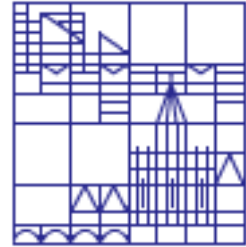


Universität Konstanz
Fachbereich Informatik und Informationswissenschaft
Wintersemester 2002/2003
Gutachter:
Prof. Dr. Rainer Kuhlen
Prof. Dr. Marc H. Scholl



Sicherheitsaspekte des Aufbaus eines Internet-Portal-Systems am Beispiel des Portals zur Informationsethik nethics.net

Richard Wonka
01/423573
Schiffstraße 3
78464 Konstanz
richard.wonka@uni-konstanz.de

Zusammenfassung

Dieses Papier behandelt die Sicherheitsfragen, die sich bei der Implementation eines Serversystems für das Portal für Informationsethik nethics.net stellen.

Neben den Fragen nach der technischen Sicherheit, stellen sich hierbei auch solche, die aus Definitionen von Sicherheit aus dem Begriffsfeld der Informationswissenschaften her rühren. Es zeigt sich, dass die Antworten auf diese Fragen einen direkten Einfluss auf die Beantwortung der Fragen nach technischer Sicherheit haben.

Besonderes Augenmerk soll dabei auf dem Einfluss des Entwicklungsmodells auf die Sicherheitsmerkmale von Software liegen. Die Entwicklungsmodelle proprietärer und *closed Source* Software werden hierbei mit denen von *open Source* Software im Allgemeinen und freier Software im Besonderen verglichen.

abstract

This paper deals with security-related questions that occurred while implementing a server-system for the Web-Portal on information ethics nethics.net.

Next to the security issues of a purely technical nature, it deals with questions that are not so much technical, but derive from definitions of security in the field of information science and information ethics. It shows, that answering these specific questions has a direct effect on the answers concerning technical security.

Of special concern is the influence that the general concept of software development has on security issues of the same software. The concepts used in the development of proprietary and closed-source software are compared to those applied in the development of open source software in general and free software in particular.

Inhaltsverzeichnis

I	Definitionen	3
1	Anbieter sind auch Nutzer	3
2	Unterscheidungen von Software	4
3	Allgemeine Konstruktionsprinzipien	6
4	Netzwerkmodelle	9
5	Sicherheit	10
6	Der erweiterte Sicherheitsbegriff	12
II	Technische Sicherheit	14
7	Ermittlung des Schutzbedarfs	15
8	Hardware	20
9	Das Netzwerk	24
10	Wahl der Software	31
11	Zugriffskontrolle	39
12	Aufrechterhaltung	45
III	Die erweiterte Sicherheit	47
13	Informationelle Selbstbestimmung	47

14 Rechtliche Absicherung	49
A Klauseln zum Haftungsausschluss	52
B Beispielszenarien des BSI	57

Abbildungsverzeichnis

1	Software-Kategorien nach deren Lizenz	4
2	Die Referenzmodelle im Vergleich	10
3	SSL Tunnel	28

Tabellenverzeichnis

1	Schutzbedarfsfestellung nach dem BSI Grundschriftzhandbuch .	21
2	Relevante Dienste für ein Portalsystem im WWW	30
3	Rechtevergabe für anonyme Besucher	42
4	Rechtevergabe für eingetragene Benutzer	43
5	Rechtevergabe für Direktveröffentlicher	44
6	Rechtevergabe für Lektoren	45
7	Rechtevergabe für Administratoren	45

Einleitung

Der Begriff der Sicherheit wird durch technische Sicherheit nicht ausgefüllt, sondern bedarf auch aus einer nicht-technischen Warte einer eingehenderen Betrachtung. Dieser erweiterte Blick hat aber auch Einfluss auf technische Entscheidungen, die beim Prozess des Aufbaus eines Portals anstehen. So zeigt sich, dass Überlegungen aus dem Themenkomplex der Informationswissenschaft und (Informations-)Ethik klare Konsequenzen für die Entscheidungsfindung in technischen Fragen haben.

Was diese Arbeit *nicht* kann

Diese Arbeit kann - allein aufgrund ihres Umfangs - keine detaillierte Anleitung zur Erstellung eines sicheren Serversystems geben. Darüber wurden schon viele Bücher geschrieben, deren Inhalt hier zu wiederholen kaum sinnvoll wäre.

Noch können in diesem Papier alle Sicherheitsfragen, die beim Aufbau eines solchen Systems aufkommen detailliert betrachtet, geschweige denn beantwortet werden.

Die Entscheidungen, die beim Erstellen eines Serversystems zu treffen sind, sind von vielen Faktoren abhängig, von denen die Sicherheit nur einer - wenn auch ein oft unterschätzter - ist. Deshalb kann diese Arbeit auch nicht vorgeben, wie solche Entscheidungen auszufallen haben.

Was diese Arbeit kann

Diese Arbeit kann einen Überblick verschaffen, welche Fragen beim Aufbau eines Serversystems im Bezug auf dessen Sicherheit - und auf die Sicherheit der Betreiber und Nutzer - gestellt werden können und sollten. Auch wird sie den Leser bei den Überlegungen begleiten, die diese Fragen aus technischer wie auch aus nicht-technischer Sicht anstoßen.

Auch kann diese Arbeit aufzeigen, dass sich aus der Entwicklung hin zu einer Wissens- und Informationsgesellschaft, also einer Verfügbarkeit und Anwendbarkeit hoch komplexer Serverbetriebssysteme für eine breite Bevölkerungsschicht, Fragen nach der Sicherheit ergeben, die nicht rein technischer Natur sind und deren Beantwortung nicht nur eines technischen Hintergrundes bedarf, da die technischen und die nicht-technischen Aspekte der Sicherheit in engem Zusammenhang stehen.

Aufgabenstellung

Die Fragestellung für diese Arbeit entstand aus einem Projektpraktikum in Zusammenarbeit mit Alexander Holupirek im Zuge des Studienganges *Information Engineering* an der Universität Konstanz. In Teams absolvieren die Studierenden hier ein Projektpraktikum, bei dem die während des Studiums erlangten Kenntnisse und Fertigkeiten im Rahmen eines betreuten Projektes umgesetzt werden sollen.

Unter der Anleitung von Prof. Rainer Kuhlen und Joachim Griesbaum war es unsere Aufgabe, für den Internetauftritt des Vereins für Informatikethik Nethics ein Konzept zu erarbeiten, das die Integration eines Web Content Management Systems vorsah.

Die erstrebte Funktionalität sollte für den Verein die technische Basis liefern, um als Portal fungieren zu können. Unabdingbare Voraussetzung war also vor allem die Möglichkeit der Interaktion und Einbringung der Nutzer. Die Anforderungen waren also

- Die Bereitstellung eines Web-Autorensystems. Den Nutzern des Portals soll es ermöglicht werden, mit einem Mindestmaß technischen Vorwissens Inhalte auf der Website zu veröffentlichen.
- Die Bereitstellung eines Kommunikationsforums. Den Nutzern des Portals soll es ermöglicht werden, die Website zu benutzen, um themenbezogene öffentliche Diskussionen zu führen.
- Die Möglichkeit der Erweiterbarkeit, um eventuell aus dem Einsatz erwachsenden Ansprüchen gerecht werden zu können.

Das Ergebnis der Konzeptionierung und Implementation einer solchen Plattform sollte beispielhaft für andere kleinere *non governmental Organizations* (NGO), also auf ähnlich geartete und skalierte Anforderungen übertragbar sein.

Diese Arbeit wird sich mit den konzeptionellen Überlegungen auseinandersetzen, die sich mit der Sicherheit eines solchen Systems beschäftigen, und in weiten Teilen die Entscheidungen, die bei der Implementation eines Beispielsystems gefallen sind beschreiben. Besonderes Augenmerk wird dabei auf die Wahl der verwandten Software geworfen.

Das beschriebene ist jedoch nicht das System, das zum Sommersemester 2003 öffentlich zugänglich wird. Die Integration eines Portalsystems in bestehende komplexe Systeme wie das der Universität Konstanz bedarf einiger anderer - vor allem praktischer - Überlegungen und soll nicht Thema dieser Arbeit sein.

Teil I

Definitionen

Zunächst wird im Weiteren das den nachfolgenden Ausführungen zugrunde liegende Verständnis einiger Begriffe und Konzepte erläutert - und dabei zum Teil auch das traditionelle Verständnis dieser Begriffe erweitert. Der so aufgespannte konzeptionelle Raum soll dann als Basis für die weiteren Betrachtungen dienen.

Auf diesen Grundbegriffen aufbauend wird die Sicherheit und die Sicherung von Informationssystemen aus verschiedenen Blickwinkeln betrachtet und deren Umsetzung an einem Beispielsystem beschrieben.

1 Anbieter sind auch Nutzer

Traditionell wird bei Dienstleistungen im Allgemeinen und Informationsdienstleistungen im Speziellen zwischen Anbietern und Nutzern unterschieden. Diese Trennung wird jedoch in der Informationstechnologie durch die weite Verbreitung von IT und den einfachen Zugang zu Systemen, die es ermöglichen, Informationsdienste anzubieten, nach und nach aufgehoben. Die Entwicklung der Gesellschaft zu einer Wissens- und Informationsgesellschaft macht die Mitglieder dieser Gesellschaft zu Nutzern und Anbietern von Information zugleich.

Leistungsfähige Personal Computer und deren Anbindung an schnelle globale Netzwerkstrukturen machen das Ausnutzen der Freiheiten der (oder Rechte auf) Information und Kommunikation immer einfacher. Jeder, der eigene Inhalte im Internet verbreitet, sei es auf der eigenen Homepage oder durch die aktive Teilnahme am Kommunikationsprozess in Foren oder auf Portalen, erweitert seinen Wirkungsbereich von dem eines Nutzers um den eines Anbieters von Information.

Bemerkbar wird diese Tatsache auch daran, dass es nach der großen Welle der Kommerzialisierung des Internets eine Rückentwicklung dahin gibt, dass die Zahl der Angebotene Dienste steigt, hinter denen nicht finanzielle Interessen, sondern der Transport von Inhalten steht und viele NGOs das Internet nutzen, um sich und/oder ihre Dienste einer weltweiten Öffentlichkeit zu präsentieren.

Es ist aber nicht die generelle Steigerung der informationstechnischen Kompetenz der Anbieter, die dies möglich macht, sondern paradoxerweise

die steigende Komplexität der Anwendungen, welche solche Dienstleistungen möglich machen. Die (künstliche) Vereinfachung der Handhabung komplexer Serversoftware zum Beispiel durch den Einsatz graphischer Benutzeroberflächen und 'halbintelligenter' Automatismen fördert also eine Situation, in der die Anbieter selbst zu Nutzern vermittelnder Informationsdienstleistungen werden. Sie nutzen eine Dienstleistung (die Serversoftware), die sie befähigt, eine (möglicherweise andere) Dienstleistung anzubieten.

Die Software, die solchen Dienstleistungen zugrunde liegt, soll also im Weiteren als eine Teilmenge der Informationsdienstleistungen betrachtet werden.

2 Unterscheidungen von Software

Bei der Betrachtung der Sicherheitsaspekte der Anwendung von Software werden wir besonderes Augenmerk auf die unterschiedlichen Lizenzmodelle werfen, denen die Nutzer dieser Software unterworfen sind. Wie sich zeigen wird, haben diese Lizenzmodelle direkte Auswirkungen auf Sicherheitsaspekte der Dienstleistung selbst. Hierbei werden wir uns weitgehend nach den in [FSF02] definierten Kategorien richten, die von der *Free Software Foundation* formuliert wurden. Es wird jedoch ein vereinfachtes Modell benutzt, das einige Feinheiten übersieht, die aus unserer Warte vernachlässigbar sind.

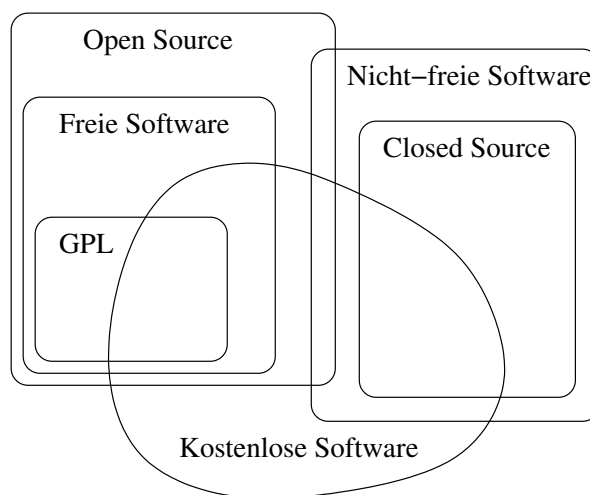


Abbildung 1: Software-Kategorien nach deren Lizenz

Proprietäre Software zeichnet sich dadurch aus, dass ihre Nutzung, Veränderung oder Verbreitung eingeschränkt erlaubt oder verboten ist oder dass dazu eine explizite Erlaubnis notwendig ist, oder ihre Verbreitung nur so eingeschränkt möglich ist, dass effektiv nicht mehr von freier Veränderbarkeit oder Erlaubnis zur Verbreitung die Rede sein kann. An dieser Stelle weichen wir etwas von der Definition in [FSF02] ab und bezeichnen proprietäre Software als *nicht-freie* Software.

Closed Source bezeichnet jene Softwareprodukte, deren Quellcode nur dem Hersteller zugänglich und nicht für die Öffentlichkeit einsehbar ist. Sie ist eine Teilmenge der proprietären Software.

Open source Software beschreibt solche Software, deren Quellcode für jedermann zur Einsicht offen liegt. Sie schließt auch solche mit ein, deren Code zwar zur Einsicht offen liegt, aber aus Gründen des Copyright nicht vom Anwender verändert, angepasst, oder weitergegeben werden darf.

Somit lässt Open source Software eine Kontrolle zu, die bei closed source Software nicht zu erreichen ist. Doch ist dies zum Teil nur eine passive Kontrolle, wie beim Beispiel der Verschlüsselungssoftware PGP, deren Quellcode zwar der Öffentlichkeit zur Einsicht offen liegt, deren Lizenz es aber untersagt, diesen Code zu verändern und die veränderte Version weiter zu geben. Open Source Software ist nicht notwendigerweise auch freie Software.

Freie Software ist eine Teilmenge der open source Software. Der Quellcode *freier Software* wie der im Sinne der *GNU General public License* (GPL)¹ (→ [FSF91]) liegt auch offen, die Lizenz schränkt aber die Möglichkeiten nicht ein, das Produkt weiter zu verteilen oder Veränderungen am Produkt vorzunehmen, und die veränderte Version zu nutzen oder auch weiter zu verbreiten und so eventuell zur zukünftigen Sicherheit der Software selbst beizutragen. Freie Software Lizenzen lassen es auch zu, dass man freie Software weiterverkauft, solange man den Käufern die gleichen Rechte zugesteht, wie man sie selbst hat.

Kostenlose Software Zur Vermeidung von Mißverständnissen sei hier darauf hingewiesen, dass kostenlose Software nicht notwendigerweise auch zur *freien Software* gehört. Das Verhältnis der Verschiedenen Kategorien ist zur Verdeutlichung in Abbildung 1 skizziert.

¹Es gibt noch viele weitere ähnlich geartete Lizenzen, eine dokumentierte Liste ist unter <http://www.gnu.org/licenses/license-list.html> zu finden.

3 Allgemeine Konstruktionsprinzipien

Ein Grundproblem aller Systeme (und damit auch von Softwaresystemen) ist, dass es äußerst unwahrscheinlich ist, dass sie fehlerfrei sind, wobei jeder Fehler im System die Sicherheit des Systems gefährdet. Eine Möglichkeit, die Wahrscheinlichkeit und Anzahl von Fehlern zu minimieren, ist die Beachtung der Prinzipien, die 1975 von Saltzer und Schröder in [SS75] formuliert wurden. Da die Prinzipien in englischer Sprache formuliert wurden, werden im Folgenden, um Ungenauigkeiten in der Übersetzung zu vermeiden, teilweise ihre ursprünglichen Bezeichnungen verwendet.

***Economy of Mechanism* (minimale Funktionalität)**

Es soll nur soviel Funktionalität zur Verfügung stehen, wie benötigt wird. Dies vereinfacht eine Kontrolle des Systems und steigert die Übersichtlichkeit. Werden die Zugriffskontrollen eines Systems nach diesem Prinzip entworfen, so wird damit nicht die implementierte, sondern die greifbare Funktionalität eingeschränkt, was die Anzahl der Optionen für einzelne Benutzer auf das aufgaben-angemessene Maß beschränkt und Fehlbedienungen vorbeugt. Das Ergebnis ist ein Umsetzung des folgenden Prinzips.

***Least Privilege* (geringstmögliche Rechte)**

Dieses Prinzip hängt stark mit dem der *minimalen Funktionalität* zusammen. Jeder Nutzer eines Systems sollte genau so viele Rechte besitzen, wie er benötigt, um seine Aufgabe zu erfüllen. Die Erfüllung der Aufgabe sei in unserem Beispiel die Bereitstellung der von einem Portalsystem geforderten Dienste.

***Fail-safe Defaults* (explizite Erlaubnis)**

Die Zugriffsrechte werden dermaßen vergeben, dass grundsätzlich jede Nutzung des Systems verboten ist und jede Funktion explizit erlaubt werden muss. Dieses Prinzip kann als sichere Grundregel für die Implementation der *minimalen Funktionalität* durch den Einsatz von Zugriffskontrollen betrachtet werden.

Separation of Privilege (Aufteilung der Rechte)

Ein Sicherungsmechanismus, der von zwei von einander unabhängigen Schlüsseln abhängt ist robuster als einer, der von nur einem abhängt. Durch die Anwendung dieses Prinzips werden Rechte feinkörniger vergeben. Ein Beispiel für dieses Prinzip ist die Benutzung zweier Schlüssel für Bankschließfächer, von denen einer bei der Bank und einer beim Nutzer des Schließfaches verbleibt. Die Auftrennung der Rechte ist das grundlegende Prinzip hinter Nutzer- und Gruppen-basierten Zugriffskontrollen, bei denen der Zugriff auf eine Datei z. B. davon abhängt, dass erstens der Nutzer einer bestimmten Gruppe angehört und zweitens der Eigentümer der Datei den Zugriff für Angehörige der Gruppe freigegeben hat.

Complete Mediation (vollständige Kontrolle)

Die Zugriffskontrolle wird global eingesetzt. D. h. jede Aktion und jeder Zugriff werden auf ihre Zulässigkeit geprüft, bevor sie ausgeführt werden. Dies beinhaltet idealerweise auch die Überprüfung aktueller Zugriffe (z. B. geöffneter Dateien) bei einer Änderung der Rechtevergabe, allerdings ist eine dermaßen umfassende Kontrolle z. B. im Feld der Betriebssysteme nur bei sehr wenigen Systemen implementiert. Dies beinhaltet jedoch keine Kontrolle der behandelten Daten. Die Kontrolle ist allein auf die zugehörigen Berechtigungen bezogen.

Least common Mechanism (minimale gemeinsame Mechanismen)

Je weniger Funktionen sich einen Mechanismus teilen, desto geringer ist die Auswirkung einer Fehlfunktion oder Sicherheitslücke dieses Mechanismus auf das Gesamtsystem. Auf diese Art kann eine eventuelle Beeinträchtigung lokal begrenzt werden.

Die Minimierung der gemeinsamen Mechanismen verringert auch unbeabsichtigten Informationsfluss im System.²

²Dieses Prinzip steht im direkten Gegensatz zur effizienten Programmierung, für die es ein Grundprinzip ist, bereits implementierte Funktionen wiederzuverwenden, wo immer dies möglich ist.

***Psychological Acceptability* (Verständlichkeit und Akzeptanz)**

Die Bedienoberfläche eines Systems sollte so entworfen sein, dass ein Nutzer die Sicherheitsmechanismen möglichst intuitiv auf die richtige Weise bedienen kann. Sie sollte mit dem mentalen Modell, das der Benutzer von der Software aufbaut, möglichst übereinstimmen und erwartungskonform sein. Die Bildung dieses mentalen Modells ist jedoch nur sehr eingeschränkt beeinflussbar. Generell ist es hierbei sinnvoll, sich an etablierte Standards in der Gestaltung von Bedienoberflächen zu halten.³ Man spricht hier auch vom Prinzip der geringstmöglichen Überraschung.

Bei der Implementation vereinfacht die Anwendung der vorgenannten Prinzipien das Erreichen dieses Ziels - allen voran die Anwendung des Prinzips der geringstmöglichen Rechte, welches dazu beiträgt, die Bedienoberfläche und die Funktionalität des Systems auf das notwendige Minimum zu reduzieren und damit die Bedienung zu simplifizieren. Hat ein Nutzer die geringstmöglichen Rechte, so wird durch das Fehlen von nicht-kontextueller Funktionalität die Verständlichkeit des Systems gesteigert, was auch die *gefühlte* Sicherheit des Nutzers erhöht.

***Open Design* (Offener Entwurf)**

Die Sicherheit eines Systems sollte nicht von der Geheimhaltung seiner Sicherheitsmechanismen abhängen. Dies stünde im Konflikt mit der Öffentlichmachung des Systems selbst. Das gegensätzliche Prinzip, dessen Berechtigung weithin angezweifelt wird, ist das der *Security by Obscurity*, auf dem viele aktuelle proprietäre Softwaresysteme basieren, und das aus sicherheitstechnischer Sicht das Prinzip aller closed Source Softwaresysteme ist.⁴ Das Prinzip des offenen Entwurfes ist maßgeblich für aktuelle kryptographische Verfahren. Bestehende Verfahren und Standards, deren Entwurf oder Entwurfskriterien nicht offen liegen, werden hier immer mehr von offenen Verfahren abgelöst. Der Verschlüsselungsstandard *Data Encryption Standard* (DES), z. B., dessen Designkriterien von der US-Amerikanischen *National Securi-*

³Vorausgesetzt, diese sind nachvollziehbar und bedürfen nicht eines sinnvollen Ersatzes, dieses Urteil liegt im Ermessen des Herstellers.

⁴Grassmuck führt hierzu das Beispiel der Telefonkarten der Deutschen Telekom an, deren Sicherheitsgeheimnis durch die millionenfache Verteilung der Karten innerhalb kurzer Zeit entdeckt und von Kartenfälschern ausgenutzt wurde.

ty Agency (NSA) nie veröffentlicht wurden, wurde 1997 durch die Initiative des *National Institute of Standards and Technology* (NIST) durch den *Advanced Encryption Standard* (AES) abgelöst. Ein klar formuliertes Kriterium für die Auswahl des ablösenden Algorithmus war dabei die Transparenz des Entwurfes.

Diese Prinzipien werden im folgenden eine Bewertungs- und Entscheidungsgrundlage bilden. Bei der Analyse von Software und deren Entwicklungsprozess werden wir darauf eingehen, ob die oben beschriebenen Kriterien dabei eine Rolle spielen, oder nach welchen anderen Regeln das jeweilige Objekt entworfen ist. Sollen Entscheidungen über die Nutzung oder die Konfiguration von Systemen getroffen werden, so werden wir die obigen Prinzipien als richtungsgebend betrachten.

4 Netzwerkmodelle

Um einen systematischen Überblick über die technischen Aspekte der Sicherheit im Netzwerk zu ermöglichen, bietet sich das ISO-OSI Schichtenmodell, wie in [PS99] beschrieben, an. Es dient der abstrakten Beschreibung der Verbindung und Kommunikation zwischen Informationssystemen *Open Systems Interconnection* (OSI) und wurde von der *International Standards Organisation* (ISO) zum Standard der Beschreibung von Rechnernetzwerken erhoben.

Dieses Referenzmodell beschreibt sieben Schichten, denen jeweils eine distinkte Rolle in der Kommunikation zwischen Nutzern der Infrastruktur zugewiesen ist. Jede dieser Schichten tritt nur mit den ihr direkt über- oder untergeordneten Schichten in Verbindung, der Nutzer interagiert nur mit der 'obersten' siebten Schicht.

Im Internet, sowie in allen Formen kleinerer Netze wie LANs, und WANs ist jedoch ein Modell implementiert, bei dem die Anzahl der Schichten reduziert, die Interaktionen zwischen den Schichten jedoch komplexer sind, da die Schichten-Hierarchie keine Aussage über das Zugriffsverhalten der Schichten und die möglichen (und praktizierten) Interaktionen zwischen den Schichten macht.

Dieser de-facto-Standard wird in [Dav88] und [Eck03] als das TCP/IP - Referenzmodell beschrieben.⁵ In Abbildung 2 wird deutlich, dass die Funk-

⁵Im Gegensatz zu [Eck03] fasst Davidson in [Dav88] das TCP/IP-Modell als einen Sonderfall des ISO-OSI Modells auf, bei dem die Schichten vier bis sieben zu einer zusammengefasst sind. Dies erhält den Gedanken des ISO-OSI-Modells, ist im Kontext dieser Arbeit jedoch nur aus begrifflicher Sicht interessant.

tionen der Darstellung und Steuerung von den Anwendungen selbst übernommen werden, was zur Entwicklung einer Vielzahl verschiedener Anwendungsprotokolle geführt hat.

Wir werden im weiteren ausschließlich vom TCP/IP Referenzmodell ausgehen, da das ISO-OSI-Modell Schichten beschreibt, die für die Implementation eines WWW-Portals keine Bedeutung haben. Die einzelnen Schichten dieses Referenzmodells werden in Teil II im Hinblick auf mögliche Formen des Angriffs und die Schutzmöglichkeiten davor betrachtet.

ISO-OSI	TCP/IP
Anwendung	Anwendung
Darstellung	⋮
Steuerung	⋮
Transport	Transport
Vermittlung	Netzwerk
Sicherung	Verbindung
..0101001..	

Abbildung 2: Die Referenzmodelle im Vergleich

5 Sicherheit

Bevor wir uns eingehend mit dem Thema der Sicherheit beschäftigen, müssen wir festlegen, worum es sich dabei handelt.

Zum einen gibt es hier die Definitionen, die in gängigen Nachschlagewerken zur Informationstechnik wie z. B. [RP99] zu finden sind und die ein verbreitetes Bild der Sicherheit in der IT anbieten. Hier ist der Begriff der Sicherheit - hier immer auf Information⁶ bezogen - wie folgt definiert:

“Sicherheit ist ein geplantes Ausmaß (Soll-Sicherheit) bzw. vorhandenes Ausmaß (Ist-Sicherheit) an *Vertraulichkeit*, *Integrität*, *Verfügbarkeit* und *Verbindlichkeit*.” [Hei99]

⁶Wir halten uns dabei an das pragmatische Primat der Informationswissenschaften. Wir betrachten auch die Speicherung von Daten als eine Übertragung handlungsrelevanter Daten durch die Zeit.

Diese Definition ist weiter gefasst als jene von Eckert in [Eck03], wo die Verfügbarkeit nicht als Teil der Definition von Sicherheit angesehen wird. Das Wesen von Web-basierten Informationssystemen ist jedoch solcher Art, dass ihre Verfügbarkeit über ihre Existenz oder Nicht-Existenz entscheidet. Offensichtlich hat ein Internet-Portal, das nicht verfügbar ist, keine Präsenz und ohne die Präsenz keine *Existenz* im Internet, wir gehen deshalb im Weiteren von der oben zitierten Definition aus.

Nach dieser Definition sind *Vertraulichkeit*, *Integrität*, *Verbindlichkeit* und *Verfügbarkeit* folgendermaßen definiert:

Vertraulichkeit ist der Zustand, in dem Zugang zu Information nur autorisierten Personen möglich ist. Information soll einem eindeutig definierbaren Empfängerkreis zugeordnet werden können, und einem eindeutig definierbaren Kreis *nicht* zugänglich sein.

Integrität wird dadurch definiert, dass übertragene Information bei der Übertragung nicht verändert wird. Integrität ist dann gegeben, wenn gesandte Information auf dem Weg vom Produzenten zum Rezipienten nicht verändert wird. Dies schließt nicht eine Veränderung des Darstellungsformates aus, da das Wesen der Information im Inhalt, nicht aber in dessen Darreichungsform liegt.⁷

Verbindlichkeit (Auch: Nicht-Abstreitbarkeit) ist dann gegeben, wenn der Produzent und der Rezipient von Information eindeutig identifiziert und der Informationsfluss eindeutig nachvollzogen werden kann. Die Authentizität ist ein Teil der Verbindlichkeit, der beschreibt, dass der Produzent von Information der ist, der er behauptet zu sein.

Verfügbarkeit ist gewährleistet, wenn ein Informationssystem die geforderte Leistung zum geforderten Zeitpunkt und am geforderten Ort liefert. Bei einem Internetportal z. B. ist eine dieser geforderten Leistungen die Auslieferung von Daten an anfragende Systeme.⁸

Diese Definitionen schränken den Begriff der Sicherheit nur zum Teil auf jene Sicherheit ein, die mit rein technischen Mitteln gewährleistet werden kann. Es ist zwar theoretisch möglich, all diese Faktoren tatsächlich zu sichern, doch ist diese Möglichkeit angesichts der Komplexität moderner Informationsmaschinen eine eben theoretische, die unter realistischen Bedingungen kaum wahrgenommen werden kann.

⁷Das Format wird bei der Übertragung über ein elektronisches Netzwerk meist zwangsweise einige Male verändert.

⁸Auch - vor allem bei technischen Systemen: Betriebssicherheit

Die Sicherheit von Information kann auch in einem öffentlichen System durch den Einsatz von kryptographischen Werkzeugen gesichert werden. Der Einsatz dieser Werkzeuge ist jedoch relativ wenig verbreitet und stellt für ein Internetportal keine Lösung dar, da ein Portal, wie oben erläutert, von der Partizipation lebt. Wenn die Teilnahme an der Gestaltung eines Portales für einen Großteil der Nutzer die Installation und Anwendung von krypt(ograph)ischer Software voraussetzt, ist dies eine Hürde, die viele von einer Teilnahme abhalten wird.

Auch hier stößt also die (theoretische) technische Machbarkeit auf praktische Grenzen.

Folglich ist die Sicherheit eines Portalsystems tatsächlich nicht allein durch technische Maßnahmen zu gewährleisten. Die Vertraulichkeit, Authentizität und Integrität der angebotenen Daten ist schon durch die - notwendigen - Möglichkeiten eines Administrators, sie einzusehen oder gar zu verändern, bedroht. Die Notwendigkeit der technischen Administration technischer Systeme, also die Notwendigkeit, ihre Funktion und damit ihre Verfügbarkeit zu sichern, trägt demnach ein Risiko für andere Aspekte der Sicherheit in sich.

6 Der erweiterte Sicherheitsbegriff

Die Definitionen in Abschnitt 5 decken den klassischen Begriff der Sicherheit ab, wie er in der Informationstechnologie gebräuchlich ist, lassen jedoch einen Teilbereich unberührt, der in der Informationswissenschaft definiert wird.

Sicherheit ist in diesem Kontext auch als das Gegenteil einer *Un*-Sicherheit im Umgang mit Informationssystemen zu betrachten. Diese Unsicherheit entsteht aus der Tatsache, dass die Komplexität der Informationssysteme und -Dienstleistungen eine Situation schafft, in der die Anwender und Nutzer von Informationssystemen und -Diensten deren Funktion nicht zur Gänze selbst durchschauen können.

Die Entscheidung, einen Dienst zu nutzen ist also nicht abgesichert, sondern beruht auf einem Vertrauen, das dem System entgegengebracht wird und das nicht oder "nur geringfügig auf Sicherheit verschaffende[m] Wissen" [Kuh99] beruht.

Dieser Unsicherheit im Umgang mit Informationsdiensten sind alle Nutzer ausgesetzt, sie vergrößert sich jedoch mit jeder vermittelnden Stufe, auf der ein Anbieter im Vertrauen auf die von ihm genutzten Dienste selbst Dienste anbietet. Damit führt ein Übergang vom Anbieter zum Nutzer, wie er in Abschnitt 1 eingeführt wurde, zwangsläufig zu einer Vergrößerung der

Unsicherheit und damit zu einem größeren Bedarf für Vertrauen.

Die Frage nach dem Vertrauen in Informationsdienste im Allgemeinen und Software im Besonderen hat dabei eine besondere Eigenart. Des Vertrauens, das weithin den Informationsdiensten - z. B. der Software auf dem Heimischen PC - entgegengebracht wird, sind sich die wenigsten Nutzer bewusst. Es wird in vielen Fällen eher unreflektiert geschenkt, was sich durch die geringe öffentliche Sensibilisierung bezüglich des Themenkomplexes der Sicherheit erklären lässt. Es ist also nicht notwendigerweise so, dass mit jeder Stufe auch die *gefühlte* Unsicherheit zunimmt.

Kuhlen definiert den Begriff der informationellen Autonomie in [Kuh02] nicht nur als die Fähigkeit, selbst auf benötigte informationelle Ressourcen zuzugreifen, sondern auch als die Fähigkeit, “ *diese [Informations-] Arbeit bewusst und kontrolliert an entsprechende Ressourcen zu delegieren*”.⁹

Damit ist informationelle Selbstbestimmung ein existenzieller Bestandteil der Sicherheit im Umgang mit Informationsmaschinen, mit der wir uns hier auseinander setzen.

Wie kann nun diese Form der Sicherheit für die Betreiber und die Besucher eines Portals gesichert werden?

Die einem Portalsystem zugrunde liegende Software ist als eine Ressource zu betrachten, an die im Idealfall Informationsarbeit bewusst und kontrolliert delegiert werden kann. Ebenso ist aber das Serversystem, das den Portaldienst zur Verfügung stellt eine solche Ressource. Es ist also unabdingbar, dass auch die Delegation der Informationsarbeit an dieses System bewusst und kontrolliert geschieht.

Wie wir festgestellt haben, entzieht closed Source Software sich effektiv nahezu gänzlich der Kontrolle, während proprietäre Open Source Software nur eine passive Form der Kontrolle zulässt. Somit kann informationelle Selbstbestimmung durch den Einsatz proprietärer Software nur begrenzt erreicht werden.

⁹Nach dieser Definition und in Anbetracht der Verbreitung des oben geschilderten Nutzerverhaltens ist die informationelle Autonomie vieler Nutzer zu einem hohen Grade eingeschränkt!

Teil II

Technische Sicherheit

Die technische Sicherheit eines Systems lässt sich unterteilen in die sichernden Eigenschaften der Hardware und die sicherheitsrelevanten Eigenschaften der Software, doch zunächst stellt sich die Frage, welche Komponenten für den Aufbau eines Internet-Portalsystems notwendig sind. Die Beantwortung dieser Frage hängt offensichtlich vom geplanten Einsatz des Systems ab. Wir werden hier von einem reinen WWW-Portal ausgehen, das außer dem Auftritt im World-Wide-Web keine Dienste anbietet.

Ganz offensichtlich ist ein Zugang zum Internet notwendig. Um Diesen nutzen zu können, bedarf es zusätzlich eines Rechnersystems, das ausreichend Speicher- und Rechenkapazität zur Verfügung stellt, um ein Portal darauf zu betreiben. Wie viel dabei ausreichend ist, hängt stark von der Nutzung des Portals ab; es ist also auch die Skalierbarkeit des Systems zu beachten.

Den letzten Schritt der Grund-Anforderungen macht ein Betriebssystem, das eine effiziente Datenverwaltung ermöglicht und es erlaubt, einen Webserver zu betreiben.

Wie von Holupirek in [Hol03] erläutert wird, kennzeichnen ein Portal nicht so sehr die technischen Gegebenheiten, als vielmehr die Bildung und die Aktivitäten einer *Community*, also einer Gruppe von Benutzern, die sich aktiv an der Gestaltung der Inhalte der Seite beteiligt. Diese Bildung einer Community bedarf der Möglichkeit, eigene Inhalte auf dem Portal zu veröffentlichen, was für den Einsatz eines *Web Content Management Systems* (WCMS) spricht.

- Internetzugang
- Hardware, i. e. Serversystem mit ausreichend (skalierbarer) Speicher- und Rechenkapazität
- Betriebssystem und Zusatzsoftware

Mit diesem Anforderungskatalog kann nun die Betrachtung des Systems in Angriff genommen werden. In den folgenden Abschnitten werden wir zunächst versuchen, festzustellen, welchen Gefährdungen ein solches System potenziell ausgesetzt ist.

7 Ermittlung des Schutzbedarfs

Bevor die Planung der Sicherung eines Serversystems in Angriff genommen wird, ist es sinnvoll, sich ein Bild der möglichen Gefährdungen zu machen, gegen die das System zu schützen ist. Zu diesem Zweck beschreibt das *Bundesministerium für Sicherheit in der Informationstechnik* (BSI) im *Grundschutzhandbuch* (GSH) [BSI01] detailliert Gefährdungen, Analyseverfahren und Maßnahmen im Zusammenhang mit seinem Beschäftigungsgebiet, der Sicherheit von IT-Systemen.

Gefährdungen

Eine primäre Zielgruppe des Grundschutzhandbuchs des BSI sind die Informationssysteme der öffentlichen Verwaltung, also ein recht spezieller Teilbereich aller möglichen Systeme, für die die Leistungen des BSI noch weiter gehen, als bis zur Bereitstellung des Grundschutzhandbuchs. Doch bietet das Grundschutzhandbuch Anhaltspunkte für systematische Überlegungen über die Gefährdung und den Schutz auch für die Systeme von NGOs wie dem Verein Nethics.

Während es den Umfang dieser Arbeit sprengen würde, den vollen Sicherheitszyklus des GSH zu durchlaufen, so soll doch die Schutzbedarfsanalyse und der Gefährdungskatalog des GSH als Anhaltspunkt bei der Vorbereitung des Systems dienen, um eine Gewichtung der Maßnahmen vornehmen zu können.

Nach dem Grundschutzhandbuch des BSI Gefährdungen sind Informationssysteme Gefährdungen in den folgenden Kategorien ausgesetzt. Die einzelnen Kategorien sind jeweils durch Beispiele illustriert. Die genauen Beschreibungen sind beim Internetauftritt¹⁰ des BSI einzusehen.

Höhere Gewalt *Unzulässige Temperatur und Luftfeuchte, Personalausfall, Datenverlust durch starke Magnetfelder oder Beeinträchtigungen durch Großveranstaltungen,...*

Technisches Versagen *Spannungsschwankungen oder Ausfall der Stromversorgung, schlechte oder fehlende Authentikation,...*

Organisatorische Mängel *Fehlende oder unzureichende Regelungen oder unzureichende Kenntnis über Regelungen, unberechtigte Sammlung personenbezogener Daten, ungeeignete Verwaltung von Zugangs- und Zugriffsrechten,...*

¹⁰<http://www.bsi.bund.de/gshb/deutsch/etc/inhalt.htm>

Menschliche Fehlhandlungen *Vertraulichkeits-/Integritätsverlust von Daten durch Fehlverhalten der IT-Benutzer, speichern von Passwörtern unter Windows for Workgroups und Windows 95, Sorglosigkeit im Umgang mit Informationen, ungeeigneter Umgang mit Passwörtern,...*

Vorsätzliche Handlungen *Manipulation/Zerstörung von IT-Geräten oder Zubehör, trojanische Pferde, Mitlesen von E-Mails, Nichtanerkennung einer Nachricht,...*

Vorsätzliche Handlungen sind Angriffe, die hier noch genauer betrachtet werden sollen. Zunächst lässt sich unterscheiden, wo die Quelle eines Angriffes liegt. Es wird hier zwischen Angriffen von innen und Angriffen von außen unterschieden:

Angriffe von innen geschehen durch Mitarbeiter oder eine sonstige vertraute Personen - ein ausgesprochen häufiges Szenario. Diese Angriffe gehen von der Sabotage der Hardware am Arbeitsplatz bis hin zum Datendiebstahl oder der System-weiten Infizierung der Daten durch Viren, Würmer oder Trojanische Pferde.

Angriffe von außen decken ein weites Feld von Szenarien ab. Das eine Ende dieses Spektrums bilden nicht zielgerichtete Attacken durch technisch nicht bis wenig versierte Benutzer, die im Internet verfügbare *Exploits* (Programme, die bekannte Sicherheitslücken ausnutzen) z. T. vollkommen unreflektiert oder aus reinem Spieltrieb benutzen, so genannte *Script-Kiddies*. Das andere Ende des Angreiferspektrums bilden professionelle *Cracker*, die oft über herausragende technische Fähigkeiten verfügen.

Weiterhin können Angriffe anhand des Einflusses, den sie auf die kommunizierten Daten haben und damit anhand der Sicherheitskriterien, die sie im Kommunikationsprozess verletzen, unterschieden werden.

Passive Angriffe (*Lauschen*) Der Angreifer macht sich Daten im Informationsfluss zugänglich, verändert dies aber nicht. Die Vertraulichkeit der Kommunikation ist eingeschränkt, nicht aber Integrität, Verfügbarkeit oder Verbindlichkeit.

Aktive Angriffe Der Angreifer greift aktiv in den Kommunikationsprozess ein. Er verändert die Kommunikation, indem er im Kommunikationsprozess Information entfernt, verändert oder hinzufügt. Aktive Angriffe können alle Facetten der Sicherheit verletzen.

Einordnen des Systems

Das BSI hat neben der Beschreibung möglicher Gefährdungen auch Beispielszenarien formuliert, anhand deren Zutreffen oder Nicht-Zutreffen eine qualitative Einordnung in eine der Schutzbedarfsklassen *niedrig bis mittel, hoch* oder *sehr hoch* vorgenommen werden kann.

Zur Einteilung werden die Schäden und Folgeschäden aus Anwendersicht durch den Verlust von Vertraulichkeit, Integrität oder Verfügbarkeit in sechs Kategorien betrachtet. Die zu betrachtenden Schäden sind dabei sowohl die materieller, als auch ideeller Art.

1. Verstoß gegen Gesetze/Vorschriften/Verträge
2. Beeinträchtigung des informationellen Selbstbestimmungsrechts
3. Beeinträchtigung der persönlichen Unversehrtheit
4. Beeinträchtigung der Aufgabenerfüllung
5. Negative Außenwirkung
6. Finanzielle Auswirkungen

Die Beispielszenarien für die drei Schutzbedarfsklassen, anhand derer die folgende Einteilung erfolgt, sind in Anhang B aufgeführt.

Anhand des Grundschutzhandbuchs soll die Schutzbedarfskategorie für das zu erstellende System ermittelt werden.

Dazu wird das System aus dem Blickwinkel der sechs Schadensszenarien betrachtet und die zu erwartenden Schäden des Verlusts der Sicherheitsaspekte Vertraulichkeit, Integrität und Verfügbarkeit ermittelt werden. Diese Analyse basiert darauf, dass die Frage *“was wäre, wenn. . .”* gestellt wird. Die Aussagen, die danach getroffen werden können sind also qualitativer Art.

Zunächst wird betrachtet, welcher rechtliche Schaden in Form von Vertragsverletzungen, oder dem Verstoß gegen gültiges Recht den Nutzern eines Systems durch die Verletzung der verschiedenen Teilbereiche entstände.

Verstoß gegen Gesetze/Vorschriften/Verträge

- Verlust der Vertraulichkeit: Es gibt keine gesetzlichen Auflagen, die die Vertraulichkeit der Daten auf dem Portal verlangen. Es gibt jedoch die selbst auferlegte Verpflichtung, personenbezogene Daten nicht an Dritte weiter zu geben. Eine Verletzung dieser Pflicht kann zu einem Vertrauensverlust der Nutzer führen und sollte verhindert werden.

- Verlust der Integrität: Ein Verlust der Integrität der Daten auf Nethics hätte keine absehbaren rechtlichen Folgen.
- Verlust der Verfügbarkeit: Nethics ist von keiner Seite her verpflichtet, seine Daten verfügbar zu halten. Eine Verletzung der Verfügbarkeit bliebe somit unter diesem Gesichtspunkt folgenlos.

Beeinträchtigung des informationellen Selbstbestimmungsrechts

- Verlust der Vertraulichkeit: Die Nethics zur Verfügung stehenden Daten, deren Vertraulichkeit garantiert ist, sind die E-Mail-Adressen der Nutzer. Sie lassen aller Wahrscheinlichkeit kaum tief greifende Rückschlüsse auf die Nutzer zu. Der schwerste zu erwartende Missbrauch dieser Daten ist die unzulässige Zusendung von Werbe-E-Mail.
- Verlust der Integrität: Sollten die auf Nethics gespeicherten Daten verfälscht oder manipuliert werden, ist mit einer Einschränkung der informationellen Selbstbestimmung der Nutzer nicht zu rechnen. (Die Verfälschung dieser Daten hätte andere Auswirkungen.)
- Verlust der Verfügbarkeit: Durch die Natur der in Frage kommenden Daten ist auch hier großer Schaden ausgeschlossen.

Beeinträchtigung der persönlichen Unversehrtheit

Nethics ist eine rein informationelle Dienstleistung. Der Themenbereich umfasst keinerlei Daten, von denen anzunehmen ist, dass sie mit der persönlichen Unversehrtheit der Nutzer zusammenhängen.

- Verlust der Vertraulichkeit: Eine physische oder psychische Schädigung von Personen durch den Verlust der Vertraulichkeit ist nur schwer denkbar
- Verlust der Integrität: Die Daten und Abläufe auf Nethics stehen in keinem Zusammenhang mit dem Wohlergehen der Nutzer. Eine Verfälschung oder Manipulation hätte keine gesundheitlichen Folgen.
- Verlust der Verfügbarkeit: Auch das Ausbleiben der Verfügbarkeit ist nicht mit dem Wohlergehen eines Beteiligten zu verbinden.

Beeinträchtigung der Aufgabenerfüllung

- Verlust der Vertraulichkeit: Die Aufgaben, die auf Nethics erfüllt werden, sind nicht von der Vertraulichkeit der behandelten Daten abhängig. Das Portal dient eher der Veröffentlichung.
- Verlust der Integrität: die Integrität der Daten muss gewährleistet sein, damit Nethics seine Funktion als Portalsystem erfüllen kann. Es entsteht jedoch kaum materieller oder ideeller Schaden, wenn diese Funktion mit verfälschten Daten wahrgenommen wird. Eine fälschliche Zuordnung von Daten zu Personen würde vermutlich kaum Schaden verursachen¹¹.
- Verlust der Verfügbarkeit: Das Portalsystem ist die öffentliche Vertretung des Vereins Nethics. Das heißt ein Ausfall des Portals hätte für den Verein einen Ausfall des Auftritts zur Folge.

Negative Außenwirkung

- Verlust der Vertraulichkeit: Sollten die wenigen vertraulichen Daten im System ihre Vertraulichkeit verlieren und dieser Zustand öffentlich bekannt werden, so wäre ein genereller Vertrauensverlust zu erwarten, da der Nutzer bei der Registrierung zugesichert bekommt, seine Daten würden nicht an dritte weitergegeben. Vermutlich würden Besucher eher davon absehen, sich bei dem Portal zu registrieren.
- Verlust der Integrität: Eine Verfälschung der Inhalte des Systems könnte kurzzeitig zu Ansehensverlusten führen. Sie wäre aber angesichts der momentanen Benutzerzahl relativ leicht zu erkennen und zu beheben.
- Verlust der Verfügbarkeit: Die Folgen eines Verlustes der Verfügbarkeit des Systems hätte kaum Vertrauenseinbußen zur Folge. Ein Ausfall von 24 Stunden wäre jedoch grenzwertig und nicht als tolerabel einzuschätzen.

Finanzielle Auswirkungen

- Verlust der Vertraulichkeit: Beim Verlust der Vertraulichkeit der auf dem System gespeicherten Daten ist nicht mit Regressforderungen zu

¹¹Dieser Punkt muss jedoch neu überdacht werden, sollte Nethics zu einem zentralen Informationsknoten werden.

rechnen. Die Menge der als vertraulich behandelten E-Mail Adressen macht einen Versuch der wirtschaftlichen Nutzung von außen unwahrscheinlich.

- Verlust der Integrität: Eine Manipulation der veröffentlichten Daten hätte vermutlich kaum finanzielle Folgen. Die verbreiteten Inhalte sind kaum von finanzieller Relevanz.
- Verlust der Verfügbarkeit: Die Institution Nethics verfolgt keine finanziellen Interessen, es sind keine existenzbedrohenden Geldmittel im Einsatz.

Betrachtet man die Bedeutung des Systems im Gesamtzusammenhang der Institution Nethics e. V. , so wird erkennbar, dass das Portalsystem für den Verein Nethics ersatzlos ist und deshalb eine gute Sicherung der dauerhaften Verfügbarkeit des Systems von Nöten ist.

Alle diese Fragen und Antworten bedürfen einer wiederholten, regelmäßigen Kontrolle, da sie bei einer eventuellen Änderung der Betreiberpolitik oder einem Veränderten öffentlichen Status des Portals eventuell anders beantwortet und die Maßnahmen gegebenenfalls entsprechend angepasst werden müssen.

Vergleicht man nun die die Aussagen dieser Analyse mit den vom BSI vorgelegten Beispielszenarien in Anhang B, so ergibt sich ein Bild, das in den meisten Facetten einen geringen bis mittleren Schutzbedarf, im Gegensatz zur Außenwirkung, deren Analyse einen hohen Schutzbedarf nahe legt. Die Zusammenfassung der Aussagen ist in Tabelle 1 aufgeführt.

8 Hardware

Zunächst wollen wir feststellen, welche Anforderungen die Sicherung eines Internet-Portalsystems an die Hardware dieses Systems stellt.

Bei der Betrachtung der Hardwareseite wollen wir vor allem die Sicherung der Verfügbarkeit betrachten. Die Vertraulichkeit der Daten - die der auf physischen Datenträgern gespeicherten, wie auch die der Übertragung von und zum Server - ist zum größten Teil ein Softwareproblem. Hardwareseitig sind hier nur Maßnahmen zum Diebstahlschutz und die Sicherung der Übertragungsmedien, wie sie weiter unten noch erläutert wird, relevant.

Für alle modernen Betriebssysteme wird von deren Entwicklern versucht, eine größtmögliche Integrität der gespeicherten Daten zu gewährleisten, um

Szenario	Schaden und Folgen
Verstoß gegen Gesetze, Vorschriften, Verträge	Verstöße gegen Vorschriften und Gesetze haben nur geringfügige Konsequenzen. Es kommt zu geringfügigen Vertragsverletzungen mit nur geringen Konventionalstrafen.
Beeinträchtigung des informationellen Selbstbestimmungsrechts	Ein möglicher Missbrauch personenbezogener Daten hat nur geringfügige Auswirkungen auf die gesellschaftliche Stellung oder die wirtschaftlichen Verhältnisse des Betroffenen.
Beeinträchtigung der persönlichen Unversehrtheit	Eine Beeinträchtigung der persönlichen Unversehrtheit erscheint nicht möglich.
Beeinträchtigung der Aufgabenerfüllung	Die Beeinträchtigung der Aufgabenerfüllung würde von den Betroffenen für kurze Zeit als tolerabel eingeschätzt werden. Die maximal tolerierbare Ausfallzeit ist nicht größer als 24 Stunden.
Negative Außenwirkung	Eine breite Ansehens- oder Vertrauensbeeinträchtigung ist zu erwarten.
Finanzielle Auswirkungen	Der finanzielle Schaden bleibt für die Institution tolerabel.

Tabelle 1: Schutzbedarfsfeststellung nach dem BSI Grundschutzhandbuch

die Funktionsfähigkeit der Systeme zu sichern, die ja ebenfalls von den Datenträgern abhängen.¹² Alle verbreiteten modernen Betriebs- und Backupsysteme

¹²Es ist dies jedoch keine Garantie des Herstellers.

me sind mit integritätssichernden Technologien implementiert. Damit wird dieser Faktor im wesentlichen durch die Wahl der Software bestimmt, weshalb diese Problematik hier ausgeklammert wird.

Die Verfügbarkeit von Daten ist in zweierlei Hinsicht interessant: Auf der einen Seite gibt es hier eine Verfügbarkeit im laufenden Betrieb, die bei stark geforderten Systemen auch mit der Geschwindigkeit des Datenflusses zusammen hängt. Der bereits erläuterte Zusammenhang zwischen der Verfügbarkeit und der Existenz Web-basierter Dienste führt dazu, dass gerade die Serversysteme von Anbietern solcher Dienste in hohem Maße von dieser Verfügbarkeit abhängig sind.

Auf der anderen Seite kann der Begriff der Verfügbarkeit auch allgemeiner als die Möglichkeit des Zugriffs auf gespeicherte Daten betrachtet werden - in diesem Fall des Zugriffs auf die Allgemeinheit der Daten, die dem Portalsystem verfügbar gemacht wurden. *Diese* Verfügbarkeit wiederum kann noch feiner gekörnt werden, indem man zwischen der Archivierung und der Sicherung (im Sinne eines Backups) von Daten unterscheidet.

Um die Verfügbarkeit von Datensicherungen zu gewährleisten, ist zum einen eine geeignete Backup-Strategie zu wählen, zum anderen ist es sinnvoll, für eine räumliche Trennung des laufenden Systems zu sorgen. Die Lagerung der Backup-Medien (so die Entscheidung auf tragbare Medien gefallen ist) ist mit den selben Überlegungen verbunden, wie die über den Standort des Systems, die weiter unten noch erläutert werden.

Die Verfügbarkeit im laufenden System ist durch den Einsatz von *Redundant Array of inexpensive Disks* (RAID)¹³ - Systemen zuverlässig zu erreichen. Es handelt sich dabei um eine logische Speichereinheit, die sich aus mehreren Festplatten oder vergleichbaren Medien zusammensetzt und Redundanz wie auch hohen Datendurchsatz gewährleistet.

Ein weiterer Aspekt der Verfügbarkeit ist die Verfügbarkeit des Gesamtsystems. Diesem Aspekt wird offensichtlich durch Redundante Systeme genüge getan, die den Ausfall eines oder mehrerer Serversysteme kompensieren können. Da oft diese Redundanz auch genutzt wird, um die Verfügbarkeit im Sinne des Datenflusses zu vergrößern, ist ein Verlust eines Serversystems meist mit Einbußen in der Performance des Gesamtsystems verbunden. Ein weiterer Schritt in dieser Richtung ist die Trennung von Webserver, Datenbank und Dateiserver, die das Konstruktionsprinzip des *least common Mechanism* umsetzt. Ein eventueller Angreifer muss, um ein effektiv getrenntes System zu kontrollieren die Kontrolle über alle Systeme gewinnen, um die Funktion des Gesamtsystems zu beeinflussen und stößt dabei auf den Effekt der Rechte-Aufteilung, die hier vorgenommen wurde. Wird aber nur die Kon-

¹³Auch: *Redundant Array of independent Disks*

trolle über einen Teil der Subsysteme erlangt, so kann zwar die Sicherheit eines Teils der Daten, nicht aber die der Gesamtdaten verletzt werden. Es ist dies eine schadensminimierende Maßnahme, die nicht generell schützt, sondern die Folgen eines Bruches der Schutzmaßnahmen verringert.¹⁴

Die Archivierung von Daten ist eher aus rechtlicher Hinsicht, z. B. zur Beweissicherung interessant. Für den laufenden Betrieb eines Serversystems ist sie aber nicht unbedingt notwendig, da es meist die aktuellen Daten sind, die für die Nutzer den Wert eines Portals ausmachen - es sei denn, die Archivierung ist Teil der angebotenen Dienstleistung und ein Ziel des Portals. Eine Einführung zu Backup-Strategien und Archivierung findet sich u. a. in [SV03].

Die Authentizität von Daten auf der Hardware-Ebene zu kontrollieren ist gerade zum Zeitpunkt der Entstehung dieser Arbeit das Thema vieler Diskussionen. Es wird sich herausstellen, wie weit sich das Konzept des *Digital Rights Management* (DRM)¹⁵ verbreiten und die *trusted Computer Platform Alliance* (TCPA) die Entwicklung des Digitalen Informationsraumes beeinflussen wird. Eine Bearbeitung dieser Thematik ist in [Hol02] zu finden, weiterführende Literatur u. a. in [Sch02] oder auf der offiziellen Website der TCPA [TCPA02].

Physischer Zugriff Der für Angriffe empfindlichste Punkt der technischen Sicherheit ist der physische Zugriff auf das System selbst. Generell muss davon ausgegangen werden, dass Angreifer mit physischem Zugang zum System jede Form von Zugriff bekommen können. Deshalb sollte der Zugang zum Standort sicherheitskritischer Informationssysteme kontrolliert sein.

Nachdem der Schutzbedarf unseres Systems umrissen werden konnte, zeigt sich, dass ein handelsüblicher Heim-PC als Basis des Portalsystems die gestellten Anforderungen an die Datensicherheit auf Hardware-Ebene erfüllt. Dies ist nicht überraschend, da es ja die weitreichende Verfügbarkeit leistungsfähiger Rechnersysteme ist, der sich dieses Papier zu einem Teil widmet. Die Rechen- und Speicherkapazität, die zum Betreiben der ausgewählten Serversoftware notwendig ist, wird erfahrungsgemäß von jedem aktuellen System erbracht.¹⁶

¹⁴Eine eingehende Bearbeitung dieser Thematik ist in [ano00] zu finden.

¹⁵Auch: *Digital Restrictions Management* [Sta02]

¹⁶Im Zuge des Projektpraktikums, das der Entstehung dieser Arbeit voraus ging, wurde ein Beispielsystem auf einem PC der Pentium-II-Klasse realisiert.

9 Das Netzwerk

Zur Betrachtung des Netzwerkes werden wir die Schichten des TCP/IP-Referenzmodells betrachten und auf Gefährdungen und Schutzmöglichkeiten untersuchen. Dabei werden die Schichten des Referenzmodells von der Verbindungsschicht an aufeinander aufbauend betrachtet.

Die Verbindungsschicht

Um die Sicherheit der Internet-Anbindung des Portals zu analysieren, sollte man sich Klarheit über die Übertragungsmedien verschaffen, die bei der Verbindung zum *Internet Service Provider* (ISP) genutzt werden.

Kupferkabel Dies ist die am häufigsten anzutreffende Form der Vernetzung von Computersystemen, die in verschiedenen Fällen Anwendung findet. Von der Telefonleitung eines Nutzers zu Hause zu dessen *Internet Service Provider* (ISP) bis hin zur hoch leistungsfähigen¹⁷ Verkabelung unter Verwendung mehrerer Stränge verdrellter Doppel-Adern.

Koaxialkabel Ältere Netzwerkstandards (wie *thin-Ethernet*), oder Datenverbindungen über das Kabelnetz. Diese spielen bei LANs und WANs kaum mehr eine Rolle, werden aber regional zur Anbindung an das Internet über die öffentlichen Breitbandkabel genutzt. (Dies vor allem in Nordamerika)

Glasfaserleitungen werden in Hochgeschwindigkeits-Netzwerken eingesetzt, da sie eine äußerst hohe Bandbreite zur Verfügung stellen können. vereinzelt sind auch Haushalte mit Glasfaserleitungen an ihren Internet Service Provider angebunden. Dies ist die z. Zt. sicherste Form der (Roh-)Datenübertragung, da hier nur eine vernachlässigbare elektromagnetische Strahlung erzeugt wird, die bei anderen Übertragungsmedien (bei räumlichem Zugang) relativ leicht abgehört werden kann.¹⁸

Funkverbindungen Über Satelliten- oder Richtfunk-Anlagen oder Kurzstrecken-Funktechnologien wie *wireless local Area Network* (WLAN)s oder der relativ neue Standard Bluetooth. Diese Netzwerktechnologie erfreut sich immer größerer Beliebtheit, ist jedoch die anfälligste für

¹⁷... mit Übertragungsgeschwindigkeiten von bis zu einem Gigabit pro Sekunde

¹⁸das zur Übertragung genutzte Licht ist selbst eine Elektromagnetische Strahlung, die aber außerhalb des Leiters als nicht sinnvoll messbar betrachtet werden kann.

Angriffe von außen, da hier eine Sende- oder Empfangsstation in Reichweite des Netzwerks den Datenverkehr abhören oder auch Daten in das Netz einspeisen kann.

Bis auf die lichtgestützte Übermittlung bei Glasfaserleitungen (und lasergestützten Systemen, die hier aufgrund ihrer relativ geringen Verbreitung aber nicht berücksichtigt werden sollen) sind all diese Verfahren anfällig für Abhörattacken, die die Elektromagnetische Abstrahlung der Übertragungsmedien nutzen. Diese Art von Lauschangriffen sind mit technisch relativ einfachen Mitteln zu realisieren und machen einem potentiellen Angreifer alle übertragenen Daten zugänglich. Die Manipulation auf diesem Wege ist dagegen als schwierig zu betrachten. Um solchen Angriffen vorzubeugen, sollten Datenleitungen soweit wie möglich nicht öffentlich zugänglich sein.

Soweit keine Funk-gestützte Verbindung betroffen ist, ist mit der physischen Abschottung des Serversystems und der Verkabelung somit die Sicherheit der beiden Schichten vor einem Angriff von außen gegeben.

Bei der Datenübertragung per Funk kann eine physische Abschottung nur sehr schwer erreicht werden, was Funknetze besonders empfindlich gegenüber Lauschangriffen macht. Hier kann die Sicherheit der Bit-Übertragungsschicht nur sehr eingeschränkt gewährleistet werden. Die Sicherung von Information muss also soweit wie möglich in einer höheren Schicht erfolgen. Damit wird jedoch eine Sicherung derjenigen Daten ausgeschlossen, die für die Funktion der Verbindungsschicht benötigt werden. Die potenziell auch für Angreifer interessante Adressierung von Datenpaketen - sowohl Host- und auch Port-Nummer - bleiben dabei also unverschlüsselt.

Einen Sonderfall stellen Sicherungsprotokolle dar, die vor allem für die Verbindung zweier Systeme - im Gegensatz zur Vernetzung mehrerer - genutzt werden. Die wichtigsten Vertreter dieser Art sind das *Peer-to-Peer-Protocol* (PPP) und sein naher Verwandter, das *peer-to-peer-Protocol over Ethernet* (PPPoE), das eine direkte Verbindung zweier Rechner in einem Netzwerk erlaubt.¹⁹ Diese Protokolle werden bei weitaus den meisten Verbindungen über Telefonleitungen genutzt, wie sie bei der Verbindung von Privathaushalten an deren ISP genutzt werden, verbreitet sind z. B. ISDN²⁰ oder SDSL/ADSL²¹.

Da in den meisten Fällen eine Authentifikation zwischen den Partnern einer PPP-Verbindung gewünscht ist, kann hier schon ein Angriff auf die Vertraulichkeit sensibler Authentifikationsdaten oder die Verfügbarkeit der

¹⁹Tatsächlich setzt PPPoE schon auf einem anderen Verbindungsschicht-Protokoll auf

²⁰integrated Services Digital Network

²¹Symmetric, bzw. asymmetric Digital Subscriber Line

Verbindung geschehen, wenn ein Zugriff auf die Datenleitungen besteht. Deshalb wurden für diese Ebene der Kommunikation kryptographische Maßnahmen eingeführt, die das Abhören sensitiver Daten verhindern sollen. Wichtig ist hierbei vor allem das *Challenge Handshake Protocol* (CHAP), das eine relativ sichere Alternative zur unverschlüsselten Authentifikation durch das *Password Authentication Protocol* (PAP) darstellt.

Die Netzwerkschicht

Aufbauend auf dem *Address Resolution Protocol* (ARP), das es ermöglicht, die Logische Adresse eines Rechners im Netz anhand einer einmaligen Identifikationsnummer der zugehörigen Netzwerkkarte zu finden, wird vor allen Dingen das *Internet Protocol* (IP) des TCP/IP-Referenzmodells genutzt, um die Verbindungen zwischen beteiligten Rechnern herzustellen. Das *Internet Protocol* ermöglicht eine eindeutige logische Adressierung von Systemen im Netzwerk und bildet so die Grundlage für gezielte Kommunikation. Eine Verfälschung dieser Adressen ist einfach zu realisieren, da IP keine Sicherungsmaßnahmen wie Konsistenzprüfung der Pakete vorsieht.

Eine Sicherung der Netzwerkschicht wird durch das Protokoll *IP secure* (IPSec) gewährleistet, das die Integrität und Vertraulichkeit der Informationen zwischen jeweils zwei Kommunikationspartnern (*Peers*) auf dieser Ebene sichern kann. Dieses Protokoll wird in der nächsten Version von IP (IPv6) integriert sein. Interessant ist diese Sicherung bei funkgestützten Datenübertragungen. In unserem Modell wird sie jedoch keine Rolle spielen, da das zu implementierende System keine drahtlosen Verbindungen zu clients nutzen und die Abhörsicherheit des Internetzugangs gesichert sein wird.

Werden Datenpakete bei Verwendung von IP mit einer falschen Absenderadresse versehen, wie im obigen Beispiel beschrieben, so spricht man von *Maskierung*. Die Verfälschung der Absenderadresse kann nicht nur dazu genutzt werden, Rechnersysteme von einer sinnvollen Interaktion mit dem Netz abzuhalten, sondern auch um unautorisierten Zugriff zu Systemen zu erlangen (*IP-Spoofing*).

Die Veränderung von Empfängeradressen entspricht einer Umleitung der Pakete an einen anderen Rechner, dies ist dann missbräuchlich, wenn Die Paketadressen von einem Netzknoten auf ihrem Weg zum Empfänger verändert werden. Dies ist entweder ein Angriff auf die Verfügbarkeit (wenn die umgeleiteten Pakete ihren Bestimmungsort nicht erreichen), die Vertraulichkeit (wenn die Pakete über ein "lauschendes" System umgeleitet werden) oder die Integrität der gesandten Daten (wenn die Pakete auf ein System umgeleitet werden, das den Inhalt der Pakete verändert und sie dann zum Zielsystem

weiter sendet).

Ebenfalls als dieser Ebene zugehörig ist das *Internet Control Message Protocol* (ICMP), es ist vor allem wegen *Denial of Service* (DOS)-Attacken interessant, da es genutzt werden kann, um Verbindungen zwischen Rechnern zu stören, oder zu übernehmen (*Session Hijacking*). Eine mögliche Form des Angriffs über ICMP ist zum Beispiel das Versenden von so genannten *Ping Requests* an viele Rechner im Internet. Die versandten Pings tragen aber als Absender die Adresse des Opfers, das daraufhin von *ACK*²² - Meldungen, die als Antwort auf das *Ping* gesandt werden überflutet und handlungsunfähig gemacht wird.

Die Transportschicht

Je höher wir in der Hierarchie des Referenzmodells steigen, umso detaillierter wird die Information, die darin transportiert wird,

Die wichtigen Protokolle der Transportschicht sind das *Transmission Control Protocol* (TCP)²³ und das *User Datagram Protocol*. Sie lassen eine Adressierung bestimmter Dienste eines Rechnersystems über so genannte Port-Nummern zu.

UDP ist ein *verbindungsloses* oder *unzuverlässiges* Protokoll, das unbenutzte Veränderungen an einem Datenstrom sehr leicht macht, da es keine Validitäts- oder Integritätsprüfung der Daten vorsieht, und der Empfang von Paketen vom Empfänger nicht quittiert wird. Es spielt vor allem bei Echtzeit-Anwendungen wie der Netzwerktelefonie (Voice over IP) oder Videoübertragung eine Rolle. TCP ist ein so genanntes *zuverlässiges* Protokoll, das Konsistenzprüfungen und Quittierung von Paketen vorsieht. Eine Manipulation von Daten auf dem Übertragungsweg ist deshalb erschwert.

SSL

Die *secure Sockets Layer* (SSL) ist eine Sicherheitsschicht, die auf einem Verbindungsorientierten Transportprotokoll wie TCP aufsetzt. Ab dieser Ebene kann Verschlüsselung effizient eingesetzt werden, um die zu übertragende Information zu schützen. Während es kaum sinnvoll ist, auf den darunter liegenden Ebenen z. B. die Adressen von Paketen zu verschlüsseln - da auf diese Weise die Zustellung der Pakete unmöglich gemacht wird - ermöglicht

²²Auch: *Pong*

²³auch: Transport Control Protocol

der Einsatz von SSL die Gewährleistung von Vertraulichkeit und Authentizität von Information.

Die Authentizität der Kommunikationspartner kann dadurch gesichert werden, dass Client und Server beim *SSL-Handshake* Zertifikate austauschen, die - wenn Sie von einer Vertrauten Instanz wie z. B. einer zentralen Zertifikationsbehörde ausgestellt wurden - die Identität des Partners belegen. Zumeist wird SSL jedoch nur dazu eingesetzt, die Vertraulichkeit der Kommunikation zu gewähren. Die Kommunizierenden einigen sich dazu beim *Handshake* auf einen geheimen Schlüssel, den *Session Key*, der während der Kommunikation zur Verschlüsselung der transportierten Daten benutzt wird.

An dieser Stelle wird auch deutlich, dass SSL nicht auf einem verbindungslosen Protokoll wie UDP aufsetzen kann. UDP ist ein verbindungsloses Protokoll, das keine Unterscheidung zwischen verschiedenen *Sessions* zulässt.

Abbildung 3 Zeigt, wie die unverschlüsselten (Nutz-)Daten von einer beliebigen Anwendung - in unserem Falle dem Webserver - an die verschlüsselnde SSL weitergegeben werden und ab diesem Punkt verschlüsselt im Netz kommuniziert werden, bis sie auf der Seite des Clients wieder entschlüsselt werden. Dabei gehen wir natürlich davon aus, dass ausreichend starke Verschlüsselung zum Einsatz kommt. Es entsteht ein so genannter Tunnel, der es den im Internet zwischengeschalteten Rechnern (den Routern) unmöglich macht, die verschlüsselten Inhalte zu lesen. Nur die für den Transport notwendige Information bleibt für die Transporteure offen. Diese Herangehensweise folgt dem *need-to-know*-Prinzip.

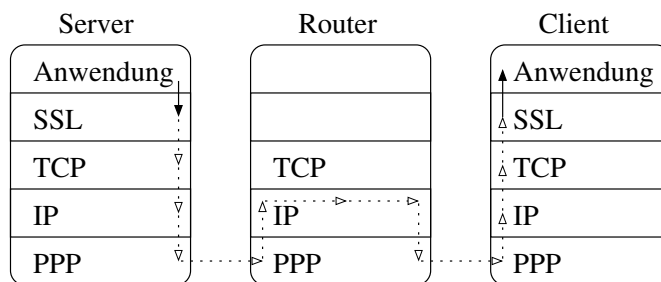


Abbildung 3: SSL tunnelt die Kommunikation zwischen Server und Client

Für viele netzwerkbasierte Anwendungen wurden eigene Protokolle definiert, die - je nach Konstruktion der Anwendung und des Protokolls - Angriffsmöglichkeiten offen lassen. Prinzipiell ist SSL nicht auf einen bestimmten Dienst festgelegt, sondern ist geeignet, um verschiedenste Protokolle der Anwendungsschicht abzusichern. Vor allem die Tatsache, dass Protokolle der Anwendungsschicht unabhängig vom Vorhandensein von SSL funktionieren,

macht SSL zu einem Sicherungswerkzeug, dessen Einsatz wenig Aufwand erfordert. Durch die Implementation dieser transparenten Zwischenschicht kann die Kommunikation im Netzwerk einer jeden Anwendung, die auf TCP aufsetzt, gesichert werden.

Ist der Einsatz von Anwendungen geplant, für die eigene Protokolle implementiert wurden, so müssen die in Betracht kommenden Protokolle einer eingehenden Überprüfung unterzogen werden. Die Überlegungen, die weiter unten über die Auswahl geeigneter Software gemacht werden sind hier direkt auf die Implementation der Protokolle übertragbar.

Die Anwendungsschicht

Die Anwendungsschicht ist der bei weitem komplexeste Teil des Referenzmodells. Die Entwicklung einer unübersehbaren Vielfalt von netzwerkbasieren Programmen hat zu einer - geringeren, doch immer noch nicht überschaubaren - Vielfalt von Netzwerkprotokollen geführt, die auf dieser Ebene anzusiedeln sind.

Eine Sicherung nach Außen ist durch die Firewall-Software bereits implementiert, doch gibt es auf der Ebene der Anwendungen ein weites Betätigungsfeld, wenn man es sich zum Ziel setzt, alle Anwendungen ähnlich genau zu kontrollieren, wie dies beim Netzwerk möglich ist.

Da jedoch kaum ein Betreiber eines Portalsystems die Sicherheit seiner Anwendungen vollständig und alleine überprüfen kann, geschieht die Sicherung der Anwendungsschicht zum größten Teil durch die Wahl der Software.

Bis hin zur Anwendungsschicht ist ein effektiver Schutz vor Angriffen aus dem Netzwerk mit Firewall-Software zu erreichen. Firewall-Software kontrolliert den ein- und ausgehenden Datenfluss der Netzwerkschnittstellen von Rechnersystemen und ermöglicht es, Netzwerk-Pakete nach Typ, nach Absender- oder Empfänger-Adresse oder nach Absender- oder Empfänger-Port-Nummer zu unterscheiden und umzuleiten, abzublocken, oder zu verändern.

Bei der Konfiguration der Firewall ist es ratsam, das Prinzip der *fail safe Defaults* zu beachten, d. h. der Prozess der Konfiguration erfolgt in der Weise, dass zunächst jeder Zugriff vom System auf das Netzwerk und aus dem Netzwerk auf das System gesperrt wird. Dann werden die für die Funktion notwendigen Ports der Reihe nach freigeschaltet. In unserem Falle sind dies die Folgenden:

Soll das Portalsystem ausschließlich als WWW-Portal dienen, so sind dies die einzigen Dienste, die vom Internet aus erreichbar sein müssen. UDP-Pakete werden vollständig blockiert.

Protokoll	Port-Nummer	Dienst
TCP	80	HTTP
TCP	443	HTTPS

Tabelle 2: Relevante Dienste für ein Portalsystem im WWW

Aus organisatorischen Gründen kann es sinnvoll sein, auch z. B. die Ports 53 (TCP) und/oder 22 (TCP) für Pakete zum Verbindungsaufbau (SYN-Pakete) zu öffnen, um den *Domain Name Service* (DNS) zur Namensauflösung, respektive die *Secure Shell* (SSH) für Fernwartungsdienste zu nutzen.²⁴ Wir gehen jedoch davon aus, dass diese Dienste nicht angeboten werden sollen. Es ist leicht machbar, die angebotenen Dienste auf beliebigen anderen Ports anzubieten, und damit zu 'verstecken', doch kann dies kein Ersatz für die sichere Konfiguration der Serversoftware sein. Das Test-Tool *nessus* z. B. erkennt die Signaturen vieler verschiedener Dienste auch, wenn sie auf unüblichen Ports horchen.

Durch den Einsatz eines dedizierten Firewall-Rechners, der im lokalen Netz vermittelnd zwischen Webserver und Internet steht, kann die Sicherheit gegenüber dem Internet noch vergrößert werden. Dies noch mehr durch die Einrichtung einer so genannten *demilitarized Zone* (DMZ), die auch zum lokalen Netz hin geschützt ist und von außen nur sehr schwer zu erreichen ist. Damit kommt auch hier die Aufteilung der Systeme zum Einsatz, wir folgen dem Prinzip des *least common Mechanism*.

Das implementierte Beispielsystem setzt zur Verschlüsselung der übertragenen Passwörter - und optional aller Verbindungen zum Portal - HTTPS, also HTTP über SSL ein, das bei der Nutzung von Diensten im World Wide Web offensichtlich dem unverschlüsselten HTTP vorzuziehen ist. Dies umso mehr, je mehr die zu übertragenden Daten vertraulich behandelt werden sollten; zum Beispiel bei der Übertragung von Passwörtern. Das Abhören oder die Manipulation anderer Datenübertragungen im Internet birgt zwar weniger Missbrauchspotenzial, doch soll unser Portal seinen gesamten Dienst auch verschlüsselt anbieten. Obwohl bei weitem die meisten WWW-Browser die Sicherung der Übertragung durch SSL unterstützen, soll auch ein unver-

²⁴Der unverschlüsselte Dienst Telnet, der lange Zeit für solche Aufgaben benutzt wurde, wurde inzwischen auch um eine SSL-Option erweitert, damit er über eine gesicherte Verbindung nutzbar ist, SSH hat jedoch andere Vorteile derent wegen wir SSH den Vorzug geben.

schlüsselter Login möglich sein. Die informationelle Selbstbestimmung gebietet es, den Nutzer entscheiden zu lassen, welche seiner Daten schützenswert sind und welche nicht.

Die Überprüfung der Sicherheit einer Netzwerk-Konfiguration ist relativ einfach. Ob die gestellten Anforderung an die Sicherheit des Systems eingehalten werden, ist schon mit (relativ) simplen Netzwerk-Analyse-Tools wie *netcat*, oder bedienungsfreundlicheren wie *ethereal* oder dem mächtigen Sicherheits-Test-Tool *nessus*²⁵ möglich. Sie lassen eine detaillierte Analyse des Netzwerkverkehrs und - im Falle von *nessus* - einen aggressiven Test der Sicherung der Infrastruktur und des gesamten Datenverkehrs zu.

10 Wahl der Software

Das ausladende Feld der Software-Auswahl soll hier nur aus dem Gesichtspunkt der Sicherheit heraus betrachtet werden. Finanzielle Aspekte, die bei der Software-Auswahl in der Praxis oft eine große Rolle spielen, sollen dabei ebenso bewusst unbeachtet bleiben, wie Überlegungen, die von einem eventuell bestehenden System ausgehen. Diese Voraussetzung ist relativ naiv, da Faktoren wie die Gewöhnung an ein bestehendes System oder die persönlichen Präferenzen und Vorkenntnisse der Entscheidungsträger in der Praxis einen großen Einfluss auf solche Entscheidungen haben. Wir wollen jedoch versuchen, die Entscheidung nicht von solchen Faktoren abhängig zu machen.

Auf den ersten Blick ist die Wahl der Software - vom Betriebssystem bis hin zur spezialisierten Anwendung - eine vor allem technische Entscheidung. Es zeigt sich aber, dass nicht-technische Eigenschaften von Informationssystemen eine immer wichtigere Rolle spielen und technische Entscheidungen beeinflussen.

Es ist nahezu unmöglich, in der Vielfalt der Dienste und Anwendungen, die für ein modernes Informationssystem eingesetzt werden sollen, selbst sicher zu stellen, dass alle Anforderungen, denen die Software gerecht werden soll, auch erfüllt sind.

Kaum ein Hersteller von Software bietet eine umfassende Garantie für deren Funktion oder gar Eignung für einen bestimmten Zweck,²⁶ wodurch eine - häufig proprietärer Software zugesprochene - Sicherheit durch die Möglichkeit des Rechtsweges weitestgehend ausgeschlossen wird.

“Ihre Software [die kommerzieller Anbieter] wird angeboten,

²⁵<http://netcat.sourceforge.net>, <http://www.ethereal.com>, <http://www.nessus.org>

²⁶[Mic03],[FSF91]

wie sie ist (asis), ohne die darüber hinausgehende Garantie, dass sie für den intendierten Einsatz taugt und mit so weitgehenden Garantie- und Haftungsausschlüssen, wie es die jeweilige nationale Jurisdiktion zulässt.” [Gra02]

Beispiele für diese Praxis sind anhand der *End User License Agreement* (EULA) für Microsoft Windows 2000 Pro und der GPL in Anhang A aufgeführt. Es fällt dabei auf, dass Hersteller proprietärer Software diesen Haftungsausschluss häufig an recht unauffälliger Stelle erwähnen. So wird das EULA von Windows 2000 Pro beispielsweise einmal zur Laufzeit einer Installation des Betriebssystems automatisch angezeigt, nämlich während der Installation in einer Reihe von Dialogen, die üblicherweise der Reihe nach *weggeklickt* werden. (Dass ein Anwender diese Dialoge tatsächlich durchliest, ist die Ausnahme). Eine Kopie des EULA liegt auch im Verzeichnis `C:\winnt\system32\eula.txt`, im laufenden Betrieb wird dem Nutzer allerdings durch einen Dialog implizit davon abgeraten, sich den Inhalt dieses Verzeichnisses anzeigen zu lassen, was ihn offensichtlich davon abhält, zufällig auf die Datei - eine unter mehr als 1500 in diesem Verzeichnis - zu stoßen.

Wenn auch die GPL vermutlich noch geringere Gewährleistung anbietet, als es ein kommerzieller Anbieter nach gültigem Recht tun muss, so ist doch die Tatsache, dass ausgesprochen aufdringlich auf diesen Umstand hingewiesen wird, eher eine *vertrauensbildende* als eine *vertrauensschädigende* Maßnahme.

Die GPL verlangt, dass zusätzlich zu dem in Anhang A zitierten Passus - der zusammen mit der Gesamten Lizenz mit jeder Kopie der Software an den Empfänger weitergegeben werden muss - dem Benutzer interaktiver Software an prominenter Stelle eine Meldung wie folgende angezeigt wird:²⁷

```
Gnomovision version 69, Copyright (C) <year> <name of author>
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.
```

Eine versichernde Gewährleistung für die Funktionalität von Software ist also kaum oder gar nicht zu bekommen. Es stellt sich die Frage, wo nun sonst Sicherheit zu finden ist. Im Folgenden wird deutlich werden, dass Sicherheit nach unserer technischen Definition - wenn überhaupt - dann nur bei solchen Produkten theoretisch erreichbar ist, deren Source Code offen liegt.

²⁷Nutzer freier Softwaresysteme stoßen ausgesprochen häufig auf solcherlei Hinweise

Beschäftigen wir uns zunächst mit dem Auffinden von Fehlern in Software.

Wird ein Fehler oder eine Sicherheitslücke erst dann offenbar, wenn der Fehler auftritt oder die Sicherheitslücke ausgenutzt wird, kann die Fehleranalyse schon schwierig sein, wenn sie nicht durch das Auftreten des Fehlers oder das Ausnutzen der Sicherheitslücke unmöglich geworden ist. Im schlimmsten Fall ist eine Analyse nicht einmal im Ansatz möglich, da der Zugriff zum System versperrt oder die für die Analyse notwendigen Daten verloren sind.

Software, deren Quellcode nicht offen liegt (*closed Source-* oder *Blackbox-*Software), bietet ausschließlich die Möglichkeit, Analysen auf deren Fehlfunktionen zu gründen. Ein Fehler muss erst auftreten, um eine Fehlfunktion oder Sicherheitslücke überhaupt zu entdecken, da keine qualifizierte Funktionsanalyse vor der Inbetriebnahme der Software möglich ist. Auch wenn es üblich ist, dass Software vor ihrem (kritischen) Einsatz zunächst auf einem unkritischen System getestet wird, ist dies Mangels der Möglichkeit, alle möglichen Szenarien zu testen ein Manko.

In einem vollkommen intransparenten System, das mehrere *closed Source* Programme beherbergt ist es bereits problematisch, herauszufinden, in *welchem* Prozess ein Fehler aufgetreten ist. Tatsächlich ist eine exakte Fehlerbestimmung schon dann nicht mehr möglich wenn in einem - ansonsten offenen - System *ein* Blackbox-Prozess genutzt wird, der nicht mit völliger Sicherheit als Fehlerquelle ausgeschlossen werden kann. Durch eine eingehende Prüfung kann jede andere Fehlerquelle ausgeschlossen werden, doch wenn die übrige - offene - Software sich als fehlerfrei entpuppt, kann nur noch festgestellt werden, *dass* der Blackbox-Prozess fehlerhaft ist, nicht aber *genau wo*.

Dies stellen auch Miller und Forrester bei einem Stresstest von Windows-NT- und Windows-2000-Systemen fest. Bei den Tests wurden die Systeme zum einen dauerhaft zufälligen Tastatur- und Maussignalen, eine Belastung - die deswegen relevant ist, weil sie durch normale Benutzung der Systeme auch entstehen kann - und zum anderen einer zufälligen Folge von Systemnachrichten ausgesetzt.²⁸

Dieser Test folgte einem analogen Test verschiedener UNIX-artiger Systeme, der in [MKL⁺95] beschrieben ist. Beim Test der UNIX-artigen waren die Sourcen für einige der Systeme verfügbar, was den Ausführenden für diese Systeme eine detaillierte Fehlersuche und -Beschreibung ermöglichte. Der Detaillierungsgrad war so groß, dass den Herstellern der Software ein präziser

²⁸Die sog. Win32-Messages sind ein dem Betriebssystem eigenes Konstrukt zur internen Kommunikation zwischen laufenden Anwendungen. Sie nicht mit den Benutzermeldungen zu verwechseln.

Katalog von Fehlerbeschreibungen vorgelegt werden konnte.

Die Analyse der Ergebnisse der Tests, die ohne Zugriff auf den Quellcode stattfanden, musste dagegen recht allgemein gehalten werden. So beschließen Miller et al. ihre Analyse der Windows NT Systeme:

“The lack of general access to application source code prevented us from making a more detailed report of the causes of program failures.” [MF90]

Ist es dem Betreiber aber nun gelungen herauszufinden, welches closed Source Programm einen bestimmten Fehler verursacht hat, so ist es ihm immer noch nicht möglich, diesen selbst zu beheben. Nachdem der Hersteller der Software verständigt wurde, bleibt also nur ein einstweiliges Umgehen des fehlerhaften Prozesses - so dieses möglich ist - oder der Verzicht auf die Nutzung der Software. Es hängt in diesem Moment vom Hersteller ab, ob und wann die fehlerbehaftete Software genauer überprüft und eventuell verbessert wird. Bei kritischen Anwendungen ist dies offensichtlich keine erstrebenswerte Situation. Es besteht meist die Möglichkeit, gegen einen Aufpreis, einen Service-Vertrag mit dem Hersteller der Software abzuschließen, der dafür eine bevorzugte Behandlung solcher Fehlfunktionen garantiert, wobei davon auszugehen ist, dass eine Verkürzung der Reaktionszeit eine Erhöhung des Entgeltes mit sich bringt.

Open Source Software bietet die Möglichkeit, ihre Funktionalität systematisch zu prüfen, wobei diese Prüfung durchaus schon vor dem Einsatz der Software geschehen kann. Diese Möglichkeit ist aber mit sehr hohem Aufwand verbunden, wenn es sich um ein größeres Softwarepaket handelt. Selbst ein großes Team von Entwicklern würde sehr lange brauchen, um beispielsweise den kompletten Quellcode des Webservers 'apache' eingehend zu analysieren.

Ein Programm, kann auch erst nachträglich zu einem Open Source Programm geworden sein, d. h. es wurde bei bei geheimem Quellcode als proprietäres Produkt entworfen und der Quellcode nachträglich veröffentlicht. Die Überprüfung solcher Projekte ist ebenso aufwändig wie die oben beschriebene.

Wesentlich effizienter ist es offensichtlich, Software schon während ihrer Entwicklung einer dauernden Prüfung zu unterwerfen.

Die Tatsache, dass eine effektive nachträgliche Kontrolle von Software dermaßen aufwändig ist, macht eine solche Kontrolle für die meisten Benutzer unrentabel, es ist also allgemein gebräuchlich, dass die *Kontrolle* der Software ausgeklammert und durch *Vertrauen* kompensiert wird.

An dieser Stelle sticht das Entwicklungsmodell, das einem Großteil der freien Software zu Grunde liegt, hervor. Nicht nur liegt der Quellcode dieser

Software von Beginn an offen, auch ist meist von Beginn an eine Vielzahl von Entwicklern an dessen Entstehung und Analyse beteiligt.

Dieses Modell der Freilegung des Quelltextes von Software hat dazu geführt, dass auch sehr große Softwarepakete, deren Quellcode schon zu Beginn des Projektes offen lag - wie auch das obige Beispiel des Webservers 'apache' - seit ihrer ersten Vorversion von einer Vielzahl von Programmierern (die meist auch direkt an ihrer Entwicklung beteiligt waren und sind - man spricht hier von einer *Entwicklergemeinde*) geprüft und getestet wurden. Dies schließt natürlich Fehler nicht aus, ihre Entdeckung wird aber durch die große Öffentlichkeit äußerst wahrscheinlich. Die Erfahrung hat gezeigt, dass Sicherheitsprobleme solcher Projekte meist sehr schnell - i. e. innerhalb weniger Tage, wenn nicht gar Stunden - gelöst und eine verbesserte Version frei angeboten wird.

Miller et al. haben in ihrem Test UNIX-artiger Betriebssysteme und der darauf laufenden Programme außer einer Reihe proprietärer Betriebssysteme auch das unter der GPL veröffentlichte freie Betriebssystem *GNU is Not Unix* (GNU) mit einem Linux-Kernel, der ebenfalls unter der GPL steht, getestet. Während die beschriebenen Tests "*far from elegant*" [MKL⁺95] sind, kommen Miller et al. zu Ergebnissen, die die oben gemachten Überlegungen stützen.

"It is also interesting to compare results of testing the commercial systems to the results from testing 'freeware' GNU and Linux. The seven commercial systems in the 1995 study have an average failure rate of 23%, while Linux has a failure rate of 9% and the GNU utilities have a failure rate of only 6%." [MKL⁺95]

Zukunftssicherheit Grundsätzlich sollte bei der Auswahl jedweder Software beachtet werden, dass sie vom Hersteller oder der Entwicklergemeinde regelmäßig gepflegt wird.

Die Geschichte der Sicherheit einer Software sollte einsehbar und nachvollziehbar sein. Aus ihr kann z. B. die Reaktionszeit der Entwickler auf Fehler - vom Bekanntwerden einer Sicherheitslücke bis zum Schließen derselben durch ein Update - beurteilt werden.

Es kann dabei auch beurteilt werden, wie wahrscheinlich es ist, dass ein System, sollte es genutzt werden, auch auf lange Sicht weiter vom Hersteller unterstützt werden wird.

Offensichtlich gibt es keine Garantie, dass die Entwicklergemeinde ein Projekt auf lange Sicht weiterentwickeln und pflegen wird. Doch ist die Situation bei proprietärer Software kaum anders. Oft findet sich zwar zum Beispiel bei der Auflösung eines Software-Herstellers ein anderer, der die Projekte des

sich auflösenden Betriebes übernimmt, doch werden auch diese oft nicht weiterverfolgt. Solche Übernahmen geschehen zum Teil aus dem Grund, dass die Software eines ehemaligen Konkurrenten nicht in andere Hände fallen soll, oder dass der Übernehmende Betrieb sich einen technologischen Vorteil aus der Analyse verspricht und die übernommenen Programme zu Gunsten seines Produktes 'sterben' lässt.

Nicht selten ist aber nicht einmal das Ausscheiden eines Herstellers aus dem Markt notwendig, damit die Unterstützung eines Softwareproduktes endet. Das Textverarbeitungsprogramm Word ist seit mehreren Versionsnummern dafür berüchtigt, dass die unterschiedlichen Versionen des Programms zueinander inkompatibel sind. Da einige freie Textverarbeitungen wie OpenOffice oder inzwischen einen beachtlichen Grad der Kompatibilität zu *allen* Versionen des Programms erreicht haben, obwohl es für sie keinen Zugang zum Quellcode gibt, ist anzunehmen, dass es sich hierbei nicht um eine technische Unmöglichkeit handelt.

Wiederum stoßen wir also auf eine Vertrauensfrage beim Vergleich der verschiedenen Kategorien von Software.

Die Entscheidung

Ein Ausschlusskriterium bei der Entscheidung für ein Bestimmtes Produkt ist die Bereitstellung der geforderten Funktionalität. Es zeigt sich, dass in allen Softwarekategorien Produkte verfügbar sind, die alle Funktionalität, die von unserem System erwartet wird, bereitstellen.

Da wir wissen, dass es äußerst aufwändig ist, sich Sicherheit über die Funktion von Software zu verschaffen - so aufwändig, dass wir davon ausgehen können, dass es für uns nicht praktikabel ist, dies in Angriff zu nehmen - und wir zudem um die qualitäts- und vertrauensbildenden Mechanismen in den verschiedenen Kategorien von Software und deren Entwicklungsmodellen wissen, ist es jetzt möglich, eine qualifizierte Entscheidung über die Wahl der zu verwendenden Software zu treffen.

Die Verfolgung des Ziels, der informationellen Autonomie möglichst nahe zu kommen, also u. a. in der Lage zu sein, Informationsarbeit bewusst und kontrolliert zu delegieren, ist ein weiterer Faktor, der bei dieser Entscheidung hilfreich ist.

Der Faktor der Kontrolle schließt allerdings die Kategorie der closed Source Software aus, da eine Kontrolle - und damit ein kontrolliertes Delegieren der Informationsarbeit - hier nur im Ansatz möglich ist.

Es bleibt also die open Source Software, die sich aus proprietärer und freier Software zusammensetzt; die Frage nach der informationellen Autonomie

spricht aber auch gegen die proprietäre open Source Software, da die geforderte Kontrolle hier bestenfalls passiv erfolgen kann, während freie Software auch aktiv und mitwirkend kontrolliert werden kann.

Die Zukunftssicherheit ist bei allen Bewerbern ähnlich fragwürdig, die Frage danach also wenig hilfreich im Entscheidungsprozess, doch es ist nicht zuletzt der schlichte Faktor der Leistungsfähigkeit und relativen Fehlerfreiheit im Vergleich mit anderer Software, der uns eine Entscheidung zugunsten der freien Software treffen lässt.

Es wird der *stable*-Zweig²⁹ der Debian³⁰ GNU/Linux-Distribution installiert. Das (Software-)Paketmanagement von Debian unterscheidet zwischen den Kategorien *main*, *contrib* und *non-free*, welche ähnlich den von uns bisher verwandten Softwarekategorien sind. Diese Unterscheidung macht es für uns einfach, vollständig auf die Verwendung nicht-freier Software zu verzichten.

Doch auch in anderer Hinsicht erfüllt und unterstützt diese Distribution die von Saltzer und Schröder formulierten Konstruktionsprinzipien auf konsequentere Weise, als viele andere Distributionen. Die Standard-Installations-Prozedur erzeugt ein Betriebssystem, in dem nur die notwendigsten Dienste angeboten werden, die es zulassen, das System auf den geforderten Stand zu erweitern, was eine Anwendung der *Economy of Mechanism* ist. Dies verhindert eine Situation, wie die folgende:

“Schließlich werden Produkte, die von sich aus relativ sicher sein können, mit unsicheren Voreinstellungen und Installationswerten ausgeliefert.

Rieger führt als Beispiel die damals [1999] aktuelle SuSE-Distribution an, die nach einer Standardinstallation unnötig viele Ports offen ließ. Auch wenn es möglich ist, ein SuSE-Linux wasserdicht zu machen, zeigt es sich doch bei allen Computersystemen, dass viele Endanwender die Standardeinstellungen nie verändern.” [Gra02]

Das zitierte Beispiel macht die Bedeutung der Prinzipien der *Economy of Mechanism* und der *fail-safe Defaults* deutlich. Ein System, das im Grundzustand weniger Funktionalität bereitstellt, und das voraussetzt, dass jede Funktion explizit nach den eigenen Wünschen konfiguriert und aktiviert wird, erfüllt diese Prinzipien bei weitem besser, als eines, das sich wie oben beschrieben verhält.

²⁹Debian kann außer in der stabilen auch in experimentelleren (Entwickler-)Versionen installiert werden, wovon bei diesem System aber abgesehen wurde.

³⁰<http://www.debian.org>

Solcherlei unsichere Voreinstellungen werden oft mit dem erhöhten Komfort begründet, den sie für den Anwender bringen³¹, stehen jedoch sehr häufig im direkten Konflikt mit der Systemsicherheit, die ebenfalls von den Anwendern erwartet wird. Dieses Konfliktes zwischen Komfort und Sicherheit sind sich aber viele Anwender nicht bewusst.

³¹Diese Situation entsteht oft aus dem Wunsch der Anwender, Dienste anzubieten und zu nutzen, ohne die Kompetenz für ihre Einrichtung erlangen zu müssen.

11 Zugriffskontrolle

Die Verwaltung von Rechten und der Schutz vor Missbrauch, sowie das Ermöglichen individueller Einstellungen werden üblicherweise durch die Definition von Benutzern und Benutzergruppen und die Verteilung von Benutzer- und Gruppenprivilegien auf diese Gruppen ermöglicht. Das Fehlen einer solchen Benutzer- und Gruppeneinteilung lässt für alle, die auf ein System zugreifen, dieselben Aktionen zu. Beispiele für diese Art der (nicht-)Regelung sind MS-DOS oder einfache Webseiten ohne Loginfunktionalität.

Eine Unterscheidung verschiedener Benutzer ist zwangsläufig mit einer Authentifikationsprozedur verbunden, die es zulässt, einen Nutzer eindeutig zu identifizieren. Im Internet und auf Betriebssystemebene geschieht die Identifikation von Benutzern meist durch eine Kombination von Benutzername und zugehörigem Passwort (*Authentifikation durch Wissen*).

Alternative Ansätze zur Authentifikation durch Wissen sind die biometrischen Verfahren, die Anhand verschiedener physiologischer einzigartiger Merkmale wie zum Beispiel der Struktur der Iris oder der Retina des Auges Personen identifizieren können. Diese sind im Internet nur schwer einsetzbar, da ja die Bestätigung der Authentifikation auch über das Internet verschickt werden müsste, und diese Daten wiederum einer Authentizitätsprüfung bedürfen.

Schlüsselbasierte Verfahren sind für den Einsatz in Netzwerken bestens geeignet. Sie lassen es zu, die Verbindlichkeit, Vertraulichkeit und Integrität von Nachrichten (also Information) zu sichern. Die Verfahren die das vermögen sind so genannte asymmetrische Verfahren, deren Grundprinzip es ist, dass es für jeden Teilnehmer einen *öffentlichen* und einen *privaten* Schlüssel gibt. Die Verteilung der Schlüssel lässt sich leicht aus deren Namen schließen. Während der private, oder "geheime", Schlüssel nur dem Eigentümer bekannt sein darf, kann und soll der öffentliche Schlüssel jedem Zugänglich sein. Die Schlüssel verhalten sich so zueinander, daß eine Nachricht, die mit einem der Schlüssel verschlüsselt wurde, *nur* mit dem anderen *ent*-schlüsselt werden kann.

Ein Vorteil von schlüsselbasierten Verfahren ist der, dass es möglich ist, die Authentifikation ohne die Übertragung sensitiver Information wie Passwörter es sind zu gestalten. Ein populäres Verfahren dazu ist der *Diffie-Hellman* Schlüsseltausch, der nur einer Übertragung der öffentlichen Schlüssel bedarf.³²

Schlüssel-basierte Authentifikation findet auch bei der Implementation der SSL Anwendung.

³²Eine Eingehende Behandlung dieser Thematik ist z. B. in [MvOV01] zu finden.

Konsequent eingesetzte Zugriffskontrollen spielen auch für die *gefühlte* Sicherheit des Nutzers eine Rolle. Nach dem *need to know* - Prinzip sollen für Benutzer jeder Gruppe nur diejenigen Funktionen sichtbar und damit greifbar sein, für die der Benutzer in der jeweiligen Gruppe die Nutzungsrechte hat. Dies resultiert in einer Vereinfachung der Benutzungsoberfläche und schränkt den Zugriff auf diejenigen Funktionen des Informationssystems ein, auf die ein Zugriff für den Nutzer sinnvoll ist. Zu keiner Zeit kann also eine Aktion begonnen werden, die dann aufgrund mangelnder Berechtigungen abgebrochen werden muss. Dies vermittelt ein Gefühl der Kontrolle und Sicherheit, da der Nutzer das Verhalten des Systems einschätzen kann, und erhöht die Motivation, das System weiter zu nutzen.

Wenn also ein WCMS eingesetzt werden soll, so ist grundsätzlich ein WCMS notwendig, das zwischen verschiedenen Nutzern unterscheiden kann und eine Authentifikation zulässt. Dies zählt zur Grundfunktionalität, die ein WCMS bereitstellen sollte und kann vorausgesetzt werden.

Wie das WCMS die Vertraulichkeit der Authentifikationsdaten bei deren Übertragung sichert, oder ob hier überhaupt eine Sicherung vorgesehen ist, ist dank der in Abschnitt 9 beschriebenen Transparenz von SSL nicht von Belang. Durch den Einsatz von SSL kann die Authentifikation vom WCMS gesichert vollzogen werden, wenn auch das System selbst keine Sicherheitsmechanismen vorsieht. In diesem Falle wird eine Login-Prozedur über HTTPS herangezogen, um die Vertraulichkeit und gegebenenfalls die Authentizität der Authentifikationsdaten zu gewährleisten.

Ist eine Unterscheidung von verschiedenen *Nutzern* möglich, so können die Rechte der Nutzer entsprechend gesetzt werden, um eine Inhaltskontrolle wie weiter unten beschrieben umzusetzen. Um diese Umsetzung jedoch auch bei einer großen Anzahl von Benutzern beherrschbar zu halten, ist eine Verwaltung von *Gruppenrechten* notwendig. Dabei wird Benutzern die Zugehörigkeit zu bestimmten Gruppen zugewiesen, welchen jeweils unterschiedliche Zugriffsregeln zugewiesen sind, glücklicherweise gehört auch dies zu den Funktionen, die ein WCMS meist von Haus aus mitbringt.

Im Folgenden wird eine mögliche Vergabe von Zugriffsrechten erläutert, die eine feinkörnige Unterscheidung der Nutzer zulässt und eine Implementation der in Abschnitt 13 hergeleiteten Mechanismen der Inhaltskontrolle erlaubt.

Die Berechtigungen im laufenden System sind eine Projektion der Konstruktionsprinzipien, die in Abschnitt 3 eingeführt wurden. Wichtig sind in diesem Falle vor allem die *fail-safe Defaults*, *Economy of Mechanism* und das *need-to-know*-Prinzip.

Zur Erläuterung dieser Rechtevergabe werden in Portalen häufig anzutreffende Funktionen betrachtet. Hier sind dies:

- Artikel oder Kurzberichte, Textbeiträge
- Kommentare zu Artikeln
- FAQ
- Diskussionsforen
- Ein Kalender, um relevante Termine zu verwalten
- Private Nachrichten zur internen Kommunikation³³

Anonyme Besucher

Alle Besucher des Portals sind zunächst anonym und die Funktionalität der Portalsoftware steht ihnen nur zu einem Teil zur Verfügung. Anonyme Besucher können auf nahezu alle Daten, die dem Portal zur Verfügung stehen lesend zugreifen, nicht aber aktiv an der Gestaltung der Inhalte teilnehmen. Sie haben die Rechte, Artikel, FAQs, Forenbeiträge, Kommentare und Termine im Kalender zu lesen, sind aber nicht berechtigt, Änderungen vorzunehmen oder eigene Beiträge zu veröffentlichen. Entsprechend diesen Berechtigungen, sind für anonyme Besucher der Seite nur diejenigen Menüs und Links sichtbar, die ihnen lesenden Zugriff gewähren, nicht aber jene, die auf eine verändernde oder schreibende Funktionalität hinweisen.

Anonyme Besucher sind jedoch die einzigen, für die die Funktionalität des Login oder der Registrierung zugänglich ist. An deren Stelle tritt für angemeldete Besucher offensichtlich eine Logout-Funktionalität.

Eingetragene Benutzer

Jeder Besucher, der sich durch Angabe eines gültigen Benutzernamens und des dazugehörigen Passwortes authentifizieren kann, erhält den Status eines eingetragenen Benutzers. Jeder Besucher der Seiten kann über ein automatisiertes Formular eingetragener Benutzer werden. Für jede Vergabe von Privilegien der anderen Gruppen ist ein Benutzerkonto Voraussetzung.

Bei der Anmeldung der Benutzer halten wir uns an das *need-to-know*-Prinzip, d. h. die Daten, mit denen die Benutzer sich beim Portal registrieren, unterliegen keiner Prüfung. Die Anmeldung auf den Seiten kann unter

³³Es ist dies ein interner Dienst, der nicht mit dem E-Mail-Dienst verwechselt werden sollte.

Funktion	allgemeine Rechte
Artikel	lesen
Kommentare	lesen
FAQ	lesen
Foren	lesen
Termine	lesen
Private Nachrichten	kein Zugriff

Tabelle 3: Rechtevergabe für anonyme Besucher

Pseudonym erfolgen, was einen eventuellen Missbrauch der Daten ad absurdum führt.

Das Portal stellt eingetragenen Benutzern einen erweiterten Funktionsumfang zur Verfügung, was sich zunächst an erweiterten Wahlmöglichkeiten in den Menüs bemerkbar macht. Die neuen Menüeinträge führen zu jenen Funktionen, die es den Benutzern möglich machen, sich aktiv am Portal zu betätigen und sich einzubringen.

Mit der Möglichkeit der Identifikation ergibt sich die Möglichkeit der privaten Kommunikation. Alle Benutzer können, wenn sie den Benutzernamen des Kommunikationspartners kennen, ihm private Nachrichten senden, die ihm beim nächsten Login vorgelegt werden.

Es ist allen registrierten Benutzergruppen erlaubt, Kommentare zu Artikeln abzugeben, die dann (ohne eine Prüfung) an die Artikel angehängt werden. Wie die Kommentare zu Artikeln, werden auch Forenbeiträge und Termin-Einträge im Kalender ohne vorherige Prüfung veröffentlicht.

Benutzer können in einem Formular eigene Textbeiträge (Artikel) verfassen, die dann an vom System den Lektoren vorgelegt werden. Ebenso kann diese Benutzergruppe die Inhalte der FAQ beeinflussen, indem sie Fragen vorschlägt, die dann von Lektoren beantwortet und veröffentlicht werden können.

Direktveröffentlicher

Es ist aus technischer Sicht ein kleiner Schritt vom eingetragenen Benutzer zum Direktveröffentlicher. Es ändert sich hier nur ein Detail. Direkt-

Funktion	allgemeine Rechte
Artikel	lesen, einreichen
Kommentare	lesen, schreiben
FAQ	lesen, Fragen stellen
Foren	lesen, schreiben
Kalender	lesen, Ereignisse eintragen
Private Nachrichten	lesen, schreiben

Tabelle 4: Rechtevergabe für eingetragene Benutzer

veröffentlicher haben einen ihnen zugewiesenen Bereich, sei er nun thematisch oder technischer Art,³⁴ in dem Sie Inhalte, also Textbeiträge ohne Zwischenprüfung veröffentlichen können.

Ihnen wird das Vertrauen ausgesprochen, dass die Inhalte, die sie veröffentlichen relevant und nicht missbräuchlich sind.

Direktveröffentlicher haben keinen Einfluss auf die Veröffentlichungen anderer, d. h. sie haben ausschließlich Schreib-, nicht aber Lösch- oder Änderungsrechte.

Lektoren

Lektoren bekommen Lese, Schreib-, Änderungs- und Löschrechte für die ihnen zugewiesenen Teilbereiche, die sie betreuen.

Sie bekommen Artikel zur Prüfung, die von eingetragenen Benutzern zur Veröffentlichung vorgeschlagen wurden und können ggf. Änderungen vornehmen.³⁵, oder die Artikel nicht zur Veröffentlichung freigeben. Die Änderungen, die ein Lektor vornehmen kann sind rein inhaltlicher Art, es wäre eine Verletzung der Authentizität der Daten, ihm die Möglichkeit zu geben, einem Artikel einen anderen Autor zuzuordnen. Allein die Tatsache, dass es dem Lektor möglich ist, eingereichte Beiträge inhaltlich zu verändern, ist eine Gefährdung der Authentizität dieser Beiträge. An dieser Stelle wird

³⁴z. B. ist es möglich, dass ein Direktveröffentlicher nur in den FAQ direkt schreiben, aber keine Artikel unbesehen veröffentlichen darf.

³⁵oder Änderungen von den Autoren fordern.

Funktion	allgemeine Rechte	Zuständigkeitsbereich
Artikel	lesen	lesen, schreiben
Kommentare	lesen, schreiben	lesen, schreiben
Foren	lesen	lesen, schreiben
FAQ	lesen, Fragen stellen	lesen, Fragen stellen, Fragen beantworten
Kalender	lesen, Ereignisse eintragen	lesen, Ereignisse eintragen
Private Nachrichten	lesen, schreiben	lesen, schreiben

Tabelle 5: Rechtevergabe für Direktveröffentlicher

ein Kompromiss eingegangen, um einen anderen Aspekt der Sicherheit zu gewährleisten.

Lektoren können in den ihnen zugewiesenen Teilbereichen offensichtlich ebenso direkt veröffentlichen, da die Bereiche nach den Kompetenzfeldern der Lektoren ausgesucht werden. Will ein Lektor eines Teilbereiches A allerdings im Teilbereich B veröffentlichen, so muss er den Umweg über den zuständigen Lektor gehen. Dieses Berechtigungsmodell ermöglicht eine Qualitätssicherung ist aber nicht geeignet, um eine durchgängige Kontrolle aller Inhalte zu gewährleisten. Dies ist Ausdruck des Vertrauens, das den Lektoren und Direktveröffentlichern ausgesprochen wird, ihre Privilegien nicht zu missbrauchen

Administratoren

Administratoren haben grundsätzlich alle Rechte und Privilegien. Sie sind vor allem für die technische Wartung des WCMS zuständig. Sie haben alle vom System unterstützten Rechte und Möglichkeiten. Hiervon sind wichtige Beispiele die Nutzer- und Rechteverwaltung. Diese erlaubt es ihnen, das oben aufgeführte Modell zu implementieren oder Benutzer bestimmten Gruppen zuzuweisen, dem System hinzuzufügen oder zu löschen. Administratoren haben auch Zugriff auf die Konfiguration des WCMS selbst, wie die Möglichkeit,

Teilbereich	allgemeine zugelassen	im Zuständigkeitsbereich
Artikel	lesen, einreichen	administrieren
Foren	lesen, schreiben	administrieren
FAQ	lesen, Fragen stellen	administrieren
Kommentare	lesen, schreiben	administrieren
Kalender	lesen, Ereignisse eintragen	administrieren
Private Nachrichten	lesen, schreiben	lesen, schreiben

Tabelle 6: Rechtevergabe für Lektoren

Module des WCMS einzufügen oder Spracheinstellungen zu verändern.

Teilbereich	allgemeine Rechte
Alle	administrieren

Tabelle 7: Administratoren haben alle Rechte

12 Aufrechterhaltung

Die Sicherung eines Systems endet nicht nach seiner Erstellung. Eine entscheidende Rolle spielt dabei auch das Erkennen und die Analyse von Angriffen. Um diese zu ermöglichen, ist es notwendig, die Benutzte Software derartig zu konfigurieren, dass in Log-Dateien relevante Ereignisse verzeichnet werden. Sinnvoll ist das erstellen dieser Dateien erst, wenn sie regelmäßig, auch unter Zuhilfenahme von Filterprogrammen, auf die Spuren von Angriffen hin überprüft werden. Es ist möglich, dass ein Verlust der Vertraulichkeit oder Integrität der Daten hier bemerkbar wird, eventuell noch bevor dadurch Schaden entsteht.

Der Großteil der Filterprogramme, die geeignet sind, um Administratoren bei der Analyse der Log-Dateien zu unterstützen, fällt in die Kategorie der

Intrusion Detection Systeme (IDS). IDS sind Programme oder Programmsammlungen, die auf unterschiedliche Weise den Betrieb des Systems überwachen und auf Anomalien oder Angriffe reagieren können. Diese Reaktionen können von einer einfachen Benachrichtigung zu einem beliebigen Medium bis hin zu defensiven Maßnahmen wie der Sperrung von IP-Adressen oder dem Ausloggen eines Benutzers reichen. Unter dem Pseudonym “Anonymous” hat der Autor von [ano00] ein umfassendes Kompendium über diesen Themenbereich erstellt.

Durch die schnellen Veränderungen, denen die IT unterworfen ist, ist auch für die Verantwortlichen notwendig, über aktuelle Gefahren und Risiken, Sicherheitslücken und -Updates informiert zu bleiben. Hierzu ist das Internet nahezu die einzige Quelle, die eine ausreichend schnelle Reaktion zulässt. Es gibt diverse Newsgroups, E-Mail-Dienste und Websites, die sich mit dieser Problematik beschäftigen, für ein System wie das hier beschriebene ist unter anderem <http://security.debian.org> relevant.

Teil III

Die erweiterte Sicherheit

Vom konkreten, nachprüfbar und eindeutigen Feld der technischen Sicherheit unterscheidet sich die nicht-technische Sicherheit dadurch, dass sie vollkommen abhängig von den Motiven der Verantwortlichen ist und sich vor allem durch deren Handeln äußert. Sie ist nicht immer überprüfbar und ihr Fehlen nicht immer offensichtlich. Wenn auch einem guten Teil der Ideale der nicht-technischen Sicherheit durch die Anwendung freier Software nahe gekommen werden kann, so liegt immer noch eine Verantwortung bei den Betreibern des Portals, die aus deren Verfügungsgewalt über die ihnen überlassenen Daten entsteht.

Schon die Auswahl der verwandten Software bedarf des Vertrauens der Anwender. Dieser Bedarf ist noch größer bei den Endanwendern von Rechnersystemen, die - selbst wenn ihnen die Möglichkeit gegeben ist, den Dienstleistungen zugrunde liegende Mechanismen zu überprüfen - davon in den seltensten Fällen Gebrauch machen, und das meist aus dem einfachen Grund, dass ihnen die Kompetenz dazu fehlt.

Die Umsetzung der nicht-technischen Sicherheit kann nicht durch technische Maßnahmen geschehen, sondern allenfalls von ihnen unterstützt werden.

Die Verantwortlichen können sich aber selbst zu solchem Handeln verpflichten, welches ihren formulierten und veröffentlichten Sicherheits-Ansprüchen gerecht wird. Diese Verpflichtung geschieht immer in dem Maße freiwillig, wie sie über das gesetzlich geforderte Maß hinausgeht.

Eine freiwillige Verpflichtung bietet jedoch keinerlei reelle Sicherheit für den Benutzer, sondern muss als vertrauensbildende Maßnahme betrachtet werden, durch die die Zielsetzungen der Verantwortlichen klar definiert und veranschaulicht werden.

13 Informationelle Selbstbestimmung

Informationelle Selbstbestimmung³⁶ beinhaltet nach der Auffassung des Bundesverfassungsgerichts, dass es dem einzelnen obliegt, zu entscheiden ob und an wen er personenbezogene Daten weiter gibt.

Es ist gängige Praxis, dass Nutzer von Informations-Dienstleistungen das

³⁶Auch: informationelle Autonomie

Privileg, diese zu Nutzen durch die Preisgabe personenbezogener Information erkaufen, die als solche oft in keinem Zusammenhang mit dieser Nutzung steht, sondern vom Dienstleister gespeichert und zu seinen Zwecken weiter verwandt wird.

Die Freigabe dieser Daten erfolgt zwar freiwillig, doch geschieht sie oft vor allem aus der Abwägung heraus, dass dem Nutzer ohne diese Freigabe die Nutzung des Dienstes verwehrt bleibt. Die Freiwilligkeit solcher Zusagen ist dann besonders in Frage zu stellen, wenn sie die Nutzung eines Dienstes betreffen, zu dem es keine gleich- oder mehrwertige Alternative gibt, die ohne diese Konzession verfügbar ist.³⁷

Die Zustimmung zur Weiterverwendung solcher Informationen suchen die Dienstbetreiber *"[...] vor allem mit dem Argument, dass sie [die Nutzer] die Angaben ruhig machen können, da diese vertraulich und/oder ausschließlich in ihrem (angenommenen) Interesse behandelt würden. "* [Kuh98]

Diese Praxis stellt eine Einschränkung der informationellen Autonomie dar, die auf dem Ziel der Gewinnmaximierung der Betreiber der Dienstleistungen beruht. Eine solche Einschränkung ist gerade für ein Portal zur Informationsethik nicht zu rechtfertigen, weshalb wir davon absehen, derlei Daten zu sammeln.

Content-Control

Um die Möglichkeit des Missbrauchs des Portals zu verringern, ist eine routinemäßige Kontrolle der durch das System verbreiteten Inhalte notwendig.

Das möglicherweise aufkommende Datenvolumen lässt an den Einsatz technischer Hilfsmittel wie Wortfilter oder automatisierter statistischer Analysen zur inhaltlichen Überwachung denken, doch sind diese Hilfsmittel bisher vor allem Gegenstand der Forschung und noch nicht zu einem Grade heran gereift, der sie für diese Aufgabe zu mehr als einer Hilfestellung macht.

Die semantische Analyse textueller Inhalte ist bei weitem zu komplex, um von aktuellen Systemen zuverlässig gemeistert zu werden. Noch weiter ist der Weg, den die inhaltliche Erfassung anderer Kommunikationsformen bis zur zuverlässigen Funktionalität zu gehen hat. Bilder oder audiovisuelle Datenströme entziehen sich momentan nahezu vollkommen einer automatisierten Kontrolle.

Zur Unterstützung der automatisierten Inhaltskontrolle bietet das Content-Management-System zur Unterstützung der automatisierten Inhaltsüberwachung einfache Wortfilter an.

³⁷ ... oder dem Nutzer nicht bekannt ist, dass es diese Alternative gibt.

Eine solche Zensur ließe sich durch die Themenbezogenheit des Portals rechtfertigen, da es sich bei Nethics nicht um ein Portal *zur* freien Meinungsäußerung, sondern um ein Portal, das sich - unter anderem - *mit* freier Meinungsäußerung *beschäftigt* handelt.

Die Themenstellung von Nethics ist aber derartig, dass gerade solche Worte, die potentiell einem Wortfilter zum Opfer fallen würden, die also ethisch in einem vernetzten Informationsumfeld problematisch sind, hier zur Diskussion stehen.

Der Einsatz von Wortfiltern in einem Solchen Umfeld ist erschwert, da missbräuchliche Beiträge nicht durch das bloße Auftauchen dieser Vokabeln zu erkennen sind, sondern das Erkennen einer semantischen Analyse bedarf.³⁸

Wo Benutzer direkt Inhalte verbreiten können, wird wiederum die Praxis zeigen, ob hier auch potentiell gefährliche Inhalte - wie z. B. Würmer - verbreitet werden, oder Missbrauch (z. B. off-Topic-Einträge) die Benutzbarkeit des Forums einschränkt.

Die inhaltliche Kontrolle muss also persönlich erledigt werden. Die wichtigste Rolle obliegt dabei den Lektoren, denen die Aufgabe zukommt, jene Artikel zu lesen und zu bewerten, die von eingetragenen Benutzern zur Veröffentlichung eingereicht werden.

14 Rechtliche Absicherung

Das Gesetz über die Nutzung von Telediensten [Bun01] legt in §8 fest, dass die Betreiber von Telediensten nur begrenzt Verantwortung für Handlungen und Veröffentlichung Dritter tragen. Betreiber eines Portals wie Nethics haben dieser Regelung zu Folge keine Verpflichtung, die Inhalte der auf ihren Seiten von den Benutzern zur Verfügung gestellten Daten zu überprüfen.

Dies ist vor allem sinnvoll, da die Überprüfung der Datenmengen die schon bei Seiten mit einem mittleren Datenaufkommen so aufwändig ist, dass die Betreiber oftmals nicht über die Arbeitskraft verfügen, die notwendig wäre, um dieser Aufgabe gerecht zu werden, wie oben schon ausgeführt wurde.

Die Verantwortung gegenüber dem Gesetzgeber ist jedoch nicht die Einzige, die der Betreiber eines Forums trägt. Offensichtlich ist es auch im Interesse des Betreibers, nur solche Inhalte auf seinen Seiten verfügbar zu machen, deren Verbreitung auch seinen Konsens findet.

³⁸Die Praxis des Einsatzes des Portals wird zeigen, ob ein Einsatz von Wortfiltern zu einem späteren Zeitpunkt notwendig wird.

Sieber hat 1999 in [Sie99] eine eingehende Betrachtung über diesen Themenbereich angestellt. Es ist bei der Lektüre jedoch zu beachten, dass die Rechtslage sich in der Zwischenzeit verändert hat. Der “elektronische Rechtsraum” ist ein Thema geworden, das stark diskutiert wird und dessen rechtliche Regelung noch in den Kinderschuhen steckt.

Fazit

Ein Internet-Portal-System wie nethics.net zu erstellen, erfordert mehr als nur die technischen Kenntnisse und Fähigkeiten, derer es bedarf, um ein Serversystem zu implementieren und die Inhalte zu veröffentlichen.

Auch Inhalte aus den Bereichen der Informationswissenschaft und der (Informations-)Ethik bedürfen einer Betrachtung und haben Auswirkungen, die nicht nur ideeller Natur sind, sondern sich in der Implementation des Gesamtsystems bemerkbar machen.

Die Fragen nach den Sicherheitsaspekten von Vertrauen und informationeller Autonomie beeinflussen den Entscheidungsbaum, der bei der Konzeption eines Internet-Portalsystems entsteht, in hohem Maße.

Im Besonderen stellt sich heraus, dass alle Ziele der Sicherheit, die des Vertrauens und des Erreichens der informationellen Autonomie, wie auch die der Gewährleistung von Vertraulichkeit, Integrität, Verfügbarkeit und Verbindlichkeit (welche die technische Sicht der Sicherheit widerspiegeln) die Wahl der zu verwendenden Software beeinflussen.

A Klauseln zum Haftungsausschluss

Windows 2000 Pro EULA

11. AUSSCHLUSS DER GEWÄHRLEISTUNG. Die unten genannte beschränkte Garantie ist die einzige ausdrückliche Garantie, die Ihnen gegeben wird. Sie ersetzt alle anderen ausdrücklichen Garantien (falls zutreffend), die von der Dokumentation oder der Packung gegeben werden. Mit Ausnahme der beschränkten Garantie und im größtmöglichen durch das anwendbare Recht gestatteten Umfang stellen Microsoft und deren Lieferanten das Produkt und gegebenenfalls Supportleistungen WIE BESEHEN UND OHNE GARANTIE AUF FEHLERFREIHEIT zur Verfügung. Sie schließen hiermit alle anderen Garantien und Pflichten, gleich ob ausdrücklich, konkludent oder gesetzlich, einschließlich, aber nicht beschränkt auf (falls zutreffend) jede konkludente Garantie der Handelsüblichkeit, Eignung für einen bestimmten Zweck, Genauigkeit oder Vollständigkeit von Antworten, Ergebnisse, fachmännischen Bemühungen, Virenfreiheit und Fahrlässigkeit - alles bezüglich des Produkts - sowie der Bereitstellung von Supportleistungen oder der Tatsache, dass keine Supportleistungen erbracht worden sind, aus. ES WIRD AUCH JEDE GARANTIE FÜR EIGENTUM, UNGESTÖRTE NUTZUNG, UNGESTÖRTEN BESITZ, ÜBEREINSTIMMUNG MIT DER BESCHREIBUNG ODER NICHTVERLETZUNG VON RECHTEN DRITTER IN BEZUG AUF DAS PRODUKT AUSGESCHLOSSEN.

12. AUSSCHLUSS VON FOLGE-, ZUFÄLLIGEN UND BE-

STIMMTEN ANDEREN SCHÄDEN. IM GRÖSSTMÖGLICHEN DURCH DAS ANWENDBARE RECHT GESTATTETEN UMFANG SIND MICROSOFT ODER DEREN LIEFERANTEN IN KEINEM FALL HAFTBAR FÜR IRGENDWELCHE SPEZIELLEN, ZUFÄLLIGEN, INDIRECTEN ODER FOLGESCHÄDEN WELCHER ART AUCH IMMER (EINSCHLIESSLICH, ABER NICHT BESCHRÄNKT AUF SCHÄDEN AUS ENTGANGENEM GEWINN, VERLUST VON VERTRAULICHEN ODER ANDEREN INFORMATIONEN, GESCHÄFTSUNTERBRECHUNG, PERSONENSCHÄDEN, VERLUST VON PRIVATSPHÄRE, VERLETZUNG VON VERTRAGSPFLICHTEN (EINSCHLIESSLICH PFLICHTEN NACH TREU UND GLAUBEN ODER SORGFALTSPFLICHTEN), FAHRLÄSSIGKEIT SOWIE VERMÖGENS- ODER SONSTIGE SCHÄDEN), DIE AUS DER VERWENDUNG DES PRODUKTS ODER DER TATSACHE, DASS ES NICHT VERWENDET WERDEN KANN, ODER AUS DER BEREITSTELLUNG VON SUPPORTLEISTUNGEN ODER DER TATSACHE, DASS KEINE SUPPORTLEISTUNGEN ERBRACHT WORDEN SIND, ODER ANDERWEITIG AUS ODER IN VERBINDUNG MIT EINER BESTIMMUNG DIESES EULAS RESULTIEREN ODER IN IRGEND EINEM ZUSAMMENHANG DAMIT STEHEN, SELBST IM FALLE VON VERSCHULDEN, UNERLAUBTEN HANDLUNGEN (EINSCHLIESSLICH FAHRLÄSSIGKEIT), VERSCHULDENSUNABHÄNGIGER HAFTUNG, VERTRAGSBRUCH ODER VERLETZUNG DER GARANTIE VON MICROSOFT ODER DEREN LIEFERANTEN, UND SELBST WENN MICROSOFT ODER DER LIEFERANT AUF

DIE MÖGLICHKEIT SOLCHER SCHÄDEN HINGEWIESEN WURDE.

13. HAFTUNGSBESCHRÄNKUNG UND ANSPRÜCHE.

Ungeachtet aller Schäden, die Sie aus welchen Gründen auch immer erleiden mögen (einschließlich, aber nicht beschränkt auf alle oben angesprochenen Schäden sowie alle direkten oder allgemeinen Schäden) ist die gesamte Haftung von Microsoft und deren Lieferanten unter allen Bestimmungen dieses EULAs und Ihr ausschließlicher Anspruch für alles oben genannte (außer für Ansprüche bei Nachbesserung oder Nachlieferung, die von Microsoft bei einer Verletzung der beschränkten Garantie gewählt wird) beschränkt auf den tatsächlich von Ihnen für das Produkt gezahlten Betrag oder US-\\$ 5,00, je nachdem, welcher Betrag höher ist. Die vorstehenden Beschränkungen und Ausschlüsse (einschließlich der Abschnitte 11 und 12 weiter oben und wie in der beschränkten Gewährleistung angegeben) gelten im größtmöglichen durch das anwendbare Recht gestatteten Umfang, auch wenn ein Anspruch dadurch seinen wesentlichen Zweck verfehlt.

[Mic03], Abschnitte 11 bis 13

GNU General Public License (GPL)

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. [FSF91], Abschnitt

11

B Beispielszenarien aus dem Grundschutzhandbuch des BSI

Schutzbedarfskategorie “niedrig bis mittel”

Verstoß gegen Gesetze, Vorschriften, Verträge	<ul style="list-style-type: none"> • Verstöße gegen Vorschriften und Gesetze mit geringfügigen Konsequenzen • Geringfügige Vertragsverletzungen mit maximal geringen Konventionalstrafen
Beeinträchtigung des informationellen Selbstbestimmungsrechts	<ul style="list-style-type: none"> • Eine Beeinträchtigung des informationellen Selbstbestimmungsrechts würde durch den Einzelnen als tolerabel eingeschätzt werden. • Ein möglicher Missbrauch personenbezogener Daten hat nur geringfügige Auswirkungen auf die gesellschaftliche Stellung oder die wirtschaftlichen Verhältnisse des Betroffenen.
Beeinträchtigung der persönlichen Unversehrtheit	<ul style="list-style-type: none"> • Eine Beeinträchtigung erscheint nicht möglich.
Beeinträchtigung der Aufgabenerfüllung	<ul style="list-style-type: none"> • Die Beeinträchtigung würde von den Betroffenen als tolerabel eingeschätzt werden. • Die maximal tolerierbare Ausfallzeit ist größer als 24 Stunden.
Negative Außenwirkung	<ul style="list-style-type: none"> • Eine geringe bzw. nur interne Ansehens- oder Vertrauensbeeinträchtigung ist zu erwarten.
Finanzielle Auswirkungen	<ul style="list-style-type: none"> • Der finanzielle Schaden bleibt für die Institution tolerabel.

Schutzbedarfskategorie "hoch"

Verstoß gegen Gesetze, Vorschriften, Verträge	<ul style="list-style-type: none"> • Verstöße gegen Vorschriften und Gesetze mit erheblichen Konsequenzen • Vertragsverletzungen mit hohen Konventionalstrafen
Beeinträchtigung des informationellen Selbstbestimmungsrechts	<ul style="list-style-type: none"> • Eine erhebliche Beeinträchtigung des informationellen Selbstbestimmungsrechts des Einzelnen erscheint möglich. • Ein möglicher Missbrauch personenbezogener Daten hat erhebliche Auswirkungen auf die gesellschaftliche Stellung oder die wirtschaftlichen Verhältnisse des Betroffenen
Beeinträchtigung der persönlichen Unversehrtheit	<ul style="list-style-type: none"> • Eine Beeinträchtigung der persönlichen Unversehrtheit kann nicht absolut ausgeschlossen werden.
Beeinträchtigung der Aufgabenerfüllung	<ul style="list-style-type: none"> • Die Beeinträchtigung würde von einzelnen Betroffenen als nicht tolerabel eingeschätzt. • Die maximal tolerierbare Ausfallzeit liegt zwischen einer und 24 Stunden.
Negative Außenwirkung	<ul style="list-style-type: none"> • Eine breite Ansehens- oder Vertrauensbeeinträchtigung ist zu erwarten.
Finanzielle Auswirkungen	<ul style="list-style-type: none"> • Der Schaden bewirkt beachtliche finanzielle Verluste, ist jedoch nicht existenzbedrohend.

Schutzbedarfskategorie “sehr hoch”

Verstoß gegen Gesetze, Vorschriften, Verträge	<ul style="list-style-type: none"> • Fundamentaler Verstoß gegen Vorschriften und Gesetze • Vertragsverletzungen, deren Haftungsschäden ruinös sind
Beeinträchtigung des informationellen Selbstbestimmungsrechts	<ul style="list-style-type: none"> • Eine besonders bedeutende Beeinträchtigung des informationellen Selbstbestimmungsrechts des Einzelnen erscheint möglich. • Ein möglicher Missbrauch personenbezogener Daten würde für den Betroffenen den gesellschaftlichen oder wirtschaftlichen Ruin bedeuten.
Beeinträchtigung der persönlichen Unversehrtheit	<ul style="list-style-type: none"> • Gravierende Beeinträchtigungen der persönlichen Unversehrtheit sind möglich. • Gefahr für Leib und Leben
Beeinträchtigung der Aufgabenerfüllung	<ul style="list-style-type: none"> • Die Beeinträchtigung würde von allen Betroffenen als nicht tolerabel eingeschätzt werden. • Die maximal tolerierbare Ausfallzeit ist kleiner als eine Stunde.
Negative Außenwirkung	<ul style="list-style-type: none"> • Eine landesweite Ansehens- oder Vertrauensbeeinträchtigung, evtl. sogar existenzgefährdender Art, ist denkbar.
Finanzielle Auswirkungen	<ul style="list-style-type: none"> • Der finanzielle Schaden ist für die Institution existenzbedrohend.

Index

- AES, 9
- Angriffe, *siehe* Gefährdungen
- ARP, 26
- Authentizität, 23
- Autonomie, *siehe* Selbstbestimmung

- Bluetooth, 24
- BSI, 15, 17, 57

- CHAP, 26
- Content-Control, 48

- DES, 8
- Diffie-Hellman, 39
- DMZ, 30
- DNS, 30
- DOS, 27
- DRM, 23

- EULA, 32, 52

- Firewall, 30
- Free Software Foundation, 4

- Gefährdungen, 15
 - Angriffe, 16
- GNU, 35
- GPL, 5, 32, 35, 55
- Grundschutzhandbuch, 17
- GSH, 15

- Haftung, 17, 32, 49, 52
- Hardware, 20

- ICMP, 27
- IDS, 46
- Integrität, 10, 11
- IP, 26
- IP-Spoofing, 26
- IPSec, 26
- ISO, 9

- ISP, 24

- Konstruktionsprinzipien
 - complete Mediation, 7
 - Economy of Mechanism, 6
 - fail-safe Defaults, 6
 - least common Mechanism, 7
 - least Privilege, 6
 - open Design, 8
 - psychological Acceptability, 8
 - Separation of Privilege, 7
- Kryptographie, 8

- Maskierung, 26

- Netzwerk, 9, 24, 29
- NGO, 2
- NIST, 9
- NSA, 9

- OSI, 9

- PAP, 26
- PPP, 25
- PPPoE, 25

- RAID, 22

- Schichtenmodell, 9
 - Anwendung, 29
 - Netzwerkschicht, 26
 - Transport, 27
 - Verbindung, 24
- Schutzbedarf, 15
- Security by Obscurity, 8
- Selbstbestimmung, 13, 47
- Sicherheit, 10
 - erweiterter Begriff, 12, 47
 - technische, 14
- Software

- Hersteller, 31
- Kategorien, 4, 31
- SSH, 30
- SSL, 27, 30, 39

- TCP, 27, 29
- TCP/IP, 10
- TCPA, 23
- thin-Ethernet, 24
- Transmission Control Protocol, *sie-*
he Transport Control Pro-
tocol
- Tunnel, 28

- UDP, 27–29
- Unsicherheit, 12
- User Datagram Protocol, 27

- Verbindlichkeit, 10, 11
- Verfügbarkeit, 10, 11, 22
- Verschlüsselung, 27
- Vertraulichkeit, 10, 11

- WCMS, 14
- WLAN, 24

- Zugriffskontrolle
 - Administratoren, 44
 - Anonyme Besucher, 41
 - Direktveröffentlicher, 42
 - eingetragene Benutzer, 41
 - Lektoren, 43

Literatur

- [ano00] ANONYMOUS: *Linux Hacker's Guide*. Markt&Technik, 2000.
- [BSI01] BSI - BUNDESAMT FÜR SICHERHEIT IN DER INFORMATIONSTECHNIK: *IT-Grundschutzhandbuch*, 2001.
- [Bun01] BUNDESMINISTERIUM DER JUSTIZ: *Gesetz über die Nutzung von Telediensten - TDG*, 12 2001. Gesetzestext.
- [Dav88] DAVIDSON, JOHN: *An Introduction to TCP/IP*. Springer-Verlag, New York Berlin Heidelberg, 1988.
- [Eck03] ECKERT, CLAUDIA: *IT-Sicherheit*. Oldenbourg Wissenschaftsverlag, München, Wien, 2003.
- [FSF91] FSF - THE FREE SOFTWARE FOUNDATION, INC.: *The GNU General Public License*, 1991. <http://www.fsf.org/licenses/gpl.html>.
- [FSF02] FSF - THE FREE SOFTWARE FOUNDATION, INC.: *Categories of Free and Non-Free Software*, 12 2002. <http://www.gnu.org/philosophy/categories.html>.
- [Gra02] GRASSMUCK, VOLKER: *Freie Software zwischen Privat- und Gemeineigentum*. BPB - Bundeszentrale für politische Bildung, 1. Auflage, 2002.
- [Hei99] HEINRICH, L. J.: *Informationsmanagement*. In: RECHENBERG, PETER und GUSTAV POMBERGER [RP99], Kapitel F4, Seiten 1065–1080.
- [Hol02] HOLUPIREK, ALEXANDER: *Die Trusted Computer Platform Alliance*. Arbeit zum Seminar Informationspolitik, 2002.
- [Hol03] HOLUPIREK, ALEXANDER: *Aufbau eines Internetportals mit freier Software*, 5 2003.
- [Kuh98] KUHLEN, RAINER: *Trust - Vertrauen, Informationsethische Basis elektronischen Marktgeschehens*, 1998. ISSN 0942-2625.
- [Kuh99] KUHLEN, RAINER: *Was bedeutet informationelle Autonomie oder wie kann Vertrauen in elektronische Dienste in offenen Informationsmärkten gesichert werden?* Suhrkamp Taschenbücher Wissenschaft, Frankfurt, 1999.

- [Kuh02] KUHLEN, RAINER: *Informationskompetenz und Vertrauen als Grundlage informationeller Autonomie und Bildung. Was bedeutet die fortschreitende Delegation von Informationsarbeit an Informationsassistenten?* Thomas Christaller and Josef Wehner, 08 2002.
- [MF90] MILLER, BARTON und JUSTIN E. FORRESTER: *An Empirical Study of the Robustness of Windows NT Applications Using Random Testing*. Technischer Bericht, Computer Sciences Department, University of Wisconsin, 1990.
- [Mic03] MICROSOFT, INC.: *Microsoft Windows 2000 Pro End user license agreement*, 2003. C:\winnt\system32\eula.txt.
- [MKL⁺95] MILLER, BARTON, DAVID KOSKI, CJIN PHEOW LEE, VIVEKANANDA MAGANTY, RAVI MURTHY, AJITKUMAR NATARAJAN und JEFF STEIDL: *Fuzz Revisited: A Re-examination of the Reliability of UNIX Utilities and Services*. Technischer Bericht, Computer Sciences Department, University of Wisconsin, 1995.
- [MvOV01] MENEZES, ALFRED J., PAUL C. VAN OORSCHOT, and SCOTT A. VANSTONE: *Handbook of applied cryptography*. CRC Press, 5. edition, 2001.
- [PS99] PLATTNER, B. und P. SCHULTHESS: *Rechnernetze*. In: RECHENBERG, PETER und GUSTAV POMBERGER [RP99], Kapitel C6, Seiten 381–407.
- [RP99] RECHENBERG, PETER und GUSTAV POMBERGER (Herausgeber): *Informatik-Handbuch*. Carl Hanser Verlag, 2. Auflage, 1999.
- [Sch02] SCHNEIER, BRUCE: *Palladium and the TCPA*. <http://www.counterpane.com/crypto-gram-0208.html#1>, 8 2002.
- [Sie99] SIEBER, ULRICH: *Verantwortlichkeit im Internet*. Verlag C. H. Beck, München, 1999.
- [SS75] SALTZER, JEROME H. und MICHAEL D. SCHROEDER: *The Protection of Information in Computer Systems*. Proceedings of the IEEE, 63(9):1278–1308, Sep 1975.

- [Sta02] STALLMAN, RICHARD M.: *Some Confusing or Loaded Words and Phrases that are Worth Avoiding*. <http://www.gnu.org/philosophy/wordstoavoid.html#DigitalRightsManagement>, 10 2002.
- [SV03] STEINWEDE, ANDREAS und AXEL VAHLDIK: *Sicherungskopie - Strategien gegen Datenverlust*. c't - Magazin für Computertechnik, 8:156–158, 2003.
- [TCPA02] TRUSTED COMPUTER PLATFORM ALLIANCE, TCPA: *Official Website of TCPA*. <http://www.trustedcomputing.org/tcpaasp4/index.asp>, 10 2002.

Danke

Prof. Rainer Kuhlen und Joachim Griesbaum für die Förderung unserer (Alex Holupireks und meiner) Fähigkeiten und Interessen

Elvira Weber für so ziemlich alles

Alex Holupirek für harsche Kritik, den Boden unter den Füßen und den Blick aus einer anderen Richtung

Sebastian Grünert, Benno Buchczyk, Nils Wigglinghaus und Goliath für ein Leben abseits des Schreibtisches

Christian Panse, Sebastian Rexhausen für scharfe Augen

Die Freie Software Gemeinde: Unter anderem Bram Moolenaar et al. for *vim*; D. E. Knuth et al. for $\text{T}_{\text{E}}\text{X}/\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$; Glyph & Cog, LLC for *xpdf*; Rob Caelters, Raymond Penners for *workrave*; ...