls mit den besten Routing-Parametern z. B. OSPF (**Open Shortest Path First**);
- ✓ Deaktivieren der Routing-Updates an den User-Ports; dadurch wird beim Sniffen in dem Netzwerk die Bekanntgabe der Netzstruktur verhindert;
- ✓ Authentifizierung der dynamischen Routing-Updates von den Nachbar-Routern;
- ✓ Logging aller dynamischen Routingänderungen.

Die Komponenten-Verfügbarkeit und -Sicherheit

Der letzte Punkt behandelt kurz die Aspekte der Appliances (s. a. Abschnitt 20.2) und Redundanz von Netzwerkgeräten. Auch auf diesen Komponenten ist es erforderlich, nicht benötigte Dienste (z. B. TELNET und HTTP bei Nutzung von SSH oder HTTPS) zu deaktivieren und somit Angriffs-möglichkeiten auf die Infrastruktur einzuschränken. Jeder Prozess, der inaktiv ist, ist ein poten-zielles Risiko weniger.

Verfügbarkeit (Redundanz) ist ein wichtiger Bestandteil in einem Sicherheitskontext. Wenn bestimmte Dienste im Netzwerk nicht mehr erreichbar sind, ist mitunter der wirtschaftliche Schaden genauso hoch wie bei einem Virenangriff.

Protokolle und Hardware, die eine Verfügbarkeit garantieren, sind:

- ✓ redundante Netzteile bei systemkritischen Komponenten,
- ✓ redundante Switch/Router-Hardware (z. B. Switch-Stack und virtueller Switch/Router),
- ✓ redundante physikalische Verbindungen (u. a. IEEE 802.1AX Link Aggregation),
- ✓ redundante Routingprotokolle (z. B. **Virtual Router Redundancy Protocol VRRP**),
- ✓ Software-defined Networking (SDN) als Virtualisierung der Netzwerkdienste,
- ✓ Software-defined Storage (SDS) als Virtualisierung der Speichersysteme.

Die Nutzung der beschriebenen Maßnahmen in Abhängigkeit von der vorhandenen Hard- und Firmware erhöht signifikant die Sicherheit in Ihrem Intranet und die Kommunikation mit dem Internet.

0		B		Cookie	71
0900-Dialer	68	Backdoor	68	Core Wars	66
0-Day-Exploits	225	Bannergrabbing	25	Cracker	18
3		Baran, Paul	14	CVE Numbering Authority, CNA	50
3DES	120	Bastion-Host	164	D	
8		Biometrie	218	Darknet	225
802.11i	197	BIOS	54	Darwin	66
A		Black-Hat	18	Dateianhang, Virus	56
Access Control List, ACL	166	Blockchiffre	116	Dateiedungen anzeigen	85
Access-Points	192, 195	Blowfish	120	Dateiviren	56
Access-Points, Platzierung	200	Bootsektoviren	54	Datenpanne, Folgen	16
Access-Points, Reichweite	200	Botnetz	39, 53, 68	Datenschutz	6
ACK	22	Bots	39	Datensicherheit	6
ACL	166	Bridging-Modus	193	Datensicherheit, gesetzliche Grundlagen	9
Advanced Encryption Standard, AES	120, 198	Browser Hijacking	76, 77	DDoS	23, 39
Advanced Thread Detection	172	Browser, Sicherheits- einstellungen	89	DDoS, Smurf-Attack	39
Adware	69	BSI	10, 13, 69, 70	Deep Packet Inspection	167
AES	120, 198	BSI-Standard	100	Defense Information Systems Agency, DISA	175
AH	151, 186	Buffer Overflow	33	Defensive Programmierung	222
Alternative E-Mail-Clients	89	Bullrun	19	Demilitarisierte Zone, DMZ	164
Alternatives Betriebssystem	205	Bundesamtes für Sicherheit in der Informations- technik, BSI	10, 13, 69, 70	Denial of Service	23
Anomalieerkennung	175	BVB	13	Denial of Service, DoS	39
Antiviren-CD	96	Byte Substitution, ByteSub	121	DES	116, 117
Antivirus on Access	93	C		Data Encryption Standard, DES	116, 117
Antivirus on Demand	93	CA	142	DH-Schlüsseltausch	131
Antivirus, Auswahlkriterien	92	Caesar-Chiffre	110	Dialer	68
Antivirus, Client/Server	94	CAST	120	Diffie, Whitfield	127
Antivirus, Computer scannen	95	CBC	124	Diffie-Hellman	150, 199
Antivirus, Echtzeitschutz	93, 97	CENELEC	103	Diffie-Hellman-Schlüsseltausch	128
Antivirus, Erkennungsquote	92	CERT	50	Digital Signature Algorithmus, DSA	132
Antivirus, Heuristik	93	Certificate Authority, CA	142, 148	Digitale Signatur	136
Antivirus, Testlabore	92	CFB	124	Direct Sequence Spread Spectrum, DSSS	193, 194
Antivirussoftware	91	Challenge Handshake Authenti- cation Protocol, CHAP	215	DISA	175
Appliances	164, 224	Challenge-Response-Verfahren	146	Distributed Denial of Service, DDoS	39
Application Control	167	CHAP	215	Distributed-Reflected-Denial-of- Service-Attacke, DRDoS	40
Application Level Firewall	168	Chosen Ciphertext	139	DMZ	164
Armored Code	63	Cipher Block Chaining, CBC	124	Domänenzertifikat	148
Armored Viren	63	Cipher Feedback, CFB	124	DoS	23, 39, 178
ARPA	14	Claws-Mail	155	DSA	132
ARP-Spoofing	43	COBIT	102	DSSS	194
Assets	99	Cohen, Fred	52	Dumpster Diving	28
Asymmetrische Schlüssel	135	Common Criteria	101	E	
Asymmetrische Verfahren	127	Common Vulnerabilities and Exposures, CVE	50, 70	EAP	215
Authentication Header, AH	151, 186	Common Vulnerability Scoring System	50	ECB	123
Authentication Server	212	Computer Emergency Response Team, CERT	50	Edge, Sicherheitseinstellungen	90
Authenticity	7	Computer scannen	95	Einwegfunktion	127
Authentifikation	7	Confidentiality	6	Electronic Codebook	123
Availability	8				
avast!, Antivirus	95				

Elektronische Kriegsführung	21	Gpg4win	155	ISO 9000	102, 222
EI-Gamal	132	GpgEX	155	ISO/IEC 13335	100
E-Mail-Clients, alternative	89	GpgOL	155, 161	ISO/IEC 15408	101
Encapsulating Security Payload, ESP	151, 186	GRE	185	ISO/IEC 19790	101
End-to-End-Verschlüsselung	153	Grey-Hat	18	ISO/IEC 2700X	101
Ergänzenden Vertragsbedingungen für die Beschaffung von Informationstechnik	13	Hacking	18	IT-Grundschutz-Kompendium	99
Erpressung	48	Half-Open Scan	23	ITIL	102
Erzeugendes Element	130	Handshake, 3-Way	21	ITSEC	101
ESP	151, 186	Hashfunktion	136, 137	IT-Sicherheitsbeauftragter	105
EU-Datenschutzgrundverordnung	12	Hellman, Martin E.	127	IT-Sicherheitsgesetz	9
EU-Datenschutzrichtlinie	12	Heuristik	64	IT-Sicherheitsstandards	99
Exklusives ODER	118	High-Interaction-Honeypots	181	IV	124
Expansionspermutation	118	Hijacking	42, 43	J	
Exploit	31	Hintertür	47	JavaScript-Viren	59
Exploitcode, Aufbau	34	Honeynet	180, 181, 182	K	
Exponentielfunktion	129	Honeypot	180, 181	Kasiski, Friedrich Wilhelm	114
Extended Validation TLS	145	IDEA	120	Kaskadeneffekt	119
Extensible Authentifikation Protocol, EAP	215	IDS	174, 175	Kaspersky Security Bulletin	69
Extranet-VPN	184	IDS, hostbasiertes	176	KDC	187
F		IDS, netzwerkbasiertes	176	Kerberos	187, 211
Failover	173	IEEE 802.11i	198	Kerberos Key Distribution Center, KDC	187
False Negatives	221	IEEE 802.1X	216	Kerberos-Tickets	212
False Positives	221	IEEE802.1X	199	Kerckhoffs	108
False-Negative-Rate	220	IKE	151, 187	Key-Server	141
Fernzugriff	68	Independent WLAN	192	Kleopatra	155
FHSS	194	Infektionsroutine	53	Kriterienwerke für IT-Sicherheit	99, 103
FIN Scan	25	Information Security Awareness	51	Kritische Infrastruktur	10, 11
Fingerabdruck-Scanner	219	Infrastructure WLAN	192	Kryptoanalyse	107
Fingerprinting	25	Initial Permutation	117	Kryptografie	107
Fingerprinting, passives	26	Initialisierungsvektor, IV	124	Kryptologie	107
FIPS 140-2	101	Integrität	7	L	
FIPS 197	120	Integrity	7	L2F	185
Fire-Back	178	Internet Key Exchange, IKE	151, 187	L2TP	185
Firefox	90, 207	Internet Security Association and Key Management Protocol, ISAKMP	151, 187	Layer 2 Forwarding, L2F	185
Firewalls	163, 164	Internet, Entwicklung	14, 15	Layer 2 Tunneling Protocol, L2TP	185
Five Eyes	19	Internet-Würmer	66	Link-Balancing	173
Frequency Hopping Spread Spectrum, FHSS	194	Intranet-VPN	184	Low-Interaction-Honeypots	181
Funktionssicherheit	6	Intrusion-Detection-System	174	M	
G		Intrusion-Detection-System, IDS	175	MAC-Filterung	196
Geburtstagsangriff	140	Intrusion-Prevention-System, IPS	178	macOS	205
Generic Proxy	169	Intrusion-Reaction-System	178	Makroviren	57
Generic Routing Encapsulation, GRE	185	IP Security	151	Malware	52
Gesetzliche Grundlagen	9	IPS	178	Man-in-the-middle-Angriff	43
Gnu Privacy Assistant, GPA	155	IPsec	151, 186, 189	Man-in-the-Middle-Attack	139
Gnu Privacy Guard, GnuPG	154	IP-Spoofing	40	Manipulation	48
GnuPG	154	IRS	178	MARS	120
Gnu-Privacy-Projekt	154	ISAKMP	151, 187	MD5	137
GPA	155	ISN	22, 42		

Meltdown	70	Perimeternetzwerk	164	Restwert	128		
Message Digest 5	137	Personal Firewall	170	Reverse Social Engineering	29		
Microsoft Defender	81	PGP	142	Rijndael	120		
Microsoft Outlook sichern	88	Phishing	29, 77	Risikomanagement	105		
MixColumns	121	Phishing-Mails	79	Roaming	193		
Modul	128	PKCS	136	Root-CA	143		
Modulo-Operation	128	PKI	141	Rootkit	38		
Monokultur	204	PMail	208	ROT	111		
Morris, Robert T.	66	Point-to-Point-Protocol, PPP	184	ROT-13	112		
MPLS	190	Point-to-Point-Tunnelling-Protocol	184	RSA	133		
N							
NAT	169	Policy	163	S			
Nationale Cyber-Sicherheitsrat, NCS	13	Policy, Security	104	S/MIME	153		
Nationales Cyber-Abwehrzentrum, NCAZ	13	Polymorphe Viren	63	SA	187		
Network Address Translation, NAT	169	Port-Forwarding	151	Safety	6		
Netzwerke, soziale	78	Portscan	21	Sandboxie	224		
Netzwerkscan	21	PPP	184	S-Box	119		
Neumann, John von	52	Pretty Good Privacy, PGP	142, 154	Schadensmöglichkeiten	15		
Nmap	227	Prinzip der minimalen Rechte	223	Schlüssel exportieren	157		
Non-Executable-Stack	223	PRISM	20	Schlüssel importieren	159		
Non-Repudiation	7	Privacy	6	Schlüsselinfrastruktur	142		
NOP	35	Private Key	150	Schlüsselmanagement	127		
NOP-Sliding	35	Programmablauf	32	Schlüsselpaar generieren	155		
NSA	136	Programmaufbau	31	Schlüsseltausch	126		
NUL Scan	25	Protection	6	Secure Shell, SSH	150		
Nutzlast	53	Proxies	169	Secure Sockets Layer, SSL	145		
O							
Obscurity	109	Proxy	168	Security Association	187		
OFB	125	Psychometrie	218	Security Parameters Index, SPI	187		
OFDM	195	PUA	69	Security Policy	104		
One-Time-Pad	116	Potentially Unwanted Applications, PUA	72	Selbstsigniertes Zertifikat	148		
Online Social Engineering	28	Public Key	127, 150	Selbstverschlüsselnde Viren	62		
OpenPGP	142, 153, 154	Public Key Infrastructure, PKI	141	SELinux	224		
OpenVPN	190	Public Key Cryptography Standards	136	SERPENT	120		
Orthogonal Frequency Division Multiplexing, OFDM	195	Public-Key-Verschlüsselungsverfahren	132	Service Set Identifier, SSID	195		
Output-Feedback, OFB	125	PUP	69, 72, 98	Session Hijacking	43		
Overlayviren	61	Q					
Overwrite-Infektion	56	Qualifiziertes Zertifikat	148	SHA	138		
P							
Paketfilter	165	Radius	199	SHA-1	138		
PAP	215	RADIUS	215	SHA-2	138		
Password Authentication Protocol, PAP	215	Ransomware	67	SHA-3	138, 145		
Passwortknacker	229	RAS	214	Shared Secret Keys	196		
Patch	225	RC4	122	Shellcode, Aufbau	36		
Payload	53	RC6	120	ShiftRow	121		
Pegasus Mail	208	Remote Access Server, RAS	214	Sicherheits-Policies	51		
Perfect Forward Secret	198	Remote-Access-VPN	184	Sicherheitsprobleme	45, 46, 47, 49, 51		
R							
RAS	214	Sicherheitsrichtlinie	104				
RC4	122	Sicherheitsvorschriften	51				
RC6	120	Sicherheitszonen anpassen	89				
Remote Control	68	Signatur	93				
Remote-Control-Tool	226	Signatur, digitale	136				
Repeating-Modus	193	Signaturanalyse	60, 176				
Replay-Attacke	42	Signaturgesetz	13, 154				
S							
Signaturprüfung	60, 176						
Signaturverordnung	13						
Single-Point-of-Knowledge	46						

Skriptviren	57	Transportmodus	188	Wireless LAN	191
Skytale, Stabchiffre	109	Transportverschlüsselung	153	WPA	197
Slackviren	60	Trapdoor	128	WPA2	198
Smartcard	217	Triple DES	120	WPA3	198
Smurf-Attacks, DDoS	39	Trojaner	39, 67, 73	WPA-Enterprise	197
Snapshot	206	Trust Center	148	WPA-PSK	197
Sniffer	40	Trusted Introducer	142	WPS	200
Snort	179	Terminate and Stay Resident,		Würmer	66
Social Engineering	27, 28	TSR	55		
Spam-Filter	80	Tunnelmodus	188	X	
Spear-Phishing	79	TWOFISH	120	X.509	142, 153
Spectre	70			XKeyscore	20
Speicherresidente Viren, TSR	55	U		X-MAS Scan	25
Speicherverwaltung	33	Unified Messaging	210	XOR	118, 128
SPI	187				
Sporage	47	V		Z	
Sponsoring	75	VBS	58, 86	Zero-Day	225
Spybot	82	VBS-Skript Default ändern	86	Zertifikat	141, 142
Spyware	67, 72, 73, 75, 77	Verbindlichkeit	7	Zonealarm	171
SSH	150, 151	Verfügbarkeit	8		
SSH-Clients	151	Vernam, Gilbert S.	115		
SSID	195	Verschlüsselnde Viren	62		
SSID-Broadcast	195	Verschlüsselung, hybride	135		
SSL	145, 146	Verteilter Scan	26		
SSL aus Client-Sicht	147	Vertraulichkeit	6		
SSL aus Server-Sicht	147	Vertretungsregelung	45		
Stabchiffre	110	Vigenère, Blaise de	113		
State	121	Viren entfernen	97		
Stateful Inspection Firewall	172	Viren, Bauplan	53		
Stateful Packet Inspection		Viren, Geschichte	52		
Firewall, SPI	167, 172	Viren, Tarnmechanismen	59		
Stealth Scan	22	Virenprävention	86, 87, 89		
Stealth-Viren	64	Virensucher, Suchmethoden	64		
Stromchiffre	116	Virtual Private Network,			
Sub-CA	142	VPN	152, 183		
Substitution	119	Virtualisierung	206		
SYN	22	Virus im Dateianhang	56		
SYN-ACK Scan	24	Visual Basic Script	58		
Synchronize-Flag	22	VPN	152, 183		
		Vulnerability Assessment	226		
		Vulnerability Scanner	227		
T					
Tarnroutine	54	W			
Temporal Key Integrity Protocol,		Wardriving	27		
TKIP	197	Web of Trust	142, 160		
TGS	212	Webbugs	88		
TGT	212	Webmail	210		
Thunderbird	87	White-Hat	18		
Ticket Granting Server, TGS	212	Wi-Fi Protected Access, WPA	197		
Ticket Granting Ticket, TGT	212	Wi-Fi Protected Setup, WPS			
TKIP	197	Windows-Packet-Capture	179		
TLS	215	WinPCap	179		
TLS 1.3	149	Wired Equivalency Protocol,			
Token	217	WEP	196		
Total Cost of Ownership, TCO	226	Wireless Distribution System	193		
Tracking Cookie	72				
Transport Layer Security, TLS	215				

Impressum

Matchcode: NWSI_2019

Autor: Siegmund Dehn

Produziert im HERDT-Digitaldruck

11. Ausgabe, April 2019

HERDT-Verlag für Bildungsmedien GmbH
Am Kümmerling 21-25
55294 Bodenheim
Internet: www.herdt.com
E-Mail: info@herdt.com

© HERDT-Verlag für Bildungsmedien GmbH, Bodenheim

Alle Rechte vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder einem anderen Verfahren) ohne schriftliche Genehmigung des Verlags reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Dieses Buch wurde mit großer Sorgfalt erstellt und geprüft. Trotzdem können Fehler nicht vollkommen ausgeschlossen werden. Verlag, Herausgeber und Autoren können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Wenn nicht explizit an anderer Stelle des Werkes aufgeführt, liegen die Copyrights an allen Screenshots beim HERDT-Verlag. Sollte es trotz intensiver Recherche nicht gelungen sein, alle weiteren Rechteinhaber der verwendeten Quellen und Abbildungen zu finden, bitten wir um kurze Nachricht an die Redaktion.

Die in diesem Buch und in den abgebildeten bzw. zum Download angebotenen Dateien genannten Personen und Organisationen, Adress- und Telekommunikationsangaben, Bankverbindungen etc. sind frei erfunden. Eventuelle Übereinstimmungen oder Ähnlichkeiten sind unbeabsichtigt und rein zufällig.

Die Bildungsmedien des HERDT-Verlags enthalten Verweise auf Webseiten Dritter. Diese Webseiten unterliegen der Haftung der jeweiligen Betreiber, wir haben keinerlei Einfluss auf die Gestaltung und die Inhalte dieser Webseiten. Bei der Bucherstellung haben wir die fremden Inhalte daraufhin überprüft, ob etwaige Rechtsverstöße bestehen. Zu diesem Zeitpunkt waren keine Rechtsverstöße ersichtlich. Wir werden bei Kenntnis von Rechtsverstößen jedoch umgehend die entsprechenden Internetadressen aus dem Buch entfernen.

Die in den Bildungsmedien des HERDT-Verlags vorhandenen Internetadressen, Screenshots, Bezeichnungen bzw. Beschreibungen und Funktionen waren zum Zeitpunkt der Erstellung der jeweiligen Produkte aktuell und gültig. Sollten Sie die Webseiten nicht mehr unter den angegebenen Adressen finden, sind diese eventuell inzwischen komplett aus dem Internet genommen worden oder unter einer neuen Adresse zu finden. Sollten im vorliegenden Produkt vorhandene Screenshots, Bezeichnungen bzw. Beschreibungen und Funktionen nicht mehr der beschriebenen Software entsprechen, hat der Hersteller der jeweiligen Software nach Drucklegung Änderungen vorgenommen oder vorhandene Funktionen geändert oder entfernt.