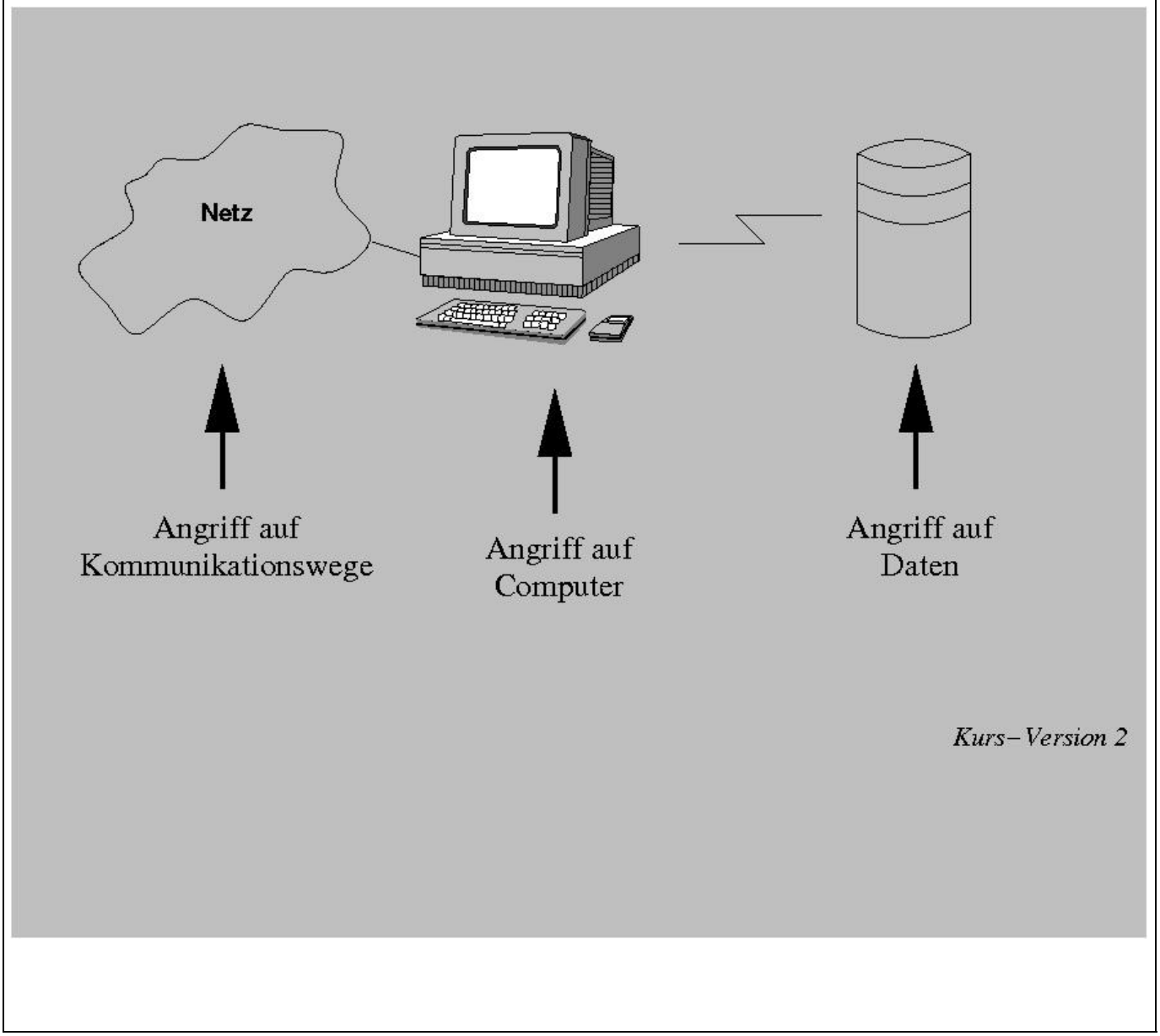
Kurs 1866: Sicherheit im Internet

Kurseinheit 1: Sicherheit in der Informationstechnik

Autor: Stefan Wohlfeil



000 000 000

Inhalt

Kurseinheit 1

Sicherheit in der Informationstechnik

Der Autor: Dr. Stefan Wohlfeil, geb. 12.12.1964

- Studium der Informatik mit Nebenfach Elektrotechnik an der Universität Kaiserslautern (1984–1991)
- Wissenschaftlicher Mitarbeiter am Lehrgebiet Praktische Informatik VI der FernUniversität Hagen (1991–1998)
- Promotion zum Dr. rer. nat. (1997)
- Mitarbeiter in der Deutsche Bank AG, Abteilung TEC — The Advanced Technology Group (1998–2002)
- Professor an der Fachhochschule Hannover, Fakultät IV, Abteilung Informatik; Arbeitsgebiet: Sichere Informationssysteme (seit 2002)

1.1 Einführung

Liebe Fernstudentin, lieber Fernstudent,
herzlich willkommen beim Kurs über Sicherheit im Internet!

Diese Einführung soll Ihnen einen Überblick darüber geben, worum es im vorliegenden Kurs geht. Der Kurs liefert eine Einführung in das Gebiet der *Sicherheit*. Dabei wird es um einzelne Computer, vernetzte Computer und das *Internet* gehen. Sie erfahren, welche Sicherheitsprobleme dort existieren und welche Möglichkeiten Sie haben, sich diesen Problemen entgegenzustellen.

Inhalt des Kurses und Vorkenntnisse: Dieser Kurs richtet sich an Informatik-Studierende[1] und setzt die Kenntnis einiger Inhalte aus einem Informatik-Grundstudium voraus. Konkret sollten Sie bereits wissen, wie ein Computer prinzipiell aufgebaut ist, was ein Betriebssystem typischerweise macht

[1] Hierzu gehören alle Studierenden deren Curriculum einen Informatikbestandteil enthält wie beispielsweise auch Studierende der Wirtschaftsinformatik.

und welche Möglichkeiten sich durch die Vernetzung, wie beispielsweise im Internet, für Anwender bieten. Diese Themen werden im Kurs *(01801) Betriebssysteme und Rechnernetze* behandelt.

Die Kurseinheit 1 beschäftigt sich mit den Grundlagen des Themas *Sicherheit*. Die folgenden Fragen werden diskutiert:

- *Warum* ist das Thema Sicherheit überhaupt bedeutsam?
- Was *bedeutet* der Begriff „Sicherheit“ im Zusammenhang mit Computern eigentlich?
- Welche Probleme sind zu lösen?

In der Literatur werden eine Reihe von „klassischen Bedrohungen“ vorgestellt, auf die im Kurs auch eingegangen wird. Weiterhin wird der Aufbau und die Funktionsweise des Internets kurz vorgestellt. Konkret werden einige wichtige Protokolle und Dienste des Internet besprochen. Den Abschluß der ersten Kurseinheit bildet eine Beschreibung von ausgewählten konkreten Bedrohungen der Computersicherheit, wie beispielsweise Viren.

Die Kurseinheit 2 beschäftigt sich mit den Grundfunktionen, die die Hardware von Systemen und die Betriebssysteme selbst im Bereich Sicherheit anbieten. Außerdem werden dort die Grundlagen von Verschlüsselungsverfahren erklärt. Sie sind für die Lösung des *Vertraulichkeits-* und des *Integritäts-Problems* ein wichtiges Hilfsmittel. Es wird weiterhin auf Authentifizierungsverfahren eingegangen. Dazu gehört auch das Konzept der *digitalen Signaturen*.

In Kurseinheit 3 wird das Thema Sicherheit aus der Perspektive eines World Wide Web (WWW) Benutzers beleuchtet. Die Fragen, wie versende/empfange ich sicher eine email, bzw. wie surfe ich sicher im Netz werden beantwortet. Oft muß man auch an Rechnern arbeiten, die weit entfernt vom eigenen Schreibtisch stehen. Der Studentenrechner `bonsai` der FernUniversität ist ein Beispiel hierfür. Wie man das sicher tun kann wird auch in Kurseinheit 3 erklärt. Abschließend bekommen Sie einige Hinweise, wie Sie Ihren privaten PC zu Hause sicherer machen können.

Die Kurseinheit 4 beschäftigt sich mit dem Thema Sicherheit aus der Perspektive eines System-Administrators und eventuell auch Web-Anbieters. Hier werden Verfahren und Systeme vorgestellt, mit denen ein internes Netz (auch Intranet genannt) so an das Internet angeschlossen wird, daß keine unbefugten Zugriffe und Modifikationen möglich sind. Neben dem Konzept der *firewall* wird auch auf organisatorische und prozedurale Aspekte eingegangen.

Ergänzende Materialien: Das Thema Sicherheit im Internet ist derart umfangreich, daß es in diesem Kurs nur in Ausschnitten behandelt werden kann. Ziel des Kurses ist es, daß Sie die Grundlagen des Themengebietes kennenlernen und Sie sich dann darauf aufbauend tiefer in die Materie einarbeiten können. Dazu gibt es verschiedene weitere Informationsquellen.

Bücher

Die Menge an Büchern zum Thema Security wächst sehr schnell. Ausgehend vom Literaturverzeichnis dieses Kurses sollten Sie in der Universitätsbibliothek das eine oder andere Buch ausleihen und durchschauen. Aktuellste Bücher

kann man bei den verschiedenen Buchhändlern im Internet suchen. Dort findet man u. U. auch Rezensionen der Bücher vor.

Überhaupt ist das Internet eine nahezu unerschöpfliche Quelle an Informationen zum Thema Security. Im Kurs werden eine Reihe von Verweisen auf interessante Seiten im Internet genannt. Wenn Sie Zugang zum Internet haben, nehmen Sie sich doch die Zeit und schauen sich die eine oder andere Seite an. Ich hoffe, daß die Verweise noch stimmen, wenn Sie den Kurs lesen. Das Internet ändert sich ständig, so daß es gut sein kann, daß Sie einmal eine Seite nicht finden. In diesem Fall sollten Sie eine der vielen Suchmaschinen wie *altavista*, *fireball*, *google*, *yahoo* usw. konsultieren. Vielleicht hat sich ja nur die Adresse der Seite leicht verändert. Informieren Sie dann auch bitte die Kursbetreuer, damit der Kurstext aktualisiert werden kann. Die Namen und Kontaktmöglichkeiten der Kursbetreuer wurden Ihnen im Anschreiben zusammen mit dieser Kurseinheit genannt.

Internet

An der FernUniversität Hagen ergänzen der Kurs *(01868) Sicherheit im Internet 1 – Ergänzungen* und der Kurs *(01867) Sicherheit im Internet 2* diesen Kurs. In Kurs *(01867) Sicherheit im Internet 2* werden unter anderem diese Themen besprochen:

Fortsetzungskurse im nächsten Semester

- Konkrete Bedrohungen und Angriffe gegen Rechner
- Bezahlverfahren im Internet
- Authentisierung durch biometrische Merkmale
- Virtual Private Networks (VPN)
- Intrusion Detection Systeme (IDS)
- Advanced Hashing
- Überblick über wichtige Gesetze und Verordnungen, die im Internet von besonderer Bedeutung sind
- Entwurf und Implementierung sicherer Systeme

In Kurs *(01868) Sicherheit im Internet 1 – Ergänzungen* werden dann diese Themen behandelt:

- Computer Forensik
- Anonymität
- Biometrie
- Zugriffskontrollen, Benutzerauthentisierung
- Sicherheit in Telekommunikationsnetzen (GSM, Voice over IP)
- Aktive Inhalte (ActiveX, Java, JavaScript)

Informationen zu den Prüfungsmöglichkeiten dieses Kurses finden Sie in den Studien- und Prüfungsinformationen Ihres Studienganges an der FernUniversität oder bei der Studienberatung. Dort erfahren Sie, in welchen Studiengängen dieser Kurs eingesetzt wird, in welchen Modulen dieser Kurs ein Bestandteil ist, in welcher Form Sie Leistungsnachweise oder Prüfungen ablegen können bzw. müssen usw.

1.2 Einleitung

1.2.1 Warum ist Sicherheit erforderlich?

Das Thema „Sicherheit in der Informationstechnik“ hat in den letzten Jahren mehr und mehr an Bedeutung gewonnen. Einer der Hauptgründe dafür ist die große Popularität des Internet. Für viele Menschen ist das Internet nicht nur das Informationsmedium, als das es ursprünglich entwickelt wurde, sondern immer öfter auch das Medium für private Geschäftstätigkeiten aller Art. Bücher, Flug- und Eisenbahn-Tickets können nicht nur im Geschäft gekauft werden sondern auch bequem von zu Hause aus. Über virtuelle Auktionshäuser wie beispielsweise *eBay* werden inzwischen täglich enorme Mengen von Gütern aller Art versteigert. Neben Privatleuten beteiligen sich auch Gewerbetreibende als Bieter, ebenso wie als Anbieter. Auch Bankgeschäfte wie Überweisungen, Einrichten/Ändern von Daueraufträgen oder sogar An- und Verkäufe von Wertpapieren lassen sich über das Internet abwickeln. Die Vorteile für die Konsumenten sind vielfältig:

private Geschäftstätigkeiten

- Man ist nicht mehr an die Ladenöffnungszeiten gebunden, sondern kann rund um die Uhr tätig sein.
- Man muß nicht mehr persönlich im Geschäft vorbei schauen, sondern kann seine Geschäfte bequem von zu Hause erledigen.
- Verschiedene Angebote lassen sich einfacher vergleichen. Die Konkurrenz ist immer nur „einen Mausklick“[2] entfernt.

Aber nicht nur Privatleute, sondern auch viele Firmen benutzen das Internet für ihre Geschäftszwecke. War man zuerst nur durch eine „web site“ präsent und hat sich und seine Produkte vorgestellt, so nutzt man das Internet bzw. die Internet-Technologie heute auch für die Abwicklung von Geschäften. In der Automobil-Industrie sind die Hersteller und ihre Zulieferer über das Internet miteinander verbunden und tauschen so beispielsweise Bestellungen aus. Der Vorteil für die Unternehmen besteht darin, daß viele solcher Tätigkeiten automatisierbar sind. Dadurch lassen sich Kosten einsparen. Den Umfang der wirtschaftlichen Bedeutung, die das Internet heute (Januar 2009) erreicht hat verdeutlichen die folgenden Zahlen:

wirtschaftliche Bedeutung

[2] Tatsächlich ist es nicht ganz so einfach. Die wenigsten Anbieter werden einen Verweis (engl. **link**) auf ihre Konkurrenz mit anbieten. Man muß diese Adressen also erst einmal finden. Suchmaschinen, wie *google*, oder Verzeichnisdienste, wie *Yahoo*, helfen bei der Suche.

- Alleine in Deutschland wurden 2003 ca. 12,4 Millionen Bankkonten online geführt.[3] Das sind etwa 80% der Girokonten bei privaten Banken. Bis zum Ende des Jahres 2006 hat sich die Zahl der online geführten Konten auf 35,3 Millionen (von ca. 91 Millionen Konten) erhöht. Etwa jeder zweite Internet-Benutzer führt sein Konto online.

- Im Jahr 2002 wurden in Deutschland etwa 767 Millionen Online-Überweisungen getätigt. Dabei wurden etwa 660 Milliarden Euro bewegt[4]. Im Jahr 2006 wurden bereits etwa 1,8 Milliarden Online-Überweisungen getätigt und dabei 1685 Milliarden Euro bewegt.

- Das U.S. Census Bureau veröffentlicht Statistiken aus den USA. Im dritten Quartal des Jahres 2008 wurden etwa 34,4 Milliarden US-Dollar Umsatz im E-commerce gemacht. Abbildung 1.1 zeigt die stetig steigenden Umsatzzahlen sowie die Spitzen zu Weihnachten. Dabei zählt ein Handel als E-commerce, wenn entweder die Bestellung oder die Preisvereinbarung über das Internet oder ein anderes elektronisches Netz (z. B. Extranet oder EDI) erfolgt.

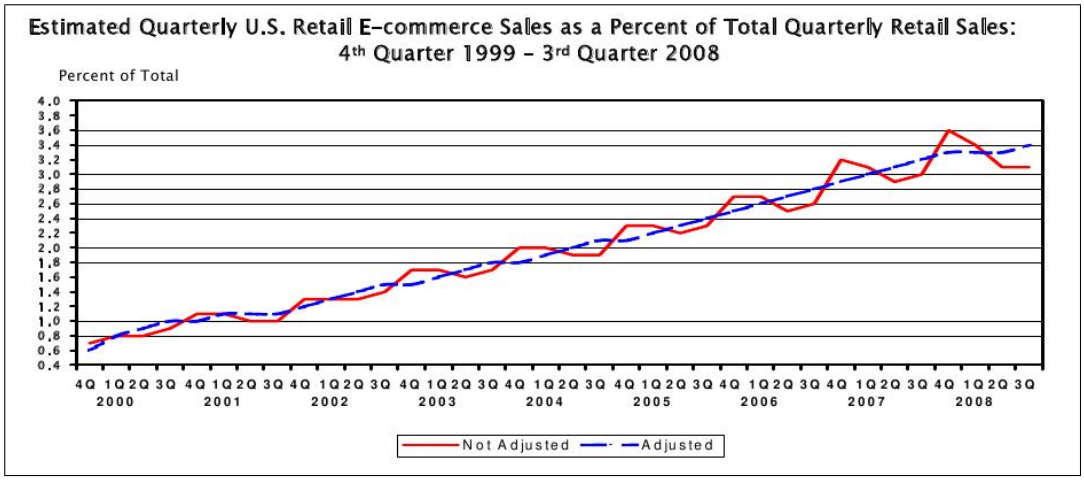


Abbildung 1.1: E-Commerce Anteil an den Gesamtumsätzen in den USA

- Der Buchhändler *Amazon* hat nach eigenen Angaben im Weihnachtsgeschäft 2008 am Spitzentag 15.12.08 weltweit mehr als 6,3 Millionen Bestellungen angenommen. Das entspricht etwa 73 Bestellungen pro Sekunde. Auf Deutschland entfielen davon etwa 1,2 Millionen Produktbestellungen, was etwa 14 Bestellungen pro Sekunde entspricht. Im Jahr 2007 hat Amazon knapp 14,9 Milliarden US-Dollar Umsatz (engl. **net sale**) gemacht. Im Jahr 2008 stieg der Umsatz trotz der Finanz- und Wirtschaftskrise auf ca. 19,2 Milliarden US-Dollar.

[3] Quelle: Bundesverband Deutscher Banken (`http://www.bdb.de/`)

[4] Quelle: Deutsche Bundesbank (`http://www.bundesbank.de/`): *Statistiken zum Zahlungsverkehr 1998-2002 (Stand Januar 2004)* und *Statistiken zum Zahlungsverkehr 2002-2006 (Stand Januar 2008)*

- Der Online-Auktionator *eBay* hat im Jahr 2007 Einnahmen (engl. **net revenue**) in Höhe von 7,6 Milliarden US-Dollar gehabt.[5] Der Gesamtwert der Waren die über *eBay* verkauft wurden betrug etwa 60 Milliarden US-Dollar.

Man sieht, daß das Internet eine große wirtschaftliche Bedeutung gewonnen hat und daß die wirtschaftliche Bedeutung kontinuierlich wächst. Ohne die entsprechende Sicherheit der beteiligten Informations-Technologie-Systeme (IT-Systeme) könnten also immense wirtschaftliche Probleme entstehen.

Aktuelle Sicherheitsprobleme: Sicherheit ist aber auch deshalb ein wichtiges Thema, weil heute schon viele Sicherheitsprobleme auftreten. Sie stören die normale Computernutzung, richten nicht unerhebliche Schäden an und finden daher auch mehr und mehr Beachtung in der Presse. Ein **Computer Emergency Response Team (CERT)** ist eine Anlaufstelle, die Meldungen über Sicherheitsvorfälle entgegen nimmt und über mögliche Abwehrmaßnahmen informiert. Im Internet findet man unter der Adresse `http://www.cert.org/` eine Statistik der gemeldeten Vorfälle.

Computer Emergency Response Team (CERT)

In Deutschland betreibt das Deutsche Forschungsnetz (DFN) auch solch ein Team. Zu seinen Aufgaben gehören neben der Betreuung der DFN-Mitglieder auch die Bereitstellung von Informationen über Sicherheitsvorfälle und Hilfsmitteln zur Bekämpfung von Sicherheitsproblemen. Außerdem betreibt es eine Zertifizierungsstelle (siehe auch Abschnitt 2.7). Im Internet finden sie es unter der Adresse `http://www.cert.dfn.de/`

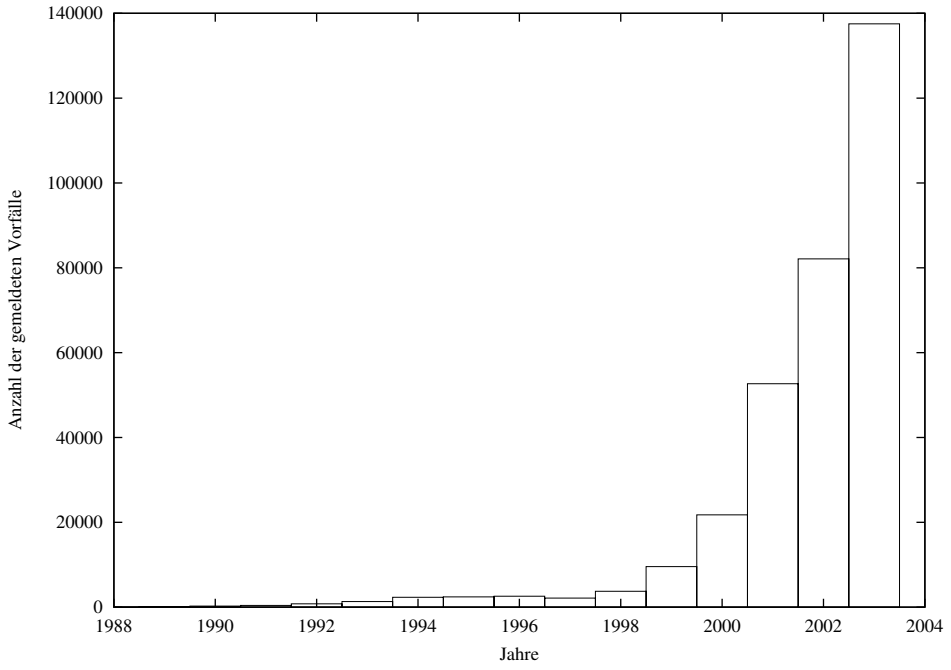


Abbildung 1.2: Dem CERT gemeldete Sicherheitsvorfälle

[5] Quelle: eBay Annual Report 2007

Abbildung 1.2 zeigt graphisch die Zahlen, die das CERT als Tabelle im Internet publiziert. Die steigende Zahl der Vorfälle kann nicht alleine auf ein gestiegenes Bewußtsein bzgl. Computersicherheit zurück geführt werden. Wenn mehr Personen auf Sicherheitsvorfälle achten, dann werden natürlich auch mehr Vorfälle erkannt. Eine leichte Steigung der gemeldeten Vorfälle wäre also auch bei einer eigentlich konstanten Zahl an Vorfällen zu erwarten. Dadurch alleine läßt sich der Anstieg der Zahlen in Abbildung 1.2 allerdings nicht erklären. Es ist tatsächlich eine steigende Zahl an Sicherheitsvorfällen vorhanden.

Hierfür gibt es zwei wesentliche Gründe: (1) Ein Grund ist die Verfügbarkeit von einfach zu bedienenden Angriffswerkzeugen, beispielsweise Virus-Konstruktions-Programmen oder konkreten „hacker tools“. Damit kann auch ein weniger qualifizierter Angreifer Schwachstellen in Systemen ausnutzen. In diesem Fall spricht man oft von sog. **script kiddies**, die verfügbare Programme zur Ausnutzung von Schwachstellen laufen lassen und beliebige Rechner angreifen. (2) Der zweite Grund ist wirtschaftlicher Natur. Der massenhafte Versand von unerwünschten Werbe-emails (engl. **spam**) ist für die Versender ein potentiell lukratives Geschäft. Vergleichsweise niedrigen Kosten steht ein großer Nutzen gegenüber. Allerdings versuchen Anwender permanent, die steigende Flut von Werbe-emails einzudämmen. Beispielsweise werden emails von bekannten „Spammern“ grundsätzlich blockiert. Dies kann aber dadurch umgangen werden, indem die „Spammer“ nicht unter eigenem Namen agieren, sondern erfolgreich angegriffene Rechner und deren Benutzer hierzu mißbrauchen. Die unerwünschten Werbe-emails kommen dann nicht mehr von einem Rechner (den man als Absender evtl. nicht zulassen kann), sondern von beliebigen, vorher nicht bekannten Rechnern von Privatnutzern. Hacker vermieten solche **Bot-Netze** an zahlende Kunden um deren Werbung über die gehackten Rechner zu verbreiten.

script kiddies

Bot-Netze

Das *SANS Institute* (siehe `http://www.sans.org/`) veröffentlicht jedes Jahr eine Liste mit den *Top 20 Security Risks*. Sie gruppiert die Liste der Probleme des Jahres 2007 in

- Client-side Vulnerabilities (Web Browsers, Office Software, Email Clients, Media Players, etc.)
- Server-side Vulnerabilities (Web Applications, Windows Services, UNIX and Mac OS Services, Backup Software, Anti-virus Software, Management Servers, Database Software, etc.)
- Security Policy and Personnel (Excessive User Rights and Unauthorized Devices, Phishing, Unencrypted Laptops and Removable Media, etc.)
- Application Abuse (Instant Messaging, Peer-to-Peer Programs, etc.)
- Network Devices (VoIP Servers and Phones, etc.)
- Zero Day Attacks

Man sieht, daß es eine Vielzahl von Schwachstellen und Problemen gibt. Da ständig neue Probleme auftauchen und glücklicherweise einige alte Probleme gelöst scheinen ändert sich auch die Klassifikation der Liste mit der Zeit.

Conficker

Ein Beispiel für ein großes Sicherheitsproblem aus dem Jahr 2008 ist der Wurm *Conficker* alias Downadup. Er nutzt eine Schwachstelle in allen Betriebssystemen von MS Windows 2000 bis MS Windows Server 2008 aus. Der Remote Procedure Call (RPC) Dienst enthält einen Fehler, so daß ein Angreifer eine spezielle Nachricht an den Rechner schicken kann, um dann die komplette Kontrolle über diesen Rechner auszuüben. Der Angreifer braucht keine Benutzerkennung oder irgendein Paßwort auf dem System zu kennen oder auszuspähen. Bereits im Jahr 2003 hatte der Wurm *MSBlaster* alias Lovsan eine vergleichbare Lücke in MS Windows ausgenutzt und großen Schaden angerichtet. Der RPC Dienst wird bei der Datei- und Druckfreigabe in MS Windows benutzt.

Sich persitent machen

Als erstes kopiert sich der Wurm als DLL-Datei in das Betriebssystem des infizierten Rechners, trägt sich in die Registry ein und sorgt somit dafür, daß er beim Systemstart immer wieder neu gestartet wird.

In ihrem Malware Protection Center beschrieb *Microsoft* die weiteren Auswirkungen von Conficker wie folgt:

> This malware mostly spreads within corporations but also was reported by several hundred home users. It opens a random port between port 1024 and 10000 and acts like a web server. It propagates to random computers on the network by exploiting MS08-067. Once the remote computer is exploited, that computer will download a copy of the worm via HTTP using the random port opened by the worm. The worm often uses a .JPG extension when copied over and then it is saved to the local system folder as a random named dll.
>
> It is also interesting to note that the worm patches the vulnerable API in memory so the machine will not be vulnerable anymore. It is not that the malware authors care so much about the computer as they want to make sure that other malware will not take it over too...

Als Verbreitungs-Server arbeiten

Andere aussperren

Jeder infizierte Rechner wird somit zum neuen Verteiler für den Wurm im Gegensatz zu anderer Schadsoftware, die ein neu infizierter Rechner immer von demselben Server laden würde.

Anschließend ruft der Wurm HTML-Seiten von einigen Web-Servern ab, um daraus das aktuelle Datum und die Uhrzeit zu entnehmen. Außerdem erfährt er so, welche im Internet sichtbare IP-Adresse der infizierte Rechner hat. (Der Rechner könnte in einem privaten Netz mit privaten IP-Adressen liegen und über einen Router der Network Address Translation (NAT) durchführt mit dem Internet verbunden sein.) Eine URL zum o. g. Web-Server auf dem infizierten Rechner verteilt der Wurm dann an andere Rechner die er zu infizieren versucht.

verbreiten

Letztlich versucht der Wurm, sich (1) im lokalen Netz, auf angeschlossene USB devices, etc. zu verbreiten und (2) schickt der Wurm dann auch noch An-

fragen an einige zufällig generierte URLs um von dort weitere (Schad-)Software nachzuladen. Diese URLs werden zufällig generiert, damit man die Rechner nicht einfach sperren kann. Ein Hacker kann nun eine dieser zufällig generierten DNS-Domains für sich registrieren, dort Schadsoftware aufspielen und abwarten bis infizierte Rechner versuchen diese Schadsoftware zu laden und zu starten. Die Firma *F-Secure* hat einige solche DNS-Domains für sich registriert und dann gezählt wieviele infizierte Computer dorthin Anfragen schicken. Am 16. Januar 2009 schätzte man bei *F-Secure* die Zahl der mit dem Conficker-Wurm infizierten Rechner auf ca. 8,9 Millionen!

Das Nachladen von Funktionen haben die Autoren von Conficker noch zusätzlich geschützt. Die nachzuladenden (Schad-)Funktionen müssen mit einem speziellen RSA-Schlüssel (siehe Abschnitt 2.4.2) digital signiert sein. Die Autoren von Schutzsoftware können diesen Nachlademechanismus also nicht dazu benutzen, dem Wurm eine „Selbstzerstörungsfunktion“ unter zu schieben.

Der einzige Schutz vor dem Wurm besteht darin, den Betriebssystem-Patch von Microsoft rechtzeitig zu installieren.

Schutz

Gesetzliche Verpflichtungen: Am 1. Mai 1998 trat das Gesetz zur Kontrolle und Transparenz im Unternehmensbereich (KonTraG) in Kraft. Es zwingt börsennotierte Aktiengesellschaften zur Risikovorsorge durch die Etablierung eines Risiko-Management- und Kontrollsystems. Das heißt, daß diese Firmen gezwungen sind, sich mit den Risiken, die durch ihre IT-Systeme entstehen können, auseinander zu setzen. Natürlich reicht es nicht, sich nur mit den Risiken auseinander zu setzen. Die Unternehmen müssen die erkannten Gefahrenpotentiale auch verringern. Da 100-prozentige Sicherheit nicht erreichbar ist, müssen die Restrisiken auf ein vertretbares Maß reduziert werden. Verbleibende Gefahren sollen durch Überwachungsmaßnahmen kalkulierbar werden.

KonTraG

Im Geschäftsleben hat die Risikovorsorge aber auch handfeste finanzielle Gründe. Banken müssen für jeden Kredit, den sie vergeben, einen bestimmten Anteil an Eigenkapital vorhalten. Das dient dazu, daß beim Zahlungsausfall eines Gläubigers die Bank nicht in Liquiditätsprobleme kommt. Die Höhe dieses Eigenkapitalanteils richtet sich nach der Höhe des Risikos. Kreditnehmer mit hohem Risiko zwingen eine Bank zu höheren Eigenkapitalrücklagen. Daher werden Banken solchen Kreditnehmern ungünstigere Zinskonditionen anbieten. Das *Basel Committee on Banking Supervision* hat solche Vorschriften unter dem Namen **Basel II** erarbeitet. Insbesondere die sog. operationellen Risiken werden zukünftig beurteilt. Ein Kreditnehmer, dessen Geschäfte stark von der IT abhängen und der wenig IT-Sicherheitsvorkehrungen trifft, muß also mit höheren Kreditkosten rechnen.

Basel II

Aber auch Sie persönlich müssen aufpassen, daß Angreifer Ihren Rechner nicht dazu mißbrauchen, anderen einen Schaden zuzufügen. Möglicherweise tragen Sie daran eine Mitschuld, wenn Sie die erforderliche Sorgfalt mißachten. Dann könnten Sie auf Schadenersatz verklagt werden.

Persönliche Arbeitserleichterung: Sobald Sie auf dem eigenen Rechner ein Sicherheitsproblem haben, müssen Sie sich darum kümmern. Falls Sie nichts

tun, können Sie Probleme der vielfältigsten Art bekommen. Ihr Rechner arbeitet nicht mehr für Sie, er schädigt andere Benutzer oder ähnliches. Um diese Probleme zu vermeiden müssen Sie also etwas tun. Das kostet mindestens Zeit, die Sie möglicherweise mit Erfreulicherem verbringen wollen. Daher ist es einfacher, Probleme von Anfang an zu vermeiden, als sich später mit der Problembehebung auseinander zu setzen. In der Medizin heißt das: „Vorbeugen ist besser als heilen."

1.2.2 Was heißt eigentlich Sicherheit?

Umgangssprachlich versteht man unter Sicherheit in der Regel einen Zustand ohne Gefahren. Im dtv-Lexikon ist Sicherheit wie folgt definiert:

> **Sicherheit, 1)** Zivilrecht: Bürgschaft, Pfand oder jeder Vermögenswert, der zur Sicherheitsleistung gebracht wird. **2)** objektiv das Nichtvorhandensein von Gefahr, subjektiv die Gewißheit, vor möglichen Gefahren geschützt zu sein.

Für diesen Kurs ist die juristische Bedeutung uninteressant. Die zweite Interpretation unterscheidet zwischen *objektiver Sicherheit* und *subjektiver Sicherheit*. In beiden Fällen ist von Gefahren die Rede. Im Rahmen von IT-Sicherheit spricht man an Stelle von Gefahren häufiger von **Bedrohungen**. Neben den eher abstrakten Bedrohungen sind auch die potentiellen **Schäden** zu betrachten. Diese lassen sich in der Regel einfacher quantifizieren. In Abschnitt 1.2.3 wird kurz vorgestellt, *was* eigentlich bedroht ist. Es geht konkret um die Ziele von **Angriffen** (engl. **attacks**). In Abschnitt 1.2.4 werden einige Kriterien vorgestellt, anhand derer man Bedrohungen klassifizieren kann. Anschließend geht es in Abschnitt 1.2.5 um die „klassischen Bedrohungen" der IT-Sicherheit. Hier werden die typischen Probleme, wie Integrität, Vertraulichkeit, Authentizität usw. diskutiert.

Bedrohungen

Schäden

Angriffe

1.2.3 Angriffsziele

Die Sicherheit von IT-Systemen kann auf verschiedene Weise gefährdet sein. Abbildung 1.3 zeigt, an welchen Stellen die Sicherheit gefährdet werden kann. Der erste Angriffspunkt liegt an den Zugangswegen zu einem Computer. In einem vernetzten System wie dem Internet sind nicht alle Kommunikationswege unter der Kontrolle einer einheitlichen Instanz. Deshalb ist damit zu rechnen, daß einzelne Übertragungsleitungen abgehört werden. So kann jemand Ihre Kommunikation „mitlesen".

Der zweite Angriffspunkt ist der Computer selbst. Bei Mehr-Benutzer-Computern können andere Benutzer Programme laufen lassen, die evtl. Ihre eigenen Programme beeinträchtigen oder Ihre eigenen Daten verfälschen oder ausspionieren. Außerdem könnte der Computer lahmgelegt werden und dadurch nicht mehr für Sie zur Verfügung stehen.

Das dritte Ziel von Angreifern sind die Daten selbst. Durch unbefugten Zugriff auf Datenbanken oder die Datenträger (Festplatten, Bänder, CD-ROM)

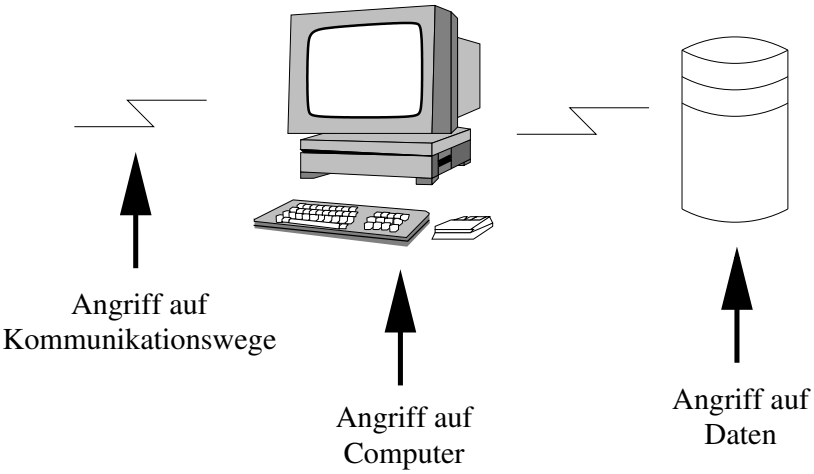


Abbildung 1.3: Ziele von potentiellen Angriffen auf die Sicherheit

können Daten ausspioniert oder manipuliert werden. Aber auch durch die Messung von elektromagnetischen Wellen, die von einem Bildschirm immer[6] abgestrahlt werden, kann man den Bildschirminhalt (und damit die Daten auf dem Bildschirm) rekonstruieren.

1.2.4 Systematik der Bedrohungen

nicht-technische und technische Bedrohungen

Bei den Bedrohungen kann man unterschiedliche Klassen differenzieren. Ein erstes Unterscheidungskriterium ist, ob es sich um *technische Bedrohungen* handelt oder ob eine *nicht-technische Bedrohung* vorliegt. Zu den einfachen technischen Bedrohungen gehört z. B. die kosmische Strahlung. Auch wenn es eher selten vorkommt, so kann diese Strahlung den Wert einzelner Bits verändern und so Daten verfälschen. Auch andere elektrische Probleme, z. B. auf Übertragungsleitungen, oder elektromagnetische Probleme bei Funkübertragungen gehören hierzu. Diesen Problemen kann man durch die Einführung von Redundanz, beispielsweise durch Prüfbits oder fehlerkorrigierende Codes begegnen. Ein nicht-technisches Problem wäre dagegen eine Person, die absichtlich Daten verfälscht und dann überträgt.

beabsichtigte und unbeabsichtigte Bedrohungen

Dies führt zum zweiten Unterscheidungsmerkmal. Bedrohungen können *beabsichtigt* oder *unbeabsichtigt* entstehen. Eine unbeabsichtigte Bedrohung ist es, wenn ein Mitarbeiter aus Unwissenheit ein wichtiges Paßwort auf einen post-it Zettel schreibt und diesen auf den Monitor des Computers klebt. Im allgemeinen zählen die meisten Bedienungsfehler zu dieser Kategorie. Im Gegensatz dazu ist jede Form von Spionage zu den beabsichtigen Bedrohungen zu zählen. Auch die sogenannten *denial of service attacks* sind beabsichtigte Bedrohungen. Hier versuchen Angreifer einen Dienst für die legitimen Benutzer nicht mehr verfügbar zu machen, beispielsweise durch Systemüberlastungen.

aktive und passive Bedrohungen

Ein drittes Merkmal unterscheidet *aktive* und *passive* Bedrohungen. Zu den passiven Bedrohungen zählt jede Form des Abhörens. Funkübertragungen lassen sich durch eine zusätzliche Antenne einfach mitschneiden und beeinträch-

[6] Zumindest bei Bildschirmen auf Basis von Elektronenstrahlröhren.

tigen den regulären Empfang nicht. Auch das heute sehr oft benutzte *Ethernet* (siehe auch Abschnitt 1.3.1) überträgt in seiner Grundform jedes Datenpaket an alle angeschlossenen Computer. Normalerweise ignoriert ein Computer alle Datenpakete, die nicht an ihn selbst adressiert sind. Es ist jedoch sehr einfach, einen Computer so zu programmieren, daß er alle Pakete an einen bestimmten anderen Computer im Netz mitprotokolliert. Passive Bedrohungen sind schwer zu entdecken. Im Gegensatz dazu wird bei aktiven Bedrohungen direkt eingegriffen. Das Erzeugen neuer Nachrichten, das Unterdrücken oder Verzögern von Nachrichten und die Fälschung von Nachrichten sind aktive Bedrohungen. Sie lassen sich i. d. R. einfacher entdecken als passive Bedrohungen.

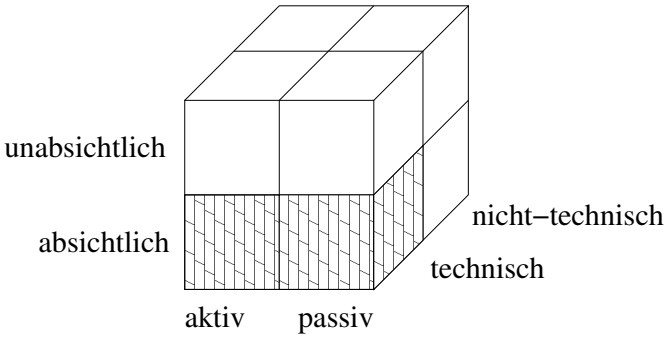


Abbildung 1.4: Klassifikation von Bedrohungen

Abbildung 1.4 visualisiert die Systematik der Bedrohungen. Im Rahmen dieses Kurses wird der Schwerpunkt der Betrachtungen auf den gemusterten Bereichen (vorne unten im Würfel) liegen.

Übungsaufgabe 1.1 Ordnen Sie die folgenden Bedrohungen in die gerade vorgestellte Systematik ein:

1. Ausfall der Stromversorgung durch Witterungseinflüsse.
2. Datenverlust durch Versand einer Diskette mit der Post an eine falsche Adresse.
3. Von einem Girokonto wird unerlaubt Geld auf ein anderes Konto überwiesen. Der Auftrag wurde über das Internet eingegeben.

1.2.5 Klassische Bedrohungen

Hauptzweck der Erstellung von Computernetzen ist es, den Austausch von Daten zwischen verschiedenen Computern so einfach wie möglich zu machen. In diesem Abschnitt wird daher von einer normalen, ungestörten Kommunikation wie in Abbildung 1.5 ausgegangen. Ein Sender verschickt eine Nachricht an

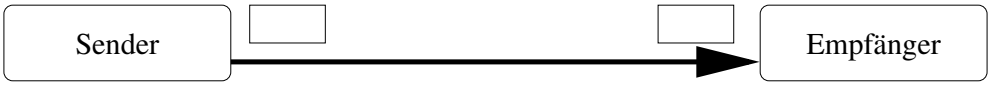


Abbildung 1.5: Ungestörter Nachrichtenaustausch

einen Empfänger und die Nachricht kommt dort unversehrt an. In den folgenden Abschnitten wird dann aufgezeigt, welche Probleme beim Datenaustausch auftreten können.

Unbefugter Informationsgewinn: Der unbefugte Informationsgewinn ist ein Angriff auf die **Vertraulichkeit** (engl. **confidentiality**) der übertragenen Daten. Bei der elektronischen Abwicklung von Geschäften, z. B. dem Kauf von Aktien, möchten die beteiligten Parteien i. d. R. nicht, daß Dritte davon erfahren. Der unbefugte Informationsgewinn ist ein passiver Angriff der absichtlich oder unabsichtlich erfolgen kann. Abbildung 1.6 zeigt das Prinzip des unbefugten Informationsgewinns.

Vertraulichkeit

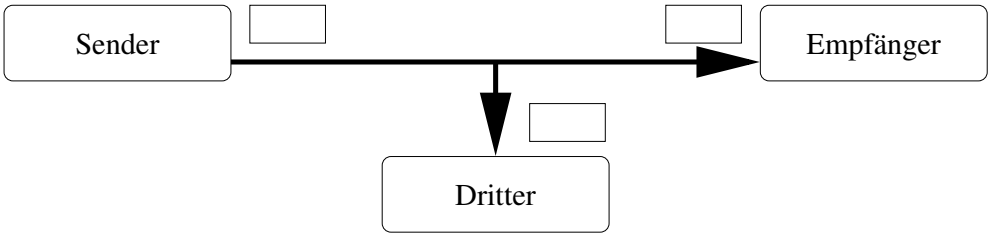


Abbildung 1.6: Unbefugter Informationsgewinn

Das „klassische“ Gegenmittel gegen diese Bedrohung besteht darin, die Nachricht für den Dritten unleserlich, unverständlich oder unerkennbar zu machen. Durch **Verschlüsselung** kann man Nachrichten unverständlich machen. Versteckt man eine Nachricht in einer anderen, unverfänglichen und i. d. R. größeren Nachricht, so spricht man von **Steganographie**.

Verschlüsselung

Steganographie

Der unbefugte Informationsgewinn ist nicht auf das Abhören von Datenübertragungen begrenzt. Wie bereits in Abschnitt 1.2.3 dargestellt, kann sich ein Unbefugter auch direkt Zugang zu einem Computer mit wichtigen Daten verschaffen. Diese kann ein Angreifer dann einsehen oder kopieren und mitnehmen. In großen Rechenzentren sind die Computer daher in verschlossenen Räumen untergebracht. Der Zugang zu diesen Räumen ist genau reglementiert und wird u. U. durch Sicherheitsschleusen geregelt. Durch die steigende Verwendung von mobilen Computern, wie Notebooks oder *Persönlichen Digitalen Assistenten (PDAs)*, steigt jedoch die Gefahr von Diebstählen nicht nur der Geräte, sondern auch der darauf gespeicherten Daten.

PDA

Weiterhin ist unbefugter Informationsgewinn auch dadurch möglich, daß sich ein Angreifer Zugang zu den Datenträgern selbst verschafft. Die Festplatten sind normalerweise genauso sicher untergebracht wie die Computer selbst. Aber die Sicherungen, z. B. auf Magnetbändern, werden aus Sicherheitsgründen auch an anderen Orten als der Computer selbst gelagert. Eventuell sind diese Orte nicht so gut gesichert wie ein Rechenzentrum.

Unbefugte Modifikation: Die unbefugte Modifikation ist ein Angriff auf die **Integrität** (engl. **integrity**) der übertragenen Daten. Eine vom Sender als Kaufauftrag abgeschickte Nachricht könnte unterwegs von einem Dritten in einen Verkaufsauftrag umgewandelt werden. Die unbefugte Modifikation ist

Integrität

ein aktiver Angriff der i. d. R. absichtlich erfolgt. Doch auch durch technische Probleme ist eine unbeabsichtigte Modifikation möglich.

Abbildung 1.7: Unbefugte Modifikation

Abbildung 1.7 zeigt das Prinzip der unbefugten Modifikation einer Nachricht. Der Empfänger erhält eine andere Nachricht als der Sender abgeschickt hat.

Gegen unbefugte Modifikationen kann man sich auf verschiedene Arten schützen. Durch die Einführung von Redundanz kann ein Empfänger erkennen, ob Daten auf dem Transportweg verändert wurden. Dazu werden an den Nutzinhalt zusätzliche Daten gehängt, die bestimmten Bedingungen genügen. Ein Beispiel hierfür sind die Paritätsbits von Hauptspeicher-Bausteinen. Bei gerader Parität wird das Paritätsbit so gesetzt, daß die Zahl der Einsen in einem Datenwort (einschließlich Paritätsbit) gerade ist. Folgende Gründe machen dieses Verfahren im Bereich der Sicherheit jedoch nicht einsetzbar:

- Einfache Verfahren erkennen nur bestimmte Veränderungen an den Daten, z. B. nur die Veränderung eines Bits. Andere Veränderungen bleiben dagegen unentdeckt.
- Bei einer beabsichtigten Modifikation der Daten kann ein Angreifer nicht nur die Nutzdaten sondern auch die Redundanzdaten verändern. Für den Empfänger sieht die Nachricht dadurch korrekt aus. Voraussetzung hierfür ist, daß der Angreifer das Verfahren kennt, mit dem die Redundanzdaten berechnet werden.

Eine weitere Schutzmöglichkeit ist wiederum die Verschlüsselung der Daten mit einem geeigneten Verfahren. Ohne den Schlüssel zu kennen, kann ein Angreifer dann keine sinnvolle Veränderung der Daten vornehmen. Verändert man wahllos einige Bytes einer gut verschlüsselten Nachricht, so entsteht bei der Entschlüsselung meistens etwas Unleserliches.

Angriffe auf die Integrität der Daten können nicht nur bei der Übertragung von Daten auftreten. Auch hier kann ein Unbefugter durch den direkten Zugang zu einem Computer, auf dem wichtige Daten gespeichert sind, diese Daten verändern. Eine weitere Gefährdung der Integrität von Daten sind ungeübte Benutzer. Durch die falsche Bedienung der Programme können Daten unbeabsichtigt modifiziert werden.

Authentizität

Unbefugte Erzeugung: Die unbefugte Erzeugung von Nachrichten ist ein Angriff auf die **Authentizität** (engl. **authenticity**). Dabei erzeugt jemand eine Nachricht und gibt darin vor, jemand anderes zu sein. Beispielsweise könnte Ihr Nachbar eine Bestellpostkarte auf Ihren Namen ausfüllen und damit bei irgendeinem Versandhändler eine Bestellung in Ihrem Namen aufgeben. Es kommt auch immer noch häufig vor, daß Personen einen Anruf von einem vorgeblichen Bankmitarbeiter erhalten. Dieser fragt den Angerufenen dann mit

einer vorgeschobenen Begründung nach der Geheimzahl der ec-Karte. Geben Sie Ihre Geheimzahl oder ein Paßwort *niemals* jemand anderem preis!

Abbildung 1.8 zeigt das Prinzip der unbefugten Erzeugung. Ohne daß der Sender aktiv wird, erhält der Empfänger eine Nachricht. Die unbefugte Erzeugung ist ein aktiver Angriff und erfolgt überwiegend absichtlich. Er kann bei Konfigurations- oder Bedienungsfehlern auch unabsichtlich erfolgen.

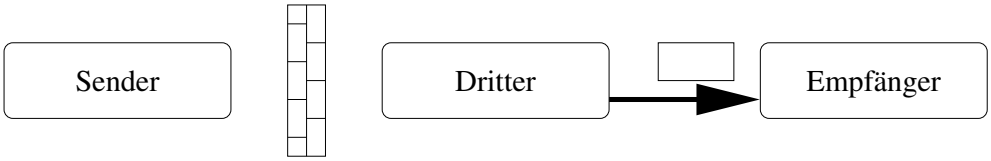


Abbildung 1.8: Unbefugte Erzeugung von Nachrichten

Normalerweise überzeugt man sich von der Identität einer Person durch einen Ausweis, wie z. B. einen Personalausweis. Um festzustellen, ob ein Brief tatsächlich vom vorgeblichen Absender kommt, bedient man sich der *eigenhändigen Unterschrift*. Diese Methoden haben den Vorteil, daß sie schwer zu fälschen oder zu kopieren sind. Man kann mit dem Auge recht gut feststellen, ob man ein Originaldokument oder eine Fotokopie vor sich hat. In der digitalen Welt der Computer kann man den Unterschied zwischen „Original-Bit“ und „Kopie-Bit“ jedoch nicht mehr feststellen. Man ist dort auf andere Verfahren angewiesen, auf die wir in Abschnitt 2.6 noch genauer eingehen werden.

Das Problem der Authentizität ist auch eng verwandt mit dem Problem der **Nicht-Zurückweisbarkeit** (engl. **non-repudiation**) von Nachrichten. Dabei geht es darum, daß weder der Sender noch der Empfänger die stattgefundene Kommunikation nachträglich abstreiten (zurückweisen) können. Konkret bedeutet dies, daß

Nicht-Zurückweisbarkeit

- der Empfänger beweisen kann, daß die Nachricht tatsächlich vom vorgegebenen Absender kommt (vorhandene eigenhändige Unterschrift) und
- der Sender beweisen kann, daß die Nachricht tatsächlich beim geplanten Empfänger und nicht bei jemand anderem angekommen ist (vgl. Einschreiben mit Rückschein bei der Post).

Unbefugte Unterbrechung: Die unbefugte Unterbrechung ist ein Angriff auf die **Verfügbarkeit** (engl. **availability**) von Daten, Computern und Kommunikationsmitteln. Wenn Ihr Nachbar Ihr Telefonkabel durchschneidet, so steht Ihnen dieses Kommunikationsmittel nicht mehr zur Verfügung. Insbesondere bei zeitkritischen Geschäften ist die Verfügbarkeit der Systeme sehr wichtig. Aktienkurse können sich beispielsweise sehr schnell ändern und eine Verzögerung Ihrer Kauf-/Verkaufsorder durch die Nicht-Verfügbarkeit der Kommunikationsstrecke kann enorme wirtschaftliche Konsequenzen haben. Angriffe auf die Verfügbarkeit sind aktive Angriffe, die i. d. R. absichtlich erfolgen.

Verfügbarkeit

Angriffe auf die Verfügbarkeit können sich auch auf Computer selbst richten. Dabei werden speziell konstruierte Nachrichten an einen Computer geschickt, sodaß bei der Bearbeitung der Nachricht das Betriebssystem abstürzt.

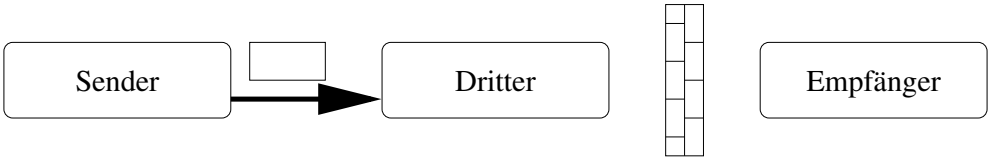


Abbildung 1.9: Unbefugte Unterbrechung

Bis der Computer neu gestartet ist (evtl. ist dazu ein manueller Eingriff erforderlich!) stehen die Dienste dieses Computers nicht mehr zur Verfügung.

Das klassische Gegenmittel gegen diese Attacken besteht darin, sogenannte „Hochverfügbare Systeme“ (engl. **high availability**) aufzustellen. Dem eigentlichen Rechnersystem stellt man ein zweites, redundantes System mit identischer Konfiguration zur Seite. Fällt ein System aus, übernimmt das andere den Betrieb. Das macht man nicht nur bei Rechnern, sondern auch anderen Komponenten, wie Stromversorgung, Festplatten, Internet-Anschlüssen usw. Bei Ausfällen aufgrund technischer Probleme (z. B. Ausfall der Festplatte, Blitzeinschlag, o. ä.) hat sich dieses Mittel bewährt. Gezielte Attacken, die Fehler im Betriebssystem ausnutzen und den Computer dadurch zum Absturz bringen, kann man durch ein identisches Back-up System nicht bekämpfen.

Allgemein gesagt entwirft man Architekturen, die sich nicht durch einen Fehler an einer einzelnen Stelle lahm legen lassen. Solche Stellen nennt man auf Englisch **single point of failure**. Sie sind zu vermeiden. Das Thema sichere Architekturen wird im Kurs *(01867) Sicherheit im Internet 2* ausführlicher besprochen.

single point of failure

Zusammenfassung der zu schützenden Eigenschaften:

Vertraulichkeit (engl. **confidentiality**)**:** Daten sind nur für befugte Personen zugänglich.

Integrität (engl. **integrity**)**:** Daten sind korrekt und unverändert.

Authentizität (engl. **authenticity**)**:** Daten stammen vom vorgeblichen Erzeuger.

Verfügbarkeit (engl. **availability**)**:** Daten können von befugten Personen gelesen/bearbeitet werden.

1.2.6 Nicht-technische Aspekte von Sicherheit

Im vorigen Abschnitt wurde der Begriff der Sicherheit in dem Sinne vorgestellt, daß ein sicheres System für den Benutzer *verläßlich* sein muß. Konkret bedeutet dies, daß die Funktionsweise des Systems den gestellten Anforderungen genügt. Der Benutzer kann sich auf die Korrektheit der Ergebnisse (z. B. der übertragenen Daten oder der Identität des Kommunikationspartners) ebenso verlassen wie auf die Verfügbarkeit des Systems. Neben dieser technischen Sicht gibt es auch die Sicht der Betroffenen. Aus dieser Sicht ist ein System sicher, wenn es für die Betroffenen *beherrschbar* ist. Dies bedeutet, daß der Einzel-

Verläßlichkeit

Beherrschbarkeit

ne und auch die gesamte Gesellschaft vor den unerwünschten Auswirkungen neuer Technologien und Systeme geschützt werden müssen [MR99].

Neue Technologien und Systeme verändern das Verhalten der Menschen, die diese Systeme benutzen. Die neuen Möglichkeiten der Informationsverarbeitung beeinflussen auch die Struktur der Gesellschaft. Die Erfindung des Buchdrucks mit beweglichen Lettern durch Gutenberg hat schon einmal das Leben der Menschen und die Struktur der Gesellschaft grundlegend verändert. Ähnliches kann durch die Verbreitung von Personal-Computern (PCs) und des Internet wieder vorkommen. Seit dem 11. September 2001 werden in den USA zum Zweck der Terrorabwehr vermehrt Daten gesammelt (Patriot Act).

Dazu kommen die Daten, die viele Menschen freiwillig von sich selbst in „Sozialen Netzwerken“ wie *XING*, *StudiVZ*, *facebook*, *myspace* usw. einstellen. Neben Namen, Anschrift und weiteren Kontaktdaten werden dort auch berufliche Informationen und persönliche Hobbys und Interessen eingetragen. Manchmal findet man dort auch einen Spitznamen einer Person der in eigentlich anonymen Diskussionsforen als Benutzerkennung auftaucht. Die Beiträge unter dieser Benutzerkennung könnten dann der Person mit dem Spitznamen zugeordnet werden. Der Schutz der Privatsphäre (engl. **privacy**) eines Menschen ist durch IT-Technik möglicherweise gefährdet.

privacy

Wenn man einmal zusammenstellt, in welchen Bereichen Daten über Personen erfaßt werden, ergibt sich eine lange Liste:

Finanzdaten: Wo hat man Konten? Wieviel verdient jemand? Wohin wird Geld überwiesen? Wo hebt man Geld ab, d. h. wo hält man sich auf? Wo kauft man ein (und bezahlt mit Karte)?

Konsumdaten: Mit Hilfe von Rabattkartensystemen erfassen Geschäfte die Einkäufe ihrer Kunden. Was wird wann und wo gekauft? Daraus lassen sich beispielsweise Ernährungsgewohnheiten ableiten, an denen Krankenkassen Interesse haben könnten.

Kommunikationsdaten: Mit wem telefoniert jemand, wer bekommt email von wem? Telefongespräche und email können abgehört werden (und werden es auch). Bei Mobiltelefonen kann man auch orten, wo sich der Teilnehmer aufhält. Wohin surfen Benutzer im Internet? In welchen Diskussionsgruppen beteiligen sich Personen?

Aufenthaltsdaten: Die Zahl von Überwachungskameras (Flughafen, Kaufhaus, öffentliche Plätze, Hotels, Banken usw.) steigt ständig und gibt Auskunft über den Aufenthaltsort von Personen. Bei Flugreisen muß man sich auch ausweisen und in den USA u. U. auch biometrische Daten (Fingerabdrücke) von sich selbst abgeben.

Persönliche Informationen: In sozialen Netzen findet man neben dem Namen einer Person oft auch Fotos, Informationen zur Ausbildung (den eigenen Kenntnissen und Fertigkeiten), zum Arbeitsplatz (in welcher Firma, welche Position, welche Tätigkeit) und zu den sonstigen Interessen, z. B. den Hobbys.

data mining

Diese Liste ist nicht vollständig. Trotzdem können diese Datenmengen zusammengefaßt und mit Hilfe von **data mining** durchsucht und korreliert werden. Daraus kann man

> erschreckend komplette Dossiers über sie erhalten: über Lebensgewohnheiten, Interessen und Vorlieben, Lebensstile, persönliche Probleme und sexuelle Orientierungen, politische Neigungen, finanzielle Verhältnisse, Familienstand usw. [DIE ZEIT Nr. 34, 12. August 2004]

Der Aufbau der Computer- und Netz-Technologie ist für den Einzelnen im Grundsatz verständlich. Trotzdem ist das Ausmaß der Veränderungen für die Gesellschaft nur zum Teil vorhersehbar. Die Auswirkungen des E-commerce sind nur schwer vorhersehbar, insbesondere die Auswirkungen auf die bisher als Verkäufer oder allgemein im Vertrieb tätigen Menschen.

Software-Fehler

Daneben gibt es ein weiteres Problem. Die technische Zuverlässigkeit von Computern (sie können riesige Datenmengen in sehr kurzer Zeit fehlerfrei verarbeiten, z. B. Zahlenkolonnen addieren oder Gleichungssysteme lösen) verleitet die Benutzer zu einer unzulässigen Verallgemeinerung. Es resultiert ein falsches Vertrauen in die Objektivität der Datenverarbeitung nach dem Motto: „Was der Computer berechnet hat wird schon stimmen."

Die Vernetzung der Systeme und die immer weiter steigende Komplexität der Systeme stellt die Beherrschbarkeit in Frage. Da man nur noch selten das gesamte System durchschaut und versteht, übersieht man möglicherweise Schwachstellen und Mißbrauchsmöglichkeiten. Und welcher PC-Benutzer weiß heute schon, was sein Betriebssystem im Hintergrund so alles macht? Ist man wirklich sicher, daß private Daten nicht heimlich an Dritte übertragen werden?

Neben der technischen Sicht der Sicherheit, also der *Verläßlichkeit* von Systemen darf auch die Benutzer-(bzw. Betroffenen-)Sicht, also die Frage der *Beherrschbarkeit* von Systemen nicht vergessen werden. In diesem Kurs geht es allerdings nicht um Fragen der Beherrschbarkeit, sondern zunächst um die Verläßlichkeit der Systeme.

Interessenkonflikt

An dieser Stelle soll auch erwähnt werden, daß die Wahrnehmung von Gefahren, d. h. letztlich die Sicherheit, auch von der Perspektive des Wahrnehmenden abhängt. Ein Netzbetreiber hält es eventuell für gefährlich, wenn jemand anonym Zugang zum Netz erhält. Wie könnte sich der Betreiber vor unliebsamen Teilnehmern schützen, wer bezahlt den Betreiber für die in Anspruch genommenen Dienste?

Auf der anderen Seite möchten Sie als „Web-Surfer" nicht unbedingt mit Ihrem wahren Namen auftreten. Dadurch verhindern Sie, daß ein Aktivitäts- und Interessenprofil von Ihnen erstellt wird. Ein aus Ihrer Sicht sicheres System erlaubt also Anonymität, ein aus Sicht des Betreibers sicheres System würde Anonymität lieber verbieten.

Solche widersprüchlichen Sichten ergeben sich nicht nur bei verschiedenen Beteiligten, sondern auch aus den unterschiedlichen Situationen, in denen ein Beteiligter agiert. Während einer Recherche oder bei anderer Informationsbeschaffung möchten Sie gerne anonym bleiben. Auf der anderen Seite wollen

und erwarten Sie beispielsweise bei Bankgeschäften, daß Sie und die Bank keinesfalls anonym bleiben. Sie möchten wissen, daß Sie auch tatsächlich mit der Bank kommunizieren und von einer Bank erwarten Sie, daß die Bank keine anonymen Aufträge zu Lasten ihres Kontos ausführt.

Übungsaufgabe 1.2 Welches sind die vier zu schützenden Eigenschaften in einem System? Erklären Sie deren Bedeutung mit eigenen Worten.

1.3 Netze

In diesem Abschnitt werden der Aufbau und die Struktur von Computernetzen, insbesondere des Internets diskutiert. Die im Internet verfügbaren Dienste und die dort benutzten Kommunikationsprotokolle werden vorgestellt. Es wird auf die einzelnen Bereiche jedoch nur so weit eingegangen, wie es für das Verständnis des Kurses erforderlich ist.

1.3.1 Lokale Netze

Schon seit Jahren werden Computer miteinander vernetzt. Alle Computer einer Abteilung, eines Lehrgebietes oder einer beliebigen anderen geschlossenen Benutzergruppe werden in einem lokalen Netz (engl. **(LAN) Local Area Network**) miteinander verbunden. So können die Benutzer bestimmte Ressourcen wie Drucker oder Dateien gemeinsam benutzen und sehr einfach Daten austauschen. In lokalen Netzen werden unterschiedliche Topologien eingesetzt. Im folgenden stellen wir diese Topologien kurz vor und betrachten dabei insbesondere die Auswirkungen auf die Sicherheit.

Netztopologien

Sterntopologie: In einem Netz mit Sterntopologie werden alle Computer an einen zentralen Punkt angeschlossen (siehe Abbildung 1.10). Aufgrund der Ähnlichkeit mit einem Rad nennt man den Mittelpunkt auch Nabe (engl. **hub**). Ein Beispiel für ein Sternnetz ist das ATM (engl. **Asynchronous Transfer Mode**) der Telefongesellschaften. Der zentrale Punkt ist ein elektronischer Vermittler (engl. **switch**), der dedizierte Verbindungen zwischen den angeschlossenen Computern schaltet. Die Kommunikationsdaten fließen also nur vom Sender zum Switch und zum Empfänger. Auf die Sicherheit hat dies folgende Auswirkungen:

- Fällt ein Computer aus, so können die anderen Computer weiterhin kommunizieren (Verfügbarkeit).
- Fällt der Switch aus, so ist keine Kommunikation mehr möglich (Verfügbarkeit).
- Da die Daten nur den Sender, den Switch und den Empfänger passieren, können andere Computer diese Kommunikation nicht stören oder abhören (Vertraulichkeit).

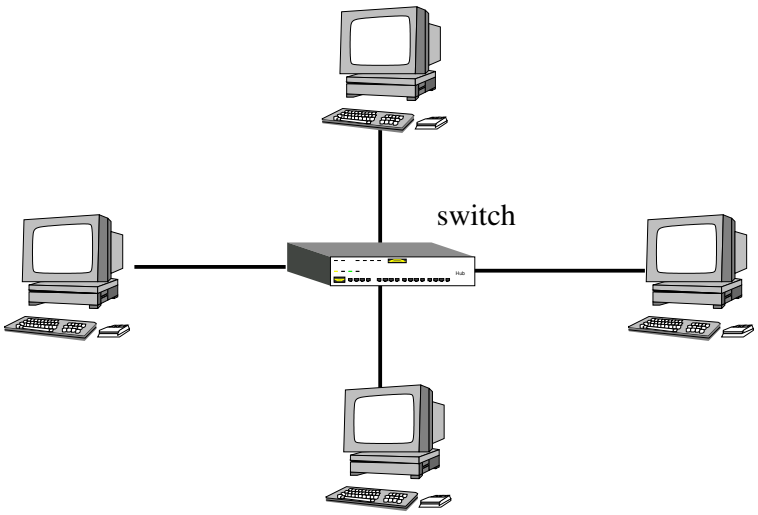


Abbildung 1.10: Prinzip der Sterntopologie

Das setzt allerdings voraus, daß der Switch die Adressen der angeschlossenen Systeme kennt. Ethernet-Switches lernen diese Adressen im laufenden Betrieb. Trifft ein Paket ein, so steht darin die Adresse des Absenders. Soll der Switch anschließend ein anderes Paket an genau diese Adresse schicken, dann kennt der den zugehörigen Anschluß (engl. **port**). Der Switch ist also darauf angewiesen, daß die angeschlossenen Computer kooperativ sind und nicht beispielsweise gefälschte Absenderadressen benutzen.

Ringtopologie: Sind alle Computer eines lokalen Netzes in einer geschlossenen Schleife angeordnet, so spricht man von einer Ringtopologie (siehe Abbildung 1.11). Ein Beispiel für ein Ring-Netz ist der *Token Ring* von *IBM*. Die

token

Computer benutzen eine spezielle Nachricht, genannt **token**, um die Nutzung zu koordinieren. Will ein Computer Daten verschicken, so wartet er auf das token und sendet dann die Daten an seinen „rechten“ Nachbar. Die Nachricht wird von Computer zu Computer weitergereicht. Der Empfänger erstellt eine Kopie der Nachricht, und der Absender erhält die Nachricht wieder zurück. Danach gibt der Sender das token, und somit die Sendeerlaubnis, an seinen „rechten“ Nachbarn weiter. Will kein Computer senden, kreist das token mit hoher Geschwindigkeit. Auf die Sicherheit hat dies folgende Auswirkungen:

- Fällt ein Computer oder *eine einzige* Verbindung aus, so ist die Kommunikation unterbrochen (Verfügbarkeit).
- Eine Nachricht passiert alle angeschlossenen Computer und kann im Prinzip auf jedem Computer gelesen werden (Vertraulichkeit). Jeder Computer könnte die Nachricht auch verfälschen (Integrität) oder unterdrücken.

Bustopologie: Bei der Bustopologie sind alle Computer an das gleiche Kabel angeschlossen (siehe Abbildung 1.12). Ein Beispiel für ein Bus-Netz ist das

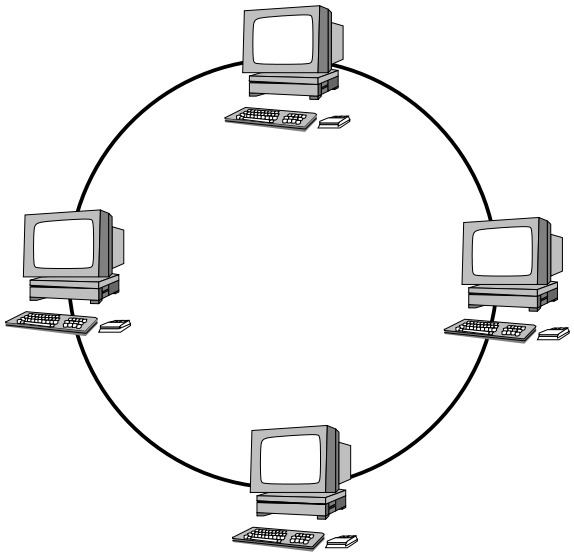


Abbildung 1.11: Prinzip der Ringtopologie

Ethernet. Es kann in seiner Grundform die Daten mit 10 MBit/s Geschwindigkeit übertragen. Inzwischen gibt es auch *Fast-Ethernet* mit 100 MBit/s Übertragungsrate und auch *Gigabit-Ethernet* mit 1 GBit/s Übertragungsrate. Wenn ein Computer Daten versendet, dann können alle an das Kabel ange-

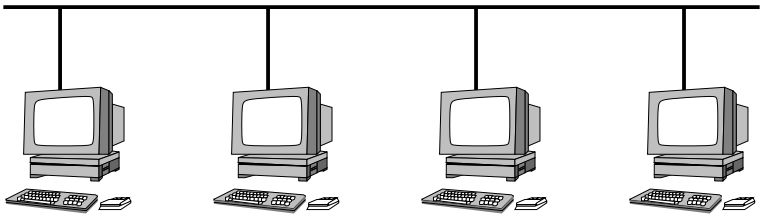


Abbildung 1.12: Prinzip der Bustopologie

schlossenen Computer diese Nachricht lesen. Wollen zwei Computer gleichzeitig senden, so kommt es zu einer Kollision, und die Computer müssen die Übertragung abbrechen und später wieder aufnehmen. Auf die Sicherheit hat dies folgende Auswirkungen:

- Fällt ein Computer aus, so können die anderen Computer weiterhin miteinander kommunizieren (Verfügbarkeit).
- Läuft ein Computer „Amok“, so können die anderen Computer u. U. nicht mehr kommunizieren, da die Leitung belegt ist (Verfügbarkeit).
- Jeder angeschlossene Computer kann *alle* Nachrichten auf dem Kabel mitlesen (Vertraulichkeit).

1.3.2 Vernetzte Netze

Lokale Netze können nicht beliebig groß werden. Außerdem gibt es nicht „die beste“ Netztechnologie für alle Anwendungsfälle. Es besteht also der Bedarf

unterschiedliche Netze miteinander zu verbinden. Dazu stehen folgende Gerätetechnologien zur Verfügung:

Repeater: Ein Repeater ist ein Gerät, das zwei gleichartige lokale Netze miteinander verbindet und zu einem Gesamt-Netz macht. Dazu verstärkt und übertragt der Repeater *alle* Signale (also auch Störungen) zwischen den beiden Netzen.

Bridge: Eine Bridge arbeitet im Prinzip wie ein Repeater. Sie gibt aber nicht alle Signale weiter, sondern leitet nur vollständige und fehlerfreie Datenrahmen weiter. Daher kann eine Bridge anhand der Absenderadressen in den Rahmen mit der Zeit lernen, welcher Computer in welchem Netz liegt. Die Bridge leitet die Rahmen daher nur dann weiter, wenn der Zielcomputer in einem anderen Netz liegt.

switch

Dies setzt voraus, daß die Struktur und die Datenrahmen in beiden Netzen gleich sind. Für ein Ethernet realisieren sogenannte Switches eine Menge von einelementigen Subnetzen für jeden angeschlossenen Computer, der über Bridges mit allen einelementigen Subnetzen verbunden ist. Dadurch kann die Bustechnologie Ethernet sternförmig verkabelt sein.

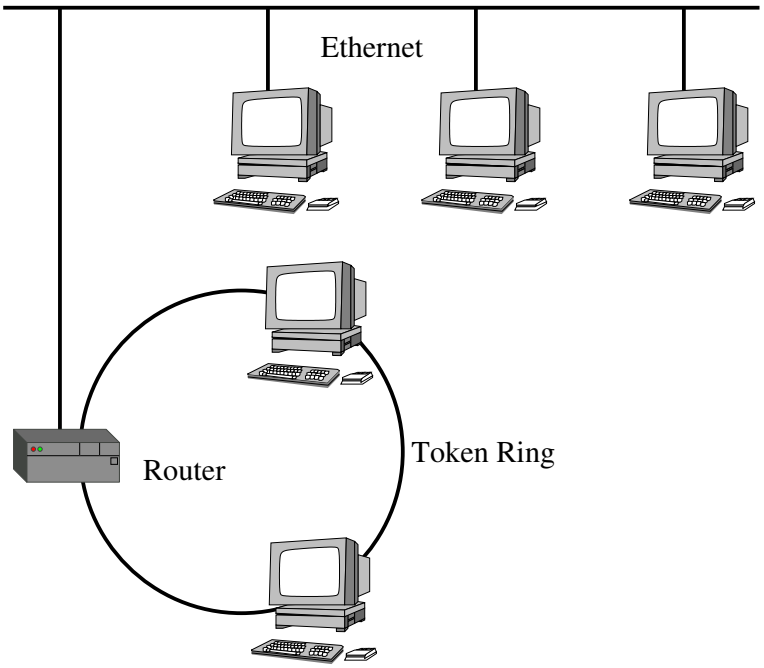


Abbildung 1.13: Verbindung von Ethernet und Token Ring durch einen Router

Router: Ein Router ist ein dedizierter Computer für den Zusammenschluß von Netzen. Er kann Netze unterschiedlicher Technologien mit verschiedenen Medien, Adreßschemata oder Rahmenformaten verbinden. Dazu hat er eine getrennte Schnittstelle für jeden Netzanschluß, d. h. eine Netzwerkkarte für jeden Netztyp. Ein Router kann also ein Busnetz wie das Ethernet mit einem Ringnetz wie dem Token Ring verbinden (siehe Abbildung 1.13).

1.3.3 Das Internet-Protokoll

Damit man Daten zwischen unterschiedlichen Netzen austauschen kann, muß man sich vorher auf bestimmte Dinge wie Datenformate, Adressierung, Verbindungsaufbau usw. einigen. Diese Einigungen werden in einem sogenannten **Protokoll** festgehalten. Die am häufigsten implementierte Protokollfamilie ist das *TCP/IP*.

Protokoll

Protokollschichten: Die Protokolle der TCP/IP Familie sind hierarchisch in Schichten organisiert. In Abbildung 1.14 sind die Teile dargestellt, die bei einer Kommunikation beteiligt sind. Auf der obersten Ebene stehen die An-

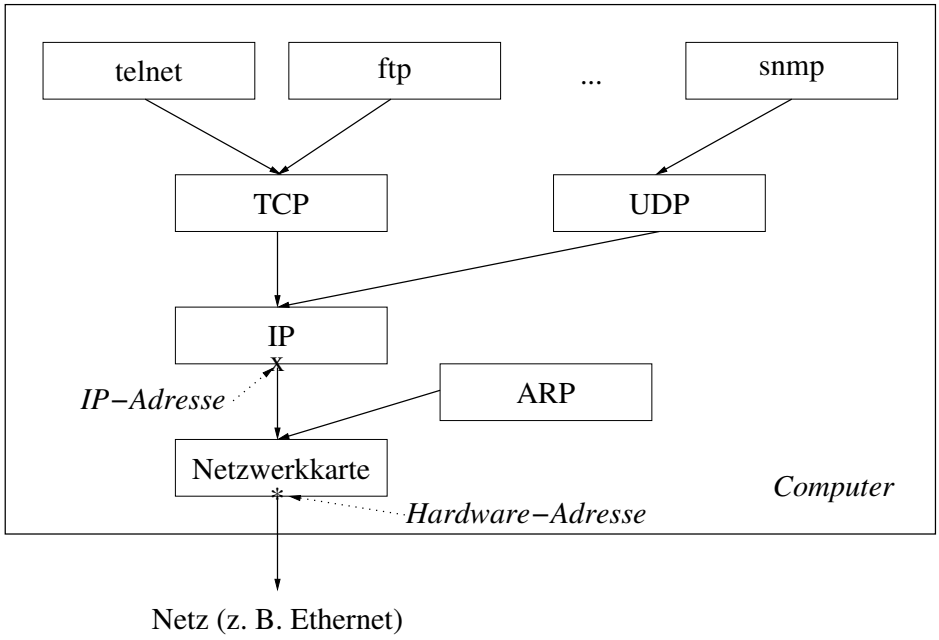


Abbildung 1.14: Ablauf der Kommunikation bei TCP-IP

wendungsprogramme (wie telnet, ftp oder andere), die den Bedarf haben, mit einem anderen Computer Daten auszutauschen. Diese Programme rufen dazu Funktionen aus der *Transmission Control Protocol (TCP)* bzw. der *User Data Protocol (UDP)* Schicht auf. Das Anwendungsprogramm übergibt eine Anwendungsnachricht.

TCP
UDP

Die Funktionen der TCP-Schicht zerlegen diese Nachricht und leiten TCP-Segmente an die darunterliegende *Internet Protocol (IP)* Schicht weiter. In der IP-Schicht muß auch die IP-Adresse des Computers selbst bekannt sein.

IP

Die IP-Schicht gibt ein IP-Paket (versehen mit einer IP-Ziel-Adresse) an die Netzwerkkarte weiter. Mit Hilfe des *Address Resolution Protocol (ARP)* und der auf dem Computer gespeicherten Tabelle wird die Hardware-Adresse des Zielcomputers ermittelt. Damit versehen geht dann beispielsweise ein Ethernet-Frame auf den Weg vom Computer an das Netz.

ARP

Adressierung: Jede Netzwerkkarte hat eine eigene eindeutige Adresse. Diese hängt von der verwendeten Netztechnologie ab und ist bei Ethernet bei-

spielsweise 6 Bytes lang. Für das Internet abstrahiert man von den Hardware Adressen und benutzt ein eigenes Adreßschema, die sogenannten **IP-Adressen**.

IP-Adressen

Eine IP-Adresse ist eine eindeutige 4 Byte lange Binärzahl. Sie besteht aus zwei Teilen. Das *Präfix* identifiziert das Netz, an das der Computer angeschlossen ist. Das *Suffix* identifiziert den Computer innerhalb des Netzes. Da lokale Netze aus unterschiedlich vielen Computern bestehen können, ist die Länge des Suffix nicht fest vorgegeben. IP-Adressen notiert man, indem jedes Byte als Dezimalzahl geschrieben wird. Zwischen die Zahlen wird ein Punkt geschrieben.

Die IP-Adresse eines Computers wird bei der Konfiguration eingegeben und lokal gespeichert.[7] Zur ARP-Schicht gehört eine Tabelle der folgenden Form:

IP-Adresse	**Hardware-Adresse**
10.71.144.1	08-00-28-00-38-A9
10.71.144.2	08-00-39-00-2F-C3
10.71.144.3	...
...	

Der Inhalt dieser Tabelle wird jedoch nicht vom Administrator gepflegt, sondern sie füllt sich automatisch. Möchte der Computer ein Paket an eine IP-Adresse schicken, deren Hardware-Adresse nicht in der Tabelle steht, so wird das Paket zunächst zurückgestellt. Der Computer schickt eine Anfrage an alle angeschlossenen Computer (engl. **broadcast**) und fragt nach der Hardware-Adresse zu dieser IP-Adresse. Einer der angeschlossenen Computer erkennt nun seine IP-Adresse und antwortet mit seiner Hardware-Adresse. Anschließend kann das IP-Paket mit der richtigen Hardware-Adresse versehen, in ein Netzpaket verwandelt und abgeschickt werden.

Aber auch IP-Adressen sind für Benutzer nicht einfach zu merken. Um Namen für Computer vergeben zu können, gibt es das **Domain Name System (DNS)**. Hierin werden symbolische Namen für Computer und deren zugehörige IP-Adressen verwaltet. Beispiele für DNS-Namen sind: `gremlin.fernuni-hagen.de` oder `www.deutsche-bank.de`. DNS-Namen sind hierarchisch aufgebaut. Ausgehend von einer leeren Wurzel (siehe Abbildung 1.15) setzt sich ein Name aus den Knoten eines Pfads durch den Baum zusammen. Dieser Baum gibt *nicht* die Netzstruktur wieder, sondern kann vom Verantwortlichen für einen Knoten beliebig unterhalb des Knotens gestaltet werden.

Domain Name System (DNS)

Früher wurde diese Zuordnung lokal auf jedem Computer gespeichert. Eine zentrale Stelle registrierte alle Veränderungen und verteilte die neue Zuordnungstabelle. Bei der derzeitigen Anzahl von Computern im Internet und der hohen Änderungsgeschwindigkeit ist dieser Mechanismus nicht mehr praktikabel.

Heute wird die Zuordnung von IP-Adresse zu Computername von einem **DNS-Server** erledigt. Dieser arbeitet in einem Verbund mit anderen

DNS-Server

[7] Um diese aufwändige Arbeit zu sparen benutzen viele Administratoren eine Technik mit der IP-Adressen automatisch an Rechner vergeben werden. Sie heisst Dynamic Host Configuration Protocol (DHCP).

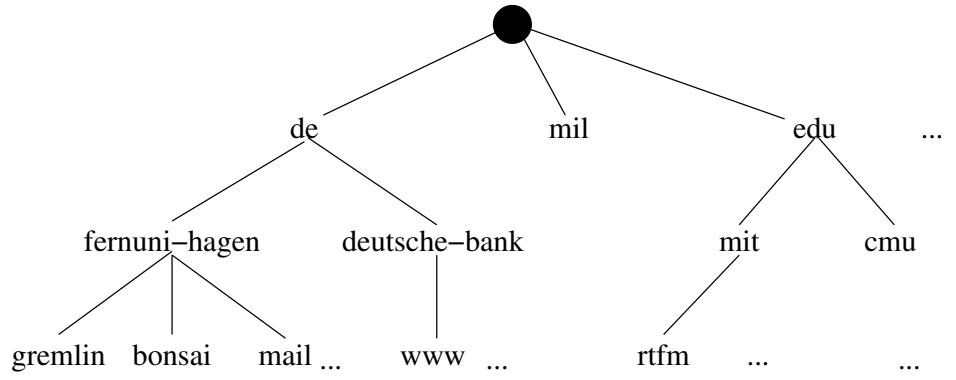


Abbildung 1.15: Beispiel für einen DNS-Namensraum

DNS-Servern zusammen. Falls zu jedem Knoten aus dem Baum aus Abbildung 1.15 ein DNS-Server gehört, so muß dieser nur seine Kinder und seinen Vorgänger kennen. Wenn dann vom Computer `bonsai.fernuni-hagen.de` auf `www.deutsche-bank.de` zugegriffen werden soll, so entsteht eventuell folgende Abfragekette: `bonsai` fragt den DNS-Server von `fernuni-hagen`, der fragt den DNS-Server von `de`, der fragt den DNS-Server von `deutsche-bank` und der kennt die Adresse von `www`. Die IP-Adresse wird dann zurückgeschickt. Damit das Netz nicht ständig von solchen Anfragen belastet wird, merken sich die DNS-Server angefragte Adressen. Wenn also auch der Computer `gremlin.fernuni-hagen.de` die Adresse von `www.deutsche-bank.de` wissen will, so kann der DNS-Server der FernUniversität diese Anfrage aus seinem Zwischenspeicher (engl. **cache**) beantworten.

Dieses Schema birgt leider auch gewisse Risiken. Wenn Sie beispielsweise mit dem Computer `gremlin.fernuni-hagen.de` kommunizieren wollen, kann Ihr DNS-Server eventuell eine falsche IP-Adresse zurückgeben. Sie würden dann Daten an einen anderen Computer schicken, der nur vorgibt `gremlin.fernuni-hagen.de` zu sein. Aber auch IP-Adressen sind kein sicherer Authentifizierungs-Mechanismus. Wie oben schon erwähnt, wird die Tabelle zum ARP durch eine broadcast-Anfrage gefüllt. So kann auch ein anderer Computer in Ihrem lokalen Netz vorgeben, Ihre IP-Adresse zu haben. An Sie geschickte Daten landen dann auf einer anderen Maschine. Beachten Sie außerdem, daß der Administrator oder auch der Benutzer die IP-Adresse seines Computers selbst konfigurieren kann.

Für die verschiedenen Dienste bzw. Anwendungsprogramme ist es nicht nur erforderlich zu wissen, mit welchem Computer man kommunizieren möchte, sondern man muß auch mit schicken können, welchen Dienst man benutzen möchte. Dazu sind in TCP/IP die sogenannten *ports* vorgesehen. Ein Port ist eine Nummer. Empfängt ein Computer eine Nachricht, so kann die TCP/IP-Schicht bereits erkennen, welches Programm diese Nachricht erhalten muß. Der Anwendungsprogrammierer muß sich also nicht mit Datenpaketen befassen, die gar nicht für diese Anwendung gedacht sind. Die Portnummer wird an den Rechnernamen angehängt und durch einen Doppelpunkt von Namen getrennt. Eine Rechneradresse inklusive Portnummer sieht dann beispielsweise so aus:

port

```
gremlin.fernuni-hagen.de:80
```

1.3.4 Die Internet-Dienste

In diesem Abschnitt werden einige typische Hilfsprogramme und Internet-Dienste vorgestellt. Es wird insbesondere auf die sicherheitsrelevanten Aspekte der Dienste eingegangen.

Nslookup: Mit dem Programm *nslookup* können sie einen DNS-Server abfragen. Sie können zu einem Rechnernamen die zugehörige IP-Adresse erfahren, oder sie können zu einer IP-Adresse die zugehörigen Rechnernamen abfragen. Die Abbildung von Rechnernamen auf IP-Adressen ist nicht unbedingt bijektiv. Häufig sind mehrere Namen derselben IP-Adresse zugeordnet. Ein Beispiel für *nslookup*:

```
>nslookup -silent www.fernuni-hagen.de
Server:         141.71.30.1
Address:        141.71.30.1#53

Non-authoritative answer:
www.fernuni-hagen.de    canonical name = cl-www.fernuni-hagen.de.
Name:   cl-www.fernuni-hagen.de
Address: 132.176.114.181
```

In diesem Beispiel kommt die Antwort vom DNS-Server mit der Adresse 141.71.30.1. Der Web-Server der FernUniversität hat zwei Namen (`cl-www` und `www`). Der Vorteil ist, daß man den Web-Server bei Bedarf auf einer anderen Maschine installieren kann und dann nur den DNS-Eintrag ändern muß. Ihre lokal gespeicherten Verweise zeigen dann nach wie vor auf den richtigen Computer.

Unter UNIX ersetzt das Programm *dig* zukünftig *nslookup*. Die Option `-silent` in obigem Kommando unterdrückt eine Erinnerungsmeldung von `nslookup`, daß zukünftig `dig` benutzt werden soll. Neben den Informationen über die IP-Adresse zu einem Namen gibt *dig* auch Informationen über den DNS-Server aus. Beispielsweise kann der DNS-Server eine IP-Adresse von einem anderen DNS-Server übermittelt bekommen haben oder er kann sie aus seinem lokalen Cache gelesen haben. In diesem Fall (Cache) könnte die Adresse evtl. nicht mehr gültig sein, da sie geändert wurde und sich diese Tatsache noch nicht bis zum Cache des DNS-Servers verbreitet hat.

Ping: Nachdem sie nun die IP-Adresse bzw. den Namen eines Computers kennen, können sie mit dem Programm *ping* testen, ob der Computer eingeschaltet ist und IP-Pakete empfangen und zurückschicken kann. Sie können den Computer entweder mit seiner IP-Adresse oder seinem DNS-Namen identifizieren.

```
>ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.207 ms
```

```
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.043 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.041 ms
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.098 ms
64 bytes from localhost (127.0.0.1): icmp_seq=5 ttl=64 time=0.090 ms

--- localhost ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3999ms
rtt min/avg/max/mdev = 0.041/0.095/0.207/0.061 ms
```

In diesem Beispiel wird der lokale Rechner abgefragt. Beachten Sie, daß der Administrator eines Rechners den Rechner auch konfigurieren kann, daß der Rechner niemals auf pings antwortet.

Finger: Mit dem Programm *finger* können Sie herausfinden, wer an einem entfernten Computer angemeldet ist. Weiterhin können sie damit zusätzliche Informationen zu einem Benutzer erlangen. Da sich diese Informationen zur Vorbereitung von Angriffen benutzen lassen, wird der finger-Dienst i. d. R. deaktiviert.

finger port: 79

Traceroute: Mit dem Programm *traceroute* können Sie den Weg verfolgen, den IP-Pakete von Ihrem lokalen Rechner zu einem entfernten Rechner (engl. **remote host**) nehmen. Es ist in erster Linie für Administratoren gedacht. Aber auch „normale“ Benutzer können es starten und dann etwas über die Struktur des Internets erfahren. Beachten Sie jedoch, daß sich diese Struktur ständig ändern kann und die Pakete beim nächsten Aufruf vielleicht einen ganz anderen Weg nehmen könnten. Das folgende Beispiel zeigt den Weg, den IP-Pakete aus der Abteilung Informatik der FH Hannover zum Studentenrechner bonsai der FernUniversität Hagen zurücklegen.

```
> traceroute bonsai.fernuni-hagen.de
traceroute to bonsai.fernuni-hagen.de (132.176.114.21), 30 hops max, 40 byte packets
 1  inner-gw-i.inform.fh-hannover.de (141.71.30.62)  0.288 ms   0.198 ms   0.128 ms
 2  outer-gw-i.inform.fh-hannover.de (141.71.31.30)  0.357 ms   0.366 ms   0.414 ms
 3  rzswitch10.rz.fh-hannover.de (141.71.1.240)  1.019 ms   0.754 ms   0.805 ms
 4  igwserv.rz.fh-hannover.de (141.71.7.2)  0.779 ms   0.628 ms   0.671 ms
 5  141.71.8.2  1.261 ms   1.093 ms   1.222 ms
 6  clustergate-907-rf.rrzn.uni-hannover.de (130.75.9.9)  2.752 ms   2.571 ms   2.703 ms
 7  gwingate-cgc.rrzn.uni-hannover.de (130.75.9.245)  3.675 ms   3.404 ms   3.629 ms
 8  ar-hannover1.g-win.dfn.de (188.1.46.1)  4.031 ms   3.962 ms   3.914 ms
 9  cr-hannover1-ge5-1.g-win.dfn.de (188.1.88.1)  4.452 ms   4.201 ms   4.061 ms
10  cr-essen1-po0-0.g-win.dfn.de (188.1.18.49)  9.775 ms   9.198 ms   9.005 ms
11  ar-essen1-ge0-0-0.g-win.dfn.de (188.1.86.2)  9.659 ms   8.775 ms   9.736 ms
12  C65-GWIN.fernuni-hagen.de (132.176.100.1)  11.467 ms   11.357 ms   11.261 ms
13  bonsai.fernuni-hagen.de (132.176.114.21)  11.465 ms   11.339 ms   11.520 ms
```

Man erkennt an den ersten beiden Zeilen (inner gateway und outer gateway), daß die Abteilung Informatik der FH Hannover eine Firewall mit DMZ betreibt. Das Thema Firewall wird in Kurseinheit 4, Abschnitt 4.3 besprochen.

Einige Netzbetreiber konfigurieren ihre Router so, daß man auch mit traceroute keine Informationen darüber erhält, welchen Weg die Pakete nehmen. Alle an das Internet angeschlossene Firmennetze leiten die für traceroute erforderlichen Pakete nicht ins Internet. Man erhält somit keine Informationen über die interne Netzstruktur einer Firma.

telnet port: 23

Telnet: Das Programm *telnet* ist ein Terminal-Programm mit dem Sie sich zu einem beliebigen anderen Computer im Internet verbinden können. Dabei verhält sich *telnet* so als säßen Sie an einem Terminal, das direkt an den anderen Computer angeschlossen ist. Die Benutzer-Kennung und das Paßwort werden über das Internet übertragen. Auch während der Sitzung werden die Benutzereingaben und die Antworten des Computers *unverschlüsselt* übertragen. Je nachdem welchen Weg die Pakete nehmen, können alle Computer auf diesem Weg die Daten mitprotokollieren. Statt *telnet* wird heute überwiegend *ssh* benutzt. Es wird in Abschnitt 3.4.1 vorgestellt.

Man kann *telnet* aber einsetzen um Netzprotokolle zu testen. Dazu sagt man *telnet* mit welchem Port man verbunden werden will indem man hinter dem Rechnernamen die Portnummer eingibt. Das Kommando `telnet www.fernuni-hagen.de 80` verbindet sie direkt mit dem Web-Server. Sie können dann die unten gezeigten HTTP-Nachrichten eintippen und die Antworten des Servers als Text angezeigt bekommen. Dasselbe macht ihr Web-Browser im Prinzip auch, aus einem Klick wird eine Anfragenachricht und die Antwortnachricht wird im Fenster schön formatiert angezeigt.

ftp port: 21

File Transfer: Mit dem Programm *ftp* können Sie Dateien zwischen einem lokalen Computer und einem entfernten Computer hin und her kopieren. Das Programm erwartet beim Aufbau der Verbindung die Eingabe einer Benutzer-Kennung und eines Paßworts. Diese werden, wie bei *telnet*, *unverschlüsselt* verschickt. Damit Sie nicht auf jedem ftp-Server eine Benutzer-Kennung einrichten müssen, gibt es die anonymen Kennungen `ftp` und `anonymous`. Diese brauchen kein Paßwort (man bittet Sie i. d. R., Ihre eigene email Adresse als Paßwort anzugeben) und erlauben eingeschränkten Zugriff auf den Server.

Übungsaufgabe 1.3 Welche Sicherheitsrisiken können bei *ftp* auftreten?

WWW

World Wide Web: Waren die bisher vorgestellten Anwendungen und Dienste textbasiert und terminalorientiert, so ist das *World Wide Web (WWW)* multimediabasiert und an graphischen Benutzeroberflächen orientiert. Zusätzlich bietet das WWW Möglichkeiten des *Information Retrieval*. Als Benutzer brauchen sie ein Client-Programm, das auch als *Web-Browser* bezeichnet wird, um auf die Inhalte von Web-Servern zugreifen zu können. Die Kommunikation zwischen Web-Client und Web-Server ist durch das *Hypertext Transfer Protocol (HTTP)* definiert. Darin ist festgelegt, welche Form die Anfragen eines Web-Clients haben und wie ein Web-Server darauf antwortet. Dokumente auf Web-Servern sind in der *Hypertext Markup Language (HTML)* geschrieben. HTML ist eine SGML-Anwendung, d. h. es gibt eine *Document Type Definition (DTD)* vom World Wide Web Consortium (W3C) in der die zulässigen Dokument-Bestandteile und ihre Organisation festgelegt sind. Für die Zukunft ist es geplant, nicht nur HTML-Dokumente, sondern allgemeine Dokumente in der *eXtensible Markup Language (XML)* zu erlauben. XML ist eine Weiterentwicklung von SGML, bei der man aus den Problemen mit SGML gelernt und die Anforderungen des WWW berücksichtigt hat. Weitere Informationen zu

HTTP port: 80

HTML

XML

den Dateiformaten finden Sie auch im Kurs *(01873) Daten- und Dokumentformate*.

Web-Server und die Dokumente auf ihnen werden durch *Uniform Resource Locators (URL)* adressiert. Eine URL besteht i. d. R. aus drei Teilen: URL

1. Dem Dienst, der benutzt werden soll,
2. dem Web-Server, der kontaktiert werden soll und
3. dem Dokument, das angesprochen werden soll.

Die folgende Tabelle zeigt einige Beispiele von URLs.

Dienst	Web-Server	Dokument
http	://www.fernuni-hagen.de	/FeU/Fachbereiche/fachbereiche_f.html
http	://www.deutsche-bank.de	
ftp	://ftp.fernuni-hagen.de	/

Die Kommunikation zwischen Client und Server basiert auf dem Frage-Antwort-Prinzip (engl. **request response**). Anfragen bestehen aus einem Anfragekopf und einem Anfragerumpf. Diese sind durch eine Leerzeile getrennt. Die erste Zeile eines Anfragekopfs enthält die Methode, das Objekt, auf das diese Methode angewendet werden soll, und die Versionsnummer des HTTP Protokolls. Die weiteren Kopfzeilen enthalten Informationen wie beispielsweise das Datum, die Codierung der Zeichen usw. Ein Beispiel sieht wie folgt aus:

```
GET /index.html HTTP/1.1
Host: 10.71.144.4

```

In diesem Beispiel ist der Anfragerumpf leer. Hinter der Zeile `Host:` kommt also noch eine Leerzeile.

Die Antwort des Servers besteht aus einem Antwortkopf und einem Antwortrumpf, die wiederum durch eine Leerzeile getrennt sind. Die erste Zeile der Antwort ist eine Statuszeile. Die weiteren Kopfzeilen sind teilweise wie im Anfragekopf. Der Antwortrumpf enthält meistens die angeforderte HTML-Seite. Ein Beispiel:

```
HTTP/1.1 200 OK
Date: Fri, 06 Aug 1999 14:57:16 GMT
Server: mod_perl/1.18 Apache/1.3.4 (Unix)  (SuSE/Linux) ...
Last-Modified: Fri, 06 Aug 1999 14:27:12 GMT
Accept-Ranges: bytes
Content-Length: 4464
Content-Type: text/html

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML3.2//EN" "strict.dtd">
<HTML>
  <HEAD>
         ...
```

```
  </HEAD>
  <BODY>
        ...
  </BODY>
</HTML>
```

Die gesamte Kommunikation findet unverschlüsselt statt. Weiterhin kann weder der Client sicher sein, daß er tatsächlich mit dem Server kommuniziert, mit dem er kommunizieren möchte, noch kann der Server sicher sein, daß der Client derjenige ist, der er vorgibt zu sein (Authentizität).

Electronic Mail: Die Möglichkeit, schnell und einfach elektronische Nachrichten (engl. **email**) auszutauschen, hat sicherlich stark zur Popularität des Internets beigetragen. Die Art und Weise, wie email verschickt wird, ist im *Simple Mail Transfer Protocol (SMTP)* festgehalten. Der Benutzer benötigt ein Client-Programm, in dem er seine email schreibt, verwaltet, empfängt und verschickt. Das Client-Programm kommuniziert dazu mit einem Mail-Server, der email entgegen nimmt und entweder direkt zustellt oder an einen weiteren Mail-Server weiterleitet. Die Übertragung einer email vom Mail-Client zum Mail-Server, bzw. von Mail-Server zu Mail-Server läuft wie folgt ab:

SMTP port: 25

Versand

1. Der Mail-Client öffnet eine Verbindung zum Mail-Server. Konkret bedeutet dies, daß an den Port 25 des Mail-Servers eine Nachricht geschickt wird.

   ```
   S: HELO gremlin.fernuni-hagen.de
   ```

   Das vorangestellte „`S: `“ kennzeichnet Nachrichten vom Mail-Client an den Server. Antworten des Servers werden durch ein „`R: `“ gekennzeichnet. Diese Zeichen werden *nicht* übertragen, sondern sind zum besseren Verständnis der Beispiele von mir eingefügt worden. In der oben genannten Nachricht identifiziert sich der email verschickende Computer gegenüber dem Mail-Server.

2. Als nächstes führt der Mail-Client die eigentliche Übertragungstransaktion aus. Diese besteht aus drei Schritten:

   (a) Zunächst wird ein einleitendes Kommando geschickt.

   ```
   S: MAIL FROM: <wohlfeil@gremlin.fernuni-hagen.de>
   ```

   Der Mail-Server beginnt eine neue Transaktion und initialisiert seinen Zustand und seine internen Puffer.

   (b) Anschließend schickt der Mail-Client ein oder mehrere „Empfänger“-Kommandos:

   ```
   S: RCPT TO: <rolf.klein@fernuni-hagen.de>
   R: 250 OK
   S: RCPT TO: <asdfg@fernuni-hagen.de>
   ```

```
R: 550 no such user here
S: RCPT TO: <swohlfeil@acm.org>
R: 250 OK
```

Der Mail-Server überprüft und speichert die Empfängeradressen. Das Beispiel zeigt, daß ein Mail-Server bestimmte Empfängeradressen, wie `asdfg`, u. U. sofort verwerfen kann. Der Server speichert diese Adressen, um die anschließend vom Client übergebene Nachricht an diese Adressen zu schicken.

(c) Zuletzt wird die eigentliche Nachricht übertragen. Diese Übertragung wird durch das DATA Kommando eingeleitet. Das Ende der Nachricht wird durch eine Zeile, die nur einen Punkt enthält gekennzeichnet.

```
S: DATA
R: 354 start mail input; end with <CRLF>.<CRLF>
S: From stefan.wohlfeil@gremlin.fernuni-hagen.de Fri ...
S: Return-Path: <stefan.wohlfeil@fernuni-hagen.de>
S: Date: Fri, 11 Jun 1999 15:05:26 +0200 (MEST)
S: From: Stefan Wohlfeil <stefan.wohlfeil@gmx.net>
S: To: rolf.klein@fernuni-hagen.de
S: Cc: swohlfeil@acm.org
S: Subject: Mail Uebertragung mit SMTP
S: Content-Type: text/plain; charset="iso-8859-1"
S: Content-Transfer-Encoding: 8bit
S:
S: Hallo Herr Klein,
S: ...
S: MfG
S: Stefan Wohlfeil
S: .
S:
R: 250 message accepted for delivery
```

3. Der zweite Schritt kann nun mehrmals wiederholt werden. Hat der Mail-Client alle emails abgeliefert, so beendet er die Verbindung zum Mail-Server.

```
S: QUIT
R: mail.fernuni-hagen.de closing connection
```

Die oben dargestellten Beispiele zeigen das Prinzip der Mail-Übermittlung, nicht unbedingt den Wortlaut der Antworten eines Mail-Servers. Vom Sicherheitsstandpunkt gibt es einige Anmerkungen.

Sicherheit

Zunächst wird bei emails im Internet zwischen einem Umschlag (engl. **envelope**) und der eigentlichen Nachricht unterschieden. Der Umschlag wird durch die Schritte 2.(a) und 2.(b) erstellt. Für den Mail-Server ist nur der Umschlag für die Zustellung der email von Bedeutung.

Die Nachricht selbst besteht auch aus zwei Teilen. Alle Zeilen vor der ersten Leerzeile sind der Nachrichten-Kopf (engl. **message header**), die Zeilen dahinter der Nachrichten-Rumpf (engl. **message body**). Im Nachrichten-Kopf werden Absender- und Empfängerangaben wiederholt. Diese sollten mit den Angaben auf dem Umschlag übereinstimmen, müssen es aber nicht. Unverlangt zugeschickte Werbe-email (engl. **spam**) haben im Nachrichten-Kopf häufig gefälschte Einträge.

Weiterhin findet die komplette Kommunikation zwischen Mail-Client und Mail-Server *unverschlüsselt* statt. Jedermann auf dem Weg der Nachricht kann mitlesen und dadurch erfahren, mit wem und worüber Sie kommunizieren.

Bei der Weiterleitung einer email trägt jeder der beteiligten Mail-Server in den Nachrichten-Kopf eine Zeile `Received:` ein. Darin wird festgehalten, daß und wann diese email den Server erreicht hat. Damit kann bei Zustellungsproblemen der Weg einer email nachvollzogen werden. In den Nachrichten-Kopf kann aber auch der Absender Informationen eintragen, beispielsweise auch eine Zeile `Received`. Damit kann der Weg der email verschleiert werden.

Empfang

Das gesamte email System ist auch darauf eingerichtet, daß ein Empfänger nicht permanent online ist. In diesem Fall wird eine email auf dem letzten am Weg liegenden Mail-Server gespeichert. Sie können mit ihrem Mail-Client dann direkt diesen Server kontaktieren, um angekommene email auf den eigenen Computer zu laden und neue email vom Computer an den Mail-Server zu übertragen.

POP3
IMAP4

Diese Kommunikation ist im *POP3 (Post Office Protocol)*-Protokoll bzw. auch im neuen *IMAP4*-Protokoll geregelt. Die Idee dieser Protokolle ist, daß der Client eine Verbindung zum Mail-Server aufbaut. Im POP3-Protokoll schickt der Client dann einen Benutzernamen und ein Paßwort an den Mail-Server. Beides wird im Klartext über das Netz übertragen. Anschließend kann der Client die angekommenen emails auf seinen Computer übertragen, auf dem Server dann löschen und neue emails vom Computer zum Server schicken.

Da email sehr häufig mit POP3 abgeholt wird, unterstützen moderne email-Server auch durch SSL verschlüsselte Verbindungen. SSL wird in Abschnitt 3.3.1 genauer behandelt.

1.4 Konkrete Gefahren

In diesem Abschnitt lernen Sie einige der konkreten Bedrohungen kennen, denen ihr Computer heutzutage ausgesetzt ist. Dazu gehören die auch in den Nachrichten immer öfter vorkommenden *Viren*, die sogenannten *trojanischen Pferde* und der Mißbrauch von Paßwörtern. Sie werden in den folgenden Unterabschnitten vorgestellt.

Insgesamt ist dies allerdings nur ein erster Einblick. Vertiefende Informationen zu weiteren Angriffsmöglichkeiten auf Computer werden in Kurs *(01867) Sicherheit im Internet 2* behandelt.

1.4.1 Viren

Prinzip: Viren sind Computerprogramme, die sich selbst kopieren (vervielfältigen) können und sich auf diesem Weg vermehren. Neben dieser Grundfunktion enthalten Viren auch andere Funktionen. Diese anderen Funktionen können unterschiedlichen Zwecken dienen:

- Sie können Schäden aller Art auf ihrem Computer anrichten, indem sie Dateien löschen oder deren Inhalt verändern.
- Viren können versuchen, sich selbst zu tarnen und zu verstecken.
- Die zusätzlichen Funktionen können aber auch „nur“ den Benutzer durch seltsame Ausgaben (Bildschirmmeldungen oder akustische Signale) verunsichern und erschrecken.

Schäden: Ein Virus ist für denjenigen, der ihn auf dem Computer hat, immer eine unangenehme Sache. Man kann eigentlich nie genau wissen, was ein Virus letztlich machen wird. Um das herauszufinden, müssen menschliche Experten das Virus genau analysieren. Aus der Theoretischen Informatik ist bekannt, daß Computerprogramme im allgemeinen nicht einmal herausfinden können, ob ein gegebenes Programm bei einer gegebenen Eingabe terminiert (Unentscheidbarkeit des Halte-Problems). Es besteht also wenig Hoffnung, daß Viren-Scanner jemals genau herausfinden können, welchen Schaden ein Virus möglicherweise anrichten kann. Ein Benutzer hat also keine andere Wahl, als das Virus aus seinem System zu entfernen. Dies kostet auf jeden Fall Zeit und Mühe.

Schadensumfang

Von Viren verursachte Schäden können unterschiedlich groß sein. Im einfachsten Fall löst das Virus ein akustisches Signal bei jedem Tastendruck aus, falls gerade der 18. eines Monats ist. Ein größerer Schaden entsteht, wenn das Virus versucht Programmdateien zu löschen, die Sie ausführen möchten. Da diese Schadenfunktion für den Benutzer offensichtlich ist, kann man das Virus entfernen und die gelöschten Programme wieder installieren. Natürlich sind hierzu die Original-Datenträger oder die hoffentlich regelmäßig angelegten Sicherungskopien (engl. **backup**) erforderlich.

Zu den schlimmsten Schäden gehört es, wenn ein Virus unbemerkt Daten von ihrem Computer liest und diese dann per email oder verpackt in HTTP requests verschickt. Sollte das Virus Ihre Benutzer-Kennung (engl. **account**) oder Ihr Paßwort (engl. **password**) ausspioniert haben, kann sich ein Dritter auf ihrem Computer anmelden und dann beliebigen Schaden anrichten.

Virentypen: Zu den ersten Viren überhaupt gehörten die sogenannten **Bootsektor-Viren (BSV)**. Wenn ein Computer eingeschaltet wird, dann muß als erstes das Betriebssystem geladen werden. Damit man nun auf einem Computer mehrere Betriebssysteme benutzen kann, und damit man ein Betriebssystem einfach aktualisieren (engl. **to update**) kann, ist das Betriebssystem i. d. R. nicht im ROM (Read Only Memory) vorhanden. Statt dessen

Bootsektor-Viren (BSV)

Master Boot Record (MBR)

enthält das ROM ein kleines Programm (Urlader), das die Aufgabe hat das Betriebssystem zu laden. Der Urlader sucht auf fest vorgegebenen Speichermedien nach dem eigentlichen Ladeprogramm (engl. **loader**). Diese fest vorgegebene Stelle nennt man **Master Boot Record (MBR)**. Bei Disketten (engl. **floppy disk**) ist das der erste Sektor auf der Diskette, und bei Festplatten (engl. **hard disk**) ist es Sektor 1 auf Zylinder 0, Kopf 0.

Der Master Boot Record einer Festplatte enthält weiterhin Informationen über die Aufteilung der Festplatte (engl. **partition table**). Das Ladeprogramm aus dem MBR startet nun das Betriebssystem, das auf einer der Partitionen gespeichert ist (siehe Abbildung 1.16). Falls mehrere Betriebssysteme auf der Festplatte installiert sind, bietet das Ladeprogramm eine Auswahl an, und der Benutzer kann entscheiden, welches Betriebssystem gestartet werden soll. Zu Linux gehört das Ladeprogramm GRUB (GRand Unified Bootloader), und es erlaubt das Laden unterschiedlicher Betriebssysteme.

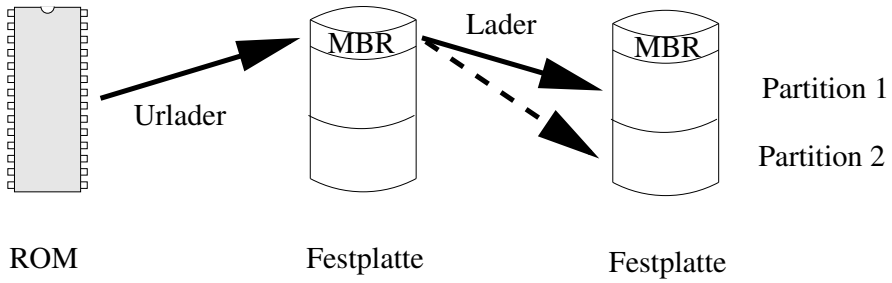


Abbildung 1.16: Boot Vorgang bei einem Computer

Abbildung 1.16 zeigt den prinzipiellen Ablauf beim Starten eines Computers. In einem PC ist der Urlader so implementiert, daß er zunächst auf dem Diskettenlaufwerk nach einem Lader im Bootsektor sucht, anschließend auf der Festplatte und evtl. anschließend auch noch auf dem CD-ROM Laufwerk. Im BIOS des PC kann man diese Reihenfolge evtl. auch ändern.

Wie kann nun ein PC mit einem Bootsektor-Virus infiziert werden? Am Anfang steht i. d. R. eine infizierte Diskette, eine infizierte CD/DVD oder ein infizierter USB stick, die irgendwelche Daten enthalten. Sie legen die Diskette/CD/DVD ein (schließen den USB stick an), kopieren die Daten, arbeiten mit dem Computer, vergessen die Diskette/CD/DVD bzw. den USB stick und schalten am Ende den Computer wieder aus. Bis jetzt ist noch nichts passiert. Beim nächsten Einschalten startet der Urlader nicht den Lader aus dem MBR der Festplatte, sondern den infizierten Lader aus dem Bootsektor der Diskette/CD/DVD bzw. dem USB stick. Das Virus führt dann die folgenden Schritte aus:

1. Es lädt sich in den Hauptspeicher und trägt sich in den Interrupt 13h (disk read/write) ein.

2. Es kopiert den Original MBR der Festplatte an einen anderen freien Platz auf der Platte.

3. Es kopiert sich selbst in den MBR der Festplatte.

4. Es startet den Lader aus dem „Original“-MBR der Festplatte.

Für den Benutzer sieht es so aus, als starte der Computer wie immer. Schließt man nun einen neuen USB stick an und versucht den Inhalt zu lesen oder zu schreiben, so kommt das Virus ins Spiel. Zum Lesen oder Schreiben ruft das Programm das Betriebssystem auf (siehe auch Kurs *(01801) Betriebssysteme und Rechnernetze*) und das Betriebssystem löst den Interrupt aus, womit das Virus aktiviert ist. Das Virus überprüft, ob der USB stick bereits infiziert ist. Ist er es noch nicht, dann kopiert sich das Virus in den Bootsektor des USB sticks. Auf diese Art verbreitet sich das Virus.

Da das Virus gestartet wird, bevor das Betriebssystem geladen ist, können die Schutzfunktionen des Betriebssystems nicht wirksam werden. Ein Virentest-Programm kann das Virus auch nachträglich nicht mehr erkennen. Dazu müßte es den Master-Boot-Record der Festplatte lesen. Dieser Lesezugriff läuft aber letztlich wieder über den Interrupt, in den sich das Virus „eingeklinkt“ hat. Das Virus liefert bei einer Leseanfrage also einfach den kopierten „Original“-MBR zurück. Viren, die sich auf diese oder eine andere Art tarnen, nennt man auch **stealth virus**. Bei der Übertragung auf einen anderen Rechner schützt sich das Beagle Virus beispielsweise dadurch, daß es sich nur verschlüsselt überträgt. Virenscanner auf dem Übertragungsweg können die Nachricht dann nicht entschlüsseln und das Virus erkennen. Erst der Empfänger entschlüsselt die Nachricht und startet möglicherweise das Virus.

stealth virus

Ein PC wird also nur durch das Starten von einem infizierten Boot Medium mit einem Boot-Sektor-Virus infiziert. Um das Virus zu entdecken und zu entfernen, brauchen Sie eine *garantiert* virenfreie Boot-Diskette. Diese enthält die entsprechenden Hilfsprogramme, die einen virenfreien MBR rekonstruieren und auf die Festplatte schreiben können. Von Klaus Knopper gibt es eine Linux Distribution, genannt Knoppix, die man direkt von CD/DVD booten kann. Darauf basiert eine CD/DVD des Heise Zeitschriftenverlags die Antivirenprogramme enthält. Bootet man dieses System und ist der Rechner ans Internet angeschlossen, dann bestehen gute Chancen mit Hilfe der aktuellen Virensignaturen die Schädlinge wieder zu entfernen.

Knoppix

Der sicherste Weg, ein kompromittiertes System wieder in einen sauberen Zustand zu versetzen ist allerdings die komplette Neuinstallation mit einem „sauberen“ Installationsmedium. Nur mit aktuellen Sicherungskopien gelingt die Wiederherstellung ohne Datenverlust.

Der zweite Virustyp sind die sogenannten **Dateiviren**. Ein Dateivirus infiziert einzelne Dateien, vorzugsweise Programmdateien (also `.exe` und `.com` Dateien unter DOS). Startet ein Benutzer ein infiziertes Programm, so lädt sich das Virus in den Hauptspeicher und versucht, dort seßhaft (resident) zu werden. Das heißt, daß auch nach der Beendigung des Wirt-Programms das Virus weiterhin im Speicher bleibt und weiter läuft. Startet der Benutzer weitere Programme, so versucht das Virus, auch diese Programme zu infizieren. Dazu verändert das Virus die Programmdatei, beispielsweise indem der Viruscode an das Ende der Datei angehängt wird. An den Anfang der Programmdatei fügt das Virus dann auch einen Aufruf des Viruscodes am Ende der Datei ein. Siehe dazu auch Abbildung 1.17.

Dateiviren

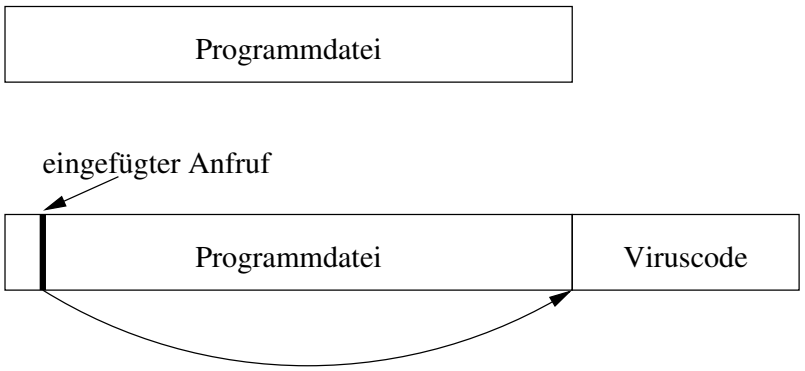


Abbildung 1.17: Infektion einer Programmdatei

Dabei verändert sich die Dateigröße des Programms und man kann daran die Infektion erkennen. Ein cleveres Virus versucht, diese Veränderung zu verbergen. Dazu kann es beispielsweise in der Programmdatei einen Block zusammenhängender Nullen suchen und sich dort hinein kopieren. Beachten Sie, daß Abbildung 1.17 *nicht* maßstabsgetreu ist. Der Virus-Code ist deutlich kleiner als der Programm-Code. Eine passende Folge von Nullen zu finden, ist also gar nicht so unwahrscheinlich. Beim Start des Programms wird dieses in den Hauptspeicher geladen und der Virus-Code gestartet. Neben den oben bereits genannten Schritten überschreibt sich der Virus-Code im geladenen Programm wieder mit Nullen, damit das Programm wie üblich funktioniert. Ein Dateivirus könnte nicht nur andere Programmdateien infizieren, sondern es könnte auch versuchen, sich selbst in den Boot Sektor zu kopieren.

Wenn ein Dateivirus gestartet wird, dann ist das Betriebssystem bereits geladen und das Virus muß die Schutzmechanismen des Betriebssystems überwinden. Die mangelhaften Schutzmechanismen in DOS und damit auch in MS Windows (außer bei Windows NT und seinen Nachfolgern Windows 2000, Windows XP und Windows Vista, die nicht mehr auf DOS basieren) machen es Dateiviren leicht. Unter UNIX oder einer der aktuellen Windows-Varianten kann sich ein Dateivirus nicht so ohne weiteres resident in den Hauptspeicher legen. Und es kann auch nicht einfach eine Programmdatei infizieren, wenn der Benutzer zwar das Ausführungsrecht (engl. **execute permission**) hat, aber kein Schreibrecht (engl. **write permission**). Dies bedeutet nicht, daß es unter UNIX keine Dateiviren geben kann, sondern nur, daß die Verbreitung dort deutlich schwieriger als beispielsweise unter DOS/Windows ist.

Vorhandene Dateiviren werden entfernt, indem man den Computer von einem garantiert virenfreien Medium startet und dann alle infizierten Programme wieder neu installiert. Im Zweifelsfall müssen alle Programme neu installiert werden.

Ein Dateivirus kann man durch den Start eines infizierten Programms auf dem eigenen Computer bekommen. Das infizierte Programm kann über ein USB-Gerät oder auch über das Netz (per email oder auch durch einfaches Surfen im Internet) auf den eigenen Computer kommen. Computer, die als Server für Programmdateien in einem Netz arbeiten sind besonders gefährdet.

Makroviren

Der dritte Virentyp sind die sogenannten **Makroviren**. Sie sind im Prin-

zip auch Dateiviren, denn sie befallen Dateien. Während „klassische“ Dateiviren i. d. R. Programmdateien infizieren, stecken Makroviren in Dokumentdateien, wie beispielsweise Textverarbeitungsdokumenten, Tabellenkalkulationsdokumenten oder Präsentationsdokumenten. Die zugehörigen Programme enthalten Funktionen zum Erstellen von Makros. Makros sind kleine Programme, die dem Anwender die Arbeit erleichtern sollen. Zuerst bestand die Funktion von Makros darin, die vom Benutzer gedrückten Tasten zu speichern und diese Folge später wiederholt und automatisch abzuspielen. Moderne Anwendungsprogramme bieten spezielle Makrosprachen, die auf Programmiersprachen wie C oder BASIC basieren. Die Office-Programme der Firma *Microsoft* bieten beispielsweise die Sprache *VBA (Visual Basic for Applications)* zur Entwicklung von Makros an. Diese Makrosprachen erlauben nicht nur das Aufzeichnen von Tastendrücken, sondern auch komplexe Kontrollstrukturen (Sequenz, Auswahl, Wiederholung und Prozeduraufrufe) und den Zugriff auf Dateien.

Definition: Makro

Makros werden zusammen mit dem Dokumentinhalt in *einer* Datei gespeichert. Öffnet ein Benutzer diese Datei, so werden der Inhalt und der Makrocode geladen.

Inzwischen verwenden alle Office-Programme von *Microsoft* dieselbe Makrosprache. Makroviren können daher nicht nur in Textdokumenten, sondern auch in Tabellenkalkulationsdokumenten oder auch Präsentationen stecken. Im Gegensatz zu Bootsektor- oder „klassischen“ Dateiviren, die i. d. R. in Assembler oder einer anderen low level Programmiersprache geschrieben werden, werden Makroviren in einer einfacher zu lernenden höheren Programmiersprache geschrieben. Jedes Dokument, das Sie heute aus dem Internet laden, per email zugeschickt bekommen oder von einer Diskette/CD-ROM kopieren, könnte ein Makrovirus enthalten. Zum Schutz vor Makroviren bieten sich folgende Möglichkeiten:

Schutz vor Makroviren

1. Installation eines Virenscanners, der auch Makroviren erkennen und entfernen kann. Da immer wieder neue Makroviren auftauchen, reicht es nicht, den Scanner einmal zu installieren. Man muß den Virenscanner auch regelmäßig aktualisieren. Weitere Hinweise hierzu folgen in Abschnitt 3.5.

   Virenscanner einsetzen

2. Von *Microsoft* sind inzwischen auch Versionen ihrer Office-Programme verfügbar, die ausschließlich Dokumente anzeigen können. Sie können die Dokumente damit nicht bearbeiten. Weiterhin können diese Programme auch keine Makros ausführen. Ein Makrovirus kann in diesen Programmen also nicht aktiv werden.

   reader Programm benutzen

3. Wenn das Dokument von ihnen nicht weiter bearbeitet werden soll, dann ist ein Dokumentformat wie das Portable Document Format (PDF) von *Adobe* die bessere Wahl.

4. Seit etwa 2003 können Makros auch digital signiert sein. Durch prüfen der Signatur kann der Benutzer verifizieren, von wem das Makro stammt und daß es nicht verändert wurde. So kann man Makros von vertrauenswürdigen Personen zulassen, während Makros von anderen Autoren

nicht ausgeführt werden. Konfigurieren Sie Ihre Office-Installation entsprechend.

Makroviren sind recht einfach zu schreiben und haben vielfältige Ausbreitungsmöglichkeiten. Sie sind unabhängig vom Betriebssystem und brauchen nur ihre zugehörige „Wirtsanwendung“. Vernünftigen Schutz vor Makroviren bieten derzeit Virenscanner, die a) regelmäßig aktualisiert werden und b) *alle* Dateien auf Viren überprüfen sowie die sichere Konfiguration der Office-Programme, so daß unbekannte Makros nicht ausgeführt werden.

Übungsaufgabe 1.4 Welche Virengefahren können in CDs oder DVDs stecken, die Zeitschriften häufig beigelegt sind?

1.4.2 Würmer

Wirtsprogramme

Während Viren sich mit Hilfe anderer Programme, sog. **Wirtsprogramme**, verbreiten verbreiten sich Würmer eigenständig. Sie müssen also vom Benutzer mindestens einmal explizit gestartet werden. Dann führen sie ihre Schadensfunktion aus und verbreiten sich weiter. Die Schadensfunktion kann nun auch darin bestehen, daß der Wurm dafür sorgt, daß er später automatisch immer wieder gestartet wird. Unter MS Windows kann er sich in den Autostart Ordner kopieren oder die Registry verändern.

Typische Verbreitungswege für Würmer sind email oder HTTP. Dazu enthält der Wurm beispielsweise seinen eigenen SMTP- oder HTTP-Server. Aber auch andere Verbreitungswege sind denkbar. In einem MS Windows-Netz kann ein Wurm auch nach freigegebenen Laufwerken suchen und sich dort unter einem Tarnnamen installieren. Die Tarnnamen sind so gewählt, daß ein Benutzer dazu verleitet wird, einen Doppelklick auf die Datei auszuführen. Obwohl eigentlich ein Programm, gibt sich der Wurm dann auch einmal den Namen eines Bildes. Bei bestimmten Schwachstellen kann sich ein Wurm auch ganz ohne Unterstützung des Benutzers verbreiten. Der Conficker-Wurm ist so ein Beispiel, da er eine Schwachstelle im RPC-Dienst ausnutzte.

1.4.3 Trojanische Pferde

Ein Programm, das neben seiner eigentlichen Funktion auch weitere Funktionen ausführt, nennt man *Trojanisches Pferd*, falls die weiteren Funktionen dem Benutzer *nicht* bekannt sind und er deren Ausführung auch nicht bemerkt. Normalerweise stellt diese weitere Funktion für den Benutzer eine Gefahr dar. Ein trojanisches Pferd kann in jedem Programm stecken, in einem Textverarbeitungssystem, einem Editor, einem Bildschirmschoner oder kleinen Hilfsprogrammen zur Datei- oder Netzwerkverwaltung. Aber auch im Betriebssystem oder der Systemsoftware können unbekannte und nicht dokumentierte Funktionen stecken.

Im Internet werden sehr viele Programme von den unterschiedlichsten Autoren für die unterschiedlichsten Zwecke angeboten. Seien Sie vorsichtig, wenn

sie unbekannte Programme aus dem Netz laden und bei sich installieren wollen. Gerade wenn diese Programme nur in Binärform, also für Menschen unleserlich, verteilt werden, können sie über die Funktionen des Programms nur spekulieren. Anfang 2009 wurde beispielsweise über das BitTorrent-Netz eine trojanisierte Version der Bürosoftware *iWork2009* der Firma *Apple* verbreitet. Das Installationsprogramm hat neben der Bürosoftware auch einen Trojaner installiert.

Ein weiteres beliebtes Gebiet für Trojaner sind Multimedia-CODECs. Ein CODEC ist eine Software, die Multimediadaten wie Sound oder Filme in ein digitales Format wie MP3 für Musik oder MPEG für Filme COdiert oder DECodiert. Neben den beiden genannten Formaten gibt es auch viele andere Formate. Multimediasoftware ist daher modular programmiert, d. h. sie kann einfach durch Installation zusätzlicher CODECs erweitert werden und dann auch neue Formate abspielen. Auf einigen Internetseiten werden nun Multimediadaten angeboten, für die der Benutzer einen zusätzlichen CODEC aus dem Netz laden und installieren soll. Hier besteht die Gefahr, daß man neben einem CODEC auch weitere unerwünschte Funktionen installiert.

Dieses modulare Konzept findet sich auch bei den Web-Browsern wieder. Sie können durch sogenannte Plug-Ins in ihren Funktionen erweitern werden. Ob ein Plug-In aber neben der offensichtlichen Funktion nicht auch die Benutzereingaben beim Internet-Banking irgendwohin weiterleitet kann ein normaler Benutzer kaum und selbst Experten nur schwierig herausfinden.

Da man einem Programm nicht per se ansehen kann, was es letztlich machen wird[8], kann man nicht auf die automatische Entdeckung von trojanischen Pferden hoffen. Man kann als Benutzer nur versuchen, Veränderungen im System zu erkennen. Dazu erstellt man sich eine Datenbank, die zu allen installierten Programmen Informationen wie

- das Datum der letzten Änderung,
- die Größe der Programmdatei und
- spezielle Prüfsummen (Hash-Codes) der Programmdatei

enthält. Diese Daten werden dann regelmäßig geprüft. Stellt man Veränderungen fest[9], so müssen die veränderten Programme überprüft und gegebenenfalls neu installiert werden.

Trojanische Pferde finden sich nicht nur in Programmdateien, sondern unter Umständen auch in Dokumentdateien. Diese enthalten neben dem eigentlichen Dokumentinhalt weitere Informationen, die man als Anwender dort nicht vermutet. Dateien von *Microsoft Office* beispielsweise sind in einem codierten Format abgespeichert, das nur *Microsoft* selbst kennt. Neben dem jeweiligen Dokumentinhalt enthalten sie auch Informationen wie die Programmversion

[8] Sie erinnern sich: Das Halteproblem ist *nicht* entscheidbar!

[9] Ein Angreifer kann zwar das Datum der letzten Änderung oder die Dateigröße manipulieren, bei guten Prüfsummen kann der Angreifer die Veränderungen aber nicht mehr so einfach tarnen.

mit der das Dokument erstellt wurde, den Pfadnamen unter dem das Dokument auf der Festplatte gespeichert ist, Namen der Ersteller und Bearbeiter usw. Als Benutzer kann man nicht ausschließen, daß nicht auch Informationen über andere installierte Programme oder sonstige vertrauliche Daten in einem unverfänglichen Textdokument stecken. Verschickt man solche Dateien per email, so gibt man u. U. auch Informationen preis, die man eigentlich nicht preisgeben möchte.

Ab der Version *Microsoft Office 2007* gibt es auch ein Werkzeug *Document Inspector*. Es ist dazu gedacht, vertrauliche und versteckte Informationen aus einem Dokument zu entfernen. Zu diesen Informationen gehören (1) persönliche Daten (z. B. Autor) und allgemeine Dokumenteigenschaften, (2) Kommentare und Informationen zur Versionsgeschichte, (3) Kopf- und Fußzeilen oder sog. watermarks, (4) versteckter Text, versteckte Zeilen/Spalten in Tabellen, (5) unsichtbare Inhalte oder (6) Präsentationsnotizen.

Fazit: Viren, Würmer und trojanische Pferde sind Beispiele für Schadsoftware (engl. **malware**). Aktuelle Schadsoftware kombiniert die Techniken der Viren, Würmer und Trojaner und ergänzt diese Techniken um automatische Aktualisierungsfunktionen. Dadurch kann die Schadsoftware weitere Komponenten nachladen und dadurch auch mutieren. Die Häufigkeit der unterschiedlichen Schadprogrammtypen zu zählen ist daher nicht mehr sinnvoll. Man kann jedoch davon ausgehen, daß die Zahl der Schadprogramme kontinuierlich steigen wird.

1.4.4 Paßwortmißbrauch

Paßwörter (auch Kennwörter genannt) werden zur Authentifikation von Benutzern eingesetzt. Bei der Authentifikation geht es um die Prüfung, ob der Benutzer tatsächlich derjenige ist, der er vorgibt zu sein. Im Prinzip gibt es für diese Prüfung mehrere Möglichkeiten:

Biometrie
1. Überprüfen eines unverwechselbaren und schwer zu fälschenden biometrischen Merkmals, wie beispielsweise eines Fingerabdrucks. Dieses Thema wird in Kurs *(01867) Sicherheit im Internet 2* vertieft.

Besitz
2. Kontrolle eines schwer zu fälschenden Gegenstandes, wie beispielsweise eines Personalausweises.

Wissen
3. Überprüfung, ob die Person eine bestimmte Information, wie beispielsweise eine Geheimnummer, kennt.

Bei der Benutzer-Authentifikation an einem Geldautomaten werden die Merkmale *Besitz* und *Wissen* geprüft. Der Benutzer muß eine gültige ec-Karte haben und eine Geheimzahl (genannt PIN) kennen. Nur wenn beide Merkmale erfolgreich geprüft wurden, ist die Authentifikation gelungen.

Bei der Authentifikation gegenüber einem Computer greift man heute i. d. R. nur auf die Kontrolle des Merkmals *Wissen* zurück. Als Benutzer kann

Sicherheitsproblem
bzw. muß man ein Paßwort auswählen. Das Problem mit Paßwörtern ist, daß

jeder der das Paßwort von Benutzer *xy* kennt, sich selbst als Benutzer *xy* ausgeben kann. Von einem fremden Paßwort kann man auf verschiedenen Wegen erfahren:

Raten: Man kann beispielsweise prüfen, ob der Benutzer seinen Benutzernamen auch als Paßwort eingetragen hat. Vielleicht hat ein Benutzer auch seine Telefonnummer, den Namen des Partners, der Kinder, der Eltern, sein Autokennzeichen oder eine andere öffentlich bekannte Information über sich selbst als Paßwort verwendet.

Ausprobieren: Man kann Computer auch so programmieren, daß sie nacheinander alle Wörter eines Wörterbuchs, systematisch ausprobieren. Noch allgemeiner kann man hergehen und *alle* Zeichenketten über dem Alphabet der zulässigen Paßwortzeichen generieren und als Paßwort ausprobieren. Für ein Paßwort der Länge n, bei dem jedes Zeichen aus einem Vorrat von x Zeichen stammen darf, gibt es jedoch x^n Möglichkeiten.

Ausspähen/Abhören: Bei der Eingabe des Paßworts kann man beobachtet werden. Insbesondere bei Paßwörtern, die man selbst nicht ändern kann oder darf, ist es besonders wichtig, bei der Eingabe nicht beobachtet zu werden. Falls ein unbeobachtet eingegebenes Paßwort erst im Klartext über ein Netz an einen entfernten Computer zur Prüfung geschickt wird, kann das Paßwort auf diesem Weg ausgespäht und kopiert werden.

In diese Kategorie fallen auch alle Versuche von Hackern, mittels gefälschter Nachrichten jemanden zur Preisgabe eines Paßwortes zu verleiten. Im Internet wird das häufig auch *phishing* genannt. Die Nachrichten können Telefonanrufe oder aktuell auch emails sein. Ein Angreifer gibt darin vor, ein Administrator von einer dem Benutzer bekannten Institution zu sein. Der Benutzer soll am Telefon sein Paßwort dann direkt nennen.

phishing

Alternativ wird in einer email behauptet, daß technische Probleme bei einem Diensteanbieter im Internet aufgetreten seien und der Benutzer seine Daten zur Aktualisierung bzw. Überprüfung erneut eingeben müsse. Dazu steht dann häufig eine lange URL in der Nachricht, die der Benutzer anklicken soll. Der Anbieter könnte dabei die eigene Bank, *eBay*, *amazon* usw. sein. Jeder Anbieter, bei dem man eine Benutzerkennung mit Paßwort besitzt kann betroffen sein. Klickt man nun DIE URL an dann erscheint eine Seite die so aussieht als wäre sie vom Anbieter. In Wirklichkeit stammt sie vom Angreifer und dient nur dem Zweck, Geheimnisse des Benutzers (Paßwörter, Kreditkartennummern, etc.) zu erlangen.

Man sollte sich also immer *wirklich* sicher sein, mit welchem Rechner man verbunden ist, wenn man im Internet surft. Hinweise wie man das macht werden in Abschnitt 3.3 vorgestellt.

Beim Thema „Ausprobieren“ ist zu beachten, daß ein Hacker sich heute gar nicht mehr die Arbeit machen muß, ein eigenes Programm zum Paßwörtersuchen oder -ausprobieren zu schreiben. Solche Programme kann man direkt aus dem Internet laden. Im folgenden soll auf die prinzipielle Arbeitsweise solcher Programme etwas genauer eingegangen werden.

Dazu zunächst einige Hintergrundinformationen, wie Paßwörter unter UNIX behandelt werden. Paßwörter werden *nicht* im Klartext gespeichert. Statt dessen wird in der Datei `/etc/passwd` für jeden Benutzer eine Zeile mit bestimmten Einträgen angelegt. Zu diesen Einträgen gehören unter anderem:

- der Benutzername (engl. **user ID**),
- der Vor- und Nachname des Benutzers,
- das Heimatverzeichnis (engl. **home directory**) des Benutzers,
- die Gruppe, der der Benutzer zugeordnet ist,
- ein initialer Wert, genannt „Salz“ (engl. **salt value**) und
- das verschlüsselte Paßwort.

Wie das Paßwort verschlüsselt wird, zeigt Abbildung 1.18. Der salt value wird vom System generiert und hängt von der Uhrzeit ab, zu der der Benutzer sein Paßwort setzt. Dieser Wert wird an das vom Benutzer eingegebene geheime Paßwort angehängt (engl. **to append**). Der so entstandene Wert wird als Schlüssel benutzt. Mit diesem Schlüssel wird eine Folge von 12 Null-Zeichen mit einer Abwandlung des DES Algorithmus (siehe auch Abschnitt 2.3.3) verschlüsselt. Das Ergebnis wird noch einmal mit dem Schlüssel verschlüsselt. Nach 25 Verschlüsselungsrunden wird das Endergebnis in die Datei `/etc/passwd` eingetragen.

Der Sinn des salt value liegt darin, daß nun zwei Benutzer dasselbe Paßwort wählen können *ohne* daß auch die verschlüsselten Einträge in `/etc/passwd` identisch sind. Dazu hätten die Benutzer das Paßwort nämlich auch zur selben Zeit eingeben müssen.

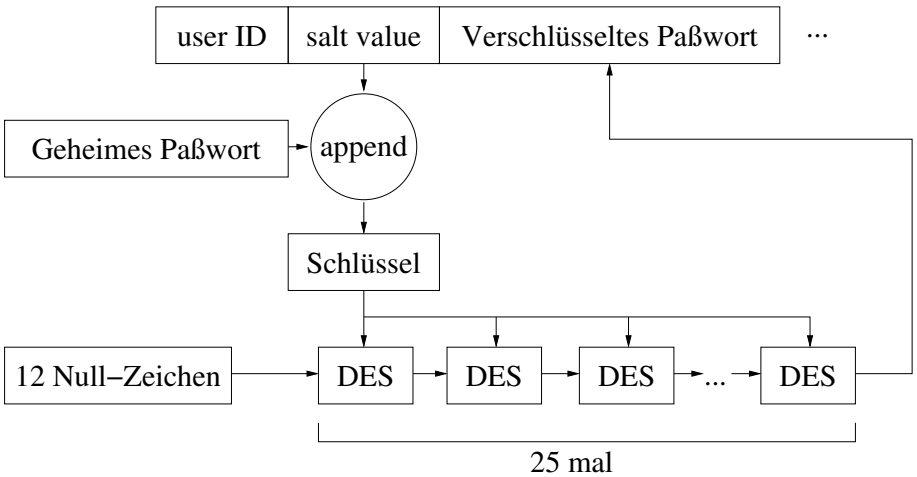


Abbildung 1.18: Ablauf beim Speichern eines Paßworts

crack
john

Die im Internet verfügbaren Programme *crack* und *john* probieren nach diesem Verfahren (25-maliges Verschlüsseln) einfach eine große Menge von möglichen geheimen Paßworten (mit jedem möglichen salt value) aus. *john* kann auch MS Windows Paßwörter suchen. Dazu gibt es große Wörterbücher im

Internet, die gerne gewählte Paßwörter[10], Wörter aus unterschiedlichen Sprachen, Filmtitel, Namen usw. enthalten. Weiterhin prüft *crack* auch bestimmte Modifikationen der Wörter aus dem Wörterbuch. Beliebte Ersetzungen wie „i“ durch „1“ oder „o“ durch „0“ werden genauso geprüft wie unterschiedliche Groß-/Kleinschreibungen. Die Erfolgsquote solcher Angriffe kann bis zu 25% betragen!

Auch wenn heutige Betriebssysteme selbst das verschlüsselte Paßwort dem Zugriff der Benutzer entziehen[11] so sind Paßwort-crack-Programme nach wie vor eine ernste Bedrohung der Sicherheit. Bei der Auswahl eines Paßwortes sollten Sie sich daher an die folgenden Regeln halten:

- Das Paßwort steht nicht in einem erkennbaren Zusammenhang mit Ihnen. Also keine Telefonnummern; keine Namen von Familienmitgliedern, Freunden, Kollegen; keine Autokennzeichen; keine Hobbys; usw.

- Das Paßwort ist ausreichend lang (mindestens 6 Zeichen, besser sogar 8), damit systematisches Ausprobieren zu lange dauert.

- Das Paßwort stammt nicht aus einem Wörterbuch. Fügen Sie beispielsweise Sonderzeichen (Ziffern, Satzzeichen, o. ä.) mitten in ein Wort ein, so haben Sie ein längeres, nicht im Wörterbuch stehendes und trotzdem nicht zu schwer zu merkendes Paßwort gefunden.

- Vom Hersteller voreingestellte Paßwörter werden unbedingt geändert.

- Paßwörter sind nicht zu lange gültig. Irgendwann hat ein Angreifer beim Ausprobieren aller Möglichkeiten vielleicht doch Erfolg. Sie sollten Ihre Paßwörter daher regelmäßig ändern. Auch wenn Sie das Gefühl haben, daß jemand Ihr Paßwort geknackt hat, sollten Sie unbedingt sofort das Paßwort ändern.

- Paßwörter sollten nicht notiert werden. Falls es doch erforderlich sein sollte, dann nur in einem verschlossenen versiegelten Umschlag, der an einem sicheren Ort deponiert wird.

Leider gilt für Paßwörter aber auch folgende Regel: Regel

> Gut zu merkende Paßwörter lassen sich i. d. R. einfach „knacken“. Schwer zu „knackende“ Paßwörter können sich Benutzer nicht einfach merken.

Eine einfache Möglichkeit diese Regel zu brechen besteht darin, sich einen Satz an Stelle des Paßwortes zu merken. Aus diesem Satz kann man dann das Tipp!

[10] Viele Systemadministratoren sind auch Science Fiction Fans und kennen die einschlägige Literatur. Paßwörter aus solchen Büchern sind also selbst als Paßwort eines Administrators gar nicht so selten.

[11] Unter UNIX gibt es die sogenannten *shadow passwords*. Die verschlüsselten Paßwörter sind dann nicht mehr in der Datei `/etc/passwd` abgelegt sondern in einer Datei die nicht für jedermann lesbar ist.

Paßwort ableiten, beispielsweise indem das Paßwort aus den Anfangsbuchstaben der Wörter und den Satzzeichen besteht. Aus dem Satz: „Das einfachste Paßwort auf der Welt.“ würde dann die Gedankenstütze für das Paßwort „DePadW.“ entstehen.

Ergänzend können Sie sich dann noch ein oder zwei Sonderzeichen überlegen und in das oben entstandene Paßwort einbauen. Dann haben Sie ein sicheres und hoffentlich schwer zu knackendes Paßwort gefunden, das Sie sich vermutlich auch gut merken können.

1.5 Zusammenfassung

Nach dem Durcharbeiten dieser Kurseinheit sollten Sie folgendes gelernt haben:

- Warum man sich mit dem Thema Computer und Sicherheit befassen sollte.
- Welche Bedeutung das Wort Sicherheit im Zusammenhang mit Computern eigentlich hat.
- Welche Systematik in den Bedrohungen steckt und welche Eigenschaften man von sicheren Systemen erwartet.
- Wie Rechnernetze im Prinzip funktionieren und welche Protokolle und Dienste im Internet benutzt, bzw. angeboten werden.
- Welche konkreten Gefahren Ihnen und Ihrem Computer heute schon drohen.

Lösungen der Übungsaufgaben

Übungsaufgabe 1.1

1. Der Ausfall der Stromversorgung ist eine technische Bedrohung. Stromausfall aufgrund der Witterungsbedingungen ist unabsichtlich und er gehört zu den aktiven Bedrohungen. Er ist deshalb aktiv, da etwas passiert und man das auch sehr schnell merkt.

2. Datenverlust, weil jemand eine Diskette per Post an eine falsche Adresse verschickt ist eine nicht-technische Gefahr. Sie kann beabsichtigt oder unbeabsichtigt sein und ist auf jeden Fall auch eine aktive Bedrohung, da sie aktives Handeln voraussetzt.

3. Wenn unerlaubt Geld von einem Girokonto gebucht wird, so ist das eine nicht-technische Bedrohung, da es sich im Prinzip um einen normalen Vorgang handelt. Wenn die Überweisung auf ein bestimmtes Konto erfolgt, so handelt es sich um eine absichtliche und aktive Bedrohung.

Übungsaufgabe 1.2 Die zu schützenden Eigenschaften in Systemen sind:

Vertraulichkeit: Dieses Schutzziel dient der Sicherung der Privatsphäre. Es bedeutet, daß der Inhalt bei einer Kommunikation nur dem Absender und dem beabsichtigtem Empfänger bekannt sein darf. Man wird hier evtl. sogar fordern, daß auch die Tatsache, daß eine Kommunikation statt gefunden hat nur den Beteiligten bekannt werden darf.

Integrität: Dieses Schutzziel dient der Sicherung des Vertrauens in den Inhalt der Kommunikation. Absender und Empfänger sollen sich sicher sein, daß die Nachricht auf dem Transportweg nicht verändert wurde.

Authentizität: Dieses Schutzziel dient der Sicherung des Vertrauens in die Identität der an einer Kommunikation Beteiligten. Konkret soll sich der Kommunikationspartner sicher sein, daß der andere Partner tatsächlich derjenige ist, der er vorgibt zu sein.

Verfügbarkeit: Dieses Schutzziel dient der Sicherung des Vertrauens in die „Technik“. Konkret erwartet man, daß die Dienste zur Verfügung stehen, wenn man sie benutzen möchte.

Übungsaufgabe 1.3 Bei *ftp* treten mindestens die folgenden Sicherheitsrisiken auf:

- Die Übertragung des Paßwortes kann abgefangen werden und ein Dritter kann dann mit diesem Paßwort evtl. vertrauliche Daten vom ftp-Server laden.

- Der Dritte kann auch Daten, die durch Urheberrecht geschützt sind auf den ftp-Server kopieren. Dadurch macht sich der Betreiber des ftp-Servers evtl. strafbar (Raubkopien).

- Der account und das Paßwort können nicht nur für ftp gelten, sondern auch ganz normale Benutzerkennungen sein. Dann kann sich ein Angreifer per *telnet* auf dem ftp-Server anmelden und dort beliebige Kommandos ausführen.

Übungsaufgabe 1.4 Auf einer fremden CD können alle drei vorgestellten Virentypen enthalten sein. Zunächst könnte die CD einen Boot Sektor Virus enthalten, der bei normaler Benutzung nicht auffällt. Normalerweise bootet man seinen Rechner ja nicht von CD.

Weiterhin können die auf der CD vorhandenen Programme Viren enthalten. Mit einem aktuellen Viren-Scanner sollte man also die CD zuerst komplett testen, bevor man eines der Programme darauf startet oder installiert.

Viele der CDs enthalten auch einzelne Artikel aus der Zeitschrift. Diese liegen als Portable Document Format (PDF) oder als *Office*-Dokumente, z. B. für *Word* vor. Deshalb können in diesen Dateien auch Makroviren stecken.

Literatur

[And08] Ross J. Anderson. *Security Engineering*. Wiley und Sons, 2. Auflage, 2008.

[Ano98] Anonymous. *Maximum Security*. SAMS, Indianapolis, Indiana, 2. Auflage, 1998.

[Ano03] Anonymous. *Hacker's Guide*. Markt+Technik Verlag, München, Deutschland, 2003. Übersetzung von Maximum Security, 4th ed.

[bac99] Stephan Roßbach. *Der Apache Webserver*. Addison-Wesley, Bonn, Deutschland, 1999.

[Bau97] Friedrich L. Bauer. *Entzifferte Geheimnisse. Methoden und Maximen der Kryptologie.* Springer-Verlag, Heidelberg, Deutschland, 1997.

[BPH02] L. Bassham, W. Polk und R. Housley. Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation Lists (CRL) Profile. `ftp://ftp.rfc-Herausgeber.org/in-notes/rfc3280.txt`, April 2002.

[Bra99] John R. T. Brazier. Possible NSA Decryption Capabilities. *DuD Datenschutz und Datensicherheit*, 23(10):576–581, Oktober 1999.

[BSI08a] BSI. IT-Grundschutz-Vorgehensweise. BSI-Standard 100-2, Version 2.0, Mai 2008.

[BSI08b] BSI. Managementsysteme für Informationssicherheit (ISMS). BSI-Standard 100-1, Version 1.5, Mai 2008.

[BSI08c] BSI. Risikoanalyse auf der Basis von IT-Grundschutz. BSI-Standard 100-3, Version 2.5, Mai 2008.

[CZ95] D. Brent Chapman und Elizabeth D. Zwicky. *Building Internet Firewalls*. O'Reilly & Associates Inc., Sebastopol, CA, 1995.

[Dem00] W. Edwards Deming. *Out of the Crisis*. MIT Press, 2. Auflage, Oktober 2000.

[DR99] Joan Daemen und Vincent Rijmen. AES Proposal: Rijndael. `http://www.esat.kuleuven.ac.be/~rijmen/rijndael/rijndaeldocV2.zip`, 1999.

[DR02] Joan Daemen und Vincent Rijmen. *The Design of Rijndael*. Springer Verlag, Berlin Heidelberg, 2002.

[Eck07] Claudia Eckert. *IT-Sicherheit*. Oldenbourg Wissenschaftsverlag GmbH, München, 5. Auflage, 2007.

[Ele98] Electronic Frontier Foundation. *Cracking DES: Secrets of Encryption Research, Wiretap Politics & Chip Design*. O'Reilly & Associates Inc., Sebastopol, CA, 1998.

[Ert03] Wolfgang Ertel. *Angewandte Kryptographie*. Fachbuchverlag Leipzig im Carl Hanser Verlag, München, 2003.

[Ges04] Alexander Geschonneck. *Computer Forensik*. dpunkt.verlag, Heidelberg, 2004.

[Gon99] Marcus Goncalves. *Firewalls: A Complete Guide*. McGraw-Hill, Oktober 1999.

[GORP04] Helmar Gerloni, Barbara Oberhaitzinger, Helmut Reiser und Jürgen Plate. *Praxisbuch Sicherheit für Linux-Server und -Netze*. Hanser Verlag, München, 2004.

[HPFS02] R. Housley, W. Polk, W. Ford und D. Solo. Internet X.509 Public Key Infrastructure. `ftp://ftp.rfc-Herausgeber.org/in-notes/rfc3280.txt`, April 2002.

[MR99] Günter Müller und Kai Rannenberg, Herausgeber. *Multilateral Security in Communications*. Addison Wesley Longman GmbH, München, 1999.

[Sch96] Bruce Schneier. *Angewandte Kryptographie. Protokolle, Algorithmen und Sourcecode in C.* Addison-Wesley, Bonn, Deutschland, 1996.

[Sch00] Bruce Schneier. *Secrets & Lies. IT-Sicherheit in einer vernetzten Welt*. dpunkt.Verlag / Wiley-VCH, Heidelberg, Weinheim, Deutschland, 2000.

[SGG08] Abraham Silberschatz, Peter Galvin und Greg Gagne. *Applied Operating System Concepts*. John Wiley, Inc., New York, NY, USA, 8. Auflage, 2008.

[SKW+99] Bruce Schneier, John Kelsey, David Wagner, Chris Hall, Niels Ferguson und Doug Whiting. *Twofish Encryption Algorithm : A 128-Bit Block Cipher*. John Wiley and Sons, 1999.

[Spe06] Ralf Spenneberg. *Linux Firewalls mit iptables & Co*. Addison-Wesley, München, Deutschland, 2006.

[SSA+08] Alexander Sotirov, Marc Stevens, Jacob Appelbaum, Arjen Lenstra, David Molnar, Dag Arne Osvik und Benne de Weger. MD5 considered harmful today. `http://www.win.tue.nl/hashclash/rogue-ca/`, Dezember 2008.

[Sta98] William Stallings. *Cryptography and Network Security*. Prentice Hall, Upper Saddle River, New Jersey, 2. Auflage, 1998.

[Sta00] William Stallings. *Network Security Essentials*. Prentice Hall, Uppser Saddle River, New Jersey, 2000.

[Tan02] Andrew S. Tanenbaum. *Moderne Betriebssysteme*. Pearson Studium, 2002.

[Vie09] John Viega. *the myths of security*. O'Reilly, Sebastopol, CA, 2009.

[Wol07] Sebastian Wolfgarten. *Apache Webserver 2. Installation, Konfiguration, Programmierung*. Addison-Wesley, 2007.

[WWS02] Tobias Weltner, Kai Wilke und Björn Schneider. *Windows-Sicherheit, Das Praxisbuch*. Microsoft Press, Konrad-Zuse-Str. 1, D-85716 Unterschleißheim, 2002.

[WY05] Xiaoyun Wang und Hongbo Yu. How to Break MD5 und Other Hash Functions. In Ronald Cramer, Herausgeber, *Advances in Cryptology - EUROCRYPT 2005*, number 3494 in Lecture Notes in Computer Science, Seiten 19–35, Berlin, 2005. Springer-Verlag.