Sicherheitsrisiken und Sicherheitsmechanismen im Internet

Studienarbeit

Peter Fecht
Universität Hamburg
Fachbereich Informatik

betreut von

Prof. Dr. Florian Matthes
Technische Universität Hamburg-Harburg
Arbeitsbereich Softwaresysteme

01. Juli 1997

Inhaltsverzeichnis

1 Einführung 1

1.1 Hintergrund und Motivation 1

1.2 Ziel der Arbeit 1

1.3 Gliederung der Arbeit 2

2 Sicherheitsanforderungen an betriebliche Informationssysteme 3

2.1 Begriffsbildung 3

2.2 Sicherheitskonzepte 4

2.3 Aufbau von Zugriffskontrollsystemen 5

3 Sicherheitsrisiken im Internet 7

3.1 Kommunikationsverlauf im Internet 8

3.2 Angriffsformen 8

3.3 Untersuchung der Sicherheitsrisiken von gängigen Internet-Diensten 9

3.3.1 Transport- und Netzwerkschicht : Die TCP/IP- Familie 9

3.3.2 Anwendungs-, Darstellungs und Sitzungsschicht 11

4 Existierende Sicherheitsmechanismen und Implementationen 13

4.1 Sicherheitserweiterungen von Anwendungen 13

4.1.1 Identifikation und Authentisierung 13

4.1.2 Kryptographie 15

4.1.3 Authentisierung durch asymmetrischer Kryptographie 18

4.1.4 Schlüsselverteilung und Verwaltung 21

4.2 Sicherheitserweiterungen von Internet-Diensten 22

4.2.1 Sicherheitserweiterungen des *Domain Name System* 22

4.3 Sicherheitsprotokolle der Anwendungs- und Darstellungsschicht : Das Secure Socket Layer - *Protokoll 26*

4.3.1 Das Secure Socket Layer - Protokoll Version 2.0....26
4.3.2 Modifikationen der SSL Version 3.0....32
4.3.3 Implementationen des SSL-Protokolls....32
4.4 Sicherheitserweiterungen der Netzwerkschicht : Das Internet Protokoll Version 6....33
4.4.1 Adressformate....33
4.4.2 Sicherheitserweiterungen....34
4.4.3 Schlüsselverwaltung....38
4.4.4 Authentisierung zwischen Benutzer und Host....38
5 Zusammenfassung und Ausblick....40
Anhang A Literaturverzeichnis....42
Anhang B Abkürzungsverzeichnis....45
Anhang C Der Standardisierungsprozeß der Internet-Organisationen....45

Abbildungsverzeichnis

Abbildung 2.1 : Elementare Sicherheitsanforderungen 3
Abbildung 2.2 : Aufbau eines Zugriffskontrollsystems 5
Abbildung 3.1 : ISO/OSI-Schichtenmodell.....8
Abbildung 4.1 : Funktionsweise des Frage / Antwort- Verfahrens.....14
Abbildung 4.2 : Aufbau des DNS und Datensatz - Zuordnung.....23
Abbildung 4.3 : Aufbau von Anfrage und Antwort 24
Abbildung 4.4 : Das SIG RDATA Format.....25
Abbildung 4.5 : Das KEY RDATA Format 25
Abbildung 4.6 : Verbindungen zu SSL 27
Abbildung 4.7 : Beziehung zwischen SSL und dem OSI-Modell.....28
Abbildung 4.8 : Struktur des SSL-Pakets.....29
Abbildung 4.9 : Verlauf des SSL-Handschlags.....30
Abbildung 4.10 : Format einer IPv6 - Adresse.....33
Abbildung 4.11 : Format des *Authentication Header* 35
Abbildung 4.12 : Format des ESP - Kopfes 37
Abbildung 5.1 : Internet - Organisationen.....45
Abbildung 5.2 : Der Standardisierungsprozeß von Internet-Dokumenten 46

Kapitel 1

Einführung

„Es ist einfach, ein Computersystem sicher zu betreiben. Sie müssen bloß alle Wählverbindungen abklemmen, ausschließlich direkt angeschlossene Terminals zulassen, diese Terminals und den Computer selbst in einen abgeschlossenen Raum bringen sowie eine Wache vor die Tür stellen."

F.T. Grampp und R.H. Morris

1.1 Hintergrund und Motivation

Aufgrund des starken Wachstums des Internets und der Nutzung durch Privatpersonen, wird die kommerzielle Nutzung für Firmen immer interessanter. Um über das Internet Kundenkontakte pflegen zu können, oder eine Abrechnung und Bezahlung von Dienstleistungen oder Produkten zu ermöglichen, muß die Identität von Anbietern und Kunden verifiziert werden können, und eine vertrauliche Übertragung von Daten gewährleistet sein.

Ein weiterer Wachstumsbereich ist die interne Nutzung von Internettechnologien, die als Intranet bezeichnet wird. Falls Firmen oder Organisationen den internen Informationsaustausch über ein Intranet realisieren möchten, sind zuverlässige Sicherheitsmechanismen eine notwendige Voraussetzung.

Es gibt somit vielfältige Anwendungsmöglichkeiten von Sicherheitsmechanismen im Internet und in Intranetzwerken. Das Internet als offenes, heterogenes Netzwerk fördert jedoch auch die Entwicklung zahlreicher, unabhängiger, teilweise sogar inkompatibler Mechanismen durch unterschiedliche Firmen und Organisationen.

1.2 Ziel der Arbeit

In dieser Arbeit wird ein Überblick über Sicherheitsmechanismen gegeben, die zur Kommunikation im Internet und in Intranetzwerken genutzt werden können. Hierbei werden vorrangig speziell für das Internet entwickelte Mechanismen, im allgemeinen Protokolle, jedoch auch allgemein anwendbare Verfahren, beispielsweise Verschlüsselungsalgorithmen, berücksichtigt.

Die einzelnen Mechanismen werden im Rahmen eines Überblicks vorgestellt. Hierzu wird die Funktionsweise beschrieben, ohne näher auf Detaillösungen und

Spezialfälle einzugehen. Die funktionale Beschreibung basiert auf Spezifikationen und nicht auf Implementationen, da diese von mehreren Firmen und Organisationen vorgenommen werden und voneinander abweichen können.

Da die Betrachtung von Sicherheitsaspekten im Internet erst seit Beginn der kommerziellen Nutzung verstärkt betrieben wird, liegen für viele Mechanismen und Verfahren noch keine Implementationen und Erfahrungswerte vor, so daß eine Bewertung der Stärken und Schwächen teilweise nur auf Basis der Spezifikation erfolgen kann. Werden bei Implementationen Sicherheitslücken aufgedeckt, muß allerdings unterschieden werden, ob sie auf der Spezifikation des Verfahrens oder auf dem Design der Implementation beruhen.

Die untersuchten Sicherheitsmechanismen sollen einen allgemeinen Schutz der Kommunikation Dritten gegenüber gewährleisten. Die Sicherheit von Kommunikationsverbindungen kann jedoch nur gewährleistet werden, falls die beteiligten Rechner vertrauenswürdig sind. Der Schutz dieser Rechner ist über kommuniaktionsunabhängige Mechanismen sicherzustellen und ist somit nicht Gegenstand dieser Arbeit.

Die im folgenden beschriebenen und bewerteten Sicherheitsmechanismen und Implementationen sollen als Entscheidungshilfe beim Design von schutzbedürftigen Anwendungen verwendet werden können.

1.3 Gliederung der Arbeit

Zunächst werden in Kapitel 2 die allgemeinen Sicherheitsanforderungen an betriebliche Informationssysteme vorgestellt. In Kapitel 3 erfolgt eine Beschreibung der Sicherheitsrisiken im Internet. Hierzu wird der Kommunikationsverlauf im Internet auf Basis des ISO/OSI-Schichtenmodells analysiert. Das Kapitel 4 dient der Vorstellung und Analyse existierender Sicherheitsmechanismen und Implementationen. Es folgt in seiner Struktur den Schichten des OSI-Modells.

Kapitel 2

Sicherheitsanforderungen an betriebliche Informationssysteme

Es werden zunächst die grundlegenden Begriffe erläutert, um darauf aufbauend die wichtigsten Basiskonzepte vorzustellen, die zur Erfüllung von Sicherheitsanforderungen verwendet werden können. Im Anschluß erfolgt eine Darstellung der einzelnen Komponenten eines Zugriffskontrollsystems.

2.1 Begriffsbildung

Eine sehr allgemeine Definition von Sicherheit ist „die Verhinderung unbefugter Aktivitäten, mit oder durch einen Computer und der zugehörigen Peripherie.“ [ChBe94]

Diese Definition kann spezialisiert werden, wenn bekannt ist, welche Komponenten einer EDV-Anlage vor wem geschützt werden sollen. In lokalen Netzen ist die wichtigste zu schützende Komponente die Datenbasis. Im Internet gewinnen die Kommunikationseinrichtungen eines Computers an Bedeutung, da ein Angreifer Zugriff auf die Ressourcen anderer Systeme erhalten kann, wenn er die Identität eines vertrauenswürdigen Systems vortäuscht. Die Stärke der einzusetzenden Sicherheitskonzepte ergibt sich aus der Qualifikation der zu erwartenden Angreifer. Die elementaren Anforderungen an ein sicheres System zeigt folgende Abbildung [Opp92]. [ChBe94, HaAt94]

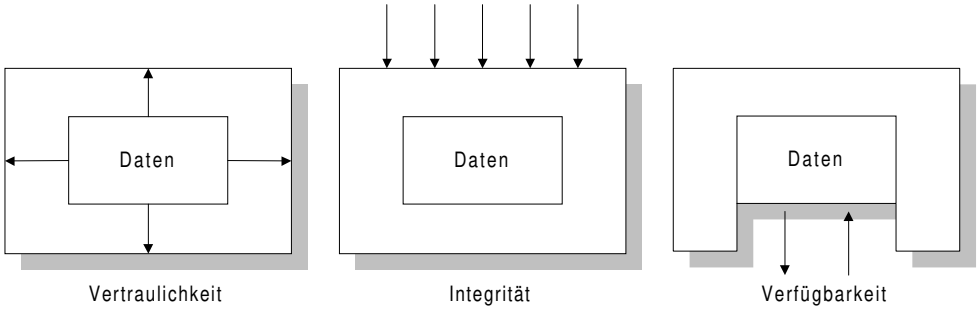

Abbildung 2.1 : Elementare Sicherheitsanforderungen

Vertraulichkeit *(confidentiality)*

Ihr Ziel ist der Schutz der Information, so daß Unbefugte keine Kenntnisse über die Bedeutung der Information erhalten, selbst falls der Zugriff auf den Datenträger (z.B. Disketten, Netzwerkpakete) gelingen sollte.

Integrität *(integrity)*

Die Integrität von Informationen gewährleistet, daß nur autorisierte Modifikationen an Daten vorgenommen werden, damit die Konsistenz der Datenbasis erhalten bleibt.
Die Einhaltung dieser Anforderung ergibt sich aus den Teilzielen des Schutzes vor Modifikationen durch nicht autorisierte Benutzer, des Schutzes vor inkonsistenten Modifikationen durch autorisierte Benutzer und der Einhaltung der Konsistenzbedingungen voneinander abhängiger Daten.

Verfügbarkeit *(availability)*

Diese Anforderung sichert zu, daß Informationen nicht unbefugt gelöscht oder beeinträchtigt werden können.

2.2 Sicherheitskonzepte

Die Einhaltung obiger Anforderungen kann durch den Einsatz vielfältiger Maßnahmen und Methoden sichergestellt werden, die auf folgenden Basiskonzepten beruhen.

Identifikation

Bevor Benutzer Zugang zu Systemen mit sicherheitsrelevanten Informationen erhalten, müssen sie sich in der Regel identifizieren. Hierzu wird meistens eine Benutzerkennung vergeben, die bei der Systemanmeldung anzugeben ist. Die Kenntnis einer Benutzerkennung sollte allerdings nicht ausreichen, um Zugang zu Systemen zu erhalten, da diese auch Unbefugten zur Verfügung stehen können. [Kaß95, ChBe94]

Authentisierung

Der Authentisierungsvorgang beschreibt die Verifikation der Daten, die während der Identifikation vom Benutzer übermittelt wurden. In der Regel geschieht dies durch Zuordnung eines Paßworts zur Benutzerkennung (Authentisierung durch Wissen).
Andere Möglichkeiten der Identifikation und Authentisierung sind Chipkarten (Authentisierung durch Besitz) oder bei besonders hohen Sicherheitsbedarf die Authentisierung durch persönliche Merkmale (z.B. Fingerabdrücke, Retinaabtastung). Konnte die Identität des Benutzers verifiziert werden, erhält er während der Autorisierung seiner Rolle entsprechende Zugriffsrechte auf Systemressourcen. [ChBe94, HaAt94]

Autorisierung

Hat ein Benutzer während der Authentisierung die Zugangsberechtigung zu einem System erlangt, darf er dennoch nicht alle Ressourcen nutzen. Vielmehr werden ihm nur Berechtigungen für Ressourcen erteilt, die zur Erledigung seiner Aufgaben notwendig sind.[1]

[1] Dieses Verfahren wird als *least privilege principle* bezeichnet.

Außerdem kann sich die Art der Berechtigung unterscheiden. Im einfachsten Fall erfolgt eine Unterteilung in Lese- und/oder Schreiboperationen. [ChBe94, Kaß95]

Protokollierung

Die Aufzeichnung sicherheitsrelevanter Ereignisse (als *auditing* bezeichnet) ist hilfreich bei der Prüfung, ob Benutzer Zugriff auf nicht freigegebene Ressourcen erlangen wollten, oder auf eine Benutzerkennung mehrere Anmeldeversuche mit unterschiedlichen Paßwörtern durchgeführt wurde.
Die Analyse von gelungenen oder versuchten Systemeinbrüchen kann dazu führen, das System sicherer zu gestalten. Außerdem können Einbrüche verhindert werden, indem Benutzerkennungen nach einer vorgegebenen Anzahl erfolgloser Anmeldeversuche gesperrt werden. [Kaß95]

Kryptographie

Die Kryptographie wird oft als ultimatives Werkzeug zur Gewährleistung von Vertraulichkeit angepriesen. Diesem Anspruch kann sie jedoch nur bei wohlüberlegtem Einsatz gerecht werden. Die Verschlüsselung von Daten dient primär der Gewährleistung von Vertraulichkeit. In diesem Bereich verrichten symmetrische Verschlüsselungsalgorithmen[2] gute Dienste.
Der Einsatz von asymmetrischen Algorithmen[3] macht die Kryptographie zu einem hervorragenden Werkzeug der Authentisierung. Die zwei größten Probleme beim Einsatz der Kryptographie sind die Stärke des Algorithmus und des verwendeten Schlüssels und die Verwaltung und Verteilung der Schlüssel.[4] [ChBe94]

2.3 Aufbau von Zugriffskontrollsystemen

Die Aufgabe von Zugriffskontrollsystemen ist es, die Funktionen und Daten zur Durchführung der oben beschriebenen Sicherheitskonzepte zur Verfügung zu stellen. Es läßt sich logisch in die drei Komponenten Datenbasis, Überwachungssystem und Autorisierungssystem zerlegen.

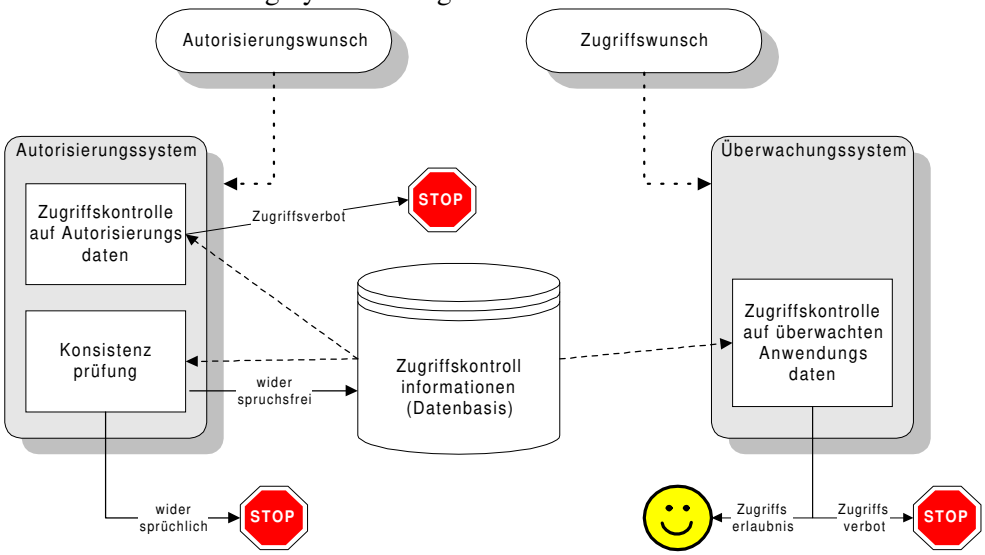

Abbildung 2.2 : Aufbau eines Zugriffskontrollsystems

[2] siehe auch Kap. 4.1.2 „Kryptographie“, Abschnitt „Symmetrische Kryptographie“
[3] siehe auch Kap. 4.1.2 „Kryptographie“, Abschnitt „Asymmetrische Kryptographie“
[4] siehe auch Kap. 4.1.4 „Schlüsselverteilung und Verwaltung“

In der Datenbasis werden die Zugriffsrechte gespeichert, die vom Autorisierungssystem verwaltet und vom Überwachungssystem benötigt werden, um Entscheidungen über Zugriffserlaubnisse beziehungsweise -verbote zu treffen. Da diese Datenbasis anwendungsunabhängig für das gesamte System und alle Benutzer erstellt wird, ist aufgrund der zu erwartenden Größe und der vom Zugriffsmodell abhängenden Komplexität der zu speichernden Zugriffsregeln, Datenbankfunktionalität nötig.

Das Autorisierungssystem verwaltet die Zugriffsregeln und hat zu prüfen, ob Erweiterungen oder Modifikationen dieser Regeln vom Benutzer durchgeführt werden dürfen.

Das Überwachungssystem greift schließlich auf die Datenbasis zurück, um entscheiden zu können, ob ein Benutzer für die angeforderten Anwendungsdaten Nutzungsrechte hat. [Kaß95]

Kapitel 3

Sicherheitsrisiken im Internet

Auf Basis der im vorhergehenden Kapitel angeführten Sicherheitsanforderungen und Konzepte zur Gewährleistung von Sicherheit erfolgt einer Untersuchung der im Internet genutzten Anwendungen, Protokolle und Techniken auf Sicherheitsrisiken. Hierzu werden die Ergebnisse der Analysen von Cheswick und Bellovin aus [ChBe94] verwendet.

Im Internet angewandte Verfahren und Techniken werden in Dokumenten mehrerer Internet-Gremien beschrieben. Der Oberbegriff dieser Dokumente lautet *Request for Comments (RFC)*. Da nur ein geringer Teil der RFC's tatsächlich Internet-Standards beschreibt, ist bei der Referenzierung von RFC's der Status zu beachten, da sich aus diesem die Vertrauenswürdigkeit und Stabilität des Dokuments herleiten läßt. Eine detaillierte Beschreibung des Standardisierungsprozesses der Internet-Gremien erfolgt in Anhang A. [Brad96]

3.1 Kommunikationsverlauf im Internet

Die Aufgaben, die während des Kommunikationsverlauf im Internet anfallen, können unter Verwendung des OSI-Modells den sieben aufeinander aufbauenden Schichten zugeordnet werden. Jede Schicht bietet hierbei ihre Dienste der nächsthöheren an und nimmt die Dienste der nächstniedrigeren in Anspruch. Die folgende Abbildung zeigt die Einordnung der wichtigsten Internet-Dienste in das OSI-Schichtenmodell.[5]

<table>
<tr><td>OSI-Schicht</td><td colspan="6">Anwendungen / Protokolle / Techniken</td></tr>
<tr><td>Anwendung</td><td>File-Transfer</td><td>Electronic Mail</td><td>Terminal Emulation</td><td>Usenet News</td><td>WWW Services</td><td>Domain Name Service</td></tr>
<tr><td>Darstellung</td><td rowspan="2">File Transfer Protocol

(FTP)
RFC 959</td><td rowspan="2">Simple Mail Transfer Protocol
(SMTP)
RFC 821</td><td rowspan="2">Telnet Protocol

(Telnet)
RFC 854</td><td rowspan="2">Network News Transfer Protocol
(NNTP)
RFC 977</td><td rowspan="2">Hyper Text Transfer Protocol
(HTTP)
RFC</td><td rowspan="2">Domain Name System

(DNS)
RFC 1034</td></tr>
<tr><td>Sitzung</td></tr>
<tr><td>Transport</td><td colspan="4">Transmission Control Protocol (TCP)
RFC 793</td><td colspan="2">User Datagram Protocol (UDP)
RFC 768</td></tr>
<tr><td>Netzwerk</td><td>Address Resolution Protocol (ARP)
RFC 826</td><td colspan="3">Internet Protocol (IP)
RFC 791</td><td colspan="2">Internet Control Message Protocol (ICMP) RFC 792</td></tr>
<tr><td>Sicherung</td><td colspan="6">Ethernet, Token Ring, FDDI etc.</td></tr>
<tr><td>Bit-Übertragung</td><td colspan="6">Doppelader, Koaxkabel, Lichtwellenleiter, drahtlose Übertragung</td></tr>
</table>

In Kapitel 3.2 werden im Internet mögliche Angriffsformen beschrieben, bevor in Kapitel 3.3 die Untersuchung der im OSI-Schichtenmodell eingeordneten Dienste auf Sicherheitsrisiken erfolgt.

3.2 Angriffsformen

Es gibt vielfältige Möglichkeiten, unbefugt in den Besitz von Daten zu gelangen und diese zu manipulieren, zu mißbrauchen oder den Schutz verschlüsselter Daten zu durchbrechen. Die wichtigsten, in den folgenden Kapiteln referenzierten Methoden, werden hier kurz beschrieben. [ChBe94, HaAt94]

Lauschen *(eavesdropping)*

Ein Lauscher hört die Kommunikationsverbindungen ab und gelangt so in den Besitz von Daten, die als Ausgangspunkt weiterer Angriffe dienen können. Ein erfolgreicher Lauschangriff ist die Voraussetzung für einen „Mann in der Mitte“ - Angriff.

[5] Für eine detailliertere Beschreibung des Aufbaus und der Funktionsweise des OSI-Schichtenmodells siehe [Kern92].

Mann in der Mitte *(man in the middle)*

Der Angreifer schaltet sich zwischen zwei Kommunikationspartner und spielt jedem den jeweiligen Partner vor. Auf diese Weise erlangt der Angreifer Informationen und kann Täusch- und Wiederholungsangriffe durchführen.

Täuschen

Ein Angreifer fügt Nachrichten ein oder löscht oder modifiziert legitime Nachrichten.

Wiederholung *(replay)*

Eine legitime Nachricht wird zu einem späteren Zeitpunkt wiederholt.

Zeit-Manipulation

Falls Protokolle zur Verifikation von Paketen die aktuelle Zeit benötigen, kann die Manipulation der Zeitvorstellung des Angriffsziels dazu verwendet werden, Wiederholungsangriffe auszuführen.

Wörterbuch-Angriff *(dictionary)*

Es wird versucht, von einem bekannten Paßwort auf andere Paßwörter zu schließen.

Vollständiges durchsuchen *(brute force)*[6]

Das Ausprobieren aller Schlüssel eines Schlüsselraums.

Kryptoanalyse

Das Entschlüsseln von Nachrichten, ohne im Besitz des Schlüssels zu sein.

3.3 Untersuchung der Sicherheitsrisiken von gängigen Internet-Diensten

Dieses Kapitel unterteilt sich in Dienste der Transport- und Netzwerkschicht und in die Anwendungen beziehungsweise den unterstützenden Diensten der Darstellungs- und Sitzungsschicht.

3.3.1 Transport- und Netzwerkschicht : Die TCP/IP[7]- Familie

Die TCP/IP-Protokollfamilie befindet sich in der Mitte des Schichtenmodells. Sie stellt die Verbindungen zwischen den Applikationen (z.B. *mail*, *login*, Videoserver)[8] und den Gerätetreibern für Netzwerkkarten, Modem und anderer Kommunikationshardware her.

[6] weiterführend siehe [Brad95]
[7] **T**ransmission **C**ontrol **P**rotocol / **I**nternet **P**rotocol
[8] *mail* und *login* sind UNIX-Anwendungen

Internet - Protokoll *(IP)*[9]

Die höheren Schichten benutzen das paketvermittelnde Internet-Protokoll um Daten zu empfangen oder zu versenden. Die IP-Pakete setzen sich aus dem IP-Kopf *(IP-header)* und dem Rumpf *(IP-body)* zusammen, der aus Nutzdaten besteht. Der Paketkopf beinhaltet die Quell- und Zieladresse, einige Bits zur Aktivierung von Optionen und eine Prüfsumme. Die Prüfsumme dient dazu, den Paketkopf und nicht die Integrität der Nutzdaten zu sichern. Eine Manipulation der Nutzdaten wird vom Internet - Protokoll somit nicht erkannt und muß von höheren Schichten verhindert werden. Zwei der optionalen Felder sind aus Sicherheitsgesichtspunkten interessant. Die Geheimhaltungsklassifikation (*security label*) erlaubt es, IP-Paketen eine Geheimhaltungsstufe[10] und eine Kategorie[11] mitzugeben. Mit dieser Option lassen sich mehrere Sicherheitskonzepte realisieren. Im Datennetz kann sichergestellt werden, daß Pakete nur auf Verbindungen übertragen werden, die für ihre Geheimhaltungsstufe freigegeben sind. Gegebenenfalls kann die Geheimhaltungsstufe einer Verbindung durch verschlüsselte Übertragung erhöht werden. Auch die Verwendung der Daten wird eingeschränkt, da ein Prozeß niedrigerer Geheimhaltungsstufe die Nutzdaten eines Pakets nicht auslesen darf[12].

Ein typisches IP-Paket umfaßt nur einige hundert Byte, so daß größere Datenmengen auf mehrere Pakete verteilt werden müssen. Dieser Vorgang wird Fragmentierung genannt und findet wie sein Gegenstück in der TCP-Schicht statt. Da das IP nicht verbindungsorientiert arbeitet, kann eine korrekte Übertragung von in mehreren Paketen zerlegten Daten allerdings nicht sichergestellt werden. Pakete können somit verlorengehen, dupliziert werden oder in der falschen Reihenfolge das Ziel erreichen. Eine sichere Übertragung zusammenhängender Pakete muß also von höheren Schichten gewährleistet werden.

Ein Sicherheitsrisiko bei Verwendung des Internet-Protokolls besteht darin, daß die Korrektheit der Quelladresse nicht garantiert werden kann. Zwar wird von vielen Betriebssystemen die Angabe einer falschen Quelladresse nicht zugelassen, aber verlassen kann sich der Empfänger auf die Gültigkeit nicht, so daß sich Sicherheitsmechanismen, wie zum Beispiel die Authentisierung, auf höhere Protokollschichten stützen.

Address Resolution Protokoll *(ARP)*[13]

Bei der Übertragung von IP-Paketen über ein Ethernet ergibt sich ein weiteres Problem, da die 32 Bit breiten IP-Adressen in das 48 Bit-Ethernet-Format umgesetzt werden müssen. Diese Zuordnungen werden vom *Address-Resolution* Protokoll vorgenommen. Im Ethernet wird der Empfänger eines Pakets per ARP-*broadcast*[14] der IP-Adresse gesucht. Es ist also möglich, das ein System sich mit einer gefälschten ARP-Antwort meldet und auf diese Weise den Datenverkehr auf sich umlenkt. Die Übertragung von IP-Paketen im Ethernet ist also nur sicher, solange ausschließlich vertrauenswürdige Maschinen im lokalen Netz senden dürfen.

Transmission Control Protokoll *(TCP)*[15]

[9] weiterführend siehe RFC 791
[10] zum Beispiel „Vertraulich“, „Geheim“, „Streng geheim“
[11] bei militärischer Nutzung zum Beispiel „Nuklearwaffen“ oder „Kryptographie“
[12] weiterführend siehe [Amor94]
[13] weiterführend siehe RFC 826
[14] Als *broadcast* wird der Versand einer Nachricht an alle bekannten Benutzer bezeichnet.
[15] weiterführend siehe RFC 793

Das *Transmission Control* Protokoll stellt basierend auf dem IP eine gesicherte virtuelle Verbindung zur Verfügung. Die Client-/Server-Prozesse kommunizieren über Ports miteinander. Ist eine Server-Portnummer kleiner als 1024, wird sie vom Administrator genutzt und kann in korrekt administrierten Systemen als sicher angesehen werden. Die zu übertragenden Pakete erhalten eine Laufnummer, damit verlorene oder beschädigte Pakete erneut übertragen und die ursprüngliche Sendefolge im Zielsystem wiederhergestellt werden kann.

User Datagram Protokoll *(UDP)*[16]

Das *User Datagram* Protokoll stellt Applikationen den Datagrammdienst des IP zur Verfügung. Er arbeitet verbindungslos und ohne Fehlerbehandlung. Da keine virtuellen Verbindungen aufgebaut werden, übernimmt nur ein Serverprozeß die Kommunikation. UDP-Pakete können einfach gefälscht werden, da kein Verbindungsaufbau mit Handschlag *(handshake)* stattfindet. Somit müssen Applikationen, die mit dem Datagrammdienst arbeiten, geeignete Sicherheitsvorkehrungen treffen.

Internet Control Message Protokoll *(ICMP)*[17]

Das *Internet Control Message* Protokoll dient dazu, das Verhalten von TCP- und UDP-Verbindungen zu beeinflussen. Es kann Hosts und Routern über Umleitungsnachrichten *(redirect messages)* andere Routen empfehlen. Hören Router auf gefälschte ICMP-Pakete, kann der Datenverkehr zu anderen Zielen umgelenkt werden. Aus diesem Grund sollten nur Hosts ICMP-Pakete auswerten.

Routing Information Protokoll *(RIP)*[18]

Bei Routing-Protokollen, wie zum Beispiel dem *Routing Information* Protokoll, besteht immer die Gefahr, daß Routen manipuliert werden. So kann einem angegriffenem Host ein vertrauenswürdiges System vorgetäuscht werden, falls sich dieser nur auf die Authentizität der IP-Adresse verläßt.

3.3.2 Anwendungs-, Darstellungs und Sitzungsschicht

Die Anwendungsschicht stellt Benutzern eine Vielzahl von Diensten zur Verfügung, die auf den Protokollen der Darstellungs- und Sitzungschicht basieren. Die wichtigsten dieser Dienste werden im folgenden untersucht.

Simple Mail Transport Protokoll *(SMTP)*[19]

Das *Simple Mail Transport* Protokoll wird meist für den Mail-Transport im Internet eingesetzt. Es ist ein einfaches Protokoll, welches den Austausch von 7-Bit-ASCII[20]-Zeichen ermöglicht.
Der Empfänger einer Nachricht hat keine Möglichkeit die Absenderadresse zu verifizieren. Sollen sicherheitsrelevante Informationen per SMTP versandt werden, müssen höhere Protokollebenen geeignete Mechanismen bereitstellen.
Ein weiteres Problem kann der Inhalt der Mail darstellen, falls dieser nach *Multipurpose Internet Mail Extensions (MIME)*[21] kodiert sind. In diesen Fall ist es möglich, in Mails Programmkode einzubinden, der auf Empfängersystemen ausgeführt

[16] weiterführend siehe RFC 768
[17] weiterführend siehe RFC 792
[18] weiterführend siehe RFC 1721-24
[19] weiterführend siehe RFC 821
[20] **A**merican **S**tandard **C**ode of **I**nformation **I**nterchange
[21] weiterführend siehe RFC 1896

wird. Weitere Sicherheitsprobleme des MIME-Standards werden in den MIME-Spezifikationen beschrieben.

Telnet[22]

Der Telnet-Dienst bietet einen einfachen Terminalzugang zu Systemen. Zur Authentifikation wird in der Regel ein Paßwort benötigt. Finden zwischen Maschinen, die sich vertrauen, Sitzungen statt, kann „secure telnet“ eingesetzt werden, um die übertragenen Daten zu verschlüsseln. Allerdings werden meist Sitzungen mit Maschinen gestartet, zu denen kein Vertrauensverhältnis besteht. Die übertragenen Daten, inklusive der Paßwörter, können in diesem Fall aufgezeichnet und manipuliert werden.

World Wide Web (WWW)

Unter dem Oberbegriff *World Wide Web* werden mehrere Informationsprotokolle zusammengefaßt. Neben *gopher*[23] und *wide area information servers (WAIS)24* zählt das *hyper text transfer protocol (HTTP)25* zum bekanntesten Vertreter dieser Klasse. Die Protokolle unterscheiden sich zwar im Detail, stimmen jedoch in einigen wichtigen Punkten überein.

Der Kommunikationsverlauf besteht aus Anfragen, die ein Host an einen WWW-Server sendet und Antworten vom Server, die unterschiedliche Formate haben können. Das bekannteste sind Textdateien im HTML[26]-Format, die auch MIME-Formate beinhalten können. Hieraus ergeben sich zahlreiche Probleme, da per MIME vom Anwender unbemerkt unterschiedlichste Informationen übertragen werden können.

Ein weiteres Problem besteht in der Referenzierung der WWW-Dateien, die über URL’s[27] erfolgt. Die Überprüfung des WWW-Servers, ob die angeforderte Datei zum Transfer freigegeben wurde, arbeitet nicht zuverlässig, so daß eine Authentisierung des anfordernden Host stattfinden müßte.

[22] weiterführend siehe RFC 854
[23] weiterführend siehe RFC 1436
[24] weiterführend siehe RFC 1614
[25] weiterführend siehe RFC 2068
[26] **H**yper **T**ext **M**arkup **L**anguage (weiterführend siehe RFC 1866)
[27] **U**nique **R**esource **L**ocator (weiterführend siehe RFC 1738)

Kapitel 4

Existierende Sicherheitsmechanismen und Implementationen

In diesem Kapitel werden existierende Sicherheitsmechanismen und Implementationen beschrieben, die von Diensten des OSI-Schichtenmodells genutzt werden können. In die Beschreibung der Mechanismen fließen teilweise Bewertungen ein, um eine Auswahl von zu nutzenden Sicherheitstechniken treffen zu können. Die Struktur dieses Kapitels folgt den Schichten des OSI-Modells.

4.1 Sicherheitserweiterungen von Anwendungen

Damit auf Anwendungsebene die Sicherheitsanforderungen von betrieblichen Informationssystemen erfüllt werden können, müssen Sicherheitsmechanismen zur Gewährleistung von Identifikation, Authentisierung und Autorisierung genutzt werden.

4.1.1 Identifikation und Authentisierung

Es gibt mehrere unterschiedliche Klassen von Authentisierungsmechanismen, die von keiner bis zu sehr starker Authentisierung reichen. Unterschiedliche Probleme erfordern unterschiedliche Mechanismen, so daß sich zwischen den Klassen keine strenge hierarchische Abhängigkeit festlegen läßt.

Keine Authentisierung

Ein System ohne Authentisierungsmechanismen aufzubauen ist nur sinnvoll, falls der Rechner keinem Netzverbund angehört oder die Benutzer gleiche Rechte haben und ein Mißbrauch durch nichtberechtigte Benutzer ausgeschlossen werden kann.

Schwache Authentisierungsmechanismen

Die einfachste Form der Authentisierung ist die Abfrage von Paßwörtern. Diese Paßwörter können vom Benutzer aus dem Gedächtnis eingegeben werden oder dem Authentisierungsmechanismus durch Nutzung anderer Medien übergeben werden. Gebräuchlich sind zum Beispiel Magnet- oder Chipkarten.

Bei Einsatz von biometrischen Verfahren werden biologische Merkmale als Paßwort genutzt[28]. Da es jedoch keine eindeutigen Signaturen biologischer Eigenschaften gibt, müßte der Authentisierungsprozeß mit Toleranzen arbeiten. Ein System das seine Benutzer nicht eindeutig authentisieren kann ist allerdings nicht sinnvoll, so daß als Lösung die Mitführung der biologischen Signatur auf einer Chipkarte in Frage käme. Im Internet wird gegenwärtig keine biometrische Authentisierung durchgeführt.

Einfache Authentisierungssysteme werden als enthüllend *(disclosing)* bezeichnet, da das Paßwort bei der Übertragung durch Netzwerke Lauschern gegenüber enthüllt wird. Eroberte Rechner können auf diese Weise leicht genutzt werden um weitere Paßwörter in Erfahrung zu bringen oder Zugang zur Datenbank des Sicherheitssystems zu erlangen. [HaAt94, ChBe94]

Starke Authentisierungsmechanismen

Um Angriffe durch Wiederverwendung alter Paßwörter abwehren zu können, wurden nicht-enthüllende Authentisierungssysteme entworfen, die weiterhin auf Paßwortabfragen basieren. Die einfachste Möglichkeit ein nicht-enthüllendes System zu realisieren, besteht darin, Paßwörter zu generieren, die nach einmaliger Benutzung ungültig werden. Dies ist eine sehr starke Verteidigung gegen Lauschangriffe.

Wird zur Generierung der Einmal-Paßwörter zusätzliche Hardware oder Software verwendet, lassen sich sehr sichere Frage / Antwort - Systeme *(challenge / response systems)* implementieren. Der Benutzer erhält ein Gerät mit gespeichertem, geheimen Schlüssel, welches nach Eingabe der vom Host gestellten Frage eine Antwort errechnet (das Einmal-Paßwort). Die Antwort wird dem Host mitgeteilt und von diesem verifiziert.[29]

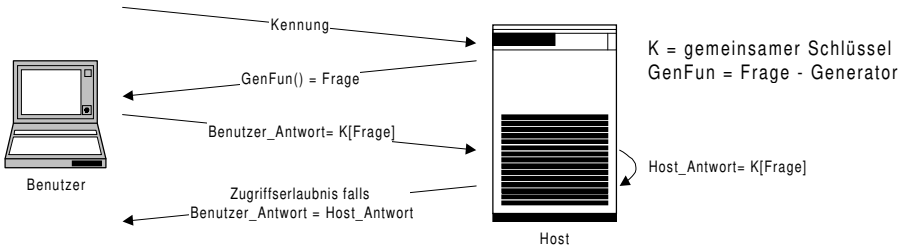

Abbildung 4.1 : Funktionsweise des Frage / Antwort- Verfahrens

Das Frage / Antwort - Verfahren basiert auf einem Gerät, welches gestohlen werden kann. Es wird somit ein zusätzlicher Schutz in Form einer PIN[30] benötigt, die den Benutzer des Geräts als rechtmäßigen Besitzer ausweist. Bei Eingabe einer falschen PIN würde eine ungültige Antwort generiert werden.

Ein Einmal-Paßwort-Verfahren, das ohne spezielle Hardware auskommt, wurde von Lamport entwickelt. Es basiert auf einer Funktion F, die mit geringem Aufwand berechnet werden kann, deren Umkehrung aber effektiv unmöglich zu berechnen ist.

[28] zum Beispiel Fingerabdruck, Retinascan oder Stimmanalyse

[29] Das Frage / Antwort - Verfahren leitet sich von Freund-Feind-Erkennungsgeräten militärischer Flugzeuge ab [Diff88]

[30] Eine *Personal Identification Number* wird häufig zum Schutz von Magnet- und Chipkarten verwendet

Der Benutzer und der Host haben einen gemeinsamen, geheimen Schlüssel K. Damit der Benutzer sich x-mal authentisieren kann, berechnet der Host folgende Funktion:

$$F^{x}(K)$$

Bei der ersten Anmeldung verwendet der Benutzer das Paßwort $F^{x-1}(K)$, und der Host überprüft die Gültigkeit folgender Gleichung :

$$F(F^{X-1}(K)) = F^{X}(K)$$

War das Paßwort $F^{x-1}(K)$ korrekt, speichert der Host diesen Wert, um als nächstes Paßwort $F^{X-2}(K)$ verifizieren zu können.

Das von Bellcore entwickelte S/Key-Authentisierungssystem[31] basiert auf dem Verfahren von Lamport. Es erlaubt dem Benutzer sich mehrere Paßwörter als Liste ausgeben zu lassen, falls dieser keine Möglichkeit zur Berechnung hat. Eine Gefahr besteht in der schlechten Wahl des geheimen Schlüssels, so daß dieser über Wörterbuch-Angriffe ermittelt werden könnte. [HaAt94, ChBe94, Lamp81]

Stärkste Authentisierungsmechanismen

Der starke Wachstum von vernetzten Rechnerumgebungen hat dazu geführt, daß noch stärkere Authentisierungsmechanismen nötig werden, da Benutzer in offenen Netzwerken die übertragenen Informationen abfangen und manipulieren oder Informationen mit gefälschtem Absender einspeisen können.

Leistungsfähigere Authentisierungssysteme nutzen die Rechenleistung der Beteiligten aus. Die Authentisierung kann unidirektional erfolgen, zum Beispiel bei der Authentisierung von Benutzern gegenüber einem Hostrechner, oder bidirektional, so daß der Benutzer die Identität des Hostrechners verifizieren kann.

Einige Authentisierungssysteme nutzen kryptographische Techniken und richten während des Authentisierungsprozesses ein geteiltes Geheimnis *(shared secret)* ein, welches für den folgenden Informationsaustausch genutzt werden kann. Dem Benutzer wird zum Beispiel ein Schlüssel erstellt, der ihm ermöglicht, weitere Ressourcen des Systems zu nutzen, ohne erneut eine Authentisierung durchlaufen zu müssen.
Diese Systeme können dem Benutzer auch Vertraulichkeit garantieren, da Informationen über unsichere Verbindungen verschlüsselt übertragen werden können. Im Anschluß an eine allgemeine Einführung in die Kryptographie wird die häufig genutzte Authentisierung durch öffentliche Schlüssel vorgestellt. [HaAt94, ChBe94]

4.1.2 Kryptographie

Dieses Kapitel bietet nur eine kurze Einführung in das Gebiet der Kryptographie und dient hauptsächlich der Begriffsdefinition und Bewertung der Stärke von kryptographischen Algorithmen. Auf die Funktionsweise dieser Algorithmen wird nicht näher eingegangen. Für detailliertere Beschreibungen siehe [Denn82, Schn94].

[31] weiterführend siehe [Hall94]

Kryptographische Mechanismen werden häufig in Netzwerken eingesetzt, um Authentisierung mit oder ohne der Gewährleistung von Vertraulichkeit zu ermöglichen. Es gibt zwei grundlegende Arten von Kryptographie, welche im folgenden erläutert werden. Ein grundlegendes Problem der Kryptographie ist die Verteilung der Schlüssel an die Beteiligten, da Unbefugte auf diese keinen Zugriff erlangen dürfen.

Moderne Kryptosysteme bestehen aus einer Funktion, die einen Klartext P und einen Schlüssel K auf ein Chiffrat C abbilden. Es ergibt sich folgende Notation :

$$C \leftarrow K\,[P]$$

Zur Entschlüsselung wird eine Umkehrfunktion und ein Schlüssel K^{-1} benötigt :

$$P \leftarrow K^{-1}\,[C]$$

Angreifer versuchen in der Regel die Schlüssel K und K^{-1} zu bestimmen. Bei einem starken Kryptosystem sollte es unmöglich sein, die Schlüssel durch analytische Maßnahmen zu bestimmen, so daß den Angreifern nur das vollständige Durchsuchen des Schlüsselraums bleibt.
Die Güte eines Verschlüsselungsalgorithmus wird also von seiner Funktionsweise, die eine analytische Bestimmung von Schlüsseln ausschließen muß, und der Breite von Schlüsseln, aus der sich die Mächtigkeit des Schlüsselraums ergibt, bestimmt.
Die Schlüsselbreite wird üblicherweise in Bit angegeben, wobei zu beachten ist, daß die effektive Schlüsselbreite, also die Anzahl der tatsächlich zur Verschlüsselung genutzten Bits, deutlich geringer sein kann. [ChBe94]

Symmetrische Kryptographie

Symmetrische Kryptographie[32] schließt alle Systeme ein, die denselben Schlüssel zum Ver- und Entschlüsseln benutzen. Es gilt somit

$$K = K^{-1}$$

Somit kann ein Unbefugter mit dem Schlüssel sowohl verschlüsselte Nachrichten entschlüsseln, als auch Nachrichten verschlüsseln und diese vertrauenswürdig erscheinen lassen. Erlangt eine dritte Partei Kenntnis über Schlüssel des Systems ist die Vertrauenswürdigkeit nicht mehr gewährleistet.

Der weit verbreitete *Data Encryption Standard (DES)* [33] -Algorithmus ist wahrscheinlich der beste symmetrische Verschlüsselungsalgorithmus. Er arbeitet mit sechzehn Wiederholungen von Permutationen und Substitutionen, die anschließend per „Exklusiv-Oder“ mit der Eingabe verknüpft werden. Der Algorithmus benutzt einen 64 Bit breiten Schlüssel[34].
Da bei dieser Schlüsselbreite die Durchsuchung des Schlüsselraums zum Erfolg führen kann, ist es sinnvoll die *Triple-DES* -Verschlüsselung zu verwenden. Sie arbeitet mit zwei Schlüsseln K_1 und K_2 :

$$C \leftarrow K_1\,[K_2^{-1}\,[K_1[P]]]$$

Werden K_1 und K_2 gleichgesetzt, bleibt der Algorithmus kompatibel zu DES.

[32] auch als klassische, konventionelle oder *Private-Key*-Kryptosysteme bezeichnet
[33] weiterführend siehe [NBS77, NBS80]
[34] Die effektive Schlüsselbreite beträgt 56 Bit.

Der *IDEA*[35] - Algorithmus ähnelt dem DES in seiner Struktur. Er nutzt jedoch nicht nur die „Exklusiv-Oder"-Verknüpfung, sondern kombiniert diese mit „Modulo-Addition" und „Modulo-Multiplikation". Die Schlüsselbreite beträgt 128 Bit, so daß eine Durchsuchung des Schlüsselraums nicht durchführbar ist. [ChBe94, HaAt94]

Asymmetrische Kryptographie

In den späten 70er Jahren führte ein Durchbruch in der Kryptographie zur Verfügbarkeit von asymmetrischen kryptographischen Algorithmen. Diese Klasse von Algorithmen benutzt zur Ver- und Entschlüsselung von Daten zwei verschiedene Schlüssel. In solchen Systemen gilt also $K \neq K^{-1}$. Außerdem kann aus dem öffentlichen Schlüssel *(public key)* K nicht der private Schlüssel *(private key)* K^{-1} abgeleitet werden.

Besitzt der Benutzer A die Schlüssel ergibt sich für die Verschlüsselung:

$$C \leftarrow E_A [P]$$

und für die Entschlüsselung

$$P \leftarrow D_A [C].$$

Die Benutzer verteilen ihren öffentlichen Schlüssel und halten ihren privaten Schlüssel geheim. Einem Benutzer kann eine Nachricht gesendet werden, indem sein öffentlicher Schlüssel zur Verschlüsselung benutzt wird. Nur der Empfänger kann die Nachricht mit seinem privaten Schlüssel entschlüsseln.

Das Problem der Schlüsselverteilung hat sich hierdurch sehr vereinfacht, da der öffentliche Schlüssel unverschlüsselt über diverse Dienste veröffentlicht werden kann.

Der bekannteste und wichtigste asymmetrische Verschlüsselungsalgorithmus basiert auf der Arbeit von Rivest, Shamir und Adleman und wird als *RSA*[36] abgekürzt. Der Algorithmus basiert auf der Schwierigkeit, sehr große Zahlen zu faktorisieren. Zur Nutzung von RSA werden zwei große Primzahlen p und q benötigt[37]. Zunächst muß $n = pq$ berechnet werden. Außerdem wird eine zufällige, natürliche Zahl d benötigt, die relativ prim zu $(p-1)(q-1)$ ist, sowie ein e für das gilt:

$$ed \equiv 1 \pmod{(p-1)(q-1)}$$

Das Paar (e, n) wird als öffentlicher Schlüssel und das Paar (d, n) als privater Schlüssel genutzt. Die Verschlüsselung eines Klartexts P erfolgt durch Exponentiation mit *e modulo n* :

$$C \leftarrow P^e \pmod{n}$$

Die Entschlüsselung verwendet statt dessen d als Exponent:

$$\begin{aligned} P \leftarrow C^d \pmod{n} &\equiv (P^e)^d \pmod{n} \\ &\equiv P^{ed} \pmod{n} \\ &\equiv P \pmod{n} \end{aligned}$$

[35] weiterführend siehe [Lai92]

[36] Rivest, Shamir und Adleman gründeten zur Vermarktung des Algorithmus die RSA Data Security Inc.

[37] p und q sollten mehrere hundert Bit lang sein

Der Sicherheit von asymmetrischen Kryptosystemen stehen allerdings zwei große Nachteile gegenüber. Die Schlüssel sind im Vergleich zu konventionellen Systemen sehr groß, woraus sich Probleme bei der Eingabe oder Übermittlung ergeben können. Der zweite Nachteil ist der langsame Ver- und Entschlüsselungsprozeß. Dieser Nachteil kann vernachlässigt werden, falls solche Systeme nur zur Schlüsselverteilung eingesetzt werden. Der übertragene Schlüssel kann dann zur Verschlüsselung mit schnelleren symmetrischen Systemen genutzt werden. [ChBe94, HaAt94]

Kryptographische Prüfsummen

Kryptographische Prüfsummen gewährleisten Datenintegrität und Authentisierung. Einige Protokolle bilden Prüfsummen über Schlüssel und die zu authentisierende Informationen. Dies sichert die Echtheit des Datenursprungs, allerdings nicht die Echtheit der übertragenen Informationen. Da diese Prüfsummen nur sehr schwer zu fälschen sind, ergibt sich ein sehr starker Authentisierungsmechanismus. Das größte Problem bei der Implementation von kryptographischen Prüfsummen ist wiederum die Schlüsselverteilung. [HaAt94]

Digitale Unterschriften *(digital signatures)*

Eine digitale Unterschrift ist ein kryptographischer Mechanismus, welcher das elektronische Äquivalent zur eigenhändigen Unterschrift ist. Sie gewährleistet gegenüber dem Empfänger die Echtheit des Absenders, kann aber auch dazu dienen Nachrichten einem Absender zuzuordnen, selbst wenn dieser den Versand bestreitet.[38]

Eine digitale Unterschrift gewährleistet Authentisierung, jedoch nicht Vertraulichkeit. Dieser Mechanismus kommt bei der Authentisierung durch öffentliche Schlüssel zum Einsatz, die im folgenden Kapitel erläutert wird. [HaAt94]

4.1.3 Authentisierung durch asymmetrische Kryptographie

Die *RSA Public Key* - Kryptographie ist zur Authentisierung und Verschlüsselung weit verbreitet. Im folgenden wird die Verwendung zur Authentisierung unter Bezug auf [RSA93, Nets96a, Nets97a, ChBe94] demonstriert.

Der Verschlüsselungsalgorithmus benutzt das im vorhergehenden Kapitel eingeführte Schlüsselpaar, bestehend aus öffentlichen und privaten Schlüssel.
Werden Daten mit dem privaten Schlüssel verschlüsselt, können sie nur mit dem öffentlichen Schlüssel entschlüsselt werden. Umgekehrt können mit dem öffentlichen Schlüssel verschlüsselte Daten nur vom Besitzer des privaten Schlüssels entschlüsselt werden. Diese Eigenschaft macht die asymmetrische Kryptographie so nützlich.

Öffentlicher und privater Schlüssel

Zur Authentisierung einer Identität wird der Person eine Nachricht geschickt, die vom Empfänger nach Verschlüsselung mit dem privaten Schlüssel zurückgeschickt wird. Sie kann nun mit dem bekannten öffentlichen Schlüssel entschlüsselt werden. Stimmen die Nachrichten überein, muß die verschlüsselte Nachricht von B stammen, da nur er den entsprechenden privaten Schlüssel kennt.

[38]Diese Eigenschaft wird als *non-repudiation* bezeichnet.

A→B	Nachricht
B→A	{Nachricht}B's privater Schlüssel

Allerdings sollte B eine von A gesandte Nachricht nicht einfach verschlüsseln, da dies nachweislich nur von ihm erfolgt sein kann, da er als einziger den privaten Schlüssel kennt. Eine bessere Methode ist, eine Prüfsumme über der Nachricht zu errechnen, und diese verschlüsselt zurückzusenden. Diese Prüfsumme wird als *digest* bezeichnet und hat die folgenden Eigenschaften:

- Aus der Prüfsumme kann nicht die ursprüngliche Nachricht zurückgewonnen werden.
- Unterschiedliche Nachrichten erhalten unterschiedliche Prüfsummen.

Diese Technik wird digitale Unterschrift genannt.

Digitale Unterschrift

Diese Methode muß jedoch noch modifiziert werden, da B sonst die Nachricht von A unterschreibt. Die von B unterschriebenen Daten sollten auch von B stammen.

A→B	Anfrage an B
B→A	Nachricht
	{ Prüfsumme [Nachricht] } B's privater Schlüssel

A kann die Prüfsumme entschlüsseln und mit der selbst errechneten Prüfsumme der unverschlüsselt übertragenen Nachricht vergleichen.

Es muß allerdings noch das Problem der Schlüsselverteilung gelöst werden, da bei der Übertragung von öffentlichen Schlüsseln die Identität des Eigentümers authentisiert werden muß. Um dies gewährleisten zu können, werden Zertifikate verwendet.

Zertifikate *(certificates)*

Ein Zertifikat beinhaltet folgende Informationen :

- den Aussteller des Zertifikats
- den Inhaber des Zertifikats
- den öffentlichen Schlüssel
- einige Zeitstempel

Das Zertifikat wird mit dem privaten Schlüssel des Ausstellers unterschrieben *(signed).* Der öffentliche Schlüssel des Ausstellers ist allgemein bekannt, woraus folgt, daß auch der Aussteller ein Zertifikat besitzt. Zertifikate sind eine übliche Methode um öffentliche Schlüssel an Namen zu binden.

A→B	Anfrage
B→A	B's Name, **B's Zertifikat**
A→B	Beweisanforderung
B→A	Nachricht
	{ Prüfsumme [Nachricht] } B's privater Schlüssel

Person A kann nun bei Erhalt der ersten Nachricht von Person B das Zertifikat mit dem öffentlichen Schlüssel des Ausstellers entschlüsseln und erhält so den Namen und öffentlichen Schlüssel von Person B. Als Beweis erhält sie eine Nachricht und

einen mit B's privaten Schlüssel verschlüsselte Prüfsumme. Ist die entschlüsselte Prüfsumme und die selbst errechnete Prüfsumme identisch, kann Person A sicher sein, mit Person B zu kommunizieren und seinen öffentlichen Schlüssel zu besitzen.

Geheime Schlüssel

Sobald sich Person A der Identität Person B's sicher ist, kann sie Person B eine Nachricht senden, die nur Person A entschlüsseln kann.

A→B { Geheimnis } B' öffentlicher Schlüssel

Das Geheimnis kann nur mit B's privaten Schlüssel entschlüsselt werden. Der Austausch von Geheimnissen ist eine andere mächtige Verwendungsmöglichkeit der asymmetrischen Kryptographie.

Aus dem Geheimnis kann nun wiederum ein Schlüssel für einen Verschlüsselungsalgorithmus generiert werden. Dies kann durch Anwendung einer komplexen Funktion geschehen oder einfach durch direkte Verwendung des Geheimnis. Der gewonnene Schlüssel wird als geheimer Schlüssel (*secret key)* bezeichnet und von symmetrischen Kryptographiealgorithmen (z.B. DES, RC4, IDEA usw.) benutzt.

Da nur Person A und Person B diesen Secret Key besitzen, können nun Nachrichten mit symmetrischen Algorithmen verschlüsselt und ausgetauscht werden.

Message Authentication Code (MAC)

Wird der Datenverkehr zwischen A und B von einer dritten Person C abgehört[39], besteht die Gefahr, daß C die ausgetauschten Daten manipuliert. Da ihm der geheime Schlüssel unbekannt ist, kann er die Nachrichten nicht entschlüsseln. Allerdings kann er die Daten manipulieren und hoffen, daß A oder B sie als gültige Nachricht anerkennen. Um eine Manipulation von Daten ausschließen zu können, wurde der *Message Authentication Code (MAC)* eingeführt.

MAC := Prüfsumme [Nachricht , Geheimnis]

Der MAC ist eine Prüfsumme, die über den Daten unter Verwendung eines Geheimnisses errechnet wird. Ein gängiger Prüfsummen-Algorithmus ist der MD5 von RSA., der mit einer 128 Bit breiten Prüfsumme arbeitet und ein erraten des MAC unmöglich macht.

Da C das Geheimnis nicht kennt, kann er nach einer Manipulation der Daten keine gültige Prüfsumme errechnen. Somit erkennen A und B die Manipulation und können die Kommunikation abbrechen.

Nach Einführung des MAC ergibt sich folgender Protokollverlauf:

A→B	Anfrage
B→A	B's Name, B's Zertifikat
A→B	Beweisanforderung
B→A	Nachricht { Prüfsumme [Nachricht] } B's privater Schlüssel

[39] siehe auch „Mann in der Mitte" - Angriff in Kap. 3.2 „Angriffsformen"

A→B	{ Geheimnis } B's öffentlicher Schlüssel
B→A	{ Nachricht , MAC } geheimer Schlüssel

Person C kann die Nachrichten weder lesen noch manipulieren, er kann sie jedoch speichern und zu einem späteren Zeitpunkt wiederholen.[40]

Wiederholungsangriff

Um einen Wiederholungsangriff bemerken zu können, werden den Nachrichten zufällige Elemente beigefügt, die bei einer Wiederholung der Nachrichten zur Erkennung des Angriffs führen. Diese Elemente können zum Beispiel. Zeitstempel sein, die von Dritten nicht manipuliert werden können, da sie in die verschlüsselte Nachricht eingebettet werden.

4.1.4 Schlüsselverteilung und Verwaltung

Die Geheimhaltung von Schlüsseln ist eine elementare Anforderung von Kryptosystemen. Da Schlüssel jedoch verbreitet werden müssen, um seinen Kommunikationspartnern die Entschlüsselung von Daten zu ermöglichen, sind Verfahren nötig, die eine sichere Verteilung von Schlüsseln gewährleisten können.

Ein Internet-Standard zur Schlüsselverteilung existiert nicht, es beschäftigen sich jedoch mehrere Dokumente im Entwurfsstadium *(drafts)*[41] mit dieser Problematik. Die vorgeschlagenen Verfahrensweisen werden bereits von einigen kommerziellen Produkten[42] genutzt

Der in Kapitel 4.2.1 beschriebene *Domain Name System - Service* kann ebenfalls zur Verteilung von Schlüsseln verwendet werden.

Der *Kerberos-Authentisierungs-Service*[43] ist der bekannteste Dienst zur Authentisierung und Schlüsselverteilung. Er basiert auf dem symmetrisch verschlüsselnden DES-Algorithmus. Ein vertrauenswürdigen Host wird hierbei als dritte Partei genutzt, der alle geheimen Schlüssel der Benutzer und Dienste kennt. Der Kerberos-Host generiert Zertifikate, die es Benutzern erlauben sich gegenüber anderen Systemen, die den Kerberos-Service nutzen, als vertrauenswürdig auszuweisen. Der Kerberos-Host muß absolut sicher sein, da die Enthüllung eines Schlüssels dazu führt, daß Zugang zu sämtlichen angeschlossenen Systemen gewährt wird.

Auf asymmetrischen Algorithmen basiert *das Diffie-Hellmann-Schlüsselaustauschverfahren*[44] und die bereits beschriebene *RSA-Public-Key-Authentisierung*. Diese Verschlüsselungssysteme erfordern jedoch viel Rechenzeit und sind somit zur Verschlüsselung von Paketen innerhalb eines Netzwerkes ungeeignet. Ihre asymmetrische Eigenschaft hingegen eignet sich hervorragend zum Austausch von geheimen Schlüsseln. Ein Vorteil bei der Nutzung von asymmetrischen Techniken ist der Wegfall des zentralen Schlüssel-Servers.

[40] siehe auch Wiederholungsangriff in Kap. 3.2 „Angriffsformen"
[41] siehe [Blak96]
[42] beispielsweise OSF DCE 1.1, SESAME, OMG CORBA, Lotus Notes, Novell Netware
[43] weiterführend siehe [KoNe93, Will96]
[44] weiterführend siehe [DiHe76]

4.2 Sicherheitserweiterungen von Internet-Diensten

Zur Erfüllung von Sicherheitsanforderungen können teilweise auch Systemdienste des Internets herangezogen werden. Der *X.500 - Directory-Service* kann in Verbindung mit *X.509* zur Schlüsselverteilung verwendet werden. Auf die Nutzung dieser Dienste wird hier nicht weiter eingegangen. Weitere Informationen können den Dokumenten [BaKi, CC88a, CC88b, ISO88] entnommen werden.

Der im folgenden Kapitel beschriebene *Domain Name System* - Service kann zur Authentisierung und Schlüsselverteilung genutzt werden. Detailliertere Beschreibungen des DNS-Dienstes befinden sich in [EaKa97, Stahl87, Lott87, Mock87a, Mock87b].

4.2.1 Sicherheitserweiterungen des *Domain Name System*

Die Sicherheitserweiterungen des *Domain Name System (DNS)* - Protokolls unterstützen drei verschiedene Dienste :

- Verteilung von Schlüsseln
- Authentisierung von Daten
- Authentisierung von Transaktionen und Anfragen *(requests)*

Nicht unterstützt werden Dienste, die den Zugriff auf das DNS durch Zugriffskontrolllisten oder ähnliche restriktive Maßnahmen beschränken. Der Grundgedanke des DNS ist die Verteilung öffentlicher Informationen und die Gleichbehandlung aller Anfragen an das System. Diese Eigenschaft darf sich auch nach der Einführung von Sicherheitsmaßnahmen nicht ändern.

Außerdem wird vom DNS Vertraulichkeit nicht implizit unterstützt. Sollte diese Sicherheitseigenschaft von Benutzern benötigt werden, kann sie von anderen Diensten bereitgestellt werden [45]

Aus der Spezifikation des DNS ergibt sich eine hierarchische Struktur mit den entsprechend zugeordneten Ressourcen-Datensätzen *(resource records, RR)*, die im folgenden erläutert werden.

[45] z.B. *Internet Protocol Security (IPSEC)*, siehe auch Kap. 4.4 „Sicherheitserweiterungen der Netzwerkschicht“

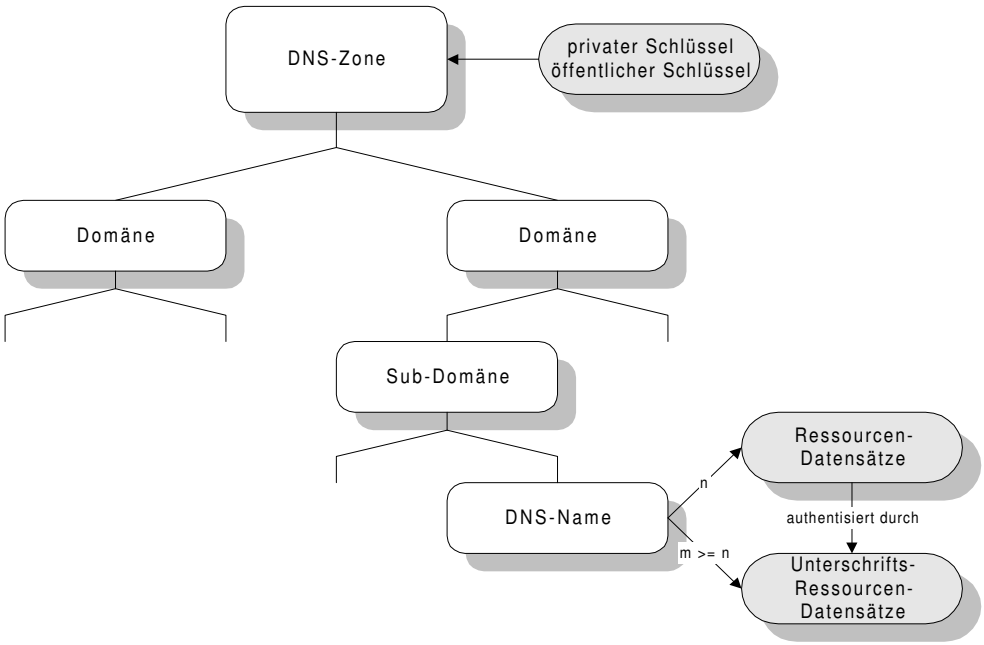

Abbildung 4.2 : Aufbau des DNS und Datensatz - Zuordnung

Schlüsselverteilung

Die Zuordnung von Namen zu Schlüsseln wird mit Ressourcen-Datensätzen durchgeführt. Dieser Mechanismus erlaubt es, den DNS zur Verteilung öffentlicher Schlüssel einzusetzen. Dieser Dienst kann von unterschiedlichen Sicherheitsmechanismen genutzt werden (z.B. DNS-Datenauthentisierung, IPSEC[46]).

Authentisierung von DNS-Einträgen

Die Authentisierung von DNS-Einträgen wird mit asymmetrischer Kryptographie durchgeführt. Hierzu werden die Ressourcen-Datensätze mit dem privaten Schlüssel einer Zone verschlüsselt. Benutzer des DNS-Dienstes benötigen zur Entschlüsselung der digital unterschriebenen Anwort den entsprechenden öffentlichen Schlüssel der Zone und können dann eine Authentisierung der Antwort durchführen. Die öffentlichen Schlüssel von Zonen können über das DNS angefordert werden. Da die übertragenen Schlüssel natürlich verschlüsselt sind, muß mindestens ein öffentlicher Schlüssel dem Benutzer lokal zur Verfügung gestellt werden.

Der private Schlüssel wird zum besseren Schutz Off-Line gehalten und nur periodisch zur Aktualisierung der Unterschriften eingesetzt. Der private Schlüssel gehört zur Zone und nicht zu einzelnen DNS-Servern, die Kopien der Daten bereitstellen. Somit kann selbst dann noch Authentisierung gewährleistet werden, falls einer oder mehrere Server ihre Vertrauenswürdigkeit verlieren.

Authentisierung von Transaktionen und Anfragen

Die oben beschriebene Authentisierung von DNS-Einträgen gewährleistet zwar die Korrektheit von empfangenen Ressourcen-Datensätzen, bietet aber keinen Schutz für DNS-Anfragen oder Nachrichtenköpfe.
Die Authentisierung von Transaktionen und Anfragen kann folgende Eigenschaften zusichern :

- die Antwort kommt vom kontaktierten DNS-Server

[46] siehe auch Kap. 4.4 „Sicherheitserweiterungen der Netzwerkschicht“

- die Antwort bezieht sich auf die zugehörige Anfrage
- die Antwort wurde während der Übertragung nicht manipuliert

Zur Authentisierung wird der Antwort, die aus der Anfrage und den gefundenen Ressourcen-Datensätzen besteht, vom DNS-Server ein Unterschrifts - RR *(signature-RR, SIG-RR)* hinzugefügt.

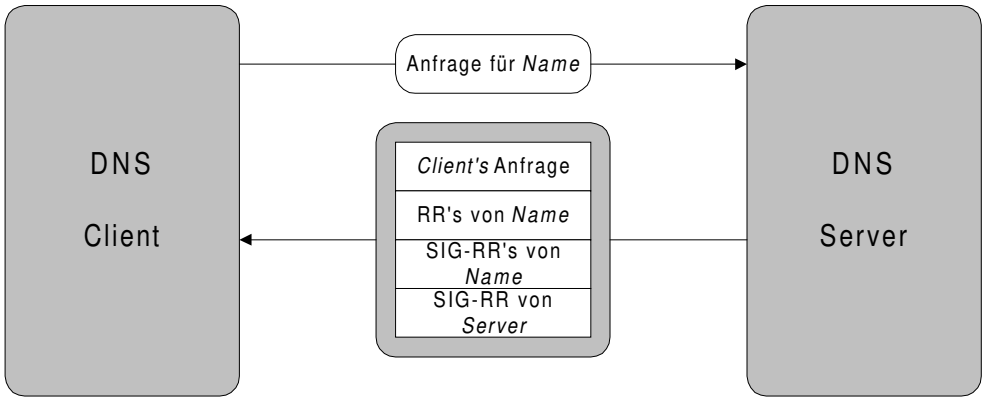

Abbildung 4.3 : Aufbau von Anfrage und Antwort

Derselbe Mechanismus wurde zur Authentisierung von Anfragen spezifiziert, ist aber noch nicht in implementierte DNS-Protokolle integriert.
Die vom Authentisierungsalgorithmus benötigten Schlüssel gehören zum Host und nicht zur Zone. Der öffentliche Schlüssel des Hosts kann über den DNS-Dienst angefordert werden. Da die Unterschrifts-RR anfragespezifisch generiert werden, muß der private Schlüssel online als Software- oder Hardwarelösung zur Verfügung stehen.

Typen von Ressourcen-Datensätzen

Der Unterschrifts-RR beinhaltet die Typen der unterzeichneten Ressourcen-Datensätze, den Namen des Unterzeichners, den Zeitpunkt der Unterzeichnung und der Gültigkeit der Unterschrift, die Lebensdauer der Unterschrift, den benutzten Verschlüsselungsalgorithmus und die aktuelle Unterschrift.
Jedem DNS-Eintrag wird mindestens ein SIG-RR für jeden untergeordneten RR-Typ zugewiesen.

Im Schlüssel-RR (*key resource record ,KEY-RR)* wird der einem Namen zugeordnete Schlüssel gespeichert. Es handelt sich hierbei um einen öffentlichen Schlüssel, da nur diese im DNS gespeichert werden. Der Schlüssel kann einer Zone, einem Host oder einem Benutzer zugeordnet sein. Der KEY-RR wird, genau wie jeder andere RR, durch einen SIG-RR authentisiert. DNS-Implementationen müssen zu jeden einem Namen zugeordneten Typ mindestens zwei gleichzeitig gültige Schlüssel speichern können.

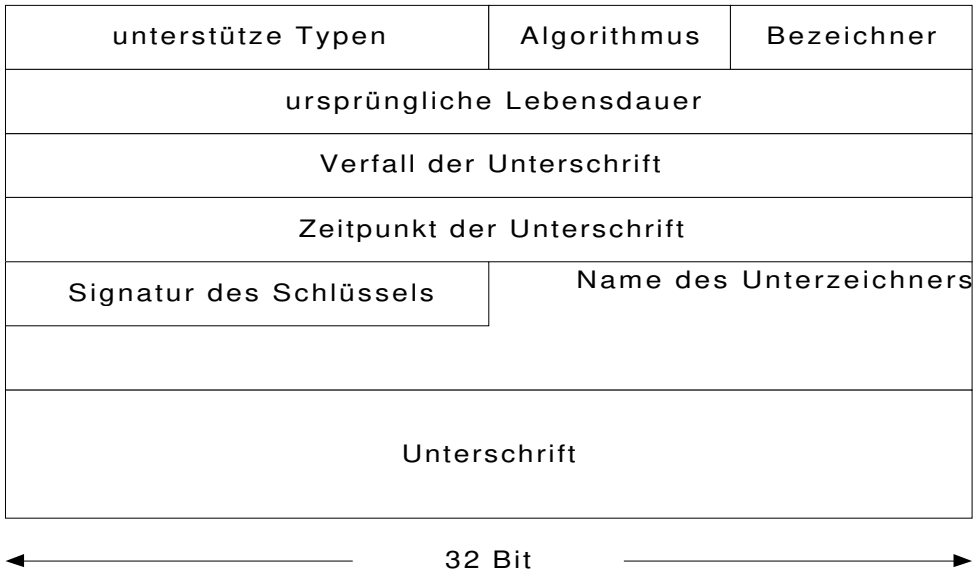

Abbildung 4.4 : Das SIG RDATA Format

Die in einem KEY-RR gespeicherten Daten (KEY RDATA) bestehen aus Kennzeichnungsbits, einem Protokolloktett, der Nummer des Algorithmus und dem öffentlichen Schlüssel. Die Daten haben folgendes Format :

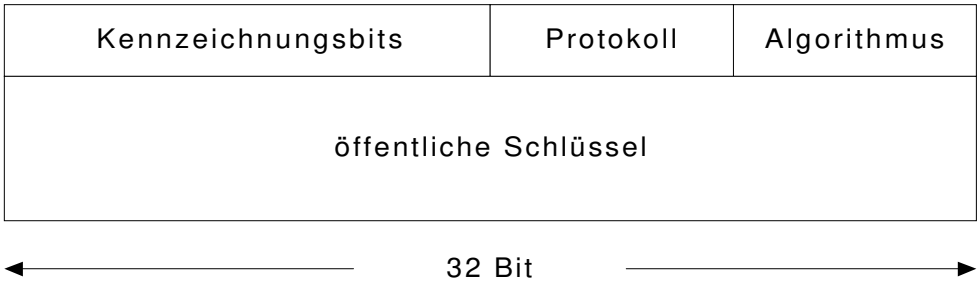

Abbildung 4.5 : Das KEY RDATA Format

Die Kennzeichnungsbits müssen vor weiterer Verwendung der RDATA ausgewertet werden, da sich aus ihnen das Format der restlichen Daten ergibt. Die Bedeutung der Kennzeichenbits wird ausführlich in [EaKa97] beschrieben. Aus den Zuständen läßt sich zum Beispiel ablesen, für welchen Authentisierungsmechanismus der öffentliche Schlüssel verwendet wird und ob der Besitzer eine Zone, ein Host oder ein Benutzer ist. Aus dem Protokoll-Feld und Algorithmus-Feld ergibt sich, mit welchen Internet Protokollen und Verschlüsselungsalgorithmen der öffentliche Schlüssel zusammenarbeitet.

4.3 Sicherheitsprotokolle der Anwendungs- und Darstellungsschicht : Das Secure Socket Layer - *Protokoll*

Das *Secure Socket Layer* - Protokoll *(SSL)* wurde von der Firma Netscape als Sicherheitserweiterung des Web-Browsers *Navigator*[47] entwickelt. Die sichere Übertragung von Daten unter Verwendung diverser Anwendungen[48] ist die Basis einer kommerziellen Nutzung des Internets.

Netscape beschreibt das SSL-Protokoll wie folgt:

> „The SSL Protocol is designed to provide privacy between two communicating applications (a client and a server). Second, the protocol is designed to authenticate the server, and optionally the client.“ [Hick95]

Die aktuelle Version ist SSL 3.0. Sie ist jedoch abwärtskompatibel zur Version 2.0, welche auch noch genutzt wird. Im Anschluß an die Beschreibung der Version 2.0 wird auf die Unterschiede zur Version 3.0 eingegangen. Das Kapitel endet mit einem Überblick über gegenwärtige Implementationen und die zukünftige Entwicklung des Secure Socket Layer - Protokolls.

Dieses Kapitel basiert auf der SSL-Spezifikation [Hick95] und den Implementationsbeschreibungen [HuYo97, BrDa96].

4.3.1 Das Secure Socket Layer - Protokoll Version 2.0

Das SSL-Protokoll kann in Verbindung mit jedem Transportprotokoll genutzt werden und ist somit unabhängig von TCP/IP. Die Nutzung von SSL erfolgt durch unterschiedliche Anwendungsprotokolle wie HTTP, FTP oder Telnet.

Die Authentisierung erfolgt bei SSL durch Zertifikate und dem in Kapitel 4.1.3 vorgestellten asymmetrischen Verfahren unter Nutzung der RSA-Verschlüsselung. Zur Verschlüsselung der Daten einer der symmetrischen Algorithmen RC4, RC2, DES, Triple DES oder IDEA.[49]

Die Verbindungen zwischen SSL, Anwendungen und dem Netzwerk zeigt folgende Abbildung.

[47] Informationen zur Firma Netscape und ihren Produkten sind unter *http://www.netscape.com* erhältlich.
[48] Zu den wichtigsten gehören Web-Browser auf Basis des HyperText-Transfer-Protokolls, sowie die Dateiübertragung mittels des File-Transfer-Protokolls und Telnet-Anwendungen
[49] Referenzen für Algorithmen

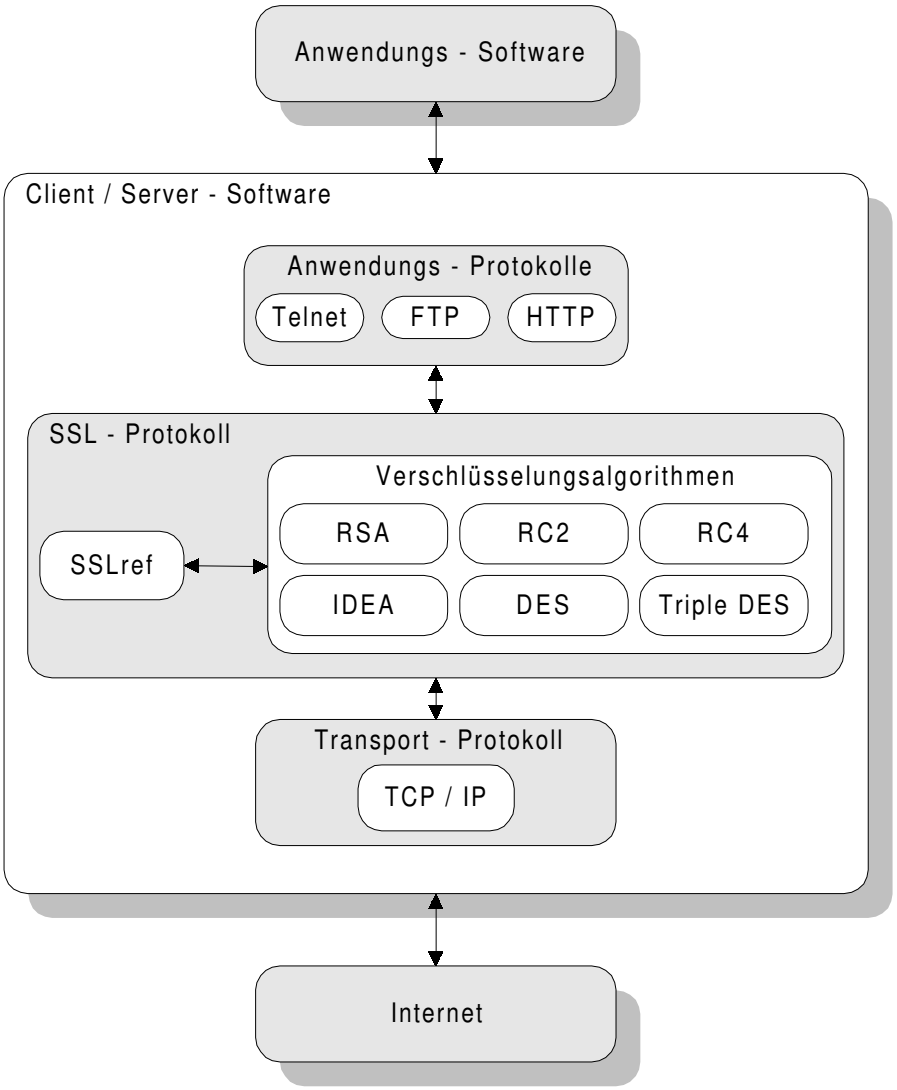

Abbildung 4.6 : Verbindungen zu SSL

Für ein neues Kommunikationsprotokoll ist es wichtig, konform zu einem bestehenden und etablierten Protokollstapel zu sein. Auf diese Weise kann es einfach eingebunden werden oder kann andere Protokolle ersetzen. Die bekannteste Abstraktion eines Protokollstapels sind die sieben Schichten des ISO/OSI-Modells[50].

Das SSL-Protokoll wird innerhalb dieses Schichtenmodells der Anwendungs- und der Darstellungsschicht zugeordnet.

Die Sicherheitsfunktionaliät dieser Schichten wird in [Hals95] wie folgt beschrieben:

„Zusätzlich zur Übertragung von Informationen stellt die *Anwendungsschicht* Dienste zur Verhandlung von Verschlüsselungsmechanismen und zur Authentisierung eines angehenden Kommunikationspartners zur Verfügung."

„Eine weitere Funktion der *Darstellungschicht* betrifft die Datensicherheit. In einigen Anwendungen werden Daten vor der Übertragung mit einem Schlüssel verschlüsselt, welcher nur der empfangenden Darstellungsschicht bekannt ist. Die letztere entschlüsselt die empfangenden Daten unter Nutzung des zugehörigen

[50] ISO/OSIModell

Schlüssels, bevor diese an den Empfänger weitergegeben werden. Dies ist gegenwärtig nicht Bestandteil des Standards."

In der folgenden Abbildung ist die Analogie zwischen dem Aufbau des SSL-Protokolls und der Funktionalität der zuvor beschriebenen OSI-Schichten zu erkennen.

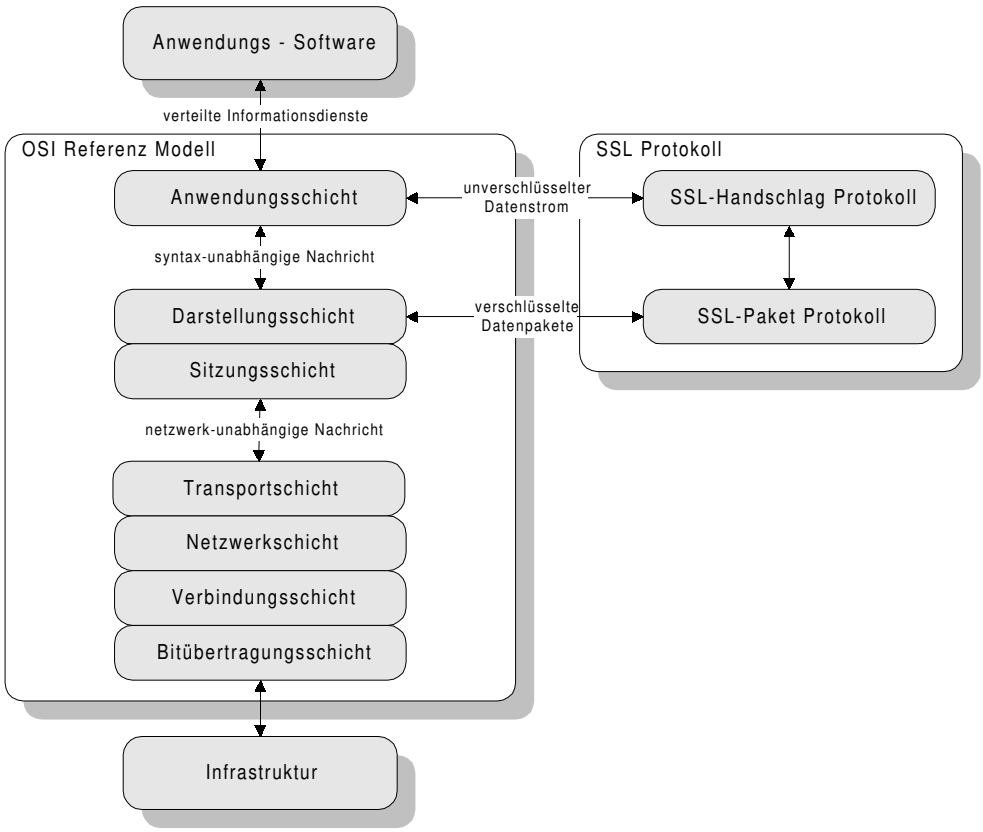

Abbildung 4.7 : Beziehung zwischen SSL und dem OSI-Modell

Das SSL-Protokoll besteht aus den beiden Teilen SSL-Handschlag Protokoll (*SSL-Handshake Protocol, SSLHP)* und SSL-Paket Protokoll (*SSL-Record Protocol SSLRP*).

Das SSLHP handelt den zu benutzenden symmetrischen Verschlüsselungsalgorithmus aus und führt gegebenenfalls die Authentisierung des Clients und/oder Servers durch.

Das SSLRP unterteilt den Datenstrom in Pakete und verschlüsselt diese mit dem ausgehandelten Algorithmus oder empfängt Pakete und entschlüsselt sie.

Wie aus der obigen Abbildung hervorgeht, erfüllen die beiden Protokolle SSLHP und SSLRP die Anforderungen der Anwendungs- und Darstellungsschicht des OSI-Modells. Da in den meisten Protokollen noch keine Sicherheitsfunktionalität implementiert wurde, wird das SSL-Protokoll als Aufsatz solcher Protokolle verwendet und soll nicht als Ersatz dienen.

Das SSL-Record Protokoll

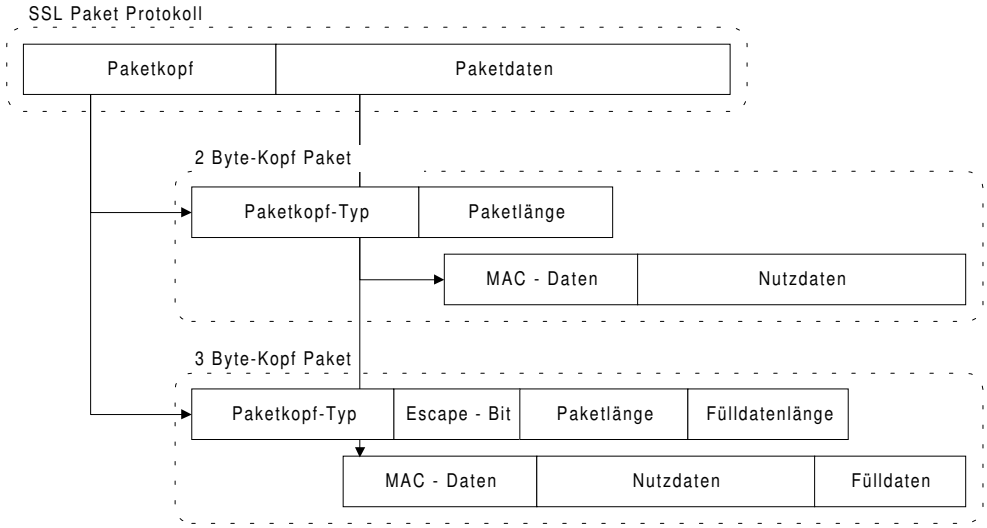

Abbildung 4.8 : Struktur des SSL-Pakets

Ein SSL-Paket besteht aus einem Kopf *(Header)* und den Daten. Die Länge des Kopfes kann 2 Bytes oder 3 Bytes betragen. Die 2-Byte-Version wird genutzt, wenn keine Fülldaten verwendet werden.

Das *Escape*-Bit wird in der Version 2.0 des SSL-Protokolls nicht unterstützt.

Die maximale Länge eines Pakets beträgt bei der 2-Byte-Version 32767 Bytes und bei der 3-Byte-Version 16383 Bytes.

Der Datenteil eines Pakets besteht aus einem MAC[51] , den Nutzdaten und gegebenenfalls Fülldaten. Die Fülldaten werden nur in Verbindung mit Block-Chiffren[52] verwendet. Sie dienen dazu, den Datenteil auf ein Vielfaches der Blockgröße der Chiffre zu verlängern. Falls die Länge des Datenteils bereits ein Vielfaches ist, oder eine Datenstrom-Chiffre verwendet wird, kann die 2-Byte Version des SSL-Pakets genutzt werden.

Der MAC ist eine Prüfsumme, der über dem geheimen Schreibschlüssel des sendenden Teilnehmers, den Nutzdaten, den Fülldaten und einer Sequenznummer in dieser Reihenfolge berechnet wird. Die Sequenznummer ist ein 32 Bit breiter Integer, der nach versenden einer Nachricht erhöht wird.

Das SSL-Handschlag Protokoll

Das SSL-Handschlag Protokoll besteht aus zwei Phasen. Die erste Phase ist für die Auswahl einer Chiffre, den Austausch des Hauptschlüssels (*master key*) und die Authentisierung des Servers zuständig. In der zweiten Phase findet, falls angefordert, die Authentisierung des Clients statt, und der Handschlag wird beendet.

Nach Abschluß des Handschlags werden Nutzdaten zwischen dem Client und Server ausgetauscht. Alle Nachrichten des Handschlags und der Übertragung von Nutzdaten finden über das SSL-Paket Protokoll statt.

[51] siehe auch Kap. 4.1.3 „Authentisierung durch asymmetrische Kryptographie“, Abschnitt „MAC“

[52] Als Chiffre wird der Arbeitsmodus eines Verschlüsselungsalgorithmus bezeichnet.

Der Verlauf des Handschlags ist abhängig von der Nutzung der Client-Authentisierung und der verwendeten Sitzung. Über eine Sitzungs-ID können frühere Sitzungen fortgeführt werden, so daß sich der Authentisierungsprozeß vereinfacht.

Der folgende Protokollverlauf zeigt die Fortführung einer Sitzung mit Authentisierung des Clients.

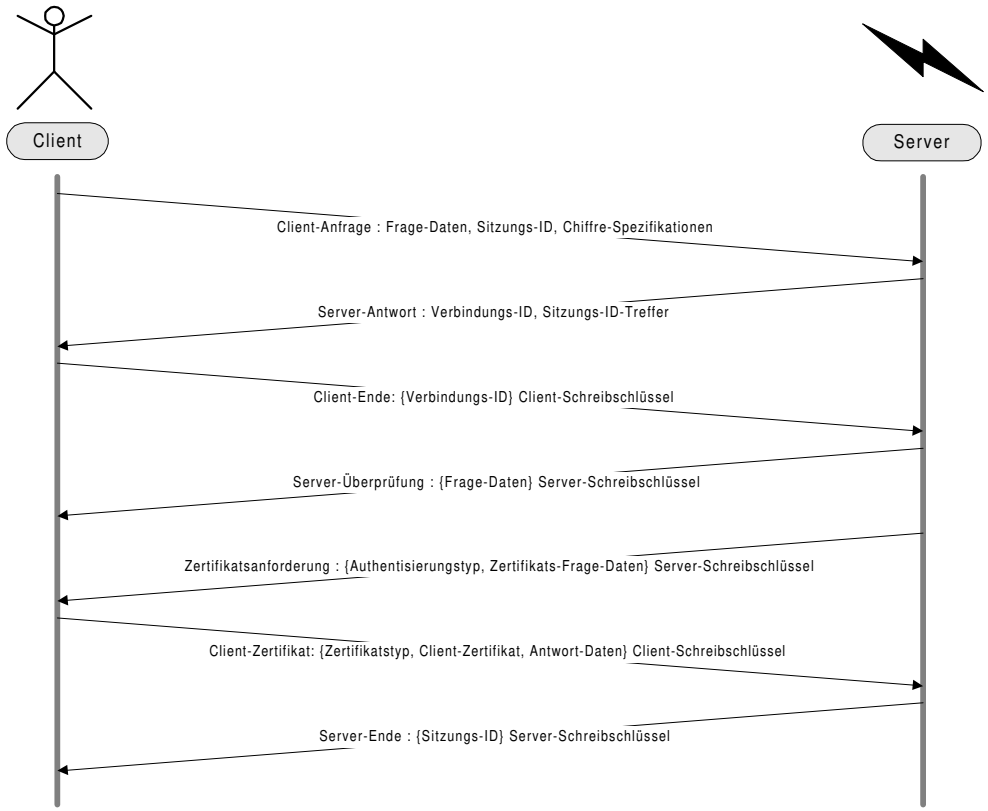

Abbildung 4.9 : Verlauf des SSL-Handschlags

Die *Client-Anfrage* besteht aus Frage-Daten, einer Sitzungs-ID und den Chiffre-Spezifikationen. Die Frage-Daten werden für die spätere Authentisierung des Servers benötigt. Mit der Sitzungs-ID kann der Server den bereits übertragenen Hauptschlüssel und Chiffre-Typ bestimmen. Die Chiffre-Spezifikationen werden nur benötigt, falls der Server die Daten der vorhergehenden Sitzung nicht mehr gespeichert hat. In diesem Fall erfolgt der Handschlag ohne Nutzung der Sitzungs-ID.

Wurde die Sitzungs-ID vom Server gefunden, übergibt die *Server-Antwort* dem Client eine Verbindungs-ID und die boolesche Variable Sitzungs-ID-Treffer mit dem Wert „wahr“.

Aus dem Hauptschlüssel generieren der Client und der Server ein Schlüsselpaar mit folgenden Eigenschaften :

Client-Schreibschlüssel = Server-Leseschlüssel
Client-Leseschlüssel = Server-Schreibschlüssel

Ab diesem Zeitpunkt werden zwischen dem Client und Server nur noch verschlüsselte Nachrichten ausgetauscht. Der Client verschlüsselt eine Nachricht mit seinem Schreibschlüssel, die der Server mit seinem Leseschlüssel entschlüsseln kann und umgekehrt.

Die *Client-Ende* Nachricht besteht aus der Verbindungs-ID, die vom Server übertragen wurde. Diese wird mit dem Client-Schreibschlüssel verschlüsselt. Die Verbindungs-ID wird im weiteren Protokollverlauf als Zufallswert verwendet, der zur Abwehr diverser Angriffsarten dient[53].

Die *Server-Überprüfung* -Nachricht beinhaltet die ursprünglich vom Client übertragenen Frage-Daten. Der Empfang und die Entschlüsselnd dieser Nachricht durch den Client ist der letzte Schritt der Server-Authentisierung, da nur der echte Server den privaten Schlüssel zur Entschlüsselung des Hauptschlüssels besitzt.

Die *Zertifikatsanforderung* wird vom Server gesendet. Sie besteht aus dem Authentisierungstyp und den Zertifikats-Frage-Daten. Der Authentisierungstyp bestimmt eine *Message-Digest*-Funktion und einen Verschlüsselungsalgorithmus. Die Zertifikats-Frage-Daten werden vom Server generiert und vom Client unterzeichnet, indem er das Ergebnis einer Hash-Funktion über den Frage-Daten mit seinem privaten Schlüssel verschlüsselt.

Die *Client-Zertifikat* Nachricht überträgt den Zertifikatstyp, das Client-Zertifikat und die Antwort-Daten. Zur Authentisierung des Clients entschlüsselt der Server die Anwort-Daten mit dem öffentlichen Schlüssels des Client, den er dem Zertifikat entnimmt. Die Antwort-Daten vergleicht er mit seinem Ergebnis der Hash-Funktion.

Mit der *Server-Ende* Nachricht wird die Handschlag-Phase beendet. Die Nachricht enthält die Sitzungs-ID.

Während des Handschlag-Protokolls können auch Fehlermeldungen übertragen werden, deren Ursache entweder behoben werden kann oder zum Abbruch der Kommunikation führt.

Nach Abschluß des Handschlag-Protokolls erwarten die Parteien Pakete mit Anwendungsdaten.

Stärken des SSL-Protokolls

Brute-Force Angriffen kann das SSL-Protokoll widerstehen, da die eingesetzten Chiffren mit Schlüsselbreiten arbeiten, die ein Durchsuchen des Schlüsselraums, selbst mit zukünftiger Technologie, unmöglich machen.

Wiederholungsangriffe sind erfolglos, da vom SSL-Server eine 128 Bit breite Verbindungs-ID generiert wird, und diese dem Client verschlüsselt mitgeteilt wird. Eine wiederholte Nachricht würde also vom Client oder Server ignoriert werden, da sie keine gültige Verbindungs-ID besitzt.

Der *Mann-in-der-Mitte-Angriff* scheitert an der asymmetrischen Verschlüsselung. Der vom Client übertragene Hauptschlüssel kann nur mit dem privaten Schlüssel des Servers entschlüsselt werden. Eine dritte Partei müßte also ein Zertifikat fälschen und im Besitz des privaten Schlüssels einer Zertifikats-Behörde sein. Dieser Schlüssel kann wiederum nur durch einen Brute-Force-Angriff gegen die Zertifikatsbehörde erlangt werden.

Schwächen des SSL-Protokolls

[53] siehe auch Abschnitt „Stärken des SSL-Protokolls“

Der offensichtlichste Schwachpunkt des Protokolls sind *Brute-Force-Angriffe gegen schwache Chiffren*. Zu den schwachen Chiffren gehören alle Algorithmen, die Schlüssel mit geringer Bitbreite verwenden. Die Exportgesetze der USA erforderten die Implementation solcher Algorithmen, da die stärksten Chiffren nur innerhalb der USA genutzt werden dürfen.

Die fehlende *Neuverhandlung von Sitzungsschlüsseln* ist ein ernstzunehmender Schwachpunkt, falls das SSL-Protokoll in Verbindung mit langfristig kommunizierenden Anwendungen[54] eingesetzt wird. Der Hauptschlüssel und das generierte Schlüsselpaar sind für die Dauer der Verbindung gültig und erleichtern Dritten Angriffe auf die Verschlüsselung. Sinnvoller wäre es, die Schlüssel in bestimmten Intervallen auszutauschen.

4.3.2 Modifikationen der SSL Version 3.0

Die aktuelle Version 3.0 des SSL Protokolls ist abwärtskompatibel zum Vorgänger, wurde aber in den folgenden Punkten erweitert und verbessert.

- Es werden weniger Handschlag-Nachrichten ausgetauscht, um den Prozeß zu beschleunigen.
- Es werden zusätzliche Verschlüsselungsalgorithmen und Verfahren unterstützt.
- Die Verwendung von Chipkarten zur Authentisierung wurde vorgesehen.[55]
- Die Verhandlung von Client-Zertifikaten, die vom Server akzeptiert werden, wurde vereinfacht.

4.3.3 Implementationen des SSL-Protokolls

Netscape Produkte

Die Netscape-Produktpalette gibt es in mehreren Versionen, da einige Chiffren Exportbeschränkungen unterliegen. Die frei erhältlichen, internationalen Versionen des Netscape Navigator verwenden überhaupt keine starken Chiffren. Die Netscape-Software unterstützt das SSL-Protokoll in den Versionen 2.0 und 3.0 .

SSLref

SSLref ist die Referenzimplementation des SSL Protokolls von Netscape. Sie soll Anwendern ermöglichen Programme zu entwickeln, die das SSL Protokoll unterstützen. Eine kommerzielle Nutzung dieser Anwendungen ist allerdings nicht erlaubt. Außerdem greift wieder die Exportbeschränkung, so daß SSLref nur innerhalb der USA vertrieben werden darf.

SSLeay

SSLeay ist eine frei erhältliche Implementation des SSL Protokolls. Sie wurde vom australischen Kryptographen Eric Young implementiert, weicht allerdings in einigen Punkten von der Referenzimplementation ab.

[54] zum Beispiel FTP oder Telnet
[55] das verwendete Verfahren heißt *Fortezza*

4.4 Sicherheitserweiterungen der Netzwerkschicht : Das Internet Protokoll Version 6

Die Informationen zur Spezifikation und Anwendung des Internet Protokolls Version entstammen den Dokumenten [MeSi95, DeHi95, HiDe95, Atki95a+b+c, Hind95].

Es gibt eine Vielzahl von Ursachen die dazu führen, daß ein neues, leistungsfähigeres Internet-Protokoll benötigt wird. Aus diesen ergeben sich die Anforderungen an die Spezifikation des *Internet-Protokolls Version 6 (IPv6)*[56]
Zu den erheblichen Problemen des aktuellen Internet Protokolls zählt die unzureichende Größe des Adressraums und die schlechte Skalierbarkeit der Bandbreite. Die wichtigste Anforderung an das neue Protokoll ist die Abwärtskompatibilität zu IPv4.

Aus dem ungeheuren Wachstum des Internets während der letzten Jahre ergeben sich Probleme im Bereich der Adressierung und des Routings. Eine Hierarchie von Domänen mit weltweit eindeutigen Internetadressen läßt sich auf Dauer mit IPv4 nicht realisieren. Das Internet wächst jedes Jahr um den Faktor 2, obwohl es bislang hauptsächlich vom Computermarkt genutzt wird und die Erschließung neuer Bereiche (z.B. Einbindung von Consumer Electronics bis hin zur Steuerung von Gebäudeelektrik/-elektronik) diesen Faktor noch vergrößern wird. Außerdem werden tragbare Computer mangels skalierbarer Protokolle und leistungsfähigen Übertragungsmedien bislang nur sporadisch an Netzwerke gebunden. Sollten Protokolle Verbindungen über beliebige Medien aufbauen und diese effizient nutzen können, wird sich der Anteil tragbarer Computer im Internet drastisch vergrößern und es werden zuverlässige Mechanismen zur Authentisierung und Verschlüsselung benötigt.

4.4.1 Adressformate

Das IPv6 ist der Nachfolger des IPv4 und somit abwärtskompatibel. Änderungen wurden nur in notwendigen Bereichen durchgeführt.

Die Adresslänge wurde von 32 Bit im IPv4 auf 128 Bit erhöht, um eine vielschichtigere Adresshierarchie und einen ausreichend großen Adressraum realisieren zu können.

3 Bits	n Bits	m Bits	o Bits	p Bits	o-p Bits
010	Registrierungs ID	Dienstleister ID	Teilnehmer ID	Subnetz ID	Schnittstellen ID

Abbildung 4.10 : Format einer IPv6 - Adresse

Die Felder der Adresse haben folgende Bedeutung :

- Die Bitfolge *'010'* kennzeichnet die Adresse als einzelne Dienstleister-Nachricht *(provider-unicast)*[57].
- Die *Registrierungs ID* identifiziert das Internet-Adressen-Register, welches Adressraum an die Dienstleister vergibt.
- Die *Dienstleister ID* bestimmt den zuständigen Dienstleister, der Adressen an seine Teilnehmer vergeben darf.

[56] auch als IPng (next generation-).bezeichnet
[57] Zur Beschreibung weiterer Adressformate siehe [HiDe95].

- Die *Teilnehmer ID* verweist auf Kunden von Dienstleistern.
- Eine *Subnetz ID* ist die physikalische Adresse eine Sub-Netzes. Auf einer Adresse können mehrere Sub-Netze liegen.
- Die *Schnittstellen ID* referenziert eine Schnittstelle aus der Menge, die sich aus dem Subnetz-Prefix ergibt.

4.4.2 Sicherheitserweiterungen

Das Protokoll beinhaltet Erweiterungen des Headers, die Authentisierung, Datenintegrität und Vertraulichkeit gewährleisten. Diese Erweiterungen werden in allen Versionen des IPv6 implementiert sein.

Authentication-Header

Der erste Mechanismus ist der *Authentication Header (AH)* und gewährleistet IPv6-Datagrammen Authentisierung und Integrität. Diese Erweiterung ist unabhängig von speziellen Algorithmen und unterstützt unterschiedliche Authentisierungsmechanismen. Um die weltweite Verwendung im Internet garantieren zu können, ist die Verwendung des *Keyed MD5*[58] vorgesehen.

Der AH unterstützt die sichere Kommunikation zwischen

- **Hosts**
  alle Hosts müssen in diesem Fall den AH implementieren
- **Gateways**
  dient zum Aufbau eines privaten virtuellen Netzes entlang eines nicht vertrauenswürdigen Backbones (z.B. des Internet)
- **Hosts und Gateways**
  die Gateways dienen in einem sicheren Subnetz als Sicherheitsgateway.

Ein *Sicherheitsgateway* ist ein System, das zwischen externen, nicht vertrauenswürdigen Systemen und dem internen, vertrauenswürdigen Subnetz als Kommunikationsgateway dient. Die internen Hosts können zur Kommunikation mit externen Systemen die Sicherheitsdienste des Gateways in Anspruch nehmen. In einem vertrauenswürdigen Subnetz muß sichergestellt sein, daß die Hosts und Router keinen aktiven oder passiven Angriffen ausgesetzt sind und das Übertragungsmedium (z.B: Ethernet) nicht angegriffen wird.
Da interne Systeme nur über das Gateway externe Verbindungen nutzen können, ist es ausreichend den AH auf dem Gateway zu implementieren und den internen Systemen diesen Dienst bei Bedarf zur Verfügung zu stellen. Die Anforderung geschieht über ein *Sensitivity Label*, welches das Gateway auswertet und entsprechende Sicherheitsmechanismen aktiviert.

Der AH beinhaltet Authentisierungsinformationen für sein IP-Datagramm. Diese sind Ergebnis einer kryptographischen Authentisierungsfunktion, die als Eingabe das IP-Datagramm und einen geheimen Authentisierungsschlüssel erhält. Die Berechnung der Authentisierungsdaten des IP-Datagramms erfolgt vor einer eventuellen Fragmentierung und somit auf Empfängerseite die Prüfung der Authentisierungsdaten erst nach Erhalt aller Pakete. Es gibt einige Felder in einem IP-Datagramm, die sich während der Übertragung ändern. Dazu gehört zum Beispiel das Sprunglimit *(hop limit)*, welches die Zahl von Sprüngen angibt, nach denen ein

[58] weiterführend siehe [MeSi95]

Paket vernichtet werden muß. Diese Felder werden von der Berechnung der Authentisierungsdaten ausgenommen.

Der AH steht nach allen Köpfen, die bei jedem Sprung untersucht werden und vor den Köpfen, die zwischendurch nicht untersucht werden. Bei Verwendung von IPv6 befindet sich der AH normalerweise zwischen dem Sprung-zu-Sprung-Kopf *(Hop-by-Hop-Header)* und den IPv6 Zieloptionen.

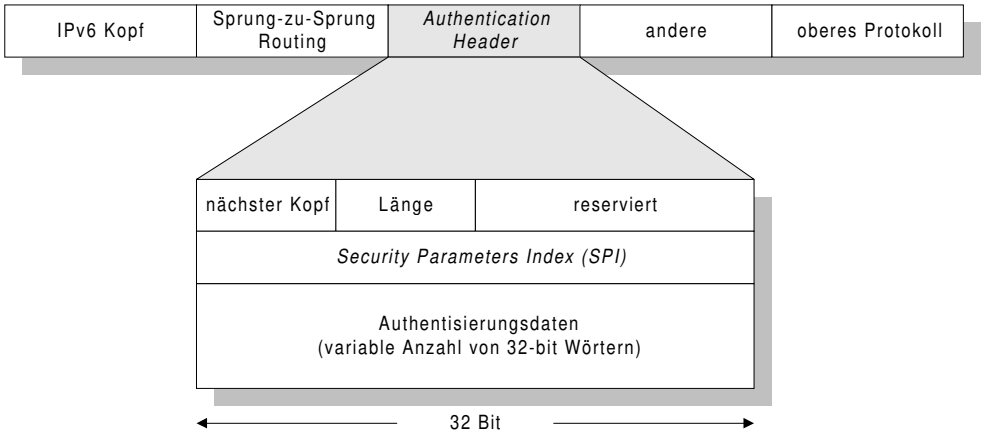

Abbildung 4.11 : Format des *Authentication Header*

Die Felder des *Authentication Header* haben folgende Bedeutung :

- **Nächster Kopf (8 Bit)**
  Dieser Wert identifiziert die Nutzlast (*payload*), die den Authentisierungsdaten folgt.

- **Nutzlastlänge (8 Bit)**
  Die Länge des Authentisierungsdaten-Feldes in 32-bit Worten.

- **Reserviert (16 Bit)**
  Dieses Feld ist für zukünftigen Gebrauch reserviert.

- ***Security Parameters Index* (32 Bit)**
  Der Wert identifiziert den SPI des Datagramms. Der Wert '0' bedeutet, daß kein SPI ausgewählt wurde. Die Werte '1' bis '255' sind von der IANA[59] für zukünftigen Gebrauch reserviert worden.

- **Authentisierungsdaten (n * 32 Bit)**
  Die Länge des Feldes ist variabel, aber immer ein Vielfaches von 32 Bit. Die Nutzung des Feldes wird von dem zugeordneten Sicherheitsverband *(security association)* vorgegeben und ist für alle Datagramme mit gleicher Zieladresse und SPI identisch.

Encapsulating Security Payload

Die zweite Sicherheitserweiterung ist der *Encapsulating Security Payload (ESP)* Kopf. Er stellt dem IP-Datagrammdienst Mechanismen zur Gewährleistung von Integrität und Vertraulichkeit zur Verfügung. Das Verfahren ist einfacher als andere Sicherheitsprotokolle, bietet dafür aber Flexibilität und Unabhängigkeit, da es mit mehreren Algorithmen arbeiten kann. Als Standardalgorithmus wird der *Data Encryption Standard (DES)* verwendet.

Das ESP kann in zwei unterschiedlichen Modi arbeiten. Der erste Modus nennt sich Tunnelmodus und kapselt das gesamte IP-Datagramm innerhalb eines ESP-Kopfes.
Der zweite Modus ist der Transportmodus. Er kapselt nur die Daten der oberen Protokollschicht (z.B. TCP oder UDP), verschlüsselt den Großteil des ESP-Inhalts und versieht das Paket mit einem neuen, unverschlüsselten IP-Kopf, der für die Übertragung der geschützten Daten im Internet zuständig ist.

Der Kopf des ESP kann überall zwischen dem IP-Kopf und dem abschließenden Transportschicht-Protokoll eingefügt werden. Der ESP besteht aus einem unverschlüsselten Kopf gefolgt von verschlüsselten Daten. Die verschlüsselten Daten enthalten die geschützten Felder des ESP und die geschützten Benutzerdaten, welche entweder ein gesamtes IP-Datagramm oder ein Rahmen einer oberen Protokollschicht sein können.

[59] **I**nternet **A**ssigned **N**umbers **A**uthority

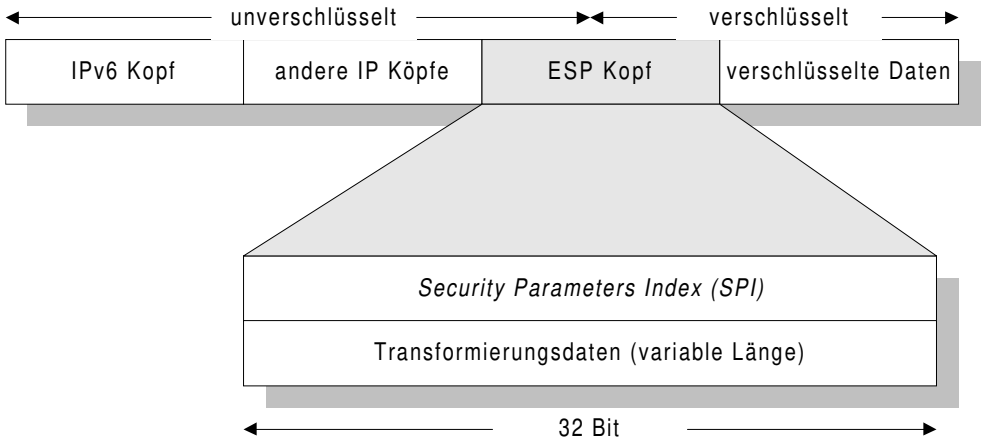

Abbildung 4.12 : Format des ESP - Kopfes

Die Algorithmen zur Authentisierung und Verschlüsselung zusammen mit dem Format der zugehörigen Transformierungsdaten werden als Transformierungen (*transforms)* bezeichnet. Das ESP-Format erlaubt es neue Transformierungen zu unterstützen, die sich aus der Verwendung anderer Verschlüsselungsalgorithmen ergeben können.

Konzept des Sicherheitsverbandes *(security association)*

Das Konzept des Sicherheitsverbandes gehört zur Basis der AH- und ESP-Dienste. Ein Sicherheitsverband ist die Summe der Sicherheitsinformationen, die einer einzelnen oder einer Menge von Netzwerkverbindungen zugeordnet sind. Sie besteht aus dem SPI und einer Zieladresse. Der SPI ist ein unstrukturierter Index von Sicherheitsinformationen. Zu den Parametern, die jeder Sicherheitsverband enthalten muß, gehören die Algorithmen, Modi und Schlüssel, die zur Authentisierung und Verschlüsselung benötigt werden. Es können noch zusätzliche Parameter angegeben werden, die aber nicht von allen Implementierungen ausgewertet werden müssen.

Der sendende Host benutzt die Benutzer-ID des Absenders und die Zieladresse um einen geeigneten Sicherheitsverband zu finden (und somit auch SPI - Werte). Der empfangende Host ermittelt aus den SPI-Werten und der Zieladresse den korrekten Sicherheitsverband. Eine AH-Implementation wird somit immer in der Lage sein, für alle korrekten eintreffenden Pakete, aus dem SPI und der Zieladresse auf den Sicherheitsverband und damit verbundene Sicherheitsinformationen zu schließen.
Ein Sicherheitsverband ist nur für eine Richtung gültig, so daß eine Verbindung zwischen zwei Hosts zwei SPI's nutzt.

Leistungsauswirkungen der Sicherheitserweiterungen

Die Nutzung des AH verringert die Leistung eines Systems, da für jedes IP-Datagramm mit AH vom Sender und Empfänger Authentisierungsdaten berechnet werden müssen und vom Empfänger zusätzlich noch der Vergleich mit den übertragenen Daten erfolgen muß.

Die Verwendung des ESP kann die Leistung von Systemen nachhaltig beeinflussen, weil die Verschlüsselung der Daten mehr Rechenleistung und Zeit erfordert. Im Tunnelmodus verringert sich gegenüber dem Transportmodus die Leistung weiter, da das gesamte IP-Paket inklusive des IP-Kopfes ver- und entschlüsselt werden muß. Die genauen Auswirkungen auf die Systemleistung ergeben sich aus den eingesetzten Algorithmen, den Schlüsselgrößen und anderen Faktoren. Bei

hohem Datenaufkommen ist eine Hardwareimplementierung des Algorithmus sinnvoll.

Kombination von Sicherheitsmechanismen

Der AH kann mit ESP kombiniert werden, um die erwünschten Sicherheitseigenschaften zu erhalten. Der AH gewährleistet immer Integrität und Authentisierung und kann die Eigenschaft *non-repudiation* gewährleisten, falls ein passender Authentisierungsalgorithmus eingesetzt wird (z.B: RSA). Der ESP gewährleistet immer Integrität und Vertraulichkeit und kann mit entsprechenden Algorithmen auch Authentisierung gewährleisten. Durch hinzufügen des AH zu einem IP-Datagramm und anschließender Verschlüsselung durch ESP ergibt sich ein besonders starker Sicherheitsmechanismus.

4.4.3 Schlüsselverwaltung[60]

Das Protokoll zur Schlüsselverwaltung ist an den AH und ESP nur über den SPI gekoppelt, so daß mehrere unterschiedliche Systeme zur Schlüsselverwaltung eingesetzt werden können. Ein Internet-Standard ist noch nicht definiert.

Schlüsselverwaltungssysteme die ihre Daten in der IP-Schicht halten sollen nicht unterstützt werden, sondern statt dessen Systeme, die mit einer der oberen Schichten (TCP oder UDP) zusammenarbeiten. Auf diese Weise wird das Schüsselverwaltungssystem strikt von den anderen Sicherheitsmechanismen getrennt und kann somit jederzeit durch andere, verbesserte Systeme ersetzt werden, ohne die Implementationen der übrigen Protokolle modifizieren zu müssen.

4.4.4 Authentisierung zwischen Benutzer und Host

Es gibt mehrere Möglichkeiten, um Benutzer gegenüber einem Host zu authentisieren. Bei Fern- oder Netzwerkzugriffen gibt es zwei Gefahrenquellen : ein Eindringling kann per Lauschangriff ID's und Paßwörter erhalten und diese später für einen Wiederholungsangriff nutzen, oder er kann aus der Form bestehender Paßwörter auf weitere schließen (Wörterbuchangriff).

Die meisten Systeme benutzen gegenwärtig unverschlüsselte Paßwörter, die über Netzwerke versandt werden (z.B. *telnet* oder *rlogin*). Sie bieten somit keinen ausreichenden Schutz vor Angriffen.

Verteidigung durch Umgrenzung

In einem umgrenzten System muß sich der Benutzer zunächst gegenüber einem externen Host authentisieren, bevor er Zugang zum internen System erlangt. Der externe Host kann zum Beispiel eine Feuerwand *(firewall)*[61] sein, die nichtenthüllende Paßwörter zur Authentisierung benutzt. Der Zugriffsschutz auf das interne System kann dann schwächer implementiert werden. Das Problem der Authentisierung wird auf diese Weise in zwei einfacher zu handhabende Teilprobleme zerlegt.

Die Verteidigung durch Feuerwände hat allerdings auch Nachteile, so daß sie nur als kurzfristige Lösung angesehen werden sollte. Das Gateway ist auf dem IP-Level nicht transparent, es müssen also alle Dienste voneinander unabhängig be-

[60] siehe auch Kap. 4.1.4 „Schlüsselverteilung und Verwaltung"
[61] zum Aufbau und Funktionsweise von Firewalls siehe [ChBe94]

handelt werden. Für Maschine-zu-Maschine-Kommunikation ist doppelte Authentisierung im allgemeinen nur schwer zu realisieren. Die im Internet üblichen Ende-zu-Ende-Protokolle könnten leicht unterbrochen werden.

Das Feuerwand-Gateway muß sehr stark gegen Eindringlinge gesichert sein, da das interne System nur geringen Schutz bietet. Benutzt das interne System unverschlüsselte Paßwörter, kann ein Angreifer, der die Feuerwand überwunden hat, durch Abhören in den Besitz aller Paßwörter des Systems gelangen.

Ein Vorteil von durch Feuerwände geschützten System ist, daß nur eine geringe Anzahl der Rechner externe Verbindungen aufbauen kann und nur diese Angriffen ausgesetzt sind. Allerdings ist die Realisierung einer Feuerwand aufwendig und vollständiger Schutz kann nicht garantiert werden. Die Konfiguration einer Feuerwand ist kompliziert, da einige Netzwerkdienste benötigt werden, andere hingegen durch Filterung der IP-Pakete unterbunden werden müssen. Dieser Filteraufwand kann dazu führen, daß die Feuerwand zum Flaschenhals in der Kommunikation wird und die Systemleistung verringert.

Kapitel 5

Zusammenfassung und Ausblick

Die Untersuchung von Sicherheitsrisiken und Sicherheitsmechanismen im Internet hat gezeigt, daß es zahlreiche Angriffsmöglichkeiten gegen Protokolle und Anwendungen gibt, denen Sicherheitsmechanismen gegenüberstehen, die sich größtenteils noch in der Entwicklungs- und Verbesserungsphase befinden.

Im Internet nutzbare Sicherheitsmechanismen lassen sich unter Betrachtung des Entwicklungsverlaufs in kommerzielle Implementationen und Spezifikationen der Internet-Gremien unterteilen.

Kommerzielle Implementationen

Es werden Mechanismen von Firmen entwickelt, die sich im Internet engagieren und Sicherheitsfunktionalität benötigen, die von Internetgremien noch nicht für die Internetarchitektur vorgesehen wurde. Zu dieser Gruppe gehört das SSL-Protokoll der Firma Netscape[62].

Der Vorteil dieser Verfahren ist es, daß frühzeitig Implementationen vorliegen und die Weiterentwicklung vorangetrieben wird, da ein kommerzieller Nutzen vorhanden ist.

Als Nachteil erweist sich die mangelnde Kompatibilität und Integrierbarkeit in die Internetarchitektur, da diese Mechanismen an den Bedarf der entwickelnden Firmen anpaßt werden.
Sind diese Produkt kommerziell erfolgreich und finden weite Verbreitung im Internet, werden sie in der Regel den Internet-Gremien als Standardisierungsvorschlag eingereicht[63]. Eine Voraussetzung zur Erlangung des Standard-Status ist die erfolgreiche Anwendung mehrerer unabhängiger Implementationen der Spezifikation[64].
Hier liegt wiederum ein Vorteil dieser Mechanismen, da schon Implementationen im Internet genutzt werden und Erfahrungswerte vorliegen.

Spezifikationen der Internet-Gremien

Den Internet-Gremien vorgelegte Standardisierungsvorschläge bestehen in der Regel aus Spezifikationen und wurden noch nicht implementiert. Da die Standar-

[62] siehe auch Kap. 4.3.1 „Das Secure Socket Layer - Protokoll Version 2.0“

[63] Das von Netscape eingereichte SSL-Protokoll wurde in einer modifizierten Form als *Transport Layer Security* - Protokoll von den Internet-Gremien zur Standardisierung vorgesehen.

[64] siehe auch Anhang Anhang C „Der Standardisierungsprozeß der Internet-Organisationen“

disierung jedoch, wie im Anhang beschrieben, auf Implementationen basiert, kann sich dieser Prozeß über einen längeren Zeitraum hinziehen. Eventuell wird eine Spezifikation schon während der Standardisierungsphase hinfällig, da von Firmen proprietäre Verfahren eingesetzt werden, und die weite Verbreitung eine Substitution verhindert.

Lücken in Sicherheitsmechanismen

Die Entwicklung und kommerzielle Nutzung von Sicherheitmechanismen im Internet befindet sich noch im Anfangsstadium. Dies läßt sich aus den vielen Meldungen von Sicherheitslücken ersehen, die der Internet-Gemeinde verdeutlichen, daß es noch keine vollständige Sicherheit gibt. Die meisten dieser Lücken basieren allerdings auf der Implementation und nicht auf der Spezifikation, und erfordern somit die ständige Weiterentwicklung der Mechanismen.

Zukünftige Entwicklung

Die zukünftigen Aufgaben liegen im Bereich der Verbesserung und Integration bestehender Sicherheitsmechanismen und der Betrachtung von Sicherheitsaspekten für die bislang keine Standardisierungsvorschläge ausgearbeitet wurden.

Ein wesentlicher Bereich ist die Verteilung von Schlüsseln und die Verwaltung von Zertifikaten. Hier existieren bislang lediglich einige Dokumente im Entwurfsstadium und mehrere kommerzielle, untereinander teilweise inkompatible, Lösungen.

Der größte Entwicklungsbereich ist zur Zeit die Abwicklung von Geldgeschäften über das Internet. Hierzu gehören sowohl die Bezahlung mittels Kreditkarte oder einer digitalen Währung, als auch die Erledigung sämtlicher Bankgeschäfte.

Zur Gewährleistung sicherer Transaktionen werden von den meisten Finanzinstituten proprietäre Lösungen bevorzugt, die teilweise auf Hardwareerweiterungen basieren.
In diesem Bereich wird sich in nächster Zeit ein kommerzieller Standard etablieren, der nicht von Internet-Gremien unterstützt wird.

Anhang A

Literaturverzeichnis[65]

[Amor94] E. Amoroso. *Fundamentals of Computer Security Technology*. Prentice Hall, 1994.

[Atki95a] R. Atkinson. *Security Architecture for the Internet Protocol*. Network Working Group, RFC 1825 (Standards Track), Aug. 1995.

[Atki95b] R. Atkinson. *IP Authentication Header*. Network Working Group, RFC 1826 (Standards Track), Aug. 1995.

[Atki95c] R. Atkinson. *IP Encapsulating Security Payload*. Network Working Group, RFC 1827 (Standards Track), Aug. 1995.

[BaKi] P. Barker and S. Kille. *The COSINE and Internet X.500 Schema*. RFC 1274.

[BaRi96] R. Baldwin, R. Rivest. *The RC5, RC5-CBC, RC5-CBC-Pad and RC5-CTS Algorithms*. Network Working Group, RFC 2040 (Informational), Okt. 1996.

[Blak96] B. Blakley. *Architecture for Public-Key Infrastructure*. IETF PKIX Working Group, Internet-Draft, Nov. 1996.

[Brad95] Jeremy Bradley. *Brute Force Attacks and Continual Technological Revolution.* University of Bristol, Department of Computer Science, Sep.1995.

[Brad96] S. Bradner. *The Internet Standards Process - Revision 3.* Network Working Group, RFC 2026 (Best Current Practice), Okt. 1996.

[BrDa96] Jeremy Bradley, Neil Davies. *The SSLP Reference Implementation Project.* University of Bristol, Department of Computer Science, 1996.

[CC88a] CCITT. *Recommendation X.509*. CCITT Blue Book, Volume VIII, Nov. 1988.

[CC88b] CCITT. *The Directory: Overview of Concepts, Models and Service*, *Recommendation X.509*. CCITT Blue Book, Nov. 1988.

[CFM+95] Silvana Castano, Mariagraza Fugini, Giancarlo Martella, Pierangela Smarati. *Database Security*. Addison-Wesley, 1995.

[ChBe94] William R. Cheswick, Steven M. Bellovin. *Firewalls and Internet Security.* AT&T Bell Laboratories, Inc., 1994.

[DeHi95] S. Deering and R. Hinden. *Internet Protocol Version 6 Specification*. Network Working Group, RFC 1883 (Standards Track), Dez. 1995.

[Denn82] Dorothy E. Denning. *Cryptography and Data Security*. Addison-Wesley, 1982.

[Diff88] Whitfield Diffie. *The first ten years of public key cryptography*. Proceedings of the IEEE, Mai 1988.

[DiHe76] Whitfield Diffie and Martin E. Hellman. *New Directions in cryptography.* IEEE Transactions on Information Theory, IT-11, Nov.1976.

[EaKa97] D. Eastlake and C. Kaufman. *Domain Name System Security Extensions.* Network Working Group, RFC 2065 (Standards Track), Jan. 1997.

[Grund97] Pia Grund-Ludwig. *Die heimlichen Machthaber im Internet*. CHIP -

[65] Der Status von *Request for Comments* ist in Klammern hinter der Nummer angegeben.

Magazin, Vogel-Verlag, Würzburg, April 1997.

[HaAt94] N. Haller and R. Atkinson. *On Internet Authentication*. Network Working Group, RFC 1704 (Informational), Okt. 1994.

[Hall94] N. Haller. *The S/Key One-time Password System*. Proceedings of the Symposium on Network & Disrtibuted Systems Security, Internet Society, Feb. 1994.

[Hals95] Fred Halsall. *Data Communications, Computer Networks and Open Systems*. Addison-Wesley, 1993.

[Hick95] Kipp Hickman. *The SSL Protocol (Version 2).* Netscape Communications Corporation, Feb. 1995.

[HiDe95] R. Hinden, S. Deering. *IP Version 6 Addressing Architecture*. Network Working Group, RFC 1884 (Standards Track), Dez. 1995.

[Hind95] Robert M. Hinden. *Internet Protocol Version 6 Overview*. SunSoft Playground, Mai 1995.

[HuCr94] E. Huizer, D. Crocker. *IETF Working Group Guidelines and Procedures*. Network Working Group, RFC 1603 (Informational), März 1994.

[HuYo97] T.J. Hudson, E.A. Young. *SSLeay Programmer Reference*. Mincom Corporation, Feb. 1997.

[ISO88] ISO/IEC. *Information Processing Systems - Open Systems Interconnection - The Directory: Overview of Concepts, Models and Service*. ISO/IEC JTC 1/SC21, International Standard 9594-1, 1988.

[Kaß95] Thomas Kaß. *Anwendungsspezifische Autorisierungsstrategien in polymorphen persistenten Programmierumgebungen : Ein Bibliotheksansatz.* Diplomarbeit, Universität Hamburg, Fachbereich Informatik, Dez. 1995.

[Kern92] Helmut Kerner. *Rechnernetze nach OSI*. Addison-Wesley, 1992.

[KoNe93] John Kohl and Cliff Neumann. *The Kerberos network authentication service (V5).* Network Working Group, RFC 1510, Sep. 1993.

[Lamp81] Leslie Lamport. *Password authentication with insecure communication*. Communications of the ACM, Nov. 1981.

[LlSi92] B. Lloyd and W. Simpson. *PPP Authentication Protocols*. RFC 1511, Sep. 1993.

[Lott87] M. Lottor. *Domain Administrators Operations Guide*. RFC 1033, Nov. 1987.

[MeSi95] P. Metzger and W. Simpson. *IP Authentication using Keyed MD5*. Network Working Group, RFC 1828 (Standards Track), Aug. 1995.

[Mock87a] P. Mockapetris. *Domain Names - Concepts and Facilities*. RFC 1034 (Standard 13), Nov. 1987.

[Mock87b] P. Mockapetris. *Domain Names - Implementation and Specifications*. RFC 1035 (Standard 13), Nov. 1987.

[NBS77] NBS. *Data encryption standard*. Federal Information Processing Standards Publication 46, Jan. 1977.

[NBS80] NBS. *DES modes of operation*. Federal Information Processing Standards Publication 81, Dez. 1980.

[Nets96a] Netscape. *Using RSA Public Key Cryptography for Internet Security*. Netscape Communications Corporation, 1996.

[Nets97a] Netscape. *Netscape SSL 2.0 Certificate Format*. Netscape Communications Corporation, 1997.

[Opp92] Rolf Opplinger. *Computersicherheit.* Vieweg-Verlag, 1992.

[Post96] J. Postel. *Internet Official Protocol Standards*. STD 1, März 1996.

[RSA93] RSA. *RSA Certificate Services, An RSA White Paper*. RSA Data Security, Inc., Juli 1993.

[Schn94] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms and Source Code in C*. John Wiley & Sons, 1994.

[Simp93] W. Simpson. *The Point to Point Protocol*. RFC 1548, Dez.1993.

[Stahl87] M. Stahl. *Domain Administrators Guide*. RFC 1032, Nov. 1987.

[Will96] André Willomat. *Authentisierung in Tycoon: Eine polymorphe Entwicklungsschnittstelle zu Kerberos*. Studienarbeit, Universität Hamburg, Fachbereich Informatik, Juli 1996.

Anhang B

Abkürzungsverzeichnis

AH	*Authentication Header*
ARP	*Address Resolution Protocol*
BCP	*Best Current Practice*
CA	*Certificate Authority*
DES	*Data Encryption Standard*
DNS	*Domain Name System*
ESP	*Encapsulating Security Payload*
FTP	*File Transfer Protocol*
HTML	*Hypertext Markup Language*
HTTP	*Hypertext Transfer Protocol*
IAB	*Internet Architecture Board*
IANA	*Internet Assigned Numbers Association*
ICMP	*Internet Control Message Protocol*
IETF	*Internet Engineering Task Force*
IP	*Internet Protocol*
IPSEC	*Internet Protocol Security*
IRTF	*Internet Research Task Force*
ISO	*International Standards Organization*
MAC	*Message Authentication Code*
MIME	*Multipurpose Internet Mail Extensions*
NBS	*National Bureau of Standards*
NNTP	*Network News TransferProtocol*
OSI	*Open Standards Interconnection*
PIN	*Personal Identification Number*
PPP	*Point to Point Protocol*
RFC	*Request for Comments*
RIP	*Router Information Protocol*
RR	*Resource Record*
SA	*Security Association*
SMTP	*Simple Mail Transfer Protocol*
SPI	*Security Parameter Index*
SSL	*Secure Socket Layer*
STD	*Standard*
TCP	*Transmission Control Protocol*
TLS	*Transport Layer Security*
UDP	*User Datagram Protocol*
W3	*World Wide Web*
WWW	*World Wide Web*

Anhang C

Der Standardisierungsprozeß der Internet-Organisationen

Die Angaben zu Internet-Organisationen und Standardisierungsprozessen entstammen den Quellen [Brad96, Grund97, Post96].

Ein Internet-Standard ist eine Spezifikation, die folgenden Anforderungen genügen muß :

- Stabilität der Spezifikation
- allgemeines Verständnis
- technisch kompetent
- Existenz von unabhängigen Implementationen, die sich im Einsatz befinden
- signifikante öffentliche Unterstützung
- erkennbarer Nutzen für einige oder alle Teile des Internet.

Bevor aus einem Vorschlag von Mitarbeitern einer Arbeitsgruppe ein Standard werden kann, durchläuft dieser mehrere Zustände (*maturity levels*). Dieser Prozeß wird als *Standards Track* bezeichnet. Die Koordination von Arbeitsgruppen und Dokumenten wird in mehreren Internet-Organisationen durchgeführt, die im folgenden vorgestellt werden.

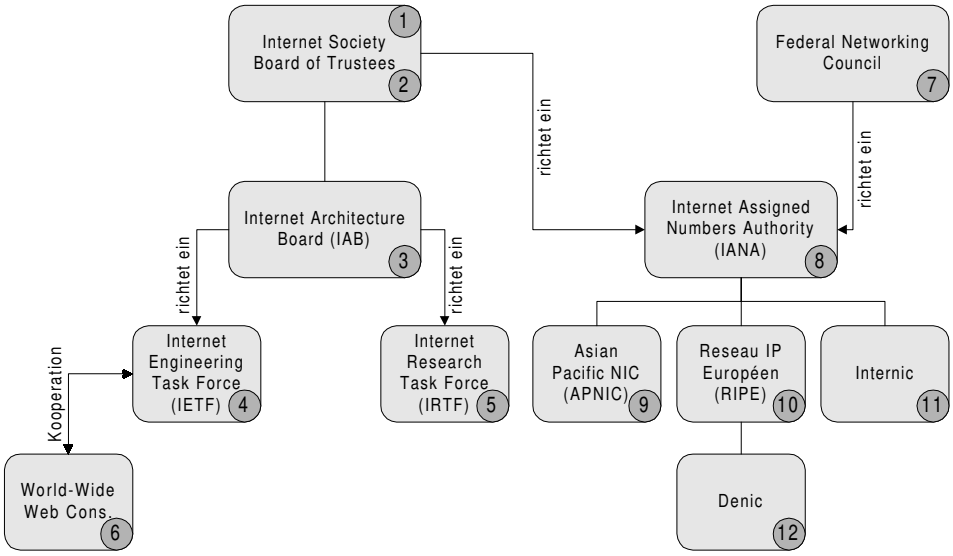

Abbildung 5.1 : Internet - Organisationen

1. *ISOC* : Die Dachorganisation des Internet. Sie richtet die Task Forces zur Klärung technischer Fragen ein.
2. *Board of Trustees* : Das wichtigste Entscheidungsgremium der ISOC.
3. *IAB* : Die oberste Expertenrunde des Internet. Sie entscheidet über Standard-Vorschläge der Task Forces.
4. *IETF* : Ein Zusammenschluß von Arbeitsgruppen, die Standards entwickeln.
5. *IRTF* : Zuständig für die zukünftige technologische Entwicklung des Internet
6. *WWW-Consortium* : Ist zuständig für Standards und Weiterentwicklung im World-Wide Web.
7. *FNC* : Zuständig für die Vergabe von Domains der Top-Level „gov“, „mil“ und „edu“.
8. *IANA* : Oberste für die Namensvergabe zuständige Organisation. Sie weist ihren Unterorganisationen Namensräume zu.
9. *APNIC* : Zuständig für die Vergabe von Internetadressen im asiatischen Raum.
10. *RIPE* : Verantwortlich für Internetadressen in Europa und angrenzenden Regionen.
11. *Internic* : Verantwortlich für nicht ländergebundene Domains. (z.B. „com“, „net“ oder „org“)
12. *Denic* : Zuständig für die Vergabe deutscher Internetadressen.

Zu Beginn erhält ein Dokument des Status *draft* . Nach mehreren Perioden von Untersuchungen und Weiterentwicklung durch die Internet-Gemeinschaft, kann ein draft zum *Standard* werden. Zusätzlich zum Standard existieren Dokumentgruppen, die zur Information oder experimentellen Zwecken dienen. Der Zustandsverlauf von Dokumenten wird in folgender Abbildung beschrieben :

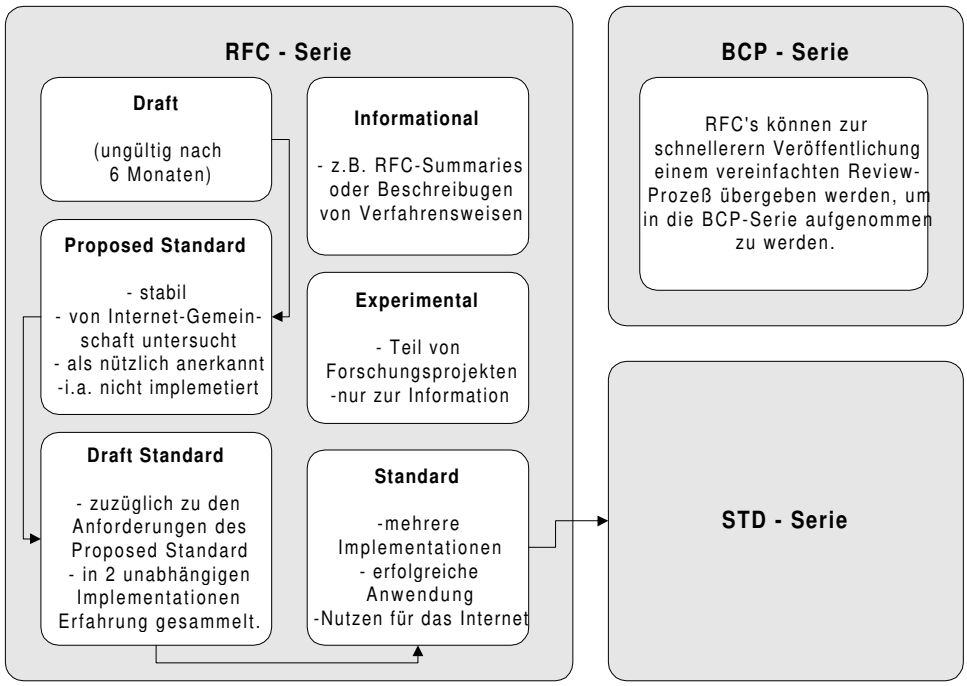

Abbildung 5.2 : Der Standardisierungsprozeß von Internet-Dokumenten

Die Verbreitung dieser Dokumente geschieht unter Aufsicht nach einem vorgeschriebenen Zeitplan auf einer Vielzahl von Internet-Servern. Das Standard Format ist ASCII-Text, ein Dokument kann aber auch in anderen Formaten vorliegen.