SICHERHEIT

IM

INTERNET

VON

THOMAS SCHMITZ

&

MICHAEL HURLER

SEMINAR

AUSGEWÄHLTE KAPITEL DER

WIRTSCHAFSINFORMATIK

(WINTERSEMESTER 1998/99)

TECHNISCHE UNIVERSITÄT DARMSTADT

LEHRSTUHL FÜR WIRTSCHAFTSINFORMATIK I

PROFESSOR DR. H. J. PETZOLD

Inhaltsverzeichnis

Literaturverzeichnis

- Anderson, Ross
  Why Cryptosystems Fail
  in: Proceedings of the First ACM Conference on Computer and Communication Security
  November 1993
  S. 215-227
  URL: http://www.d.shuttle.de/isil/crypt/txt/wcf.html , 29. Oktober 1998
- Anderson, Ross; Kuhn, Markus
  Tamper Resistance – a Cautionary Note
  in: The Second USENIX Workshop on Electronic Commerce Proceedings
  Oakland, California, November 18-21, 1996
  S. 1-11
  ISBN 1-880447-83-9
  URL: http://www.d.shuttle.de/isil/crypt/txt/andkuhn.html , 29. Oktober 1998
- Barz, Hans Wilhelm
  Kommunikation und Computernetze: Konzepte, Protokolle und Standards
  1. Auflage, 1991
  Hanser Verlag, München, Wien
  ISBN 3-446-16241-0
- CERT Advisory CA-95:01
  IP Spoofing Attacks and Hijacked Terminal Connections
  1997
  URL: ftp://ftp.cert.dfn.de/pub/csir/cert/cert_advisories/CA-95:01.IP.spoofing , 29. Oktober 1998
- CERT Advisory CA-96.20
  Sendmail Vulnerabilities
  1998
  URL: ftp://ftp.cert.dfn.de/pub/csir/cert/cert_advisories/CA-96.20.sendmail_vul ; 14. Dezember 1998
- CERT Advisory CA-96.21
  TCP SYN Flooding and IP Spoofing Attacks
  1998
  URL: ftp://ftp.cert.dfn.de/pub/csir/cert/cert_advisories/CA-96_21.tcp ; 27. November 1998
- CERT Advisory CA-96.24
  Sendmail Daemon Mode Vulnerability
  1997
  URL: ftp://ftp.cert.dfn.de/pub/csir/cert/cert_advisories/CA-96.24.sendmail.daemon.mode ; 14. Dezember 1998
- CERT Advisory CA-97.05
  MIME Conversion Buffer Overflow in Sendmail Versions 8.8.3 and 8.8.4
  1997
  URL: ftp://ftp.cert.dfn.de/pub/csir/cert/cert_advisories/CA-97.05.sendmail ; 14. Dezember 1998
- Decker, Bart De
  Unix Security & Kerberos
  in: Preneel, Bart; Govaerts, René; Vandewalle, Joos (Ed.)
  Computer Security and Industrial Cryptography

1. Auflage, 1993
Springer-Verlag Berlin, Heidelberg, New York
ISBN 3-540-57341-0

- Ellermann, Uwe
  Netzwerksicherung durch Firewalls
  DFN-CERT
  1995
  URL: http://www.cert.dfn.de/team/fire/fire.html , 1. November 1998
- Stephan Fischer, Achim Steinacker, Reinhard Bertram, Ralf Steinmetz
  Open Security – Von den Grundlagen bis zur Anwendung
  1. Auflage, 1998
  Springer-Verlag Berlin, Heidelberg, New York
  ISBN 3-540-64654-X
- Flanagan, David
  Java in a Nutshell
  2nd Edition, 1997
  O'Reilly & Associates, Inc, Cambridge, Köln, Paris
  ISBN 1-56592-262-X
- Hager, Nicky
  Secret Power - New Zealand's role in the international spy network
  1. Auflage, 1996
  Craig Potton Publishing, Nelson, New Zealand
  ISBN 0 908802 35 8
- Hendry, Mike
  Practical Computer Network Security
  1. Auflage, 1995
  Artech House Inc., Boston, London
  ISBN 0-89006-801-1
- Hickman, Kipp E. B.
  The SSL Protocol
  Netscape Corporation
  9. Februar 1995
  URL: http://sitesearch.netscape.com/eng/security/SSL_2.html ; 13. Dezember 1998
- Hosenfeld, Friedhelm; Brauer, Kai
  Kommunikation ohne Grenze
  in: c't – Magazin für Computer Technik
  Ausgabe 12 / 1995
  Verlag Heinz Heise GmbH & Co KG
  ISSN 0724-8679
  S. 330-338
- Kossel, Axel
  Innere Sicherheit – Sichere Intranet-Lösungen
  in: c't – Magazin für Computer Technik
  Ausgabe 10 / 1996
  Verlag Heinz Heise GmbH & Co KG
  ISSN 0724-8679
  S. 332-335

- Kubaitis, Ed (Ed.)
  WWW Browser Security & Privacy Flaws
  1998
  URL: http://www.ews.uiuc.edu/~ejk/browser-security.html ; 13. Dezember 1998
- Mraz, Viktor; Weidner, Klaus
  Falsch Verbunden – Gefahr durch DNS-Spoofing
  in: c't – Magazin für Computer Technik
  Ausgabe 10 / 1997
  Verlag Heinz Heise GmbH & Co KG
  ISSN 0724-8679
  S. 286 ff.
- Network Associates, Inc.
  PGP for Personal Privacy Version 5.5, User's Guide
  1998
  Zu beziehen über: http://www.pgpi.com
- Neuman, B. Clifford; Steiner, Jennifer G.
  Authentication of Unknown Entities on an Insecure Network of Untrusted Workstations
  1988
  URL: ftp://ftp.pca.dfn.de/pub/docs/kerberos/kerberos-abstract.ps.gz , 28. Oktober 1998
- Oppliger, Rolf
  Internet and intranet security
  1. Auflage 1997
  Artech House Inc., Boston, London
  ISBN 0-89006-829-1
- Raepple, Martin
  Sicherheitskonzepte für das Internet: Grundlagen, Technologien und Lösungskonzepte für die kommerzielle Nutzung
  1. Auflage, 1998
  dpunkt-Verlag, Heidelberg
  ISBN 3-9325588-14-2
- Reif, Holger
  Netz ohne Angst – Sicherheitsrisiken im Internet
  in: c't – Magazin für Computer Technik
  Ausgabe 9 / 1995
  Verlag Heinz Heise GmbH & Co KG
  ISSN 0724-8679
  S. 174-183
- RSA FAQ
  „What is SSL?"
  RSA Data Security, Inc.
  1998
  URL: http://www.rsa.com/rsalabs/faq/html/5-1-2.html ; 13.12.1998
- Schiller, J.I.
  Sicherheit im Daten-Nahverkehr
  in Spektrum der Wissenschaft
  Ausgabe 1 / 1995
  S. 50-57

- Schmidt, Jürgen
  Kidnapping im Netz – Cracker-Tool entführt Telnet-Verbindung
  Ausgabe 10 / 1997
  Verlag Heinz Heise GmbH & Co KG
  ISSN 0724-8679
  S.
- Schmidt, Jürgen
  Falsche Fährten – Mißbrauchsmöglichkeiten von ARP und ICMP
  in: c't – Magazin für Computer Technik
  Ausgabe 12 / 1997
  Verlag Heinz Heise GmbH & Co KG
  ISSN 0724-8679
  S. 246 – 248
- Schneier, Bruce
  Applied Cryptography Second Edition – Protocols, Algorithms, and Source Code in C
  2. Auflage, 1996
  John Wiley & Sons, Inc., New York, Chichester, Brisbane, Toronto, Singapore
  ISBN 0-471-11709-9
- Shaffer, Steven L.; Simon, Alan R.
  Network Security
  1. Auflage, 1994
  AP Professional, Boston, San Diego, New York, London, Sydney, Tokyo, Toronto
  ISBN 0-12-638010-4
- Shostack, Adam
  An Overview of SSL (version 2)
  1995
  URL: http://www.homeport.org/~adam/ssl.htm , 28.Oktober 1998
- Steiner, Jennifer G.; Neuman, Clifford; Schiller, Jeffrey I.
  Kerberos: An Authentication Service for Open Network Systems
  1998
  URL: ftp://ftp.pca.dfn.de/pub/docs/kerberos/kerberos-article.ps.gz , 28. Oktober 1998
- STOA-Gremium des Europäischen Parlaments
  „Eine Bewertung der Technologien für eine politische Kontrolle – Zusammenfassung Zwischenstudie September 1998"
  URL: http://www.europarl.eu.int/dg4/stoa/de/publi/166499/execsum.htm , 28. Oktober 1998
- SUN Microsystems, Inc.
  Internet-Based Secure Virtual Private Networks – The Total Cost of Ownership
  1997
  URL: http://www.sun.com/security/wp-vpn.tco/ , 2. November 1998
- Tanenbaum, Andrew S.
  Computer Networks
  Auflage, 1981
  1. Auflage, Prentice-Hall, London, Sydney, Toronto
  ISBN 0-13-165183-8
- Verschuren, Jan; Govaerts, René; Vandewalle, Joos
  ISO-OSI Security Architecture
  in: Preneel, Bart; Gouvaerts, René; Vandewalle, Joos

Computer Security and Industrial Cryptography
1993
Springer-Verlag, Berlin, Heidelberg
ISBN 3-540-57341-0
S. 179 - 192

- Ylonen, T.; Kivinen, T.; Saarinen, M.
SSH Connection Protocol
Network Working Group – INTERNET-DRAFT
1997a
URL: ftp://ftp.informatik.uni-bremen.de/pub/doc/internet-drafts/draft-ietf-secsh-connect-02.txt ,
13. Dezember 1998
- Ylonen, T.; Kivinen, T.; Saarinen, M.
SSH Protocol Architecture
Network Working Group – INTERNET-DRAFT
1997b
URL: http://www.net.lut.ac.uk/psst/draft-ietf-secsh-architecture-01.txt , 1. November 1998

Abkürzungsverzeichnis

ANSI	American National Standards Institute
ARP	Address Resolution Protocol
BIND	Berkeley Internet Name Daemon
bzw.	beziehungsweise
CA	Certification Authority
CBC	Cipher Block Chaining
CERT	Computer Emergency Response Team
CGI	Common Gateway Interface
CHAP	Challenge Handshake Authentication Protocol
CSMA/CD	Carrier Sense Multiple Access / Collision Detect
DES	Data Encryption Standard
DFN	Deutsches Forschungsnetz
d.h.	das heißt
DMZ	Demilitarisierte Zone
DNS	Domain Name Service
DSA	Data Signature Algorithm
DSS	Data Signature Standard
ECP	PPP Encryption Protocol
EDE	Encrypt Decrypt Encrypt
EKE	Encrypted Key Exchange
EU	Europäische Union
f.	folgende
FAQ	Frequently Asked Questions
ff.	fortfolgende
FTP	File Transfer Protocol
F&E	Forschung und Entwicklung
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
ID	Identifier
IDEA	International Data Encryption Algorithm
i.d.R.	in der Regel
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPSEC	IP Security
ISDN	Integrated Services Digital Network
ISO	International Standards Organization
ISP	Internet Service Provider
JVM	Java Virtual Machine
KDC	Key Distribution Center
LEAF	Law Enforcement Access Field
MAC	Message Authentication Code
MD	Message Digest
MIME	Multi Purpose Mail Extension

MIT	Massachusetts Institute of Technology
MOSS	MIME Object Security Services
NBS	National Bureau of Standards
NFS	Network Filesystem
NIST	National Institute of Standards and Technology
NSA	National Security Agency
OSI	Open Systems Interconnection
PAP	Password Authentication Protocol
PEM	Privacy Enhanced Mail
PGP	Pretty Good Privacy
POP	Post Office Protocol
PPP	Point to Point Protocol
PPTP	Point to Point Tunneling Protocol
RADIUS	Remote Authentication Dial-In User Service
RARP	Reverse Address Resolution Protocol
RAS	Remote Access Server
RC	Rivest Cipher
S.	Seite
S-HTTP	Secure HTTP
SHA	Secure Hash Algorithm
SLIP	Serial Line Internet Protocol
S/MIME	Secure Multi Purpose Mail Extension
SMTP	Simple Mail Transfer Protocol
sog.	sogenannte
SRA	Secure Remote Procedure Call Authentication
SSH	Secure Shell
SSL	Secure Socket Layer
ST	Service Ticket
STEL	Secure Telnet
STOA	Scientific and Technological Options Assessment
TCP	Transmission Control Protocol
TGS	Ticket Granting Service
TGT	Ticket Granting Ticket
TLS	Transport Layer Security
TTL	Time to Live
UDP	User Datagram Protocol
u.	und
u.ä.	und ähnliches
URL	Uniform Resource Locator
vgl.	vergleiche
VPN	Virtual Private Network
WWW	World Wide Web
z.B.	zum Beispiel
z.T.	zum Teil

Darstellungsverzeichnis

1 Einleitung

1.1 Überblick

Diese Arbeit gibt einen Überblick über das Problemfeld der „Sicherheit im Internet“. Es werden Grundkenntnisse im Bereich ISO-OSI-Netzwerkarchitektur sowie Internet-Netzwerke, d.h. TCP/IP u.ä., vorausgesetzt.[1] Es werden, soweit möglich, keine umfangreichen theoretischen Ausführungen zu Protokollen oder Diensten gegeben, die deren Vor- und Nachteile respektive deren Schwachstellen beschreiben, sondern es wird versucht, die in der Praxis häufig auftretenden Probleme darzulegen und Lösungsansätze anzubieten, die diese Schwachstellen so weit wie möglich beseitigen.

Einleitend wird eine Übersicht über notwendige organisatorische, personelle und technische Voraussetzungen und Anforderungen zur Realisierung eines Internet-Sicherheitskonzepts innerhalb eines Unternehmens gegeben. Diesem Abschnitt folgt ein Kapitel, das die notwendigen theoretischen Grundlagen Kryptographie und Signaturen erläutert. Im folgenden wird auf Schwachstellen der regelmäßig im Internet verwendeten Protokolle eingegangen. Diesem Kapitel folgen Ausführungen zum Schutz offener Systeme, d.h. von Rechnern und Rechnernetzen, die nicht nur in einem abgeschlossenen System arbeiten, sondern mit anderen Systemen verbunden sind. Danach werden die auf den Internetprotokollen aufbauenden Dienste und deren Sicherheitsprobleme erörtert. Vor der abschließenden Zusammenfassung wird dann noch eine Klasse von Problemen beleuchtet, die nur allzu häufig nicht beachtet wird: Sicherheitsprobleme durch Design- und Implementierungsfehler.

1.2 Das Problem

Warum ist nun aber „Sicherheit im Internet“ ein in zunehmendem Maße wichtiges Thema, sowohl für Privatpersonen, als auch, in ganz besonderem Maße, für Unternehmen? Die augenfälligste Antwort darauf ist die Tatsache, daß das Internet zunehmende Bedeutung im alltäglichen Leben gewinnt. Dies umfaßt im Falle von Privatpersonen i.d.R. die Nutzung von Internetangeboten und enthält bei vielen Unternehmen zusätzlich das Bereitstellen von Informationen verschiedenster Art im Internet. Dazu kommt oftmals die Verbindung verschiedener Unternehmensstandorte, die teilweise ebenfalls über das „öffentliche“ Internet realisiert ist und die zunehmende Nutzung von Business-to-Business-Lösungen über das Internet. Das Internet in seiner bisherigen Form ist allerdings primär auf die Übertragung von Informationen ausgelegt und nicht dafür, diese Informationen vor unberechtigtem Zugriff zu schützen. Dabei kann es sich um reines Mitlesen von Daten handeln, es könnte jedoch auch eine Veränderung von Informationen durch Dritte vorgenommen werden. Dies ist selbstverständlich nicht im Sinne der „ehrlichen“ Nutzer des Internets. Zusätzlich bringt eine Nutzung oder das Anbieten von Informationen im Internet häufig eine Kopplung des Firmennetzes mit dem Internet mit sich. Dies birgt die Gefahr in sich, daß Unberechtigte vom Internet in das Firmennetz eindringen und dort Informationen stehlen, verändern oder einfach nur vernichten. Dies muß im Interesse eines Unternehmens verhindert werden, da unbefugten Dritten ungewollt Informationen zugänglich werden könnten, durch deren Verwertung dem Unternehmen Schaden zufügen werden könnten.

Oft übersehen, aber ebenfalls sehr wichtig, ist auch der Schutz eines Unternehmens vor staatlichem Zu- bzw. Übergriff. Mag staatlicher Zugriff auf Daten durch die eigene Regierung noch keine Gefährdung für ein legal arbeitendes Unternehmen darstellen, so sieht dies beim Zugriff auf Informationen

[1] Eine gute Einleitung dazu findet sich in: Raepple, M. (1998), S. 26ff. und 38ff., sowie Hosenfeld, F.; Brauer, K. (1995)

des Unternehmens durch fremde Regierungen anders aus. Dazu nur drei Beispiele, von denen das erste inzwischen offiziell bestätigt ist, das zweite mehr oder minder ein offenes Geheimnis darstellt und das dritte zwar unbestätigt, aber auch unwidersprochen ist.

1. Die amerikanische Regierung in Gestalt der „National Security Agency", NSA, unterhält unter Mithilfe mehrerer Staaten, darunter Großbritannien, Australien, Kanada und Neuseeland, ein globales Abhörnetz, ECHELON genannt, mit dem sie nahezu alle Telekommunikationsverbindungen, d.h. Telefon/Telefax, Telex, Internet, Satellitenkommunikation und Mobilfunk, weltweit abhören kann.[2] Dabei wird allen Staaten, mit Ausnahme der USA, nur die Informationen zugänglich gemacht, die aus ihrem Land stammen bzw. dieses betreffen, nur die NSA sieht alle Informationen ein. Inzwischen, nach Jahren der Untätigkeit, befaßt sich mit ECHELON auch ein Untersuchungsausschuß der Europäischen Union (EU).[3] Allein schon durch die Existenz von ECHELON muß man es schon als sträflichen Leichtsinn bezeichnen, wenn Unternehmen wichtige firmeninterne Daten via Fax oder Internet ohne zusätzliche Sicherheitsvorkehrungen versenden. Angesichts von ECHELON sollte allerdings auch nicht vergessen werden, daß auch andere Staaten, in der EU allen voran Frankreich, weitreichende Abhöranlagen unterhalten.
2. Vor einigen Jahren kam es im Zusammenhang mit den Vertragsverhandlungen über den Verkauf von ICE-Hochgeschwindigkeitszügen durch die Siemens AG nach Südkorea zu unerklärlichen Vorkommnissen bezüglich der Angebotsgestaltung des französischen Mitbewerbers GEC-Alsthom, der seinen Hochgeschwindigkeitszug, den TGV, ebenfalls nach Südkorea verkaufen wollte. Der französische Konkurrent war immer in der Lage, auf ein niedrigeres Angebot durch Siemens sofort ein noch günstigeres Angebot vorzulegen. Wie inzwischen auch aus Kreisen deutscher Geheimdienste berichtet wurde, ist davon auszugehen, daß der französische Geheimdienst Faxe von Siemens, die neue Angebotsvorschläge enthielten, abgefangen und an Siemens Konkurrenten weitergeleitet hat. Außerdem kam es zur gleichen Zeit zu, erst später entdeckten, Einbrüchen in das Siemens Rechnernetz, bei denen scheinbar gezielt die Rechner angegriffen wurden, die die Angebotsdaten enthielten.
3. Das Groupware-System „Lotus Notes" ist heute eines der kommerziell meistgenutzten Softwaresysteme überhaupt. Das System bietet zum Schutz von Daten vor fremdem Zugriff eine optionale Verschlüsselungsfunktion. Notes enthält jedoch, wie für amerikanische Software üblich, in der Version für den nicht-amerikanischen Markt nur schwache Verschlüsselungsalgorithmen, was in diesem Fall bedeutet, daß die maximale Schlüssellänge stark beschränkt ist. Nun besteht seit einiger Zeit der Verdacht, daß diese relativ kurzen Schlüssel einen fixen Anteil enthalten, der wiederum der NSA bekannt sein soll. Dies würde bedeuten, daß die sowieso schon schwache Verschlüsselung für den amerikanischen Geheimdienst kein Hindernis mehr darstellt und die NSA bei Bedarf durch Notes verschlüsselte Informationen in Echtzeit entschlüsseln und diese Informationen auch amerikanischen Unternehmen zum Zwecke der Stärkung der amerikanischen Wirtschaft zugänglich machen könnte. Ein Indiz dafür, daß auch in der freien Wirtschaft schon einige wenige Unternehmen die Verschlüsselung von Notes als nicht ausreichend ansehen, kann darin gesehen werden, daß die Deutsche Bank eine zusätzliche Verschlüsselungsschicht vor Lotus-Notes-Anwendungen schaltet, die die Daten bereits außerhalb des Notes-Systems verschlüsselt.

Auch wenn die Vorfälle Nummer zwei und drei nicht wie geschildert vorgefallen sein sollten, dann sollte schon allein die Tatsache, daß diese Fälle so hätten ablaufen können, jedes Unternehmen dazu veranlassen, Vorkehrungen zu treffen, damit solche Übergriffe erst gar nicht möglich sind. Bedauer-

[2] Siehe zu weiteren Ausführungen: Hager, N. (1996)

[3] Siehe dazu: STOA-Gremium des Europäischen Parlaments (1998)

lich ist in diesem Zusammenhang die Tatsache, daß das Thema „Sicherheit im Internet“ bei vielen Unternehmen nicht forciert wird, so daß eine geeignete Infrastruktur und Unternehmenspolitik für diesen Bereich nicht etabliert werden kann.

2 Anforderungen an ein Sicherheitskonzept

2.1 Was bedeutet Sicherheit?

Um die Aspekte von Sicherheit zu betrachten, muß zuerst geklärt werden, was unter „Sicherheit im Internet“ zu verstehen ist. Sicherheit betrifft in diesem Kontext speziell den Bereich „Kommunikation“, denn i.d.R. ist Kommunikationssicherheit ein geeignetes Mittel, um Angriffe auf ein System zu verhindern. Wenn sichergestellt werden kann, daß Kommunikation zwischen zwei bestimmten Partnern stattfindet, diese Kommunikation auch nicht von Dritten verändert oder mitgelesen werden kann und auch unbefugte Dritte nicht in der Lage sind, mit Systemen zu kommunizieren, mit denen sie nicht kommunizieren sollen, ist man dem Ziel, der Sicherheit im Internet, schon deutlich näher gekommen. Nicht betrachtet werden soll in dieser Arbeit die physische Absicherung von Rechneranlagen, sowie der Einfluß von Umwelteinflüssen und Katastrophen auf die Sicherheit der Kommunikationsinfrastruktur. Genauso wenig befaßt sich die Arbeit mit der Ausarbeitung von Sicherheitsrichtlinien zur Festlegung von Zugriffsbefugnissen von Mitarbeitern, dem Datenschutz und ähnlichen Aspekten.

Bei der Betrachtung von Datenkommunikation trifft man auf zwei Ursachen für deren Mißlingen bzw. das Auftreten von Sicherheitslücken, nämlich menschliche und technische Ursachen.[4] Die durch Menschen verursachten Störungen können zufällig oder beabsichtigt sein, man könnte daher zwischen *Nachlässigkeiten* und *vorsätzlichen Eingriffen* unterscheiden. Außerdem muß zwischen *berechtigten* und *unberechtigten Personen* unterschieden werden: während Berechtigte einen Dienst im Netzwerk nutzen sollen, erhalten Unberechtigte durch Nachlässigkeit bei der Systemverwaltung oder aber durch vorsätzliche Eingriffe Zugriff auf einen Dienst. Schließlich ist zwischen *Außen- und Innentätern*, d.h. zwischen Personen außerhalb des Unternehmensnetzes und solchen, die innerhalb des Unternehmensnetzes arbeiten, zu unterscheiden.[5]

Die durch die Technik verursachten Störungen (Hard- und Software) können kurzzeitig sein, d.h. sie beheben sich nach einiger Zeit von selbst, oder sie können permanent sein, so daß Fachpersonal eingreifen muß. Dementsprechend könnte man von *Betriebstörungen* und *Systemausfällen* reden.

All diese Beeinträchtigungen sollten ernst genommen werden, da sie, je nachdem wie und wo sie auftreten, ernsthafte Auswirkungen auf ein Unternehmen haben können. Häufig neigen Mitarbeiter, die für die Sicherheit zuständig sind, dazu, Nachlässigkeiten und Betriebsstörungen weniger ernst zu nehmen, als vorsätzliche Eingriffe und Systemausfälle, mit dem Ergebnis, daß viele Methoden, die eingesetzt werden, um vorsätzliche Eingriffe zu verhindern, dazu führen, daß ernstzunehmende Nachlässigkeiten deutlich wahrscheinlicher werden. Dieser Effekt könnte beispielsweise auftreten, wenn ein Verfahren, das eingeführt wird, um eMail-Kommunikation sicherer zu gestalten, so kompliziert ist, daß die Mitarbeiter lieber darauf verzichten und ihre elektronische Post vollkommen ungeschützt verschicken.

Um ein sicheres System zu spezifizieren, sollte man nicht so sehr die Ursachen von Störungen untersuchen, als vielmehr die Auswirkungen derselben. Eine guter Ansatzpunkt, um mit den Untersuchungen zu beginnen, ist die Betrachtung der Daten, die übertragen werden und der Operationen, die aus-

[4] Siehe dazu: Hendry, M. (1995), S. 4 f.

[5] Siehe dazu: Fischer, S. et al. (1998), S. 30 f.

geführt werden, nachdem bestimmte Daten empfangen wurden.[6] Dabei muß man sich folgende Fragen stellen:

- **Wie unternehmenskritisch sind die Daten?** Welchen Effekt hätte beispielsweise eine Fehlerquote von einem Prozent bei der Datenübertragung oder der Verlust von Daten auf das Geschäftsergebnis des Unternehmens? Welchen Wert hätten die Daten für einen Wettbewerber? Oder hängt beispielsweise das **Vertrauen** in das ausführende Unternehmen bei Onlinetransaktionen vom möglichst hohen Prozentsatz korrekt ausgeführter Transaktionen ab?
- **Welchen Effekt gäbe es für das Unternehmen, wenn es nicht möglich wäre, eine bestimmte Aufgabe auszuführen?** Oder was würde geschehen, wenn die Zustellung von Daten einige Minuten oder Tage verzögert stattfinden würde?

2.2 Kriterien für Kommunikationssicherheit

Die Kriterien für Kommunikationssicherheit stellen Anforderungen an ein IT-System dar, die dieses erfüllen muß, um den Sicherheitsanforderungen seiner Benutzer zu entsprechen. Dabei werden folgende Kriterien unterschieden.[7]

2.2.1 Vertraulichkeit

Vertraulichkeit bezieht sich auf den Schutz geheimer Informationen beim Transport über das Internet vor Einblicken durch unberechtigte Dritte. Unter dieses Kriterium fallen Paßwörter, Zahlungsinformationen und auch private Korrespondenz. Teilweise gehen die Anforderungen sogar so weit, daß die bloße Existenz von Informationen nicht erkennbar sein darf. Dies wäre beispielsweise dann der Fall, wenn in einem militärischen Szenario plötzlicher starker Datenverkehr zwischen dem Hauptquartier und einer anderen Einheit aufträte. Es wäre dann wahrscheinlich, daß eine, wie auch immer geartete, Handlung bevorsteht. Man muß sich fragen, was passieren könnte, wenn beispielsweise ein Konkurrent in der Lage wäre, die übertragenen, geheimen Daten zu lesen. I.d.R. sind kryptographische Methoden ein guter Weg, um Vertraulichkeit zu sichern.

2.2.2 Datenintegrität

Datenintegrität bezieht sich auf die Möglichkeit der Veränderung von Daten. Dabei kann es sich um zufällige Veränderungen oder beabsichtigte Manipulationen von Daten durch Unberechtigte handeln. Es muß geklärt werden, welche Folgen es beispielsweise hätte, wenn ein Unberechtigter eine Nachricht abfangen und sie dann in veränderter Form weiterleiten würde. Wenn dies ernstzunehmende Folgen haben könnte, dann können entweder kryptographische Methoden eingesetzt werden, die dazu führen, daß sich eine veränderte Nachricht nicht sinnvoll entschlüsseln läßt oder man kann Prüfsummen verwenden, die von einem Angreifer nicht verändert werden können.

2.2.3 Verfügbarkeit

Verfügbarkeit bedeutet, daß nutzungsberechtigte Personen auf Informationen und Kommunikationsdienste zur rechten Zeit am rechten Ort zugreifen können. Außerdem muß gefordert werden, daß Informationen überhaupt empfangen werden können. Auf jeden Fall muß aber sichergestellt werden, daß erkannt werden kann, daß Daten fehlen, wenn sie nicht empfangen wurden. Ansonsten könnte man die Bearbeitung einer Aufgabe auf falscher Grundlage fortführen und dadurch zu falschen Ergebnissen kommen. Eine Bedrohung des Kriteriums Verfügbarkeit besteht durch die sogenannten „Denial-of-

[6] Siehe zu weiteren Ausführungen: Hendry, M. (1995), S. 11ff.

[7] Vgl. Raepple, M. (1998), S. 4 ff. und Hendry, M. (1995), S. 12 ff.

Service“-Attacken, die gerade die Nichtverfügbarkeit von Systemen für den berechtigten Nutzer zum Ziel haben.

Wie hoch der Grad an Verfügbarkeit für ein Unternehmen sein muß, hängt vom Einsatz des entsprechenden Dienstes ab. Wenn ein beliebiger öffentlicher WWW-Server ausfällt, stellt dies für das Internet keine größere Einschränkung dar. Wenn aber ein kommerzieller Internet Service Provider (ISP), der gegen Bezahlung einen Internetzugang für Unternehmen anbietet, diesen Dienst für einige Stunden nicht anbieten kann, kann dies schon Schadensersatzansprüche der Kunden nach sich ziehen, denn diese setzen i.d.R. eine ständige Erreichbarkeit und eine hohe Datendurchsatzrate voraus. Auch bedeutet es für den Betreiber eines eCommerce-Angebots entgangenen Gewinn, wenn er aufgrund eines Serverausfalls nicht in der Lage ist, seine Produkte über das Internet anzubieten.

Die Verfügbarkeit im Themenkreis Sicherheit ist zu unterscheiden von der Verfügbarkeit im Themenkreis Zuverlässigkeit. Letztere bezieht sich allgemein auf die Verfügbarkeit von Systemen unter bestimmten, i.d.R. technischen, Zuverlässigkeitskriterien, wohingegen die Verfügbarkeit unter Sicherheitsaspekten enger gefaßt wird und sich primär auf die Ausfallsicherheit aufgrund von Angriffen auf ein System bezieht.

2.2.4 Berechtigung

Häufig ist es notwendig, die Berechtigung eines Nutzers zur Verwendung eines Dienstes zu überprüfen oder sicherzustellen, daß eine Nachricht von der Person stammt, von der behauptet wird, daß sie von ihr stammt. Das Thema Berechtigung läßt sich in zwei Unterpunkte gliedern.

2.2.4.1 Authentifikation

Es muß, wenn nötig, möglich sein, daß ein Nutzer eines Dienstes einen eindeutigen Beweis seiner Identität erbringen kann. Entsprechende Mechanismen prüfen die Authentizität der vorgegebenen Identität, also die Echtheit von Personen, Organisationen oder Programmen. Ein entsprechender Dienst könnte beispielsweise bestätigen, daß eine eMail tatsächlich von dem Absender stammt, von dem sie angeblich gesendet worden sein soll. Für die Authentifikation von Personen werden häufig Paßwörter verwendet, also Zeichenfolgen, die nur der entsprechende Benutzer kennen sollte. Auch können kryptographische Verfahren eingesetzt werden.

2.2.4.2 Zugriffskontrolle

Um den Zugriff von Unberechtigten auf vertrauliche Daten zu verhindern, ist der Authentifikation häufig die Zugriffskontrolle nachgeschaltet. Sie soll erreichen, daß nur Personen, die sich entsprechend authentifiziert haben und die zum Zugriff auf bestimmte Informationen oder Dienste berechtigt wurden, diese Informationen oder Dienste auch tatsächlich nutzen dürfen.

2.2.4.3 Sicherstellung der Zustellung

Immer wenn juristische Rahmenbedingungen eine Rolle spielen, also beispielsweise bei einer Bestellung, einer Überweisung oder bei der Zustellung von Informationen zur Fristwahrung, muß eine rechtsverbindliche Kommunikation sichergestellt werden. Es muß nicht nur, gemäß 2.2.4.1, eindeutig feststellbar sein, daß eine bestimmte Person oder Institution der Absender einer Information ist, sondern es muß auch verläßlich feststellbar sein, daß eine Nachricht dem richtigen Empfänger tatsächlich zugegangen ist.

Über Authentifikation und Sicherstellung der Zustellung kann so eine verbindliche Kommunikationsbeziehung aufgebaut werden.

2.2.5 Wahrung der Anonymität

Eine Person muß die Möglichkeit haben, beispielsweise aus Gründen des Datenschutzes, Kommunikation anonym durchzuführen. Dies steht häufig dem Interesse eines Anbieters entgegen, der durch Erstellung möglichst detaillierter Kundenprofile eine Optimierung seines internetgestützten Direktmarketings, also z.B. der Werbung, die einem Nutzer auf den WWW-Seiten des Anbieters präsentiert wird, erreichen will. Auch können detaillierte Kundenprofile eine wertvolle Handelsware darstellen. Es liegt häufig nur in der Hand des Anbieters, wie vertraulich er mit Benutzerdaten umgeht und inwieweit er das informelle Selbstbestimmungsrecht eines Benutzers respektiert. Erschwerend kommt hier hinzu, daß rechtliche Regelungen in verschiedenen Staaten stark differieren.

2.3 Sicherheitsanforderungen[8]

Um ein System im Internet gegen Angriffe abzuschirmen, ist es notwendig, die Anforderungen an die Sicherheit eines Systems festzulegen. Sicherheitsmaßnahmen müssen dabei allgemein unter Abwägung unterschiedlicher Interessen geplant werden und das, wenn möglich, vom Beginn der Vorarbeiten für den Netzaufbau an, so daß Schwächen im Netzdesign frühzeitig beseitigt werden können. Die Sicherheitsmaßnahmen bestehen einerseits in der **Einschränkung von Risiken** für die zu schützenden Objekte, andererseits muß die **Benutzbarkeit** dieser Objekte so leicht wie möglich gestaltet werden, um die Akzeptanz der Sicherheitsmaßnahmen sicherzustellen. So darf beispielsweise die negative Auswirkung von Sicherheitsmaßnahmen auf die Netzwerk- oder Systemperformance nicht inakzeptabel hoch sein.[9] Diese beiden Aspekte stehen unter der Bedingung der **Realisierbarkeit**, denn die Maßnahmen müssen mit vertretbarem Personal-, Investitions- und Wartungsaufwand durchführbar sein.[10] Diese unterschiedlichen Interessen müssen gegeneinander abgewogen und entsprechend gewichtet werden.

Die Anforderungen an die Sicherheit können sehr gut durch eine **detaillierte Risikoanalyse** bestimmt werden. Diese soll alle Risiken möglichst vollständig erfassen und sie nach dem möglichen Schaden, der im Schadensfall entstehen kann, priorisieren. **Risiko** ist dabei wie folgt definiert:

$$\text{Risiko} = \text{mögliche Schadenshöhe} \times \text{Eintrittswahrscheinlichkeit}$$

Das Risiko dient dann als Maß zur Gewichtung der Schwachstellen des Systems. Es muß dann zwischen tragbaren und untragbaren Risiken unterschieden werden, und die sich daraus ergebenden, notwendigen Maßnahmen müssen entsprechend ergriffen werden.

Die Risikoanalyse läßt sich in mehrere Teilschritte untergliedern, die die unternehmensspezifischen Sicherheitsanforderungen als Ergebnis liefern:

1. Bestandsaufnahmen
2. Bedrohungsanalyse
3. Eintrittswahrscheinlichkeit und mögliche Schadenshöhe ermitteln
4. Risikobewertung

Die **Bestandsaufnahme** soll alle schützenswerten Objekte des untersuchten Systems erfassen. Dies findet i.d.R. unter Mitarbeit der für die Systeme verantwortlichen Mitarbeiter statt. Dabei sind der WWW-Server, Mailserver, Datenbankserver und Gateways, die die Verbindung zum Internet herstellen, typische Objekte der Untersuchung. Sind die Objekte erfaßt, müssen die schützenswerten Daten,

[8] Siehe dazu: Raepple, M. (1998), S. 11 ff.

[9] Siehe Shaffer, S. L.; Simon, A. R. (1994), S. 198

[10] Siehe Barz, H. W. (1991), S. 206 f.

die auf diesen Systemen gespeichert sind, identifiziert werden. Darunter fallen beispielsweise Konfigurations- und Paßwortdateien, Datenbanken oder Kommunikationsdienste.

Die **Bedrohungsanalyse** untersucht die Gefahren, die den erfaßten Objekten drohen und welchen Einfluß ein Schadensfall auf die Geschäftstätigkeit und den laufenden Betrieb des Gesamtsystems hätte. Eine solche Bedrohungsanalyse kann teilweise unter Verwendung von kommerziellen oder frei verfügbaren Programmpaketen zur Simulation möglicher Angriffsstrategien vorgenommen werden.

Die **Ermittlung der Eintrittswahrscheinlichkeiten und der möglichen Schadenshöhen** ist der schwierigste Teil der Ermittlung der Sicherheitsanforderungen, da es selten direkte Anhaltspunkte für die Angriffshäufigkeit auf einzelne Systeme gibt. Die häufig einzigen Anhaltspunkte für die Eintrittswahrscheinlichkeit sind Protokolldateien der verschiedenen Systeme. Aus diesen Protokolldateien können unter Umständen Angriffsversuche in Form von Unregelmäßigkeiten im Betrieb der Systeme erkannt werden. Außerdem gibt es nichtkommerzielle „Computer Emergency Response Teams“ (CERT), die Angriffshäufigkeiten auf verschiedene Systeme und die Abwehrmöglichkeiten dieser Angriffe ermitteln. Insgesamt erfordert die Bewertung der Eintrittswahrscheinlichkeit ein hohes Maß an Wissen und Erfahrung beim bewertenden Mitarbeiter. Die Schadenshöhe muß auf Basis der möglichen Störungen des normalen Betriebs, der möglichen Schäden durch Verlust von vertraulichen oder geheimen Daten und Image- oder Vertrauensverlusten bei tatsächlichen oder potentiellen Kunden geschätzt werden. Da der Schaden häufig nicht in Zahlen beziffert werden kann, wird oft eine Einstufung in die fünf Schadensklassen „existenzgefährdend“, „groß“, „mittel“, „gering“ und „unbedeutend“ vorgenommen. Diese Einstufung kann auch beliebig weiter verfeinert werden, so daß sie den individuellen Anforderungen entspricht.

Die **Risikobewertung** findet schließlich auf Basis des berechneten Risikos statt. Die Risiken werden dann nach ihrer Höhe priorisiert und nacheinander auf ihre Tragbarkeit untersucht. Nichttragbare Risiken müssen später im Rahmen des Sicherheitskonzepts durch entsprechende Maßnahmen möglichst vollständig beseitigt werden.

Nach Durchführung der Risikobewertung liegt ein Katalog von Sicherheitsanforderungen vor, der aber noch keine Pläne für die Implementierung von Abwehrmaßnahmen enthält. Diese Abwehrmaßnahmen werden erst im nächsten Schritt, im Rahmen des **Sicherheitskonzepts** geplant.

2.4 Sicherheitskonzept[11]

Primäre Aufgabe des Sicherheitskonzepts ist das Schließen der Sicherheitslücken, die den Sicherheitsanforderungen zuwiderlaufen. Ist ein Schließen der Sicherheitslücken nicht möglich, so muß zumindest versucht werden, das Risiko im Rahmen der Möglichkeiten zu verringern. Dabei sind **präventive Maßnahmen** zur Verhinderung eines Schadensfalles wünschenswert. Stehen solche Maßnahmen nicht zur Verfügung, sollte versucht werden, **überwachende Maßnahmen** einzusetzen, die einen Angriff bei dessen Eintritt erkennen und mögliche Abwehrmaßnahmen initiieren. Ist auch der Einsatz solcher Maßnahmen nicht möglich, muß man auf **reaktive Maßnahmen** zurückgreifen, um die Höhe des Schadens zu minimieren.
Die getroffenen Maßnahmen müssen immer im Verhältnis zum Wert des Objektes stehen, das sie schützen sollen. Da i.d.R. mit stetig steigenden Schutzkosten, die potentielle Schadenshöhe stetig abnimmt, ist es bezüglich der Schutzmaßnahmen wünschenswert, jene zu wählen, die dazu führen, daß

[11] Siehe zu weiteren Ausführungen: Raepple, M. (1998), S. 21 ff.

die Summe der Kosten für die Schutzmaßnahmen plus des geschätzten potentiellen Schadens ihr Minimum erreicht. Senkt man die Kosten für die Schutzmaßnahmen weiter unter diesen Wert ab, steigt der mögliche Schaden überproportional, und entsprechend steigen mit Minimierung der Schadenshöhe die Kosten für die Schutzmaßnahmen extrem an.
Die Erstellung des Sicherheitskonzepts setzt, genauso wie die Analyse der Sicherheitsanforderungen, Experten mit Fachwissen über technische Schutzmaßnahmen und deren Stärken und Schwächen und über gängige Standards voraus.
Das Sicherheitskonzept sollte einen langfristigen Schutz der Systeme zum Ziel haben und entsprechend auch Erweiterungsmöglichkeiten bieten. Zusätzlich sollte eine „Internet Security Policy" bestimmt werden, die Sicherheitsvorschriften und Verfahrensanweisungen bezüglich der Gefahren aus dem Internet enthält.
Endgültiges Ziel muß ein Maßnahmenbündel zur Aufrechterhaltung der Sicherheit sein, das dafür sorgt, daß regelmäßig die Sicherheitsanforderungen kritisch überprüft werden, danach das Sicherheitskonzept entsprechend den geänderten Anforderungen angepaßt wird und neue oder geänderte Maßnahmen implementiert werden.

2.5 Die Rolle der Unternehmensleitung[12]

Ein Sicherheitskonzept kann nur mit der Unterstützung seitens des gehobenen Managements realisiert und durchgesetzt werden. Diese Unterstützung ist regelmäßig durch eine Reihe von Eigenschaften der Sicherheitspolitik erkennbar. Dabei ist im folgenden unter Internetsicherheit sowohl der Einbruch in ein Unternehmensnetzes über das Internet zu verstehen, als auch das Manipulieren oder Mitlesen von Übertragungen des Unternehmens über das Internet.

- Die Unternehmensleitung ist in verantwortlicher Position in die Ausarbeitung und den Beschluß der Sicherheitsanforderungen und des Sicherheitskonzepts eingebunden.
- Der Internetsicherheit wird eine hohe Priorität innerhalb des Unternehmens zugewiesen.
- Das unabhängig von anderen Organisationseinheiten gestaltete Sicherheitskonzept wird im Rahmen des allgemeinen Budgetierungsverfahrens mit finanziellen Mitteln ausgestattet.
- Damit die Ziele und Aufgaben des Sicherheitskonzepts erreicht werden können, müssen die einzelnen Verantwortungsbereiche genau definiert werden und die notwendigen Vollmachten erteilt werden.
- Sicherheit muß ein fundamentaler Teil der Unternehmens- und Organisationsplanung sein.

Diese fünf Punkte sind eine wesentliche Voraussetzung dafür, ein effektives Sicherheitskonzept zu verwirklichen. Dabei besteht häufig eines der Hauptprobleme darin, die Unternehmensleitung überhaupt von der Notwendigkeit eines solchen Sicherheitsprogramms zu überzeugen. Diese Probleme sind primär in der technischen Natur des Sachgebiets und der Schwierigkeit begründet, den Return-on-Investment für diese Maßnahmen anzugeben, der die notwendigen Investitionen in das Sicherheitsprogramm rechtfertigen würde. Ein praktischer Ansatzpunkt, um die Zustimmung und Mitarbeit der Unternehmensleitung zur Einrichtung eines umfassenden Sicherheitskonzepts zu erreichen, kann darin bestehen, daß man, ähnlich wie beim Informationsmanagement, den Wert und die Sensibilität von Informationen bestimmt und die Ergebnisse der Unternehmensleitung darstellt, daß man Schwachstellen, Bedrohungen und den potentiell daraus resultierenden Schaden beschreibt, daß man Beispiele für Schäden bei anderen Unternehmen anführt und daß man Vorschläge für einen Ansatzpunkt für die Arbeiten zum Thema Internetsicherheit unterbreitet. Speziell beim Thema „möglicher Schaden" ist auch darauf zu achten, daß sich die Geschäftsleitung darüber im klaren ist, daß es nicht nur um den

[12] Vgl. Shaffer, S. L.; Simon, A. R. (1994), S. 198 f.

Schaden durch einen Angriff innerhalb des Unternehmens geht, sondern daß ein möglicher Schadenseintritt auch bei Kunden und Geschäftspartnern mit einem Image- und Vertrauensverlust einhergeht.

Erschwerend wirkt sich bei diesem Vorgehen das Problem der Definition und der Wertbestimmung von Information aus, allein schon wegen der starken Zeitabhängigkeit dieses Wertes, aber auch aufgrund des immateriellen Charakters von Information.

3 Kryptographie

3.1 Überblick

In diesem Abschnitt soll eine kurze Einführung in das Themengebiet Kryptographie erfolgen und eine Übersicht über die wichtigsten kryptographischen Verfahren gegeben werden, die im zivilen Umfeld verfügbar sind. Dies geschieht vor dem Hintergrund, daß die kryptographischen Verfahren eine der wesentlichen Voraussetzungen zur ”sicheren” Übertragung von Daten über offene Netze darstellen. Mit der Sicherheit der verwendeten kryptographischen Algorithmen stehen und fallen somit viele Protokolle, die den Datenverkehr im Internet sicherer machen sollen. Dabei sollen die mathematischen Aspekte nur am Rande betrachtet werden, da diese fundierte mathematische Kenntnisse in den Gebieten Informations-, Komplexitäts- und Zahlentheorie erfordern und somit den Rahmen dieser Arbeit sprengen würden.
Es sollen zwei Techniken unterschieden werden, nämlich Verschlüsselung und Message Digest, die in den folgenden zwei Abschnitten erläutert werden.

3.2 Verschlüsselung

Algorithmen zur Verschlüsselung von Daten dienen dazu, eine im Klartext vorliegende Nachricht für unbefugte Dritte ”unlesbar” zu machen, mit der Option, die Lesbarkeit für den berechtigten Empfänger wieder herzustellen. Dabei beschränkt sich dieser Text auf Verfahren für digitale Rechneranlagen. Während man in früheren Zeiten oft diese Algorithmen oder Teile von ihnen geheim gehalten hat, geht man immer mehr dazu über, sie zu veröffentlichen. Dies erklärt sich aus der Tatsache, daß früher die Geheimnisse zur Ver- und Entschlüsselung ganz oder teilweise im Algorithmus begründet lagen, während sie heute ausschließlich in den sogenannten Schlüsseln liegen. Schlüssel sind dabei nichts anderes als speziell gewählte Bitfolgen einer bestimmten Länge. Dies hat auch den Vorteil, daß sich Forscher in aller Welt mit den entwickelten Algorithmen beschäftigen können und somit einer Aufdeckung eventueller Schwachstellen Vorschub geleistet wird. Werden solche Schwachstellen nicht gefunden, kann der Anwender eines kryptographischen Verfahrens davon ausgehen, daß es sich um ein sicheres Verfahren handelt.
Es sei erwähnt, daß die in den USA und Kanada entwickelten Verfahren strengen Exportbeschränkungen der dortigen Regierungen unterliegen. Gewöhnlich sind außerhalb dieser Länder nur stark abgeschwächte Varianten der betreffenden Verfahren erhältlich, die nur einen geringen Schutz bieten.
Es sollen nun einige Verschlüsselungstechniken vorgestellt werden, wobei zwischen symmetrischen und asymmetrischen (oder Public-Key) Verfahren unterschieden werden soll. Symmetrische Verfahren verwenden sowohl zur Verschlüsselung als auch zur Entschlüsselung den gleichen Schlüssel. Asymmetrische Verfahren bedienen sich dazu zweier verschiedener Schlüssel, eines geheimen und eines öffentlichen. Nachrichten, die mit dem öffentlichen Schlüssel verschlüsselt worden sind, können nur mit dem geheimen Schlüssel wieder entschlüsselt werden und umgekehrt.

3.2.1 Symmetrische Verfahren

Ein Algorithmus mit großer praktischer Relevanz ist der Data Encryption Standard (DES), der von IBM in Zusammenarbeit mit der NSA entwickelt worden ist. Details des Algorithmus sind bis heute geheim. Das American National Standards Institute (ANSI) hat DES im Jahre 1981 zum Standard für den privaten Sektor erklärt (ANSI X3.92). Die Länge des Schlüssels für DES ist 56 Bit.[13] Durch Aus-

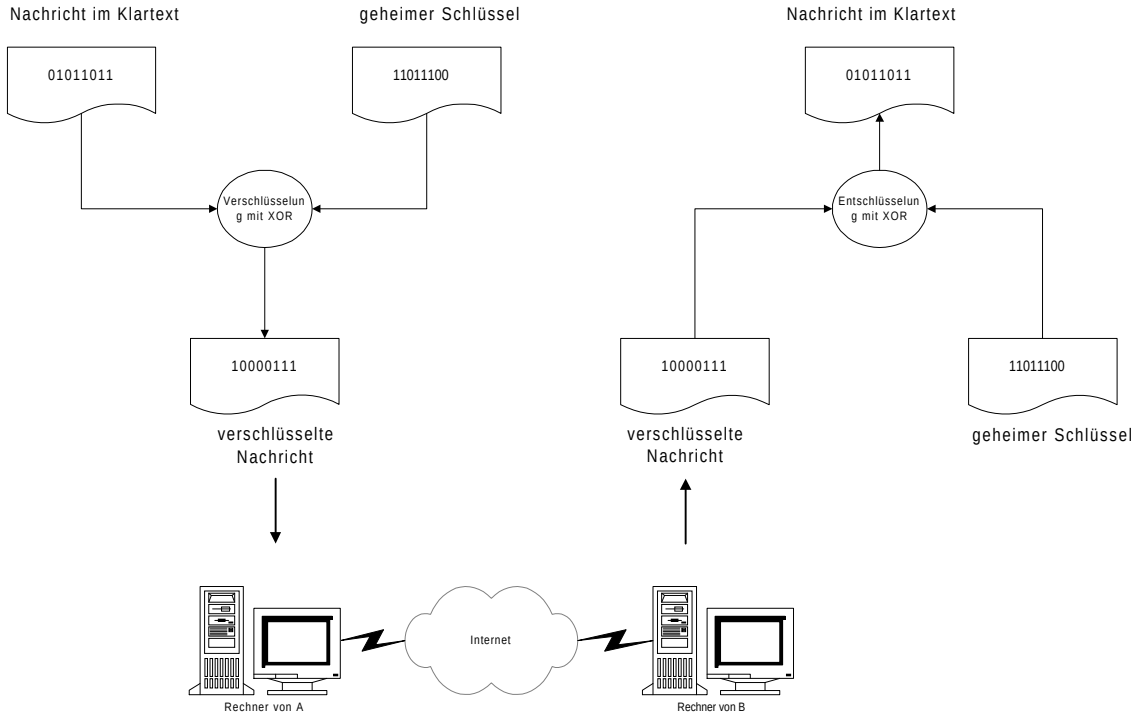
[13] Siehe zu weiteren Ausführungen: B. Schneier (1996), S. 265 - 301

probieren aller möglichen 2^{56} Kombinationen auf einer speziell dafür ausgelegten Hardware (Hardware Brute-Force Attacke) kann DES bedingt durch die geringe Schlüssellänge relativ schnell gebrochen werden. Da DES einer der meist benutzten kryptographischen Verfahren ist, ist anzunehmen, daß etwa Nachrichtendienste oder ambitionierte Unternehmen über solche Hardware verfügen.

Kosten/Schlüssellänge in Bit	40	56	64	80	112	128
$ 100.000	2 s	35 h	1 a	70.000 a	10^{14} a	10^{19} a
$ 1.000.000	0,2 s	3,5 h	37 d	7.000 a	10^{13} a	10^{18} a
$ 10.000.000	0,02 s	21 min	4 d	700 a	10^{12} a	10^{17} a
$ 100.000.000	2 ms	2 min	9 h	70 a	10^{11} a	10^{16} a
$ 1.000.000.000	0,2 ms	13 s	1 h	7 a	10^{10} a	10^{15} a
$ 10.000.000.000	0,02 ms	1 s	5,4 min	245 d	10^{9} a	10^{14} a
$ 100.000.000.000	2 µs	0,1 s	32 s	24 d	10^{8} a	10^{13} a
$ 1.000.000.000.000	0,2 µs	0,01 s	3 s	2,4 d	10^{7} a	10^{12} a
$ 10.000.000.000.000	0,02 µs	1 ms	0,3 s	6 h	10^{6} a	10^{11} a

Darstellung 1 - Mittlere geschätzte Zeit für eine Hardware Brute-Force Attacke auf DES - 1995[14]

Wegen der verhältnismäßig kurzen Schlüssellänge von DES gibt es zahlreiche Modifikationen, eine davon ist Triple-DES. Triple-DES verwendet den gewöhnlichen DES-Algorithmus, verschlüsselt die Nachricht mit einem Schlüssel, entschlüsselt sie mit einem zweiten und verschlüsselt sie wieder mit einem dritten. Die Entschlüsselung erfolgt dann durch Entschlüsselung mit dem dritten Schlüssel, Verschlüsselung mit dem zweiten und Entschlüsselung mit dem ersten. Die Idee dabei war es, DES sicherer zu machen, ohne den Algorithmus modifizieren zu müssen. Die effektive Schlüssellänge beträgt bei Triple-DES 112 Bit.[15]



Darstellung 2 - Symmetrische Verschlüsselung

[14] Vgl. B. Schneier (1996), S. 153

[15] Siehe zu weiteren Ausführungen: B. Schneier (1996), S. 358 - 363

Eine nochmals größere Schlüssellänge bietet der International Data Encryption Algorithm (IDEA) mit 128 Bit. Der Algorithmus ist vollständig offengelegt, und es ist nur eine kryptoanalytische Attacke bekannt, die IDEA gefährlich werden kann: es gibt einige sogenannte schwache Schlüssel, die nicht verwendet werden sollten. Bruce Schneier, ein anerkannter Kryptographieexperte, beschreibt den Effekt dieser Schlüssel und gibt eine sehr einfach zu implementierende Möglichkeit an, die Wahl dieser Schlüssel zu vermeiden und bezeichnet IDEA als den sichersten Algorithmus, der der Öffentlichkeit zur Zeit zur Verfügung steht.[16]
CAST als ein weiterer Vertreter der symmetrischen Algorithmen benutzt einen 64 Bit langen Schlüssel. Es ist gezeigt worden, daß der Algorithmus sicher gegen kryptoanalytische Attacken ist und somit nur mit Brute-Force gebrochen werden kann.
Neben den hier vorgestellten existiert noch eine Fülle weiterer Verfahren, die in bezug auf ihre Sicherheit z.T. sehr unterschiedlich zu bewerten sind.

3.2.2 Asymmetrische Verfahren

Im Gegensatz zu den symmetrischen Verfahren, bei denen beiden Parteien der geheimzuhaltende Schlüssel, der sowohl zur Ver- als auch zur Entschlüsselung verwendet wird, bekannt sein muß, verfolgen asymmetrische Verfahren einen grundlegend anderen Ansatz. Hier wird zwischen einem geheimen privaten und einem öffentlichen Schlüssel unterschieden, der aus dem privaten abgeleitet wird. Nachrichten, die mit dem öffentlichen Schlüssel verschlüsselt wurden, können nur mit dem privaten Schlüssel wieder entschlüsselt werden. Umgekehrt können mit dem privaten Schlüssel verschlüsselte Nachrichten ausschließlich mit dem öffentlichen Schlüssel entschlüsselt werden. Der Vorteil liegt auf der Hand: wenn eine Person *A* von den Personen *B* und *C* verschlüsselte Nachrichten empfangen möchte, erzeugt *A* ein Schlüsselpaar mit privatem und öffentlichem Schlüssel. *A* macht den öffentlichen Schlüssel sowohl *B* als auch *C* bekannt. Danach kann z.B. *B* an *A* eine Nachricht schicken, die nur *A* entschlüsseln kann, da nur *A* den privaten Schlüssel besitzt. Dabei ist es wichtig, daß *B* und *C* den öffentlichen Schlüssel aus einer vertrauenswürdigen Quelle beziehen, um eine sogenannte Man-in-the-Middle Attacke zu verhindern. Bei diesem Angriff gibt sich ein unbefugter Dritter *D* gegenüber *B* und *C* als *A* aus und teilt diesen einen anderen, als *A*s öffentlichen Schlüssel mit. Wenn *B* und *C* auf die Maskerade hereinfallen, kann *D* die Nachrichten von *B* bzw. *C* an *A* abfangen, entschlüsseln, lesen, mit dem wahren öffentlichen Schlüssel von *A* verschlüsseln und an *A* weiterleiten. So würde der Angriff vermutlich lange Zeit nicht auffallen. Asymmetrische Verfahren werden auch als Public-Key Verfahren bezeichnet.

Der bekannteste Algorithmus dieser Art ist der RSA Algorithmus, der nach seinen Erfindern Rivest, Shamir und Adleman benannt wurde. Seine Sicherheit beruht auf der Schwierigkeit der Primfaktorzerlegung großer Zahlen. Bei einer ausreichend großen Schlüssellänge gilt RSA als sicher. Dabei ist aber unbedingt zu beachten, daß die zu verwendenden Schlüssel bei gleicher Sicherheit deutlich länger sein müssen, als die der symmetrischen Verfahren. Für Daten, die langfristig geheim bleiben sollen, ist jedoch zu bedenken, daß in Zukunft eventuell erhebliche Fortschritte bei den Verfahren zur Primfaktorzerlegung erzielt werden könnten.

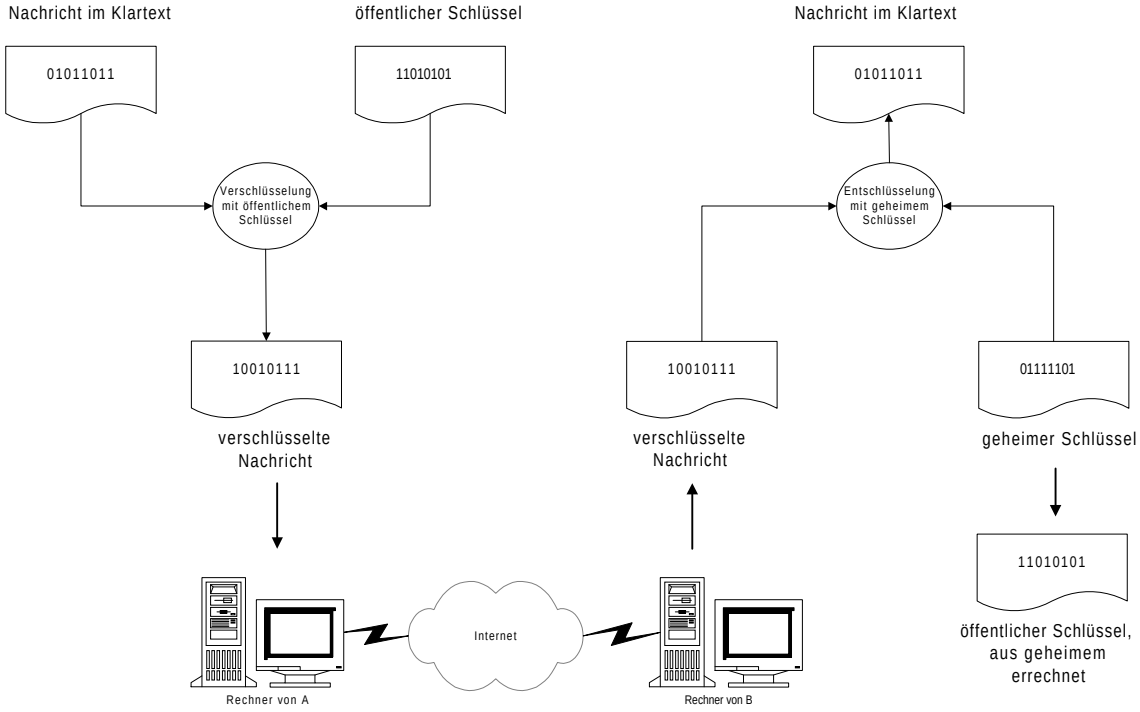
Länge symmetrischer Schlüssel	Länge asymmetrischer Schlüssel
56 Bit	384 Bit
64 Bit	512 Bit

[16] Siehe zu weiteren Ausführungen: B. Schneier (1996), S. 319 -325

Länge symmetrischer Schlüssel	Länge asymmetrischer Schlüssel
80 Bit	768 Bit
112 Bit	1792 Bit
128 Bit	2304 Bit

Darstellung 3 - Resistenz symmetrischer und asymmetrischer Schlüssel gegen Brute-Force Attacken[17]

Ein weiteres Public-Key Verfahren ist ElGamal. Dieses Verfahren unterliegt als einziges, zumindest in den USA, nicht mehr patentrechtlichem Schutz.
Ein Nachteil neben den langen Schlüsseln ist die, im Vergleich zu den symmetrischen Verfahren, hohe Komplexität im Laufzeitverhalten der Algorithmen. Asymmetrische Algorithmen sind gewöhnlich mindestens um den Faktor 1000 langsamer als symmetrische.[18] Sie eignen sich daher nur zur Verarbeitung relativ kleiner Datenmengen. Aus diesem Grund wird in der Praxis oft eine Nachricht mit einem symmetrischen Verfahren verschlüsselt, der symmetrische Schlüssel wiederum mit einem asymmetrischen verschlüsselt und an die Nachricht angehängt.



Darstellung 4 - Asymmetrische Verschlüsselung

RSA und ElGamal können auch dazu benutzt werden, eine Nachricht digital zu signieren. Dazu wird diese vom Absender mit seinem privaten Schlüssel verschlüsselt, mit der Konsequenz, daß sie nur mit dem öffentlichen Schlüssel wieder entschlüsselt werden kann. Dies ist aber, aufgrund des ungünstigen Laufzeitverhaltens, ebenfalls nur für kurze Nachrichten sinnvoll.
In der Praxis wird deshalb oft zunächst ein Message Digest[19] berechnet, der dann mit dem privaten Schlüssel eines Public-Key Verfahrens verschlüsselt wird. Darauf baut auch ein besonderes Public-Key Verfahren für digitale Signaturen auf, der Digital Signature Algorithm (DSA), der den Data Signature Standard (DSS) des National Institute of Standards and Technology (NIST) implementiert. Er ist, seinem Namen entsprechend, ausschließlich für digitale Signaturen geeignet, also nicht etwa für

[17] Vgl. B. Schneier (1996), S. 166
[18] Vgl. B. Schneier (1996), S. 33
[19] Zu Message Digest siehe Abschnitt: 3.3

das Verschlüsseln von Daten. DSA verwendet gemäß DSS die SHA-Hashfunktion[20], die mit einem Public-Key Verfahren verschlüsselt wird. Dieses Public-Key Verfahren ist eine Variante des bereits erwähnten ElGamal sowie eines weiteren Verfahrens, des Schnorr-Algorithmus. Die Schlüssellänge kann dabei zwischen 512 und 1024 Bit liegen.[21]

3.3 Message Digest

Eine Einweg-Hashfunktion *H(M)* berechnet aus einer gegebenen Nachricht *M* beliebiger Länge einen Hash oder Message Digest *h* fester Länge. Dabei müssen Einweg-Hashfunktionen die folgende Eigenschaften besitzen[22]:

- Bei gegebenem *M* ist *h* leicht zu berechnen.
- Bei gegebenem *h* ist es schwer *M* zu berechnen, so daß gilt *H(M)=h* (daher **Einweg**-Hashfunktion).
- Bei gegebenem *M* ist es schwer eine andere Nachricht *M'* zu finden, so daß gilt *H(M)=H(M')*.

Manchmal wird außerdem noch die Kollisionsfreiheit verlangt:

- Es ist schwer zwei **zufällige** Nachrichten *M* und *M'* zu finden, so daß gilt *H(M)=H(M')*.

Der Hash kann also als Fingerabdruck einer Nachricht bezeichnet werden. Verschlüsselt man diesen Fingerabdruck einer Nachricht mit dem privaten Schlüssel eines asymmetrischen Verfahrens, kann der Empfänger den Hash mit dem öffentlichen Schlüssel des Absenders entschlüsseln. Danach berechnet der Empfänger den Hash aus der empfangenen Nachricht und vergleicht ihn mit dem entschlüsselten Hash. Sind sie identisch, kann der Empfänger davon ausgehen, daß die Nachricht auf dem Weg vom Absender zum Empfänger nicht verändert oder ausgetauscht worden ist, da nur der Absender in Besitz des privaten Schlüssels ist und somit den Hash passend zum gewählten öffentlichen Schlüssel verschlüsseln konnte.

Mit Hilfe von Einweg-Hashfunktionen können auch Message Authentication Codes (MAC) berechnet werden. Dazu wird der Hash zusätzlich zur Abhängigkeit von der Nachricht von einem Schlüssel abhängig gemacht, indem z.B. der Schlüssel vor dem Hashing an die Nachricht angefügt wird. Dadurch kann bewiesen werden, daß die Nachricht von einem bestimmten Absender stammt. Dabei ist aber Vorsicht geboten, denn nicht alle Verfahren zum Berechnen von MACs nach dieser Methode gelten als sicher.

Zu den bekanntesten Hash-Algorithmen gehören MD4, sein Nachfolger MD5 sowie der Secure Hash Algorithm (SHA) der NSA.

Dem Kryptologen Hans Dobbertin ist es gelungen, MD5 teilweise zu brechen.[23] Vor diesem Hintergrund ist auch MD4 nicht sicher. Insbesondere ist MD5 nicht kollisionsfrei. Beide Algorithmen erzeugen einen 128 Bit langen Hash.

SHA ist MD 4 sehr ähnlich, erzeugt aber einen 160 Bit langen Hashwert und ist somit deutlich resistenter gegen Brute-Force Attacken. Die NSA hat mathematische Details zum SHA nicht veröffentlicht; bisher konnten keine kryptographischen Schwachstellen bei SHA gefunden werden, so daß dieser zur Zeit als der sicherste Algorithmus gilt.[24]

3.4 Verfahren zum Austausch von Schlüsseln

Das erste Verfahren zum Austausch geheimer Schlüssel war das 1976 vorgestellte Diffie-Hellman Verfahren von Whitfield Diffie und Martin Hellman. Es ist ein mathematisch relativ einfaches Verfah-

[20] Zu SHA siehe Abschnitt: 3.3

[21] Siehe zu weiteren Ausführungen: B. Schneier (1996), S. 483 - 495

[22] Vgl. B. Schneier (1996), S. 429

[23] Vgl. Network Associates, Inc. (1998), S. 110 und Newsgroup-Nachricht durch Herrn Hans Dobbertin in der Newsgroup sci.crypt am 11. Juni 1996, 14.22.03 GMT

[24] Siehe zu weiteren Ausführungen: B. Schneier (1996), S. 435 - 445

ren dessen Stärke auf der Schwierigkeit des diskreten Logarithmierens begründet ist und leicht auf mehr als zwei Parteien ausgeweitet werden kann. Diffie-Hellman in seiner ursprünglichen Form ist nicht resistent gegen Man-in-the-Middle Attacken, Abhilfe ist aber möglich.[25]

Ein weiteres Verfahren ist das Encrypted Key Exchange Verfahren (EKE), daß darauf beruht, daß die beiden Parteien, z.B. ein Client und ein Server, ein gemeinsames Geheimnis, wie ein Paßwort haben. EKE benutzt dieses Geheimnis, um zufällig erzeugte öffentliche Sitzungsschlüssel zu verschlüsseln. Es kommen also sowohl symmetrische wie asymmetrische Verfahren zum Einsatz.[26]

[25] Siehe zu weiteren Ausführungen: B. Schneier (1996), S. 513 - 516 u. S. 49f

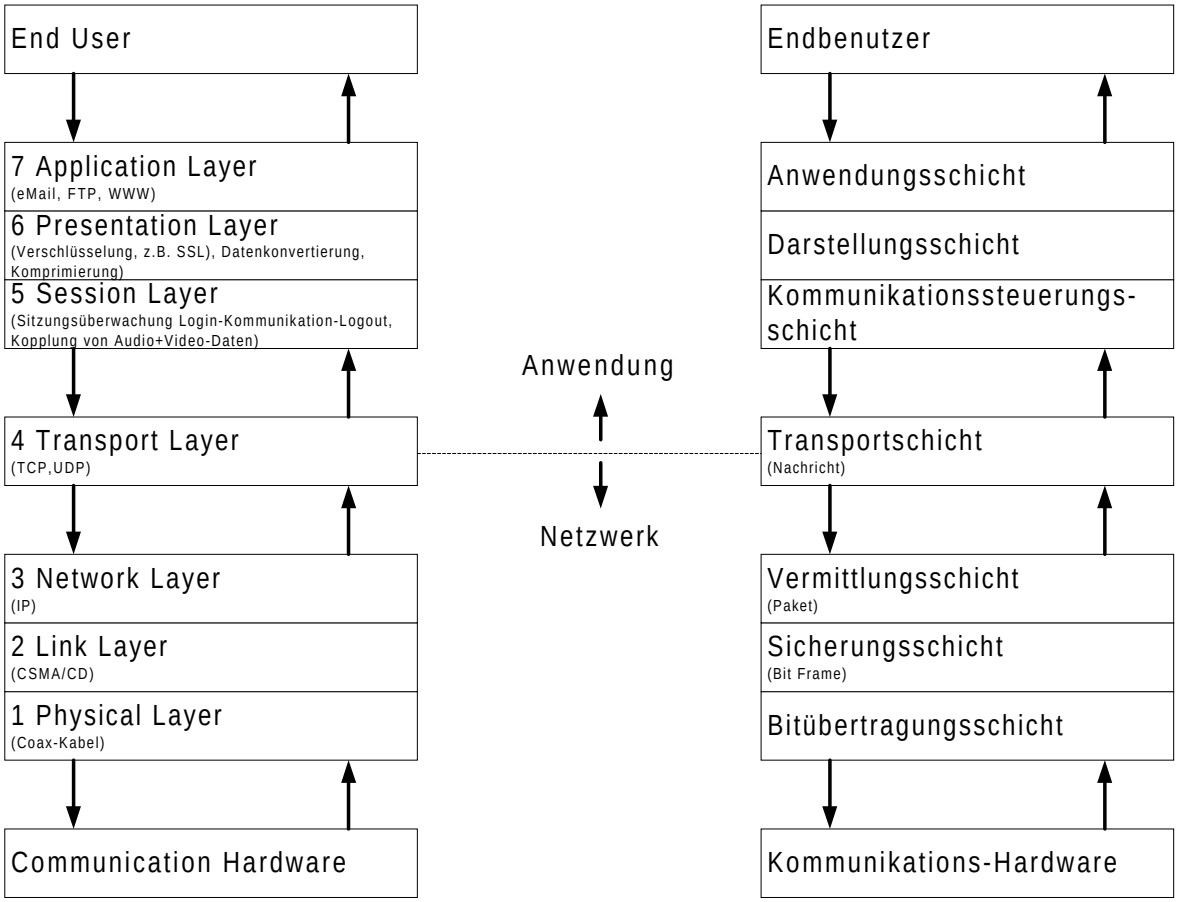
[26] Siehe zu weiteren Ausführungen: B. Schneier (1996), S.518 - 522

4 Protokolle und ihre Sicherheitsprobleme

Dieses und die folgenden Kapitel sollen einzelne Bedrohungen der Sicherheit durch die Nutzung des Internets und seiner Protokolle darlegen und Ansätze für deren Beseitigung vorstellen. Dabei werden die Angriffe nicht in allen technischen Einzelheiten dargelegt, vielmehr wird auf die Schwachstellen hingewiesen und, soweit möglich, eine Möglichkeit aufgezeigt, diese Sicherheitslücken zu verkleinern bzw. zu schließen. Dabei ist vorab festzustellen, daß die Abwehrmöglichkeiten gegen Angriff auf das Unternehmensnetz von außen wesentlich besser sind, als die Abwehr von Attacken, die aus dem Unternehmensnetz selbst stammen. Dies ist darin begründet, daß man das Unternehmensnetz zwar in gewissem Umfang gegen die Außenwelt abschirmen kann, solche Abwehrmöglichkeiten im Inneren jedoch nur im geringen Umfang vorhanden sind.[27]

4.1 Netzwerkgrundlagen[28]

Nach dem sogenannten ISO-OSI-Referenzmodell unterteilt man die Netzwerkkommunikation in sieben funktionale Schichten, die bestimmte Aufgaben bei der Kommunikation erfüllen. Dieses Referenzmodell stellt dabei eine Abstraktion von realen Implementierungen dar, die jedoch als Anschauungsobjekt sehr gut geeignet ist.



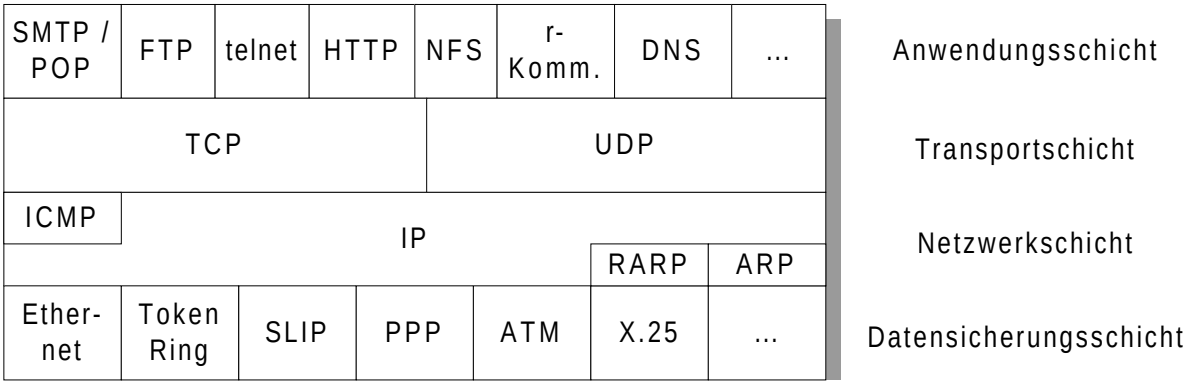
Darstellung 5 - Das ISO-OSI-Referenzmodell

In der Realität von Protokollimplementierungen sind dabei einige Schichten leer oder nochmals unterteilt, jedoch lassen sich die meisten Protokolle, so auch die im Internet gebräuchlichen, in dieses Schema einfügen.

[27] Siehe dazu Kapitel 9.1

[28] Tanenbaum, A. S (1981), S. 10-21

Die **Bitübertragungsschicht** stellt dabei die hardwarenahste Schicht dar. Hier werden Stecker-, Kabel- und elektrische Eigenschaften definiert, sowie die Kodierung von Bitfolgen mit physikalischen Signalen. Die darüberliegende **Sicherungsschicht** erfüllt die Aufgabe, eine fehlerfreie Übertragung von sogenannte Bit Frames, einer Art von hardwarenahen Datenpaketen, zwischen zwei Nachbarknoten zu gewährleisten. Als Beispiel wäre hier das Sicherungsverfahren des „Ethernet"-Netzwerkprotokolls CSMA/CD (Carrier Sense Multiple Access / Collision Detect) zu nennen, das die fehlerfreie Übertragung in einem Ethernet-Netzwerk ermöglicht. Die **Vermittlungsschicht** dient der Übertragung von Datenpaketen zwischen beliebigen Endknoten. Dies schließt die Adressierung und die Bestimmung des Übertragungsweges zwischen Start- und Endpunkt (Routing) ein. Im Internet erfüllt diese Funktion das Internet Protocol (IP). Über der Vermittlungsschicht liegt die sogenannte **Transportschicht**. Sie ermöglicht es eine logische Verbindung zwischen zwei Kommunikationspartnern aufzubauen, ermöglicht in der Regel das Verschicken von beliebig langen Nachrichten, die für den Transport über das Netzwerk in kleinere Datenpakete zerlegt und anschließend wieder zu einer großen Nachricht zusammengefügt werden. Im Internet wird diese Funktion vom Transmission Control Protocol (TCP) übernommen. Die **Kommunikationssteuerungsschicht** übernimmt die Verwaltung einer logischen Kommunikationssitzung, d.h. die Verwaltung der Kommunikation zwischen dem Login und dem Logout. Die Schicht kann ebenfalls für das Checkpointing, beispielsweise bei Transaktionsprotokollen, verwendet werden oder auch zur Kopplung von Audio- und Videodaten, beispielsweise bei einer Videokonferenz. Die **Darstellungsschicht** übernimmt Aufgaben wie die Konvertierung von Daten zwischen verschiedenen logischen Kodierungsvarianten, Komprimierung von Daten vor der Übertragung, um so weniger Daten über das Netzwerk schicken zu müssen oder die Verschlüsselung von Daten vor der Übertragung, wie sie beispielsweise die Secure Socket Layer (SSL) durchführt. Die Anwendungsschicht schließlich dient der Kommunikation zwischen spezifischen Anwendungen, wie eMail, dem File Transfer Protocol (FTP) oder dem World Wide Web (WWW).



Darstellung 6 - Funktionale Schichten und Protokolle im Internet[29]

Darstellung 6 zeigt eine Übersicht über die funktionalen Schichten und Protokolle im Internet, von denen die meisten im folgenden angesprochen werden. Mit r-Kommunikation ist hierbei die Sammlung von Unix-Programmen (rlogin, rsh, rcp), die dem entfernten Zugriff auf Rechner dienen, gemeint.

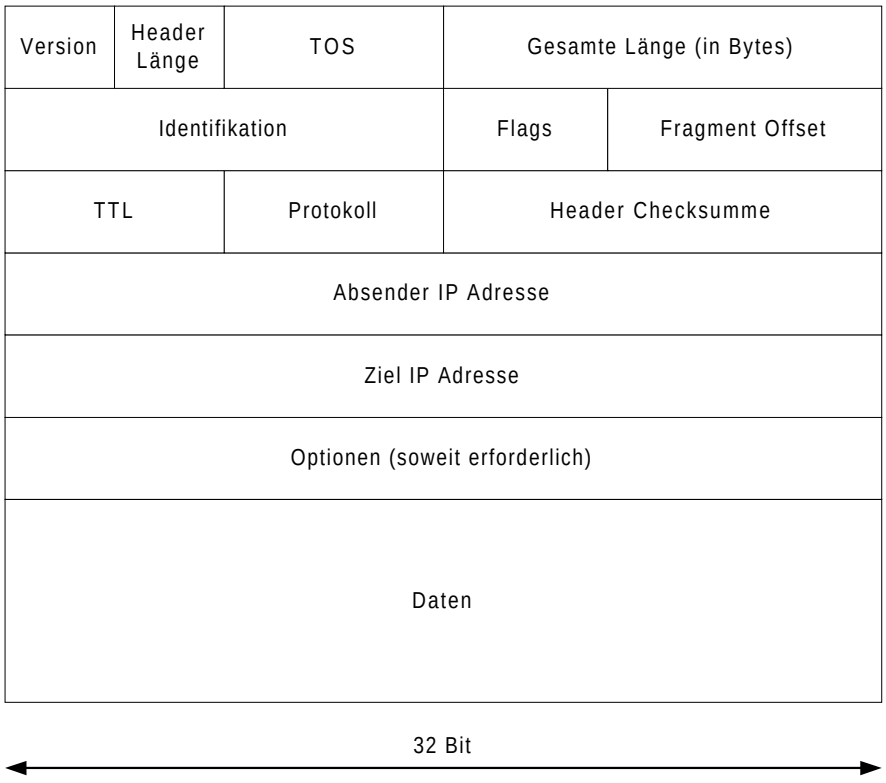
Beim Versenden von Daten werden diese von jeder Schicht in ein für diese Schicht geeignetes Format gebracht. Dieses Format wird dann von darunterliegenden Schichten jeweils in eine, für diese Schicht geeignete Form gebracht, was auch das Aufteilen von Daten in mehrere Einzelteile bedeuten kann.

[29] Vgl. Raepple, M. (1998), S. 29

Beim Empfang wird dieser Vorgang dann gerade umgekehrt, bis zuletzt wieder die originalen Nutzdaten vorliegen.

4.2 Internet Protocol, Transmission Control Protocol, User Datagram Protocol

Das **Internet Protocol**[30] (IP) ist das grundlegende Netzwerkprotokoll des Internets. Es ist speziell für die Weiterleitung von Datenpaketen, sogenannten IP-Datagrammen, von einem Punkt A zu einem Punkt B zuständig. Ein Datagramm stellt dabei die Basiseinheit für den Datentransfer dar. A und B verwenden hierbei eine weltweit eindeutige, 32 Bit lange Adresse. Die zentrale Aufgabe des IP Protokolls ist dabei, einen Weg zwischen A und B zu ermitteln, über den das Datenpaket transportiert werden kann. Dies geschieht mit Hilfe spezieller Rechner, sogenannter Router. Ob am Ziel alle abgesendeten Pakete ankommen oder ob die Sendereihenfolge bewahrt bleibt, wird hierbei nicht sicher gestellt, dies wird durch den Einsatz des **Transmission Control Protocol**[31] (TCP), das auf dem IP aufsetzt, gewährleistet. Das IP wird als verbindungslos bezeichnet, da zwischen Sender und Empfänger kein über die Nutzdaten hinausgehender Informationsaustausch stattfindet und der Empfänger bis zum Eintreffen eines Datenpaketes nichts vom Sender weiß.



Darstellung 7 - IP Paket[32]

Das Versions-Feld (4 Bit) des IP-Pakets gibt die Protokollversionsnummer an, nach deren Definition das Paket zusammengestellt wurde. Momentan ist Version 4 im Einsatz. Zwar ist Version 6 schon definiert, jedoch wird Version 4 aus Kompatibilitätsgründen noch einige Zeit verwendet werden. Die Header-Länge (4 Bit) gibt die Länge des IP-Headers ohne die Nutzdaten an. Die minimale und übliche Länge eines IP-Headers ist 20 Byte. Die Länge wird in vier Byte Schritten angegeben. Das TOS-Feld kann den „Type of Service“, den Servicetyp, angeben. Der Eintrag wird jedoch nur selten verwendet.

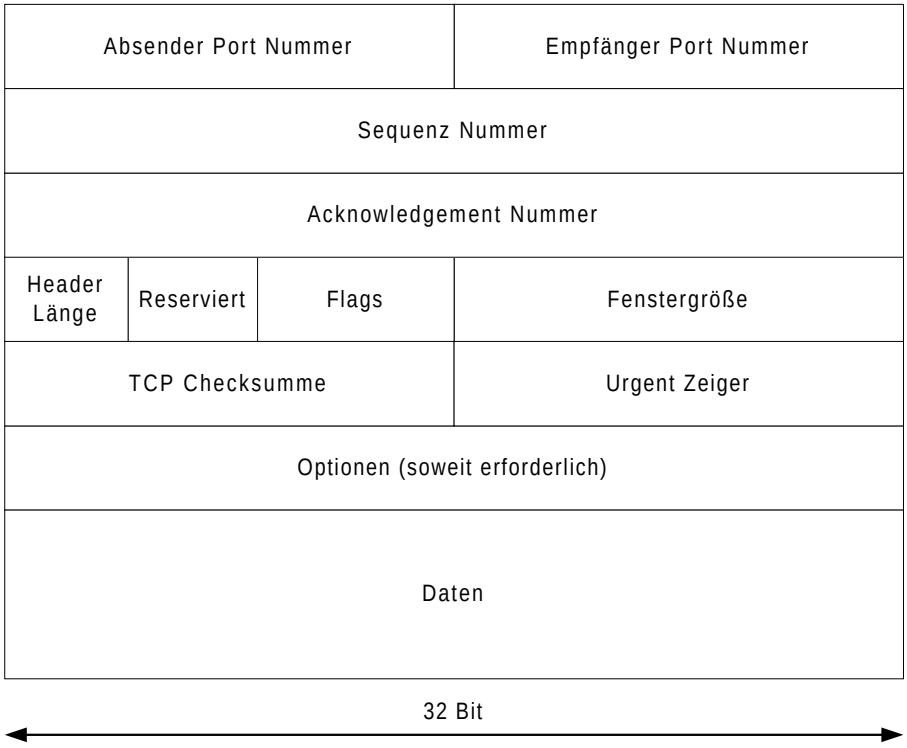
[30] Für eine detailliertere Darstellung siehe: Raepple, M. (1998), S. 38 und Oppliger, R. (1998), S. 34 ff.

[31] Siehe zu weiteren Ausführungen: Raepple, M. (1998), S. 40 f. und Oppliger, R. (1998), S. 42 ff.

[32] Oppliger, R. (1998), S. 35 ff.

Die „Gesamte Länge“ gibt die Länge des gesamten IP-Pakets, inklusive Header, an. Das Identifikations-Feld (16 Bit), die Flags (3 Bit) und der Fragment Offset (13 Bit) werden verwendet, um IP Pakete bei Bedarf in mehrere kleinere Pakete zu zerlegen und später wieder zusammenzusetzen. Das TTL-Feld (Time to Live, Lebenszeit, 8 Bit) gibt die verbleibende Lebenszeit des Paketes in Sekunden an. Wann immer das Paket von einem Rechner übertragen wird, wird der Wert um mindestens eins vermindert. Wenn das Feld den Wert 0 erreicht, wird das Paket verworfen. Das Protokoll-Feld (8 Bit) gibt an, was für Protokolldaten mit diesem IP-Paket versandt werden, dies kann u.a. das Internet Control Message Protocol (siehe Kapitel 4.3), das Transmission Control Protocol oder das User Datagram Protocol sein. Das Checksummen-Feld enthält eine Checksumme für den Header, um einfache Übertragungsfehler erkennen zu können. Die Absender- und Zieladress-Felder enthalten die entsprechenden IP-Adressen. Das Optionen-Feld wird zur Speicherung verschiedener Informationen verwendet, beispielsweise, um einen Transportweg für das Paket vorzugeben oder aber den Transportweg zurückverfolgen zu können.

Auf dem IP aufbauend und als Transportprotokolle bezeichnet, gibt es zwei Protokolle: das bereits erwähnt TCP und das **User Datagram Protocol**[33] (UDP). Das TCP ist für den „sicheren“ Transport von Datenpaketen zwischen A und B zuständig. Mit „sicher“ ist allerdings nur gemeint, daß TCP vor der Datenübertragung eine Verbindung mit dem Empfänger aufbaut, daß es Prüfsummen für die Nutzdaten eines Paketes gibt, die die Wahrscheinlichkeit zufälliger Datenveränderungen verringern, und daß die Datenpakete durchnumeriert sind, so daß eine Veränderung der Reihenfolge oder das Fehlen eines Paketes festgestellt werden kann. Auch wird der Empfang eines Datenpaketes vom Empfänger dem Sender mitgeteilt. Das UDP reicht dagegen nur die Fähigkeiten des IP an Anwendungen weiter und wird nur benötigt, damit mehrere Anwendungen gleichzeitig Datagramme an verschiedene Empfänger schicken können. Die Protokolle IP, TCP und UDP werden häufig zusammenfassend als TCP/IP bezeichnet.



Darstellung 8 - TCP-Paket[34]

[33] Für eine detailliertere Darstellung siehe: Raepple, M. (1998), S. 41 und Oppliger, R. (1998), S. 48 ff.

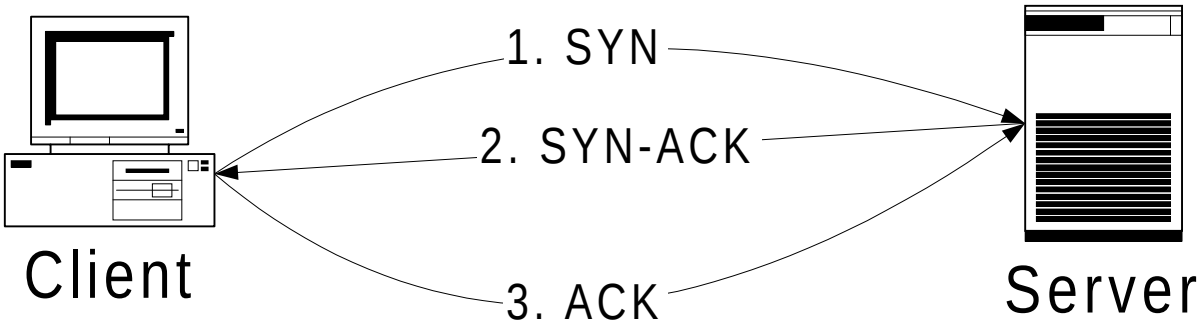
[34] Vgl. Oppliger, Rolf (1998), S. 42 ff.

Die Port-Nummern (je 16 Bit) ergeben, zusammen mit den IP-Adressen aus dem IP-Header, einen eindeutigen Kommunikationskanal zwischen Sender und Empfänger. Die Sequenz-Nummer (32 Bit) dient dazu, den Offset der Nutzdaten dieses TCP-Pakets innerhalb der aktuellen Nachricht zu bestimmen. Die Acknowledgement-Nummer (32 Bit) dient zur Bestätigung des erfolgreichen Empfangs von Daten. Das Header-Länge-Feld (4 Bit) gibt die Länge des Header des TCP-Pakets in vier Byte Schritten an. Das reservierte Feld (6 Bit) ist für zukünftige Zwecke gedacht und wird momentan nicht verwendet. Das Flags-Feld (6 Bit) enthält bestimmte Statusbits für das TCP-Paket. Diese können u.a. sein:

- ACK – ist gesetzt, wenn die Acknowledgement Nummer einen sinnvollen Wert enthält
- SYN – gibt an, daß das Sequenz-Nummer-Feld einen gültigen Startwert für den Kommunikationsaufbau enthält

Das Feld Fenstergröße (16 Bit) gibt die Anzahl von Datenbytes an, die ein Sender bereit ist, selbst zu empfangen. TCP nutzt dieses Feld zur Flußkontrolle, um die Überlastung von Knoten zu verhindern. Die Checksumme (16 Bit) dient der Entdeckung von Übertragungsfehlern. Der Urgent Zeiger gibt eine Byteposition in den Nutzdaten an, ab der Daten gespeichert sind, die vorrangig bearbeitet werden sollen. Das Optionen-Feld kann spezielle TCP-Optionen enthalten, die jedoch heutzutage nur selten benutzt werden. Das Datenfeld enthält die Nutzdaten des TCP-Pakets, bis zu 64 kByte.

Im folgenden sollen einige der Angriffspunkte auf IP, TCP und UDP aufgezeigt werden. Viele dieser Attacken basieren auf dem sogenannten **IP Spoofing**. Damit ist gemeint, daß ein Angreifer TCP/IP-Pakete mit einer gefälschten IP-Absenderadresse verschickt, um so auf Rechner zugreifen zu können, auf die er normalerweise nicht zugreifen darf.
Bei der ersten Angriffsmöglichkeit, dem sogenannten **SYN-Flooding**[35], handelt es sich um einen sogenannte **Denial-of-Service**-Angriff, d.h. bei seiner Durchführung erreicht ein Angreifer, daß der angegriffene Dienst selbst für berechtigte Nutzer nicht mehr zur Verfügung steht. Der Angriffspunkt ist der TCP/IP-Verbindungsaufbau. Dieser geschieht durch ein einfaches Protokoll zwischen Client und Server.



Darstellung 9 - TCP/IP-Verbindungsaufbau

Der Client, der eine Verbindung zu einem Server aufbauen möchte, sendet diesem ein spezielles SYN-Paket (SYN = Synchronize = Synchronisieren), d.h. ein TCP-Paket, bei dem das SYN-Bit im Feld Flags gesetzt ist und eine initiale Sequenznummer im Sequenz-Nummer-Feld gespeichert ist. Daraufhin antwortet der Server mit einem SYN-ACK-Paket (ACK = Acknowledge = Bestätigung), in diesem Paket ist sowohl das SYN-, als auch das ACK-Flag gesetzt. Dieses Paket beantwortet der Client sei-

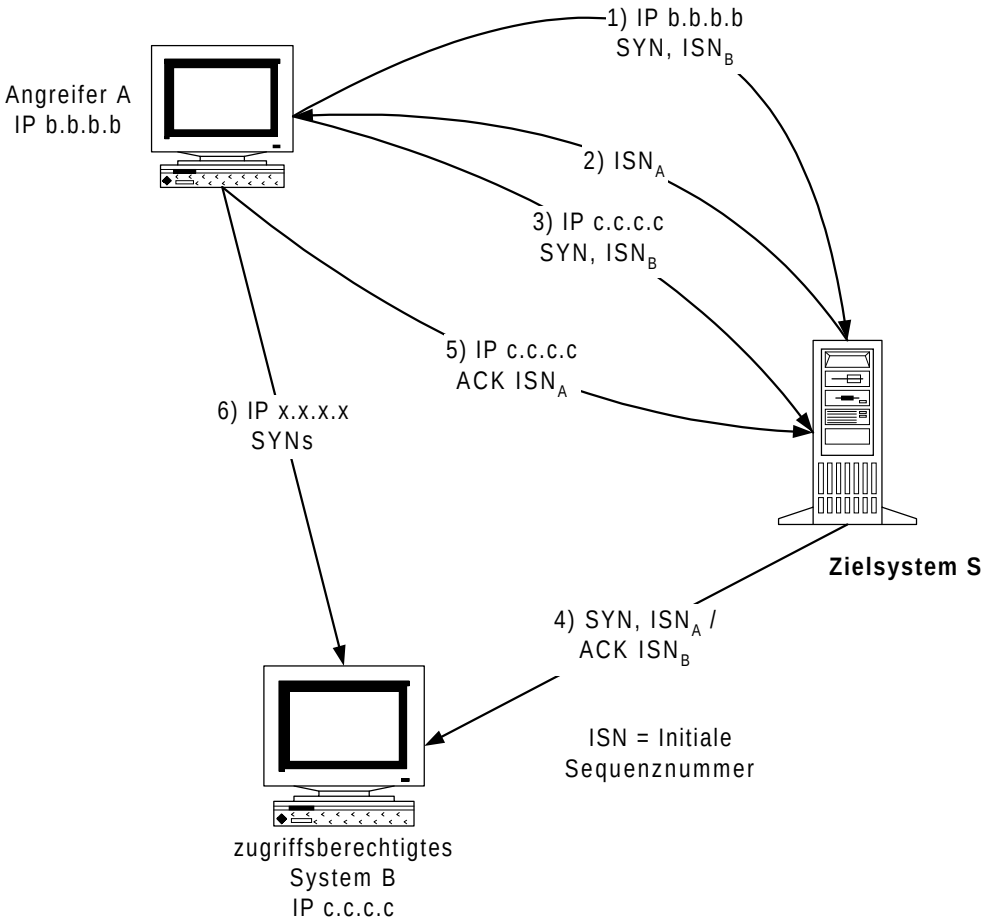
[35] Siehe zu weiteren Ausführungen: CERT-Advisory CA-96.21 (1996) und Raepple, M. (1998), S. 63

nerseits mit einem ACK-Paket (ACK-Flag gesetzt). Nach diesen drei Schritten gilt die virtuelle Verbindung zwischen Client und Server als aufgebaut. In der Zeit zwischen dem Senden des SYN-ACK-Pakets und dem Empfang des ACK-Pakets muß der Server die „halb-aufgebauten" Verbindungen in einer internen Tabelle temporär speichern. Diese Tabelle ist genau die Schwachstelle des Systems, denn sie hat nur eine endliche Größe. Das SYN-Flooding arbeitet nun so, daß ein Angreifer eine große Zahl von SYN-Paketen an einen Server sendet und jedes dieser Pakete mit einer anderen Absenderadresse versieht. Der Server schickt daraufhin SYN-ACK-Pakete an die angeblichen Absender, die jedoch nicht auf diese reagieren, da sie von einem Verbindungsaufbau ihrerseits nichts wissen. Da also die ACK-Pakete, die den Verbindungsaufbau bestätigen, ausbleiben, werden mehr und mehr Einträge der Tabelle gefüllt. Da alte Einträge erst nach einem bestimmten Zeitlimit (Timeout) aus der Tabelle gelöscht werden, muß der Angreifer nur eine hohe Anzahl von SYN-Paketen versenden, um die Tabelle, die die halb-aufgebauten Verbindungen speichert, zum Überlaufen zu bringen. Ab diesem Zeitpunkt kann kein anderer Client mehr Verbindung zum Server aufnehmen. Häufig führt ein längerer SYN-Flooding Angriff auch dazu, daß der Server den kompletten Hauptspeicher für die Tabelle der halb-aufgebauten Verbindungen verbraucht, abstürzt oder anderweitig gestört ist. Durch den Server respektive den Administrator kann ein solcher Angriff nur erkannt werden, wenn er von Dritten hört, daß ein Zugriff auf den Server nicht möglich ist, obwohl eigentlich keine Störung vorliegt oder indem der Administrator in den Logdateien des Servers entdeckt, daß die Tabelle, die die halb-aufgebauten Verbindungen speichert, ständig überläuft. Es gibt für das SYN-Flooding keine vollständige Abwehrmöglichkeit. Auch ist es i.d.R. nicht möglich, den Angreifer ausfindig zu machen, da die Absenderadressen der Datenpakete gefälscht sind und diese nahezu die einzige Möglichkeit darstellen die Herkunft zu ermitteln. Es ist ausschließlich möglich, die Angriffsmöglichkeiten einzuschränken, indem man auf einen Paketfilter[36] zurückgreift. Man kann dann zumindest dafür sorgen, daß ein Angreifer, der außerhalb des Unternehmensnetzes sitzt, keine SYN-Pakete mit Absenderadressen innerhalb des Unternehmensnetzes senden kann, indem man diese Pakete erst gar nicht ins Unternehmensnetz weiterleitet. Um Dritte außerhalb des eigenen Netzes zu schützen, ist es wiederum möglich, Angreifer, die aus dem Unternehmensnetz heraus Dritte angreifen wollen, daran zu hindern, Pakete aus dem Unternehmensnetz zu versenden, deren Absendeadresse nicht im Unternehmensnetz liegt, indem man solche Pakete beim Verlassen des Unternehmensnetzes herausfiltert.

Eine weiterer IP Spoofing-Angriff ist die sogenannte **TCP-Sequenznummern-Attacke**. Während des Verbindungsaufbaus werden im TCP-Protokoll auch die Startwerte der Sequenznummer der TCP/IP-Pakete zwischen Server und Client vereinbart. Der Angriff beginnt damit, daß der Angreifer A sowohl den Server S, als auch ein auf diesen Server zugriffsberechtigtes System B kennt. Der Angriff setzt zwar ein gutes Timing und etwas Glück seitens des Angreifers voraus, wird jedoch häufig mit Erfolg eingesetzt. Der Angriff beginnt mit einem berechtigten Zugriff des Angreifers A auf einen öffentlichen Dienst des Servers S, indem der Angreifer einen legalen Verbindungsaufbau mit SYN initiiert. Aus der Antwort auf sein SYN-Paket erfährt der Angreifer die aktuelle TCP-Sequenznummer des Servers. Der Angreifer schickt nun sehr schnell ein SYN-Paket mit dem gefälschten Absender des Systems B an den Server. Der Server S bestätigt nun dem System B den Versuch des Verbindungsaufbaus und schickt B seinerseits ein SYN mit einer Sequenznummer. Eben diese Sequenznummer kann der Angreifer A voraussagen und obwohl er die Antwort des Servers nicht empfängt, schickt er ein ACK-Paket mit der berechneten Sequenznummer an den Server. Stimmt die Sequenznummer, besteht nun eine Kommunikationsverbindung zwischen dem System B und dem Server und der Angreifer kann blind, aber ohne Kenntnis der Antworten des Servers TCP/IP-Pakete mit Befehlen an den Server schicken. Das System B würde hier jedoch den Plan des Angreifers unterlaufen, da es, nach Erhalt des

[36] Siehe Kapitel 5.1

ACK für das es keinen SYN geschickt hat, dem Server mitteilen würde, daß der Verbindungsversuch nicht vom ihm stammt. Daraufhin würde der Server die Verbindungsinformationen verwerfen. Damit genau dies nicht geschieht, blockiert der Angreifer das System B frühzeitig mittels SYN-Flooding, so daß B das ACK des Servers gar nicht empfangen kann. Für den Server ist eine TCP-Sequenznummern-Attacke nicht zu bemerken. Um sich gegen solche Angriffe zu schützen, können kryptographische Verfahren eingesetzt werden oder man verbessert die Vergabe der Sequenznummern, indem man für jede Verbindung unterschiedliche und damit nicht vorhersehbare Sequenznummern vergibt. Dies ist jedoch mit starken Eingriffen ins Betriebssystem verbunden und somit häufig nicht praktikabel, außer der Hersteller des Betriebssystems ändert die Strategie.



Darstellung 10 – TCP-Sequenznummern-Attacke[37]

Eine weitere Angriffsmöglichkeit besteht im sogenannten **Session-Hijacking**[38]. Hierbei belauscht ein Angreifer die TCP/IP-Kommunikation zwischen Server und Client. An einer bestimmten Stelle, antwortet nun der Angreifer auf ein Datenpaket des Servers, bevor dies der berechtigte Client tun kann. Auch hier verwendet der Angreifer das IP Spoofing. Von dem Moment ab, an dem sich der Angreifer in die Kommunikation eingeklinkt hat, stimmt die Synchronisation der Sequenznummer zwischen Server und berechtigtem Client nicht mehr. Daher treffen beim berechtigten Client vom Server versendete Anforderungen zum erneuten Senden der originalen Client-Anfragen ein, die jetzt aber, aus Sicht des Servers, immer eine falsche Sequenznummer tragen.. Die Pakete des Servers tragen nun aber auch, zumindest aus Sicht des Clients, immer falsche Sequenznummern. Diese bestätigt nun der Client

[37] Vgl. Raepple, M. (1998), S. 67
[38] Vgl. Raepple, M. (1998), S. 68

mit einem ACK und wiederum falscher Sequenznummer. Aus Sicht des Clients ist die Verbindung ohne ersichtlichen Grund abgebrochen. Der Benutzer schiebt dies allzu häufig auf einen Netzwerkfehler, da er an dieser Stelle zu wenig durch unternehmensinterne Maßnahmen sensibilisiert ist. Für einen Administrator erkennbar ist diese Attacke nur durch das, als „ACK-Storm“ bezeichnete, starke Ansteigen der Anzahl von ACK-Paketen. Wichtig ist, daß ein Angreifer diese Attacke nur führen kann, wenn er im gleichen TCP/IP-Segment wie der Server oder der Client sitzt oder aber auf der Route, die für die Pakete zwischen Client und Server gewählt wird, was aber oft schwer vorherzusagen ist. Session-Hijacking ist u.a. ein schwerwiegendes Problem beim „Kidnappen“ von Telnet-Sitzungen[39].

Die bisher vorgestellten Angriffe muten im Vergleich zu Angriffen auf UDP-Verbindungen kompliziert an. Da bei UDP auf Protokollebene keinerlei Kontrolle der Übertragung stattfindet, können hier Pakete verändert, gelöscht oder eingeschleust werden, ohne das dies festzustellen wäre. Wenn hier nicht auf Applikationsebene Vorsorge getroffen wird, sind solche Übertragungen als äußerst gefährdet einzustufen.

4.3 Internet Control Message Protocol

Das **Internet Control Message Protocol**[40] (ICMP) wird für Status- und Fehlermeldungen des IP, TCP und UDP verwendet. Dies geschieht immer dann, wenn Probleme im Netzwerk auftreten, wie beispielsweise die Überlastung eines Rechners, weil zu viele Pakete gleichzeitig eintreffen. Dieser hat dann die Möglichkeit, den sendenden Rechner zur Verringerung seiner Sendeleistung, d.h. der Anzahl abgesandter Pakete, aufzufordern (sog. Source Quench). Weitere Aufgaben sind die Beeinflussung des Routings (sog. Redirect), das Feststellen der Erreichbarkeit eines TCP/IP-Rechners (sog. Echo Request / Echo Reply) und auch die Mitteilung, daß ein Rechner nicht erreichbar ist (Destination Unreachable). Die ICMP-Meldungen werden in IP-Datagramme gekapselt. Da das Protokoll zur Übermittlung von netzwerkspezifischen Informationen eingesetzt wird, zählt man es, wie das IP-Protokoll zu den Netzwerkprotokollen, auch wenn die Verwendung von IP-Datagrammen eine Zuordnung zu den Transportprotokollen suggeriert. Um eine ICMP-Meldung zu versenden, wird die ICMP-Meldung in ein IP-Paket gekapselt und das Protokollfeld des IP-Pakets auf „ICMP“ gesetzt.

Eine Angriffsmöglichkeit über ICMP liegt in der Funktionalität, daß Router einem Rechner mitteilen, daß er eine andere, schnellere Route für die Versendung von Daten an ein bestimmtes Ziel verwenden soll. Der Router verwendet hierfür eine Redirect-Nachricht. Ein Angreifer kann nun einem anzugreifenden Rechner eine gefälschte Redirect-Nachricht senden, die als Absender einen, dem angegriffenen Rechner bekannten Router vorgibt (IP Spoofing) und ihm mitteilen, daß er seine Pakete über den Rechner des Angreifers senden soll. Daraufhin hat der Angreifer Zugriff auf alle Datenpakete, die der angegriffene Rechner versendet und kann diese verändern oder mitlesen. Eine weitere Attacke beruht darauf, die Routingtabelle eines anzugreifenden Rechners künstlich so weit zu vergrößern, daß das Bestimmen einer Senderoute innerhalb des Rechners extrem lange dauert. Somit fällt dieser Angriff in den Bereich der Denial-of-Service-Attacken. Viele Router und Unix-Derivate lassen sich von diesem Angriff in ihrer Funktion jedoch nicht zu stark beeinflussen, da sie hochoptimierte Algorithmen zur Bestimmung einer zu wählenden Route verwenden.[41]
Weitere Angriffsmöglichkeiten bestehen über das Vortäuschen der Unerreichbarkeit eines Zielrechners, indem gefälschte Destination Unreachable-Nachrichten verschickt werden, was den Rechner, der

[39] Siehe Kapitel 6.3

[40] Vgl. Raepple, M. (1998), S. 39 und Oppliger, R. (1998), S. 41

[41] Vgl. Schmidt, J. (12 / 1997)

diese empfängt, dazu veranlaßt, die Kommunikation mit dem Zielrechner abzubrechen. Mittels der Source Quench-Nachricht kann ein Angreifer die Sendeleistung eines Rechners extrem drosseln oder gar stoppen, indem er dem Rechner beispielsweise vorgaukelt, daß ein Router, der von diesem Rechner angesprochen wird, überlastet sei. Ein nicht zu unterschätzender Angriffspunkt ist auch der sogenannte Echo Request, der einen Rechner veranlaßt, ein Echo Reply zu senden. Diese, beispielsweise über das Betriebssystemkommando „ping“ sendbare Nachricht, kann von einem Angreifer verwendet werden, um die Netzwerktopologie zu erkunden, indem er ermittelt, welche IP-Adressen im Netz verwendet werden. Des weiteren kann man, ähnlich wie mit der SYN-Flooding-Attacke[42], mittels einer Ping-Flooding-Attacke einen Denial-of-Service-Angriff durchführen. Eine weitere Möglichkeit mittels ICMP einen Angriff auf einen Rechner durchzuführen, ist das sogenannte ICMP-Tunneling. Dieser Angriff setzt allerdings voraus, daß auf dem Zielrechner ein vom Angreifer veränderter ICMP-Server vorhanden ist. Der Angriff beruht auf der Tatsache, daß ICMP-Nachrichten normalerweise keine Nutzdaten enthalten, aber der Platz dafür vorhanden ist. Man sendet dann ein unauffälliges ICMP-Paket an den Zielrechner und der dort installierte ICMP-Server extrahiert dann aus dem Paket die Nutzdaten, die Befehle enthalten, die dann vom Server ausgeführt werden.[43]

Es gibt keine effiziente Möglichkeit sich gegen ICMP-Angriffe aus dem Inneren eines Netzes zu schützen. Die beste Möglichkeit sich gegen ICMP-Angriffe von Außen zu schützen, besteht im Einsatz eines Paketfilters[44]. Da es nicht praktikabel ist, die Verwendung von ICMP-Nachrichten gänzlich zu verbieten, da beispielsweise das Verbieten eines Echo Requests einen Kunden veranlassen könnte, zu glauben, daß ein Server nicht aktiv sei, ist nur eine sehr genaue Spezifikation von Regeln, wer an wen welche Nachrichten schicken darf, als Maßnahme für einen bedingte Schutz zu sehen.

4.4 (Reverse) Address Resolution Protocol

Das **Address Resolution Protocol** (ARP) und das **Reverse Address Resolution Protocol** (RARP)[45] dienen der Zuordnung von logischen Internetadressen zu physikalischen Netzwerkadressen und umgekehrt. Dies ist notwendig, da den Rechnern zwar eine spezielle IP-Adresse zugeordnet ist, sie jedoch im lokalen Netz über eine physikalische Netzwerkadresse angesprochen werden. Die am häufigsten eingesetzten lokalen Netzwerke sind Ethernet und Token-Ring. Dementsprechend würde eine ARP-Anfrage bezüglich einer IP-Adresse im lokalen Netz beispielsweise die Ethernet-Adresse der Netzwerkkarte des Rechners, dem diese IP-Adresse zugewiesen wurde, zurückliefern. Diese funktioniert natürlich nur, sofern die IP-Adresse im lokalen Netz verwendet wird. Das RARP liefert im umgekehrten Fall zu einer physikalischen Adresse die zugehörige IP-Adresse. RARP wird i.d.R. verwendet, wenn Rechner ohne Massenspeicher über das Netz gebootet werden und während des Hochfahrens eine IP-Adresse zugeordnet bekommen müssen.

ARP-Attacken sind immer auf das lokale Netz beschränkt, da ARP-Pakete von Routern nicht weitergeleitet werden. Die einfachste Version einer ARP-Attacke ist das Versenden eines ARP-Broadcasts, d.h. eines ARP-Paketes, das an alle Teilnehmer im aktuellen Segment gerichtet ist, der eine Anfrage bezüglich der Hardwareadresse einer unbekannten IP-Adresse enthält. Da kein Rechner und kein Router diese Anfrage beantworten kann, wird das Paket für einige Zeit ständig weitergeleitet. Dies führt zu einem hohen Verbrauch an Bandbreite, insbesondere, wenn der Angreifer viele solcher Anfragen gleichzeitig versendet. Bei diesem Angriff spricht man auch von „**Broadcast Storms**“.[46] Eine weitere Möglichkeit des Mißbrauchs besteht im **ARP Spoofing**, d.h. der Angreifer sorgt dafür, daß

---

42 Siehe Kapitel 4.2

43 Vgl. Raepple, M. (1998), S. 64 ff.

44 Siehe Kapitel 5.1

45 Siehe zu weiteren Ausführungen: Raepple, M. (1998), S. 39, Oppliger, R. (1998), S. 41 und Schmidt, J. (12 / 1997)

46 Vgl. Raepple, M. (1998), S. 63

ihm eine IP-Adresse zugeordnet wird, die ihm gar nicht gehört. Diese Attacke nutzt die Tatsache aus, daß Rechner lokale Caches mit Hardwareadressen führen und daß es legitim ist, einem Rechner, auch ohne Anfrage, eine ARP-Antwort zuzusenden, in der einer bestimmten IP-Adresse eine bestimmte Hardwareadresse zugeordnet wird. Will also der Angreifer X, daß beispielsweise ein Server im gleichen Netzwerksegment die Daten, die er normalerweise an einen Rechner A schickt, stattdessen an X versendet, so sendet ihm X regelmäßig eine ARP-Antwort mit seiner eigenen Hardwareadresse bezüglich As IP-Adresse. Dadurch, daß der Server auf diese Weise immer einen Hardwareadresseintrag für As IP-Adresse im Cache hat, fragt er nie mittels ARP-Broadcast nach der Hardwareadresse des A, sondern verschickt alle Daten direkt an X, der sich natürlich bei Anfragen auch immer mit As IP-Adresse meldet. Die einzige Möglichkeit, über die ein Administrator diesen Angriff erkennen könnte, wäre ein Vergleich der tatsächlichen Hardwareadresse mit der Adresse im ARP-Cache, was unpraktikabel ist. Es gibt jedoch eine Abwehrmöglichkeit: Betriebssysteme wie Unix oder OS/2 können mit statischen Hardwareadreßtabellen arbeiten, so daß nie ein Eintrag im ARP-Cache verändert wird. Dies bedeutet jedoch, daß bei Änderungen im Netz auf jedem Rechner die Datei mit den statischen Hardwareadressen geändert werden muß. Sollte nun ein Administrator auf die Idee kommen, statt dessen eine zentrale Tabelle auf einem Server einzurichten und die einzelnen Rechner diese über das Netz auslesen zu lassen, z.B. indem diese über ein Netzwerkdateisystem wie beispielsweise Suns NFS auf die Datei zugreifen, so muß der Administrator leider feststellen, daß NFS über UDP arbeitet und dessen Unsicherheit ist schon in Kapitel 4.2 beschrieben worden.[47]

4.5 Secure Socket Layer[48]

Die Secure Socket Layer (SSL) ist ein von dem Unternehmen Netscape Communications entwickeltes kryptographisches Protokoll. SSL dient dem sicheren Verbindungsaufbau mit Authentifizierung des Servers und optional auch des Clients, der Vereinbarung eines Sitzungsschlüssels und der anschließenden verschlüsselten Übermittlung von Daten. Dabei ist das Protokoll prinzipiell applikationsunabhängig, denn es setzt als Protokoll der Darstellungsschicht auf dem TCP/IP-Protokoll auf und kann von beliebigen Anwendungen verwendet werden. Am häufigsten wird SSL in Verbindung mit dem WWW verwendet, um dort eine sichere Übertragung von vertraulichen Daten, wie beispielsweise Kreditkarteninformationen, zu garantieren. Das System verwendet zur vorgeschriebenen Server- und optionalen Clientauthentifizierung ein asymmetrisches Verschlüsselungsverfahren in Verbindung mit der Zertifizierung der öffentlichen Schlüssel der Kommunikationspartner durch eine vertrauenswürdige Zertifizierungsorganisation. Die marktüblichen WWW-Browser werden mit einer Liste von öffentlichen Schlüsseln von solchen Zertifizierungsinstanzen ausgeliefert. Vereinfacht dargestellt, wird beim SSL-Verbindungsaufbau dem Client vom Server dessen zertifizierter, öffentlicher Schlüssel übertragen. Der Client überprüft mit seinem „eingebauten“ öffentlichen Schlüssel der Zertifizierungsorganisation die Authentizität des öffentlichen Schlüssels des Servers. Optional kann sich daraufhin auch der Client beim Server authentifizieren. Danach berechnet der Client einen Sitzungsschlüssel, der der Verwendung eines symmetrischen Verschlüsselungsverfahrens dient, kodiert diesen mit dem öffentlichen Schlüssel des Servers und überträgt diesen nun, für Dritte unverständlich, an den Server. Der Server kann nun den Sitzungsschlüssel mit seinem privaten Schlüssel dechiffrieren und somit haben nun Server und Client einen Schlüssel, mit dem sie den Datentransfer untereinander chiffrieren können und den kein Dritter kennen kann. Die einzelnen SSL-Datenpakete wiederum sind noch einmal über einen Hashwert gegen Veränderung geschützt.

---

[47] Vgl. Schmidt, J. (12 / 1997)

[48] Siehe zu weiteren Ausführungen: Shostack, A. (1995); Hickman, K. E. B. (1995); RSA FAQ (1998); Raepple, M. (1998), S. 133 ff.

SSL unterstützt die Verwendung verschiedener symmetrischer, kryptographischer Verfahren (MD2, MD5, RC2-CBC, RC4, DES-CBC, DES-EDE3-CBC) mit unterschiedlichen Schlüssellängen. Das tatsächlich zwischen Client und Server verwendete Verfahren wird zwischen den beiden Systemen ausgehandelt. Als asymmetrisches Verfahren kommt RSA zum Einsatz.

SSL gilt gemeinhin als prinzipiell sehr sicheres Verfahren, da bei Verwendung entsprechender großer Schlüssellängen (1024 Bit asymmetrisch, 128 Bit symmetrisch) auch ein Angriff mit hoher Rechenkapazität keine große Aussicht auf kurzzeitigen Erfolg hat. Auch sind anfängliche Schwachstellen in der Berechnung des Sitzungsschlüssels inzwischen beseitigt. Gerade die hohe Sicherheit von SSL und der Einsatz starker kryptographischer Verfahren haben jedoch dazu geführt, daß SSL dem amerikanischen Kriegswaffen-Export-Beschränkungen unterliegt. Dies führt dazu, daß die gängigen WWW-Browser von Netscape und Microsoft außerhalb der USA nur mit eingeschränkten Schlüssellängen der kryptographischen Verfahren erhältlich sind. Dies bedeutet den Einsatz eines maximal 512 Bit langen RSA-Schlüssels und eines 40 Bit langen symmetrischen Schlüssels. Diese Schlüssellängen gelten heute als viel zu gering um selbst nichtstaatlichen Angriffen lange standzuhalten. Darum verwundert es auch kaum, daß schon 1995 die außerhalb der USA nutzbare SSL-Implementierung der Browser von den Mitgliedern einer Mailingliste in nur 31 Stunden entschlüsselt wurde.[49]

[49] Anmerkung: Für den WWW-Browser von Netscape ist eine frei verfügbare Software erhältlich, die die Exportversion des Browsers bezüglich der kryptographischen Fähigkeiten auf die, in den USA verfügbaren Schlüssellänge modifiziert. Siehe dazu: http://www.fortify.org

5 Schutz offener Systeme: Firewalls[50]

Firewalls (deutsch: Brandschutzmauern) dienen der Abschottung eines Netzes gegen ein anderes, also beispielsweise eines Unternehmensnetzes gegen das Internet. Die Firewall soll verhindern, daß Daten aus dem einen Netz in das andere Netz gelangen, wenn dies nicht gewünscht ist. Somit sollen nicht für die „Außenwelt“ gedachte Daten nicht die Firewall überwinden können und Daten aus dem Außennetz sollen nicht in das Unternehmensnetz gelangen, solange dies nicht notwendig ist. Dabei wird durch technische und administrative Maßnahmen erreicht, daß jede Kommunikation zwischen den beiden Netzen über die Firewall geführt werden muß. Eine Schutzwirkung wird dabei nur in dem Maße erreicht, in dem keine alternativen Verbindungswege ins Internet existieren. Diese sollten durch das Sicherheitskonzept ausgeschlossen werden. Ein Modem hinter der Firewall kann beispielsweise die gesamte Schutzwirkung der Firewall unterlaufen.[51]
Bei den Firewalls wird zwischen zwei Typen unterschieden. Der eine wird als „Paketfilter“ bezeichnet, der andere als „Proxy Gateway“. Das Hauptunterscheidungsmerkmal ist dabei die Protokollebene, auf der die Schutzmaßnahmen ansetzen. I.d.R. gilt, daß der Schutz umso effektiver ist, je höher er in der Netzwerkschicht angesiedelt ist. In der Praxis kommt häufig ein Konzept zum Einsatz, daß beide Typen von Firewalls verwendet.

5.1 Paketfilter

Paketfilter, auch Packet Screens genannt, arbeiten auf der Vermittlungsschicht. Sie werten ein- und ausgehende Datenpakete ausschließlich aufgrund der Daten in den IP-, ICMP-, TCP- und UDP-Headern aus. Das bedeutet, daß die Informationen, auf deren Basis die Paketfilter über die Zulässigkeit des Transfers eines Datenpakets entscheiden, primär in der Sende- und Empfangsadresse, den verwendeten Portnummern[52], den TCP-Statusinformationen und dem Nachrichtentyp von ICMP-Paketen bestehen. Ein umfangreiches Regelwerk legt dann den Anforderungen entsprechend fest, welche Pakete in welcher Richtung passieren dürfen. Die Regeln haben dabei einen Inhalt der Form: erlaube / verbiete Zugriff von IP-Adresse a.b.c.d, Port Nummer x auf IP-Adresse e.f.g.h, Port Nummer y. Außerdem kann i.d.R. zwischen eintreffenden und ausgehende Paketen unterschieden werden. Das Problem dabei ist die Pflege des Regelwerks, denn dieses nimmt sehr schnell einen enormen Umfang an. Außerdem muß das Regelwerk mit dem Aufkommen neuer Dienst im Internet ständig angepaßt werden, wenn die Nutzung dieser Dienste gewünscht ist. Von den zwei, im allgemeinen genannten Optionen zur Gestaltung von Filterregeln

- erlaube alle Datentransfers, die nicht explizit verboten sind
- verbiete alle Datentransfers, die nicht explizit erlaubt sind

wird meist die zweite Variante favorisiert, da sie das versehentliche „Vergessen“ einer Sicherheitslükke ausschließt.

Dem Einsatz von Paketfiltern stellen sich allerdings einige Probleme entgegen. So gibt es beispielsweise bestimmte Dienste, wie das später in dieser Arbeit erwähnte „File Transfer Protocol“[53], die bei einer Datenübertragung in das Unternehmensnetz hinein, eine Verbindung aus dem Internet zum abrufenden Rechner aufbauen, was dazu führt, daß man Schwierigkeiten mit der Minimierung der Zu-

[50] Vgl. Raepple, M. (1998) S. 166 ff. und Ellermann, U. (1995)

[51] Für weitere Verfahren zum Schutz offener Systeme siehe Anhang, Seite 9-46

[52] Ports sind Unteradressen innerhalb eines Rechners, die es erlauben, mehrere unabhängige Kommunikationsverbindungen mit nur einem Rechner aufzubauen. Eine exakte Adresse besteht als aus IP-Adresse und Portnummer.

[53] Siehe Kapitel 6.5

griffsmöglichkeiten aus dem Internet in das Firmennetz hinein hat. Außerdem erzeugen Dienste, die das UDP-Protokoll verwenden, wie zum Beispiel viele Multimediaprotokolle, Probleme, da sich aus den Headern der UDP-Pakete nicht ergibt, wozu dieses Paket dient.
Paketfilter bieten einen einfachen und für den Anwender transparenten Kontrollmechanismus, der keine Rekonfiguration von bereits installierten Anwendungen erzwingt. Außerdem ist die Verwendung von Paketfiltern in der Regel sehr performant und kostengünstig, da diese Funktionalität heute bereits in den meisten Routern integriert ist. Um Probleme mit der Unterstützung von bestimmten Diensten zu verringern, gibt es inzwischen auch „intelligentere“ Paketfilter, die Kontextinformationen zu Kommunikationsverbindungen ermitteln und speichern.

Paketfilter lassen sich zwar relativ einfach einrichten, um unerlaubte Zugriffsversuche aus dem Internet abzufangen, jedoch ist ihre Schutzwirkung trotzdem nicht so hoch, wie bei Proxy Gateways, da zum einen die Inhalte der Datenpakete nicht überwacht werden und zum anderen die Struktur des Unternehmensnetzes nach außen sichtbar wird, da interne Rechner bei der Kommunikation als Kommunikationspartner sichtbar werden. Dadurch erhält ein externer Angreifer wesentlich mehr Angriffspunkte auf das Unternehmen, als notwendig. Dieses Problem läßt sich durch Proxy Gateways weitgehend beheben.

5.2 Proxy Gateways

Proxy Gateways sind dedizierte Rechner, über die alle Verbindungen zwischen dem Unternehmensnetz und dem Internet geleitet werden. Sie trennen dabei die Kommunikationsverbindungen zwischen dem Unternehmensnetz und dem Internet auf. Versucht ein Client aus dem Unternehmensnetz eine Verbindung zu einem Rechner im Internet aufzubauen, so überprüft das Proxy Gateway zuerst, ob eine Verbindung zum Zielrechner zum Zwecke der Nutzung des angeforderten Dienstes zulässig ist. Wenn dem so ist, dann sendet das Gateway die Daten an den Zielrechner, ersetzt jedoch den Absender durch seine eigene Adresse und einen eindeutig zugeordneten Port. Über eine Übersetzungstabelle kann so das Gateway eintreffende Datenpakete wieder dem zugehörigen Client zuordnen. Der Zielrechner sendet seine Antwort wieder an das Gateway, das die Daten nach möglichen weiteren Überprüfungen, an den Client weiterleitet.

Durch die Verwendung von Proxy Gateways bleibt der interne Aufbau des Unternehmensnetzes verborgen. Ein externer Angreifer hat dadurch nur das Gateway als Angriffspunkt und dieses kann man mit entsprechend hohem Aufwand vor Angriffen schützen. Die interne Netztopologie braucht dabei nicht geändert zu werden.

Proxy Gateways werden wiederum in zwei Klassen eingeteilt, die **Circuit Level Gateways** und die **Application Level Gateways**.

Circuit Level Gateways, auch als Verbindung von Packet Screen und Bastion bezeichnet, unterscheiden sich von reinen Paketfiltern primär dadurch, daß sie die direkte Verbindung zwischen dem Unternehmensnetz und dem Internet unterbrechen und somit nicht nur auf der Netzwerk-, sondern auch auf der Transportschicht arbeiten. Der Ausdruck „Bastion“ entsteht aufgrund der exponierten und besonders gesicherten Position des Circuit Level Gateways im Unternehmensnetz.

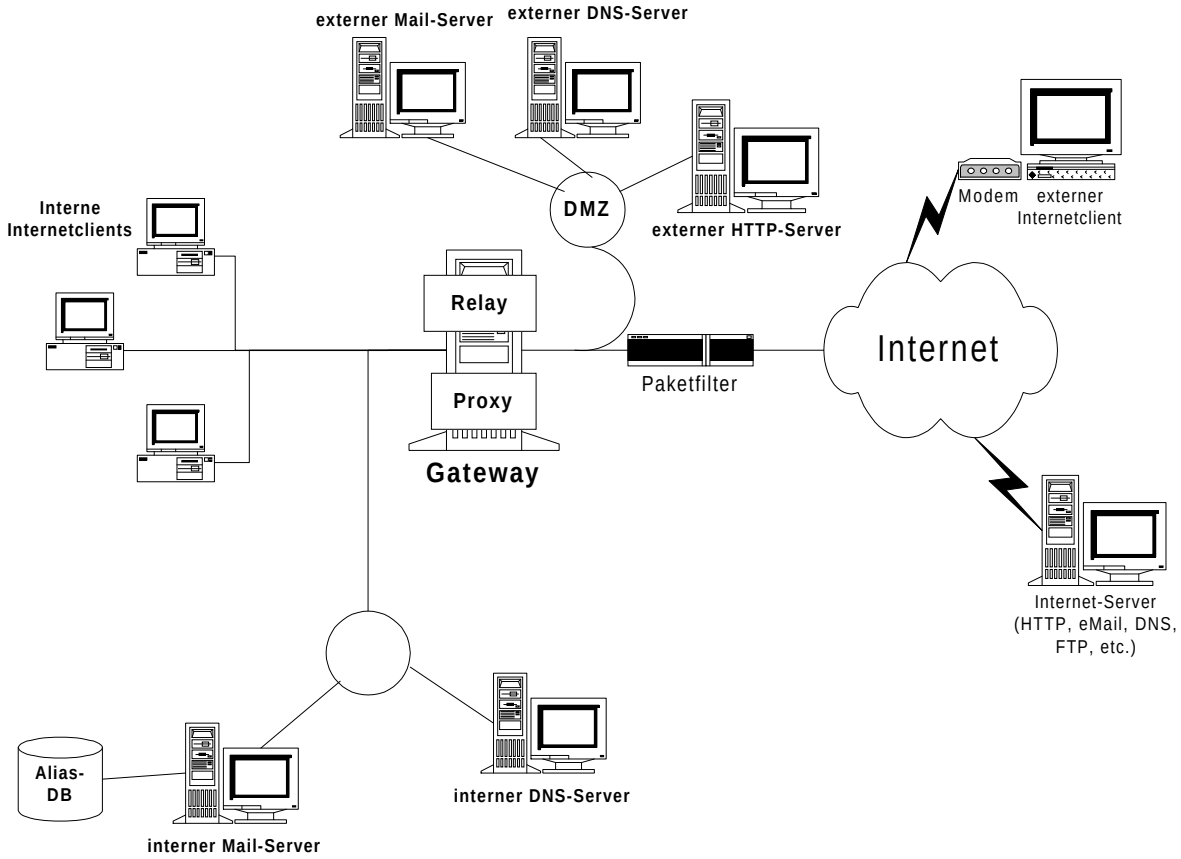
Weiter gehen hierbei die Application Level Gateways. Sie arbeiten, neben der Netzwerk- und Transportschicht, auch auf der Anwendungsschicht. Dadurch sind sie in der Lage auch anwendungsspezifische Informationen in den Filterprozeß einzubeziehen. Diese Gateways können anwendungsspezi-

fisch konfiguriert werden, d.h. man kann die Übertragung von Daten via FTP aus dem Unternehmensnetz ins Internet unterbinden und man kann bei der Übertragung von Programmen eine Virenüberprüfung erzwingen, um so den „Befall“ von Rechnern im Unternehmensnetz möglichst zu verhindern. Zusätzlich bieten Application Level Gateways häufig die Funktion eines lokalen Caches, der Informationen aus dem Internet, auf die häufig zugegriffen wird, puffert, so daß diese bei einem erneuten Zugriff direkt abgerufen werden können.

Das Problem bei Proxy Gateways ist allerdings, daß Anwendungen speziell für die Verwendung des Proxy Gateways konfiguriert werden müssen. Viele Standardanwendungen bieten heute eine solche Konfigurationsmöglichkeit, bei selbstentwickelten Anwendungen kann es allerdings notwendig sein, diese zu ändern und neu zu compilieren. Außerdem gibt es bei einem hohen Bandbreitenbedarf Performanceprobleme aufgrund der komplexen Prüfungen.

5.3 Praxisbeispiel[54]

An dieser Stelle soll kurz ein typisches Konzept für den Aufbau eines Unternehmensnetzes mit Anschluß an das Internet und Bereitstellung eines WWW-Servers für den Zugriff aus dem Internet vorgestellt werden. Dieses Konzept basiert auf dem kombinierten Einsatz eines Paketfilters und eines Proxy Gateways und der Schaffung einer sogenannten Demilitarisierten Zone (DMZ) zwischen Internet und unternehmensinternem Netzwerk.



Darstellung 11 - DMZ-Architektur[55]

[54] Vgl. Raepple, M. (1998), S. 232 ff.

[55] Aus: Raepple, M. (1998), S. 232

Der Paketfilter am äußeren Rand des Unternehmensnetzes läßt ausschließlich den Zugriff auf das Gateway und die verschiedenen externen Server zu. Alle anderen Zugriffsversuche, beispielsweise auf Rechner des internen Netzes, werden abgeblockt. Alle Zugriffe aus dem Unternehmensnetz auf das Internet werden vom Proxy Gateway unterbrochen und mit der eigenen Adresse versehen. eMail innerhalb des Unternehmens wird über den internen Mailserver abgewickelt und ist somit der Einsichtnahme eines unbefugten, externen Dritten entzogen. Nur eMail ins Internet wird an den externen Mail-Server geleitet, der diese dann versendet. Aus dem Internet eingehende eMail, wird vom externen Mail-Server empfangen und über eine spezielle Mail-Relay-Software, ein kurzes und robustes Programm, an den internen Server weitergeleitet. Dadurch wird vermieden, daß ein komplexer und fehleranfälliger eMail-Dienst zur Weiterleitung an den internen Mailserver verwendet werden muß. Der externe DNS-Server[56] gibt sich gegenüber dem Internet als primärer DNS-Server aus. Er besitzt jedoch nur minimale Informationen über den internen Aufbau des Unternehmensnetzwerks. Er kennt ausschließlich die Server in der DMZ. Informationen zu internen IP-Adressen kennt nur der interne DNS-Server. Dieser wird sowohl von den Rechnern in Unternehmensnetz, als auch vom Proxy Gateway zur Ermittlung von internen Rechneradressen verwendet. Wird aus dem inneren Unternehmensnetz eine Anfrage bzgl. eines Rechners im Internet gestellt, so leitet der interne DNS-Server die Anfrage an den externen DNS-Server weiter, der dann, sobald er eine Antwort auf die Anfrage ermittelt hat, diese Antwort an den internen DNS-Server weiterleitet. Durch diese Konzeption sind interne DNS-Informationen für einen externen Angreifer nicht verfügbar.

[56] Zu DNS siehe Kapitel 6.1

6 Dienste und ihre Sicherheitsprobleme

6.1 DNS

Da IP-Nummern nur schwer einzuprägen sind, wurde der sogenannte Domain Name Service (DNS) eingeführt, mit dessen Hilfe es möglich ist, alphanumerische Bezeichnungen für Rechner, z.B. www.firma.de, wieder in IP-Adressen zu übersetzen. Die Bezeichnung vor dem ersten Punkt ist dabei der Rechnername, alles nach dem ersten Punkt wird als Domainname bezeichnet. Dabei sind aber einem bestimmten DNS-Server nur ein kleiner Teil der Rechnernamen und der dazugehörigen IP-Adressen bekannt: das System ist hierarchisch aufgebaut. Wenn ein Client nun eine Anfrage an einen DNS-Server stellt, muß dieser, wenn er die Anfrage nicht selbst beantworten kann, wiederum einen DNS-Server einer höheren Verwaltungsebene hinzuziehen. Spätestens der Server auf der höchsten Verwaltungsebene kennt den DNS-Server, der für die entsprechende Domain verantwortlich ist. Der Nachrichtenaustausch erfolgt dabei aus Performancegründen auf Basis von UDP. Daher verfügt die DNS-Software über Mechanismen zur Wiederholung von Anfragen; unerwartet eintreffende Datenpakete werden ignoriert.

Ein Angriff auf einen DNS-Server verfolgt meistens das Ziel, daß dieser eine URL in eine falsche IP-Adresse auflöst (**DNS Spoofing**). Diese falsche IP-Adresse kann z.B. einem Rechner des Angreifers gehören, mit dem dann den arglosen Benutzern vorgetäuscht werden kann, daß sie in Kontakt mit dem von ihnen gewünschten System stehen. Der mögliche potentielle Schaden kann dabei sehr groß sein: Benutzer könnten z.B. auf dem System ein Update für ihr Betriebssystem erwarten, statt dessen steht dort ein, ihre Systemsicherheit gefährdendes Programm zum Download bereit. Generell läßt sich auf diese Art einem Benutzer leicht eine falsche Information unterschieben, mit allen daraus folgenden Konsequenzen, ohne daß dieser etwas davon merkt.

Eine Angriffsmethode ist die sog. **Cache-Pollution**. Aus Gründen der Effizienz speichert ein DNS-Server eine gewisse Anzahl von URLs und die dazugehörigen IP-Adressen in einem Cache, um bei einer erneuten Anfrage bezüglich dieser URL die Antwort geben zu können, ohne einen weiteren Server kontaktieren zu müssen. Ferner verfügt das Protokoll über eine Möglichkeit, neben der angeforderten Information noch weitere Informationen in die Antwort zu integrieren, um so möglichen zukünftigen Anfragen zuvor zu kommen. Ein Angreifer schließt nun seinen eigenen Webserver und seinen eigenen DNS-Server ans Internet an. Wenn sein DNS-Server zur Auflösung der URL seines Webservers kontaktiert wird, sendet der DNS-Server noch zusätzliche Informationen, die nicht mit der eigentlichen Anfrage in Zusammenhang stehen, nämlich falsche Zuordnungen von URLs, die auch aus anderen Domains stammen können, zu IP-Adressen. Der DNS-Server, der mit demjenigen des Angreifers in Kontakt getreten ist, speichert nun diese falschen Informationen in seinem Cache und löst somit bei einer Anfrage die betreffende URL falsch auf. In neueren Versionen ignoriert die DNS-Serversoftware unverlangte Informationen, wenn der Antwortende nicht für diese zuständig ist.

Eine weitere, sehr gefährliche Attacke ist das Erraten der Query-ID. Die Query-ID wird verwendet, um mehrere parallel laufende Anfragen unterscheiden zu können und zum Schutz vor gefälschten Antworten. Für letzeres ist sie jedoch kaum geeignet. Die Query-ID wird von dem anfragenden System frei gewählt, meistens werden der Einfachheit halber sequentielle Reihenfolgen verwendet. Der Angreifer muß für die hier beschriebene Attacke im Besitz einer eigenen Domäne sein, deren Namen er verwaltet. Zunächst erfragt der Angreifer vom anzugreifenden Server die IP-Adresse seines eigenen Rechnernamens. Dazu nimmt der anzugreifende Server Kontakt mit dem DNS-Server des Angreifers auf. So erfährt der Angreifer die aktuell verwendete Query-ID. Sofort stellt der Angreifer eine Frage an den anzugreifenden Server nach der IP-Adresse zu dem Rechnernamen, dessen IP-Adresse er fäl-

schen möchte. Er weiß, daß der anzugreifende Server dazu Kontakt mit dem Server aufnehmen muß, der für die Domäne dieses Rechners zuständig ist. Der Angreifer kennt ebenfalls den zuständigen DNS-Server. Sofort nach der Anfrage sendet der Angreifer eine vorgetäuschte Nachricht des zuständigen Servers. Dazu muß er die Query-ID erraten haben, die der angegriffene Server bei seiner Anfrage verwendet hat. Außerdem müssen seine Datenpakete als Absender die IP-Adresse des zuständigen Servers enthalten[57]. Selbstverständlich enthält die vorgetäuschte Antwort nicht die korrekte Zuordnung, sondern die vom Angreifer für seine Zwecke gefälschte. Wenn der angegriffene DNS-Server die Datenpakete des Angreifers vor denen des zuständigen Servers erhält, ist der Angriff gelungen, da der angegriffene Server diese Zuordnung nun im Cache hält und später ankommende Datenpakete zu dieser Anfrage verwirft. Der Angreifer kann zusätzlich zu dem eigentlichen Angriff auch noch eine SYN-Flooding-Attacke[58] gegen den zuständigen Server durchführen, damit dessen Antwort ihm nicht zuvorkommt. Der Angreifer sieht an der Antwort des angegriffenen Servers auf seine Anfrage sofort, ob seine Attacke Erfolg hatte oder ob der angegriffene Server die Antwort vielleicht aus seinem Cache beantwortet hat. Ist dies der Fall, so kann er dem TTL-Feld (time to live) in der Antwort auch die Lebensdauer der Daten im Cache entnehmen und weiß dann auf die Sekunde genau, wann es Sinn macht, den nächsten Angriff zu starten. Dieser Angriff ist deshalb sehr gefährlich, da er schwierig abzuwehren ist. Die Verwendung von sequentiellen Query-IDs und das zustandslose UDP sind hierbei die Hauptgründe. Während für IP Spoofing im Zusammenhang mit UDP keine Lösungsmöglichkeit besteht, könnte eine zufällige Vergabe der Query-IDs Abhilfe schaffen. Die Generierung echter Zufallszahlen in großen Mengen ist jedoch auf Rechnern problematisch, außerdem ist die Query-ID eine Zahl mit einer Länge von nur 16 Bit. Die daraus resultierenden 65536 Möglichkeiten lassen sich also unter Zuhilfenahme einer Denial-of-Service Attacke auf den zuständigen Server schnell ausprobieren. Bei einer angenommenen Länge der Antwortpakete von 100 Byte und einer Verbindung zwischen Angreifer und Opfer mit einer Bandbreite von 1 Mbit/s sind dafür etwa 60 Sekunden, bei einer ISDN-Verbindung etwa 20 Minuten notwendig.[59]

Sicherheitslücken in der DNS-Server-Software werden meist sehr rasch behoben, sobald sie bekannt geworden sind und eine Lösung gefunden wurde, so daß es sinnvoll ist, stets die aktuelle Version zu verwenden. Als DNS-Server kommt heute meistens der frei erhältliche Berkeley Internet Name Daemon (BIND) zum Einsatz.

6.2 Remote Access

Viele Teilnehmer des Internets stellen die Verbindung zu diesem mittels einer Wählverbindung über das öffentliche Telefonnetz zu einem Internet Service Provider (ISP) oder als Telearbeiter direkt zu ihrem Betrieb her. Dieser Vorgang wird auch als Remote Access bezeichnet. Die Wählverbindung wird durch ein Modem oder einen ISDN-Adapter auf der Seite des Benutzers und durch einen Remote Access Server (RAS) auf der Seite des ISPs hergestellt. Für diese Verbindung wird in der Regel das Point-to-Point Protocol (PPP) eingesetzt, das Serial Line Internet Protocol (SLIP) hingegen verliert bedingt durch seine Einschränkungen immer mehr an Bedeutung. Nach dem Herstellen einer PPP-Verbindung zwischen Client-Rechner und RAS beginnt die Authentifizierung des Clients, wozu verschiedene Protokolle zur Verfügung stehen: das Password Authentication Protocol (PAP), das Challenge Handshake Authentication Protocol (CHAP) und der Remote Authentication Dial In User Service (RADIUS).

Bei PAP überträgt der Benutzer seine Kennung und sein Paßwort im Klartext an den RAS, der die Angaben mit den bei ihm gespeicherten Daten vergleicht. Ist die Anfrage korrekt, bekommt der Cli-

[57] Sog. IP Spoofing; siehe auch Kapitel 4.2

[58] Siehe Kapitel 4.2

[59] Vgl. V. Mraz, K. Weidner (1997)

ent-Rechner eine IP-Adresse zugewiesen und erhält Zugriff auf das Netzwerk, an das der RAS angeschlossen ist. Alternativ können die Paßwörter auch als sichere Hashwerte auf dem RAS gespeichert sein, und der Benutzer schickt dann den Hashwert und nicht das Paßwort.

Bei CHAP hingegen generiert der RAS einen zufälligen Wert, die sogenannte Challenge und einen Nachrichtenindikator, die er beide an den Client sendet. Der Client berechnet aus diesen Werten und seinem Paßwort einen neuen Wert und sendet diesen an den RAS zurück. Da der RAS das Paßwort kennt, kann er ebenfalls den Wert mit dem Paßwort ausrechnen und vergleichen. Wichtig bei diesem Verfahren ist, daß die Challenge ein unvorhersehbarer Wert ist.

Diese beiden Protokolle eignen sich aber lediglich für kleinere Installationen mit verhältnismäßig wenigen Benutzern. Müssen etwa mehrere RAS eingesetzt werden, ist es sinnvoller, die Informationen zur Authentifizierung an wenigen zentralen Orten zu halten, um eine leichtere Administrierbarkeit zu ermöglichen. Dazu wurde von der Firma Livingston das RADIUS Protokoll entwickelt. Die Benutzerdaten werden dazu auf einem RADIUS-Server gehalten mit dem der RAS kommuniziert. Die Kommunikation kann mit einem symmetrischen Verfahren verschlüsselt werden. Dazu wird der Schlüssel sowohl in der Konfiguration des RAS als auch in der des RADIUS-Servers eingetragen. RADIUS kann, dank seiner offenen Architektur, mit verschiedenen Authentifizierungsmechanismen, von der einfachen Angabe eines wiederverwendbaren Paßworts, bis hin zum komplizierten Challenge/Respose-Verfahren, zusammenarbeiten. Stellt nun ein Client eine Verbindung mit dem RAS her, fragt dieser z.B. die Benutzerkennung und das Paßwort ab. Die erhaltenen Daten werden über die sichere Verbindung zum RADIUS-Server geschickt, der diese mit den vorliegenden Datensätzen vergleicht oder an einen weiteren Authentifizierungsmechanismus weiterleitet, so z.B. an die Paßwortüberprüfung eines UNIX-Systems. Überprüft der RADIUS-Server die Paßwörter selbst, ist unbedingt darauf zu achten, daß die Datei, die diese enthält, nur vom Systemadministrator les- und schreibbar ist, da die Paßwörter dort im Klartext gespeichert sind. Abgesehen von der Authentifizierung schreibt RADIUS für den RAS auch Logdateien mit sogenannten Start- und Stop-Records, in denen etwa der Zeitpunkt der Einwahl bzw. des Endes der Verbindung, die Dauer des Authentifizierungsvorgangs, die zugewiesene IP-Adresse und die übertragene Datenmenge festgehalten werden. Für ISPs sind diese Daten oft Grundlage für die Rechnungsstellung.

Noch zu erwähnen sei das PPP Encryption Protocol (ECP), das eine Erweiterung von PPP zur Verschlüsselung einer PPP-Verbindung darstellt. Die Algorithmen und Schlüssellängen werden dabei ähnlich wie bei SSL ausgehandelt.

Das Verfahren, daß sich mit einem Phänomen der Branche, nämlich schnell steigender Benutzerzahlen, am besten verträgt, ist RADIUS, weshalb es auch von den meisten ISPs eingesetzt wird. Auf komplizierte Authentifizierungsmechanismen wird dabei meist verzichtet; Benutzername und Paßwort werden dann im Klartext über die Telefonleitung geschickt. Die Sicherheit wird dabei als ausreichend betrachtet, da es im Vergleich zum Abhören einer Internetverbindung mit erheblichem Aufwand verbunden ist, gleiches bei einer Telefonverbindung durchzuführen. Sicherheitsmechanismen für sensible Daten (abgesehen von den Zugangsdaten für den Remote Access) sollten deshalb auf einer höheren Ebene als PPP implementiert werden, da diese auch hinter dem RAS, nämlich im Internet bestehen müssen.[60]

6.3 Terminalbetrieb

In der Praxis ist es oft der Fall, daß wichtige Systeme in einem speziellen Raum aufgestellt sind, der über eine Alarmanlage, unterbrechungsfreie Stromversorgung, Klimaanlage etc. verfügt, während die Administration von einem, sich eventuell in einem anderen Raum befindlichen Rechner aus durchge-

[60] Siehe zu weiteren Ausführungen: M. Raepple (1998), S. 178 - 186

führt wird. Dazu bedient sich der Administrator Programmen wie Telnet oder Rlogin: sie stellen eine Schnittstelle zu entfernten Systemen dar und erlauben die Eingabe von Befehlen sowie die Ausgabe auf dem lokalen Rechner. Dabei werden alle Daten üblicherweise als Klartext übertragen. Gewisse Programme können dies dazu verwenden, um alle übertragenen Daten, z.B. innerhalb eines Ethernetsegments, nach Paßwörtern zu durchsuchen und diese abzuspeichern. Außerdem erlauben diese Programmen die ”Entführung” einer Telnet-Sitzung. Dazu klinkt es sich in den Paketstrom zwischen Client und Server: die Absenderadresse wird gefälscht (IP Spoofing) und die TCP-Pakete werden mit einer korrekten Sequenznummer versehen. Die Datenpakete des ”echten” Clients haben dann keine Chance mehr, da Pakete mit gleicher Sequenznummern nur einmal angenommen werden und somit die Sequenznummern aus Sicht des Servers ungültig sind. Der Angreifer hat also die Telnet-Sitzung entführt und kann nun mit den Rechten des früheren Besitzers eventuell großen Schaden anrichten.[61]

Ein Protokoll, daß dies auf der Ebene der Transportschicht unterbindet, ist Secure Shell (SSH), das an der Universität Helsinki in Finnland entwickelt worden ist. Es unterstützt sowohl die Authentifizierung von Rechnern als auch von Benutzern. Da obengenannter Angriff nach der Authentifizierung stattfindet, wäre dies allein nicht ausreichend. SSH verschlüsselt auch die gesamte Sitzung und stellt somit sicher, daß ein Dritter diese nicht übernehmen kann. Das Verfahren soll hier kurz dargestellt werden: Der Client stellt eine Authentifizierungsanfrage an den Server, dieser antwortet mit seinem öffentlichen Host-Key, der die Verbindung an den Server bindet, und seinem öffentlichen Server-Key, der sich i.d.R. stündlich ändert und der somit die unbefugte Entschlüsselung sehr erschwert, selbst wenn der Host-Key gebrochen worden ist. Der Client vergleicht den Host-Key mit dem bei ihm gespeicherten Wert, der manuell auf dem Client hinterlegt worden sein sollte, um eine Manipulation zu erschweren. Fällt der Vergleich positiv aus, generiert er eine 256 Bit lange Zahl, die als Session-Key dient und wählt eine vom Server unterstützte symmetrische Verschlüsselungsmethode aus. Der Session-Key wird dann zunächst mit dem öffentlichen Host-Key, dann mit dem öffentlichen Server-Key verschlüsselt und an den Server geschickt. Dieser entschlüsselt die Nachricht und erhält so den geheimen Session-Key. Als Verschlüsselungsverfahren für die Authentifizierung und den Austausch des Session-Keys wird gewöhnlich RSA verwendet. Dieser wird nun für die Verschlüsselung aller übertragenen Daten verwendet, zum ersten Mal für die Bestätigung an den Client, daß nun eine verschlüsselte Verbindung besteht. Dabei ist aber bisher lediglich eine sichere Verbindung zwischen zwei Rechnern hergestellt worden, eine Authentifizierung des Benutzers hat nicht stattgefunden. Diese kann jedoch nun durch eine Übertragung des Paßworts über die sichere Verbindung problemlos erfolgen. Der Vollständigkeit halber sei erwähnt, daß SSH noch eine weitere Methode zur Benutzerauthentifizierung vorsieht, auf die hier nicht weiter eingegangen werden soll. Der größte Nachteil von SSH ist, daß es auf manuell verteilten Schlüsseln beruht; die Unterstützung einer Public-Key Infrastruktur ist aber generell nicht ausgeschlossen. SSH ist für UNIX-Plattformen frei verfügbar.[62]
An dieser Stelle soll noch ein weiteres Verfahren vorgestellt werden, nämlich Secure Telnet (STEL). STEL verwendet das Diffie-Hellman Verfahren in der Variante, die resistent gegen eine Man-in-the-Middle Attacke ist, zum Austausch eines geheimen Schlüssels. Von diesem Schlüssel wird als aktueller Session-Key ein Hashwert mittels MD5 errechnet, mit dem dann die gesamte Sitzung verschlüsselt wird. Anschließend kann dann eine Authentifizierung des Benutzers mir einer gewöhnlichen Paßwortabfrage erfolgen, da diese nun über eine sichere Verbindung stattfindet. Es werden aber noch weitere, hier nicht weiter ausgeführte Authentifizierungsmechanismen angeboten.[63]

[61] Siehe zu weiteren Ausführungen: J. Schmidt (10 / 1997) und Kapitel 4.2
[62] Siehe zu weiteren Ausführungen: R. Oppliger (1997), S. 215 - 226
[63] Siehe zu weiteren Ausführungen: R. Oppliger (1997), S. 253 - 255

Weitere Verfahren, auf die an dieser Stelle nicht näher eingegangen wird, sind die Benutzung von SSL oder der hier nicht vorgestellten Transport Layer Security (TLS) zum Aufbau einer sicheren Telnet-Verbindung, das Produkt S/RLogin von der Firma Baltimore Technologies Ltd., sowie ein frei verfügbarer Telnet-Ersatz für 4.4BSD UNIX, der von Matt Balze und Steven Bellovin bei den AT&T Bell Laboratories entwickelt worden ist und Secure RPC Authentication (SRA) von David Safford, David Hess und Douglas Lee Schales von der Texas A&M University. Zum Teil wird von diesen Verfahren aber nur eine sichere Authentifizierung unterstützt, so daß Schutz gegen eine Entführung der Session oder Mitlesen nicht immer gegeben ist.[64]

6.4 eMail

Für das Versenden und Empfangen von eMail wird das Simple Mail Transfer Protocol (SMTP) verwendet. Im UNIX-Umfeld wird dafür meistens das Programm sendmail benutzt. Die Konfiguration von sendmail gestaltet sich im allgemeinen relativ umständlich, so daß Konfigurationsfehler an der Tagesordnung sind. Außerdem werden immer wieder Fehler in der Implementierung von sendmail festgestellt, so daß es dringend angeraten ist, möglichst mit der jeweils aktuellen Version zu arbeiten. So wurde z.B. darüber berichtet, daß sich Unbefugte Zugang unter dem Namen desjenigen Users verschaffen konnten, unter dessen Benutzerkennung der sendmail-Prozeß gestartet wurde. Schlimmer noch: es wurde berichtet, daß Unbefugte Zugang unter der Systemverwalterkennung root erhielten. sendmail ist frei erhältlich und die auftretenden Probleme werden meist schnell behoben.[65]
Neben diesen Angriffsmöglichkeiten bestehen aber noch einige weitere: So kann man z.B. durch senden großer eMails das Zielsystem durch Erschöpfung der Ressourcen überlasten und damit unbrauchbar machen (Mail-Bombing, ein Denial-of-Service-Angriff). Eine weitere Möglichkeit ist das Senden einer Mail unter falschem Absender und damit das Vortäuschen einer falschen Identität. Ist die fälschlicherweise angegebene Absenderadresse einem bestimmten Benutzer zugeordnet, wird dieser zudem sämtliche Antworten anstelle des tatsächlichen Absenders erhalten.
Auch das Abrufen der in der Mailbox befindlichen eMails mit dem Post Office Protocol (POP) beinhaltet Risiken. Der Benutzer muß sich gegenüber dem POP-Server authentifizieren, so daß niemand unbefugt seine Mails einsehen, weiterleiten oder löschen kann. In der Regel geschieht dies durch die Übertragung der Benutzerkennung und des Paßwortes im Klartext, womit es einem Angreifer relativ leicht ermöglicht wird, diese auszuspionieren. Hinzu kommt, daß das für POP verwendete Benutzerkennung/Paßwort-Paar bei vielen ISPs auch für die Authentifizierung mit dem RAS verwendet wird, darüber hinaus besteht auch manchmal noch ein Telnet-Zugang zu einem Rechner des ISPs, an dem sich der Benutzer ebenfalls mit dem gleichen Paar authentifiziert. Wie man also sehen kann, besteht die Möglichkeit, daß durch die Übermittlung des Benutzerkennung/Paßwort-Paars im Klartext, die Sicherheitsmechanismen von z.B. RADIUS oder SSH ganz oder teilweise kompromittiert werden können. Abhilfe schafft hier die Verwendung von SSL zur Verschlüsselung der POP-Session[66], dies wird jedoch von vielen POP-Clients nicht unterstützt und auch von den ISPs selten angeboten, so daß stets Vorsicht geboten ist. Eine regelmäßige Änderung des Paßwortes ist also unbedingt angeraten, wenn dies dem Benutzer möglich ist.
”Die Übertragung einer E-Mail im Internet ist vergleichbar mit dem Versand von Postkarten.”[67] Der Unterschied dabei ist jedoch, daß sich der Inhalt von eMails im Gegensatz zu dem von Postkarten leicht elektronisch nach Schlüsselwörtern durchsuchen läßt und es somit einem Angreifer ermöglicht wird, für ihn interessante Nachrichten aus der Vielzahl der eMails herauszufiltern. Außerdem wird die

[64] Siehe zu weiteren Ausführungen: R. Oppliger (1997), S. 249 - 253
[65] Siehe zu weiteren Ausführungen: CERT Advisories CA-96.20, CA-96.24 und CA-97.05
[66] Zu SSL siehe Kapitel 4.5
[67] M. Raepple (1998), S. 192

eMail von dem meisten Benutzern eher mit dem Brief verglichen, so daß sich die wenigsten der Tatsache bewußt sind, daß die versendete Nachricht im Klartext sowohl zum Zielsystem übertragen als auch dort gespeichert wird. Abgesehen davon, daß der Systemadministrator des Zielsystems technisch berechtigt ist, sämtliche Dateien und somit auch die dort gespeicherten Mails zu lesen, zu verändern und zu löschen, kann die Nachricht auch auf ihrem Weg zum Zielsystem abgefangen und gelesen, verändert oder gelöscht werden. In Anbetracht dieser Tatsache muß man zu der Erkenntnis gelangen, daß viele Firmen, die heute arglos vertrauliche Dokumente und Nachrichten per eMail verschicken, ein großes Sicherheitsrisiko eingehen. Zum einen ist es dringend geboten, daß die Mitarbeiter in den Unternehmen für das Problem sensibilisiert werden, zum anderen müssen Mittel zur Verfügung gestellt werden, die den sicheren Versand einer Nachricht ermöglichen. Diese stehen heute zur Verfügung und erlauben ein Sicherheitsniveau, das dem der Zustellung von Nachrichten durch die Post oder einem Boten bzgl. der Vertraulichkeit und Unverfälschbarkeit weit überlegen ist.

Ein Verfahren zur Verschlüsselung von eMails ist Pretty Good Privacy (PGP). PGP verwendet den Ansatz der Public-Key Verfahren, wie er oben bereits erläutert ist. Als asymmetrischen Algorithmus verwendet PGP RSA mit einer Schlüssellänge von bis zu 2048 Bit. Mit diesem wird dann ein zufällig gewählter symmetrischer Session-Key verschlüsselt. Die Nachricht selbst wird mit dem Session-Key verschlüsselt. Als symmetrische Verfahren werden dabei wahlweise effektiv 112 Bit lange Schlüssel für Triple-DES oder 128 Bit lange Schlüssel für IDEA verwendet. Die langen Schlüssel stehen auch außerhalb der USA und Kanada zur Verfügung. Werden die maximal möglichen Schlüssellängen gewählt, so sind die Nachrichten nach heutiger Kenntnis in absehbarer Zeit nicht durch Unbefugte zu entschlüsseln. Neuere Versionen von PGP, bieten allerdings die Möglichkeit, zur sog. ”Company Message Recovery”, die es einer Firma ermöglicht, Daten zu entschlüsseln, für die der Schlüssel verlorengegangen ist. Das Verfahren ist sehr umstritten.[68]

PGP bietet auch an, Nachrichten mit einer digitalen Signatur zu versehen. Als Hash-Algorithmus wird dazu MD5 verwendet. Hashwerte werden dann unter Benutzung von RSA mit dem öffentlichen Schlüssel des Empfängers verschlüsselt.

Außerdem unterstützen neuere Versionen von PGP neue Verfahren zur Berechnung des Sitzungsschlüssels (Diffie-Hellman) und zur Erstellung von Signaturen (DSS). Diffie-Hellman wurde gewählt, da es frei von Patenten ist und dadurch einen kostengünstigeren kommerziellen Einsatz erlaubt. DSS wurde gewählt, da es erstens ein standardisiertes Verfahren ist und zweitens, weil es statt des MD5-Algorithmus SHA verwendet, der im Gegensatz zum MD5-Algorithmus keine bekannten Schwachstellen aufweist.

Auf Details zur Verteilung und Signierung der Schlüssel kann hier nur insofern eingegangen werden, daß die Benutzer die Schlüssel gegenseitig signieren und somit ein ”Web of Trust” entsteht. Öffentliche Zertifizierungsstellen sind nicht vorgesehen, das Computer Emergency Response Team des Deutschen Forschungsnetzes (DFN-CERT)[69] und der Verlag Heinz Heise GmbH & Co KG[70] haben jedoch Projekte zur zentralen Auskunft und Zertifizierung von öffentlichen Schlüsseln gestartet.

PGP ist für den nicht kommerziellen Einsatz kostenlos, und es steht eine graphische Benutzerschnittstelle zur Verfügung, über die das Programm leicht zu bedienen ist. Ferner existieren sogenannte Plugins für eMail-Clients, wie etwa Microsoft Outlook Express, mit denen durch Knopfdruck eine Mail verschlüsselt, entschlüsselt oder signiert werden kann.

Ein weiteres Verfahren ist die Secure Multi Purpose Mail Extension (S/MIME), die im Gegensatz zu PGP auf bereits bestehende Infrastrukturen bei den Zertifikaten setzt. ”Die Teilnahme am internationalen Briefverkehr mit S/MIME setzt ein beglaubigtes Schlüsselzertifikat von einer kostenpflichtigen

[68] Siehe auch Kapitel 6.8

[69] Siehe: http://www.cert.dfn.de/dfnpca/

[70] Siehe: http://www.heise.de/ct/pgpCA/

CA ... voraus"[71]. S/MIME selbst ist ebenfalls kostenpflichtig, und die verwendeten Schlüssellängen entsprechen den US Exportbeschränkungen, weshalb der Einsatz von S/MIME außerhalb der USA und Kanada gut überlegt sein sollte. Obwohl es z.B. in den eMail-Clients Outlook Express, Outlook 98 (beide von Microsoft) und dem Mail-Client des Netscape Communicators ab Version 4.0 bereits integriert ist, erfreut sich dieses Verfahren nicht der Beliebtheit von PGP.
Weitere Verfahren sind Privacy Enhanced Mail (PEM) sowie MIME Object Security Services (MOSS), die in der Praxis aber relativ irrelevant sind, weshalb an dieser Stelle nicht weiter darauf eingegangen wird.

6.5 FTP

Das File Transfer Protocol (FTP) dient dazu, Dateien zwischen zwei Rechnern zu übertragen, die über ein Netzwerk miteinander verbundenen sind. Dazu stellt der Client von einem nichtprivilegierten Port[72] zum Port 21 des Servers eine Verbindung her, über die er dem Server Befehle zur Steuerung des Dateitransfers übermittelt. Über diese, auch als Control Connection bezeichnete Verbindung, authentifiziert sich auch der Client gegenüber dem Server. Dies geschieht normalerweise durch die Übermittlung eines Benutzernamens und eines Paßworts im Klartext. Zur Datenübertragung öffnet der Server eine Verbindung von Port 20 zu einem nichtprivilegierten Port des Clients, die sogenannte Data Connection. Probleme ergeben sich, wenn der Client durch eine Firewall geschützt ist, da diese dann Verbindungsanfragen von beliebigen Rechnern zu sämtlichen Ports des Clients größer als 1023 zulassen müßte. Da dies oft nicht gewollt ist, bieten manche FTP-Server die Möglichkeit zum sog. passiven Datentransfer: dazu teilt der Client dem Server mittels des Befehls PASV mit, daß dieser einen TCP-Socket öffnen soll, der auf den Aufbau der Data Connection, diesmal aber auf Initiative des Clients, warten soll. Der Server teilt dem Client dann die Portnummer des Sockets über die Control Connection mit.[73]
Das Problem der Paßwortübertragung im Klartext läßt sich durch verschiedene Verfahren lösen: Zum einen besteht die Möglichkeit einer FTP-Implementierung, die SSL verwendet.[74] Die andere Möglichkeit ist die Benutzung der SSH, da diese ein sog. Port Forwarding ermöglicht, bei dem eine Verbindung durch eine mit SSH gesicherte Verbindung getunnelt wird.[75] Eine weitere Möglichkeit besteht darin, das Verhältnis zwischen Client und Server zu tauschen. Dafür startet der Benutzer auf dem Client einen FTP-Server, der nur eine Verbindung gleichzeitig zuläßt, und auf dem Server einen FTP-Client, verbindet den FTP-Client mit dem FTP-Server und startet die Übertragung. Ist diese erfolgt, beendet der Benutzer den FTP-Server. Ein Angreifer hätte also nie die Möglichkeit, sich mit diesem FTP-Server der Client-Seite zu verbinden. Es ist aber zu Bedenken, daß auf dem Client unter einer UNIX-Umgebung root-Rechte zum Starten des Serverprozesses notwendig sind. Außerdem ist trotzdem eine sichere (Telnet-)Verbindung zu dem entfernten Rechner erforderlich, da sonst das Paßwort für den entfernten Rechner beim Aufbau der Telnet-Verbindung abgehört werden kann.

6.6 HTTP

Das Hypertext Transfer Protocol (HTTP) findet, grob gesprochen, Verwendung zum Übertragen von HTML Seiten von einem HTTP-Server zu einem Webbrowser, aber auch von Steuerungsinformationen vom Browser zum Server. HTML bietet als Gestaltungsmöglichkeit sog. Forms an, in die der Be-

[71] M. Raepple (1998), S. 198

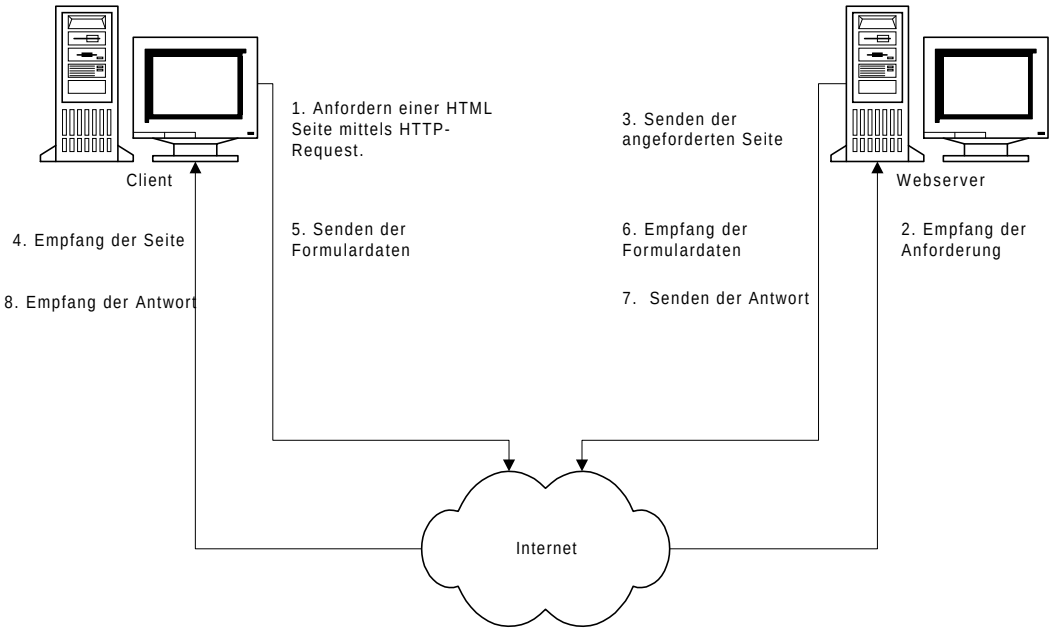
[72] Nichtprivilegierte Ports sind Ports mit einer Nummer größer als 1023. Ports mit kleinerer Nummer können unter UNIX nur Anwendungen benutzen, die vom Systemverwalter root gestartet werden.

[73] Vgl. M. Raepple (1998), S. 169f.

[74] Eine solche ist z.B. unter ftp://ftp.uni-mainz.de/pub/internet/security/ssl/SSLapps zu finden

[75] Siehe zu weiteren Ausführungen: T. Ylonen, T. Kivinen, M. Saarinen (1997a)

nutzer Daten eingeben kann, die dann an den Server übertragen werden. Auch hier verläuft die Kommunikation zwischen Browser und Server gewöhnlich unverschlüsselt. Dies wird dann problematisch, wenn der übertragene Inhalt vertrauliche Informationen enthält, wie z.B. die Nummer einer Kreditkarte beim Bestellen von Büchern im Internet.



Darstellung 12 - Ablauf eines HTTP-Requests und Verarbeitung von Formulardaten

Neben dem bereits oben vorgestellten SSL, das heute in solchen Fällen meistens zur Anwendung kommt und die Datenübertragung auf der Transportschicht absichert, existieren noch weitere Möglichkeiten, wie z.B. das Secure Hypertext Transfer Protocol (S-HTTP), das auf der Anwendungsschicht arbeitet. Es stellt eine Erweiterung des HTTP-Standards dar und ist mit diesem voll kompatibel. Das Senden einer Form an einen Server läuft mit S-HTTP dann wie folgt ab: Browser und Server gleichen ihre zur Verfügung stehenden kryptographischen Verfahren gegeneinander ab und einigen sich auf ein entsprechendes Verfahren. Daraufhin sendet der Server dem Browser sein öffentliches Schlüsselzertifikat. Der Browser verschlüsselt die zu sendenden Daten mit einem zufällig erzeugten Sitzungsschlüssel, verschlüsselt diesen mit dem öffentlichen Schlüssel des Servers und sendet die verschlüsselten Daten und den verschlüsselten Sitzungsschlüssel mit einer S-HTTP Nachricht an den Server. Dieser entschlüsselt den Sitzungsschlüssel mit seinem privaten Schlüssel und mit dem Sitzungsschlüssel die Daten. Die Antwort des Servers verschlüsselt dieser mit einem zufälligen Transaktionsschlüssel, der wiederum mit dem Sitzungsschlüssel verschlüsselt wird. Der Browser kennt den Sitzungsschlüssel und kann somit an den Transaktionsschlüssel gelangen, mit dem er wiederum die Nachricht des Servers entschlüsseln kann.

S-HTTP kann vorhandene Public-Key Infrastrukturen nutzen, benötigt diese aber nicht zwingend. Überhaupt ist das Protokoll sehr flexibel gehalten, um möglichst vielen Anforderungen zu genügen. So ist es auch möglich, Kerberos-Tickets[76] zur Authentifizierung zu verwenden oder übertragene Daten lediglich zu signieren, aber nicht zu verschlüsseln. URLs, die S-HTTP verwenden, beginnen mit ‚shttp://'.[77]

Zur Zeit dominiert SSL klar, wenn es darum geht, HTTP-Verbindungen zu sichern und das, obwohl S-HTTP der offizielle Standard der Internet Engineering Task Force (IETF) ist. Der Grund dafür mag auch in der komplizierten Konfiguration von S-HTTP im Vergleich zu SSL liegen.

[76] Siehe dazu Anhang 9.1

[77] Siehe zu weiteren Ausführungen: R. Oppliger (1997), S. 167 - 172

Bei der Einrichtung eines Webservers sollte auch bedacht werden, daß z.B. durch fehlerhafte CGI-Skripte[78] großer Schaden angerichtet werden kann. So könnte zum Beispiel der Angreifer in das Feld, in das er seine eMail-Adresse zwecks Anforderung von Produktinformationen einträgt, hinter die eMail-Adresse noch '; rm -rf /*' schreiben. Dadurch kann bei unvorsichtiger Programmierung neben dem Ausführen von sendmail auch noch das Löschen aller Dateien erreicht werden, die dem Benutzer gehören, unter dessen Username der Webserver läuft. Unter UNIX-Systemen bietet es sich deshalb an, den Server in einem Teilbaum des Dateisystems laufen zu lassen, von wo aus keine Zugriffsmöglichkeit auf andere Teilbäume besteht. Dies kann mit dem chroot-Kommando geschehen, daß für den aufrufenden Benutzer ein beliebiges Verzeichnis als Root-Verzeichnis definiert. Zugriffe auf Ebenen oberhalb des neuen Root-Verzeichnisses sind dann nicht mehr möglich. Gelingt es nun einem Angreifer, einen Webserver zu kompromittieren, hat dies keine Auswirkungen auf Dateien, die außerhalb des festgelegten Root-Verzeichnisses liegen.

6.7 Aktive Inhalte

Unter aktiven Inhalten sind hier in HTML-Seiten eingebettete Programme gemeint, wie z.B. Java Applets. Diese Inhalte sind dabei in einer höheren Programmiersprache geschrieben und können somit prinzipiell alle Operationen auf einem Rechner ausführen. Dies geht weit über die Möglichkeiten von HTML hinaus. Derjenige jedoch, der solche aktiven Inhalte auf seinen Rechner lädt und ausführt, kann aufgrund der anonymen Struktur des Internets nicht sicher sein, daß diese seinem System nicht schaden. So wäre es prinzipiell leicht möglich, ein Programm zu schreiben, daß die Festplatte des Benutzers formatiert. Daher verfügen die Konzepte für aktive Inhalte über unterschiedliche Sicherheitsmechanismen, die im folgenden vorgestellt werden.

6.7.1 Suns Java

Großer Beliebtheit erfreuen sich die sogenannten Java-Applets, bedingt durch ihre Plattformunabhängigkeit. Verfügt ein Browser über eine Java-Laufzeitumgebung, eine sog. Java Virtual Machine (JVM), kann er jedes Applet damit ausführen, unabhängig von der Umgebung, in der es entwickelt worden ist. Gewöhnliche Java-Applets sind jedoch durch das Java-Sicherheitskonzept stark beschränkt, was ihre Fähigkeiten anbelangt auf das System zuzugreifen, auf dem sie ausgeführt werden. Sie können unter anderem nicht vom lokalen Dateisystem lesen oder darauf schreiben, sie können Netzwerkverbindungen nur in einem allgemein festgelegten Rahmen initiieren, auch können sie keinen Ausdruck auf einem Drucker starten und auch nicht auf die Zwischenablage zugreifen.[79] Für manche Anwendungen scheinen diese Restriktionen zu eng zu sein. Deshalb ist es möglich, Java-Applets mit Signaturen zu versehen, die wiederum von einer öffentlichen Zertifizierungsstelle signiert sind. Lädt der Benutzer nun ein solches signiertes Applet auf seinen Rechner, wird der Browser ihn vor die Entscheidung stellen, ob er der Stelle, die das Applet signiert hat, traut. Ist dies der Fall, werden sämtliche Sicherheitsrestriktionen aufgehoben und das Applet erlangt vollen Zugriff auf alle Systemressourcen.
Für die meisten Applets sind aber Signaturen nicht erforderlich, so daß die Sicherheitsmechanismen von Java grundsätzlich einen vollständigen Schutz des Systems gewährleisten.

6.7.2 Microsofts ActiveX

ActiveX-Programme, die ausschließlich auf Windows-Umgebungen ausführbar sind, stellen grundsätzlich Applikationen mit vollen Rechten bzgl. des Zugriffs auf die Systemressourcen dar. Aus die-

[78] CGI-Skripte sind Programme, die der Webserver ausführt, um die Eingaben, die der Benutzers in HTML-Formularen eingetragen hat, zu verarbeiten.

[79] Siehe zu weiteren Ausführungen: D. Flanagan (1997)

sem Grund müssen sie mit einer Signatur des Ausstellers versehen sein, die von einer Zertifizierungsstelle zertifiziert worden ist. Akzeptiert der Benutzer die Signatur und erkennt damit den Austeller als vertrauenswürdig an, wird das ActiveX-Programm auf seinem Rechner ausgeführt.

6.7.3 Client-side Scripting

Die Idee von Client-side Scripting ist die Verlagerung von Programmlogik vom Server auf den Client. Dadurch kann erreicht werden, daß der Server von Rechenlast befreit wird, die auch der Client ausführen könnte und daß insgesamt weniger Netzlast erzeugt wird, da Anfrage und Antwort nicht über das Netz transferiert werden müssen. Als Vertreter von Client-side Scripting Languages sind hier z.B. das von Netscape entwickelte JavaScript oder das von Microsoft stammende VBScript, daß an Visual Basic angelehnt ist, zu erwähnen. Die mit diesen Sprachen entworfenen Skripte sind in nichtkompilierter Form im Text einer HTML-Datei eingebettet. Client-side Scripting Languages dürfen keinen Zugriff auf die Ressourcen des ausführenden Systems erhalten, da ihr Inhalt grundsätzlich als gefährlich angesehen werden muß. Dies gilt insbesondere deshalb, da Signaturen nicht vorgesehen sind. Eine Ausnahme davon ist das Schreiben und Lesen sog. Cookies auf der Festplatte des Clients, die es ermöglichen, den Benutzer bei einem wiederholten Besuch der Seite wiederzuerkennen. Manche sehen darin einen Angriff auf ihre Anonymität, weswegen es auch möglich ist, die Annahme von Cookies zu verweigern.

6.8 Key Escrow Systeme / Key Recovery

Die Verwendung kryptographischer Verfahren bringt nicht nur Vorteile, sondern ist auch mit Nachteilen behaftet. Oftmals ist es im Sinne der Allgemeinheit, wenn gewisse Informationen nicht geheimgehalten werden können. Dies ist z.B. dann der Fall, wenn staatliche Einrichtungen gegen das organisierte Verbrechen vorgehen. In der heutigen Praxis können Informationen in diesen Fällen z.B. durch eine richterlich anzuordnende Überwachung des Telefonanschlusses gewonnen werden. Wenn nun aber diese Informationen nicht in Telefongesprächen, sondern in mit PGP verschlüsselten eMails übertragen werden, stehen dem Staat keinerlei Möglichkeiten zur Verfügung, die es gestatten, an diese Informationen zu gelangen.

Ein anderer Fall könnte z.B. dann auftreten, wenn ein Mitarbeiter einer Firma geheime Daten mit einem starken kryptographischen Verfahren verschlüsselt hat. Hier sind verschiedene Szenarien denkbar: zum einen könnte der Mitarbeiter den Schlüssel oder das Paßwort vergessen haben, er könnte auf Geschäftsreise oder in Urlaub sein. Genauso ist es möglich, daß die Datei mit dem Schlüssel zerstört worden ist oder daß der Mitarbeiter in der Zwischenzeit verstorben ist. In jedem Fall sind die Daten unwiederbringlich verloren, wenn das verwendete Verfahren nicht die Möglichkeit beinhaltet, den Schlüssel oder die Nachricht wiederzufinden.

Auf der anderen Seite wird befürchtet, daß staatliche Organisationen sich unbefugt eine Hintertür zu Daten öffnen könnten, die vor ihnen verborgen bleiben sollen. Insbesondere wird befürchtet, daß Daten von Firmen und anderen Organisationen ein Ziel nachrichtendienstlicher Aktivitäten werden könnten.

In der Diskussion befinden sich zur Zeit mehrere Modelle, wie der Klartext verschlüsselter Nachrichten von befugten Stellen wiedergewonnen werden könnte: zum einen können von staatlicher Seite kryptographische Verfahren vollständig verboten werden, was aber als unrealistisch gilt, da es der Wirtschaft große Probleme bei dem Austausch vertraulicher Daten bereiten würde. Eine andere Möglichkeit sieht die ausschließliche Benutzung von speziellen Verschlüsselungsalgorithmen vor, die eine Hintertür für staatliche Stellen offenhält. In den USA wurde dies bereits versucht, indem die NSA einen Chip, den sog. Clipper-Chip entworfen hat, der es der NSA ermöglicht hätte, die damit verschlüsselten Daten zu dekodieren. Das Verfahren ist im folgenden grob beschrieben: Der Clipper-Chip

verschlüsselt den Schlüssel, mit dem die Daten verschlüsselt worden sind mit einem geheimen Schlüssel, dem Unit-Key, der im Chip fest gespeichert ist. Der verschlüsselte Schlüssel wird dann in einem Law Enforcement Access Field (LEAF) an die Nachricht angefügt. Da die NSA den Unit-Key kennt, kann sie aus dem LEAF den Schlüssel berechnen, mit dem die Nachricht verschlüsselt wurde und diese dann entschlüsseln (Key Recovery). Der Clipper-Chip implementiert den Skipjack-Algorithmus, der von der NSA geheim gehalten wird und nur in Hardware implementiert werden soll.[80] Clipper ist bislang nicht von der Öffentlichkeit akzeptiert worden. Im Auftrag des Bundesamtes für Sicherheit in der Informationstechnik (BSI) wurde bei Siemens der sogenannte Pluto-Chip entwickelt, der eine ähnliches Ziel wie Clipper verfolgt. Über die zukünftigen Pläne zur Einführung dieses Chips sind keine Informationen verfügbar.

Weiterhin ist die Hinterlegung des geheimen Schlüssels bei einer vertrauenswürdigen Institution vorgeschlagen worden (Key-Escrow). Das Problem dabei ist, daß es in den Augen vieler eine solche Institution nicht gibt. Auch die Verwendung kryptographisch schwacher Verfahren stellt keine befriedigende Lösung dar, da ambitionierte Unternehmen oder auch Privatpersonen diese leicht brechen könnten.

In der Diskussion wird dabei oft die praktische Durchführbarkeit dieser Verfahren vernachlässigt, da es wohl nicht anzunehmen ist, daß Kriminelle die von ihnen hinterlegten Schlüssel benutzen. Außerdem ist es auch möglich, Daten mit starken kryptographischen Methoden zu verschlüsseln und das Ergebnis dann mit den schwächeren erneut zu verschlüsseln. Der Einsatz der starken Methoden könnte so erst bei einem konkreten Entschlüsselungsversuch bemerkt werden.

[80] Siehe zu weiteren Ausführungen: B. Schneier (1996), S. 591ff.

7 Implementierungs- und Designfehler

Wie jede Software, enthalten auch sicherheitsrelevante Programme Fehler aufgrund von falschem Design oder falscher Implementierung. Die Gefahr durch solche Fehler ist besonders groß, da der Benutzer der fehlerhaften Software von der falschen Annahme ausgeht, daß seine Software sicher sei.
In relativ kurzen Abständen werden denn auch Benutzer der Webbrowser Netscape Communicator und Microsoft Internet Explorer mit immer neuen Sicherheitslücken konfrontiert.[81] Besonders häufig waren dabei in jüngster Zeit die Schutzmechanismen für das Client-side Scripting, speziell JavaScript, betroffen. Die zuletzt gefundene Sicherheitslücke erlaubt es z.B., eine beliebige Datei auf dem Rechner des Benutzers auszulesen, ohne daß diese vorher bekannt sein mußte.

Auch die Java Virtual Machine (JVM) hat schon oft eine Lücke im Sicherheitssystem der Webbrowser dargestellt. So war es zum Beispiel möglich, den Security Manager der JVM des Netscape Navigators durch ein Applet auszuschalten, woraufhin das Applet vollen Zugriff auf das System erhielt.

Dies sind nur zwei Möglichkeiten, wie an sich gute Konzepte durch eine fehlerhafte Implementierung nutzlos werden können. Auch Designfehler, insbesondere bei den mathematisch komplexen Verschlüsselungsverfahren, haben schon des öfteren dazu geführt, daß sich solche Verfahren im Nachhinein als minderwertig herausgestellt haben. Im Gegensatz zu Implemtierungsfehlern sind Designfehler aber oft schwer oder gar nicht zu beheben.

Daß nicht nur Software für den Endverbraucher, sondern auch solche für den sensiblen Bereich der Geldautomaten von Banken fehlerhaft sein kann, zeigt eine Studie, die von der Universität Cambridge, Großbritannien durchgeführt worden ist.[82] Im folgenden werden daraus einige Beispiele angeführt, die nicht in direktem Zusammenhang mit der Themenstellung dieser Arbeit stehen, sie sollen lediglich zeigen, daß die angesprochenen Probleme nicht vernachlässigbar sind: selbst banale Implementierungsfehler können zu schweren finanziellen Schäden führen. So wird z.B. darüber berichtet, daß die Software eines Geldautomaten beim Einführen einer Telefonkarte davon ausgegangen ist, daß es sich dabei um die zuvor eingeführte Karte handele. Durch das Ausspähen der Geheimnummern von Kunden konnten sich somit einige Leute schnell auf Kosten anderer bereichern. In einem anderen Fall wird davon berichtet, daß ein Geldautomat nach der Eingabe einer 14-stelligen Zahl 10 Banknoten ausgezahlt hat. Dieses Feature war ursprünglich zu Testzwecken eingeführt worden. Die entsprechende Bank hat dies dann noch zu allem Überfluß in einer Dokumentation festgehalten, so daß sich das Verfahren schnell herumsprechen konnte. Auch ist ein Fall bekannt, bei dem allen Kunden einer Bank aufgrund eines Programmierfehlers die gleiche Geheimnummer zugewiesen wurde. Insbesondere auch Mitarbeiter der betroffenen Institute haben die genannten Programmierfehlern oftmals zu eigenen Zwecken mißbraucht.
Ferner werden Beispiele genannt, in denen die Banken die vorhandene Software nicht korrekt eingesetzt haben, weil es entweder zu teuer war, deren Schnittstellen vollständig an das System der Bank anzubinden oder es den Banken dazu an Erfahrung gefehlt hat.

Die Probleme der Webbrowser werden oftmals innerhalb weniger Tage nach Bekanntwerden behoben. Die genannte Studie zeigt hingegen, daß bei den Sicherheitssystemen der Banken oft über Jahre hinweg nicht gehandelt wurde, obwohl die Probleme bekannt waren. Des weiteren wird in dieser Studie

[81] Siehe dazu E. Kubaitis (1998)
[82] Vgl. R. Anderson (1993)

darauf hingewiesen, daß es sich dabei nicht alleine um ein Problem des Bankensektors oder des zivilen Sektors im allgemeinen handelt. Vielmehr ist dem Autor indirekt bestätigt worden, daß die gleichen Probleme auch im militärischen Bereich zu finden sind.

8 Zusammenfassung

In dieser Arbeit wurde ein Überblick über die notwendigen personellen, technologischen und organisatorischen Voraussetzungen für ein funktionierendes Sicherheitskonzept für den Einsatz des Internets als Informationsträger dargestellt. Es muß festgestellt werden, daß in vielen Unternehmen diesem Thema bei weitem nicht die Aufmerksamkeit zukommt, die ihm gebührt. Dabei ist insbesondere die Problematik der unzureichenden Kompetenzen und der unzureichenden Budgets der für die Internetsicherheit verantwortlichen Mitarbeiter zu erwähnen. Diese Probleme ergeben sich nicht zuletzt aus der Problematik, den Entscheidungsträgern im Management die komplizierte technische Materie und die Risiken, die ein Vernachlässigen des Themas mit sich bringt, zu vermitteln.

Weiterhin wurden notwendige Grundlagen der kryptographischen Verfahren angesprochen, die für ein Grundverständnis der Themen Verschlüsselung und Signaturen notwendig sind. Dabei wurden symmetrische und asymmetrische Verschlüsselungsverfahren vorgestellt. Es kann festgestellt werden, daß heute ausreichend sichere Verfahren zum Schutz vertraulicher Informationen zur Verfügung stehen. Es liegt jedoch beim Benutzer, geeignete Verfahren und hinreichende Schlüssellängen zu wählen, um den von ihm benötigten Schutz zu erzielen.

Im folgenden wurde auf Schwachstellen der im Internet üblicherweise verwendeten Protokolle und Dienste hingewiesen und Möglichkeiten zur Beseitigung von Schwachstellen aufgezeigt. Des weiteren wurden Maßnahmen zum Schutz von Firmennetzen gegen Kompromittierung, insbesondere von außen, vorgestellt. Besonders wichtig ist in diesem Zusammenhang das Konzept der Firewalls, das zwischen den einfachen, aber performanten Paketfiltern und den leistungsfähigen, aber im Datendurchsatz beschränkten Proxy Gateways unterscheidet.

Abschließend wurde die Gefährdung von Sicherheitskonzepten durch fehlerhafte Implementierungen und Designfehler aufgezeigt.

9 Anhang

An dieser Stelle sollen zwei weitere Verfahren zum Schutz offener Systeme (siehe Kapitel 5) beschrieben werden, die besonders geeignet sind, Systeme zu schützen, die nicht, wie beim Konzept der Firewalls ein abgegrenztes Teilnetz bilden, bzw. deren Teilnetze so groß sind, daß ein Schutz, der sich ausschließlich nach Außen orientiert, nicht mehr ausreicht.

9.1 Kerberos[83]

Der Kerberos „Security Service“ ist ein Software-System, das der Schlüsselvergabe, der Authentifizierung und der Einrichtung sicherer Kanäle dient. Es wurde am Massachusetts Institute of Technology, in Cambridge, USA, im Rahmen des Projekts Athena entwickelt. Es basiert auf einem symmetrischen Kryptosystem und ist als Public Domain im Sourcecode verfügbar.

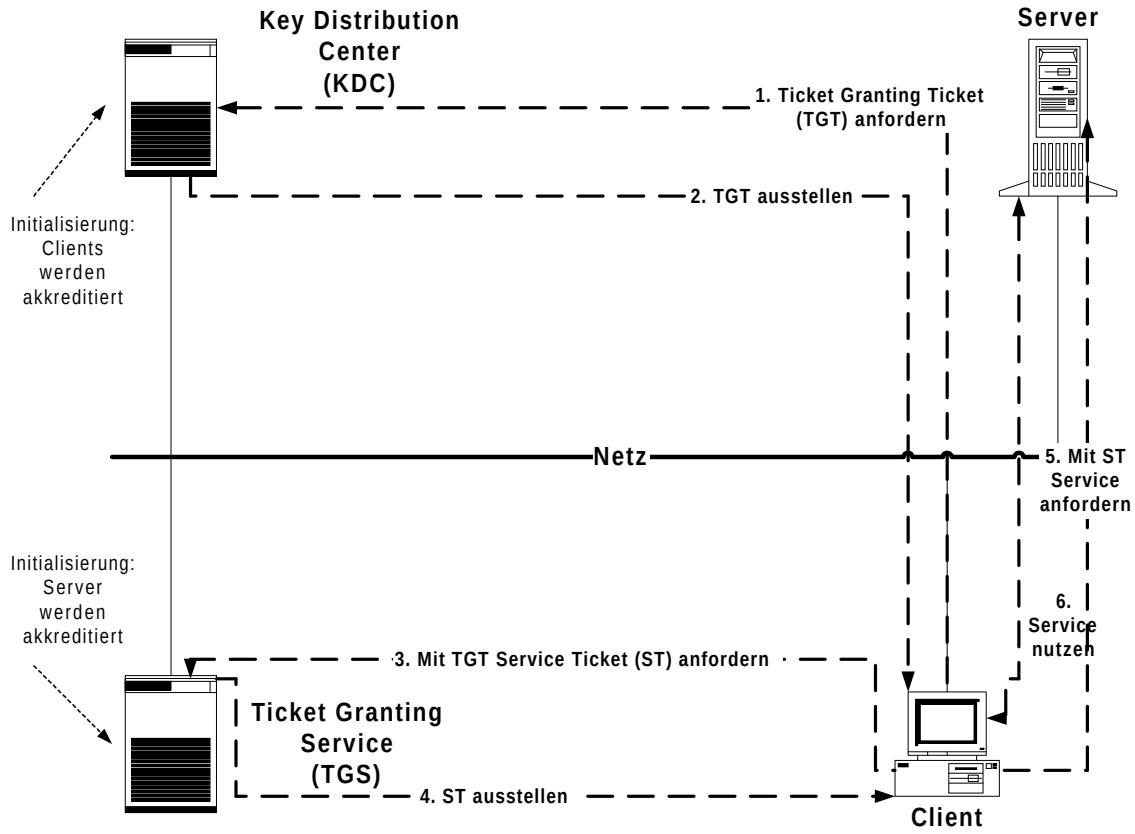
Kerberos implementiert mehrere Grundsätze, um den unerlaubten Zugriff auf das System und seine Dienste zu verhindern:

- Kommunikation findet nur verschlüsselt und mit authentifizierten Partnern statt. Dazu wird ein Sitzungsschlüssel verwendet, dessen Kenntnis als Authentitätsbeweis gilt.
- Paßwörter werden nie im Klartext übertragen und auf keinem anderen System als dem Schlüsselserver, dem sogenannten „**Key Distribution Center**“ (KDC), gespeichert.
- Alle Benutzer müssen sich beim KDC, alle Server beim „**Ticket Granting Service**“ (TGS) akkreditieren. Ein nicht akkreditiertes System, kann nicht als Server innerhalb des Kerberos-Systems auftreten, ebensowenig kann es, genau wie ein nicht akkreditierter Client, Dienste im System nutzen. Eine Nutzungsberechtigung, ein sogenanntes Ticket, ist wiederum nur in Verbindung mit einem weiteren Authentitätsnachweis, dem Sitzungsschlüssel, gültig.
- Die Gültigkeit von Tickets und Sitzungsschlüsseln ist zeitlich befristet.

Die Benutzer und ihre Paßwörter, die Schlüssel darstellen, werden dem KDC bekannt gemacht. Der TGS und dessen geheimer Schlüssel muß ebenfalls beim KDC akkreditiert werden. Server und deren geheime Schlüssel werden dem TGS bekannt gemacht. Da das KDC und der TGS die geheimen Schlüssel der Clients und Server speichern ist es notwendig diese beiden Systeme physisch zu sichern, denn wenn ein Unbefugter Zugriff auf diese Systeme erhält, ist die Sicherheit des System nicht mehr gewährleistet.

Wenn sich nun ein Client beim KDC anmeldet, dann authentifiziert er sich mit einem Initialschlüssel, der zwar aus dem Paßwort errechnet wird, jedoch nie das Paßwort im Klartext enthält. Das KDC sendet daraufhin, mit dem Benutzerschlüssel kodiert, sowohl den Sitzungsschlüssel als auch das sogenannte **Ticket Granting Ticket** (TGT) an den Benutzer. Das TGT gestattet es dem Benutzer, beim TGS die Erlaubnis – ein Ticket - zur Nutzung von Diensten im Netz anzufordern. Das TGT liegt jedoch nicht im Klartext vor, sondern ist mit dem Schlüssel des TGS kodiert, so daß nur der TGS das Ticket lesen kann. Der Client kann nun mit Hilfe des TGT beim TGS ein **Service Ticket** (ST) zur Nutzung eines bestimmten Dienstes bei einem bestimmten Server anfordern. Auch diese Kommunikation findet nur verschlüsselt statt. Das Service Ticket enthält wiederum einen Schlüssel für die Kommunikation zwischen Client und Server. Die Kommunikation wird mit dem speziell zwischen diesem Server und diesem Client in dieser Sitzung gültigen Schlüssel gesichert.

[83] Vgl. Steiner, G. et al. (1988), sowie Neumann, B. et al. (1988) und Schiller, J. I. (1995)



Darstellung 13 - Kommunikation im Kerberos-System

Kerberos setzt, wie bereits oben erwähnt, eine absolut sichere physische Sicherung des KDC und des TGS voraus. Da das System ohne KDC oder TGS nicht arbeitet, dürfen diese auch nie nicht zur Verfügung stehen. Um dies zu ermöglichen, können diese Systeme problemlos repliziert werden, da die Server zustandslos sind. Diese Replikation kann auch genutzt werden, um die Last auf mehrere Server zu verteilen. Die Clients sollten die erteilten Tickets möglichst sicher speichern, damit sie nicht geraubt werden können. Werden sie dennoch geraubt, sind sie nur begrenzt gültig. Eine Folgesitzung des gleichen Benutzers im System wird dadurch auch nicht beeinträchtigt, denn dann findet ein anderer Sitzungsschlüssel Verwendung.
Angreifbar ist das System primär dadurch, daß Paßwörter ein schlechter initialer Schlüssel für das Protokoll sind und dadurch, daß die Überprüfung der Gültigkeit eines Tickets auf Basis einer „globalen Zeit“ stattfindet, die als solches aber in verteilten Systemen nicht existiert. Die Zeit in einem solchen System kann nur ungefähr abgestimmt werden, indem die Uhren synchronisiert werden. Behindert ein Angreifer genau diesen Uhrenabgleich und ist so in der Lage, die Uhren von Teilsystemen zu beeinflussen, könnte er ein Ticket nach seinem Raub mehrfach verwenden.

In der Praxis ist Kerberos, das inzwischen in Version 5 vorliegt, am MIT seit 1986 im Einsatz. Dabei haben ca. 25.000 Benutzer über 1.200 Rechner auf das System Zugriff. Einige Schwachstellen, die in frühen Versionen aufgetreten sind, sind inzwischen beseitigt worden. Selbst Microsoft hat die Stärke von Kerberos inzwischen erkannt und wird es in Windows 2000 implementieren.

Da für Kerberos-Systeme inzwischen mehrere, inkompatible Varianten zur Verfügung stehen, ist es notwendig, daß man sich genau informiert, welche Variante man einsetzen sollte. Außerdem muß

bemerkt werden, daß Kerberos, wie man anhand des Protokolls leicht erkennen kann, nicht transparent arbeitet. D.h. die verwendete Software muß teilweise an Kerberos angepaßt sein / werden.[84]

9.2 Virtual Private Networks (VPN)[85]

Für Unternehmen stellt sich oft die Frage, wie sie mehrere Computernetze, beispielsweise von verschiedenen Niederlassungen oder Geschäftspartnern, kostengünstig verbinden können, ohne daß der Grad der Vertraulichkeit, den die Unternehmensdaten genießen, sinkt. Der Aspekt der niedrigen Kosten spricht für die Verwendung des Internets als Verbindungsmedium. Nun herrscht im Internet aber das genaue Gegenteil von Vertraulichkeit. Die Daten durchlaufen eine u.U. große Anzahl von Rechnern und können an jedem dieser Rechner kopiert, mitgelesen oder verändert werden. Um dieses Problem zu lösen, hat man die virtuellen privaten Netzwerke (Virtual Private Networks, VPN) konzipiert. Die Idee, die mit den VPN verfolgt wird, besteht darin, die Datenpakete beim Verlassen eines Teils des Unternehmensnetzes und dem darauffolgenden Eintritt in das Internet so zu verschlüsseln, daß die Nutzdaten nicht einfach zu entschlüsseln sind, ohne dabei aber den Transport der IP-Pakete über das Internet zu stören oder neue Protokolle im Internet zu benötigen. Dabei soll der Transfer über das Internet für die Rechner in den einzelnen Teilnetzen transparent stattfinden.

Auf welche Art und Weise kann man nun aber die Netzwerke sicher verbinden? Hier bieten sich zur Zeit zwei Standardvarianten an, die beide auf der Netzwerkschicht des ISO-OSI-Referenzmodells arbeiten. Die erste Variante nutzt das sogenannten Point-to-Point Tunneling Protocol (PPTP), das auf dem Point-to-Point Protocol[86] (PPP) aufbaut. Während das PPP primär zur Übertragung von nahezu beliebigen Netzwerkprotokollen über serielle Leitungen, wie beispielsweise Telefonleitungen, verwendet wird, dient das PPTP zur Übertragungen von nahezu beliebigen Netzwerkprotokollen über nahezu beliebige Protokolle, wobei hier allerdings IP als Übertragungsprotokoll am häufigsten Verwendung finden dürfte. PPTP sieht allerdings keinerlei Verschlüsselung vor, so daß diese Funktion durch zusätzliche Software, wie z.B. Erweiterungen des IP, erreicht werden muß. Die zweite Variante nutzt eine zum normalen Standard kompatible Version des IP-Protokolls, IPSEC[87] genannt. IPSEC kann nach dem „normalen“ IP-Protokoll verarbeitet werden, beispielsweise von einem gewöhnlichen Internet-Router, ohne daß sich daraus irgendwelche Probleme ergeben, der Datenteil jedoch nicht mitgelesen werden kann, da er nur ein Chiffrat enthält.

Als technische Voraussetzung muß nun in den zu verbindenden Netzwerken das Gateway zum Internet nur so modifiziert werden, daß IP-Pakete an eines der anderen zum VPN gehörenden Teilnetze erkannt werden und entsprechend des gewählten Tunneling-Protokolls in chiffrierte Datenpakete umgesetzt werden. Nach ihrem Weg durch das Internet muß nun das Gateway des Zielnetzes feststellen, daß ein anderes Teilnetzwerk des VPNs der Absender des IP-Pakets ist, muß dann das verschlüsselte Paket wieder in ein unverschlüsseltes Paket zurückwandeln und es daraufhin innerhalb des eigenen Netzes weiterleiten.

Um nicht die Sicherheitskonzepte der einzelnen Teilnetze zu unterlaufen und einen Angriff auf ein gut geschütztes Netz über eines der mit diesem Netz verbundenen Netze zu ermögliche, ist es zwingend notwendig, die Sicherheitskonzepte der zusammengeschalteten Netze aufeinander abzustimmen.

---

84 Für einen Vergleich zwischen Unix- und Kerberos-Sicherheit siehe Decker, Bart De (1993)

85 Vgl. Raepple, M. (1998), S. 174 ff.

86 Siehe Kapitel 6.2

87 Siehe zu weiteren Ausführungen: Raepple, M. (1998), S. 149 ff. und Oppliger, R. (1998), S. 160 ff., insbesondere S. 175-177

Virtual Private Networks sind für die Zukunft sicher ein vielversprechendes Konzept, um Unternehmensnetze über größere Distanzen zu verbinden. Es muß dabei nur berücksichtigt werden, daß, im Gegensatz zu einer angemieteten Standleitung, im Internet keine Mindestbandbreite vorausgesetzt werden kann. Dafür sind allerdings die Gesamtkosten langfristig oft geringer[88] als bei der dauerhaften Anmietung von Standleitungen, da sie sich i.d.R. auf die Anschaffung der entsprechenden Gateway-Hardware und die Zugangskosten zum Internet beschränken.

[88] Zu Thema Total Cost of Ownership bei VPNs siehe: SUN Microsystems, Inc. (1997)

Index

I

J

K

L

M

N

P

R

S

T

U

V

Z