Zur strafrechtlichen Beurteilung von IT-Sicherheitslücken

Seminar aus Internetrecht
ao. Univ. Prof. Dr. Zankl
WS 2005/2006

Lukas Feiler
Matrikelnummer: 0201227

Anmerkung vom 11. Februar 2009: Eine der zentralen Thesen dieser Arbeit war, dass das Tatbestandsmerkmal der „Verletzung“ einer Sicherheitsvorkehrung in § 118a Abs 1 StGB idF des StRÄG 2002 (BGBl I 134/2002) dazu führte, dass viele Arten der Zugangsverschaffung zu einem Computersystem nicht tatbestandlich waren. Wie in der Arbeit ausführlich dargestellt wird, galt dies insbesondere für eine Zugangsverschaffung unter Ausnützung von Command-Injection (5.2), SQL-Injection (5.3) oder Race Conditions (5.8). Durch das StrRÄG 2008 (BGBl I 109/2007) wurde in § 118a Abs 1 StGB das Wort „verletzt“ durch das Wort „überwindet“ ersetzt. In den EBRV 285 BlgNR XXIII. GP 7 heißt es hierzu:

> „Damit ist eine Beeinträchtigung der (Daten)Integrität der Sicherheitsvorrichtung nicht mehr erforderlich, zumal bestimmte technische Angriffsarten wie Code Injection, SQL Injection oder die Ausnützung von Race Conditions gerade zu keiner derartigen Beeinträchtigung führen.“

Dies legt den Schluss nahe, dass die vorliegende Arbeit zur Änderung des Tatbestandes des § 118a Abs 1 StGB wesentlich beigetragen hat.

Inhaltsverzeichnis

1. Einleitung

Durch den zunehmenden Einsatz von Informationstechnologie in geschäftskritischen Bereichen, hat auch das Thema IT-Sicherheit in den letzten Jahren stetig an Bedeutung gewonnen. Täglich neu auftretende Sicherheitslücken machen nicht nur eine technische sondern auch eine strafrechtliche Auseinandersetzung mit dem Thema erforderlich. Auf Grund der Neuartigkeit des Themas liegt insbesondere zu § 118a und § 126c StGB noch keine höchstgerichtliche Judikatur vor. Im Rahmen dieser Arbeit soll eine eingehende dogmatische Auseinandersetzung mit den einschlägigen Bestimmungen des österreichischen Computerstrafrechts und deren europäischen bzw. internationalen Grundlagen erfolgen. Hierbei wird insbesondere auf die Cyber-Crime Konvention des Europarates eingegangen. Ebenso Gegenstand der Erörterung ist der, für den österreichischen Gesetzgeber durch den Rahmenbeschluss 2005/222/JI des Rates vom 24. Februar 2005 über Angriffe auf Informationssysteme entstandene Handlungsbedarf.

Die relevanten Delikte des österreichischen Computerstrafrechts werden anhand eines typisierten Tathergangs systematisch dargestellt. Diesem folgend werden häufige Arten von Sicherheitslücken sowohl nach ihrer Auswirkung als auch nach ihrer technischen Beschaffenheit untersucht und mit den entsprechenden Delikten in Zusammenhang gesetzt. In einem weiteren Teil erfolgt eine strafrechtliche Beurteilung von Angriffen unter Verwendung konkreter Sicherheitslücken wie sie bei bestimmten Programmen in der Vergangenheit entdeckt wurden. Auf Grund der überaus großen praktischen Bedeutung werden zu diesem Zweck die SANS Top 20 Internet Security Vulnerabilities herangezogen.

Da Englisch im Bereich der Informatik und insbesondere im Bereich der IT-Sicherheit als Fachsprache vorherrschend ist, werden regelmäßig englische Termini technici verwendet.

2. Begriffsbestimmungen

Vorab gilt es den Begriff „Sicherheit“ im Zusammenhang mit Informationstechnologie zu bestimmen. Anerkannter Weise setzt sich dieser aus den Aspekten der Vertraulichkeit, Integrität und Verfügbarkeit von Daten und Systemen zusammen. So definiert das National Institute of Standards and Technology (NIST) des U.S. Department of Commerce “Computer Security“ als „The protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability and confidentiality of information system resources (includes hardware, software, firmware, information/data, and telecommunications)“.[1]
Die Generally Accepted Information Security Principles (GAISP)[2] enthalten keine explizite Definition des Begriffes der IT-Sicherheit, definieren jedoch – ebenso wie der Vorgänger GASSP[3] – Vertraulichkeit, Integrität und Verfügbarkeit als Ziele der primären Prinzipien („pervasive principles“) der IT-Sicherheit.[4] Diesem Begriffsverständnis folgt auch die Cyber-Crime Konvention des Europarates vom 23.11.2001[5] (ETS 185; im Folgenden: CyCC). [6]

[1] NIST SP 800-12, An Introduction to Computer Security: The NIST Handbook, S. 5; NIST SP 800-14, Generally Accepted Principles and Practices for Securing Information Technology Systems, S. 3
[2] GAISP wird von der Information Systems Security Association (ISSA) entwickelt. Version 3.0 ist unter http://www.issa.org/gaisp/_pdfs/v30.pdf online verfügbar.
[3] GASSP – Generally Accepted System Security Principles; Version 2.0 ist unter http://www.infosectoday.com/Articles/gassp.pdf online verfügbar.
[4] So grundsätzlich auch die Common Criteria for Information Technology Security Evaluation (auch Common Criteria od CC); vgl CC, Part 1: Introduction and general model, S. 9; online verfügbar unter http://www.commoncriteriaportal.org/public/files/CCMB-2005-07-001.pdf
[5] zugänglich unter http://conventions.coe.int/Treaty/Commun/QueVoulezVous.asp?NT=185&CM=8&CL=ENG

Begriffswesentlich ist jedoch, dass IT-Sicherheit niemals zu 100% gegeben sein kann. Weder die Vertraulichkeit noch die Integrität oder die Verfügbarkeit von Daten und Systemen kann jemals dauerhaft mit einer Wahrscheinlichkeit von 1 gegeben sein. Das Ziel sicherheitstechnischer Bestregungen kann es daher nur sein, das notwendigerweise bestehende Restrisiko zu minimieren.[7]

Unter IT-Sicherheitslücken ieS werden nur jene technischen Schwachstellen verstanden, die ein Eindringen in ein Computersystem oder die Störung der Funktionsfähigkeit eines solchen ermöglichen. IT-Sicherheitslücken iwS sind all jene technischen Gegebenheiten, die die Beeinträchtigung der Vertraulichkeit, Integrität oder Verfügbarkeit von Daten oder Systemen ermöglichen[8]. Im Folgenden wird der Begriff der IT-Sicherheitslücken iwS gebraucht. Auf Grund von technischen und psychologischen Faktoren ist es zwar nicht unmöglich jedoch äußerst unwahrscheinlich, dass ein Programm größeren Umfangs keine Sicherheitslücken enthält.[9] Empirische Studien[10] zeigen, dass nahezu jedes Programm Sicherheitslücken aufweist.[11]

In der Praxis besteht die Notwendigkeit bekannte Sicherheitslücken zu erfassen und zu dokumentieren. Common Vulnerabilities and Exposures (CVE) Nummern der MITRE Corporation[12] haben sich hierbei als Defacto-Standard etabliert. Alle hier in weiterer Folge angegebenen CVE-Nummern sind über http://www.cve.mitre.org abrufbar.[13]

3. Relevante Delikte nach typisiertem Tathergang

Die durch Ausnützung von Sicherheitslücken entstehenden Sachverhalte sind mannigfaltig. Es soll hier der Versuch unternommen werden, ein Modell derartiger Sachverhalte zu zeichnen und eine Systematisierung der relevanten Delikte vorzunehmen. Im Bereich der IT-Sicherheit gibt es verschiedenste Ansätze diese als Angriffe bezeichneten Vorgänge abstrakt zu erfassen. Hierbei hat sich insbesondere ein Modell namens „The 5 Ps“[14] als zweckmäßig erwiesen. Dieses gliedert einen Angriff in die Phasen Probe, Penetrate, Persist, Propagate und Paralyze. Da Gegenstand des hier zu entwickelnden Modells jedoch nicht bloß der Angriff im technischen Sinne, sondern der gesamte Tathergang ist, besteht die Notwendigkeit das Modell um die Phasen Prepare und Post-Attack zu erweitern.

---

[6] vgl die Überschrift zu CyCC Chapter II, Title 1: “Offences against the confidentiality, integrity and availability of computer data and systems” bzw. den erläuternden Bericht zur CyCC vom 8.11.2001 Rz 35.
[7] vgl NIST SP 800-30: Risk Management Guide for Information Technology Systems, 39 ff
[8] vgl die Differenzierungen in der englischen Fachsprache zwischen „Vulnerability“ und „Exposure“: http://www.cve.mitre.org/about/terminology.html
[9] *Graff/Van Wyk*, S. 18 ff
[10] vgl *Miller/Fredriksen/So* und *Miller/Koski/Lee/Maganty/Murthy/Natarajan/Steidl*
[11] So meinte der weltweite anerkannte Programmierer eines der sichersten Mail-Server Wietse Venema in einem Interview: “Defect-free software does not exist. With great effort I can reduce my own error rate to one bug in 1000 lines of code“; vgl http://www.seifried.org/security/index.php/Closet20010131_Interview_with_Wietse_Venema
[12] http://www.mitre.org
[13] Weitere Datenbanken von Sicherheitslücken sind: National Vulnerability Database - http://nvd.nist.gov, US-CERT Vulnerability Notes - http://www.kb.cert.org/vuls/, BugTraq - http://www.securityfocus.com/bid/, ISS X-Force - http://xforce.iss.net, OSVDB - http://www.osvdb.org
[14] *Cox/Gerg, S. 53 ff*

3.1. Prepare

Prepare beschreibt jene Phase, in der der Angriff selbst noch nicht initiiert ist, entsprechende Vorbereitungshandlungen jedoch schon ausgeführt werden. Hierzu gehört insbesondere das Sammeln von sog. Exploits, das sind Programme, die die Ausnützung einer Sicherheitslücke automatisieren.

3.1.1. § 126c StGB

Bei § 126c Abs 1 handelt es sich um ein Vorbereitungsdelikt der §§ 118a, 119, 119a, 126a, 126b und 148a. Es wurde durch das StRÄG 2002 (BGBl I 134/2002) zur Umsetzung des Art 6 („Misuse of devices") der CyCC eingeführt. Der objektive Tatbestand erfordert das Herstellen, Einführen, Vertreiben, Veräußern, sonst Zugänglichmachen, Sichverschaffen oder Besitzen von „Hacker-Tools"[15] (Fall 1) oder Authentifizierungsdaten (Fall 2). [16]

Abs 1 Z 1 leg cit (Fall 1) spricht von einem Computerprogramm, das nach seiner besonderen Beschaffenheit ersichtlich zur Begehung der §§ 118a, 119, 119a, 126a, 126b oder 148a geschaffen oder adaptiert worden ist, oder einer vergleichbaren solchen Vorrichtung. Durch das Erfordernis, dass das Computerprogramm oder die vergleichbare Vorrichtung nach seiner besonderen Beschaffenheit ersichtlich zur Begehung genannter Delikte geschaffen oder adaptiert worden ist, soll die Strafbarkeit bezüglich Vorrichtungen, die nur eine Doppelfunktionalität aufweisen, ausgeschlossen werden. Es ist daher erforderlich, dass das Programm bzw. die Vorrichtung nach objektiver Beurteilung primär zum Zweck der Begehung genannter Delikte geschaffen oder adaptiert wurde[17].
Daher sind Exploits, obgleich sie auch der Überprüfung der Sicherheit des eigenen Computersystems dienen können, hierunter zu subsumieren. Die weit verbreiteten Sniffer tcpdump und ethereal sind mE jedoch als klassische Computerprogramme mit Doppelfunktionalität anzusehen. Sie werden zwar von Angreifern gerne zum Mitlesen fremder Datenströme verwendet, zählen aber auch zu den essentiellen Werkzeugen der Netzwerkanalyse eines jeden Netzwerkadministrators und werden auch von Entwicklern von Netzwerkapplikationen und Protokollen bevorzugt verwendet. Als Indiz kann hierfür dienen, dass alle größeren Linux-Distributoren wie Novell/SUSE oder RedHat/Fedora beide Pakete enthalten. Der Sniffer dsniff ist hingegen nach objektiven Kriterien beurteilt, primär zur Begehung der genannten Delikte geschaffen worden, da er insbesondere in der Lage ist, die schwache Verschlüsselung mancher Protokolle zu „knacken". Wiederum als Indiz kann herangezogen werden, dass weder Novell/SUSE noch RedHat/Fedora dieses Paket ausliefern.

§ 126c Abs 1 Fall 2 erfasst durch Z 2 im Gegensatz zu Fall 1, keine Computerprogramme sondern „ein Computerpasswort, einen Zugangscode oder vergleichbare Daten, die den Zugriff auf ein Computersystem oder einen Teil davon ermöglichen".[18] Da letztgenannte Daten, mit Computerpasswörtern und Zugangscodes vergleichbar sein müssen, ergibt sich, dass nur Authentifizierungsdaten[19] unter Z 2 zu subsumieren sind. Bei Computerprogrammen iSd Z 1 handelt es sich zwar auch um Daten, die einen Zugriff ermöglichen, diese sind jedoch keine Authentifizierungsdaten (nicht mit Passwörtern und Zugangscodes vergleichbar).

---

[15] vgl den erläuternden Bericht zur CyCC vom 8.11.2001 Rz 71
[16] *Reindl* in WK, § 126c Rz 6 ff
[17] vgl den erläuternden Bericht zur CyCC vom 8.11.2001 Rz 73: "objectively designed, or adapted, primarily for the purpose of committing an offence"
[18] vgl Art 6 Abs 1 lit a, ii: "a computer password, access code, or similar data by which the whole or any part of a computer system is capable of being accessed"
[19] Als Authentifizierung bezeichnet man den Prozess der Identitätsfeststellung eines Benutzers; vgl *Garfinkel/Spafford/Schwartz*, S. 70

Aus dem Kriterium der Vergleichbarkeit lässt sich jedoch auch für Computerpasswörter und Zugangscodes iSd Z 2 ableiten, dass diese ebenso wie die vergleichbaren Daten, einen Zugang zu einem System(teil) ermöglichen müssen. Ein Passwort genügt dieser Anforderung jedoch nur, wenn die Eingabe eines Benutzernamens nicht erforderlich ist. Ist sie – was typischer Weise der Fall ist – jedoch erforderlich, so handelt es sich erst bei der Kombination von Benutzernamen und Passwort um Daten iSd § 126c Abs 1 Z 2. Ein Passwort für einen nicht genannten E-Mail-Account ist daher ebenso wenig wie die ersten 6 Zeichen eines 8-stelligen Passwortes unter Z 2 zu subsumieren, da in beiden Fällen noch kein Zugang ermöglicht wird.
Fraglich ist jedoch ob es sich um Daten handeln muss, die den Zugriff auf ein konkretes System ermöglichen oder ob eine abstrakte Eignung für einen Zugriff auf irgendein System ausreichend sein soll.[20] Da der Wortlaut der Z 2 nicht die Ermöglichung des Zugriffs auf ein „bestimmtes" bzw. „bestimmbares" oder „konkretes" System, sondern lediglich „auf ein Computersystem" erfordert, erfasst er auch Authentifizierungsdaten, die den Zugriff auf irgend ein System ermöglichen. Die Daten „Benutzername: admin, Passwort: 1234" sind daher dem Wortlaut nach unter Z 2 zu subsumieren, obgleich sie keine Bezeichnung[21] eines konkreten Computersystems enthalten. Dies ist vergleichbar mit einem Schlüssel, der ungeachtet der Tatsache, dass nicht bekannt ist welches Schloss er sperrt, noch immer unter den Begriff des Schlüssels zu subsumieren ist.
Weder die Gesetzesmaterialien, Art 6 CyCC noch der erläuternden Bericht zur CyCC enthalten diesbezüglich weitere Ausführungen. Dem, aus Rz 71 des erläuternden Berichts zur CyCC gewonnenen Argument, dass Art 6 CyCC nur bestimmten potentiell gefährlichen Handlungen („specific potentially dangerous acts") entgegenwirken soll, kann entgegnet werden, dass Authentifizierungsdaten ohne Angabe eines konkreten Systems in bestimmten Fällen sehr wohl ein erhebliches Gefahrenpotential aufweisen (vgl *5.6.2. Default-Passwörter*).

Dass die Authentifizierungsdaten unbefugten Dritten bereits bekannt sind oder im Internet veröffentlicht wurden, steht einer Subsumtion unter Z 2 nicht entgegen.

§ 126c Abs 1 Fall 2 erfordert auch nicht, dass es sich um Daten handelt, die einen Zugriff auf ein Computersystem ermöglichen, über das der Täter nicht oder nicht alleine verfügen darf. Dies trägt der Tatsache Rechnung, dass auch das eigene Passwort zur Begehung der §§ 118a, 119, 119a, 126a, 126b oder 148a geeignet ist.[22]

Die Tathandlungen sind durch Herstellen, Einführen, Vertreiben, Veräußern, sonst Zugänglichmachen, Sichverschaffen oder Besitzen überaus weit gefasst.
Die beiden letztgenannten Tathandlungen wurden erst im Zuge der Umsetzung des Rahmenbeschlusses zur Bekämpfung von Betrug und Fälschung im Zusammenhang mit unbaren Zahlungsmitteln vom 28.5.2001, ABl 2001 L 149 vom 2.6.2001 durch das StRÄG 2004 (BGBl I 15/2004) eingefügt. Bemerkenswert ist jedoch, dass Art 4 RB nur die Strafbarkeit des Sichverschaffen oder Besitzen von Computerprogrammen nicht hingegen von Passwörtern verlangt[23]. Der österreichische Gesetzgeber hat damit ohne äußeren Anlass, entgegen seiner anlässlich des StRÄG 2002 (EBRV 1166 BlgNR XXI. GP 29) ausgesprochenen Vorbehalte, dass der Besitz nicht die Schwelle erreichen würde, ab der eine

[20] Eine Auseinandersetzung mit der Problematik ist in der Literatur bislang nicht erfolgt.
[21] Beispielsweise durch eine IP-Adresse, einen Domainnamen oder eine geographische Bezeichnung (zB „der zweite Rechner von links").
[22] zB wenn es erst vom Computersystem zu dem das „eigene" Passwort Zugang gewährt, möglich ist ein drittes Computersystem anzugreifen.
[23] *Reindl* in WK, § 126c Rz 12

Kriminalisierung gerechtfertigt erscheint, auch den Besitz und das Sichverschaffen von Authentifizierungsdaten unter Voraussetzung des erforderlichen Vorsatzes kriminalisiert.

Der subjektive Tatbestand des § 126c Abs 1 erfordert neben einem Tatbestandsvorsatz in Form des dolus eventualis einen erweiterten Vorsatz. Der erweiterte Vorsatz besteht darin, dass es der Täter ernstlich für möglich halten und sich damit abfinden muss, dass das Tatmittel der Z 1 oder Z 2 zur Begehung der §§ 118a, 119, 119a, 126a, 126b oder 148a verwendet wird. Der erweiterte Vorsatz des Täters des § 126c Abs 1 hat daher auch den erweiterten Vorsatz des unmittelbaren Täters genannter Delikte zu umfassen.[24]

Nun folgend sollen die Tathandlungen des § 126c Abs 1 näher untersucht werden. Hierbei ist das Vertreiben, Veräußern und sonst Zugänglichmachen von den anderen Tathandlungen grundsätzlich zu unterscheiden. Denn die drei genannten Tathandlungen entsprechen den Tathandlungen des Beitragstäters durch sonstigen Beitrag zu einer Begehung der §§ 118a, 119, 119a, 126a, 126b oder 148a. Zu beachten ist, dass der Tatbestand des § 126c Abs 1 es nicht erfordert, dass der unmittelbare Täter bereits in das Versuchsstadium getreten ist. Aus materieller Betrachtung pönalisiert § 126c Abs 1 daher in bestimmten Fällen den sonst gem § 15 Abs 2 e contrario straflosen Versuchs des sonstigen Beitrags.
Die anderen Tathandlungen des Herstellen, Einführen, Sichverschaffen und Besitzen sind demgegenüber Handlungen, die typischer Weise der unmittelbare Täter der §§ 118a, 119, 119a, 126a, 126b oder 148a setzt.

Problematisch erscheint jedoch das Ausmaß in dem § 126c Abs 1 Fall 2 in den Bereich der Vorbereitung hineinreicht. So wäre bereits der Besitz des eigenen Passwortes oder das Anlegen eines neuen Benutzer-Accounts auf dem eigenen System bei entsprechendem Vorsatz strafbar.
Die nun folgenden Überlegungen gelten grundsätzlich für alle Tathandlungen des § 126c Abs 1, die in Bezug auf Authentifizierungsdaten gesetzt werden.

Vorbereitungshandlungen sind grundsätzlich straflos, da sie charakteristischer Weise mehrdeutig sind und daher auch keinen verlässlichen Schluss auf den Vorsatz des Täters zulassen[25]. Da ein allgemeiner Zweck des Strafrechts die Einwirkung auf menschliches Verhalten ist, muss grundsätzlich das Verhalten selbst stets Ausgangspunkt der Beurteilung sein. Im vorliegenden Fall handelt es sich um ein Delikt mit einer gesetzlichen Verhaltensbeschreibung, aus der sich die Wertung des Gesetzgebers ergibt, dass die Setzung einer der Tathandlungen in Bezug auf Authentifizierungsdaten mit entsprechendem Vorsatz eine sozial-inadäquat gefährliche Handlung darstellen soll.
Wie im Folgenden gezeigt wird, setzen jedoch alle, § 126c Abs 1 Fall 2 nahe stehenden Vorbereitungsdelikte eine vom Vorsatz unabhängige, sozial-inadäquate, objektiv bestimmbare Gefahrenlage voraus.

So begründet in § 126c Abs 1 Fall 1 die objektiv besondere Beschaffenheit des Computerprogramms die spezifische Gefahrenlage. Durch den Ausschluss von Programmen mit Doppelfunktionalität wird der Bereich der Alltagshandlungen nicht vom objektiven Tatbestand erfasst.

§ 224a ist in Bezug auf das Tatbild des § 126c Abs 1 Fall 2 sehr ähnlich gelagert. Das in § 224a gegenständliche Tatmittel der Urkunde ist ebenso wie das Passwort wesensnotwendig mit einer Doppelfunktionalität ausgestattet. § 224a erfasst mit dem Übernehmen, sich oder

[24] *Reindl* in WK, § 126c Rz 15
[25] *Fuchs*[6], S. 215

einem anderen Verschaffen, Befördern, Überlassen oder sonstig Besitzen ebenso eine Fülle von Tathandlungen. Durch das Erfordernis, dass es sich um eine falsche oder verfälschte (besonders geschützte) Urkunde handeln muss, ergibt sich im Unterschied zu § 126c Abs 1 Fall 2 jedoch die nicht alltägliche Gefahrenlage[26] bereits aus dem objektiven Tatbestand.

Vergleichbar mit § 126c Abs 1 Fall 1 stellen die §§ 227, 239 und 241c auf ein Tatmittel ab, dass nach seiner besonderen Beschaffenheit ersichtlich zur Begehung bestimmter Delikte bestimmt ist. Da die spezifische Zweckbestimmung des Tatmittels zur Deliktsbegehung objektiv gegeben sein muss, ohne dass es eines Rückgriffs auf den Vorsatz des Täters bedürfte, ergibt sich auch hier eine nicht alltägliche Gefahrenlage[27] aus dem objektiven Tatbestand selbst.

Nun sind die Tathandlungen des § 126c Abs 1 Fall 2 in Bezug auf eine spezifische Gefahrenlage zu untersuchen.
Die Herstellung von Authentifizierungsdaten erfolgt beispielsweise bei der Anlegung eines neuen Benutzer-Accounts, da hierzu die Angabe eines Benutzernamens und eines Passwortes erforderlich ist. Da sog. Mehrbenutzerbetriebsysteme, zu denen alle modernen Betriebsysteme zu zählen sind, für ihre Verwendbarkeit die Anlegung von Benutzer-Accounts erfordern, handelt es sich bei der Herstellung von Authentifizierungsdaten im Allgemeinen um keine Handlung, die eine spezifische Gefahrenlage erzeugt.
Die Tathandlung des Einführens von Authentifizierungsdaten erfolgt regelmäßig durch reisende Geschäftsleute, die auf ihren mobilen Geräten notwendiger Wiese auch Authentifizierungsdaten speichern. Es handelt sich daher auch hier typischer Weise um eine Alltagshandlung, die keine spezifische Gefahrenlage schafft.
Ein Sichverschaffen von Authentifizierungsdaten liegt beispielsweise in jenen Fällen vor, in denen einem Benutzer vom Administrator ein neues Passwort zugewiesen wird bzw. für den Benutzer ein neuer Account angelegt wird. Auch hier mangelt es im Allgemeinen an einer spezifischen Gefahrenlage.
Den Besitz von Authentifizierungsdaten verwirklicht nahezu jeder Benutzer eines Computersystems, da dessen Betrieb ohne den Besitz von Authentifizierungsdaten meist nicht möglich ist. Es handelt sich hierbei um äußerst alltägliche Handlung, ohne dass diese eine spezifische Gefahrenlage erzeugen würde.

Das Vertreiben (distribution iSd Art 6 CyCC) von Authentifizierungsdaten besteht in deren aktiver Weitergabe an Dritte. [28] Demgegenüber ist das Zugänglichmachen (making available iSd Art 6 CyCC) ein passiver Akt.[29] Beides stellt Handlungen dar, deren Ausführung – insbesondere durch einen Systemadministrator – keine spezifische Gefahrenlage schafft.
Die Veräußerung (sale iSd Art 6 CyCC) von Authentifizierungsdaten erfolgt regelmäßig zu dem Zweck Dritten einen Zugriff auf ein Computersystem zu ermöglichen, über das der Veräußerer alleine verfügungsbefugt ist. Aus der Veräußerung selbst ergibt sich daher ebenso keine spezifische Gefahrenlage.

Es handelt sich daher bei allen Tathandlungen des § 126c Abs 1 Fall 2 um Alltagshandlungen, die unumgänglich sind um die, durch § 126c geschützten Rechtsgüter (Computersysteme und Daten) zu gebrauchen. Folgender vergleichbare, fiktive Tatbestand soll dies illustrieren: „Wer einen Schlüssel mit dem Vorsatz, dass er zur Begehung des § 109 oder § 129 verwendet werde, herstellt, einführt, vertreibt, veräußert, sonst zugänglich macht, sich verschafft oder

[26] *Kienapfel/Schroll* in WK, § 224a Rz 1; *Kienapfel* in WK, § 226 Rz 3
[27] *Schroll* in WK, § 241c Rz 1
[28] Eläuternder Bericht vom 8.11.2001, Rz 72: „the active act of forwarding data to others"
[29] Eläuternder Bericht vom 8.11.2001, Rz 72: „the placing online devices for the use of others"

besitzt“. Daraus ist klar ersichtlich in welcher Nähe § 126c Abs 1 Fall 2 bei wörtlicher Interpretation tatsächlich zu einem „Gedankenverbrechen“ steht, wie es nur aus totalitären Systemen bekannt ist.[30]

Der aus § 126c Abs 1 Fall 1, den §§ 227, 239, 241c und 224a bzw. dem allgemeinen Zweck des Strafrechts auf menschliches Verhalten - und nicht auf menschliche Gedanken einzuwirken, gewonnene Zweck des § 126c Abs 1 Fall 2, ausschließlichen sozial-inadäquate Verhaltensweisen zu pönalisieren ist gegenüber den Tathandlungen des Vertreiben, Veräußern und sonst Zugänglichmachen zu relativieren. Denn bei diesen handelt es sich wie bereits erwähnt, bei materieller Betrachtung um sonstige Beiträge zur unmittelbaren Täterschaft der §§ 118a, 119, 119a, 126a, 126b oder 148a. Diese wären nach allgemeinen Regeln (§ 15 Abs 2 e contrario) straflos solange der unmittelbare Täter nicht in das Versuchsstadium getreten ist. Bezüglich der drei genannten Tathandlungen besteht der Zweck des § 126c Abs 1 Fall 2 daher darin, den versuchten sonstigen Beitrag zu pönalisieren. Da nach allgemeinen Regeln die Tathandlung des sonstigen Beitrags „an sich“ nach hA[31] nicht sozial-inadäquat sein muss, ergibt sich auch für die Beitragshandlungen des § 126c Abs 1 Fall 2, dass eine Strafbarkeit auch bei sozial-adäquaten Handlungen eintritt.

Als Ergebnis dieser teleologischer Erwägungen bezüglich § 126c Abs 1 Fall 2 ist daher festzuhalten, dass die Tathandlungen des Vertreiben, Veräußern und sonst Zugänglichmachen ohne Einschränkungen anzuwenden sind, die Tathandlungen des Herstellen, Einführen, Sichverschaffen oder Besitzen jedoch eine sozial-inadäquate Verhaltensweise voraussetzen.

Zu einem etwas anderen Ergebnis als die Methode der teleologischen Interpretation gelangt die historische Interpretation. Denn den Gesetztesmaterialien zum StRÄG 2002 (EBRV 1166 BlgNR XXI. GP, 29) und zum StRÄG 2004 (EBRV 309 BlgNR XXII. GP, 8) ist nicht entnehmbar, dass die Strafbarkeit nach § 126c Abs 1 Fall 2 bezüglich der Tathandlungen des Herstellen, Einführen, Sichverschaffen oder Besitzen auf sozial-inadäquate Verhaltensweisen beschränkt sein soll.
Bezüglich der Tathandlungen des Vertreibens, Veräußern und sonst Zugänglichmachen ist anzumerken, dass diese gem Art 6 Abs 3 CyCC zwingend umzusetzen sind, während bezüglich aller andern Tathandlungen ein Vorbehalt gem Art 42 CyCC erklärt werden kann. Daraus ist der „verstärkte“ Wille des historischen Gesetzgebers abzuleiten, die Tathandlungen des Vertreibens, Veräußerns und sonst Zugänglichmachens jedenfalls ohne Einschränkungen umzusetzen, was auch dem Ergebnis der teleologischen Interpretation entspricht.

Da die Grenze jeder Auslegung ieS[32] jedoch der äußerst mögliche Wortsinn ist und der abstrakte Wortsinn des § 126c Abs 1 Fall 2 zwingend auch alle sozial-inadäquaten Handlungen umfasst, kann dem hier ermittelten Zweck des § 126c Abs 1 Fall 2 nur durch eine teleologische Reduktion Rechnung getragen werden.

Da grundsätzlich nicht davon auszugehen ist, dass der Gesetzgeber die Normierung von „Gedankenverbrechen“ beabsichtigte, sondern wohl eher die Reichweite des Tatbestandes des § 126c Abs 1 Fall 2 auf Grund der technischen Komplexität der Materie unterschätzte, ist bezüglich der Tathandlungen des Herstellen, Einführen, Sichverschaffen und Besitzen eine teleologische Reduktion auf die Strafbarkeit von sozial-inadäquaten Handlungen vorzunehmen.

---

[30] *Fuchs*[6], S. 215
[31] *Fuchs*[6] 33. Kap Rz 57 ff
[32] *Fuchs*[6] 4. Kap Rz 16 ff

Hieraus ergibt sich der Wertungswiderspruch, dass die, von § 126c Abs 1 Fall 2 erfassten sonstigen Beitragshandlungen zu den §§ 118a, 119, 119a, 126a, 126b oder 148a, sofern diese von einem Vorsatz getragen sind, bereits bei Vorliegen einer sozial-adäquaten Handlung strafbar sind, wohingegen die Vorbereitungshandlungen des unmittelbaren Täters der §§ 118a, 119, 119a, 126a, 126b oder 148a eine sozial-inadäquate Verhaltensweise voraussetzen. Dieser Widerspruch könnte mit dem Argument aufgelöst werden, dass der Beitragstäter durch sonstigen Beitrag die Tat aus der Hand gibt, wohingegen der unmittelbare Täter noch die Möglichkeit hat, auf den weiteren Ablauf des Geschehens Einfluss zu nehmen.[33] Wenn man auch dieses Argument nicht als ausreichend erachtet, um den Wertungswiderspruch aufzulösen, so ist dieser zumindest nicht systemwidrig. Denn auch den Bestimmungstäter trifft eine weitere Strafbarkeit als den unmittelbaren Täter: die versuchte Bestimmungstäterschaft ist bereits strafbar, während der unmittelbare Täter erst mit Eintritt in das Versuchsstadium den Bereich der Strafbarkeit erreicht (§ 15 Abs 2).

Abschließend ist das Ergebnis der Auslegung iwS festzuhalten. Der objektive Tatbestand des § 126c Abs 1 Fall 1 erfasst ohne Einschränkungen die Verwirklichung jeder der in Abs 1 leg cit genannten Tathandlungen in Bezug auf Computerprogramme iSd § 126c Abs 1 Z 1. Im Rahmen des § 126c Abs 1 Fall 2 ist das Herstellen, Einführen, Sichverschaffen und Besitzen von Authentifizierungsdaten iSd Z 2 leg cit teleologisch derart zu reduzieren, dass die Strafbarkeit eine sozial-inadäquaten Handlung voraussetzt. Das Vertreiben, Veräußern und sonst Zugänglichmachen von Daten iSd Z 2 leg cit ist jedoch (wie bei Fall 1) ohne Einschränkungen vom objektiven Tatbestand erfasst.
Der subjektive Tatbestand erfasst sowohl in Fall 1 als auch in Fall 2 neben dem Tatbestandsvorsatz in Form des dolus eventualis einen bereits erörterten, erweiterten Vorsatz.

Das Herstellen, Einführen, Sichverschaffen und Besitzen des eigenen Passwortes ist daher, mangels sozial-inadäquater Verhaltensweise ungeachtet des Vorsatzes straflos. Ebenso sozial-adäquat (und daher straflos) verhält sich jemand, dem von seinem Arbeitskollegen oder Bekannten dessen Passwort mitgeteilt wurde bzw. jemand der auf Grund seiner administrativen Tätigkeit in einer Organisation Kenntnis von Passwörtern anderer Personen hat bzw. haben muss. Gleichfalls als sozial-adäquat zu beurteilen ist es, zufällig am Arbeitsplatz eines Kollegen vorbeizugehen und von dessen, unübersehbar am Monitor angebrachtem Passwort Kenntnis zu erlangen.
Sozial-inadäquat und daher strafbar würde sich jedoch jemand verhalten, der sich in die Räumlichkeiten des Unternehmens einschleicht und so Kenntnis vom, am Monitor angebrachten Passwort erlangt. Selbiges gilt, wenn das Passwort durch Sniffing, Brute Force Attacks od. Dictionary Attacks in Erfahrung gebracht wurde.

Wie später gezeigt wird, hat die Auslegung des § 126c Abs 1 erhebliche Auswirkungen auf den Umfang der Strafbarkeit nach § 118a Abs 1.

3.2. Probe

Im Rahmen des Probing, erfolgt das Sammeln von Informationen über das Angriffsziel[34]. Zum einen werden öffentliche Informationsquellen wie Websites, DNS und WHOIS-Datenbanken herangezogen, zum anderen wird häufig ein Portscan[35] oder ein Schwachstellen-Scan[36] durchgeführt. Durch Portscans ist feststellbar welche Dienste der, in weiterer Folge angegriffene Server im Netzwerk anbietet. Mittels Schwachstellen-Scans wird versucht

[33] zu diesem Argument bezüglich des Bestimmungstäters vgl Fuchs[6] 34. Kap Rz 35
[34] *Cox/Gerg, S. 54 ff*
[35] Die weiteste Verbreitung weist der Port-Scanner nmap auf; vgl http://www.insecure.org/nmap/.
[36] Der bekannteste Vulnerability Scanner ist zweifellos nessus; vlg. http://www.nessus.org.

mögliche Angriffspunkte zu identifizieren. Im Rahmen dieser Phase kann es auch zur Ausnützung von Schwachstellen kommen, die eine Informationspreisgabe ermöglichen. Da es hierdurch jedoch in aller Regel nicht zu einer Vollendung eines der in Betracht kommenden Delikte kommt und der Täter auch noch nicht in das Versuchsstadium tritt, besteht in dieser Phase des Angriffs keine Strafbarkeit.

3.3. Penetrate

In der Phase Penetrate dringt der Angreifer tatsächlich in das Computersystem ein.[37]

3.3.1. § 118a StGB

Durch das StRÄG 2002 erfolgte in § 118a die Umsetzung des Art 2 CyCC („Illegal access"). Geringfügige Änderungen der Rechtslage werden sich durch die Umsetzung des Rahmenbeschlusses 2005/222/JI des Rates vom 24. Februar 2005 über Angriffe auf Informationssysteme, ABl 2005 L 69 vom 16.3.2005 ergeben. Die Umsetzung hat gem Art 12 Abs 1 RB bis zum 16. März 2007 zu erfolgen[38].

Den objektiven Tatbestand des § 118a StGB erfüllt nur, wer sich zu einem Computersystem, über das er nicht oder nicht allein verfügen darf, oder zu einem Teil eines solchen Zugang verschafft, indem er spezifische Sicherheitsvorkehrungen im Computersystem verletzt.
Im Rahmen des objektiven Tatbestandes ist neben der Zugangsverschaffung insbesondere die Verletzung spezifischer Sicherheitsvorkehrungen im Computersystem im Kontext von Sicherheitslücken erklärungsbedürftig.

Der objektive Tatbestand erfordert die Zugangsverschaffung zu einem Computersystem oder zu einem Teil eines solchen. Der Begriff des Computersystems hat in § 74 Abs 1 Z 8 eine Legaldefinition erfahren. Diese orientiert sich an Art 1 lit a CyCC, wonach unter "computer system" Folgendes zu verstehen ist: „any device or a group of inter-connected or related devices, one or more of which, pursuant to a program, performs automatic processing of data". Gemäß § 74 Abs 1 Z 8 sind Computersysteme definiert als sowohl einzelne als auch verbundene Vorrichtungen, die der automationsunterstützten Datenverarbeitung dienen. Die Vernetzung mehrerer Computersysteme ist wiederum als ein Computersystem iSd § 74 Abs 1 Z 8 anzusehen[39]. Das größte existierende Computersystem ist daher das Internet.
Von § 118a erfasst, ist jedoch nicht nur die Zugangsverschaffung zu einem Computersystem sondern auch zu einem Teil eines solchen. Hardware als auch Software ist jedenfalls als Teil des Computersystems anzusehen. Jedoch werden auch andere Daten beliebigen Inhalts (zB Word-Dokumente oder Text-Dateien) durch ihre Speicherung im Computersystem Teil des Computersystems[40].
Ähnlich der Definition des Computersystems in Art 1 lit a CyCC bzw. § 74 Abs 1 Z 8 definiert der Rahmenbeschluss 2005/222/JI des Rates über Angriffe auf Informationssysteme den Begriff des Informationssystems in Art 1 lit a RB: eine Vorrichtung oder eine Gruppe miteinander verbundener oder zusammenhängender Vorrichtungen, die einzeln oder zu mehreren auf der Grundlage eines Programms die automatische Verarbeitung von Computerdaten durchführen sowie die von ihr oder ihnen zum Zwecke des Betriebs, der Nutzung, des Schutzes und der Pflege gespeicherten, verarbeiteten oder übertragenen Computerdaten. Im Unterschied zur geltenden Definition des Computersystems, umfasst der Begriff des Informationssystems nicht nur im System gespeicherte, sondern auch verarbeitete

[37] *Cox/Gerg, S. 64 ff*
[38] vgl http://www.bmj.gv.at/vorhaben/index.php?nav=12&st=1&th=1&sth=1&set=show&pj=254
[39] *Reindl* in WK, § 74 Rz 60
[40] *Reindl* in WK, § 118a Rz 7 f

und übertragene Daten. ME ist daher eine Novellierung des § 74 Abs 1 Z 8 zur Umsetzung des Rahmenbeschlusses erforderlich.

Eine Zugangsverschaffung erfordert grundsätzlich, dass der Täter innerhalb des Systems tätig werden kann.[41] Aus dem subjektiven Tatbestand (insbesondere dem Spionagevorsatz) ergibt sich jedoch weiters, dass der verschaffte Zugang zur Kenntnisnahme gespeicherter Daten geeignet sein muss. Besteht lediglich die Möglichkeit im System gespeicherte Daten zu verändern ohne sich von diesen Kenntnis verschaffen zu können, liegt kein Zugang iSd § 118a vor.

Weiters ist es erforderlich, dass der Täter über das Computersystem nicht oder nicht alleine verfügen darf. Es ist hierbei die Verfügungsbefugnis über das kleinste, in Betracht kommende Computersystem zu prüfen[42]. Verschafft sich jemand Zugang zu einem Computersystem, das Teil eines größeren Computersystems (zB des Internets) ist, so ist die Verfügungsbefugnis über das kleinere Computersystem maßgeblich. Andernfalls wäre das Tatbestandmerkmal der fehlenden Verfügungsbefugnis nahezu bedeutungslos, da niemand alleine über das Internet als Computersystem iSd § 74 verfügen darf.
Der Inhalt der Verfügungsbefugnis ist die unbegrenzte Nutzung und Veränderung des Computersystems[43]. Dem Verfügungsbefugen kommt damit eine Eigentümerähnliche Stellung zu. Dass einem Benutzer administrative Privilegien auf einem Computersystem eingeräumt werden, kann uU ein Indiz für das Bestehen einer Verfügungsbefugnis darstellen. Grundsätzlich kann jedoch nicht aus der Tatsache, dass der Benutzer etwas kann darauf geschlossen werden, dass er es auch darf.

So ist bei einer Zugangsverschaffung zu einem, mit dem Internet verbundenen Computersystem nicht die Verfügungsbefugnis über das Internet, das das größte existierende Computersystem iSd § 74 Abs 1 Z 8 darstellt, sondern die Verfügungsbefugnis über die kleinste Einheit, die noch als Computersystem iSd § 74 zu beurteilen ist, zu prüfen.

Nun ist das Tatbestandsmerkmal der Sicherheitsvorkehrung („security measure“ iSd CyCC) zu klären. Es handelt sich hierbei um Vorkehrungen durch deren Einsatz, die Wahrscheinlichkeit der Beeinträchtigung der Vertraulichkeit, Integrität oder Verfügbarkeit von Daten oder Systemen gemindert wird. Dass es ausschließlich auf eine Minderung der Wahrscheinlichkeit ankommt, ergibt sich aus dem Verständnis des Begriffes der IT-Sicherheit selbst. Selbiges gilt für die Abstellung auf die Aspekte Vertraulichkeit, Integrität und Verfügbarkeit. Darüber hinaus spricht auch Chapter II, Title 1 CyCC von „confidentiality, integrity and availability of computer data and systems“. Sicherheitsvorkehrungen umfassen, beschränken sich aber nicht auf Authentifizierungsmechanismen. Zu Sicherheitsvorkehrungen zählen nicht nur solche technische Gegebenheiten, die bestimmten Benutzern einen Zugang zum System od. zum Teil eines Solchen ermöglichen sollen, sondern auch jene, die für jeden (ohne Authentifizierung) eine bestimmte, beschränkte Funktionalität zur Verfügung stellen. Hierbei handelt es sich um die Anwendung des Sicherheitsprinzips „Least Privilege“[44] als Sicherheitsvorkehrung. Least Privilege bedeutet, dass jeder Prozess bzw. jedes Programm nur mit jenen Rechten ausgestattet sein soll bzw. nur jene Funktionalität implementieren soll, die für die Ausführung der Tätigkeit erforderlich sind. Daher ist jede Implementierung, die nur eine eingeschränkte Funktionalität umsetzt, eine Sicherheitsvorkehrung iSd § 118a StGB. Praktisch ist daher kaum ein Programm denkbar, das nicht auch als Sicherheitsvorkehrung

[41] *Reindl* in WK, § 118a Rz 19
[42] *Reindl* in WK, § 118a Rz 9 ff
[43] vgl *Reindl* in WK, § 118a Rz 14
[44] *Garfinkel/Spafford/Schwartz*, S. 235

anzusehen ist. Gelingt es einem Angreifer dennoch die nicht implementierte Funktionalität zu erreichen, kann eine Verletzung der Sicherheitsvorkehrung „Least Privilege“ vorliegen (zum Tatbestandsmerkmal der Verletzung siehe unten).

Die Sicherheitsvorkehrung muss sich im Computersystem befinden. Damit sollen insbesondere physische Sicherheitsvorkehrungen außerhalb des Computersystems wie eine versperrte Türe ausgeschlossen werden (EBRV 1166 BlgNR XXI. GP, 24). § 118a erfasst jedoch sehr wohl physische Sicherheitsvorkehrungen, die sich im Computersystem befinden – so zB ein Lesegerät für eine, zur Authentifizierung taugliche Smartcard.
Probleme bereiten jedoch Fälle in denen sich der Täter Zugang zu einem Computersystem dadurch verschafft, dass er eine Sicherheitsvorkehrung in einem anderen Computersystem verletzt. Derartige Konstellationen können beispielsweise bei Cross Site Scripting[45] auftreten. Sind beide Computersysteme Teil eines größeren Computersystems (idR eines Netzwerkes) so könnte besagtes größeres Computersystem als Tatobjekt betrachtet werden. Dadurch, dass der Täter sich Zugang zu einem Teil des Netzwerkes verschafft indem er eine, in diesem befindliche Sicherheitsvorkehrung verletzt, bestünde – unter den sonstigen Voraussetzungen – eine Strafbarkeit nach § 118a. Die Alternative besteht darin – parallel zur Verfügungsbefugnis – auf das kleinste Computersystem abzustellen, wodurch in derartigen Fällen keine Strafbarkeit gegeben wäre. Weder der Wortlaut noch die Gesetztesmaterialien stehen erstgenannter Interpretation entgegen. Im Unterschied zur Verfügungsbefugnis erfolgt durch die Abstellung auf das größere Computersystem keine gänzliche Entwertung des Tatbestandsmerkmals. Für erstgenannte Interpretation spricht auch, dass Netzwerke grundsätzlich gem § 74 Abs 1 Z 8 auch Schutzobjekt des § 118a sind. Daher kann die Zugangsverschaffung zu einem Computersystem durch Ausnützung einer Sicherheitslücke in einem anderen Computersystem dann zu einer Strafbarkeit nach § 118a führen, wenn beide Computersysteme in einem Computersystem verbunden sind. Sind die Computersysteme jedoch nicht miteinander verbunden, besteht keine Strafbarkeit nach § 118a.

Weiters ist das Erfordernis der Spezifität der Sicherheitsvorkehrung erklärungsbedürftig. Nach den Gesetzesmaterialien ist eine Sicherheitsvorkehrung spezifisch, wenn sie im Computersystem angebracht worden ist. Insbesondere als nicht spezifisch werden allgemeine Maßnahmen oder Vorrichtungen im Bereich der physischen Sicherheit angesehen. Nach den Materialien dient die oben erläuterte Wendung „Sicherheitsvorkehrung *im Computersystem*“ lediglich der Klarstellung.
Nach derzeit wohl hM[46] ist unter „spezifisch“ zu verstehen, dass die Sicherheitsvorkehrung individuell gestaltet und geheim sein muss. Dies lässt sich weder den Materialien, der CyCC noch dem Gesetzeswortlaut selbst entnehmen. Diesem mE zu engen Verständnis liegt ein zu eng gefasster bzw. falscher Begriff der Sicherheitsvorkehrung zugrunde. Zum einen wird eine Sicherheitsvorkehrung fälschlich nur iS eines Authentifizierungsmechanismus verstanden. Dies ist insofern erklärlich, als dass dies die einzigen Sicherheitsvorkehrungen sind, mit denen ein Benutzer bewusst interagiert. Zum Zweck der Authentifizierung wird idR ein Benutzername und ein Passwort verwendet, wobei zweiteres individuell gestaltet und geheim sein sollte. Zum anderen ist anzumerken, dass primär nicht das Passwort, sondern die im Computersystem implementierte Passwortabfrage die Sicherheitsvorkehrung darstellt. Nur das im Computersystem, idR in verschlüsselter Form gespeicherte Passwort kann als untergeordneter Teil der Passwortabfrage verstanden werden.
Dem Wort „spezifisch“ kann hier daher neben dem, aus den Materialien hervorgehenden Sinn, dass die Sicherheitsvorkehrung im Computersystem angebracht sein muss, keine weitere Bedeutung beimessen werden.

[45] s. 5.4. Cross Site Scripting (XSS)
[46] *Reindl* in WK, § 118a Rz 24

Abschließend ist das Erfordernis der Verletzung der Sicherheitsvorkehrung zu klären. Art 2 CyCC ermöglicht es den Nationalstaaten die Strafbarkeit durch das Erfordernis der Verletzung („infringing“) einzuschränken. Selbige Möglichkeit eröffnet Art 2 Abs 2 des Rahmenbeschluss 2005/222/JI des Rates über Angriffe auf Informationssysteme.
Vom Wortlaut der Bestimmung ist jedenfalls die Umgehung nicht erfasst. So ist das Verwenden einer Boot-CD zwecks Umgehung der Passwortabfrage des auf der Festplatte installierten Betriebssystems nicht strafbar. Der äußerst mögliche Wortsinn der Verletzung würde – insbesondere aus technischer Sicht – auch eine sonstige Überwindung einschließen. Den Gesetzesmaterialien (EBRV 1166 BlgNR XXI. GP 24) ist jedoch entnehmbar, dass eine sonstige Überwindung nicht ausreichend sein soll. Aus der Differenzierung der Begriffe Überwindung und Verletzung ergibt sich, dass eine Beeinträchtigung der Vorkehrung in ihrer Substanz erforderlich ist. Ist die Sicherheitsvorkehrung als Software implementiert, ist daher die Beeinträchtigung der Datensubstanz der Vorkehrung erforderlich.

Ein besonderes Problem werfen hierbei unautorisierte Zugriffe auf Computersysteme auf, die durch die Verwendung eines, wie auch immer erlangten Passwortes erfolgen. Denn durch eine Authentifizierung mit dem richtigen Passwort wird ein, dem legitimen Anmeldevorgang identischer, technischer Prozess durchlaufen, der zu keiner Beeinträchtigung der Datensubstanz der Sicherheitsvorkehrung führt.

Der Begriff der Verletzung ist jedoch auch im Rahmen einer systematischen Interpretation im Zusammenhang mit § 126c StGB zu ermitteln. Gem § 126c Abs 1 ist ua grundsätzlich strafbar, wer mit dem Vorsatz, dass ein Passwort iSd Z 2 zur Begehung des § 118a verwendet wird, ein solches herstellt, einführt, vertreibt, veräußert, sonst zugänglich macht, sich verschafft oder besitzt. § 126c Abs 1 ist daher bei materieller Betrachtung als Vorbereitungsdelikt des § 118a zu beurteilen. Ist aber die Vorbereitung gem § 126c Abs 1 strafbar, so ist davon auszugehen, dass auch eine, der Vorbereitung entsprechende Ausführung strafbar sein soll. Diese Überlegung ist im Rahmen einer systematischen Interpretation des Begriffes der Verletzung iSd § 118a Abs 1 heranzuziehen. Eine Verletzung iSd § 118a liegt daher auch in all jenen Fällen vor, in denen bereits eine Strafbarkeit im Vorbereitungsstadium nach § 126c Abs 1 zu bejahen ist. Hierfür ist jedoch nur das Herstellen, Einführen, sich Verschaffen und Besitzen als Tathandlungen des § 126c Abs 1 relevant, da nur diese als Vorbereitungshandlungen des unmittelbaren Täters des § 118a zu beurteilen sind. Die Strafbarkeit der vier genannten Tathandlungen des § 126c Abs 1 ist durch eine teleologische Reduktion auf sozial-inadäquate Verhaltensweisen beschränkt (vgl hierzu 3.1.1. § 126c StGB). Daher gilt, dass eine Verletzung einer Sicherheitsvorkehrung iSd § 118a Abs 1 nicht nur bei Beeinträchtigung der Substanz der Sicherheitsvorkehrung vorliegt, sonder auch wenn der Täter des § 118a zur Zugangsverschaffung ein Passwort verwendet, das er vorsätzlich (einschließlich des, auf die Verwendung des Passwortes zur Begehung des § 118a gerichteten, erweiterten Vorsatzes des § 126c Abs 1) sozial-inadäquat hergestellt, eingeführt, sich verschafft oder besessen hat (und somit bereits im Vorfeld nach § 126c Abs 1 strafbar war).
Dieses Interpretationsergebnis befindet sich noch innerhalb des äußerst möglichen Wortsinns, da wie bereits ausgeführt, der Begriff der Verletzung auch eine Überwindung umfassen kann. Weiters ist es in der Lage, den Gesetzesmaterialien entsprechend, Bagatellfälle von der Strafbarkeit auszunehmen, gleichzeitig jedoch eine dogmatisch stichhaltige Begründung für die Strafbarkeit des Hackings unter Verwendung von Passwörtern zu liefern.

Der subjektive Tatbestand des § 118a erfordert neben dem Tatbildvorsatz einen erweiterten Vorsatz. Dieser muss in Form der Absichtlichkeit in dreifacher Hinsicht gegeben sein.

Erstens muss der Täter in der Absicht handeln, sich oder einem anderen Unbefugten von in dem Computersystem gespeicherten und nicht für ihn bestimmten Daten Kenntnis zu verschaffen. Zweitens ist ein Verwendungsvorsatz erforderlich. Der Täter muss in der Absicht handeln die Daten selbst zu benützen, einem anderen, für den sie nicht bestimmt sind zugänglich zu machen oder zu veröffentlichen. Drittens ist es erforderlich, dass der Täter mit der Absicht handelt, sich oder einem anderen einen Vermögensvorteil zuzuwenden oder einem anderen einen Nachteil zuzufügen. Die Absicht einem anderen einen Nachteil zuzufügen ist von der Spionage- und von der Verwendungsabsicht zu differenzieren. Der Nachteil muss über die Beeinträchtigung der Geheimnissphäre hinausgehen, da diese bereits von der Spionageabsicht erfasst ist.[47] Ebenso darf der Nachteil, auf den sich die Absicht des Täters richtet, nicht in der Benützung, Zugänglichmachung oder Veröffentlichung der Daten erschöpfen. Denn dies ist bereits von der Verwendungsabsicht erfasst.

Durch den subjektiven Tatbestand kommt es zu einer sehr weit gehenden Einschränkung der Strafbarkeit nach § 118a. Ein Angreifer, der lediglich seine Fähigkeiten erproben möchte ist mangels Spionageabsicht daher nicht nach § 118a strafbar. Ebenso bleibt ein Angreifer, der nur die Privatsphäre Dritter verletzen will, jedoch keine Nachteilszufügungs- bzw. Gewinnabsicht hat, nach § 118a straflos. Selbiges gilt mangels Spionageabsicht für einen Angreifer, der nach erfolgter Zugangsverschaffung nur eine Datenlöschung anstrebt oder das Computersystem nur für Angriffe auf Dritte verwenden möchte. Letztere Sachverhaltsvariante ist von besonderer praktischer Bedeutung. Viele Angreifer sind dazu übergegangen Computersysteme zu „sammeln“, um aus diesen ein sog. Botnet[48] zu bilden. Die durchschnittliche Größe eines Botnets beträgt ca. 20.000 Computersysteme.[49] Diese auch treffend als „Zombies“[50] bezeichneten Computersysteme können vom „Besitzer“ des Botnets gleichzeitig gesteuert werden. Sie dienen dazu Phishing-Sites zu hosten, Spams zu verschicken oder einen sog. Distributed Denial of Service[51] (DDoS) Attack gegen Dritte auszuführen. Obgleich Botnets eine der gefährlichsten Waffen sind, die dem Angriff auf Computersysteme dienen können, besteht für die Bildung eines Botnets mangels Spionageabsicht keine Strafbarkeit nach § 118a. Allenfalls kommt ein Versuch eines anderen Delikts (insbesondere § 126b) oder eine Strafbarkeit nach § 126c Abs 1 in Betracht.

Abschließend ist noch darauf hinzuweisen, dass sowohl die Spionage- als auch die Verwendungs- und die Nachteilszufügung- bzw. Gewinnabsicht bereits zum Zeitpunkt der Zugangsverschaffung vorliegen müssen. Verschafft sich der Angreifer also ohne Spionageabsicht Zugang und entdeckt erst dann für ihn interessante Daten, besteht keine Strafbarkeit nach § 118a.

3.4. Persist

Persist bezeichnet die Phase, in der der Angreifer entsprechende Modifikationen am System vornimmt um sich einen späteren Zugriff auf das System zu sichern. Hierbei kommt es meist zur Installation einer sog. Backdoor[52].

[47] *Reindl* in WK, § 118a Rz 37
[48] vgl *Puri*
[49] vgl SANS NewsBites, Volume: 7, Issue: 55,
http://www.sans.org/newsletters/newsbites/newsbites.php?vol=7&issue=55#312 bzw.
http://news.com.com/Bots+slim+down+to+get+tough/2100-7355_3-5956143.html
[50] NIST SP 800-83, S. 21
[51] vgl *Puppe/Maier*
[52] NIST SP 800-83, S. 21

3.4.1. § 126a StGB

Da die Installation der Backdoor eine Veränderung von Daten über die der Angreifer nicht oder nicht allein verfügen darf erfordert, ist § 126a StGB einschlägig. § 126a wurde durch das StRÄG 1987 (BGBl 605/1987) eingeführt, dient jedoch auch der Umsetzung des Art 4 CyCC („Data interference"). Der Rahmenbeschluss 2005/222/JI des Rates über Angriffe auf Informationssysteme enthält in Art 4 („Rechtswidriger Eingriff in Daten") ebenso eine entsprechende Regelung. Zur Umsetzung des Art 4 RB scheint eine Novellierung des § 126a jedoch nicht erforderlich.

Nach § 126a ist strafbar, wer vorsätzlich einen anderen durch Unbrauchbarmachen oder Unterdrücken von Daten über die er nicht allein verfügen darf, am Vermögen schädigt. Abs 1 leg cit nennt zwar auch die Tathandlung des Veränderns, versteht diese jedoch nur als Beispielsfall des Unbrauchbarmachens. Es ist jedoch ausreichend, dass die Daten für den Berechtigten minder brauchbar werden[53]. Der Schaden besteht in der Höhe jener Summe, die zur Wiederherstellung der Daten aufgewendet werden muss. Können die Daten ohne nennenswerten Aufwand durch ein Backup wiederhergestellt werden, liegt daher gar kein Schaden vor.[54]

Die einfachste Form einer Backdoor besteht darin einen neuen administrativen Account anzulegen. Unter UNIX und UNIX-ähnlichen Betriebssystemen erfordert dies traditionell lediglich das Hinzufügen einer neuen Zeile in der Datei /etc/passwd[55]. Eine solche Passwortdatei ist für die Absicherung eines Computersystems und damit für den berechtigten Benutzer nur in einem geringen Maß brauchbar. Problematisch scheint jedoch das Tatbestandsmerkmal der Schädigung. Da eine Wiederherstellung des ursprünglichen Zustandes nur das Löschen einer Zeile erfordert, ist eine solche zu verneinen. Daher bestünde diesfalls keine Strafbarkeit nach § 126a.

Bei der komplexesten Form einer Backdoor, einem sog. Rootkit[56] werden weit reichende Veränderungen in Systemdateien und dem Betriebsystemkern, dem sog. Kernel selbst vorgenommen um die Existenz der Backdoor zu verschleiern. Hierbei kommt es stets zu einer dauerhaften Veränderung von Daten. Da ein derart „verseuchtes" und in der Kontrolle eines Angreifers stehende Betriebssystem für nahezu alle Anwendungsbereiche unbrauchbar ist, stellt die Installation eines Rootkits eine Unbrauchbarmachung von Daten iSd § 126a dar. Der Schaden des Berechtigten besteht darin, das Betriebsystem einschließlich aller Programme neu installieren zu müssen.[57]

§ 126c ist somit nur bedingt geeignet eine Strafbarkeit in dieser Phase des Angriffs zu begründen. Probleme bereiten weiters Fälle in denen die Installation der Backdoor zu Funktionsstörungen des Computersystems führt. Dies tritt in der Praxis häufig auf, da die Angreifer oft geringe Kenntnisse über das betreffende System besitzen. Liegt eine schwere Störung der Funktionsfähigkeit vor, ist der objektive Tatbestand des § 126b erfüllt. Bei näherer Betrachtung wird es jedoch meist an dem erforderlichen Tatbildvorsatz in Form des dolus eventualis mangeln. Der Angreifer wird eine schwere Störung der Funktionsfähigkeit idR für möglich halten. Da er jedoch unbemerkt bleiben will, vertraut er meist darauf, dass

[53] *Bertel* in WK, § 126a Rz 3
[54] *Bertel* in WK, § 126a Rz 5 f
[55] Bei der Verwendung von „shadow passwords" ist auch eine Zeile in /etc/shadow hinzuzufügen.
[56] NIST SP 800-83, S. 21
[57] Dies ist die einzige anerkannte Vorgehensweise nach einem erfolgreichen Hack; vgl *Garfinkel/Spafford/Schwartz*, S. 709 f

diese nicht eintreten wird. In einem derartigen Fall läge daher nur bewusste Fahrlässigkeit vor, weshalb eine Strafbarkeit nach § 126c meist ausscheiden wird.

3.4.2. § 126c StGB

Da außer bei der Verwendung eines netzwerk-basierten Einbruchserkennungssystems in aller Regel weder die Verwendung (und damit der Besitz) bestimmter Tatmittel noch die Verletzung einer Sicherheitsvorkehrung beweisbar sind, wird es oft zu keiner Strafbarkeit nach §§ 118a oder 126c in den bereits erörterten Zusammenhängen kommen.
In der Phase Persist hingegen, entstehen idR die einzigen Beweise für einen erfolgreichen Angriff. Daher ist eine allfällige Strafbarkeit nach § 126c von großer praktischer Bedeutung.

Enthält die installierte Backdoor einen Authentifizierungsmechanismus und besitzt der Angreifer die hierfür erforderlichen Daten (idR ein Passwort), so verwirklicht er das Tatbild des § 126c Abs 1 Z 2, da dieser Besitz als sozial-inadäquat zu beurteilen ist. Auch der erforderliche Vorsatz wird, angesichts der Installation der Backdoor idR gegeben sein.

3.5. Propagate

In der Phase Propagate wird das, bereits in der Kontrolle des Angreifers befindliche System dazu verwendet weitere Systeme anzugreifen[58]. Diese Angriffe erfolgen wiederum in den hier beschriebenen sieben Phasen Prepare, Probe, Penetrate, Persist, Propagate, Paralyze und Post-Attack.

3.6. Paralyze

Das Ziel des Angreifers ist von Fall zu Fall unterschiedlich. So kann es sich um das Eindringen in den internen Mail-Server, die Kompromittierung der Kundendatenbank oder einer Lahmlegung des gesamten Netzwerkes der Organisation handeln. Erreicht der Angreifer sein Ziel ist die Phase Paralyze vollendet.[59]

3.6.1. § 119 StGB

Bisweilen besteht das Ziel des Angreifers darin, in eine zentrale Komponente der Infrastruktur des Netzwerkes des Opfers einzudringen und von diesem aus den gesamten Datenverkehr „abzuhören“.
Der objektive Tatbestand des § 119 erfordert die Benützung einer Abhörvorrichtung, die an einer Telekommunikationsanlage oder an einem Computersystem angebracht oder sonst empfangsbereit gemacht wurde. Dass die Abhörvorrichtung auch in Form von Software gegeben sein kann, ergibt sich aus einer systematischen Interpretation zu § 126c Abs 1 Z 1, der als Tatmittel ein Computerprogramm nennt, dass ersichtlich zur Begehung der §§ 119, 119a geeignet ist[60]. Daher erfüllt die Benützung eines Sniffers[61] den objektiven Tatbestand des § 119.

Der subjektive Tatbestand besteht neben dem Tatbildvorsatz in einem erweiterten Vorsatz in Form der Absichtlichkeit, sich oder einem anderen Unbefugten vom Inhalt einer im Wege einer Telekommunikation oder eines Computersystems übermittelten und nicht für ihn bestimmten Nachricht Kenntnis zu verschaffen.

---

[58] *Cox/Gerg, S. 68*
[59] *Cox/Gerg, S. 68*
[60] vgl *Reindl* in WK, § 119a Rz 4
[61] zB tcpdump; http://www.tcpdump.org

Da der Inhalt einer Nachricht Gegenstand der Spionage ist, muss es sich bei den abgehörten Daten um Gedankenerklärungen handeln[62]. Bedeutend erscheint jedoch die Tatsache, dass Gedankenerklärungen technisch nicht eindeutig erfassbar sind. So ist bei den Protokollen SMTP[63], POP3[64] und IMAP[65] der Body[66] des übertragenen E-Mails nicht notwendiger Weise eine Gedankenerklärung. Denn ein automatisch generiertes E-Mail, das den Administrator über die Systemauslastung der letzten Tage informiert, wird beispielsweise nicht als Gedankenerklärung zu beurteilen sein. Die Header[67] eines E-Mails werden mit Ausnahme des Headers „Subject“[68], der den Betreff einer Nachricht enthält jedoch jedenfalls eine Gedankenerklärung darstellen. Dem Aufruf einer URL (zB http://ris.bka.gv.at/bundesrecht/) unter Verwendung des Protokolls HTTP wird idR kein Gedankeninhalt beigemessen werden können. Es sind jedoch URLs denkbar, die als Gedankenerklärung zu klassifizieren sind. Dies ist insbesondere beim Abschicken von HTML-Formularen unter der Verwendung der Request-Method GET[69] leicht denkbar (zB: http://www.example.com/gaestebuch.php?autor= Lukas+Feiler&nachricht= Dies+ist+eine+sehr+gelungene+Website). Aus der Absicht sich Kenntnis von E-Mails zu verschaffen kann daher nicht zwingend darauf geschlossen werden, dass sich der Täter Kenntnis vom Inhalt einer Nachricht iSd § 119 verschaffen wollte. Ebenso wenig kann aus der Absicht URLs zu „sniffen“ auf das Fehlen der Spionageabsicht des iSd § 119 geschlossen werden. Die technische Beschaffenheit der Daten ist daher nur bedingt zur Feststellung des erweiterten Vorsatzes tauglich.

Daraus, dass nur übermittelte Daten vom subjektiven Tatbestand erfasst sind, ergibt sich, dass es sich um, am Übertragungsweg befindliche Daten handeln muss. Bei Datenübertragungen im Wege eines Computersystems kommt es notwendiger Weise häufig zu Zwischenspeicherungen. Da sich zwischengespeicherte Daten aus funktionaler Sicht noch am Übertragungsweg befinden, erstreckt sich der Schutz des § 119 bis zum Zeitpunkt der permanenten Speicherung. Verschafft sich der Täter Zugriff zu einem System auf dem die Daten zwischengespeichert werden, kann es daher zu einer echten Konkurrenz mit § 118a kommen.[70]

Neben dem „Mithören“ des Netzwerkverkehrs, stellen sog. Keylogger eine sehr verbreitete Methode zur Beschaffung vertraulicher Informationen dar. Die Funktionsweise eines Keyloggers besteht darin, für den Benutzer unbemerkt alle Tastaturanschläge zu protokollieren. Die so erstellten Protokolle werden meist in regelmäßigen Zeitabständen an den Täter gesandt. Ein solcher Tathergang unterscheidet sich vom „Sniffing“ dadurch, dass die Daten nicht zwischen zwei Computersystemen sondern nur innerhalb eines einzigen Computersystems übertragen werden. CyCC Art 3 erfasst grundsätzlich auch die Übertragung innerhalb eines Einzelnen Computersystems. Im erläuternden Bericht zur CyCC vom 8.11.2001 Rz 55 wird den Nationalstaaten jedoch auch die Möglichkeit eröffnet nur die Übertragung zwischen Computersystemen zu erfassen. Ein entsprechender Vorbehalt ist den Gesetzesmaterialien jedoch nicht zu entnehmen. Da der Wortlaut des § 119 („im Wege eines Computersystems übermittelten Daten“) auch innerhalb eines Computersystems übertragene Daten erfasst, ist auch ein Keylogger ein geeignetes Tatmittel des § 119.

---

[62] *Lewisch* in WK, § 119 Rz 9a
[63] SMTP (Simple Mail Transfer Protocol) ist in RFC 2821 spezifiziert und dient dem Versenden von E-Mails.
[64] POP3 (Post Office Protocol Version 3) ist in RFC 1939 spezifiziert und dient dem Empfangen von E-Mails.
[65] IMAP (Internet Message Access Protocol) ist in RFC 3501 spezifiziert und dient sowohl dem Senden als auch dem Empfangen von E-Mails.
[66] vgl RFC 2822, S. 7
[67] vgl RFC 2822, S. 6 f
[68] vgl RFC 822, S. 16 f
[69] vgl RFC 2616, S. 35
[70] *Reindl* in WK, § 118a Rz 40

3.6.2. § 119a StGB

§ 119a wurde durch das StRÄG 2002 eingeführt und dient der Umsetzung des Art 3 CyCC („Illegal interception“).
Der objektive Tatbestand des § 119a Abs 1 besteht in der Benützung einer, an einem Computersystem angebrachten oder sonst empfangsbereit gemachten Vorrichtung (Fall 1) und dem Auffangen der elektromagnetische Abstrahlung eines Computersystems (Fall 2).

Der subjektive Tatbestand erfordert für beide Fälle neben einem Tatbildvorsatz, einen erweiterten Vorsatz in Form der Absichtlichkeit. Der erweiterte Vorsatz muss, ähnlich § 118a in dreifacher Hinsicht gegeben sein.[71]
Erstens muss der Täter mit der Absicht handeln sich oder einem anderen Unbefugten von im Wege des Computersystems übermittelten und nicht für ihn bestimmten Daten Kenntnis zu verschaffen. § 119a Abs 1 schützt daher im Gegensatz zu § 119 nicht nur Nachrichten mit Gedankeninhalt, sondern jegliche Daten. Der sehr weite Datenbegriff ergibt sich aus § 74 Abs 2, der Daten als personenbezogene und nicht personenbezogene Daten sowie Programme definiert. Dem Datenbegriff des § 74 Abs 2 entspricht die Definition von Computerdaten in Art 1 lit b des Rahmenbeschluss 2005/222/JI des Rates vom 24. Februar 2005 über Angriffe auf Informationssysteme: die Darstellung von Tatsachen, Informationen oder Konzepten in einer für die Verarbeitung in einem Informationssystem geeigneten Form, einschließlich eines Programms, das die Ausführung einer Funktion durch ein Informationssystem auslösen kann. Parallel zu § 119 besteht der strafrechtliche Schutz jedoch nur für den Zeitraum der Übertragung (vgl oben).
Zweitens ist es erforderlich, dass der Täter mit der Absicht handelt die Daten zu benützen, einem anderen, für den sie nicht bestimmt sind, zugänglich zu machen oder zu veröffentlichen.
Drittens ist die Absicht erforderlich durch die Verwendung der Daten sich oder einem anderen einen Vermögensvorteil zuzuwenden oder einem anderen einen Nachteil zuzufügen. Die Verwendungsabsicht als auch die Gewinn- bzw. Schädigungsabsicht entspricht der Absicht des § 118a (vgl oben).

§ 119a ist gem Abs 1 leg cit. subsidiär zu § 119. Ebenso wie eine echte Konkurrenz von § 118a mit § 119 denkbar ist, kann es zu einer solchen mit § 119a Abs 1 Fall 1 kommen. Das Abfangen elektromagnetische Abstrahlung nach § 119a Abs 1 Fall 1 stellt hingegen typischer Weise keine Zugangsverschaffung dar, weshalb es diesfalls zu keiner Konkurrenz mit § 118a kommt.

3.6.3. § 123 StGB

Nach dieser Bestimmung ist strafbar, wer ein Geschäfts- oder Betriebsgeheimnis mit dem Vorsatz auskundschaftet, es zu verwerten, einem anderen zur Verwertung zu überlassen oder der Öffentlichkeit preiszugeben.[72] Entscheidend ist die Tatsache, dass bei diesen Daten – im Unterschied zu personenbezogenen Daten – bereits das Auskundschaften den objektiven Tatbestand erfüllt.

3.6.4. § 126a StGB

Vgl bereits einführend, *3.4.1. § 126a StGB*. Die simple Datenlöschung birgt für einen Angreifer idR keine (finanziellen) Vorteile. Sie ist auch in der „Hacker-Szene“ auf Grund der leichten Machbarkeit mit wenig bis gar keinem Prestige verbunden. Bei Web-Servern kommt

[71] *Reindl* in WK, § 119a Rz 7 ff
[72] Zur Einschränkung des sehr weiten Tatbildes vgl *Lewisch* in WK, § 123 Rz 6 ff.

es jedoch häufig zu einem sog. Defacement[73]. Hierbei wird die Website des Opfers derart umgestaltet, dass der erfolgreiche Angriff für Dritte leicht ersichtlich ist.

Begreift man den, für den objektiven Tatbestand erforderlichen Schaden im Aufwand der Wiederherstellung, so ist das Vorliegen eines Schadens oft zu verneinen, da idR unter geringem Aufwand ein zuvor erstelltes Backup eingespielt bzw. die Veränderung rückgängig gemacht werden kann. Der größte, durch ein Defacemant entstehende Schaden betrifft nicht die Daten sondern das Image des Unternehmens. Das Image eines Unternehmens ist jedoch nicht Schutzgegenstand des § 126a. Daher ist eine Defacement nur nach § 126a strafbar, sofern die Wiederherstellung der Daten einen nennenswerten Aufwand darstellt.

3.6.5. § 126b StGB

Im Zuge des StRÄG 2002 (BGBl I 134/2002) erfolgte mit § 126b die Umsetzung des Art 5 CyCC („System interference").

Den objektiven Tatbestand des § 126b Abs 1 verwirklicht, wer durch Dateneingabe- oder Übermittlung die Funktionsfähigkeit eines Computersystems über das er nicht allein verfügen darf schwer stört. Das Gesetz ordnet eine Subsidiarität zu § 126a an. Der subjektive Tatbestand des § 126b erfordert lediglich einen Tatbildvorsatz in Form eines dolus eventualis. Damit sollen sog. Denial of Service[74] (DoS) Angriffe erfasst werden[75]. Diese können unterschiedlich ausgestaltet sein und setzen nicht notwendiger Weise das Bestehen einer Sicherheitslücke voraus. Die trivialste Form eines DoS-Angriffs besteht in der Überbeanspruchung endlicher physischer Ressourcen des Computersystems wie Bandbreite, Arbeitsspeicher, Festplattenspeicher oder CPU-Rechenzeit. Selbiges gilt jedoch auch für softwareimplementierte Beschränkungen wie eine maximale Anzahl der geöffnete Dateien (file handles) oder laufenden Prozesse.

Oft ermöglichen aber erst bestehende Sicherheitslücken (insbesondere Buffer Overflows) einen erfolgreichen DoS-Angriff. In diesen Fällen kommt es zu einem Totalausfall des betroffenen Dienstes, des gesamten Betriebssystems oder sogar des ganzen Netzwerkes.

Insbesondere klärungsbedürftig ist, wann eine Störung schwer iSd § 126b ist. Die CyCC gibt hierfür keine Anhaltspunkte[76]. Da es sich der Systematik nach um ein Vermögensdelikt handelt, liegt es nahe den finanziellen Aspekt in Form des Wiederherstellungsaufwandes in den Vordergrund zu stellen[77]. Hiergegen ist einzuwenden, dass die Materialien (EBRV 1166 BlgNR XXI. GP, 29) zum Ausdruck bringen, „dass es auf die Schwere der tatsächlich beeinträchtigten (oder gefährdeten) Interessen des Opfers zur Herstellung der Tatbildlichkeit nicht ankommt, weil eben (nur) auf eine schwere Störung der Funktionsweise eines Computersystems, hingegen nicht auf einen schweren Schaden durch eine Störung der Funktionsweise eines Computersystems abgestellt wird." Nach den Materialien kann ein außertatbestandsmäßiger Schaden lediglich bei der Strafzumessung nach § 32 Abs 3 StGB eine Rolle spielen.
Dass es nicht auf den Wiederherstellungsaufwand ankommt, ergibt sich auch daraus, dass dieser idR nur bei der Veränderung, Löschung, sonstigen Unbrauchbarmachung oder Unterdrückung von Daten (vgl § 126a) entsteht. Bei den Tathandlungen des § 126b (eingeben oder übermitteln) entsteht aber typischer Weise gerade kein Wiederherstellungsaufwand. Die

[73] Das bekannteste Archiv von Defacements ist http://www.zone-h.org/en/defacements.
[74] *Garfinkel/Spafford/Schwartz*, S. 767 ff; *Cooper/Northcutt/Fearnow/Frederick*, S. 189 ff
[75] Erläuternder Bericht vom 8.11.2001 Rz 67
[76] Erläuternder Bericht vom 8.11.2001 Rz 67
[77] so *Reindl* in WK, § 126b Rz 12

meisten DoS-Angriffe führen zum Absturz eines Dienstes oder des ganzen Betriebssystems. Die Wiederherstellung erfordert hier lediglich einen Neustart.[78]
Bezüglich der, von § 126b geforderten Schwere ist daher nicht auf einen allfälligen Schaden sondern ausschließlich auf die Störung selbst abzustellen. Der Absturz eines Dienstes (z.B. des HTTP-Servers) oder des gesamten Betriebssystems ist jedenfalls als schwere Störung zu beurteilen. Eine dem Absturz nahezu gleichkommende Verlangsamung des Computersystems ist ebenso als schwere Störung zu beurteilen. Je nach Art des Computersystem bzw. Dienstes wird eine unterschiedliche Minderung der Performance des Computersystems erforderlich sein.

Art 5 CyCC nennt neben der Eingabe und Übermittlung („inputting" und „transmitting") von Daten jedoch auch „damaging, deleting, deteriorating, altering or suppressing" als Tathandlungen. Da diese Tathandlungen jenen des Art 4 CyCC entsprechen, sah sie der Gesetzgeber mit § 126a als bereits umgesetzt an (vgl EBRV 1166 BlgNR XXI. GP, 28). Dies entspricht grundsätzlich den Verpflichtungen aus der CyCC, da „serious hindering" iSd Art 5 CyCC derart ausgelegt werden kann, dass das Vorliegen eines Schadens erforderlich sein soll.[79] Den Schaden im Wiederherstellungsaufwand zu begreifen erscheint ebenso zulässig.

Daher ist die schwere Störung eines Computersystems durch eingeben oder übermitteln von Daten unabhängig vom entstandenen Schaden bereits gem § 126b strafbar, wohingegen eine schwere Störung durch verändern, löschen oder sonst unbrauchbar machen oder unterdrücken von Daten gem §126a erst bei Eintritt eines Schadens in Form eines Wiederherstellungsaufwandes strafbar ist.

Dieser Wertungswiderspruch erklärt sich durch die Auffassung des Gesetzgebers, dass eine Veränderung, Löschung, Unbrauchbarmachung oder Unterdrückung von Daten ohne Schaden an den Daten selbst kaum denkbar sei.[80]

Daher sind physische Angriffe auf die Verfügbarkeit von Computersystemen ohne Verletzung der Sach- oder Datensubstanz auch dann straflos, wenn hierdurch die Funktionsfähigkeit eines gesamten Netzwerkes gestört wird. Unterbricht beispielsweise der Täter die Stromversorgung eines zentralen File-Servers so stellt dies eine schwere Störung der Funktionsfähigkeit des gesamten Netzwerkes dar. Mangels Dateneingabe- oder Übermittlung besteht keine Strafbarkeit nach § 126b. Der Täter setzt zwar damit die Tathandlung des Unzugänglichmachens von Daten iSd §126a („suppressing" iSd Art 5 CyCC), verursacht dank dem verbreiteten Einsatz von Journaling Filesystems[81] jedoch idR keinen Schaden an den Daten des File-Servers. Mangels Substanzbeeinträchtigung der Hardware des File-Servers ist auch eine Strafbarkeit nach § 125 zu verneinen.

Eine Art 5 CyCC entsprechende Regelung enthält auch der Rahmenbeschluss 2005/222/JI des Rates über Angriffe auf Informationssysteme in Art 3 („Rechtswidriger Systemeingriff"). Nach Art 3 RB ist die unbefugte vorsätzliche schwere Behinderung oder Störung des Betriebs eines Informationssystems[82], durch Eingeben, Übermitteln, Beschädigen, Löschen, Verstümmeln, Verändern, Unterdrücken oder Unzugänglichmachen von Computerdaten,

[78] Durch den verbreiteten Einsatz von Journaling Filesystems sind Beschädigungen des Dateisystems in Folge eines Systemabsturzes sehr selten; vgl *Pate* S. 201 ff
[79] vgl Erläuternder Bericht vom 8.11.2001 Rz 67: „a Party may require a minimum amount of damage to be caused in order for the hindering to be considered serious"
[80] vgl EBRV 1166 BlgNR XXI. GP, 28: „soweit eine solche Konstellation überhaupt denkbar ist"
[81] vgl *Pate*, S. 201 ff
[82] vgl Art 1 lit a RB

zumindest dann unter Strafe zu stellen, wenn kein leichter Fall vorliegt. Da es sich bei den beschriebenen physischen Angriffen typischer Weise nicht um leichte Fälle handelt, ist mE eine Novellierung des § 126b zur Umsetzung des RB erforderlich. Bemerkenswert erscheint jedoch, dass auch Art 3 RB keine Strafbarkeit für das Unterbrechen der Stromversorgung eines Computersystems vorsieht, wenn hierdurch kein anderes Computersystem in seinem Betrieb gestört wird.

3.6.6. § 225a StGB

Mit dem StRÄG 2002 (BGBl I 134/2002) erfolgte in § 225a die Umsetzung von Art 7 CyCC („Computer-related forgery"). Wie auch EBRV 1166 BlgNR XXI. GP, 30 entnehmbar[83], ist dieses Delikt parallel zu der in § 223 geregelten Urkundenfälschung ausgestaltet. Der objektive Tatbestand erfordert die Herstellung falscher Daten oder die Verfälschung echter Daten durch Eingabe, Veränderung, Löschung oder Unterdrückung von Daten. Neben einem Tatbildvorsatz ist ein erweiterter Vorsatz darauf erforderlich, dass die Daten im Rechtsverkehr zum Beweis eines Rechtes, eines Rechtsverhältnisses oder einer Tatsache gebraucht werden.

Parallel zur Urkundenfälschung sind Daten dann als falsch iSd § 225a anzusehen, wenn über die wahre Identität des Ausstellers getäuscht wird. Eine Verfälschung liegt hingegen vor, wenn bestehende Daten derart verändert werden, dass der neue Inhalt vom ursprünglichen Ersteller der Daten zu stammen scheint. „Lugdaten" können hingegen nicht unter § 225a subsumiert werden, da hierbei nicht über die Person des Ausstellers getäuscht wird.

Um über den Aussteller oder eine Erklärung des Ausstellers täuschen zu können, muss der Aussteller erkennbar sein. Daher wäre die Verfälschung eines E-Mails ohne Absender[84] nicht von § 225a erfasst. Sofern die Kopie vom Original nicht zu unterscheiden ist – was meist der Fall sein wird – kommt ihr der gleiche Schutz wie dem Original zu[85].

Aus der Parallelität zur Urkundenfälschung ergibt sich, dass der Täter mit Täuschungsvorsatz handeln muss. Auch der erläuternden Bericht zur CyCC vom 8.11.2001 Rz 81 spricht von einer „deception". Er muss es daher ernstlich für möglich halten und sich damit abfinden, dass die Daten den Beweisadressaten zu einer Willensbildung veranlassen[86]. Den subjektiven Tatbestand erfüllt daher der Täter nicht, wenn er ausschließlich den Vorsatz hat, die Daten gegenüber einem Computer oder einer Maschine einzusetzen. Jedoch ist eine mittelbare Verwendung der Daten zur Täuschung eines Menschen ausreichend (zB wird durch Eingabe verfälschter Daten die Ausgabe eines Dokumentes erreicht, das einem Menschen zur Täuschung vorgelegt wird).

3.7. Post-Attack

In der Post-Attack-Phase erfolgt die – meist finanzielle – Verwertung der „Früchte" des Angriffs. Hierzu zählt beispielsweise der Verkauf von erlangten Kundendaten.

3.7.1 § 51 DSG

Gegenstand des § 51 DSG sind personenbezogene Daten iSd § 4 Z 1 DSG, die dem Täter ausschließlich auf Grund seiner berufsmäßigen Beschäftigung anvertraut oder zugänglich geworden sind oder die er sich widerrechtlich verschafft hat. Die Tathandlungen bestehen –

[83] *Reindl* in WK, § 225a Rz 1
[84] ohne Angabe eines From-Headers; vgl RFC 2822, S. 20
[85] *Reindl* in WK, § 225a Rz 16 f
[86] *Reindl* in WK, § 225a Rz 20 ff

im Unterschied zu § 123 StGB, der bereits das Auskundschaften erfasst – im Benützen, Zugänglichmachen oder Veröffentlichen der Daten. Der subjektive Tatbestand erfordert neben dem Tatbildvorsatz einen erweiterten Vorsatz in Form der Absichtlichkeit sich einen Vermögensvorteil zu verschaffen oder einem anderen einen Nachteil zuzufügen.

Bemerkenswert ist die Tatsache, dass die Absicht des Täters einen Dritten zu bereichern nicht den subjektiven Tatbestand erfüllt.
Die Verschaffung von personenbezogenen Daten ist daher für sich nicht nach § 51 DSG strafbar.

4. Sicherheitslücken nach der Art ihrer Auswirkung

Die Unterscheidung von Sicherheitslücken nach der Art ihrer Auswirkung ist aus rechtlicher Sicht va dafür maßgeblich, welche Straftatbestände bei Ausnützung einer solchen Sicherheitslücke in Betracht kommen.

4.1. Ausführung beliebiger Befehle bzw. beliebigen Codes

Bei dieser Form von Sicherheitslücken ist es möglich durch ihre Ausbeutung beliebige Befehle bzw. beliebigen Maschinencode auf dem verwundbaren System zur Ausführung zu bringen. Die Abgrenzung zur Kompromittierung des gesamten Systems liegt grundsätzlich darin, dass der Befehl nur im Sicherheitskontext des, die Sicherheitslücke aufweisenden Programms ausgeführt wird. Da ein Programm idR nicht mit administrativen Rechten ausgestattet ist, erlangt der Angreifer derart zwar nicht Zugang zum gesamten Computersystem, jedoch zu einem Teil eines solchen. Es kommt daher insbesondere § 118a in Betracht.

4.2. Kompromittierung des gesamten Systems

Technisch können derartige Sicherheitslücken verschiedentlich ausgestaltet sein. Meist handelt es sich jedoch um ein, mit umfassenden administrativen Rechten laufendes Programm, das eine Sicherheitslücke aufweist, die die Ausführung beliebiger Befehle bzw. beliebigen Codes ermöglicht. Umfassende administrative Rechte genießt unter UNIX traditionell[87] nur der root-Account. Da durch die Ausnützung solcher Sicherheitslücken unmittelbar die Zugangsverschaffung zu einem Computersystem ermöglicht wird, ist § 118a einschlägig.

4.3. Informationspreisgabe

Hierbei ist es durch Ausnützung der Sicherheitslücke möglich die Vertraulichkeit bestimmter Information zu verletzen. Wesensnotwendig liegt jedoch unmittelbare noch keine Zugangsverschaffung zu einem Computersystem vor. Meist ermöglichen es derartige Sicherheitslücken dem Angreifer lediglich Informationen über das verwundbare Computersystem zu sammeln um weiterer Angriffe besser planen zu können (vgl 3.1. Prepare). So ermöglicht die Information an welcher Adresse im Arbeitsspeicher eine bestimmte Variable gespeichert ist, die Planung von Angriffen unter Ausnützung von Buffer Overflows. Bei der Verletzung von Sicherheitslücken, die zu einer Informationspreisgabe führen, kommt va eine Strafbarkeit nach § 15 iVm § 118a in Betracht[88]. Da die tatsächliche Zugangsverschaffung meist jedoch mit beträchtlichem zeitlichem Abstand erfolgt, ist eine Versuchsstrafbarkeit nur in seltenen Fällen zu bejahen.

[87] vgl *Garfinkel/Spafford/Schwartz*, S. 117 ff
[88] *Reindl* in WK, § 118a Rz 30 ff

4.4. Denial of Service

Durch Ausnützung dieser Art von Sicherheitslücken ist es möglich, die Funktionsfähigkeit eines Dienstes oder gar des gesamten Computersystems derart zu stören, dass es anderen Nutzern nicht mehr möglich ist, den Dienst bzw. das Computersystem zu verwenden. Hierfür ist § 126c das maßgebliche Delikt.

5. Sicherheitslücken nach der Art ihrer Beschaffenheit

Im Folgenden soll die Beschaffenheit von Sicherheitslücken insbesondere in Bezug auf das Tatbestandsmerkmal der Verletzung einer Sicherheitsvorkehrung iSd § 118a untersucht werden. Darüber hinaus soll auch eine Strafbarkeit nach anderen Delikten erörtert werden.

5.1. Manipulation des Hauptspeichers

Bei dieser Kategorie von Sicherheitslücken kann es einem Angreifer möglich sein, durch das Senden entsprechend formatierter Eingabedaten an das, die Sicherheitslücke aufweisende Programm beliebige Befehle im Sicherheitskontext des Programms auszuführen. Der Angriff kann fallweise von demselben oder einem entfernten Computersystem aus erfolgen.

Hierbei ist ein grundlegendes Verständnis der Organisation des Hauptspeichers erforderlich. Der von einem laufenden Programm verwendete Speicher besteht aus verschiedenen Teilen. Das Text-Segment enthält den, das Programm ausmachenden, kompilierten Maschinencode. Das Daten- und BSS-Segment enthält globale Variable. Der sog. Stack enthält lokale Variable, d.h. Variable, die nur innerhalb einer Funktion[89] existieren. Der sog. Heap enthält sonstige Daten, die nach der Beendigung einer Funktion weiter verfügbar sein sollen.

5.1.1. Klassische Buffer Overflows

Bei klassischen Buffer Overflows (Pufferüberläufen) ist es durch entsprechende Ausgestaltung der Eingabedaten möglich, den für die Speicherung der Daten vorgesehenen Speicherbereich (sog. Buffer) zu überfluten und dadurch Daten in andere Speicherbereiche zu schreiben wodurch der Programmfluss gestört und uU beliebiger Code ausgeführt werden kann.[90] Es wird hierbei in Stack Overflow und Heap Overflow unterschieden.

5.1.1.1. Stack Overflow

Diese Form der Sicherheitslücke hat in der Vergangenheit zu den schwerwiegendsten und am meisten Aufsehen erregenden Kompromittierungen von Computersystemen geführt.
Heute vermeiden manche Betriebssysteme diese Sicherheitslücke dadurch, dass der Stack nicht mehr ausführbar ist[91]. Dennoch besteht weiters eine praktisch überaus große Relevanz. Stack Overflows treten in Form des sog. Stack Smash und des sog. Off-by-one-Bug auf. Zum Verständnis der Funktionsweise dieser Sicherheitslücken ist ein Einblick in die Speicherorganisation des Stacks erforderlich. Einer in Ausführung befindlichen Funktion steht im Stack ein sog. Function Stack Frame zur Verfügung. Dieser enthält ua den sog. Instruction Pointer, der auf den nachfolgend auszuführenden Maschinencode (idR im Text-Segment) verweist, den Stack Frame Pointer, der den Beginn des Function Stack Frame adressiert und die innerhalb der Funktion verwendeten (lokalen) Variablen.

5.1.1.1.1. Stack Smash

[89] Als Funktion bezeichnet man vereinfacht eine Ansammlung mehrerer Befehle, die eine bestimmte Aufgabe erfüllen. Zweck einer Funktion ist es die Wiederverwendbarkeit der implementierten Funktionalität durch mehrmaligen Aufruf der Funktion zu ermöglichen.
[90] vgl *Krsul/Spafford/Tripunitara*, 2 ff
[91] vgl *Tennert*

Das Wesen der Sicherheitslücke Stack Smash[92] besteht darin, dass es dem Angreifer durch Manipulation von lokalen Variablen möglich ist, den Instruction Pointer zu überschreiben. Denn eine lokale Variable, die als Buffer fixer Länge definiert ist, wächst gleichsam in den Speicherbereich des Instruction Pointer und des Stack Frame Pointer hinein. Gelingt es dem Angreifer nun eine Speicheradresse im Instruction Pointer zu platzieren, die wiederum auf eine vom Angreifer manipulierbare lokale Variable verweist, ist es ihm möglich beliebigen Maschinencode zur Ausführung zu bringen.
Aus rechtlicher Sicht entscheidend ist, dass der Instruction Pointer, da er den Programmfluss determiniert, als Teil des Programms angesehen werden muss. Somit entspricht dessen Manipulation durch einen Stack Smash der Verletzung der Datensubstanz des Programms. Da dieses die Sicherheitsvorkehrung „Least Privilege“ implementiert, liegt eine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

5.1.1.1.2. Stack Off-by-one Bug

Im Unterschied zur soeben erörterten Ausprägung des Stack Overflow, ermöglicht ein Off-by-one Bug nicht das unmittelbare Überschreiben des Instruction Pointer sondern nur des Stack Frame Pointer. Es kommt jedoch nicht zum gänzlichen Überschreiben des Stack Frame Pointer sondern nur dessen letzten Bytes (off-by-one). Dieser Bug ist insbesondere bei Programmen, die in C geschrieben wurden aufzufinden. Denn in C schließt jede Zeichenkette mit einem sog. NULL Byte ab, weshalb eine Zeichenkette von 32 Zeichen einen Speicherbedarf von 33 Bytes aufweist. Auch heute erfolgt noch die meiste systemnahe Programmierung in C, jedoch verhindern moderne C-Compiler meist das Entstehen dieser Sicherheitslücke, was das Auftreten dieses Bugs in der Praxis mindert.
Der Off-by-one Bug weist eine höhere Komplexität als ein Stack Smash auf, führt mittelbar jedoch ebenso dazu, dass der Angreifer den Instruction Pointer beeinflussen kann und ermöglicht daher die Ausführung beliebigen Maschinencodes. Da auch der Stack Frame Pointer (mittelbar) den Programmfluss determiniert, liegt wie bei einem Stack Smash eine Verletzung der Datensubstanz der Sicherheitsvorkehrung vor.

5.1.1.2. Heap Overflow

Die Speicherorganisation des Heap ist unterschiedlich von jener des Stack. Die nun folgenden Ausführungen beziehen sich insbesondere auf die Heap-Implementierung der GNU libc[93], gelten grundsätzlich aber auch für andere Implementierungen.
Der Heap ist unterteilt in mehrere sog. Chunks. Jeder dieser Chunks speichert die Größe des vorangegangenen Chunks, die eigene Größe, die Information ob der vorangegangene Chunk derzeit verwendet wird (sog. PREV_INUSE Bit) und die Daten selbst. Ist ein Chunk nicht in Verwendung, speichert er im Datenbereich lediglich einen Pointer, der auf den nähesten, vorangegangenen freien Chunk verweist (BK-Pointer) und einen Pointer, der auf den nächsten freien Chunk verweist (FD-Pointer)[94]. Die Organisation der freien Chunks ist daher als Double Linked List implementiert – jedes Element der Liste (jeder freie Chunk) hat einen Pointer zum vorangehenden zum nächsten freien Chunk. Die Heap-Implementierung hat das Ziel mehrere kleinere Chunks nach Möglichkeit zu wenigen großen zusammenzufassen. Daher wird bei der Deallokation eines Chunks überprüft, ob nachfolgende Chunks ebenso bereits frei sind und daher alle Chunks zu einem großen zusammengefasst werden können. Ist dies der Fall, muss die Double Linked List aktualisiert werden. Wurde beispielsweise Chunk #1 und Chunk #2 zu einem Chunk zusammengefasst, muss der BK-Pointer des nächsten freien Chunks #3 nunmehr auf den gewachsenen Chunk #1 anstatt den nicht mehr existenten Chunk #2 verweisen. Ebenso muss der FD-Pointer des Chunk #1 nunmehr auf Chunk #3

[92] *Aleph One*
[93] http://www.gnu.org/software/libc/libc.html
[94] *Anonymous*

verweisen, da Chunk #2 nicht mehr existiert. Der aus dem BK-Pointer des Chunk #2 abgeleitete Speicherbereich des FD-Pointer des Chunk #1 wird mit dem Wert des FD-Pointer des Chunk #2 überschrieben. Spiegelbildlich wird der aus dem FD-Pointer des Chunk #2 ermittelte BK-Pointer des Chunk #3 mit dem Wert des BK-Pointer des Chunk #2 überschrieben.[95]

Bei einem Heap Overflow, der den Programmfluss beeinflusst, ist es möglich, durch das Überfluten eines Puffers, der im Datenbereich eines Chunks gespeichert ist, FD- und BK-Pointer des nachfolgenden Chunks zu überschreiben. Dadurch ist es möglich, einen ausgewählten Speicherbereich in der Größe von 8 Bytes (Größe eines Pointers) mit beliebigen Daten zu überschreiben. Unter Linux wird meist die, im Global Offset Table (GOT) gespeicherte Adresse einer vordefinierten Funktion (z.B: exit) überschrieben, da die Speicheradresse des GOT statisch ist. Versucht das Programm in weiterer Folge die Funktion, deren Adresse überschrieben wurde auszuführen, wird der, an besagter Adresse vom Angreifer platzierte Maschinencode zur Ausführung gebracht.

Bereits durch das Überschreiben des FD- und. BK-Pointer liegt eine Verletzung des Programms vor, da die Double Linked List der freien Chunks ein Teil des Programms darstellt. Somit ist die Verletzung einer Sicherheitsvorkehrung iSd § 118a zu bejahen.

5.1.1.2.1. Heap off-by-one und off-by-five Bugs

Bei diesen Formen des Heap Overflow, ist es nur möglich das PREV_INUSE Bit eines Chunks (Off-by-one Bug) oder auch das letzte Bit der, die Größe eines Chunks speichernden Zahl (Off-by-five Bug) zu überschreiben. Dies führt dazu, dass eine inkorrekte Zusammenführung zweier Chunks durchgeführt wird, die in weiterer Folge die Konstruktion eines neuen Chunks, mit vom Angreifer bestimmten FD- und BK-Pointern ermöglicht. Von diesem Punkt an, erfolgt der Angriff wie bereits beschrieben.

Durch das Überschreiben des PREV_INUSE Bit eines noch in Verwendung befindlichen Chunks liegt eine Verletzung des Programms vor, da die im betroffenen Chunk gespeicherten Daten Teil des Programms sind.

5.1.2. Integer Overflow

Hierbei handelt es sich tatsächlich nur um ein Mittel um in weiterer Folge Buffer Overflows zu verursachen. Bei einem Integer Overflow wird ein, im Programm verwendeter Integer[96] derart manipuliert, dass dieser zu groß oder negativ wird.
So kann beispielsweise ein Integer ohne Vorzeichen und mit einer Länge von 32-Bit nur Zahlen bis 2^{32} - 1 darstellen. Wird zu 2^{32} – 1 die Zahl 1 addiert, ist das Ergebnis 0, da nur die rechten 32 Bits maßgeblich sind. So kann eine derartige Berechnung der Größe eines dynamisch erzeugten Puffers dazu führen, dass dieser eine zu geringe Größe für die zu fassenden Daten aufweist, was einen Buffer Overflow ermöglichen würde. Eine andere Form des Integer Overflows tritt bei der Umwandlung von Integerwerten mit Vorzeichen in solche ohne Vorzeichen auf. So wird der negative Integerwert -200 hexadezimal als 0xffffff38 ausgedrückt, was dem positiven Integerwert 4.294.967.096 entspricht.

Da keine Daten überschrieben werden, sondern lediglich ein Kalkulationsfehler erzwungen wird, kommt es durch einen Integer Overflow selbst zu keiner unmittelbaren Verletzung des

[95] *McNab*, S. 304 ff
[96] Eine ganze Zahl.

Programms. Erst die, dadurch ermöglichten Buffer Overflows stellen eine solche Verletzung dar.

5.1.3. Format String Bug

Bei Vorliegen eines Format String Bugs ist es dem Angreifer möglich das Format-Argument eines Aufrufes einer Funktion wie printf()[97] od syslog()[98] zu manipulieren. Durch Verwendung entsprechender Formatspezifikationen ist es möglich auf die weiteren Parameter der Funktion zuzugreifen, wobei jeder Parameter 4 Byte im Speicher belegt. So wird durch „%7$s“ die siebente Gruppe von 4 Bytes referenziert. Wurde die Funktion hingegen nur mit 6 Parametern aufgerufen, kommt es zu einem Zugriff auf andere Speicherbereiche. Durch die Verwendung von „%n“ im Format-String ist es hingegen möglich die Anzahl der bis dahin geschriebenen Zeichen an einer beliebigen Stelle im Speicher abzulegen. Durch die Kombination beider Varianten ist es möglich gezielt bestimmte Bereiche im Speicher mit bestimmten Werten zu überschreiben und so beliebigen Code zur Ausführung zu bringen.

Durch das, für die Ausführung eigenen Codes erforderliche Überschreiben bestehender Speicherbereiche liegt eine Verletzung des Programms vor.

5.1.4. Double-free bugs

Bei einem Double-free Bug kommt es zu einer zweifachen Deallokation eines Chunks im Heap. Dies ist problematisch, da die FD- und BK-Pointer des zu deallozierenden Chunks zum Überschreiben von zwei mal je 8 Byte im Heap führen. Da nach der ersten Deallokation der Chunk idR mit anderen Chunks zu einem großen verbunden wird, enthalten die, für die zweite Deallokation verwendeten FD- und BK-Pointer bereits andere Werte. Gelingt es dem Angreifer, die als FD- und BK-Pointer behandelten Speicherbereiche mit entsprechenden Daten zu füllen, kommt es, wie bereits unter *5.1.1.2. Heap Overflow* dargestellt, zu einem gezielten Überschreiben von durch den Angreifer bestimmten Speicherbereichen. Dies ermöglicht in weiterer Folge die Ausführung beliebigen Codes.

Hierbei kommt kein Buffer Overflow zur Anwendung. Das gezielte Platzieren von Daten, die im Rahmen der zweiten Deallokation als FD- und BK-Pointer behandelt werden, stellt an sich noch keine Verletzung des Programms dar. Durch die, in weiterer Folge statt findende Überschreibung anderer Speicherbereiche mit dem Inhalt der als FD- bzw. BK-Pointer behandelten Speicherbereiche, kommt es jedoch zu besagter Verletzung.

5.2. Command-Injection

Hierbei setzt ein Programm einen Befehl im System ab, der vom Benutzer stammende Daten enthält und durch die Manipulation dieser Daten die Manipulation des ausgeführten Befehls selbst ermöglicht. So ist es bei folgendem beispielhaftem Code durch Setzen der Variablen $recipient auf den Wert *„joe@example.com; cat /etc/passwd“* nicht nur möglich – wie erwünscht – ein E-Mail zu versenden, sondern auch die Passwort-Datei eines Unix/Linux-Systems auszulesen: *eval "mail $recipient".*
In genanntem Beispiel ist bereits das Tatbestandsmerkmal des Vorliegens einer Sicherheitsvorkehrung problematisch. Grundsätzlich liegt eine Implementierung des Prinzips Least Privilege vor. Obgleich sie mangelhaft ist, verringert sie dennoch insofern die Wahrscheinlichkeit der Beeinträchtigung von Vertraulichkeit, Integrität oder Verfügbarkeit von Daten als dass der Angreifer über das Spezialwissen verfügen muss, dass er in der Variablen $recipient nach einem Strichpunkt eigene Befehle definieren kann.

[97] Unix Manual Page printf(3)
[98] Unix Manual Page syslog(3)

Mangels Beeinträchtigung der Datensubstanz des Programms ist jedoch jedenfalls die Verletzung der Sicherheitsvorkehrung zu verneinen. Daher besteht keine Strafbarkeit nach § 118a.

5.3. SQL-Injection

Diese Form der Sicherheitslücke tritt va bei Webapplikationen auf. Diese bestehen (serverseitig) idR aus zwei Teilen: dem Applikationsserver, auf dem die Geschäftslogik implementiert ist und dem Datenbankserver, der die Applikationsdaten speichert[99]. Zum Zwecke der Kommunikation mit der Datenbank verwendet der Applikationsserver die standardisierte Abfragesprache SQL[100]. Eine Datenbankabfrage enthält meist auch vom Benutzer der Applikation stammende Daten. Eine SQL-Injection Schwachstelle besteht nun darin, dass es dem Angreifer durch Eingabe entsprechend formatierter Daten möglich ist, die Datenbankabfrage zu manipulieren[101]. Es kann hierbei in folgende Angriffsvarianten unterschieden werden[102].

5.3.1. Informationspreisgabe

Folgender beispielhafter Code ermöglicht es, durch Zuweisung des Wertes „0 OR 1=1“ an die Variable userId, anstatt der Daten eines Kunden die Daten aller Kunden abzufragen:
*$sql=“SELECT * FROM customers WHERE f_userid=“ + $userId*.
Durch die Informationspreisgabe selbst erfolgt noch keine Zugangsverschaffung iSd § 118a. Handelt es sich um personenbezogene Daten, ist durch ihre Benützung, Zugänglichmachung oder Veröffentlichung der objektive Tatbestand des § 51 DSG erfüllt. Bei Daten, die eine Geschäfts- oder Betriebsgeheimnis darstellen, kommt bereits durch Auskundschaftung eine Strafbarkeit nach § 123 in Betracht. Die §§ 119 f scheiden jedoch jedenfalls aus, da diese nur am Übertragungsweg befindliche Daten schützen. Handelt es sich um Authentifizierungsdaten, ist durch das Sichverschaffen selbiger der objektive Tatbestand des § 126c erfüllt. Werden die derart erlangten Authentifizierungsdaten zur Zugangsverschaffung verwendet, liegt eine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

5.3.2. Unmittelbare Überwindung eines Authentifizierungsmechanismus

Folgender beispielhaft zur Authentifizierung dienender Code ermöglicht SQL-Injection:
$sql=“ SELECT * FROM users WHERE username=’$username’ AND password= ’$password’“. Wird der Variablen $username die Zeichenkette *„joe“* und der Variablen $password *”a’ OR ‘x’ = ‘x”* zugewiesen, ergibt sich folgende SQL-Abfrage: *SELECT * FROM user WHERE username=’joe’ AND password=’a’ OR ‘x’ = ‘x’*. Dies ermöglicht es einem Angreifer sich ohne Kenntnis des Passwortes des Benutzers joe als dieser zu authentifizieren und sich somit Zugang iSd § 118a zu einem Teil eines Computersystems zu verschaffen.
Es kommt hier jedoch zu keiner Verletzung der Sicherheitsvorkehrung des Authentifizierungsmechanismus, da die Datensubstanz desselbigen nicht beeinträchtigt wird. Auch eine Qualifikation der anstatt des Passwortes einzugebenden Zeichenkette *”a’ OR ‘x’ = ‘x”* als Computerpasswort iSd § 126c Abs 1 Z 2 ist zu verneinen, da es sich gerade um keine Authentifizierungsdaten, sondern um eine Möglichkeit den Authentifizierungsmechanismus als solchen zu umgehen handelt.

[99] vgl *Bergsten*, S. 271 ff
[100] ISO/IEC 9075 - Structured Query Language
[101] vgl *Middendorf*
[102] *Peikari/Chuvakin*, S. 374 ff

Eine derart erfolgte Zugangsverschaffung ist daher nicht nach § 118a strafbar. Allgemein gilt festzuhalten, dass die Ausnützung einer SQL-Injection Schwachstelle für sich betrachtet, keine Verletzung einer Sicherheitsvorkehrung darstellt.

5.3.3. Erstellen neuer Datensätze

Bevorzugt wird die Erstellung neuer Datensätze durch SQL-Injection zum Anlegen neuer Benutzerkonten verwendet. Nun folgende, beispielhafte Codezeile sollte nach dem Willen des Programmierers nur das Anlegen eines nicht-administrativen Accounts ermöglichen (f_isadmin ist der boolsche Wert für FALSCH zugewiesen): *$sql=”INSERT INTO accounts (f_username, f_password, f_isadmin) VALUES (‘$username’, '$password', 0)”*. Gelingt es aber einem Angreifer nun $username gleich *“new_admin’, ’secret’, 1), (’new_user“* zu setzen ergibt sich folgendes SQL-Statement: *INSERT INTO accounts (f_username, f_password, f_isadmin) VALUES (‘new_admin’, ’secret’, 1), (’new_user’, '$password', 0)*. Dies bewirkt, dass nicht nur ein nicht-administrativer Account (new_user) sondern auch ein administrativer Account (new_admin) angelegt wird.
Für die strafrechtliche Beurteilung maßgeblich ist, dass alleine durch die Erstellung eines neuen Accounts noch keine Zugangsverschaffung vorliegt und somit eine Strafbarkeit nach § 118a zu diesem Zeitpunkt allenfalls als Versuch möglich ist.[103]
Jede unautorisierte Veränderung eines Datenbestandes, der für eine Authentifizierung verwendet wird, macht den Datenbestand für den Berechtigten minder brauchbar, da hierdurch die Effektivität des Authentifizierungsmechanismus gemindert wird. Es liegt somit eine Unbrauchbarmachung von Daten iSd § 126a vor. Da das Hinzufügen einzelner Datensätze idR durch die Ausführung eines DELETE-Statements leicht rückgängig gemacht werden kann, wird das Vorliegen eines, unmittelbar durch die Unbrauchbarmachung entstandenen Schadens meist zu verneinen sein. In derartigen Fällen bestünde daher keine Strafbarkeit nach § 126a.
Da der Angreifer jedoch Authentifizierungsdaten auf sozial-inadäquate Weise besitzt, kommt eine Strafbarkeit nach § 126c in Betracht. Verwendet der Angreifer den administrativen Account um sich Zugang zu dem administrativen Teil der Applikation zu verschaffen, liegt eine Zugangsverschaffung zu einem Teil eines Computersystems durch Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

5.3.4. Datenmodifikation

Hierbei ist es einem Angreifer möglich UPDATE- oder DELETE-Statements zu manipulieren. So ermöglicht es folgender, beispielhafter Code durch Setzen der Variablen $userid auf *„1 OR 1=1“* das Passwort nicht nur eines, sondern aller Accounts zu ändern: *$sql=“UPDATE accounts SET f_password=’$password’ WHERE f_userid=“ + $userId*. Da dadurch die gesamten Authentifizierungsdaten der Applikation unbrauchbar gemacht werden, besteht unter den sonstigen Voraussetzungen eine Strafbarkeit nach § 126a. Durch den Besitz des neuen, für alle Accounts der Applikation gültigen Passwortes ist auch der objektive Tatbestand des § 126c Abs 1 erfüllt.
Wird das Passwort nun zur Authentifizierung verwendet, kommt auch eine Strafbarkeit nach § 118a in Betracht. Die Verletzung einer Sicherheitsvorkehrung ist in zweierlei Hinsicht gegeben. Zum einen ergibt sie sich aus der systematischen Interpretation zu § 126c. Zum anderen sind die Passwörter als Teil des Authentifizierungsmechanismus anzusehen, weshalb deren Überschreiben eine Beeinträchtigung der Datensubstanz einer Sicherheitsvorkehrung darstellt.
Durch das Löschen von Datensätzen kann sich ein Angreifer idR keinen Zugang zur Applikation verschaffen. Durch die Unbrauchbarmachung von Daten durch Löschung besteht

[103] vgl *Reindl* in WK, § 118a Rz 30 ff

unter den sonstigen Voraussetzungen aber eine Strafbarkeit nach § 126a. Folgender Code würde es einem Angreifer durch Setzen der Variable $userId auf *„1 OR 1=1“* ermöglichen alle anstatt nur einen Account zu löschen: *$sql=“DELETE accounts WHERE f_userid=“ + $userId*.

5.3.5. Ausführung beliebiger Befehle

Um beliebige Befehle auf dem Betriebssystem der Datenbank auszuführen, ist es erforderlich aus der Datenbank „auszubrechen“. Bei manchen Datenbanken ist dies durch die Verwendung von Stored Procedures möglich. So können auf einem Microsoft SQL Server unter Verwendung der Stored Procedure „xp_cmdshell“ beliebige Befehle im System abgesetzt werden. Bei einer derart erfolgten Zugangsverschaffung wird keine Datensubstanz beeinträchtigt und daher keine Sicherheitsvorkehrung verletzt. Es besteht somit keine Strafbarkeit nach § 118a.

5.4. Cross Site Scripting (XSS)

XSS-Sicherheitslücken sind va bei Web-Applikationen zu finden. Sie bestehen darin, dass es dem Angreifer möglich ist, clientseitigen Code (z.B. JavaScript od VBScript) in eine Web-Applikation einzuschleusen, wodurch der eingeschleuste Code beim Aufruf der Applikation durch andere Benutzer in deren Browsern zur Ausführung gelangt. Der eingeschleuste Code läuft damit im Kontext der, die Sicherheitslücke aufweisenden Applikation. Die XSS-Schwachstelle ist per definitionem eine solche der Web-Applikation, nicht des Client-Systems. Technisch ist sie idR derart ausgestaltet, dass es Benutzern der Applikation erlaubt ist, in einer Datenbank gespeicherte Daten zu erstellen bzw. zu verändern ohne, dass die Applikation die Daten auf schädlichen Code überprüft.

Handelt es sich um eine Intranetapplikation, die idR in einem, mit erweiterten Rechten ausgestatteten Sicherheitskontext läuft, kann es durch die Ausführung des eingeschleusten Codes bereits möglich sein, sich zu einem Client-Computersystem oder zu einem Teil desselben Zugang zu verschaffen. Da die Ausführung des eingeschleusten Codes gleich der Ausführung legitimen Codes erfolgt, stellt dies keine Verletzung einer Sicherheitsvorkehrung dar. Allenfalls kann es bei der Einschleusung des Codes im Server-Computersystem zur Verletzung einer Sicherheitsvorkehrung kommen. Hier liegt daher der besondere Fall vor, dass sich der Täter Zugang zu einem Computersystem verschafft indem er eine Sicherheitsvorkehrung in einem anderen verletzt. Da beide Computersystem jedoch Teil eines Netzwerkes (des Internets) sind, ist dieses als Tatobjekt zu behandeln. Da sich der Täter Zugang zu einem Teil des Netzwerkes verschafft indem er eine, in diesem befindliche Sicherheitsvorkehrung verletzt, besteht – unter den sonstigen Voraussetzungen – eine Strafbarkeit nach § 118a.

Läuft die verwundbare Applikation hingegen nicht in einem privilegierten Sicherheitskontext, was idR der Fall ist, ermöglicht die Ausführung von Code im Browser des Client-Systems ohne Ausnützung weiterer Sicherheitslücken nicht die Durchführung eines Angriffs gegen das Client-System selbst. Daher ist in derartigen Szenarien meist die Web-Applikation und nicht das Client-System Ziel des Angriffs. Es wird versucht, die Zugriffsrechte des Benutzers, in dessen Browser der eingeschleuste Code zur Ausführung gelangt, zu erhalten. Zu diesem Zweck ist es idR erforderlich entweder Benutzername und Passwort oder die sog. Session-ID zu kompromittieren. Eine Session-ID ist ein, vom Server dem Client zugewiesener Wert, den dieser bei jedem Request mitzusenden hat. Dies ermöglicht die einmalige Authentifizierung des Benutzers für eine gesamte Session[104]. Die Gültigkeit einer Session-ID endet durch

[104] Denn HTTP ist, wie in RFC 2616 spezifiziert, ein zustandsloses Protokoll.

Zeitablauf (idR 30 Minuten) oder durch die explizite Beendigung der Session durch den Benutzer (sog. Ausloggen). Sowohl Benutzername und Passwort als auch die Session-ID können sich in Cookies, zweiteres uU auch in der URL selbst befinden. Auf beides hat ein, im Browser laufender Code Zugriff. Meist führt der eingeschleuste Code einen HTTP-Request an einen Server des Angreifers aus um so die erlangten Daten dem Angreifer zur Kenntnis zu bringen.
Da der Angreifer von diesem Zeitpunkt an ein Computerpasswort oder die Session-ID (vergleichbare Daten iSd § 126c Abs 1 Z 2, die den Zugriff auf ein Computersystem ermöglichen) auf sozial-inadäquate Weise besitzt, ist hierdurch bereits der objektive Tatbestand des § 126c Abs 1 erfüllt. Verschafft sich der Angreifer Zugang zur Webapplikation indem er Passwort od. Session-ID verwendet, so verletzt er den Authentifizierungsmechanismus als Sicherheitsvorkehrung, da sein Besitz des Passwortes bzw. der Session-ID als sozial-inadäquat zu qualifizieren ist. Die Ausbeutung der XSS-Schwachstelle selbst stellt hingegen in aller Regel keine solche Verletzung dar, da die Erstellung bzw. Veränderung der Daten dem Angreifer ohne Beeinträchtigung der Datensubstanz der Web-Applikation möglich ist.
Eine Strafbarkeit nach §§ 119 f scheidet hingegen schon deshalb aus, weil sich die, an den Angreifer gesandten Daten nicht mehr am Übertragungsweg befinden.

5.5. Konfigurationsfehler

Ein solcher liegt vor, wenn der Anwender eines Programms dieses derart konfiguriert, dass es eine andere als die intendierte Funktionalität zur Verfügung stellt. Wird beispielsweise versehentlich das gesamte Laufwerk für den Zugriff aus dem Netzwerk freigegeben, so ist bereits das Vorliegen einer Sicherheitsvorkehrung bezüglich des Zugriffs auf die freigegebenen Daten zu verneinen.

5.6. Schwache Passwörter & Default-Passwörter

In Bezug auf § 118a ist hierbei va das Vorliegen einer Sicherheitsvorkehrung und deren Verletzung zu problematisieren.

5.6.1. Schwache Passwörter

Leicht zu erratende Passwörter werden als schwache Passwörter bezeichnet. Ein starkes Passwort ist mindestens 8 Zeichen lang und enthält Zahlen, Sonderzeichen, Groß- und Kleinbuchstaben. Es kann daher jede der 8 Stellen des Passworts einen von ca. 95 verschiedenen Werten annehmen. Daraus ergibt sich eine Menge von 95^8 (6.095.689.385.410.816) verschiedener Passwörter. Versucht nun ein Angreifer das richtige Passwort durch die Verwendung aller möglichen Passwörter zu erraten (sog. Brute Force Attack), betrüge der Zeitaufwand selbst bei einer Geschwindigkeit von 100 Passwörtern pro Sekunde ca. 2,1 Mio Jahre. Beschränken sich die möglichen Passwörter jedoch auf gebräuchliche Wörter zehn verschiedener Sprachen zuzüglich diverser Umformungen, ergibt sich eine Menge von nur ca. 5.000.000 verschiedenen Passwörtern[105]. Dies entspricht 0,00000008 % der ursprünglichen Passwortmenge weshalb ein Brute Force Angriff mit selbiger Geschwindigkeit nur ca. 14 Stunden benötigen würde. Eine andere Form schwacher Passwörter sind jene, die durch Kenntnisse der Privatsphäre des Benutzers erraten werden können (z.B. Geburtsdaten uU iVm Namen von Filmcharakteren od dgl). Die beiden wohl denkbar schwächsten Arten von Passwörtern sind zum einen jene, die mit dem Benutzernamen ident sind (sog. Joe-Accounts[106]) und zum anderen Passwörter, die aus einem Leer-String bestehen, d.h. aus keinem einzigen Zeichen.

[105] *Garfinkel/Spafford/Schwartz*, S. 80
[106] *Garfinkel/Spafford/Schwartz*, S. 78

Zunächst ist zu prüfen ob überhaupt eine Sicherheitsvorkehrung vorliegt. Dies ist zu bejahen, da die Wahrscheinlichkeit der Beeinträchtigung der Vertraulichkeit, Integrität oder Verfügbarkeit von Daten oder Systemen bei einer Implementierung mit Authentifizierungsmechanismus und schwachem Passwort jedenfalls geringer ist als bei einer allfälligen hypothetischen Implementierung ohne Authentifizierungsmechanismus (d.h. ohne Erforderlichkeit einer Passworteingabe). Die Schwäche des Passwortes mindert die Qualität der Sicherheitsvorkehrung, negiert diese jedoch nicht. Selbst bei einem Authentifizierungsmechanismus mit einem aus einem Leer-String bestehenden Passwort gilt, dass dies die Wahrscheinlichkeit eines erfolgreichen Angriffs mindert. Denn viele Password-Dictionaries[107] enthalten keinen Leer-String als mögliches Passwort. Somit stellt jeder Authentifizierungsmechanismus unabhängig von der Schwäche eines Passwortes eine Sicherheitsvorkehrung iSd § 118a StGB dar.

Die Verletzung genannter Sicherheitsvorkehrung ist hiervon unabhängig zu prüfen. Es kommt nur eine solche Verletzung iSd § 118a in Betracht, wie sie sich aus einer systematischen Interpretation zu § 126c Abs 1 ergibt (vgl *3.3.1. § 118a StGB*). Das Vorliegen einer Verletzung ist daher dann zu bejahen, wenn eine Strafbarkeit nach dem Vorbereitungsdelikt des § 126c Abs 1 Fall 2 besteht. Hierfür ist eine Subsumtion des gegenständlichen, schwachen Passwortes unter § 126c Abs 1 Z 2 erforderlich. Entgegen dem ersten Anschein, handelt es sich sogar bei einem, aus einem Leer-String bestehenden Passwort um ein Passwort iSd § 126c Abs 1 Z 2, da dessen Eingabe zur Authentifizierung erforderlich ist. Darüber hinaus wird es technisch ebenso wie alle anderen Passwörter behandelt und meist in verschlüsselter Form gespeichert.

Wird ein Passwort zur Zugangsverschaffung verwendet, dass mit einem, § 126c Abs 1 entsprechenden Vorsatz (einschließlich des, auf die Verwendung des Passwortes zur Begehung des § 118a gerichteten, erweiterten Vorsatzes) vom Täter auf sozial-inadäquate Weise hergestellt, eingeführt, sich verschafft oder besessen wurde, so liegt eine Verletzung iSd § 118a vor.

5.6.2. Default-Passwörter

Hersteller liefern aus Gründen der Bequemlichkeit oft Produkte mit Default-Passwörtern aus. Ändert der Administrator nach erfolgter Installation das Passwort nicht, so ist es jedem Angreifer, der Kenntnis des Default-Passwortes des Herstellers hat, möglich sich zu dem System Zugang zu verschaffen.

Auch hier gilt, dass der Authentifizierungsmechanismus niederer Qualität noch immer eine Sicherheitsvorkehrung darstellt. Denn im Verhältnis zu gar keinem Authentifizierungsmechanismus wird die Wahrscheinlichkeit eines erfolgreichen Angriffs durch einen Authentifizierungsmechanismus mit einem weithin bekannten oder sogar im Internet veröffentlichten Passwort gemindert.

Problematisch kann in diesen Fällen jedoch die Verletzung der Sicherheitsvorkehrung sein. Diese muss darin bestehen, dass der Täter ein Passwort verwendet, das er mit einem, § 126c Abs 1 entsprechenden Vorsatz (wiederum einschließlich des, auf die Verwendung des Passwortes zur Begehung des § 118a gerichteten, erweiterten Vorsatzes) auf sozial-inadäquate Weise hergestellt, eingeführt, sich verschafft oder besessen hat. So wird die Kenntnis von Default-Passwörtern, die zum Allgemeinwissen eines Informatikers gehören (zB „public“ und

[107] Diese werden im Rahmen sog. Dictionary Attacks verwendet um das richtige Passwort zu „erraten“.

„private“ für das Protokoll SNMP[108]), auch für Nicht-Informatiker als sozial-adäquat angesehen werden müssen. Ebenso sozial-adäquat ist die Kenntnis eines Default-Passwortes eines Produktes mit dem sich der Täter aus beruflichen Gründen beschäftigt. Daraus ergibt sich, dass bei der Verwendung desselben Default-Passworts zum Zweck eines Angriffs, der Systemadministrator, der auf sozial-adäquate Weise von diesem Kenntnis erlangt hat, straflos bleibt, sonstige Personen hingegen aufgrund ihrer sozial-inadäquaten Kenntnis des Default-Passwortes strafbar werden können. Dies mag unbillig erscheinen, entsprich aber jenem Fall, in dem einem Kollegen das Passwort mitgeteilt wurde, der andere es jedoch durch einen Brute Force Angriff erlangt hat. Bezüglich weiterer Beispiele vgl *3.1.1. § 126c StGB*.

5.7. Weak File Permissions

Für jeden Benutzer werden - sowohl unter Unix[109] als auch unter Windows[110] - die Zugriffsmöglichkeiten auf eine Datei durch Datei-spezifische Zugriffsrechte, sog. File Permissions determiniert. Zu weit reichenden Zugriffsberechtigungen werden hierbei als Weak File Permissions bezeichnet[111]. Diese entstehen meist durch unachtsame Benutzer bzw. Administratoren oder durch den Fehler eines Programmierers. Widrigsten Falls kann es jedem Benutzer des Computersystems möglich sein die betreffende Datei zu lesen und zu überschreiben.

Bezüglich des Lesens der Datei kommen, je nach Inhalt derselben die §§ 123, 126c (letzterer bezüglich Passwort-Dateien) in Betracht. Die §§ 119 und 119a scheiden hingegen jedenfalls aus, da es sich nicht um, am Übertragungsweg befindliche Daten handelt. Der erfolgreiche Zugriff auf eine Datei ist eine Zugangsverschaffung zu einem Teil eines Computersystems iSd § 118a. Da die File Permissions den Zugriff gestatten, ist das Vorliegen einer Sicherheitsvorkehrung jedoch bereits zu verneinen.

Bezüglich des (Über-)Schreibens der Datei kommen die §§ 126a, 126b in Betracht. Da sich ein Angreifer insbesondere durch die Veränderung von Systemdateien leicht einen administrativen Zugang zu dem Computersystem verschaffen kann, ist auch eine Strafbarkeit nach § 118a näher zu prüfen. Hierfür ist zunächst das Vorliegen einer Sicherheitsvorkehrung zu untersuchen. Wie bereits ausgeführt, stellen die File Permissions keine Sicherheitsvorkehrung dar. Bei der Veränderung von Programmdateien wäre auch an die Sicherheitsvorkehrung Least Privilege zu denken. Diese im Programm implementierte Sicherheitsvorkehrung besteht jedoch nur in der Beschränkung der Funktionalität des laufenden Programms, schützt hingegen in keiner Weise vor Veränderungen der Programmdatei.
Daher kommt es bei der Ausnützung von Weak File Permissions zu keiner Verletzung einer Sicherheitsvorkehrung. Verschafft sich ein Angreifer derart Zugang zu einem Computersystem oder dem Teil eines solchen, besteht keine Strafbarkeit nach § 118a.

5.8. Race Conditions

Die Sicherheitslücke „Race Condition“[112] beschreibt eine Situation, in der das verwundbare Programm zwei von einander abhängige Operationen derart zeitlich nacheinander ausführt, dass es einem Angreifer möglich ist, nach Ausführung der ersten Operation, jedoch vor Ausführung der zweiten, den Programmfluss zu stören oder zu manipulieren, d.h. den „Wettlauf“ zu gewinnen.

---

[108] Simple Network Management Protocol, spezifiziert in RFC 1157
[109] *Frisch*, S. 26 ff
[110] *Russel/Crawford/Gerend*, S 290 ff
[111] vgl CVE-1999-0559
[112] *Garfinkel/Spafford/Schwartz*, S. 507; *Graff/Van Wyk*, S. 10

Wird beispielsweise eine Datei zunächst mit Schreibrechten für alle Benutzer erstellt und erst im zweiten Schritt die Zugriffsmöglichkeit auf den aktuellen Benutzer beschränkt, kann ein Angreifer zwischen diesen beiden Operationen die Datei verändern.[113]
Die wohl häufigste Variante einer Race Condition ist ein sog. time-of-check-to-time-of-use (TOCTTOU) Bug. Hierbei macht sich der Angreifer die zeitliche Differenz zwischen der Abfrage eines Zustandes (time of check) und der Verwendung der abgefragten Daten (time of use) zunutze.[114] So könnte ein Programm im ersten Schritt überprüfen ob eine Datei bestimmte Daten enthält und nur bejahenden Falls die Datei verarbeiten. Wird die Datei von einem Angreifer zwischen den beiden Schritten verändert, bleibt dies unbemerkt.

Für die Beurteilung nach § 118a ist entscheidend, dass zum Zeitpunkt der Ausnützung der Race Condition die zweite Operation noch nicht ausgeführt wurde. Eine allfällige, durch diese zweite Operation getroffene Sicherheitsvorkehrung ist unbeachtlich, da nur eine Verletzung bestehender Sicherheitsvorkehrungen für die Erfüllung des Tatbestandes in Betracht kommt. Die Ausnützung einer Race Condition bedeutet lediglich der zweiten Operation zuvorzukommen. Dies ist für sich genommen, mangels Beeinträchtigung der Datensubstanz einer bestehenden Sicherheitsvorkehrung, jedenfalls nicht vom Tatbestand des § 118a erfasst. Bezüglich der, durch die Race Condition ermöglichte Veränderung von Daten, vgl 5.7. Weak File Permissions.

5.9. Logische Fehler

Hierbei liegt ein Fehler in der Implementierten Logik vor. Wollte ein Programmierer beispielsweise mit folgender Zeile PHP-Code jedem Zugriff gewähren, der sich mit dem Benutzernamen admin und wahlweise dem Passwort secret1 oder secret2 authentifiziert, liegt ein logischer Fehler vor: *if ($username == 'admin' && $password == 'secret1' || $password == 'secret2').* Denn gemäß der Spezifikation der verwendeten Programmiersprache PHP geht der Operator && (logisches Und) dem Operator || (logisches Oder) vor, weshalb bei jedem Benutzernamen in Verbindung mit dem Passwort secret2 die Bedingung erfüllt wäre.

Die Ausnützung solcher Fehler stellt, mangels Beeinträchtigung der Datensubstanz des Programms keine Verletzung einer Sicherheitsvorkehrung dar.

6. Die SANS Top 20 Internet Security Vulnerabilities

Das SANS Institute[115] ist weltweit eine der führenden Organisationen im Bereich der Ausbildung und Zertifizierung von IT-Sicherheitsfachkräften. Die SANS Top 20 Internet Security Vulnerabilities wurden von SANS ua gemeinsam mit dem U.S. Computer Emergency Response Team[116] (US-CERT) des Department of Homeland Security, der US Air Force, dem U.K. National Infrastructure Security Co-Ordination Centre[117] (NISCC), dem Canadian Cyber Incident Response Centre[118] (CCIRC) und der Korea Information Security Agency (KISA) erstellt.
Es handelt sich um eine Sammlung der kritischsten Sicherheitslücken. Das Anliegen der meisten Angreifer ist es nicht, in ein bestimmtes System sondern in möglichst kurzer Zeit in möglichst viele Systeme einzudringen. Daher bedienen sie sich bekannter und weit

[113] Unter Unix könnten die beiden Operationen aus den Befehlen touch und chmod bestehen. Um die Race Condition zu vermeiden, wären umask und touch zu verwenden.
[114] vgl *Lowery*
[115] http://www.sans.org
[116] http://www.us-cert.gov
[117] http://www.niscc.gov.uk
[118] http://www.ocipep-bpiepc.gc.ca/ccirc/

verbreiteter Sicherheitslücken. Die bei weitem meisten erfolgreichen Angriffe lassen sich daher auf eine Ausnützung der SANS Top 20 Internet Security Vulnerabilities zurückführen.

Gegenstand der Erörterung ist Version 6.01 der SANS Top 20 Internet Security Vulnerabilities, veröffentlicht am 28. November 2005[119]. Die erste Version dieser Liste von Sicherheitslücken wurde im Juni 2000 als SANS Top 10[120] veröffentlicht.
Die englischsprachige Gliederung der SANS Top 20 wird im Folgenden beibehalten um eine übersichtliche Darstellung der Sicherheitslücken zu ermöglichen.
Ausgangspunkt der Untersuchung der einzelnen Sicherheitslücken war stets die National Vulnerability Database[121].

6.1. Top Vulnerabilities in Windows Systems

6.1.1. Windows Services

Manche Teile des Betriebssystems Windows sind als Dienste (Services) implementiert. Diese im Hintergrund laufenden Programme sind für den Anwender nicht sichtbar. So erfolgt beispielsweise die Ausführung und Reihung von Druckaufträgen durch den Dienst „Druckerwarteschlange“ (Print Spooler Service). Die weiteren betroffenen Dienste sind MSDTC and COM+, Plug and Play, Server Message Block, Exchange SMTP, Message Queuing, License Logging, WINS, NNTP, NetDDE und Task Scheduler.

6.1.1.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2005-2120, CVE-2005-1984, CVE-2005-1983, CVE-2005-1978, CVE-2005-1206, CVE-2005-0045, CVE-2005-0560, CVE-2005-0059, CVE-2005-0050, CVE-2004-0567, CVE-2004-0574, CVE-2004-0206, CVE-2004-0212 und CVE-2004-1080 bestehen in Form eines möglichen Buffer Overflows. Jede der genannten Sicherheitslücken ermöglicht es von einem entfernten System aus, beliebige Befehle auf einem verwundbaren Computersystem auszuführen. Dies erfüllt die Voraussetzung der Zugangsverschaffung. Die Sicherheitsvorkehrung besteht darin, dass die genannten Dienste die Funktionalität beliebige Befehle auszuführen nicht implementieren. Diese wird bei Buffer Overflows durch das Überschreiben von Speicherbereichen, die Teil des laufenden Dienstes und damit Teil der Sicherheitsvorkehrung sind, verletzt. Das Tatbestandsmerkmal der Verletzung einer Sicherheitsvorkehrung im Computersystem iSd § 118a StGB ist daher erfüllt, wodurch bei Vorliegen der sonstigen Voraussetzung die Zugangsverschaffung durch Ausnützung einer der genannten Sicherheitslücken gem. § 118a StGB strafbar ist.

6.1.1.2. Denial of Service

Der Microsoft Distributed Transaction Coordinator (MSDTC) Proxy enthält die Sicherheitslücke CVE-2005-2119, in Form eines logischen Fehlers, der darin besteht, dass unabhängig vom konkreten Speicherbedarf stets 4 Kilobyte alloziert werden. Dies ermöglicht es einem entfernten Angreifer bestimmte Speicherbereiche zu überschreiben, jedoch nicht Befehle zur Ausführung zu bringen. Mangels Zugangsverschaffung kommt § 118a daher nicht in Betracht. Durch das Überschreiben der Speicherbereiche kann es jedoch zu einem Denial

[119] http://www.sans.org/top20/
[120] vgl http://www.sans.org/top20/2000/; eine exzellente Analyse der SANS Top 10 bietet *Cooper/Northcutt/Fearnow/Frederick* S. 39 ff
[121] http://nvd.nist.gov; Informationen über einzelne CVE-Nummern sind direkt über http://nvd.nist.gov/nvd.cfm?cvename=*CVE-Nummer* erhältlich. Zur erleichterten Abfrage mehrerer CVE-Nummern empfiehlt sich das, anlässlich der nun folgenden Untersuchung von Sicherheitslücken entwickelte Werkzeug CVEParser. Es wurde unter der LGPL unter http://rocketscience.lukasfeiler.com/CVEParser/ veröffentlicht.

of Service kommen. Da den MSDTC Proxy zum Absturz zu bringen eine schwere Störung der Funktionsfähigkeit des betreffenden Computersystems darstellt, erfüllt der Angreifer dadurch – die fehlende Verfügungsbefugnis über das Computersystem vorausgesetzt – den objektiven Tatbestand des § 126b.

6.1.2. Internet Explorer

Microsoft Internet Explorer ist mit einem Marktanteil von ca. 85%[122] der bei weitem am häufigsten genutzte Web-Browser. Durch seine große Verbreitung stellt er ein sehr beliebtes Angriffsziel dar.

6.1.2.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2003-1048, CVE-2004-0216, CVE-2004-0566, CVE-2004-0842, CVE-2004-1050, CVE-2005-0055, CVE-2005-0554, CVE-2005-0555, CVE-2005-1211, CVE-2005-1988[123], CVE-2005-1990[124], CVE-2005-2087 und CVE-2005-2127 ermöglichen die Ausführung beliebiger Befehle durch Ausnützung eines Buffer Overflows. Dadurch liegt eine Zugangsverschaffung durch Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

6.1.2.2. Zugangsverschaffung durch Ausnützung einer Race Condition

Da es bei der Ausnützung einer Race Condition zu keiner Verletzung einer Sicherheitsvorkehrung kommt, ist die Zugangsverschaffung zu einem Computersystem unter Ausbeutung der Sicherheitslücke CVE-2005-0553 nicht vom Tatbestand des § 118a erfasst.

6.1.2.3. Überwindung von Sicherheitszonen

Internet Explorer verwendet ein Konzept namens „Security Zones“[125]. Hierbei sind die Zonen „Local Intranet“, „Internet“, „Trusted Sites“, „Restricted Sites“ und „Local Machine“ mit unterschiedlichen Sicherheitseinstellungen versehen. Gelingt es einer Website aus der Internet-Zone Befehle in Form von Scripts in der Zone Local Intranet auszuführen, kann dies einem Angreifer Zugang zu dem Computersystem des Benutzers verschaffen.

Die Sicherheitslücke CVE-2005-0053[126] ermöglicht die Umgehung einer im Internet Explorer implementierten Sicherheitsvorkehrung, die nur das Kopieren bestimmter Dateitypen per Drag & Drop aus der Internet-Zone ermöglicht. Die Sicherheitsvorkehrung kann durch die Verwendung bestimmter Dateierweiterungen, die jedenfalls zugelassen werden, umgangen werden. Erst in Kombination mit anderen Sicherheitslücken stellt CVE-2005-0053 eine erhebliche Gefahr dar. Durch die Ausnützung von CVE-2005-0053 selbst kommt es weder zu einer Zugangsverschaffung noch zu einer Verletzung einer Sicherheitsvorkehrung.

6.1.3. Windows Libraries

Als Libraries bzw. Bibliotheken werden wieder verwendbare Programmteile bezeichnet. Diese werden meist in Dateien mit der Endung .dll gespeichert, weshalb auch die Bezeichnung „DLLs“ üblich ist. Sowohl von Microsoft entwickelte Programme als auch Produkte von Drittherstellern verwenden für standardisierte Aufgaben wie die Interpretation von HTML-Code regelmäßig die, mit dem Betriebssystem Windows verfügbaren Bibliotheken. Dies führt dazu, dass eine Sicherheitslücke in einer Windows-Bibliothek eine Auswirkung auf eine Reihe unterschiedlichster Programme haben kann.

---

[122] Quelle: http://marketshare.hitslink.com/report.aspx?qprid=3
[123] vgl http://www.kb.cert.org/vuls/id/965206
[124] vgl http://www.securityfocus.com/bid/14511/discuss
[125] *Bott/Siechert/Stinson*, S. 818 ff
[126] vgl http://www.securityfocus.com/bid/11466/discuss

6.1.3.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2004-0201, CVE-2004-0200, CVE-2004-0214, CVE-2004-0575, CVE-2004-0597, CVE-2004-1049, CVE-2004-1244, CVE-2005-0057, CVE-2005-1208, CVE-2005-1219, CVE-2005-2118, CVE-2005-2122, CVE-2005-2123, CVE-2005-2124 und CVE-2005-2128 ermöglichen es durch Buffer Overflows beliebigen Code auf einem verwundbarem Computersystem zur Ausführung zu bringen. Bei der Ausnützung dieser Sicherheitslücken kommt es daher zu einer Zugangsverschaffung iSd § 118a durch Verletzung einer Sicherheitsvorkehrung.

6.1.3.2. Umgehung von Zugriffsbeschränkungen

Die Sicherheitslücke CVE-2003-1041 ermöglicht es, auf dem Computersystem des Opfers gespeicherte Programme zur Ausführung zu bringen. Sie besteht darin, dass es durch besondere Formatierung einer URL möglich ist, eine Sicherheitsvorkehrung, die die Ausführung lokaler Programme verhindern soll, zu umgehen. Da die URL auf Grund ihrer besonderen Formatierung vom angegriffenen System bloß nicht als unzulässig erkannt wird, liegt keine Verletzung einer Sicherheitsvorkehrung vor. Eine Strafbarkeit nach § 118a ist daher zu verneinen.

6.1.3.3. Umgehung von dateitypenbasierten Sicherheitsvorkehrungen

Microsoft Internet Explorer öffnet grundsätzlich nur unbedenkliche Dateitypen wie HTML oder Bilddateien automatisch. Potentiell gefährliche Dateien werden hingegen nur nach erfolgter Genehmigung durch den Benutzer geöffnet bzw. ausgeführt. Für die Bestimmung des Typs einer Datei ist neben der Dateierweiterung (zB .html) die sog. CLSID (Class ID)[127] relevant. Die Sicherheitslücke CVE-2004-0420[128] besteht nun darin, dass es durch Angabe einer falschen CLSID möglich ist, beliebige Dateien ohne Genehmigung des Benutzers unmittelbar zur Ausführung zu bringen.
Da hier keine Verletzung sondern bloß eine Umgehung einer Sicherheitsvorkehrung vorliegt, ist der objektive Tatbestand des § 118a nicht erfüllt. Eine Zugangsverschaffung unter Ausnützung von CVE-2004-0420 ist daher straflos.

Wird eine lokale Datei mit nicht registrierter Dateierweiterung geöffnet, kann eine in der Datei gespeicherte CLSID beeinflussen welches Programm hierzu verwendet werden soll. Die Sicherheitslücke CVE-2005-0063[129] besteht darin, dass es dadurch möglich ist, Sicherheitsvorkehrungen, die auf Dateierweiterungen basieren zu umgehen. Mangels Verletzung einer Sicherheitsvorkehrung besteht keine Strafbarkeit nach § 118a.

6.1.3.4. Überwindung von Sicherheitszonen

Bezüglich des Konzepts von Sicherheitszonen vgl bereits 6.1.2.3. Überwindung von Sicherheitszonen. Die Sicherheitslücke CVE-2004-1043[130] ermöglicht es, eine Website zu gestalten bei deren Aufruf beliebige Befehle in der Sicherheitszone „Local Machine“ ausgeführt werden. Hierzu wird ein neues HTML Help-Fenster erzeugt, dass bei Namensgleichheit mit einem bestehenden HTML Help-Fenster in dessen Sicherheitszone ausgeführt wird. Die Wahl eines bereits verwendeten Namens stellt keine Verletzung einer Sicherheitsvorkehrung dar. Die vorliegende Überwindung einer Sicherheitsvorkehrung erfüllt somit nicht den objektiven Tatbestand des § 118a.

[127] vgl http://www.w3.org/Addressing/clsid-scheme
[128] vgl http://www.kb.cert.org/vuls/id/106324
[129] vgl http://www.idefense.com/intelligence/vulnerabilities/display.php?id=231
[130] vgl http://www.kb.cert.org/vuls/id/972415

6.1.3.5. Script Injection

Die Sicherheitslücken CVE-2005-1191 und CVE-2005-2117 ermöglichen es beliebige Befehle auszuführen, wenn eine entsprechend formatierte, lokale Datei zur Voransicht gebracht wird. Die, sich in der Web View Library (webvw.dll) befindende Sicherheitslücke besteht darin, dass Meta-Daten wie der Name des Autors einer Datei vor der Weiterverarbeitung in HTML-Code nicht hinreichend überprüft werden. Da die Funktionalität grundsätzlich auf eine Voransicht beschränkt ist, liegt die Sicherheitsvorkehrung Least Privilege vor. Mangels Beeinträchtigung der Datensubstanz besagter Sicherheitsvorkehrung, liegt keine Verletzung iSd § 118a vor.

6.1.4. Microsoft Office and Outlook Express

Microsoft Office ist die weltweit gebräuchlichste sog. „Productivity Suite". Sie besteht ua. aus den Programmen Outlook, Word, PowerPoint, Excel, Visio, FrontPage und Access. Outlook Express ist ebenso wie Outlook ein E-Mail-Programm. Es ist jedoch nicht teil des Office-Pakets, weist einen geringeren Funktionsumfang auf und wird bei jeder Installation von Windows automatisch mitinstalliert.

6.1.4.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücke CVE-2004-0848 besteht in Form eines möglichen Buffer Overflows. Durch eine speziell formatierte URL, die eine Datei mit der Endung „.doc" oder „.rtf" referenziert, kann ein Puffer zum Überlaufen gebracht und können so in weiterer Folge beliebige Befehle ausführt werden.
CVE-2005-0044 ermöglicht ebenso die Ausführung beliebigen Codes durch Ausnützung eines Buffer Overflows. Das Opfer muss lediglich dazu gebracht werden ein entsprechendes OLE[131]-Object zu öffnen.
Durch die Sicherheitslücke CVE-2005-1213 in Microsoft Outlook Express ist es einem von Outlook Express befragten NNTP[132] Server möglich durch Ausnützung eines Buffer Overflows beliebigen Code zur Ausführung zu bringen.
Da alle drei genannten Sicherheitslücken in Form eines Buffer Overflows bestehen, erfüllt ihrer Ausnützung die Tatbestandsmerkmale der Zugangsverschaffung durch Verletzung einer Sicherheitsvorkehrung iSd § 118a.

6.1.5. Windows Configuration Weaknesses

Für den Bereich der Windows-Konfigurationsfehler enthält SANS Top 20 keine konkreten CVE-Nummern. Diese Konfigurationsfehler lassen sich jedoch in drei Gruppen teilen.

6.1.5.1. Schwache Verschlüsselung von Passwörtern

Um eine Abwärtskompatibilität zu erreichen, sind sowohl Windows NT, 2000 als auch XP (nicht hingegen Windows 2003 Server) nach der Erstinstallation so konfiguriert, dass alle Passwörter lokal als LM Password Hash (LANMAN Hash) gespeichert werden. Bei LM Password Hashes kommen nur schwache Verschlüsselungsverfahren zum Einsatz, weshalb ein „Knacken" der Verschlüsselung durch Brute Force od. Dictionary Attacks in relativ kurzer Zeit möglich ist. Verschafft sich jemand derart ein Passwort, erfüllt er den objektiven Tatbestand des § 126c. Verwendet der Angreifer das so erlangte Passwort zur Zugangsverschaffung, so liegt eine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

[131] Object Linking and Embedding: diese Technik ermöglicht die Einbindung externer, mit Office erstellter Datenquellen.
[132] Network News Transfer Protocol, spezifiziert in RFC 977.

6.1.5.2. Default-Server-Konfigurationen

Manche Default-Einstellungen in Microsofts Serverplattform IIS (Internet Information Services) stellen eine Sicherheitslücke dar. So sind manche Accounts wie IUSR_<Computername> mit zu weit gehenden Rechten ausgestattet. Darüber hinaus sind manche Dienste wie FTP, NNTP oder SMTP auch wenn kein Bedarf für diese besteht in der Default-Konfiguration aktiviert, was die Angriffsfläche erhöht.
Ein Hacking unter Ausnützung dieser Sicherheitslücken erfüllt nicht den Tatbestand des § 118a, da es an einer Sicherheitsvorkehrung mangelt.

6.1.5.3. Schwache Passwörter und Default-Passwörter

Nach Installation der Microsoft Data Engine (MSDE) oder des Microsoft SQL Server Desktop (MSDE2000) wird ein administrativer Datenbank-Account namens „sa“ mit einem leeren Passwort erstellt. Da dessen Angabe im Rahmen der Authentifizierung erforderlich ist handelt es sich um ein Passwort iSd § 126c Abs 1 Z 2. Die Verwendung des Passwortes zur Zugangsverschaffung stellt daher eine Verletzung iSd § 118a dar. Das Vorliegen einer Sicherheitsvorkehrung ist ebenso zu bejahen, da durch das, aus einem Leer-String bestehende Passwort die Wahrscheinlichkeit eines erfolgreichen Angriffs im Verhältnis zum Fehlen jeglichen Passwortes gemindert wird. Eine Zugangsverschaffung durch Ausnützung dieser Sicherheitslücke ist daher geeignet den objektiven Tatbestand des § 118a zu erfüllen.

6.2. Top Vulnerabilities in Cross-Platform Applications

6.2.1. Backup Software

Durch die zunehmende Zentralisierung von Backup-Funktionalitäten auf wenige Systeme (sog. Backup Server), die die Datensicherung aller übrigen Computersysteme besorgen, ist die auf Backup Servern eingesetzte Backup Software ein überaus beliebtes Ziel von Angriffen geworden. Über den Backup Server ist meist der Datenbestand des gesamten Unternehmens zugänglich.[133]

6.2.1.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2004-1172, CVE-2005-0260, CVE-2005-0491, CVE-2005-0581, CVE-2005-0582, CVE-2005-0773, CVE-2005-1009, CVE-2005-1019, CVE-2005-1018, CVE-2005-1272, CVE-2005-1547, CVE-2005-2051, CVE-2005-2079, CVE-2005-2535, CVE-2005-2996 und CVE-2005-3116 ermöglichen die Ausführung beliebiger Befehle unter Ausnützung eines Buffer Overflows. CVE-2005-2715 besteht in Form eines Format String Bugs mit selbiger Auswirkung. Die Ausnützung dieser Sicherheitslücken stellt somit eine Verletzung einer Sicherheitsvorkehrung iSd § 118a dar.

6.2.1.2. Backdoor Accounts und statische Passwörter

Bei den Sicherheitslücken CVE-2005-0349 und CVE-2005-0496 hat der Hersteller (Computer Associates bzw. Arkeia) seine Produkte mit einem (nicht dokumentierten) Backdoor Account ausgestattet. CVE-2005-2611 beschreibt die Tatsache, dass verteilte Komponenten der Backupsoftware VERITAS Backup Exec eine Authentifizierung mittels eines statischen Passwortes durchführen.
Für alle drei Sicherheitslücken gilt, dass es sich um Authentifizierungsdaten iSd § 126c Abs 1 Z 2 handelt. Deren Verwendung zur Zugangsverschaffung stellt daher eine Verletzung einer Sicherheitsvorkehrung iSd § 118a dar.

[133] Die Möglichkeit einer teilweisen Migration dieses Risikos besteht im Off-Site-Storage der verwendeten Speichermedien; vgl *Preston*. S. 54 ff

6.2.1.3. Umgehung der Authentifizierung

Die Sicherheitslücke CVE-2005-0357 besteht darin, dass die betroffene Backupsoftware auch die Authentifizierungsmethode AUTH_UNIX[134] verwendet. Dadurch ist es einem Angreifer möglich durch Angabe einer User-ID den Authentifizierungsmechanismus zu umgehen. Mangels Verletzung einer Sicherheitsvorkehrung ist die Zugangsverschaffung daher nicht nach § 118a strafbar.

6.2.1.4. Überwindung von Autorisationsmechanismen

CVE-2005-0358 beschreibt einen Fehler der betroffenen Backupsoftware in der Autorisation von bereits authentifizierten Benutzern. Nach erfolgreicher Authentifizierung erhält der Benutzer ein Token, das auch das Ausmaß seiner Berechtigungen enthält. Die Sicherheitslücke besteht nun darin, dass ein für einen nicht-administrativen Account ausgegebener Token zu einem administrativen Token gewandelt werden kann.[135] Hierbei liegt keine Verletzung einer Sicherheitsvorkehrung in Form einer Beeinträchtigung der Datensubstanz des Programms vor, da der ausgegebene und in weiterer Folge manipulierte Token nicht als Teil des Programms zu beurteilen ist. Die so erfolgende Überwindung des Autorisationsmechanismus erfüllt daher nicht den objektiven Tatbestand des § 118a.

6.2.1.5. Umgehung von Zugriffsbeschränkungen

Die Sicherheitslücke CVE-2005-0583 ermöglicht es beliebige Dateien auf einem verwundbaren System zu erzeugen. Die Beschränkung auf ein bestimmtes Verzeichnis kann durch mehrmalige Verwendung der Zeichenfolge „../“, die das jeweils darüber liegende Verzeichnis referenziert, umgangen werden. Da diese Umgehung zu keiner Beeinträchtigung der Datensubstanz des Programms führt, liegt keine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

6.2.1.6. Zugang ohne Authentifizierung

Die Sicherheitslücke CVE-2005-0771 besteht darin, dass Modifikationen der Registry[136] mittels RPC-Request ohne vorherige Authentifizierung möglich sind. Mangels Schutz durch einen Authentifizierungsmechanismus ist bereits das Vorliegen einer Sicherheitsvorkehrung zu verneinen.

6.2.1.7. Denial of Service

Die Sicherheitslücke CVE-2005-0772 ermöglicht es einem Angreifer durch das Senden einzelner Datenpakete den Backup Agent eines zu sichernden Systems zum Absturz zu bringen. Hierdurch ist eine Sicherung des betreffenden Systems nicht mehr möglich. Da die Möglichkeit eine Sicherung eines Computersystems vorzunehmen eine wesentliche Funktionalität des Systems darstellt, liegt eine schwere Störung iSd § 126b vor.

6.2.2. Anti-Virus Software

Durch den verbreiteten Einsatz von Anti-Viren-Lösungen auf Desktops, Servern (insbesondere Mail Servern) und Gateways, sind sie zu einem beliebten Angriffsziel geworden.

---

[134] *Garfinkel/Spafford/Schwartz*, S. 410

[135] vgl http://www.kb.cert.org/vuls/id/407641

[136] Die Registry ist eine von Microsoft Windows verwendete zentrale Datenbank zur Speicherung von Systeminformationen; vgl *Russel/Crawford/Gerend*, S. 1400 ff

6.2.2.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2005-3029, CVE-2005-2385, CVE-2005-2957, CVE-2005-3154, CVE-2005-2450, CVE-2005-2920, CVE-2005-1693, CVE-2005-2720, CVE-2005-2041, CVE-2004-2405, CVE-2005-3664, CVE-2005-0350, CVE-2005-3664, CVE-2005-3142, CVE-2005-0643, CVE-2005-0644, CVE-2005-2768, CVE-2005-0249 und CVE-2005-0533, CVE-2005-1693 ermöglichen die Ausführung beliebigen Codes durch Ausnützung eines Buffer Overflows bzw. eines Format String Bugs. Dies stellt eine Verletzung einer Sicherheitsvorkehrung iSd § 118a dar.

6.2.2.2. Umgehung von Zugriffsbeschränkungen

Auf Grund der Sicherheitslücken CVE-2005-3030, CVE-2005-2384 und CVE-2005-2670 ist es möglich beliebige Dateien auf einem verwundbaren System zu erzeugen. Die Anti-Viren Software muss lediglich dazu gebracht werden ein komprimiertes Archiv nach Viren zu untersuchen. Durch die Verwendung relativer, darüber liegende Verzeichnisse referenzierende Pfade (zB „../../“) kann eine Extraktion in beliebige Verzeichnisse bewirkt werden. Da es hierbei zu keiner Beeinträchtigung der Datensubstanz der Anti-Viren Software kommt, liegt keine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

6.2.2.3. Eskalation bestehender Privilegien

Die Sicherheitslücke CVE-2005-3663[137] beschreibt eine, bei Kaspersky Anti-Virus 5.0 auftretende Situation, in der auf Grund eines Programmierfehlers die Datei C:\program.exe ausgeführt wird – sofern diese besteht. Gelingt es daher einem regulären Benutzer des betroffenen Systems, die Datei C:\program.exe zu erzeugen, so kann er beliebige Befehle im Sicherheitskontext der Anti-Viren Software zur Ausführung bringen und sich so erweiterten Zugang zu dem Computersystem verschaffen. Da diese Zugangsverschaffung zu einem Teil eines Computersystems ohne Verletzung einer Sicherheitsvorkehrung erfolgt, ist eine Strafbarkeit nach § 118a zu verneinen.

6.2.2.4. Exkurs: Malware – Verletzung einer Sicherheitsvorkehrung durch Social Engineering?

Im Zusammenhang mit Anti-Viren Software wird im Folgenden auch eine Zugangsverschaffung durch den Einsatz von Malware[138] erörtert. Malware dient als Überbegriff für Viren, Würmer[139], trojanische Pferde[140] und andere Schädlinge. Von besonderem rechtlichem Interesse sind jene Schädlinge, die nicht automatisch, sondern erst durch den Benutzer zur Ausführung gelangen. So erfolgt ein „Angriff“ mit einem trojanischen Pferd dadurch, dass der Benutzer zur Installation eines Programms verleitet wird, das neben der erwünschten Funktionalität auch für den Benutzer schädlichen Code enthält. Auch viele E-Mail-Viren erfordern, dass der Benutzer ein Attachment öffnet, das von selbst nicht zur Ausführung gelangen würde.[141]
Ein Wesensmerkmal dieser Angriffe ist daher den Benutzer dazu zu bringen eine bestimmte Handlung zu setzen. Dies bezeichnet man als Social Engineering[142]. Problematisch erscheinen

[137] http://www.idefense.com/intelligence/vulnerabilities/display.php?id=340
[138] vgl NIST SP 800-83, S. 15 ff
[139] Bezüglich Definitionen der Begriffe Virus und Wurm, vgl *Garfinkel/Spafford/Schwartz*, S. 736; NIST SP 800-83, S. 15 ff
[140] Entgegen der mythologischen Bedeutung ist leider auch die Bezeichnung „Trojaner“ üblich; vgl NIST SP 800-83, S. 18 f
[141] vgl *Newberry*, S. 4
[142] *Peikari/Chuvakin*, S. 199 ff; vgl CERT Advisory CA-1991-04 Social Engineering, http://www.cert.org/advisories/CA-1991-04.html

in diesem Zusammenhang insbesondere die Tatbestandsmerkmale der Zugangsverschaffung und der Verletzung einer Sicherheitsvorkehrung iSd § 118a.

Der Begriffskern der Zugangsverschaffung erfasst jene Sachverhalte, in denen der Angreifer zum Zeitpunkt der Zugangsverschaffung auch bestimmen kann in welcher Form er in dem Computersystem tätig werden will. Die beim Einsatz von Malware verwirklichten Sachverhalte unterscheiden sich hiervon durch ein zeitliches Moment, denn die nach erfolgter Zugangsverschaffung ausgeführten Aktionen sind bereits zum Zeitpunkt der Programmierung der Malware determiniert. Da es dennoch im Belieben des Programmierers der Malware steht, in welcher Form er in betroffenen Computersystemen tätig wird, kann eine Subsumtion unter den Begriff der Zugangsverschaffung erfolgen. Diesem Interpretationsergebnis stehen weder die Gesetztesmaterialien noch die CyCC entgegen, da sie bezüglich des zeitlichen Moments der Zugangsverschaffung keine Regelung enthalten.

Weiters ist es fraglich ob es zu einer Verletzung einer, im Computersystem befindlichen Sicherheitsvorkehrung kommt. Hierbei kommen durch die Malware ausgeführte Operationen grundsätzlich nicht in Betracht, da zu diesem Zeitpunkt bereits eine Zugangsverschaffung vorliegt. Da es sich um Social Engineering handelt, kommt es tatsächlich nur zu einer Verletzung der Sicherheitsvorkehrung Mensch.[143] Der Mensch ist jedoch keine Sicherheitsvorkehrung „im Computersystem". Derartige Sachverhalte sind mit jenem Fall zu vergleichen, in dem der Täter einen Angestellten eines Unternehmens anruft, sich als Administrator ausgibt und ihn anweist den Befehl *„cat /etc/passwd | mail joe@example.com"* auszuführen um sich so alle Benutzeraccounts per E-Mail zukommen zu lassen.

Es gilt daher festzuhalten, dass Angriffe durch Social Engineering mangels Verletzung einer Sicherheitsvorkehrung *im Computersystem*, nicht den objektiven Tatbestand des § 118a erfüllen. Somit ist auch ein Angriff durch Malware, der Social Engineering voraussetzt grundsätzlich nicht nach § 118a strafbar.

Eine Strafbarkeit von Social Engineering Angriffen kann sich allenfalls ergeben, wenn die durch den Benutzer ausgelösten Operationen zur Installation einer Backdoor führen. Die erforderliche Verletzung einer Sicherheitsvorkehrung ist in zwei Varianten denkbar. Zum einen kann es bei der Installation der Backdoor zu einer Beeinträchtigung der Substanz bestehender Programme oder des Betriebssystems kommen, was eine Verletzung der Sicherheitsvorkehrung Least Privilege darstellt. Dies ist jedoch bei der Installation einer Backdoor nicht notwendigerweise der Fall.[144] Die zweite Variante der Verletzung einer Sicherheitsvorkehrung besteht darin, dass die Backdoor einen Authentifizierungsmechanismus implementiert und die vom Angreifer sozial-inadäquat besessenen Authentifizierungsdaten somit unter § 126c Abs 1 Z 2 zu subsumieren sind. Werden diese nachfolgend zur Zugangsverschaffung gebraucht liegt eine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

6.2.3. PHP-based Applications

PHP[145] ist die beliebteste, serverseitige Scriptsprache für die Entwicklung von Web-Applikationen. PHP-Applikationen weisen häufig SQL-Injection und Command-Injection Sicherheitslücken auf (vgl bereits 5.3. SQL-Injection und 5.2. Command-Injection).

[143] vgl *Mitnick/Simon/Wozniak*
[144] So könnte eine Backdoor unter Linux beispielsweise als Daemon mit eigenen Start-Up-Scripts unter /etc/rc.d/ ausgestaltet sein. Dessen Installation würde keine Veränderung bestehender Dateien erfordern und daher keine Verletzung iSd § 118a darstellen.
[145] http://www.php.net

6.2.3.1. Race Conditions

Die Sicherheitslücke CVE-2004-0594[146] beschreibt eine Race Condition im Zuge der Allokation und Initialisierung bestimmter, von PHP verwendeter Datenstrukturen. Diese tritt ein, wenn der Programmablauf auf Grund des Erreichens des maximal zulässigen Speicherverbrauchs[147] abrupt beendet wird. Zunächst muss es dem Angreifer gelingen seinen Maschinencode in jenem Speicherbereich des Heap zu platzieren, der dann für die Allokation der Datenstruktur verwendet wird. Die Race Condition besteht darin, nach der Speicherallokation, jedoch vor der Initialisierung des Speicherbereichs einen Abbruch des Programmablaufs zu erwirken. Gelingt dies wird aus dem noch nicht initialisierten – und damit noch aus den Daten des Angreifers bestehenden – Speicherbereich ein sog. Destructor Pointer ausgelesen. Dieser verweist auf, im Rahmen der Dekonstruktion der Datenstruktur auszuführenden Code. Da der Angreifer den Inhalt des Destructor Pointer bestimmen kann, ist es ihm möglich auf eigenen Maschinencode zu verweisen und diesen dadurch zu Ausführung zu bringen. Da es bei der Ausnützung dieser Race Condition zu keiner Beeinträchtigung der Datensubsatz des Programms kommt, liegt keine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

Bei CVE-2005-3389[148] handelt es sich ebenso um eine Race Condition. Sie ermöglicht es eine als „register_globals“ bekannte Funktionalität zu aktivieren. Dadurch werden vom HTTP-Client (dem Browser) kommende Daten als globale Variable zur Verfügung gestellt. Wurde dies bei der Programmierung einer PHP-Applikation nicht berücksichtigt, kann es zu Veränderungen des logischen Programmablaufs kommen.
Die Sicherheitslücke besteht darin, dass innerhalb der Funktion parse_string()[149]die Konfigurationsoption register_globals zeitweilig aktiviert wird. Gelingt es dem Angreifer das Überschreiten des maximalen Speicherbedarfs zwischen Aktivierung und Deaktivierung von register_globals zu erwirken, wird die Funktion parse_string() abgebrochen, wodurch register_globals aktiviert bleibt. Da es durch die Ausnützung der Race Condition zu keiner Beeinträchtigung der Datensubstanz des Programms kommt, stellt sie keine Verletzung einer Sicherheitsvorkehrung iSd § 118a dar.
Die Aktivierung von register_globals ermöglicht es einem Angreifer, ansonsten uninitialisierten, globalen Variablen einen Wert zuzuweisen. Dies eignet sich insbesondere zur Umgehung von Authentifizierungsmechanismen. Wird eine globale, uninitialisierte Variable dafür verwendet den Authentifizierungsstatus zu speichern, kann deren Wert durch den Angreifer gesetzt werden. Da diese Umgehung keine Verletzung einer Sicherheitsvorkehrung darstellt, besteht keine Strafbarkeit nach § 118a.

6.2.3.2. Überschreiben Globaler Variablen

Die Sicherheitslücke CVE-2005-3390[150] beschreibt, dass es unter Verwendung der Funktionalität des File Uploads[151] möglich ist globale Variable zu überschreiben. Dies ermöglicht es einem Angreifer eigenen Script-Code zur Ausführung zu bringen[152]. Auch eine PHP-Anwendung implementiert die Sicherheitsvorkehrung Least Privilege. Da der Wert von Variablen unmittelbar Auswirkungen auf den logischen Programmfluss hat, sind Variable als Teil der Anwendung zu beurteilen. Das Überschreiben von Variablen ist daher eine Verletzung einer Sicherheitsvorkehrung iSd § 118a.

---

[146] vgl http://marc.theaimsgroup.com/?l=bugtraq&m=108981780109154&w=2
[147] Dieser ist durch die Konfigurationsoption memory_limit bestimmt; vgl http://at.php.net/ini.core
[148] vgl http://www.hardened-php.net/advisory_192005.78.html
[149] vgl http://www.php.net/manual/en/function.parse-str.php
[150] vgl http://www.hardened-php.net/advisory_202005.79.html
[151] RFC 1867
[152] vgl http://www.hardened-php.net/globals-problem

6.2.4. Database Software

Zur Speicherung größerer Mengen strukturierter Daten sind meist Datenbanken das Mittel der Wahl.

6.2.4.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2004-0638, CVE-2004-1363, CVE-2004-1371, CVE-2004-1774, CVE-2004-0628, CVE-2004-0836, CVE-2005-0684, CVE-2005-1274, CVE-2005-2558, CVE-2005-0247 und CVE-2004-1372 ermöglichen die Ausführung beliebiger Befehle durch Ausnützung eines Buffer Overflows. Es handelt sich hierbei um eine Verletzung einer Sicherheitsvorkehrung iSd § 118a.

6.2.4.2. Eskalation von Privilegien

Die Sicherheitslücke CVE-2004-0637[153] ermöglicht es einem authentifizierten Benutzer beliebige Statements in einer Oracle-Datenbank mit Rechten des administrativen Accounts DBA auszuführen. Dies ist möglich, da das Paket ctxsys.driload öffentlich ist, die darin enthaltenen Prozeduren mit DBA-Rechten ausgestattet sind und keine Überprüfung der auszuführenden SQL-Statements erfolgt. Da bereits das Vorliegen einer Sicherheitsvorkehrung zu verneinen ist, verwirklicht eine Zugangsverschaffung unter Ausnützung von CVE-2004-0637 nicht den objektiven Tatbestand des § 118a.

CVE-2004-1338[154] betrifft ebenso Oracle. Diese äußerst komplexe Sicherheitslücke ergibt sich aus dem Zusammenwirken dreier vordefinierter Tabellen eines Delete-Triggers[155] und eine Packages[156]. Die mit einem Delete-Trigger versehene Tabelle #1[157] ermöglicht es jedem Benutzer eine Löschung von Datensätzen auszuführen. Dadurch kann jeder Benutzer die Ausführung des Delete-Triggers bewirken. Dieser liest aus den Tabellen #2 und #3 Funktionen aus und bringt diese mit erhöhten Privilegien zur Ausführung. Die Tabellen #2 und #3 können von unprivilegierten Benutzern nicht unmittelbar verändert werden. Das Package PRVT_CMT_CBK enthält jedoch Prozeduren, die die Einfügung neuer Datensätze in die Tabellen #2 und #3 ermöglichen. Die, in dem Package PRVT_CMT_CBK, enthaltenen Prozeduren werden nicht mit den Privilegien des ausführenden Benutzers sondern mit den Privilegien des Erstellers des Packages ausgeführt. Da dieser Schreibrecht für die Tabellen #2 und #3 besitzt, ist es jedem Benutzer durch Verwendung des Packages PRVT_CMT_CBK möglich, den Tabellen #2 und #3 neue Datensätze hinzuzufügen. Da wie bereits erwähnt jeder Benutzer die Ausführung des Delete-Triggers der Tabelle #1 bewirken kann und dieser, die in den Tabellen #2 und #3 gespeicherten Funktionen mit erhöhten Privilegien ausführt, kann jeder Benutzer eine Erweiterung seiner Zugriffsrechte erwirken. Es liegt dadurch eine Zugangsverschaffung iSd § 118a vor. Da diese nur durch das Hinzufügen jedoch nicht durch das Überschreiben von Datensätzen erfolgt liegt lediglich eine Überwindung jedoch keine Verletzung der Sicherheitsvorkehrung Least Privilege vor. Mangels Verletzung einer Sicherheitsvorkehrung ist der objektive Tatbestand des § 118a daher nicht erfüllt.

Auf Grund der Sicherheitslücke CVE-2004-0795[158] ist es einem lokalen Benutzer des Betriebssystems möglich, beliebige Befehle mit den Rechten des DB2-Administrators

[153] http://www.idefense.com/intelligence/vulnerabilities/display.php?id=136
[154] http://archives.neohapsis.com/archives/vulnwatch/2004-q4/0051.html
[155] Es handelt sich hierbei um Programmcode der in der Datenbank beim Löschen eines Datensatzes ausgeführt wird; vgl *Urman*, S. 457 ff
[156] eine Sammlung mehrerer Prozeduren; vgl *Urman*, S. 385 ff
[157] Der Übersichtlichkeit halber wurde auf die korrekte Benennung der Tabelle als SDO_TXN_IDX_INSERTS verzichtet.
[158] vgl http://marc.theaimsgroup.com/?l=bugtraq&m=107885081414173&w=2

auszuführen. Die Sicherheitslücke besteht darin, dass der DB2 Remote Command Server alle über eine Named Pipe[159] erhaltenen Befehle mit erweiterten Rechten ausführt und besagte Named Pipe jedem authentifizierten Benutzer einen Schreibzugriff ermöglicht. Hier ist bereits das Vorliegen einer Sicherheitsvorkehrung iSd § 118a zu verneinen.

6.2.4.3. Umgehung von Zugriffsbeschränkungen

Die ebenso Oracle betreffende Sicherheitslücke CVE-2004-1364 ermöglicht es auf Libraries zuzugreifen, die sich außerhalb des dafür vorgesehenen Verzeichnisses $ORACLE_HOME\bin befinden. Dies erfolgt durch relative Referenzierung darüber liegender Verzeichnisse mittels der Zeichenfolge „../“. Die Sicherheitslücke besteht darin, den Pfad der zu ladenden Library nicht hinreichend zu überprüfen. Gelingt es einem Angreifer sich derart erweiterten Zugang zu einem Teil des Computersystems zu verschaffen, kommt § 118a in Betracht. Eine Sicherheitsvorkehrung ist insofern gegeben, als dass grundsätzlich nur Libraries aus $ORACLE_HOME\bin geladen werden können. Mangels Beeinträchtigung der Datensubstanz dieser Sicherheitsvorkehrung, ist durch Ausnützung der Sicherheitslücke CVE-2004-1364 der objektive Tatbestand des § 118a jedoch nicht erfüllt.

6.2.4.4. Umgehung der Authentifizierung

CVE-2004-1365 beschreibt, dass Oracle beim Laden bestimmter Libraries keine Authentifizierung erfordert und so die Ausführung beliebiger Befehle im Sicherheitskontext des Oracle-Users ermöglicht.
Auf Grund der Sicherheitslücke CVE-2004-0627 ist es möglich den Authentifizierungsmechanismus eines MySQL-Servers zu umgehen. Der Client teilt dem Server im Rahmen einer Authentifizierungsanforderung das Passwort und dessen Länge mit. Die Sicherheitslücke besteht darin, dass bei einer Passwortlänge von 0, kein einziges Zeichen mit dem echten Passwort verglichen wird und so der Algorithmus zu dem Ergebnis gelangt, dass keine Unterschiede zwischen dem echtem und dem eingegebenen Passwort bestehen würden.

In beiden Fällen stellt der Authentifizierungsmechanismus eine Sicherheitsvorkehrung dar. Da diese jedoch nicht verletzt sonder bloß umgangen wird, ist der objektive Tatbestand des § 118a nicht erfüllt.

6.2.4.5. Umgehung der Autorisierung

Durch die Sicherheitslücke CVE-2005-0244 in PostgreSQL ist es möglich ohne Überprüfung der erforderlichen Berechtigungen beliebige Befehle im System abzusetzen. Mangels Verletzung einer Sicherheitsvorkehrung erfüllt eine solche Zugangsverschaffung zu einem Teil eines Computersystems nicht den Tatbestand des § 118a.

6.2.4.6. Informationspreisgabe

Die Sicherheitslücke CVE-2004-1366 besteht darin, dass Oracle das Passwort des Accounts SYSMAN unverschlüsselt in einer, von jedem Benutzer des Computersystems lesbaren Datei speichert. Durch Ausnützung dieser Sicherheitslücke kommt es zu einer Verwirklichung des objektiven Tatbestandes des § 126c Abs 1 Fall 2. Wird das so erlangte Passwort verwendet um sich Zugang zur Datenbank zu verschaffen, liegt eine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

[159] Ein Mittel der Kommunikation zwischen zwei Prozessen.

6.2.4.7. Denial of Service

Die Sicherheitslücke CVE-2004-1369 ermöglicht es durch das Senden entsprechend formatierter Anfragen, den Oracle TNS Listener zum Absturz zu bringen. Hierdurch sind keine weiteren Verbindungen zur Oracle-Datenbank über das Netzwerk mehr möglich. Da es sich um eine schwere Störung der Funktionsfähigkeit des Computersystems handelt, ist bei Fehlen der Verfügungsbefugnis der objektive Tatbestand des § 126b erfüllt.

6.2.4.8. SQL-Injection

Auf Grund von CVE-2004-1370 ist es jedem regulären Benutzer einer Oracle-Datenbank möglich beliebige SQL-Befehle mit erhöhten Privilegien auszuführen. Bestimmte, mit erweiterten Rechten ausgestattete Prozeduren ermöglichen eine SQL-Injection. Mangels Verletzung einer Sicherheitsvorkehrung (vgl 5.3. SQL-Injection) ist der objektive Tatbestand des § 118a nicht erfüllt.

6.2.4.9. Deaktivierung von Audit-Funktionen

Die Sicherheitslücke CVE-2005-1495 ermöglichte es einem authentifizierten Benutzer unter bestimmten Umständen Oracles Audit-Funktionen bezüglich einer Tabelle zu deaktivieren. Dies erlaubt einem Angreifer unerkannt zu bleiben und erhöht so die Wahrscheinlichkeit eines erfolgreichen Angriffs. Oracles Audit-Funktionen stellen zwar eine Sicherheitsvorkehrung iSd § 118a dar, ihre Deaktivierung entspricht mangels einer Beeinträchtigung der Datensubstanz jedoch keiner Verletzung. Eine Zugangsverschaffung unter Ausnützung der Sicherheitslücke CVE-2005-1495 erfüllt daher nicht den objektiven Tatbestand des § 118a.
Sind die, von den Audit-Funktionen generierten Daten tatsächlich für den Berechtigten von erheblichem Interesse, so kann deren Deaktivierung allenfalls eine schwere Störung der Funktionsfähigkeit eines Computersystems iSd § 126b darstellen.

6.2.5. File Sharing Applications

Peer to Peer (P2P) File Sharing Programme erfreuen sich zunehmender Beliebtheit. Wird Software aus P2P-Netzen bezogen, besteht ein erhöhtes Risiko, dass sie Viren oder trojanische Pferde enthält. Zu dieser Problematik vgl bereits 6.2.2.4. Exkurs: Malware – Verletzung einer Sicherheitsvorkehrung durch Social Engineering?

6.2.5.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2004-1114, CVE-2004-1286, CVE-2004-1892, CVE-2004-2433 und CVE-2005-0595 bestehen in Form eines möglichen Buffer Overflows. CVE-2005-1806 stellt einen Format String Bug dar. Alle genanten Sicherheitslücken ermöglichen durch das Überschreiben bestehender Speicherbereiche die Ausführung beliebigen Codes. Dies ist als Zugangsverschaffung durch Verletzung einer Sicherheitsvorkehrung iSd § 118a zu beurteilen.

6.2.6. DNS Software

Das DNS (Domain Name System) ist jener Teil des Internets, der die Übersetzung von Domainnamen in IP-Adressen und vice-versa ermöglicht. Die meisten Dienste des Internets (insbesondere E-Mail und Web) sind daher von DNS abhängig. Bei DNS handelt es sich um eine weltweite, dezentrale Datenbank, bestehend aus unzähligen DNS-Servern (auch Name-Server genannt)[160].

Größere Organisationen betreiben idR ihren eigenen Name-Server, der zwei wesentliche Funktionen erfüllt. Zum einen speichert er die Domainnamen der Organisation und ihre

[160] *Albitz/Liu,* S. 11 ff

Übersetzung in die entsprechenden IP-Adressen und hält diese für alle anderen Name-Server abrufbereit. Zum anderen wird er von der Organisation gleichsam als Einstiegspunkt in das Domain Name System verwendet – alle von der Organisation ausgehenden Übersetzungen eines Domainnamens in eine IP-Adresse werden daher über besagten Name-Server abgewickelt.

6.2.6.1. Cache Poisoning

Die Sicherheitslücken CVE-2005-0817 und CVE-2005-0877 ermöglichen ein sog. Cache Poisioning. Als DNS-Caching[161] wird die zeitlich begrenzte Zwischenspeicherung von einmal abgefragten Informationen anderer Name-Server bezeichnet. Jeder Name-Server betreibt Caching um eine höhere Leistungsfähigkeit zu erreichen und die Anfragelast im weltweiten Domain Name System zu reduzieren. Unter Cache Poisioning versteht man die nicht autorisierte Einflussnahme auf die zwischengespeicherten DNS-Daten anderer Server. Meist erfolgt dies durch das unaufgeforderte Senden gefälschter DNS-Responses. Obgleich die so erhaltenen Daten niemals angefordert wurden, werden sie oft dennoch in den Cache aufgenommen. Dies führt dazu, dass es einem entfernten Angreifer möglich ist, den Cache derart zu manipulieren, dass alle Anfragen nach der Auflösung eines bestimmten Domainnamens (zB www.amazon.com) mit eine IP-Adresse seiner Wahl beantwortet werden.

Zunächst ist eine Strafbarkeit des Cache Poisioning selbst zu untersuchen. Wegen der mangelnden Möglichkeit, sich von im Cache befindlichen Daten Kenntnis zu verschaffen, liegt keine Zugangsverschaffung iSd § 118a vor.
Da auch kein vermögenswertes Interesse an den, konkret im Cache befindlichen Daten besteht, ist die Erfüllung des Tatbestandes des § 126a ebenso zu verneinen. Ein durch das Cache Poisioning entstehender mittelbarer Schaden ist nicht Gegenstand des § 126a.[162]

Zweck eines Cache Poisoning ist meist die Durchführung einer Man-in-the-middle-Attack. Das Wesen eines solchen Angriffs besteht darin, dass sich der Angreifer gleichsam zwischen Client und Server positioniert und sich der Client, ohne dies zu wissen, nicht mit dem intendierten Server sondern mit dem Server des Angreifers verbindet. Die vom Client erhaltenen Daten protokolliert der „Man-in-the-middle“ und sendet sie an den Server weiter. Von diesem kommende Daten werden ebenso protokolliert und spiegelbildlich an den Client weitergeleitet.
Für eine Strafbarkeit nach § 119 oder § 119a ist jedenfalls zu prüfen, ob die vorliegende Konstruktion der Benützung einer Vorrichtung, die an dem Computersystem angebracht oder sonst empfangsbereit gemacht wurde, entspricht. Da der Begriff der Vorrichtung jedes technisches Hilfsmittel umfasst[163], handelt es sich bei dem, vom Täter als „Man-in-the-middle“ eingesetzten Server um eine Vorrichtung iSd §§ 119 f. Die Durchführung des Cache Poisoning entspricht der Tathandlung des „sonst empfangsbereit machen“.
Da der Täter die Daten am Weg vom Client zum Server abfängt, befinden sich die Daten zum Tatzeitpunkt am Übertragungsweg. Daher kommt bei Daten, die eine Gedankenerklärung darstellen (zB idR E-Mails) eine Strafbarkeit nach § 119 und bei allen anderen Daten – entsprechender erweiterter Vorsatz vorausgesetzt – eine Strafbarkeit nach § 119a in Betracht.
Verschafft sich der Angreifer Authentifizierungsdaten, ist der objektive Tatbestand des § 126c Abs 1 Fall 2 erfüllt. Werden diese in weiterer Folge zu einer Zugangsverschaffung zu einem Computersystem (od. dem Teil eines solchen) verwendet, liegt eine Verletzung der Sicherheitsvorkehrung des Authentifizierungsmechanismus vor, weshalb der objektive Tatbestand des § 118a erfüllt sein wird.

---

[161] *Albitz/Liu,* S. 34 ff
[162] vgl *Bertel* in WK, § 126a Rz 5
[163] vgl *Lewisch* in WK, § 119 Rz 4

Eine weitere Möglichkeit durch Cache Poisoning an vertrauliche Informationen zu gelangen, besteht darin nicht als „Man-in-the-middle“ zwischen Client und Server zu fungieren, sondern dem Client unmittelbar den erwarteten Dienst anzubieten. In diesen Fällen stellt der Server des Täters daher den Endpunkt der Kommunikation dar. Fraglich ist jedoch ob die §§ 119 f auch für diese Fälle gelten, in denen die Nachrichten bzw. die Daten nicht „mitgelesen“ sondern „umgeleitet“ werden. Da sich die Daten noch am Übertragungsweg befinden, würde der Wortlaut der §§ 119 f derartige Fälle sehr wohl erfassen. Hierfür spricht auch die Überschrift zu CyCC Art 3 „Illegal *interception*“ bzw. die Überschrift zu § 119a „Missbräuchliches *Abfangen* von Daten“. Denn das Abfangen von Daten erfasst auch eine Umleitung der Daten. Daher kommt mE auch in diesen Fallkonstellationen eine Strafbarkeit nach den §§ 119 f in Betracht. Wer die Anwendbarkeit der §§ 119 f in diesen Fällen verneint, könnte allenfalls über das subsidiäre Delikt des § 120 Abs 2a zu einer Strafbarkeit gelangen.[164] § 120 Abs 2a schützt jedoch nur im Wege einer Telekommunikation übertragenen Nachrichten. Der objektive Tatbestand erfordert im Unterschied zu §§ 119 f nicht bloß die Verwendung einer Abhörvorrichtung sondern das Aufzeichnen, Zugänglichmachen oder Veröffentlichen der Nachricht. Ebenso wie nach § 119 ist als erweiterter Vorsatz eine Spionageabsicht erforderlich.

6.2.7. Media Players

Zur Ausnützung der hier beschriebenen Sicherheitslücken ist es erforderlich, dass das Opfer eine Datei in einem Media Player öffnet. Dies wird dadurch erleichtert, dass die meisten Browser den entsprechenden Media Player automatisch öffnen um eine Medien-Datei abspielen zu können.

6.2.7.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2004-0550, CVE-2004-1094, CVE-2004-1481, CVE-2005-0189, CVE-2005-0191, CVE-2005-0455, CVE-2005-0611, CVE-2005-0755, CVE-2005-1766, CVE-2005-2052, CVE-2005-2710, CVE-2005-0043, CVE-2005-1248, CVE-2004-1119, CVE-2004-1150, CVE-2004-1896, CVE-2005-2310, CVE-2004-0431, CVE-2004-0926, CVE-2005-2753, CVE-2005-2754 und CVE-2004-1244[165] ermöglichen die Ausführung beliebiger Befehle durch Ausnützung von Buffer Overflows bzw. Format String Bugs. Dies ist als Zugangsverschaffung durch Verletzung einer Sicherheitsvorkehrung iSd § 118a zu beurteilen.

6.2.7.2. Fehlende Zugriffsbeschränkungen

Auf Grund der Sicherheitslücke CVE-2005-2743 ist es auch nicht vertrauenswürdigen Java Applets möglich Funktionen der System-Bibliotheken auszuführen[166]. Da es die Entwickler verabsäumt haben, diese Funktionalität auf vertrauenswürdige Applets zu beschränken, liegt keine Sicherheitsvorkehrung iSd § 118a vor.

Die Sicherheitslücke CVE-2004-1324 besteht darin, dass es einem Angreifer unter Verwendung des Microsoft Windows Media Player 9.0 ActiveX Control möglich ist die Künstler- und Titelinformationen einer Musikdatei zu verändern. Dies kann nur dann zu einer Zugangsverschaffung führen, wenn eine weitere Sicherheitslücke besteht, die dazu führt, dass die Künstler- und Titelinformationen als Script-Code ausgeführt werden. Die Ausnützung von CVE-2004-1324 selbst erfüllt somit nicht den Tatbestand des § 118a.

---

[164] vlg. hierzu *Reindl* in WK, § 120 Rz 31a ff

[165] Diese Sicherheitslücke wurde auch unter *6.1.3. Windows Libraries* behandelt.

[166] vgl http://lists.apple.com/archives/security-announce/2005/Sep/msg00002.html

6.2.7.3. Unzureichende Überprüfung von Eingabedaten

Die Sicherheitslücke CVE-2005-2628 betrifft Macromedia Flash 6 und 7. Durch eine entsprechend konstruierte SWF-Datei ist es möglich beliebigen Code zur Ausführung zu bringen. Die Sicherheitslücke besteht darin, dass der Index für ein Array, in dem Funktions-Pointer gespeichert werden, aus der SWF-Datei ohne Überprüfung ausgelesen wird[167]. Enthält die Datei einen Index, der größer als das Array ist, wird der Pointer einer auszuführenden Funktion aus anderen Speicherbereichen des Heaps ausgelesen. Hierdurch ist es dem Angreifer möglich eigenen Maschinencode zur Ausführung zu bringen. Da der geschilderte Vorgang nicht zu einer Verletzung der Datensubstanz des Programms führt, wird die Sicherheitsvorkehrung Least Privilege nicht verletzt. Eine Strafbarkeit nach § 118a ist daher zu verneinen.

6.2.7.4. Umgehung von Sicherheitsvorkehrungen

Die Sicherheitslücke CVE-2004-0820 ermöglicht es einem Angreifer beliebige Befehle in der Sicherheitszone „Local Machine“ auszuführen, wenn der Benutzer einen Winamp-Skin des Angreifers öffnet. Bei einem sog. „Skin“ des Medienplayers Winamp handelt es sich um eine Datei, die die graphische Benutzeroberfläche des Medienplayers verändert. Grundsätzlich liegt hier eine Sicherheitsvorkehrung in Form der Implementierung des Prinzips Least Privilege vor, da ein Skin lediglich das Erscheinungsbild der Anwendung verändern, jedoch keine Befehle ausführen kann. Die Sicherheitslücke besteht nur darin, dass ein Skin das Laden einer bestimmten URL im Browser-Fenster des Medienplayers bewirken kann. Im Browser-Fenster geladener Script-Code wird dann in der Sicherheitszone „Local Machine“ ausgeführt. Da hiermit weit reichende Rechte verbunden sind liegt eine Zugangsverschaffung iSd § 118a vor. Da diese jedoch ohne Verletzung der Sicherheitsvorkehrung Least Privilege erfolgt ist eine Strafbarkeit nach § 118a zu verneinen.

6.2.7.5. Sicherheitslücken unbekannter Beschaffenheit

Die Sicherheitslücken CVE-2005-2054 und CVE-2005-2055 ermöglichen eine Zugangsverschaffung zu einem Computersystem. Es handelt sich hierbei um Sicherheitslücken der proprietären Software RealPlayer. Da der Hersteller keine näheren Informationen über die Beschaffenheit der Sicherheitslücken veröffentlicht hat[168], kann nicht festgestellt werden, ob die Ausnützung dieser Sicherheitslücken eine Verletzung iSd § 118a darstellt.
Es handelt sich hierbei um das allgemeine Problem, dass im Strafverfahren nicht nur die Zugangsverschaffung und das Vorliegen einer Sicherheitsvorkehrung sondern auch das Tatbestandsmerkmal der Verletzung bewiesen werden muss. Ist nicht bekannt welche Sicherheitslücke für die Zugangsverschaffung verwendet wurde bzw. ob diese eine Verletzung einer Sicherheitsvorkehrung darstellt, hat – in dubeo pro reo – ein Freispruch zu erfolgen.

6.2.8. Instant Messaging Applications

6.2.8.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2004-0597, CVE-2004-0636, CVE-2005-0562, CVE-2005-3265 und CVE-2005-3267 ermöglichen es einem Angreifer beliebige Befehle unter Ausnützung eines Buffer Overflows auszuführen. Dies stellt eine Zugangsverschaffung durch Verletzung einer Sicherheitsvorkehrung iSd § 118a dar.

[167] vgl http://www.securityfocus.com/archive/1/archive/1/415789/30/0/threaded
[168] vgl http://service.real.com/help/faq/security/050623_player/EN/

6.2.8.2. Zugangsverschaffung durch Täuschung des Benutzers

Die Sicherheitslücke CVE-2005-0243 besteht darin, dass Yahoo! Messenger lange Dateinamen nicht zur Gänze anzeigt[169]. Dies kann dazu führen, dass der Benutzer über den Dateityp getäuscht wird, und deshalb die Datei öffnet. Eine Sicherheitsvorkehrung kann darin erblickt werden, dass dem Benutzer bekannt sein soll welche Datei er öffnet. Die Datensubstanz dieser Sicherheitsvorkehrung wird jedoch durch die Verwendung eines zu langen Dateinamens nicht verletzt. Daher ist der objektive Tatbestand des § 118a nicht erfüllt. Zur nahe liegenden Problematik des Social Engineering vlg. bereits 6.2.2.4. Exkurs: Malware – Verletzung einer Sicherheitsvorkehrung durch Social Engineering?.

6.2.9. Mozilla and Firefox Browsers

Da Firefox auf den Codequellen von Mozilla basiert, betreffen manche Sicherheitslücken beide Browser gleichermaßen.

6.2.9.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2005-0592, CVE-2005-2701, CVE-2005-2705 und CVE-2005-2871 ermöglichen es, durch Ausnützung eines Buffer Overflows beliebige Befehle auszuführen. Dies erfüllt die Tatbestandsmerkmale der Zugangsverschaffung durch Verletzung einer Sicherheitsvorkehrung iSd § 118a.

6.2.9.2. Command-Injection

Die Sicherheitslücke CVE-2005-2968 betrifft sowohl Firefox als auch Mozilla. Beide Browser werden mit einem Shell-Script[170] ausgeliefert, das es anderen Applikationen erleichtern soll eine URL im gewünschten Browser zu öffnen. Das Shell-Script ist jedoch derart fehlerhaft, dass die, als Argument übergebene URL uU von der Shell interpretiert wird[171]. Daher ist eine Command-Injection möglich. Besagtes Shell-Script implementiert zwar die Sicherheitsvorkehrung Least Privilege, durch die Command-Injection wird diese jedoch – mangels Beeinträchtigung der Datensubstanz des Scripts – nicht verletzt. Es besteht daher keine Strafbarkeit nach § 118a.

6.2.9.3. Ausführen von „Chrome JavaScript“

Mozilla und Firefox unterscheiden in „Web JavaScript“ und in „Chrome JavaScript“. Während ersteres nur die von JavaScript bekannten, beschränkten Funktionalitäten aufweist, ist Chrome JavaScript mit allen Rechten und Funktionalitäten des Browsers ausgestattet. Chrome Javascript ist daher insbesondere in der Lage beliebige Dateioperationen vorzunehmen und Netzwerkverbindungen herzustellen.[172] Gelingt es einem Angreifer Code als Chrome JavaScript zur Ausführung zu bringen, stellt dies eine Zugangsverschaffung iSd § 118a dar. Die Tatsache, dass eine Website grundsätzlich kein Chrome Javascript ausführen kann, stellt eine Sicherheitsvorkehrung iSd Prinzips Least Privilege dar.

Die Sicherheitslücken CVE-2005-0752, CVE-2005-1155, CVE-2005-1158, CVE-2005-1531, CVE-2005-2262, CVE-2005-2267, CVE-2005-1156 iVm CVE-2005-1157 und CVE-2005-1476 iVm CVE-2005-1477 bestehen darin, dass Firefox in bestimmten Fällen anstatt einer URL auch eine „JavaScript-URL“ in der Form von „javascript:<code>“ öffnet. Der derart

[169] vlg. http://secunia.com/secunia_research/2005-2/advisory/

[170] Als „Shell“ bezeichnet man den Kommandozeileninterpreter eines UNIX-Systems. Eine ausführbare Datei, die, für die Shell interpretierbare Befehle enthält, bezeichnet man daher als „Shell-Script“. Vgl *Newham/Rosenblatt*, S. 83 ff

[171] vlg http://www.mozilla.org/security/announce/mfsa2005-59.html

[172] vgl http://gnusto.mozdev.org/code/component-vs-chrome.html

ausgeführte Code läuft in diesen Fällen als Chrome JavaScript. Da es jedoch hierbei zu keiner Verletzung einer Sicherheitsvorkehrung kommt, ist der Tatbestand des § 118a nicht erfüllt.

CVE-2005-1160 ermöglicht es Chrome JavaScript durch das Überschreiben bestimmter Eigenschaften und Methoden von DOM[173]-Objekten zur Ausführung zu bringen. Die Sicherheitslücke besteht darin, dass bereits als Chrome JavaScript laufender Code die Methoden von DOM-Objekten mit den Rechten von Chrome JavaScript aufruft ohne die Integrität der Methoden zu überprüfen[174]. Da das DOM als Teil jener Sicherheitsvorkehrung anzusehen ist, die den Aufruf von Chrome JavaScript grundsätzlich verhindert, stellt das Überschreiben bestehender DOM-Methoden eine Verletzung einer Sicherheitsvorkehrung iSd § 118a dar.

CVE-2005-2269 weist Ähnlichkeiten mit der gerade erörterten Sicherheitslücke CVE-2005-1160 auf. Der Unterschied besteht jedoch darin, dass Methoden bestehender DOM-Objekte nicht verändert werden müssen. Für die Ausbeutung von CVE-2005-2269 ist es ausreichend ein neues Objekt zu erzeugen, das vom Browser als reguläres HTML-Element behandelt wird.[175] Werden die vermeintlich nativen Methoden als Chrome JavaScript ausgeführt, gelangt tatsächlich der Code des Angreifers zur Ausführung. Da es bei dieser Vorgehensweise nicht erforderlich ist Teile des DOM zu überschreiben, liegt keine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

Die Sicherheitslücke CVE-2005-2270 besteht darin, dass Firefox und Mozilla beim Klonen eines JavaScript-Objekts Basisobjekte derart fehlerhaft klonen, dass es durch das neu entstandene Objekt möglich ist, auf zuvor nicht zugängliche Funktionalitäten der Basisobjekte zuzugreifen. Dies erfolgt durch ein sog. Hinaufwandern der Prototype Chain[176] bis zu einem, mit „Chrome-Rechten“ ausgestatteten Objekt. Da es dadurch zu keiner Beeinträchtigung einer Datensubstanz kommt, liegt keine Verletzung einer Sicherheitsvorkehrung iSd § 118a vor.

CVE-2005-2706 ermöglicht es Seiten mit „Chrome-Rechten“ zu laden. In Verbindung mit anderen Sicherheitslücken, die eine Überwindung der „Same Origin Policy“[177] ermöglichen, kann es so zur Ausführung von Chrome JavaScript kommen. Da es bei der Ausnützung von CVE-2005-2706 zu keiner Verletzung einer Sicherheitsvorkehrung kommt[178], ist die Erfüllung des Tatbestandes des § 118a von der Beschaffenheit der zweiten Sicherheitslücke abhängig.

6.2.9.4. Webpage Spoofing

Die Sicherheitslücken CVE-2005-1937, CVE-2005-2268, CVE-2005-2602, CVE-2005-2707 und CVE-2005-0593 ermöglichen es einem Angreifer auf unterschiedliche Weise einem Benutzer den Eindruck zu vermitteln, dass er sich auf einer von ihm angewählten, vertrauenswürdigen Website befindet, obgleich es sich tatsächlich um die Website des Angreifers handelt. Dies kann dazu führen, dass der Benutzer der Website des Angreifers und damit dem Angreifer selbst Informationen (zB Authentifizierungsdaten) preisgibt. Bezüglich der Anwendbarkeit der §§ 119 f und einer Strafbarkeit nach §§ 126c und 118a vlg. 6.2.6.1. Cache Poisoning.

---

[173] Document Object Model; vgl http://www.w3.org/DOM/
[174] vgl http://www.mozilla.org/security/announce/mfsa2005-41.html
[175] vgl http://www.mozilla.org/security/announce/mfsa2005-55.html
[176] JavaScript implementiert das Konzept der Vererbung als Prototype Inheritance; vgl *Flanagan*, S. 133 ff
[177] vgl http://www.mozilla.org/projects/security/components/same-origin.html
[178] vgl http://www.mozilla.org/security/announce/mfsa2005-58.html#about

6.2.10. Other Cross-platform Applications

Sicherheitslücken von Cross-Platform Applications, die keiner der bisher behandelten Kategorien entsprechen, werden an dieser Stelle behandelt. Die Nennung der Sicherheitslücke CVE-2005-0583 in dieser Kategorie ist als Versehen der Autoren der SANS Top 20 zu beurteilen, da CVE-2005-0583 bereits unter „C1. Backup Software“ (vgl *6.2.1. Backup Software*) behandelt wird.

6.2.10.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2005-0581, CVE-2005-0582, CVE-2005-2551, CVE-2005-1543, CVE-2005-2668, CVE-2005-1825, CVE-2005-1826, CVE-2005-3252 und CVE-2005-1471 ermöglichen es durch Ausnützung eines Buffer Overflows beliebige Befehle auszuführen. Dies stellt eine Zugangsverschaffung durch Verletzung einer Sicherheitsvorkehrung iSd § 118a dar.

6.2.10.2. Argument-Injection

Die Sicherheitslücken CVE-2005-0418 und CVE-2005-0836 ermöglichen es, dass die Java Virtual Machine (JVM) bei der Ausführung eines Applets unter Deaktivierung der Sandbox-Funktionalität gestartet wird. Als „Sandbox“ wird jenes Sicherheitskonzept der JVM bezeichnet, mit dem der darin laufende Code auf ungefährliche Operationen beschränkt wird. So sind Zugriffe auf das Dateisystem aus der Sandbox nicht möglich.[179] Die Sicherheitslücken bestehen darin, dass bestimmte, in einer JNLP[180]-Datei angegebenen Argumente für den Aufruf der JVM nicht auf ihren Inhalt geprüft werden.[181] Ähnlich der Command-Injection kommt es hierbei zu keiner Beeinträchtigung der Datensubstanz der Sicherheitsvorkehrung. Daher ist der objektive Tatbestand des § 118a nicht erfüllt.

6.2.10.3. Umgehung von Zugriffsbeschränkungen

Durch die Sicherheitslücke CVE-2004-1029 ist es einem Java-Applet möglich aus der Sandbox der JVM „auszubrechen“. Die JVM verwendet intern einige private Java-Klassen, die einem Applet grundsätzlich nicht zugänglich sind[182] – dies stellt eine Sicherheitsvorkehrung iSd § 118a dar. Eine fehlende Zugriffsprüfung ermöglicht es jedoch JavaScript über den Datenaustausch mit Java Zugriff auf diese Klassen zu erhalten. Dadurch kann die Sandbox deaktiviert werden, wodurch das Applet keinen Beschränkungen mehr unterworfen ist. Hierdurch kann die vorliegende Sicherheitsvorkehrung umgangen werden. Mangels Verletzung ist der Tatbestand des § 118a nicht erfüllt.

Die Sicherheitslücke CVE-2005-1974 ermöglicht es einem Java-Applet seine eigenen Privilegien zu erhöhen. Da Sun keine näheren Details über die Beschaffenheit der Sicherheitslücke veröffentlicht hat, kann nur vermutet werden dass ihre Ausbeutung keine Verletzung einer Sicherheitsvorkehrung iSd § 118a darstellt. Vgl hierzu bereits 6.2.7.5. Sicherheitslücken unbekannter Beschaffenheit.

6.3. Top Vulnerabilities in UNIX Systems

Unter „Unix“ werden hier alle bekannten Unix-Derivate wie Solaris/SunOS, AIX, HP-UX, IRIX, SCO Unix, FreeBSD, OpenBSD, NetBSD und Linux verstanden[183].

[179] Eine ausführliche Darstellung des Java Sandbox-Designs findet sich in *Graff/Van Wyk*, 51 ff
[180] Java Network Launching Protocol; vgl http://java.sun.com/products/javawebstart/download-spec.html
[181] vgl http://marc.theaimsgroup.com/?l=full-disclosure&m=111117284323657&w=2
[182] vgl http://www.idefense.com/intelligence/vulnerabilities/display.php?id=158
[183] vgl zur Entwicklung der verschiedenen Unix-Derivate *Frisch* S. xii ff und *Spafford*, UNIX and Security: The Influences of History

6.3.1. UNIX Configuration Weaknesses

In dieser Kategorie sind keine Sicherheitslücken mit CVE-Nummern enthalten. Das dominierende Thema im Zusammenhang mit unzulänglichen Konfigurationen ist die SSH-Authentifizierung.

6.3.1.1. SSH-Authentifizierung

Die Secure Shell (SSH) ist ein, auf einem Server laufender Dienst, der einen entfernten Zugriff auf den Server ermöglicht. SSH hat sich als defacto-Standard etabliert, da die gesamte Kommunikation zwischen Client und Server mittels asymmetrischer und symmetrischer Kryptographieverfahren verschlüsselt wird[184].
Meist ist eine Authentifizierung mittels Benutzername und Passwort möglich. Aus diesem Grund stehen Brute Force Attacks auf der Tagesordnung[185]. Zur Strafbarkeit von Brute Force Attacks vgl bereits *5.6.1. Schwache Passwörter*.

6.3.2. Mac OS X

Mac OS X ist ein kommerzielles UNIX-Betriebssystem des Unternehmens Apple. Die in dieser Kategorie genannte Sicherheitslücke CVE-2005-0418 wurde bereits unter 6.2.10.2. Argument-Injection erörtert.

6.3.2.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2005-0126, CVE-2005-1721, CVE-2005-2501, CVE-2005-2502, CVE-2005-2507 und CVE-2005-2518 ermöglichen die Ausführung beliebigen Codes durch die Ausnützung eines Buffer Overflows. Dies ist als Zugangsverschaffung unter Verletzung einer Sicherheitsvorkehrung iSd § 118a zu beurteilen.

6.3.2.2. Command-Injection

Die Sicherheitslücke CVE-2005-1342 ermöglichte es dem Täter beliebige Befehle auszuführen. Er muss hierzu lediglich das Opfer dazu bewegen einen Link anzuklicken, der mit der Zeichenkette „x-man-page:“ beginnt. Dadurch wird ein virtueller Terminal[186] aufgerufen, der es unterlässt die übergebene URL vor dem Öffnen zu überprüfen. Durch die Verwendung bestimmte Sonderzeichen (sog. Escape Characters) ist es dem Angreifer möglich Befehle im virtuellen Terminal abzusetzen. Da die Sicherheitsvorkehrung mittels „x-man-page:“ beginnenden URLs keine beliebigen Befehle ausführen zu können nicht in ihrer Datensubstanz beeinträchtigt wird, liegt keine Verletzung iSd § 118a vor.

6.3.2.3. Safari Browser

Mac OS X wird mit dem Webbrowser Safari ausgeliefert. Dieser weist einige Sicherheitslücken auf.

Auf Grund der Sicherheitslücke CVE-2005-1474 ist es möglich sog. Dashboard Widgets automatisch und ohne Zustimmung des Opfers auf dessen Computersystem herunter zuladen und zu installieren.[187] Bei einem Dashboard Widget handelt es sich um kleine, auf dem Desktop laufende Anwendungen, die in HTML, JavaScript und CSS programmiert werden. Da ein Dashboard Widget grundsätzlich mit allen Rechten des angemeldeten Benutzers ausgestattet ist, ermöglicht die Ausnützung der Sicherheitslücke CVE-2005-1474 eine

[184] *Barrett/Silverman* S. 45 ff
[185] vlg. http://searchsecurity.techtarget.com/originalContent/0,289142,sid14_gci1094140,00.html
[186] vergleichbar mit einem Fenster der DOS-Eingabeaufforderung
[187] vgl http://www.securityfocus.com/bid/13694

Zugangsverschaffung iSd § 118a. CVE-2005-1474 besteht darin, dass Safari Dashboard Widgets als unbedenkliche Inhalte behandelt und daher die Installation zulässt. Eine Sicherheitsvorkehrung ist dadurch gegeben, dass Safari die automatische Installation von Anwendungen grundsätzlich verhindert. Da im Fall der Installation von Dashboard Widgets diese Sicherheitsvorkehrung in ihrer Datensubstanz jedoch nicht beeinträchtigt wird, liegt keine Verletzung iSd § 118a vor.

Die Sicherheitslücken CVE-2005-2516 und CVE-2005-2522 ermöglichen es in RTF- bzw. PDF-Dateien Sicherheitsvorkehrungen zum Schutz vor bestimmten potentiell gefährlichen URLs zum umgehen. Da keine Verletzung einer Sicherheitsvorkehrung vorliegt besteht keine Strafbarkeit nach § 118a.

CVE-2005-2517 führt zu einer Informationspreisgabe. In bestimmten Fällen sendet Safari ein bereits ausgefülltes HTML-Formular an die nächste angewählte Seite. Wurde beispielsweise http://www.example.com/formular.php im Browser aufgerufen und das dadurch dargestellte Formular ausgefüllt, so werden beim Aufruf der URL http://www.example.net alle Formulardaten mitgesandt. Da der Betreiber der zweiten Website (http://www.example.net) jedenfalls keine Vorrichtung iSd §§ 119 f benützt, scheidet eine Strafbarkeit nach diesen Bestimmungen aus. Eine Strafbarkeit könnte sich allenfalls aus § 120 Abs 2a ergeben. Voraussetzung ist hierfür, dass es sich bei den Daten um Nachrichten, d.h. Gedankenerklärungen handelt und, dass diese vom Betreiber der zweiten Website aufgezeichnet, einem anderen Unbefugten zugänglich gemacht oder veröffentlicht werden. Darüber hinaus ist zum Tatzeitpunkt eine Spionageabsicht als erweiterter Vorsatz erforderlich.

6.3.2.4. Erstellung von Programmen mit SUID bzw. SGID

Unter Unix ist jede Datei mit bestimmten Zugriffsrechten versehen, wobei sie stets einem bestimmten Benutzer („Owner“) und einer bestimmten Gruppe („Group“) zugewiesen ist.[188] Da jedes Programm in Form einer Datei vorliegt, können Berechtigungen bezüglich des Programms auf Dateiebene definiert werden. Wird ein Programm ausgeführt, so läuft dieses grundsätzlich jedoch mit den Rechten jenes Benutzers, der es gestartet hat. Zwei Ausnahmen dieses Prinzips stellen SUID (Set User-ID) und SGID (Set Group-ID) dar.[189] Wird das SUID-Bit einer Datei gesetzt, kann diese mit den Berechtigungen des Owners anstatt des ausführenden Benutzers gestartet werden. Wurde das SGID-Bit gesetzt, kann die Datei mit den Berechtigungen der Group gestartet werden, der die Datei zugewiesen ist. Ist der administrative Account „root“ Owner einer ausführbaren Datei, deren SUID-Bit gesetzt ist, so kann die Datei auch durch andere Benutzer derart gestartet werden, dass ihr die umfassenden Rechte des root-Accounts zukommen.[190] Dies kann dann zu Problemen führen, wenn die ausgeführte Datei zB eine Command-Injection ermöglicht, wodurch ein normaler Benutzer beliebige Befehle mit umfassenden administrativen Rechten ausführen könnte.
Aus diesem Grund ist bei keinem der mit Mac OS X ausgelieferten Programme das SUID- oder SGID-Bit gesetzt. Die Sicherheitslücke CVE-2005-0970 besteht darin, dass es einem lokalen Benutzer dennoch möglich ist das SUID- oder SGID-Bits von bestehenden Dateien zu setzen[191]. Da der Benutzer hierzu jedoch Schreibrechte für die betreffende Datei haben müsste, besteht keine Möglichkeit unmittelbar durch die Ausnutzung von CVE-2005-0970

[188] *Frisch*, S. 27 ff
[189] Das SUID- und SGID-Bits entsprechen der oktalen Zahlen 4000 bzw. 2000 und sind in der Ausgabe des Befehls „*ls -l*“ durch den Buchstaben „s“ an der Stelle des Execute-Bits erkennbar. Vgl *Garfinkel/Spafford/Schwartz*, S. 140, 145 ff
[190] Vorausgesetzt die Benutzer haben die Berechtigung die Datei auszuführen.
[191] vgl http://lists.apple.com/archives/security-announce/2005/Apr/msg00000.html

eine Eskalation von Privilegien zu erwirken. CVE-2005-0970 eröffnet lediglich die Möglichkeit andere Sicherheitslücken wie eine mögliche Command-Injection wirkungsvoller auszunutzen.

6.3.2.5. Zugangsverschaffung durch Täuschung des Benutzers

Die Sicherheitslücke CVE-2005-1331 besteht darin, dass der, in Mac OS X enthaltene AppleScript Editor Script-Code in bestimmten Fällen nicht zutreffend anzeigt. Dies ermöglicht insofern eine Täuschung, als dass ein Script-Code, der einem Benutzer zugesandt wird, bei Überprüfung im AppleScript Editor als ungefährlich erscheinen wird, da der angezeigte Code nicht dem tatsächlichen Code entspricht[192]. Kommt es zu einer Ausführung des Codes, liegt eine Zugangsverschaffung iSd § 118a vor. Eine Sicherheitsvorkehrung ist dadurch gegeben, dass der AppleScript-Code vor der Ausführung überprüft werden kann. Da diese in ihrer Datensubstanz durch Ausnützung von CVE-2005-1331 nicht beeinträchtigt wird, liegt keine Verletzung iSd § 118a vor, weshalb eine Strafbarkeit zu verneinen ist.

6.3.2.6. Umgehung sonstiger Zugriffsbeschränkungen

Durch CVE-2005-1337 ist es einem Angreifer möglich über einen mit „help://“ beginnenden Link beliebige Befehle auszuführen. Die Sicherheitslücke befindet sich im Programm Apple Help Viewer. Dieser wird beim Klicken auf den mit „help://“ beginnenden Link geöffnet, und führt auch übergebenen JavaScript-Code mit erweiterten Privilegien aus. Die allgemeine Beschränkung von JavaScript-Code auf ungefährliche Operationen kann hierdurch umgangen werden. Mangels Beeinträchtigung der Datensubstanz der Sicherheitsvorkehrung liegt keine Verletzung iSd § 118a vor.

6.4. Top Vulnerabilities in Networking Products

6.4.1. Cisco IOS and non-IOS Products

Cisco bringt für die meisten seiner Router[193] und Switches[194] das Betriebsystem IOS (Internetwork Operating System) zum Einsatz. Es handelt sich hierbei um kein UNIX-Derivat sondern um ein Betriebssystem eigener Art. Nahezu 85% der Router und Switches, die Teil des globalen Internet Backbones sind verwenden IOS. Aus diesem Grund stellen Sicherheitslücke in IOS eine der größten Gefahren für die globale Infrastruktur des Internet dar.[195]

6.4.1.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücken CVE-2005-2451, CVE-2005-2841 und CVE-2005-2244 ermöglichen es einem Angreifer unter Ausnützung eines Buffer Overflows beliebige Befehle auszuführen. Die beiden erstgenannten Sicherheitslücken betreffen IOS, CVE-2005-2244 hingegen Cisco CallManager[196]. Da es hierdurch zu einer Verletzung einer Sicherheitsvorkehrung kommt, besteht – unter den sonstigen Voraussetzungen – eine Strafbarkeit nach § 118a.

---

[192] vgl http://lists.apple.com/archives/security-announce/2005/May/msg00001.html

[193] Ein Hardwaregerät, das die Kommunikation zwischen zwei Netzwerken ermöglicht; vgl *Held*, S. 365 ff

[194] Ein Hardwaregerät, das die Kommunikation mehrerer Computersysteme innerhalb eines lokalen Netzwerkes ermöglicht. Bezüglich des Unterschieds zwischen Hubs und Switches (auch „Switching Hubs“) vgl *Held*, S. 312 ff

[195] Am 27. Juli 2005 präsentierte Michael Lynn auf der BlackHat Conference 2005 Möglichkeiten der Ausbeutung von Buffer Overflows in IOS. Vgl hierzu @Risk: The Consensus Security Alert - Volume 4, Issue #30, http://www.sans.org/newsletters/risk/display.php?v=4&i=30#exploit2. Cisco hatte bereits im Vorfeld der Konferenz versucht den Vortrag zu unterbinden. In weiterer Folge reichte Cisco Klage gegen Lynn ein, die mit einer außergerichtlichen Einigung beigelegt wurde. Lynns Präsentation und eine interessante Diskussion zum Thema finden sich unter http://archives.neohapsis.com/archives/fulldisclosure/2005-07/0663.html.

[196] vgl http://www.cisco.com/en/US/products/sw/voicesw/ps556/

6.4.1.2. Backdoor Accounts und Default-Passwörter

Die Sicherheitslücke CVE-2004-0391 besteht darin, dass in bestimmten Produkten Ciscos ein nicht dokumentierter Account mit einem unveränderlichen Passwort existiert. CVE-2005-0612 beschreibt, dass Cisco Unity[197] für neu erstellte Accounts in bestimmten Fällen Default-Passwörter verwendet.[198] Die Sicherheitslücke CVE-2005-0612 ist durch einen unveränderlichen, stets gleichen SNMP[199] Community String gegeben.

Für eine Strafbarkeit nach § 118a ist zunächst das Vorliegen einer Sicherheitsvorkehrung zu prüfen. Da ein Authentifizierungsmechanismus, dessen Passwörter anzunehmender Weise im Internet veröffentlicht wurden im Verhältnis zu gar keinem Authentifizierungsmechanismus die Wahrscheinlichkeit der Beeinträchtigung der Vertraulichkeit, Integrität oder Verfügbarkeit von Daten und Systemen verringert, liegt eine Sicherheitsvorkehrung vor.
Weiters ist das Tatbestandsmerkmal der Verletzung zu prüfen. Ungeachtet der Tatsache, dass die von Cisco verwendeten Passwörter im Internet veröffentlicht wurden, sind sie unter § 126c Abs 1 Z 2 zu subsumieren. Aus § 126c Abs 1 folgt, dass Daten iSd Z 2, sofern sie auf sozial-inadäquat Weise hergestellt, eingeführt, verschafft oder besessen werden zur Begehung des § 118a tauglich sind (vgl *5.6.2. Default-Passwörter*). Daher stellt die Verwendung der auf sozial-inadäquate Weise hergestellten, eingeführten, verschafften oder besessenen Passwörter zur Zugangsverschaffung eine Verletzung einer Sicherheitsvorkehrung iSd § 118a dar. Unter den sonstigen Voraussetzungen besteht somit eine Strafbarkeit nach § 118a.

6.4.1.3. Fehlende Zugriffsbeschränkungen

Durch die Sicherheitslücke CVE-2004-0650 ist es einem Angreifer möglich einen Upload beliebiger Dateien auf einen Cisco Collaboration Server[200] zu erwirken. Da die derart auf dem Server gespeicherten Dateien nachfolgend zur Ausführung gebracht werden können, liegt eine Zugangsverschaffung iSd § 118a vor. Eine Sicherheitsvorkehrung ist darin zu erblicken, dass der Upload beliebiger Dateien grundsätzlich unterbunden ist. Diese kann jedoch durch die Verwendung des Servlet[201] UploadServlet umgangen werden, da es den Upload beliebiger Dateien zulässt. Mangels Verletzung einer Sicherheitsvorkehrung besteht keine Strafbarkeit nach § 118a.

6.4.1.4. Denial of Service

Die Sicherheitslücken CVE-2004-0589, CVE-2004-0714 und CVE-2004-1454 ermöglichen es durch die Zusendung bestimmter Daten IOS zum Absturz zu bringen. Dies stellt eine schwere Störung eines Computersystems dar. Da diese durch eine Datenübermittlung erfolgt, ist – fehlende Verfügungsbefugnis vorausgesetzt – der objektive Tatbestand des § 126b hierdurch erfüllt.

6.4.2. Juniper, CheckPoint and Symantec Products

6.4.2.1. Zugangsverschaffung durch Speichermanipulationen

Die Sicherheitslücke CVE-2004-0699 ermöglicht es durch Ausnützung eines Buffer Overflows beliebige Befehle auf einem Check Point VPN-1 Server auszuführen. Da dies eine Verletzung einer Sicherheitsvorkehrung iSd § 118a darstellt, kommt eine Strafbarkeit nach

[197] vgl http://www.cisco.com/en/US/products/sw/voicesw/ps2237/
[198] vgl http://www.cisco.com/warp/public/707/cisco-sa-20041215-unity.shtml
[199] Simple Network Management Protocol, spezifiziert in RFC 1157. Die Authentifizierung erfolgt grundsätzlich nur durch den sog. Community String, der einem Passwort entspricht.
[200] vgl http://www.cisco.com/en/US/products/sw/custcosw/ps747/index.html
[201] Ein Servlet ist in Java geschrieben und dient grundsätzlich der Implementierung von Geschäftlogik auf einem Applikationsserver; vgl *Bergsten*, S. 21 ff

§ 118a in Betracht.

6.4.2.2. Denial of Service

Durch Ausnützung der Sicherheitslücke CVE-2004-0467 ist es möglich das von Juniper für Router verwendete JunOS[202] zum Absturz zu bringen. Dies stellt jedenfalls eine schwere Störung der Funktionsfähigkeit eines Computersystems iSd § 126b dar. CVE-2004-0468 ermöglicht es einem Angreifer lediglich bestimmte beschränkte Ressourcen des Betriebssystems JunOS derart zu überbeanspruchen, dass nicht mehr alle Datenpakete ordnungsgemäß abgearbeitet werden können.[203] Erreicht die Störung der Funktionsfähigkeit das Ausmaß einer schwere Störung (vgl hierzu *3.6.5. § 126b StGB*), kommt ebenso eine Strafbarkeit nach § 126b in Betracht.

6.4.2.3. „public“ als Default SNMP Community String

Die Sicherheitslücke CVE-2004-1474 besteht darin, dass bestimmte Produkte von Symantec mit dem unveränderlichen SNMP Community String „public“ ausgeliefert werden. Dieser Community String ermöglicht sowohl einen lesenden als auch einen schreibenden Zugriff auf Einstellungen des Geräts. Daher stellt die Ausnützung dieser Sicherheitslücke eine Zugangsverschaffung iSd § 118a dar. Da die Erforderlichkeit der Eingabe des Community Strings „public“ im Verhältnis zu gar keinem Authentifizierungsmechanismus die Wahrscheinlichkeit eines erfolgreichen Angriffs mindert, ist auch das Vorliegen einer Sicherheitsvorkehrung zu bejahen. Nun ist deren Verletzung zu prüfen. Eine solche kann sich diesfalls nur aus einer systematischen Interpretation zu § 126c Abs 1 Fall 2 ergeben, wenn Authentifizierungsdaten iSd Z 2 leg cit zur Zugangsverschaffung verwendet werden, die auf sozial-inadäquate Weise mit einem § 126c Abs 1 entsprechendem Vorsatz (einschließlich des, auf die Verwendung des Passwortes zur Begehung des § 118a gerichteten, erweiterten Vorsatzes) hergestellt, eingeführt, verschafft oder besessen werden. Da die Tatsache, dass „public“ äußerst häufig von Herstellern als Default SNMP Community String verwendet wird, zum Allgemeinwissen eines Informatikers zählen, ist dessen Kenntnis wohl auch durch einen Laien nicht als sozial-inadäquat anzusehen. Daher liegt keine Verletzung einer Sicherheitsvorkehrung vor, weshalb eine Strafbarkeit nach § 118a zu verneinen ist.

6.4.3. Cisco Devices Configuration Weaknesses

In dieser Kategorie sind keine CVE-Nummern enthalten. Dessen ungeachtet stellen Konfigurationsfehler des Ciscos Internetwork Operating System (IOS) erhebliche Sicherheitslücken dar. Von strafrechtlicher Relevanz sind insbesondere:

6.4.3.1. Default SNMP Community Strings

Wird “public” oder “private” als Default SNMP Community String verwendet, so gilt zu 6.4.2.3. „public“ als Default SNMP Community String ausgeführtes. Wird jedoch ein anderer Community String verwendet, so ist die Kenntnis von diesem grundsätzlich als sozial-inadäquat zu beurteilen. Denn andere Default Community Strings als „public“ und „private“ gelten wohl nicht als Allgemeinwissen eines Informatikers. Hieraus ergibt sich bei entsprechendem Vorsatz eine Strafbarkeit nach § 126c Abs 1 Fall 2. Wird ein derart besessener Community String zur Zugangsverschaffung verwendet kommt daher auch eine Strafbarkeit nach § 118a in Betracht.

[202] vgl http://www.juniper.net/products/junos/index.html?from=HomePage-Quicklinks-to-JUNOS
[203] vgl http://www.kb.cert.org/vuls/id/JSHA-68ZJCQ

6.4.3.2. Nonexistent Default Passwords

Diese Sicherheitslücke besteht darin, dass gar kein Passwort eingegeben werden muss um Zugang zum betreffenden Gerät zu erhalten. Der Zugang ist daher durch keine Sicherheitsvorkehrung geschützt. Daher besteht keine Strafbarkeit nach § 118a.

6.4.3.3. IP Source Routing Enabled

Als Routing bezeichnet man die Vermittlung von IP-Datenpaketen von einem Netzwerk in ein anderes Netzwerk. Das, die Vermittlung durchführende Gerät bezeichnet man als Router. Soll beispielsweise ein Datenpaket von einem Computersystem in Wien zu einem Computersystem nach New York gelangen, so sind hierbei mehr als zehn Router erforderlich um das Datenpaket von einem Teil-Netz (sog. Subnet) in ein jeweils anderes zu übermitteln. Welchen Weg das Datenpaket tatsächlich nehmen wird hängt von den Konfigurationen aller betroffenen Router ab und lässt sich daher nicht vorab feststellen.
Source Routing bezeichnet die Möglichkeit ein Datenpaket mit Routing-Informationen zu versehen, so dass das Datenpaket selbst als auch die als Antwort auf das gesendete Paket vom Empfänger zurückgesandten Datenpakte, über jene Router übertragen werden, die der Sender des ursprünglichen Pakets bestimmt hat. Technisch wird in Strict Source Routing[204] und Loose Source Routing[205] unterschieden. Während ersteres die Angabe eines exakten Routing-Pfades ermöglicht, werden mit zweiterem lediglich ein oder mehrere zu verwendende Router festgelegt.

Ungeachtet zahlreicher legitimer Anwendungsbereiche[206] des Source Routing wird meist dennoch empfohlen dieses zu deaktivieren, da es dem Sender eines Datenpakets – und somit einem potentiellen Angreifer – die Möglichkeit eröffnet, in wesentliche Aspekte der Datenkommunikation einzugreifen. Die größten Gefahren des Source Routings gehen zum einen von IP-Spoofing[207] und zum anderen von der Übertragung von Datenpaketen über vertrauenswürdige Systeme aus.

Als IP-Spoofing bezeichnet man das Fälschen der Absenderadresse eines IP-Datenpakets. Ist der Absender des gefälschten Datenpakets auch an den Antwort-Paketen interessiert, so steht er vor dem Problem, dass diese nicht an ihn, sondern an die, von ihm gefälschte Absenderadresse gesendet werden.[208] Verschickt der Täter beispielsweise Datenpakete mit einer Absenderadresse von 192.168.1.10 von einem Computersystem, dem die IP-Adresse 192.168.2.20 zugewiesen wurde, so werden alle Antworten an das Computersystem mit der IP-Adresse 192.168.1.10 und nicht an den Täter gesendet. Source Routing ermöglicht es dem Täter nun alle Antwort-Pakete, obgleich sie an die gefälschte Absenderadresse (diesfalls 192.168.1.10) gesendet werden, über einen Router seiner Wahl übermitteln zu lassen, wodurch er in die Lage versetzt wird alle Antwort-Pakete „abzufangen“.
IP-Spoofing ist insbesondere dazu geeignet Sicherheitsvorkehrungen zu überwinden, die einen Zugang nur durch bestimmte, vertrauenswürdige IP-Adressen ermöglichen. Mangels Verletzung einer Sicherheitsvorkehrung besteht jedoch keine Strafbarkeit nach § 118a.

Eine weitere Möglichkeit Source Routing für Angriffe auf Computersysteme zu missbrauchen, besteht in folgendem Szenario. Angriffsziel ist ein Computersystem, das

---

[204] auch Strict Source Record Route (SSRR); vgl RFC 791, S. 18
[205] auch Loose Source Record Route (LSRR); vgl RFC 791, S. 17
[206] vgl http://www.iss.net/security_center/advice/Underground/Hacking/Methods/Technical/Source_Routing/default.htm
[207] vgl *daemon9*
[208] Dies gilt insbesondere für den Aufbau von TCP/IP-Verbindung, da diese einen sog. Three-Way-Handshake erfordern; vgl *Hunt*, S. 19

ausschließlich mit dem internen Netzwerk, jedoch nicht unmittelbar mit dem Internet verbunden ist. Besitzt ein zweites Computersystem sowohl eine Verbindung mit dem internen Netzwerk als auch mit dem Internet, so könnten mittels Source Routing Datenpakete vom Angreifer aus dem Internet über besagtes zweites System zum eigentlichen Angriffsziel gesendet werden. Hierdurch ist es möglich ein, im internen Netzwerk befindliches und grundsätzlich nicht aus dem Internet erreichbares Computersystem dennoch über andere Computersysteme zu erreichen.[209] Die diesfalls bestehende Sicherheitsvorkehrung der Abschottung eines Computersystems wird in ihrer Substanz durch diesen Angriff nicht beeinträchtigt. Es liegt daher keine Verletzung sondern bloß eine Überwindung der Sicherheitsvorkehrung vor, weshalb keine Strafbarkeit nach § 118a besteht.

6.4.3.4. IP Directed Broadcast Enabled

Die Aktivierung der Option „IP Directed Broadcast“ bewirkt, dass ein Router, der ein IP-Paket erhält, das als Zieladresse eine Broadcast-Adresse[210] trägt, weiterleitet. Das Senden eines Datenpaketes an eine Broadcast-Adresse ist gleichbedeutend mit dem Senden eines Pakets an alle Computersysteme des lokalen Netzwerkes. Diese Konfigurationsoption kann in folgendem Angriffsszenario ausgenützt werden: der Angreifer sendet ein ICMP-Paket vom Typ echo request[211] (sog. „ping“) an eine Broadcast-Adresse, verwendet jedoch als Absenderadresse mittels IP-Spoofing die Adresse eines dritten Computersystems. Alle durch die Broadcast-Adresse bezeichneten Systeme senden hierauf ICMP echo reply Pakete an den vermeintlichen Absender der echo request Pakete. Hierdurch ist es dem Angreifer möglich einen Denial of Service Attack gegen den vermeintlichen Absender der ICMP echo request Pakete derart auszuführen, dass der Angriff nicht von ihm sondern von den, durch die Broadcast-Adresse bezeichneten Systemen zu kommen scheint. Dieser Angriff wird als „smurf“[212] bezeichnet. Kommt es zu einer schweren Störung der Funktionsfähigkeit des angegriffenen Systems, so ist der objektive Tatbestand des § 126b durch die Tathandlung der Datenübermittlung erfüllt.

[209] vgl http://www.iss.net/security_center/advice/Underground/Hacking/Methods/Technical/Source_Routing/default.htm

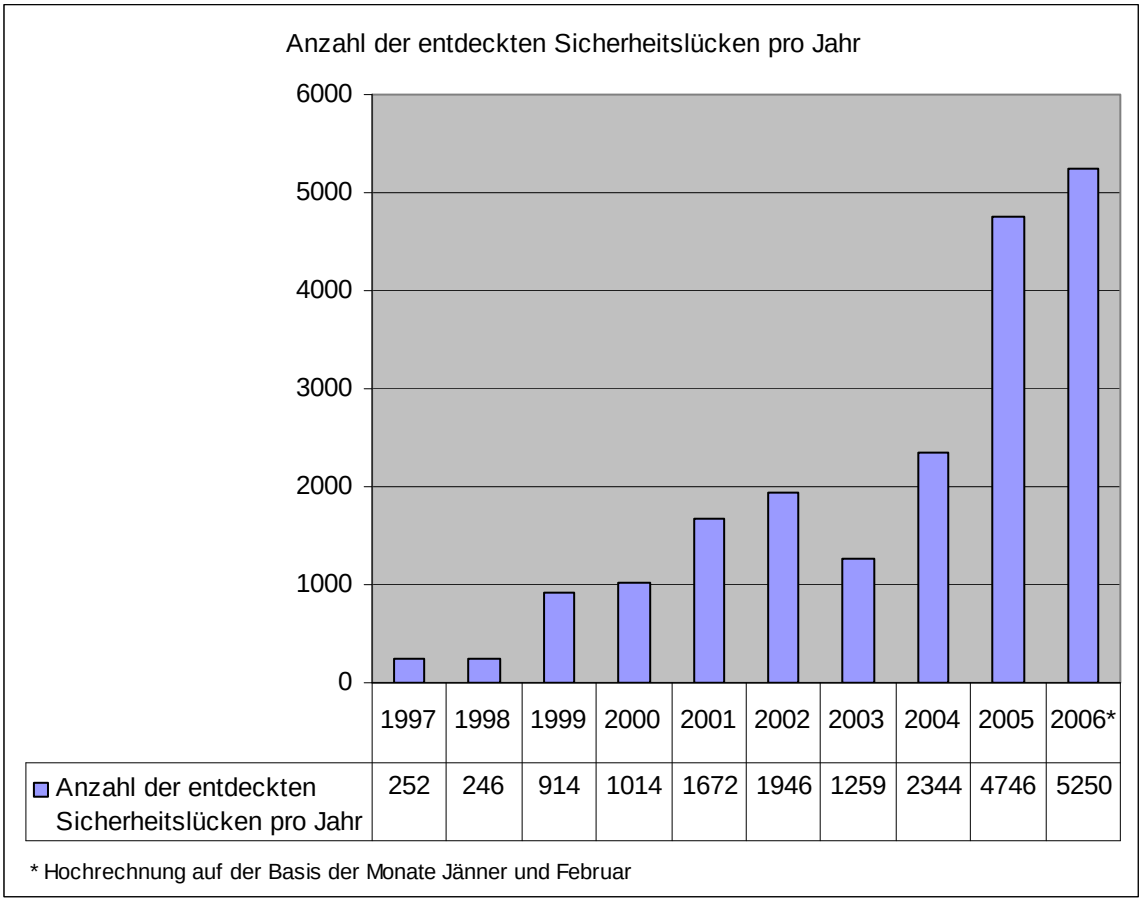
[210] Es handelt sich hierbei um eine IP-Adresse deren Host-Bits alle auf 1 gesetzt sind; vgl RFC 919; *Kirch/Dawson*, S. 79

[211] vgl RFC 792, S. 13 f

[212] CERT Advisory CA-1998-01, Smurf IP Denial-of-Service Attacks, http://www.cert.org/advisories/CA-1998-01.html

7. Statistische Auswertung

Der Veranschaulichung der wachsenden Bedeutung von Sicherheitslücken soll folgende Statistik über die Anzahl der jährlich entdeckten Sicherheitslücken dienen. Als Quelle diente die Statistics Query Page der National Vulnerability Database[213]. In den Jahren 1997 und 1998 wurden nur 252 bzw. 246 Sicherheitslücken entdeckt. Dies entspricht weniger als einer Sicherheitslücke pro Tag. Seit dem Jahr 2000 lag die Anzahl der jährlich entdeckten Sicherheitslücken stets über 1000 pro Jahr. Die im Jahre 2005 entdeckten 4746 Sicherheitslücken entsprechen 13 neu entdeckten Sicherheitslücken pro Tag. In den ersten beiden Monaten des Jahres 2006 wurden 875 Sicherheitslücken entdeckt. Eine Hochrechnung für das gesamte Jahr 2006 ergibt daher 5250 Sicherheitslücken. Dies entspricht einem Anstieg um ca. 10,7% im Vergleich zum Jahr 2005.



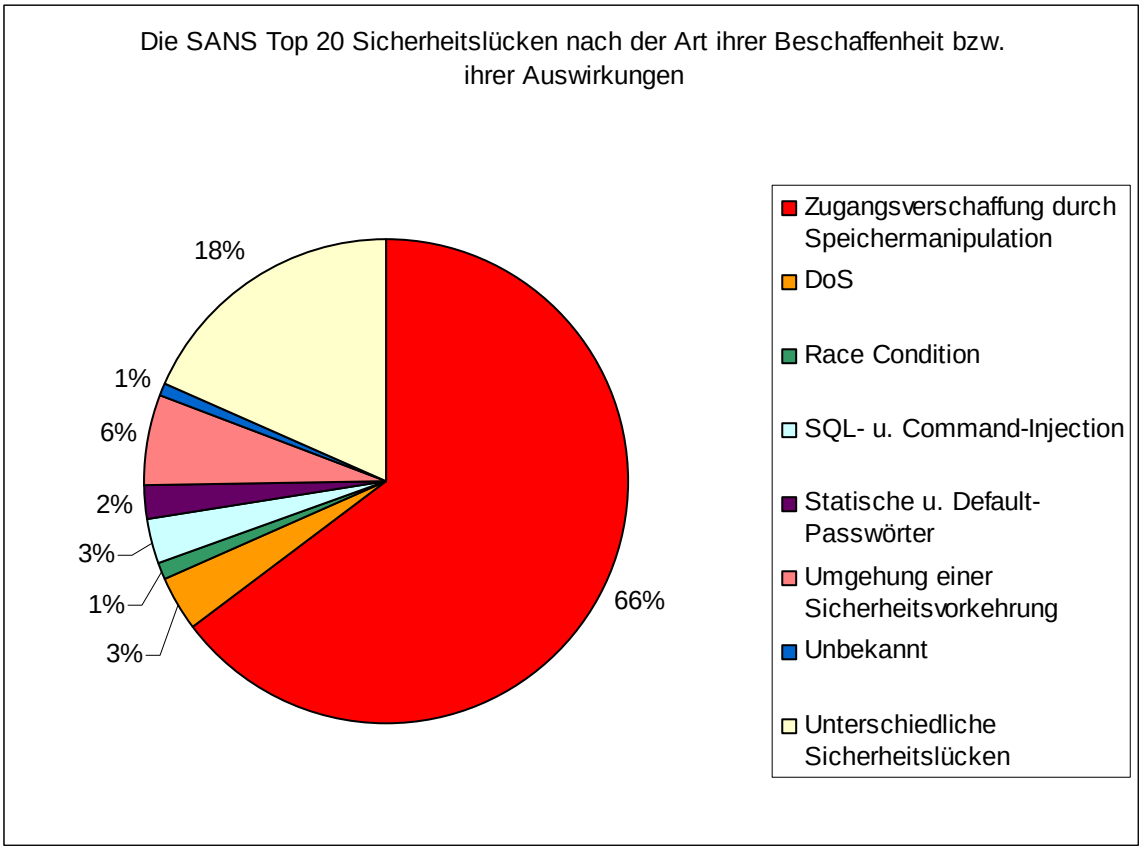
	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006*
Anzahl der entdeckten Sicherheitslücken pro Jahr	252	246	914	1014	1672	1946	1259	2344	4746	5250

* Hochrechnung auf der Basis der Monate Jänner und Februar

Die nun folgende Statistik soll die Bedeutung der unterschiedlichen Arten von Sicherheitslücken verdeutlichen. Es wurden hierbei ausschließlich Sicherheitslücken mit CVE-Nummern der SANS Top 20 Sicherheitslücken berücksichtigt. Zur Erstellung der Statistik dienten die, im Rahmen dieser Arbeit erörterten technischen Beschaffenheiten der einzelnen Sicherheitslücken. Die Statistics Query Page der National Vulnerability Database wurde nicht verwendet, da die dort vorgenommene Klassifizierungen teilweise nicht hinreichend präzise sind. Die Kategorie der Zugangsverschaffung durch Speichermanipulation erfasst die folgenden Arten von Sicherheitslücken: Buffer Overflows,

[213] http://nvd.nist.gov/statistics.cfm

Integer Overflows, Format String Bugs und Double-free Bugs. Da genannte Kategorie 66% der SANS Top 20 Sicherheitslücken ausmacht, kann ihre rechtliche Beurteilung als praktisch äußerst relevant erachtet werden.



8. Zusammenfassung

Für die strafrechtliche Beurteilung ist es von entscheidender Bedeutung, welcher Sicherheitslücken sich der Täter zum Zwecke des Angriffs auf die Vertraulichkeit, Integrität oder Verfügbarkeit von Daten oder Systemen bedient. Insbesondere im Fall der Zugangsverschaffung macht es das Tatbestandsmerkmal der Verletzung einer Sicherheitsvorkehrung iSd § 118a StGB erforderlich, die konkret verwendete Sicherheitslücke auf ihre technische Beschaffenheit zu untersuchen. Wie gezeigt werden konnte, lässt sich eine Zugangsverschaffung unter Verwendung von Buffer Overflows als einer der wichtigsten Fälle erfolgreich unter den objektiven Tatbestand des § 118a subsumieren. Andere Fälle wie Script- bzw. Command-Injection oder Race Conditions stellen hingegen keine Verletzung einer Sicherheitsvorkehrung dar. Das Hacking unter Verwendung fremder Passwörter erfordert demgegenüber eine sehr differenzierte Betrachtung. Im Ergebnis ist eine Strafbarkeit nach § 118a in jenen Fällen zu bejahen, in denen bereits im Vorbereitungsstadium eine Strafbarkeit gem § 126c Abs 1 Fall 2 bestand.
Da auch der Rahmenbeschluss 2005/222/JI des Rates vom 24. Februar 2005 über Angriffe auf Informationssysteme den Mitgliedstaaten die Möglichkeit eröffnet die „Verletzung von Sicherheitsmaßnahmen“ in den Tatbestand aufzunehmen, werden sich ähnliche dogmatische Probleme wie nach der derzeitigen österreichischen Rechtslage nach der Umsetzung der Richtlinie auch in anderen Mitgliedstaaten stellen.

Die in dieser Arbeit enthaltene strafrechtliche Beurteilung von Angriffen unter Verwendung der verschiedenen SANS Top 20 Internet Security Vulnerabilities sollte die Beurteilung der meisten, in der Praxis auftretenden Fälle wesentlich erleichtern.

9. Abkürzungsverzeichnis

aA	anderer Ansicht
Abs	Absatz
Art	Artikel
BGBl	Bundesgesetzblatt
BlgNr	Beilage(n) zu den stenographischen Protokollen des Nationalrats
bzw.	beziehungsweise
ca.	circa
CyCC	Convention on Cyber-Crime des Europarats
dh	das heißt
DSG	Datenschutzgesetz
f, ff	folgende
gem	gemäß
hA	herrschende Ansicht
Hrsg	Herausgeber
idF	in der Fassung
idR	in der Regel
idS	in diesem Sinne
ieS	im engeren Sinn
iwS	im weiteren Sinn
IP	Internet Protocol
iS	im Sinne
iSv	im Sinne von
IT	Informationstechnologie
iVm	in Verbindung mit
leg cit	legis citatae (der zitierten Vorschrift)
lit	litera
mE	meines Erachtens
Nr	Nummer
OGH	Oberster Gerichtshof
RB	Rahmenbeschluss
RV	Regierungsvorlage
Rz	Randzahl
S	Seite
sog	sogenannt, -e, -er, -es
StGB	Strafgesetzbuch
StRÄG	Strafrechtsänderungsgesetz
TCP	Transmission Control Protocol
ua	unter anderem
URL	Uniform Resource Locator
uU	unter Umständen
vgl	vergleiche
wbl	Wirtschaftsrechtliche Blätter
Z	Ziffer
zB	zum Beispiel

10. Literaturverzeichnis

Alle in dieser Arbeit enthaltenen Internetadressen beziehen sich auf 5. März 2006.

Albitz/Liu, DNS and BIND[4], O'Reilly, Sebastopol 2001

Aleph One, Smashing The Stack For Fun And Profit, Phrack Magazine, Volume Seven, Issue Forty-Nine, File 14 of 16, 1996, http://www.phrack.org/phrack/49/P49-14

Anonymous, Once upon a free()..., Phrack Magazine, Volume 0x0b, Issue 0x39, Phile #0x09 of 0x12, 2001, http://www.phrack.org/phrack/57/p57-0x09

Barman, Writing Information Security Policies, Sams, Indiana 2001

Barrett/Silverman, SSH, The Secure Shell: The Definitive Guide, O'Reilly, Sebastopol 2001

Bergsten, JavaServer Pages, O'Reilly, Sebastopol 2001

Bott/Siechert/Stinson, Microsoft Windows XP Inside Out, Deluxe Edition, Microsoft Press, Redmond 2002

Boutin, Slammed!, An inside view of the worm that crashed the Internet in 15 minutes, Wired Magazine, Issue 11.07, Juli 2003, http://www.wired.com/wired/archive/11.07/slammer.html

Bäumler, Eine sichere Informationsgesellschaft – Zur europäischen Bekämpfung der Computerkriminalität, DuD 2001, S. 348

Case/Fedor/Schoffstall/Davin, RFC 1157, Simple Network Management Protocol (SNMP), 1990, ftp://ftp.rfc-editor.org/in-notes/rfc1157.txt
(RFC 1157)

Conover, w00w00 on Heap Overflows, 1999,
http://www.w00w00.org/files/articles/heaptut.txt

Cooper/Northcutt/Fearnow/Frederick, Intrusion Signatures and Analysis, Sams, Indiana 2001

Cowan/Wagle/Pu/Beattie/Walpole, Buffer Overflows: Attacks and Defenses for the Vulnerability of the Decade, http://downloads.securityfocus.com/library/discex00.pdf

Cox/Gerg, Managing Security with Snort and IDS Tools, O'Reilly, Sebastopol 2004

Crispin, RFC 3501, Internet Message Access Protocol - Version 4Rev1, 2003, ftp://ftp.rfc-editor.org/in-notes/rfc3501.txt
(RFC 3501)

Crocker, RFC 822, Standard for the Format of Arpa Internet Text Messages, 1982,
ftp://ftp.rfc-editor.org/in-notes/rfc822.txt
(RFC 822)

daemon9, IP-spoofing Demystified, Phrack Magazine, Volume 7, Issue 48, File 14 of 18, 1996, http://www.phrack.org/phrack/48/P48-14

Donaldson, Inside the Buffer Overflow Attack: Mechanism, Method & Prevention, 2002 http://www.sans.org/rr/whitepapers/securecode/386.php

DuBois, MySQL², Sams, Indiana 2003

Eckel, Thinking in Java³, Prentice Hall PTR, Upper Saddle River 2002

Erdmann/Wegener, Dienstschutz, DNS-Sicherheit: Der Stand der Dinge, iX 12/2005, S. 142

Fielding/Gettys/Mogul/Frystyk/Masinter/Leach/Berners-Lee, RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1, 1999, ftp://ftp.rfc-editor.org/in-notes/rfc2616.txt
(RFC 2616)

Flanagan, JavaScript: The Definitive Guide³, O'Reilly, Sebastopol 1998

Frisch, UNIX System Administration[2], O'Reilly, Köln 2000

Fuchs, Österreichisches Strafrecht. Allgemeiner Teil I[6], Springer, Wien 2004

Garfinkel/Spafford/Schwartz, Practical Unix & Internet Security[3], O'Reilly, Sebastopol 2003

Geiger, Indiskretionsdelikte und neue Medien – Zur strafrechtlichen Relevanz der Überwachung privater Kommunikation mittels „Internet Monitoring Software", 2003, http://www.laga.at/rechtsprobleme/doks/indiskretionsdelikte_und_neue_medien-geiger.pdf

Graff/Van Wyk, Secure Coding: Principles and Practices, O'Reilly, Sebastopol 2003

Held, Ethernet Networks: Design, Implementation, Operation[4], Management, John Wiley & Sons, Chichester 2002

Höpfel/Ratz, Wiener Kommentar zum StGB[2], Manz, Wien 2004
(*Autor* in WK)

Hunt, TCP/IP Network Administration³, O'Reilly, Sebastopol 2002

International Information Security Foundation, Generally Accepted System Security Principles (GASSP), Version 2.0, 1999, http://www.infosectoday.com/Articles/gassp.pdf

Information Systems Security Association (ISSA), Generally Accepted Information Security Principles (GAISP), Version 3.0, 2004, http://www.issa.org/gaisp/_pdfs/v30.pdf

Kallnik/Pape/Schröter/Strobel, Das Sicherheitsloch – Buffer-Overflows und wie man sich davor schützt, c't 23/2001, S. 126

Kantor/Lapsley, RFC 977, Network News Transfer Protocol, 1986, ftp://ftp.rfc-editor.org/in-notes/rfc977.txt
(RFC 977)

Kirch/Dawson, The Linux Network Administrator's Guide², O'Reilly, Sebastopol 2000; als Open Book online verfügbar unter http://www.tldp.org/guides.html

Klensin, RFC 2821, Simple Mail Transfer Protocol, 2001, ftp://ftp.rfc-editor.org/in-notes/rfc2821.txt
(RFC 2821)

Krsul/Spafford/Tripunitara, An Analysis of Some Software Vulnerabilities, 1998, http://widsard.sourceforge.net/doc/03.pdf

Kugelmann, Die „Cyber-Crime“ Konvention des Europarats, DuD 2001, S. 215

Levine, qmail, O'Reilly, Sebastopol 2004

Lowery, A Tour of TOCTTOUs, SANS GSEC Practical v.1.4b, 2002; http://www.sans.org/rr/whitepapers/securecode/1049.php

Loney/Koch, Oracle9i: The Complete Reference, McGraw-Hill, Berkeley 2002

McNab, Network Security Assessment, O'Reilly, Sebastopol 2004

Mell/Kent/Nusbaum, NIST SP 800-83, Guide to Malware Incident Prevention and Handling, National Institute of Standards and Technology, U.S. Department of Commerce, 2005, http://csrc.nist.gov/publications/nistpubs/800-83/SP800-83.pdf

Middendorf, Datenbanken: Sicherheitslecks und wie man sie stopft; iX 11/2003, S. 110

Miller/Fredriksen/So, An Empirical Study of the Reliability of UNIX Utilities, Communications of the ACM 33, December 1990, 32-44, ftp://ftp.cs.wisc.edu/paradyn/technical_papers/fuzz.ps

Miller/Koski/Lee/Maganty/Murthy/Natarajan/Steidl, Fuzz Revisited: A Re-examination of the Reliability of UNIX Utilities and Services, 1995, http://www.opensource.org/advocacy/fuzz-revisited.pdf

Mitnick/Simon/Wozniak, The Art of Deception: Controlling the Human Element of Security, John Wiley & Sons, Chichester 2002

Mogul, RFC 919, Broadcasting Internet Datagrams, 1984, ftp://ftp.rfc-editor.org/in-notes/rfc919.txt
(RFC 919)

Myers/Rose, RFC 1939, Post Office Protocol - Version 3, 1996, ftp://ftp.rfc-editor.org/in-notes/rfc1939.txt
(RFC 1939)

Nebel/Masinter, RFC 1867, Form-based File Upload in HTML, 1995, ftp://ftp.rfc-editor.org/in-notes/rfc1867.txt
(RFC 1867)

Newberry, Virus Writers 360° - An analysis of virus writers; SANS GSEC certification Practical v1.4b, 2004; http://www.sans.org/rr/whitepapers/malicious/1523.php

Newham/Rosenblatt, Learning the bash Shell², O'Reilly, Sebastopol 1998

Northcutt/Zeltser/Winters/Fredrick/Ritchey, Inside Network Perimeter Security: The Definitive Guide to Firewalls, Virtual Private Networks (VPNs), Routers, and Intrusion Detection Systems, Sams, Indiana 2002

Otto/Parschalk, Spam- und Virenfilter - eine Notwendigkeit im Graubereich des Rechts, wbl 2005, S. 10

Pate, UNIX Filesystems: Evolution, Design, and Implementation, John Wiley & Sons, Chichester 2003

Peek/Todino-Gonguet/Strang, Learning the UNIX Operating System, O'Reilly, Sebastopol 2002

Peikari/Chuvakin, Security Warrior, O'Reilly, Sebastopol 2004

Plöckinger, Internet und materielles Strafrecht – Die Convention on Cyber-Crime, in *Plöckinger/Duursma/Helm*, Aktuelle Entwicklungen im Internet-Recht, NWV, Wien 2002

Postel, RFC 791, Internet Protocol, 1981, ftp://ftp.rfc-editor.org/in-notes/rfc791.txt
(RFC 791)

Postel, RFC 792, Internet Control Message Protocol, 1981, ftp://ftp.rfc-editor.org/in-notes/rfc792.txt
(RFC 792)

Preston, Unix Backup & Recovery, O'Reilly, Sebastopol 1999

Puppe/Maier, Von allen Seiten, Maßnahmen gegen Distributed-Denial-of-Service-Angriffe, iX 4/2005, S. 107

Puri, Bots & Botnet: An Overview, SANS GSEC Practical v.1.4b, 2003
http://www.sans.org/rr/whitepapers/malicious/1299.php

Reindl, E-Commerce und Strafrecht. Zur Strafbarkeit des Missbrauchs elektronischer Dienste, NWV, Wien 2003

Resnick, RFC 2822, Internet Message Format, 2001, ftp://ftp.rfc-editor.org/in-notes/rfc2822.txt
(RFC 2822)

Russel/Crawford/Gerend, Microsoft Windows 2000 Server Administrator's Companion[2], Microsoft Press, Redmond 2002

Schwartz, SpamAssassin, O'Reilly, Sebastopol 2004

Sieber, Strafrecht und Strafprozessrecht, in Hoeren/Sieber, Handbuch Multimedia-Recht, Beck, München 1999

Sieber, Verantwortlichkeit im Internet, Beck, München 1999

Spafford, One View of A Critical National Need: Support for Information Security Education and Research, 1997, http://homes.cerias.purdue.edu/~spaf/usgov/edu.pdf

Spafford, UNIX and Security: The Influences of History, 1995, http://www.cs.purdue.edu/coast/archive/data/categ28.html#marker408

Stanfield/Smith, Linux System Administration, Sybex, Alameda 2001

Sterne/Balenson/Branstad/Jaworski/Lee/Pfleeger/Osuna/Vechery/Walker/Winkler-Parenty/Bassham III/Jacobson/Wack, NIST SP 800-12, An Introduction to Computer Security: The NIST Handbook, National Institute of Standards and Technology, U.S. Department of Commerce, 1995 http://csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf

Stoneburner, NIST SP 800-33, Underlying Technical Models for Information Technology Security, National Institute of Standards and Technology, U.S. Department of Commerce, 2001, http://csrc.nist.gov/publications/nistpubs/800-33/sp800-33.pdf

Stoneburner/Goguen/Feringa, NIST SP 800-30, Risk Management Guide for Information Technology Systems, National Institute of Standards and Technology, U.S. Department of Commerce, 2002, http://csrc.nist.gov/publications/nistpubs/800-30/sp800-30.pdf

Stoneburner/Hayden/Feringa, NIST SP 800-27 Rev A, Engineering Principles for Information Technology Security (A Baseline for Achieving Security), National Institute of Standards and Technology, U.S. Department of Commerce, 2004, http://csrc.nist.gov/publications/nistpubs/800-27A/SP800-27-RevA.pdf

Sun Microsystems, Solaris 10 System Administrator Collection – System Administration Guide: Basic Administration, 2005, http://docs.sun.com/app/docs/coll/47.16

Swanson/Bartol/Sabato/Hash/Graffo, NIST SP 800-55, Security Metrics Guide for Information Technology Systems, National Institute of Standards and Technology, U.S. Department of Commerce, 2003, http://csrc.nist.gov/publications/nistpubs/800-55/sp800-55.pdf

Swanson/Guttman, NIST SP 800-14, Generally Accepted Principles and Practices for Securing Information Technology Systems, National Institute of Standards and Technology, U.S. Department of Commerce, 1996, http://csrc.nist.gov/publications/nistpubs/800-14/800-14.pdf

Tennert, Gegen die Wand – Mit Exec Shield gegen Buffer-Overflow-Attacken, iX 10/2004, S. 112

Urman, Oracle9i PL/SQL Programming, McGraw-Hill, Berkeley 2002

Van de Ven, Limiting buffer overflows with ExecShield, Red Hat Magazine, Issue #9, 2005, http://www.redhat.com/magazine/009jul05/features/execshield/

Welsh/Kaufman/Dalheimer/Dawson, Running Linux[4], O'Reilly, Sebastopol 2002

Younan, An overview of common programming security vulnerabilities and possible solutions, 2003, http://www.fort-knox.org/thesis.pdf

Zeder, Internet und Strafrecht, in Studiengesellschaft für Wirtschaft und Recht (Hrsg), Internet und Recht. Rechtsfragen von E-Commerce und E-Government, Linde, Wien 2002

Zwicky/Cooper/Chapman, Building Internet Firewalls², O'Reilly, Sebastopol 2000