Internet Economics IV

Burkhard STILLER
Frank EYERMANN
Arnd HEURSCH
Peter RACZ
Rute SOFIA
(Hrsg.)

Institut für Informationstechnische Systeme, IIS

Bericht Nr. 2004-04
Juli 2004

Universität der Bundeswehr München

Fakultät für

INFORMATIK

Werner-Heisenberg-Weg 39 • D-85577 Neubiberg

B. Stiller, F. Eyermann, A. Heursch, P. Racz, R. Sofia (edts.): Internet Economics IV
Technical Report No. 2004-04, August 2004
Information Systems Laboratory IIS, Departement of Computer Science
University of the Federal Armed Forces Munich
Werner-Heisenberg-Weg 39, D-85577 Neubiberg, Germany
URL: http://www.informatik.unibw-muenchen.de/reports/

Introduction

The Information System Laboratory (Institut für Informationstechnische Systeme, IIS) of the Department of Computer Science, University of the Federal Armed Forces Munich, Germany started two years ago research and teaching in the area of communications. One of the closely related topics is addressing the use and application of technology and mechanisms under economic and technical optimization measures. Therefore, during the spring term 2004 (FT 2004) a fourth instance of the Internet Economic seminar has been prepared and students as well as supervisors worked on this topic.

Still today, Internet Economics are run rarely as a teaching unit. This is a little in contrast to the fact that research on Internet Economics has been established as an important area in the center of technology and economics on networked environments. During the last ten years, the underlying communication technology applied for the Internet and the way electronic business transactions are performed on top of the network have changed. Although, a variety of support functionality has been developed for the Internet case, the core functionality of delivering data, bits, and bytes remained unchanged. Nevertheless, changes and updates occur with respect to the use, the application area, and the technology itself. Therefore, another review of a selected number of topics has been undertaken.

Content

The fourth edition of the seminar 'Internet Economics IV' deals with the use of Internet technology and additional ways to support and do business. Starting the talks, a view onto techniques of a public key infrastructure is presented. It is discussed with respect to its technology and its economic impacts in the Internet world today. The second talk addresses the area of AAA protocol, summarizing authentication, authorization, and accounting questions in the Internet. Since commercial services drive the need for security, the set of AAA services offered by AAA protocols form their basis. Service Level Agreements (SLA) define a contractual relation between a service provider and a service user. The third talk addresses SLA in the Information Technology (IT) environment and outlines main aspects for communication services as well.

Talk number four outlines an operational perspective of the Internet, looking at research networks as well as key technological issues in support of the multi-administration model applied. Autonomous systems, Internet addresses, ISPs and their tiers, as well as standardization organizations are discussed. On the application layer of todays Internet

web services have been defined. The fifth talk summarizes key aspects of XML (Extended Markup Language), Web Services and their components, and B2B/B2C aspects of those in a technical and economic snapshot. Talk number six discusses the trade-off between quality and cost, which outlining QoS (Quality-of-Service) aspects and over-provisioning views. This part summarizes for a comparison a number of known pricing models as well.

The seventh talk focuses on reputation and trust as an underlying factor for distributed systems in support of partner-to-partner communications. Those key business-enabling prerequisites and mechanisms are defined, their main characteristics are outlined, and practical examples are presented. Talk eight runs into details of advanced pricing mechanisms for content. While the set of available schemes is categorized and major differentiating factors of pricing schemes are discussed, content and its influences on prices and the economy are provided.

The ninth talk provides an overview on wireless networks and their hand-off mechanisms. Mobile IPv6 and its enhancements are discussed and evaluated with respect to their efficiency. A comparison of technological choices concludes the work. Finally, talk number ten addresses security issues in wireless networks. IEEE 802.1x as well as RADIUS are discussed. A presentation of the user view will be complemented by the complex provider operation.

Seminar Operation

As usual and well established now, all interested students worked on an initially offered set of papers and book chapters, relating to the topic titles as presented in the Table of Content below. They prepared a written essay as a clearly focussed presentation, an evaluation, and a summary of those topics. Each of these essays is included in this technical report as a separate section and allows for an overview on important areas of concern, sometimes business models in operation, and problems encountered. In addition, every student prepared a slide presentation of approximately 45 minutes to present his findings and summaries to the audience of students attending the seminar and other interested students, research assistants, and professors. Following a general question and answer phase, a student-lead discussion debated open issues and critical statements with the audience.

Local IIS support for preparing talks, reports, and their preparation by students had been granted Frank Eyermann, Arnd Heursch, Peter Racz, Rute Sofia, and Burkhard Stiller. In particular, many thanks are addressed to Arnd Heursch for his strong commitment on getting this technical report ready and quickly printed. A larger number of pre-presentation discussions have provided valuable insights in the emerging and moving field of Mobile Systems, both for students and supervisors. Many thanks to all people contributing to the success of this event, which has happened again in a small group of highly motivated and technically qualified students and people.

Neubiberg, July 2004

Inhaltsverzeichnis

Kapitel 1

Techniques of Public Key Infrastructures and their Economic Impacts

Alexander Franke

Kommerzielle Anwendungen im Internet sind heute jedem bekannt und werden mehr oder weniger genutzt. Tendenziell lässt sich eine stetige Zunahme kommerzieller Internetanwendungen feststellen. Die technischen Voraussetzungen sind für diese Anwendungen erfüllt. Damit sie sich aber auf großer Breite durchsetzen können, bedarf es einer hinreichenden Akzeptanz beim Unternehmer und auch beim Kunden. Um diese Akzeptanz zu erreichen sind sichere Internetanwendungen notwendig. Wenn z.B. ein Kunde einen Kaufvertrag eingeht, muss sichergestellt sein, dass er diesen im Nachhinein nicht leugnen kann. Ein anderes Ziel könnte sein, ein elektronisches Dokument nur dem Kunden verfügbar zu machen, der es auch erworben hat. Ziel dieser Arbeit ist es, diese exemplarischen Sicherheitsanforderungen zu ordnen und Lösungswege aufzuzeigen. Dazu werden eingangs die in Frage kommenden Sicherheitsanforderungen voneinander abgegrenzt und die kryptographischen Grundlagen gelegt, mit denen diese Anforderungen erfüllt werden können. Um aber darauf aufbauend sichere kommerzielle Anwendungen zu ermöglichen, bedarf es einer konsistenten Struktur, in der die beteiligten Kommunikationspartner interagieren. Diese Struktur wird im Rahmen der Public Key Infrastructures konkretisiert und bildet den Kern dieser Arbeit. Zum Verständnis werden einige dieser Strukturen, oder Teilmengen davon, näher betrachtet. Die Arbeit schließt mit einer kurzen Betrachtung der möglichen, ökonomischen Auswirkungen und einer spekulativen Zukunftsbetrachtung dieser Mechanismen.

Inhaltsverzeichnis

1.1 Grundlagen

1.1.1 IT-Sicherheit

Das zentrale Interesse im Bereich der IT-Sicherheit besteht darin, Informationen zu schützen. In dieser Arbeit geht es unter anderem um die Zusammenarbeit von Computern in Netzwerken. Somit ist ein Computer nicht isoliert und damit aus dem Netzwerk angreifbar. Sicherheitsprobleme entstehen auch dadurch, dass bspw. Anwendungen, die für homogene Netzwerke entwickelt wurden, weiterentwickelt werden. Der dadurch erreichte Zugewinn an Einsatzfeldern wird oft um den Preis von neuen Sicherheitslücken erkauft. Auch der Benutzer einer Anwendung selbst stellt ein Problem dar, sofern er gewollt oder ungewollt Schaden anrichten kann. Da es keine zentrale Überwachungsinstanz im Internet gibt, müssen Kommunikationsmechanismen zwischen Computern und die Architektur einer Teilmenge von Computern so gestaltet sein, dass eine hinreichend sichere Interaktion möglich ist.

All diese Sicherheitsprobleme sind noch sehr diffus in ihrer Beschreibung. Um diese Probleme behandeln zu können unterscheiden wir die, für diese Arbeit notwendigen, Sicherheitsziele [1]:

- Datenvertraulichkeit (data confidentiality) bedeutet, dass geheime Daten auch geheim bleiben sollen. Das schließt ein, dass Daten von unautorisierten Personen nicht eingesehen werden können. Voraussetzung dafür ist, das der Eigentümer der Daten spezifizieren kann, welche Benutzer die Daten einsehen dürfen.

- Datenintegrität (data integrity) bedeutet, dass Daten nicht ohne Erlaubnis des Eigentümers modifiziert werden können. Unter Modifikation von Daten verstehen wir auch das Löschen oder Hinzufügen von Daten.

- Authentizität (authentication) bedeutet, das die Identität eines Benutzers bewiesen ist, d.h. dass der Benutzer der Absender von Daten ist.

- Verbindlichkeit oder Nicht-Abstreitbarkeit (non-repudiation) bedeutet, dass jede Aktion, die ein Benutzer ausführt, auch genau diesem Benutzer zugeordnet werden kann. Der Benutzer kann somit später keine seiner Aktionen abstreiten.

Um diese Sicherheitsziele gewährleisten zu können, sind Verschlüsselungsmechanismen von zentraler Bedeutung. Wir werden im folgenden die beiden wichtigsten Verschlüsselungsverfahren erläutern. Das Verständnis dieser Verfahren ist für den weiteren Verlauf dieser Arbeit unerlässlich. Das darauf folgende Beispiel des RSA-Verfahren soll als pragmatisches Beispiel für die Beurteilung von kryptographischen Verfahren dienen.

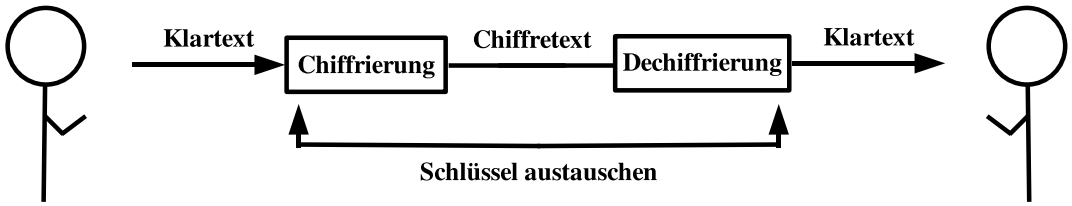

Abbildung 1.1: Secret-Key-Verfahren

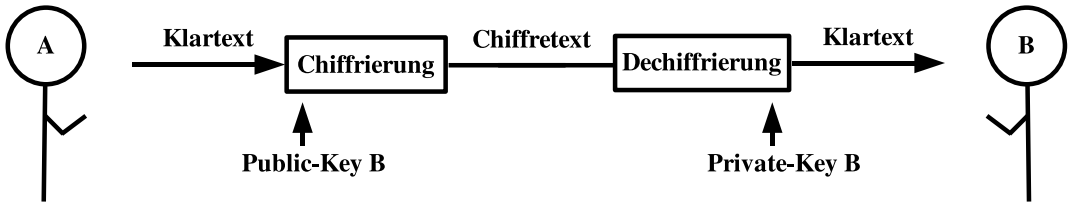

Abbildung 1.2: Public-Key-Verfahren

1.1.2 Secret-Key- oder symmetrische Verschlüsselungsverfahren

Dieses Verfahren ist in Abbildung 1.1 illustriert. Der Sender und der Empfänger verfügen über einen gleichen und geheimen Schlüssel. Gleich sind zwei Schlüssel hier auch, falls der eine Schlüssel leicht aus dem anderen Schlüssel gewonnen werden kann. Mit diesem Schlüssel wird der Klartext chiffriert. Der Chiffretext kann dann über einen unsicheren Kanal übertragen werden. Der Empfänger dechiffriert nun mit dem gleichen Schlüssel den Chiffretext und erhält wieder den Klartext. Da auf beiden Seiten der gleiche Schlüssel verwendet wird, spricht man auch von symmetrischer Verschlüsselung. Die Bezeichnung Secret-Key-Verfahren leitet sich aus der Geheimhaltung des gemeinsamen Schlüssels ab. Das ist auch ein Nachteil dieses Verfahrens, denn für das Austauschen des Schlüssels ist ein sicherer Kanal erforderlich. Es kann sogar erforderlich sein, dass sich beide Kommunikationspartner dazu treffen müssen.
Mit dem Secret-Key-Verfahren lässt sich Datenvertraulichkeit und Datenintegrität sicherstellen. Verbindlichkeit ist hier aber nicht zu erreichen da hier beide Kommunikationspartner die Daten manipulieren können. Es fehlt also die eineindeutige Zuordnung zwischen Aktion und Benutzer. Vorteilhaft ist die Schnelligkeit mit der Secret-Key-Verfahren chiffrieren und dechiffrieren können. Weiterhin bieten diese Verfahren, bei genügend großer Schlüssellänge, eine gute Sicherheit. Als sicher können zur Zeit Sommer 2004, Schlüssellängen von 192 Bit und mehr angesehen werden.

1.1.3 Public-Key- oder asymmetrische Verschlüsselungsverfahren

Im Gegensatz zu den symmetrischen Verschlüsselungsverfahren werden hier unterschiedliche Schlüssel für die Ver- und Entschlüsselung verwendet. Der Empfänger B verfügt wie in Abbildung 1.2 gezeigt über einen öffentlichen Schlüssel (Public-Key B) und einen

privaten Schlüssel (Private-Key B). Damit der Sender A eine geheime Nachricht an den Empfänger B schicken kann, verschlüsselt A den Klartext mit dem öffentlichen von B. Um die Nachricht zu entschlüsseln, verwendet dann B seinen geheimen Schlüssel. Der öffentliche Schlüssel ist i.a. auf einer Website im Internet, eine so genannte Trusted Authority, z.B. bei der Telekom hinterlegt und kann somit von jedermann zum Verschlüsseln von Nachrichten benutzt werden, die nur vom Besitzer des geheimen, privaten Schlüssels entschlüsselt werden können.

Damit dieses Verfahren, auch asymmetrisches Verfahren genannt, funktioniert, muss es nahezu unmöglich sein, eine verschlüsselte Nachricht ohne Kenntnis des privaten Schlüssels zu entschlüsseln. Mit diesem Verfahren wird die Datenvertraulichkeit und Datenintegrität sichergestellt. Weiterhin ermöglicht uns dieses Verfahren das Sicherstellen von Verbindlichkeit, denn die Public-Key-Verfahren ermöglichen es, digitale Unterschriften, oder allgemeiner Signaturen, zu erstellen. Die Public-Key-Verfahren bieten also zum einen den Vorteil, dass nur der öffentliche Schlüssel transportiert werden muss und zum anderen mit Hilfe einer digitalen Unterschrift Verbindlichkeit sichergestellt werden kann. Der öffentliche Schlüssel muss nicht vertraulich behandelt werden. Für sichere Anwendungen ist allerdings die Authentizität des öffentlichen Schlüssels sicherzustellen. Wie das geschehen kann, wird im weiteren Verlauf dieser Arbeit behandelt.
Bei diesem Verfahren beruht die Sicherheit auf der mathematischen Komplexität zum Finden des privaten Schlüssels. Nachteilig ist, dass man nicht genau weiß, wie komplex dieses Problem ist. Auch könnte es durchaus sein, dass das Entschlüsseln von Daten weit weniger komplex ist als das Finden des privaten Schlüssels. Ein weiterer Nachteil ist die schlechte Performanz der Public-Key-Verfahren. Diese Verfahren sind ca. um den Faktor 10^3 langsamer als vergleichbare Secret-Key-Verfahren. Um diesen Nachteil abzuschwächen, bieten geeignete Public-Key-Verfahren die Möglichkeit, dass zwei Kommunikationspartner über einen unsicheren Kanal zu einem gemeinsamen Geheimnis kommen können.

Dazu ist eine Funktion f mit folgender Eigenschaft nötig:

$$f(Private\_Key\_A, Public\_Key\_B) = f(Private\_Key\_B, Public\_Key\_A).$$

Das bedeutet, wenn A seinen privaten Schlüssel auf den öffentlichen Schlüssel von B anwendet ist das Ergebnis gleich dem als wenn B seinen privaten Schlüssel auf den öffentlichen Schlüssel von A anwendet. Um nun die Gesamt-Performanz der Verschlüsselung zu verbessern, verwendet man in der Praxis hybride Verschlüsselungsverfahren, ohne auf Sicherheit verzichten zu müssen. Mit Hilfe von Public-Key-Verfahren einigen sich zwei Kommunikationspartner über einen gemeinsamen und geheimen Schlüssel. Mit Hilfe dieses Schlüssels werden dann die Daten schnell und effizient mit Hilfe von Secret-Key-Verfahren verschlüsselt.

Um auch bei Signaturen die Geschwindigkeit zu steigern, wendet man hier das Public-Key-Verfahren nur auf einen Hash-Wert der Daten des Dokuments an. Bei der Hash-Funktion handelt es sich um eine Einwegfunktion, also eine Funktion die sehr schwer zu invertieren ist, die angewendet auf ein Dokument ein Ergebnis fester Länge erzeugt. Die Länge

des Hash-Wertes ist unabhängig vom Dokument. Eine wichtige Eigenschaft der Hash-Funktion ist, dass selbst Dokumente, die sich nur sehr geringfügig unterscheiden völlig unterschiedliche Hash-Werte erzeugen. Dadurch kann ein Dokument vor Manipulationen geschützt werden, d.h. seine Integrität wird gewahrt, da Manipulationen durch einen anderen Hash-Wert sofort auffallen würden. Die Hash-Funktion ist deterministisch, d.h. dasselbe Dokument erhält unverfälscht stets denselben Hash-Wert. Wenn ein Benutzer also ein Dokument signieren möchte, berechnet er zuerst den Hash-Wert des Dokuments. Auf diesen Hash-Wert wendet er seinen privaten Schlüssel an. Das Ergebnis dieser Operation ist die Signatur. Das Dokument wird dann mit der Signatur verschickt. Der Empfänger kann nun mit der Signatur feststellen ob das Dokument verfälscht wurde oder ob ein Übertragungsfehler aufgetreten ist. Dazu berechnet er zum einen selber den Hash-Wert des Dokuments und zum anderen wendet er den öffentlichen Schlüssel des Senders auf die Signatur an. Ver- und Entschlüsselung heben sich auf und der Empfänger kann durch Vergleich mit dem selbst berechneten Hash-Wert feststellen, ob das Dokument verfälscht wurde. Damit die Hash-Funktion als sicher angesehen werden kann, muss es praktisch unmöglich sein zu einem vorgegebenen Dokument ein davon verschiedenes zu finden das den gleichen Hash-Wert erzeugt. Bekannte Hash-Funktionen sind MD5, SHA und RIPEMD. MD5 steht für message digest und stammt von R.L. Rivest und S. Dusse. MD5 liefert 128-Bit-Hash-Werte. SHA steht für Secure Hash Algorithm und erzeugt 160-Bit-Hash-Werte. RIPEMD wurde von einem europäischen Konsortium entwickelt. Das deutsche Signaturgesetz empfiehlt die Hash-Funktionen SHA-1 und RIPEMD-160.

1.1.4 RSA-Verfahren

Das RSA-Verfahren, nach seinen Erfindern Rivest, Shamir und Adleman benannt, ist ein Verfahren zur Public-Key-Kryptographie. Dieses Verfahren beruht auf der Schwierigkeit des Faktorisierungsproblems. Genauer gesagt ist es einfach, das Produkt zweier großer Primzahlen zu berechnen, aber es ist fast unmöglich bzw. sehr, sehr aufwendig aus diesem Produkt ohne Kenntnis der Primfaktoren, diese zu ermitteln. Wir betrachten jetzt ein kleines Beispiel wie das RSA-Verfahren arbeitet [2]. Wir müssen zuerst ein Schlüsselpaar, bestehend aus privatem und öffentlichem Schlüssel vereinbaren. Dazu wählen wir zwei beliebige Primzahlen p und q aus. Es seien

$$p = 7 \text{ und } q = 13.$$

Wir berechnen den Modulus n mit

$$n = p \cdot q = 7 \cdot 13 = 91.$$

Nun bestimmen wir den so genannten Eulerschen ϕ-Wert von n mit

$$\phi(n) = (p-1) \cdot (q-1) = 6 \cdot 12 = 72.$$

Wir wählen ein $e < n$, das teilerfremd zu $\phi(n)$ ist. Es sei

$$e = 5.$$

Weiterhin wählen wir ein $d < n$ mit $(d \cdot e) \mod \phi(n) = 1$. Es sei

$$d = 29.$$

Das Paar $(e, n) = (5, 91)$ ist der öffentliche Schlüssel. Das Paar $(d, n) = (29, 91)$ ist der private Schlüssel. Um die Anwendung des Verfahrens zu zeigen, seien unsere Daten B als Zahl codiert. Sei $B = 2$. Die Verschlüsselung C berechnet sich wie folgt:

$$B^e \mod n = 2^5 \mod 91 = 32 = C.$$

Um die codierten Daten wieder zu entschlüsseln, wenden wir den privaten Schlüssel wie folgt an:

$$C^d \mod n = 32^{29} \mod 91 = 2 = B.$$

RSA-Verfahren können heute effizient implementiert werden und werden häufig angewendet. Der Aufwand zur Berechnung des privaten Schlüssels ist äquivalent zur Faktorisierung des Moduls n. Mit hinreichend großem Aufwand, i.a. in MIPS (million instructions per second) ausgedrückt, sind heute 512-Bit-RSA-Zahlen faktorisierbar. Daraus folgt, das für sicherheitskritische Anwendungen größere Schlüssellängen benutzt werden sollten. Der Besitzer des privaten Schlüssels hingegen kann die Daten mit vergleichsweise geringem Aufwand wieder entschlüsseln.

Aber wie sicher ist die heutige Verschlüsselung im Hinblick auf zukünftige Technologien? In der Theorie gibt es sie, die Quantencomputer. Wie weit ist die Technologie und was würde das für heutige Verschlüsselungsverfahren bedeuten? Kurz gesagt ist diese Technologie noch weit entfernt, aber sie wird längst nicht mehr als unrealistisch eingestuft. Auch kann man sagen das bisher fast jedes kryptographische Verfahren in der Vergangenheit gebrochen wurde, es also mit Sicherheit ein Trugschluss ist von heutigen Verfahren Gegenteiliges zu erwarten. Was die Quantencomputer betrifft, so sind die heutigen, fundamentalen Probleme bei dieser Technik eher praktischer Natur. Es sieht also so aus, als ob es nur noch eine Frage der Zeit ist. Die Schätzungen gehen aber von 5 bis 200 Jahren aus. Ab der Existenz von Quantencomputern sind vor allem kryptographische Methoden die auf Primfaktorzerlegung, wie das RSA-Verfahren, beruhen, bedroht. Grund dafür ist die massive Parallelität mit der Quantencomputer arbeiten können. Beim symmetrischen DES-Algorithmus würde sich der Aufwand zur Analyse nur halbieren. Ein Verdopplung der Schlüssellänge würde diesen Nachteil wieder kompensieren [3].

1.2 Public Key Infrastructures

Nachdem wir die zu erreichenden Sicherheitsziele eingegrenzt und die Grundlagen der dafür notwendigen Verschlüsselungsverfahren gelegt haben, kommen wir nun zum Kern

dieser Arbeit, den Public-Key-Infrastructures (PKI).

Definition: Public-Key-Infrastructures bezeichnet die Menge der Instanzen, die für den Einsatz asymmetrischer Kryptographie in offenen Systemen erforderlich sind.

Kernaufgaben einer PKI:

- Registrieren der Nutzer
- Ausstellen, Verwalten und Prüfen von Zertifikaten

Zertifizierungsstelle (Certification Authority, CA): Stellt durch Zertifikate die Echtheit von öffentlichen Schlüsseln und die Identität ihrer Eigentümer sicher.

1.2.1 Zertifikate

Definition: Ein Zertifikat ist eine Beglaubigung, dass ein Schlüsselpaar zu einer natürlichen Person oder einer Instanz im Netz gehört.

Zertifikate werden durch Zertifizierungsinstanzen ausgestellt. Der Inhalt eines Zertifikats ist z.B. im deutschen Signaturgesetz festgelegt. Im wesentlichen enthält ein Zertifikat immer den Zertifikatnehmer, dessen Public-Key, die verwendeten Verschlüsselungsverfahren, den Aussteller und die Gültigkeitsdauer. Ein Auszug aus dem deutschen Signaturgesetz sieht wie folgt aus:

Auszug SigG §7
(4) Die Zertifizierungsstelle hat Vorkehrungen zu treffen, damit Daten für Zertifikate nicht unbemerkt gefälscht oder verfälscht werden können. Sie hat weiter Vorkehrungen zu treffen, um die Geheimhaltung der privaten Schlüssel zu gewährleisten. Eine Speicherung privater Signaturschlüssel bei der Zertifizierungsstelle ist unzulässig.

Dass die Zertifizierungsstelle die privaten Signaturschlüssel nicht speichern darf, folgt aus der Forderung nach Verbindlichkeit.

1.2.2 Digitale Signaturen

Formal ist eine Signatur sig das Ergebnis einer Funktion s angewendet auf die zu signierenden Daten und den geheimen Schlüssel des Unterzeichners:

$$sig = s(Private\_Key, Daten).$$

Damit jeder überprüfen kann ob die Signatur auch zu den Daten passt, existiert eine Verifikationsabbildung v wie folgt:

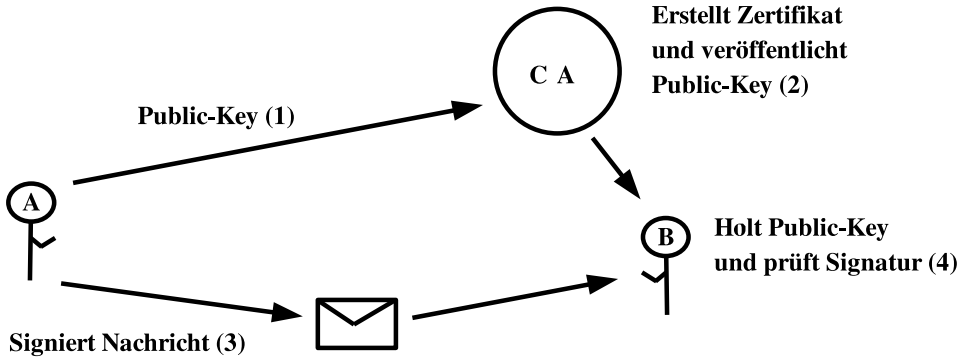

Abbildung 1.3: Digitale Signatur

$$v(Public\_Key, Daten, sig) \rightarrow \{true, false\}.$$

Das heißt jeder kann mit Hilfe der Signatur und des öffentlichen Schlüssels des Unterzeichners prüfen ob die Signatur zu den Daten passt. Der Unterzeichner kann nichts abstreiten, da die Signatur nur mit Hilfe seines privaten Schlüssels erstellt werden kann. Für den öffentlichen Schlüssel muss die Authentizität gewährleistet sein. Hierzu betrachten wir jetzt eine PKI für den Einsatz von Digitalen Signaturen. In Abbildung 1.3 möchte A eine signierte Nachricht an B senden. Um authentische Signaturen erstellen zu können, muss sich A erst ein Zertifikat besorgen. Dazu wendet sich A mit seinem Public-Key an eine Zertifizierungsinstanz CA. Die CA überprüft die Identität von A und veröffentlicht den öffentlichen Schlüssel von A in Verbindung mit seiner Identität. Nun kann A, mit seinem geheimen Schlüssel und i.a. einem Hash-Wert seiner Nachricht, seine Nachricht signieren und an B senden. Nach Erhalt der Nachricht holt sich B, in Verbindung mit der Identität von A, den Public-Key von A von der CA. B kann sich nun sicher sein, dass der Public-Key von A authentisch ist, also zu A gehört. Nun prüft B die Signatur.
Digitale Signaturen stellen somit Datenintegrität, Datenverbindlichkeit und die Identitätsfeststellung sicher. Wie wichtig digitale Signaturen für sichere Internetanwendungen, vor allem im kommerziellen Bereich, sind, deutet folgendes Zitat an:

For practical applications, digital signatures are one of the two most important cryptologic primitives. In particular with the rise of electronic commerce on the Internet and the World Wide Web, they may become even more important than the better-known schemes for message secrecy. Pfitzmann, 1996

1.2.3 Pretty Good Privacy

Pretty Good Privacy (PGP) ist der De-Facto Standard für vertrauliche und authentische E-Mail. PGP ist ein Beispiel für hybride Verschlüsselung. Abbildung 1.4 stellt den Ablauf bei der Verschlüsselung mit PGP dar. Für jede Sitzung wird ein Sitzungsschlüssel (Session-Key) erzeugt. Mit diesem Schlüssel werden die Daten symmetrisch verschlüsselt. Der Sitzungsschlüssel selbst wird dann mit dem Public-Key des Empfängers asymmetrisch verschlüsselt und zusammen mit der symmetrisch verschlüsselten Nachricht übertragen.

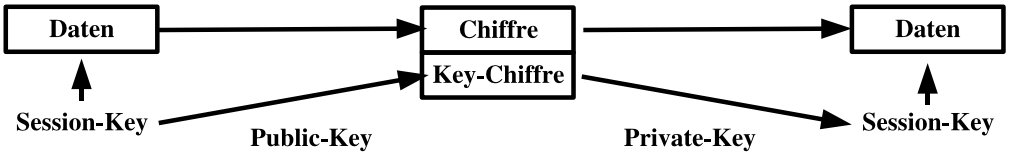

Abbildung 1.4: Pretty Good Privacy

Der Empfänger kann mit Hilfe seines Private-Keys den Sitzungsschlüssel wiederherstellen und die Daten entschlüsseln. Üblich ist heutzutage der International Data Encryption Algorithm (IDEA) mit 128-Bit-Schlüsseln für die symmetrische Verschlüsselung. Für die asymmetrische Verschlüsselung werden Schlüssellängen bis zu 4096-Bit verwendet. Deutsche Zertifizierungsinstanzen sind z.B. das Deutsche Forschungsnetz (DFN) oder die Zeitschrift c't.

1.2.4 Secure Shell

Die Secure Shell (SSH) bezeichnet sowohl ein kryptographisches Protokoll als auch dessen Implementierung. SSH ermöglicht folgende, sichere Anwendungen [7]:

- Login auf einer entfernten Maschine
- Ausführung von Kommandos auf einer entfernten Maschine
- Das Kopieren von Dateien zwischen verschiedenen Rechnern im Netz

Das aktuelle SSH-Protokoll 2.x wird durch die Secure Shell Working Group beschrieben. Die SSH-Protokolle 1.x sollten aufgrund von Sicherheitslücken und mangelnder Kryptosicherheit nicht mehr verwendet werden. Damit ein Benutzer SSH-Dienste benutzen kann, muss er sich zuerst ein Schlüsselpaar erzeugen (z.B. mit dem Befehl ssh-keygen). Der Private-Key verbleibt auf dem Rechner des Benutzers und wird zusätzlich mit einem Benutzerpasswort verschlüsselt. Dann wird der Public-Key dem Server bekannt gemacht. Der Server wiederum macht seinen Public-Key beim Benutzer bekannt. Die Anwendungen, die SSH benutzen, verbergen in der Regel diese Vorbereitungen vor dem Benutzer. Wie der Login bei SSH-Anwendungen abläuft, ist in Abbildung 1.5 gezeigt. Nachdem eine normale TCP-Verbindung zum Server aufgebaut ist, übermittelt der Server dem Client seine zwei öffentlichen Server-Schlüssel. Zwei Schlüssel daher weil der Server einen generellen öffentlichen Schlüssel zur Identifikation seiner Maschine besitzt und einen zweiten öffentlichen Schlüssel für den aktuellen Server-Prozess. Nach Erhalt der Server-Schlüssel generiert der Client einen Sitzungsschlüssel und chiffriert diesen mit den Server-Schlüsseln. Nachdem der Server den verschlüsselten Sitzungsschlüssel erhalten hat und mit seinem privaten Schlüssel entschlüsselt hat, läuft die weitere Verbindung symmetrisch verschlüsselt ab. Der Client authentifiziert sich gegenüber dem Server i.a. mit einer Challenge-Response-Authorisierung und der Server stellt dem Client eine Arbeitsumgebung bereit. Bei einer Challenge-Response-Authorisierung sendet ein Server dem Kommunikationspartner eine Zufallszahl, die dieser nach einem beiden bekannten Verfahren verschlüsselt und zurück

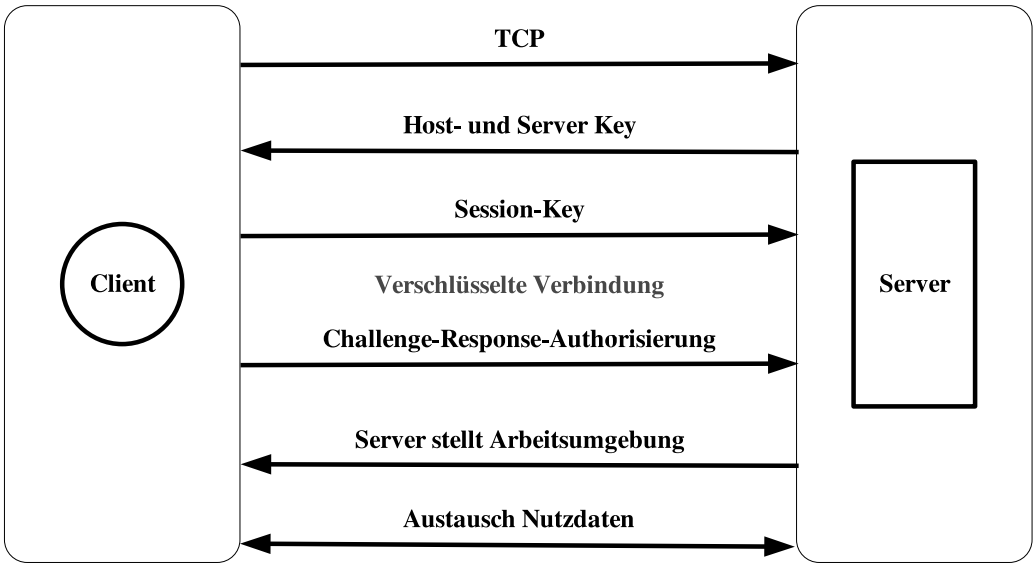

Abbildung 1.5: Login SSH

sendet. Der Server überprüft dann, ob der Kommunikationspartner die erwartete Antwort gesandt hat. Der Vorteil dieses Verfahrens liegt darin, dass das Passwort nicht bei der Authentifikation über das Netzwerk gesendet wird. Server und Client haben also ein gemeinsames Geheimnis, dass Passwort des Benutzers. Damit ist das SSH-Login beendet und der Austausch von Nutzdaten kann beginnen. Die SSH nutzt keine Certification Authority (CA). Sie speichert aber lokal öffentliche Schlüssel von anderen Rechnern mit denen sie schon kommuniziert hat und kann dadurch Man in the Middle Attacken oder Schlüsselverlust durch Neukonfiguration bei Kommunikationspartnern erkennen und den Anwender warnen bzw. darauf hinweisen.

1.2.5 Kerberos

Bei Kerberos handelt es sich um eine Entwicklung des Massachusetts Institute of Technology (MIT). Kerberos ist ein verteilter Authentifizierungsdienst. Kerberos besteht aus drei Protokollen, dem Single-Sign-on-Protocol, dem Key-Distribution-Protocol und dem Authentication-Protocol. Die Zusammenarbeit dieser Protokolle ist in Abbildung 1.6 dargestellt.

Das Single-Sign-on-Protocol beschreibt, wie sich ein Benutzer im Kerberossystem anmeldet. Dazu fordert der Client beim Authentification Server (AS) ein so genanntes Ticket-Granting Ticket (TGT) an. Die Anforderung des Clients wird nicht verschlüsselt. Der AS sendet dann das TGT und ein Zertifikat C an den Client. Das Zertifikat ist mit dem geheimen Passwort des Clients verschlüsselt und kann so nur von diesem entschlüsselt werden. Dazu muss der Client das Passwort eingeben, es wird nicht zwischengespeichert. Das Zertifikat enthält unter anderem einen Sitzungsschlüssel um die weitere Kommunikation zu

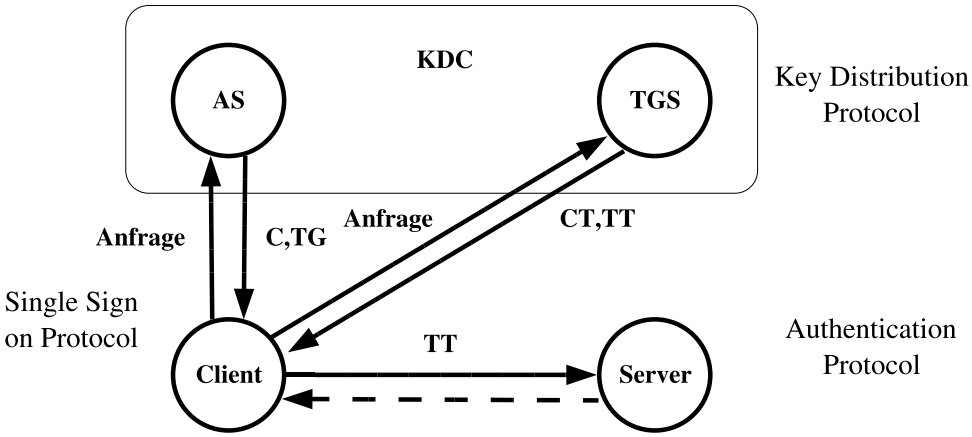

Abbildung 1.6: Kerberos

verschlüsseln und dient dem Client zur Authentifizierung. Der Sitzungsschlüssel hat eine zeitlich begrenzte Gültigkeit und funktioniert nur für den Ticket-Granting Server (TGS). Der TGS ist ein zentraler Schlüsselverteiler. Mit Hilfe des TGT kann der Client, solange der Sitzungsschlüssel gültig ist, für alle Transaktionen einen Schlüssel beim TGS holen. Das TGT ist mit dem geheimen Schlüssel des TGS verschlüsselt und enthält neben Daten zum Client auch den Sitzungsschlüssel. Damit ist das Login-Protokoll abgeschlossen. AS und TGS zusammen bezeichnet man auch als Key-Distribution-Center (KDC).

Damit der Client eine Transaktion auf einem bestimmten Server ausführen kann, tritt er nicht direkt mit dem Server in Kontakt sondern fordert beim TGS ein Transaktionsticket (TT) an. Das wird im Key-Distribution-Protocol beschrieben. Der Client schickt eine Anfrage im Klartext in Verbindung mit dem TGT zum TGS. Der TGS sendet dann dem Client das TT und ein Zertifikat CT. Dieses Ticket ist wieder nur begrenzt gültig. Der Client kann wieder das Zertifikat entschlüsseln und erhält so alle Informationen die er für die Kommunikation mit dem Server braucht.

Die Kommunikation mit dem Server wird im Authentication-Protocol beschrieben. Während der Lebensdauer des Tickets TT kann nun der Client eine Transaktion auf dem Server durchführen. Mit dem Ticket TT authentifiziert sich der Client auch auf dem Server. Nur der Server ist in der Lage das Ticket TT zu entschlüsseln.

Was macht nun Kerberos so sicher? Die Anforderung des Clients erfolgt im Klartext. Die Antwort des AS ist mit dem geheimen Passwort des Benutzers verschlüsselt und kann so nur von diesem entschlüsselt werden. Da zu keinem Zeitpunkt ein Passwort über das Netz übertragen wird, können auch keine Passwörter ausgespäht werden. Der Client kann sich so auch von der Identität des AS überzeugen da nur die AS die Antwort mit seinem geheimen Schlüssel verschlüsselt haben kann. Die AS überzeugt sich indirekt von der Identität des Clients, da nur der Client die Antwort entschlüsseln kann. Weiterhin muss sich der Client nur einmal gegenüber der AS ausweisen um andere Transaktionen

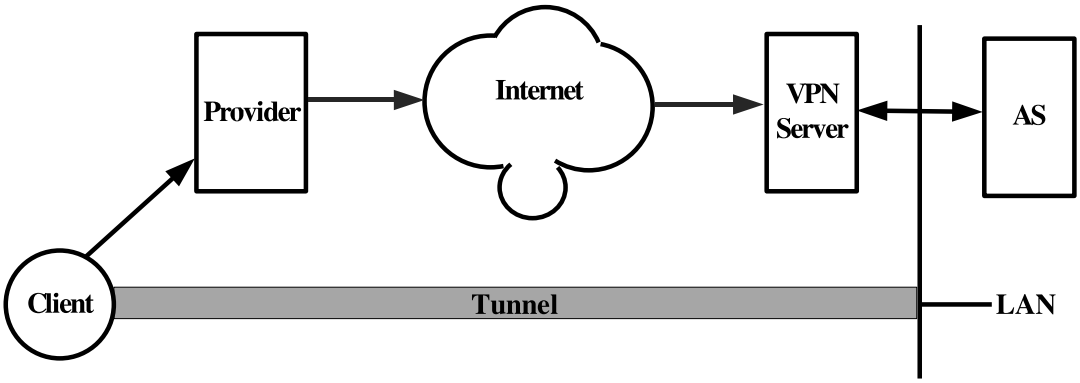

Abbildung 1.7: End-To-Site VPN

im Netz auszuführen. Dadurch muss der Client das Passwort nicht zwischenspeichern und ist vor Attacken wie Speicherabzug sicher. Für die Sicherheit von Kerberos ist natürlich die Geheimhaltung der Schlüssel notwendig. Die Passwörter der Benutzer und der Server müssen sicher im KDC gespeichert werden. Dazu ist es i.a. erforderlich das KDC physikalisch zu sichern [5]. Um Kerberos gegen Passwortraten zu sichern, sollten die Benutzer starke Passworte verwenden oder zusätzliche Mechanismen wie z.B. Smartcards benutzen. Weitere Sicherheit wird durch die begrenzte Gültigkeit der Tickets erreicht. Für das TGT sind das i.a. 8-10 Stunden und für die Servertickets 5 Minuten. Daraus folgt das Kerberos nur funktioniert wenn die Uhren auf allen Rechnern synchron gehen. Zumindest im europäischen Raum ist das durch die Funkzentraluhr unproblematisch. Auch kann die kurze Gültigkeit von 5 Minuten für die Servertickets zu Problemen führen wenn Transaktionen länger dauern [4].

1.2.6 Virtual Private Network

Virtual Private Network (VPNs) sollen sichere Verbindungen über ein unsicheres Medium (Internet) ermöglichen. In Abbildung 1.7 ist das Beispiel einer End-to-Site Verbindung gezeigt. Die End-to-Site Verbindung ist ein mögliches Szenario für VPNs. Andere Szenarien sind End-to-End oder Site-to-Site Verbindungen. Um VPNs zu ermöglichen, werden in der Praxis verschiedene Wege verfolgt. Eine Möglichkeit besteht darin, VPNs mit IPSec zu realisieren.

IPSec stellt Paketvertraulichkeit, Paketintegrität und Paketauthentizität sicher. Lokalisiert ist IPSec auf der Netzwerkebene des OSI-Referenzmodelles. IPSec unterscheidet zwei Betriebsarten, den Transportmodus und den Tunnelmodus. Im Transportmodus (Abbildung 1.8) werden i.a. die Daten des herkömmlichen IP-Paketes verschlüsselt und der IPSec-Header vor den IP-Header eingefügt. Der Transportmodus wird nur für End-to-End Verbindungen verwendet. Im Tunnelmodus (Abbildung 1.9) wird das komplette herkömmliche IP-Paket verschlüsselt und mit einem IPSec-Header versehen. Ein IP-Header wird dann dem Ganzen vorangestellt, so dass das IPSec-Paket im Tunnelmodus wie ein normales IP-Paket erscheint.

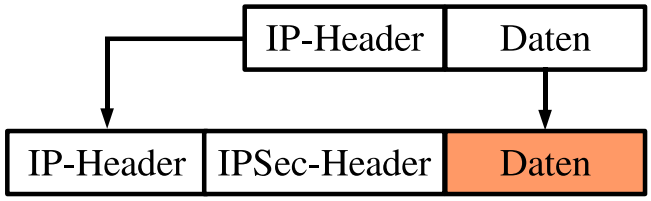

Abbildung 1.8: IPSec Paket im Transportmodus

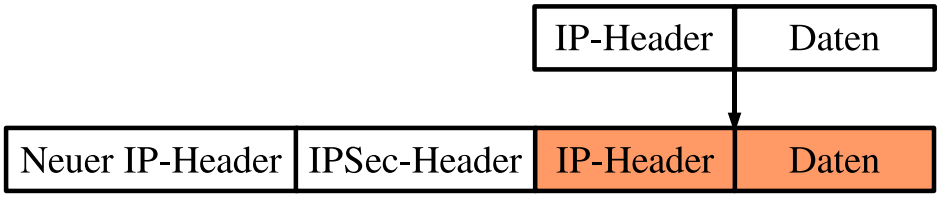

Abbildung 1.9: IPSec Paket im Tunnelmodus

Die IPSec-Architektur besteht aus einem Authentisierungs-Protokoll, dem Encapsulating-Security-Protokoll (ESP) und dem Key-Management (i.a. InterKey-Exchange-Protocol, IKE).
Das Authentisierungs-Protokoll soll Datenintegrität und Authentizität sicherstellen. Dazu wird das Datenpaket auf einen Hash-Wert abgebildet der als Teil des Authentication-Headers den Daten vorangestellt wird. Zur Erzeugung des Hash-Wertes werden gängige Verfahren wie MD5 oder SHA eingesetzt. Verwendet wird das Authentisierungs-Protokoll sowohl im Transport- als auch im Tunnelmodus. Der Sinn des Encapsulating-Security-Protokolls ist es, Datenvertraulichkeit sicherzustellen. Dazu werden im Transportmodus die Daten und der TCP/UDP-Header verschlüsselt und im Tunnelmodus zusätzlich noch der Applikationsheader. Weiterhin stellt dieses Protokoll wie oben auch Datenintegrität und Authentizität sicher. Im Key-Management wird die Verteilung und Verwaltung der Schlüssel geregelt, die für die Verwendung von IPSec notwendig sind.

Eine zweite Möglichkeit, VPNs zu realisieren, bietet das Secure-Soket-Layer (SSL). SSL geht ursprünglich auf eine Entwicklung von Netscape zurück. Das SSL-Protokoll ist zwischen der Transportschicht und der Applikationsschicht lokalisiert. Wie bei IPSec wird Datenvertraulichkeit, Datenintegrität und Authentizität sichergestellt.
SSL wird im Wesentlichen durch das Handshake-Protokoll, das Record-Protokoll und das Application-Data-Protokoll beschrieben. Im Handshake-Protokoll werden notwendige Kenntnisse, wie z.B. Verschlüsselungsverfahren, für die Kommunikation vereinbart. Weiterhin wird die Authentifizierung des Clients hier geregelt. Im Record-Protokoll werden die Daten höherer Schichten verschlüsselt und an die Transportschicht weitergegeben. Die Datenvertraulichkeit, Datenintegrität und Authentizität wird hier sichergestellt. Das Application-Data-Protokoll bereitet die Daten höherer Protokolle für die Record-Schicht auf und reicht sie an diese weiter. Der Vorteil dieser Struktur ist das nahezu jede höhere Anwendung auf Basis des Application-Protokoll implementiert werden kann und so

einfach SSL-Funktionalität erhält. Dadurch wird eine weitgehende Unabhängigkeit von Systemen oder Anwendungen erreicht, so dass es bspw. möglich ist, Weboberflächen über SSL-Clients, d.h. über verschlüsselte Verbindungen zu empfangen und zu nutzen.

Neben IPSec und SSL existieren viele weitere Softwarelösungen für VPNs wie z.B. das Point-to-Point-Tunneling-Protokoll (PPTP) oder das Layer-2-Tunneling-Protokoll (L2TP). Es werden auch spezielle Hardwareunterstützungen für VPNs angeboten, sowohl um die Performanz als auch die Sicherheit zu erhöhen. Beispiele sind spezielle VPN-Server der Firma Cisco oder Netzwerkkarten wie die Intel Pro/100S. Dieser Hardwarelösungen, wie Netzwerkkarten, eignen sich vor allem für Verschlüsselungen auf Schicht zwei des OSI-Referenzmodelles.

1.2.7 Digital Right Management

Digital Right Management (DRM) soll der Wegbereiter für die kommerzielle Nutzung des Internets sein. Bisher hat man zum Beispiel eine Gebühr für eine Kopie bezahlt und stimmte einer Lizenz zu, kurz: Pay-per-Copie. Für DRM-Anwendungen gilt das nicht mehr. Wir unterscheiden hier im wesentlichen drei Bereiche [8]:

- Pay-per-Instance: Die gegen Gebühr gekaufte Kopie ist an das persönliche Zertifikat des Nutzers gebunden. Eine Weitergabe der Kopie ohne das passende Zertifikat ist nutzlos. Wenn der Nutzer die Kopie weiter gibt, so dass ein Dritter die Kopie benutzen kann, so kann er sie selbst nicht mehr benutzen.

- Pay-per-Installation: Hier ist die Kopie an das Zertifikat des Computers gebunden.

- Pay-per-View: Die Kopie kann nur begrenzt oft benutzt werden, z.B. das Abspielen eines Musikvideos.

Um DRM-Anwendungen einzusetzen, müssen verschiedene Voraussetzungen erfüllt sein. Die Basis bildet Trusted Internet Traffic. Dazu ist es erforderlich, dass sichere Clients alle Anfragen mit unbekannten Zertifikaten ablehnen. Vermutlich werden die Internet Service Provider (ISP) dabei einen großen Anteil übernehmen. Im weiteren können die ISPs Zugangsprotokolle führen, um ggf. eine Strafverfolgung zu ermöglichen.

Mit DRM-Anwendungen verbinden sich aber auch mögliche Gefahren. Folgende Zitate lasse ich unkommentiert:

'Das Digital Rights Management von heute ist das Political Rights Management von morgen', John Perry Barlow, Künstler

'Die Frage der Information wird eine Frage des Budgets', Chaos Computer Club

1.3 Economic Impacts

Public Key Infrastructures stellen heute sichere Anwendungen bereit, um viele herkömmliche private, Geschäfts- oder Behördenvorgänge digital abzuwickeln. Die Nutzung dieser Technologien ist schnell, bequem, sicher und nahezu frei von Verwaltungskosten. Als Hauptproblem bei der Durchsetzung von kommerziellen Internetangeboten spielt die immer noch fehlende Kundenakzeptanz eine große Rolle. Viele Menschen sind sich der Möglichkeiten dieser Anwendungen gar nicht bewusst oder sehen diese als zu schwierig an. Weiterhin halten sich die Kunden beim Konsum im Internet vor allem an Produkte mit denen eine feste Vorstellung verbunden wird wie Bücher usw. Ein anderes Problem betrifft die oft noch unsichere Rechtslage (laut Gesetz müssen manche Behördendokumente über einen Zeitraum von hundert Jahren Verbindlichkeit und Integrität sicherstellen).

1.4 Zusammenfassung

Sichere Internetanwendungen sind notwendig, um eine Plattform für kommerzielle Anwendungsbereiche zu erschließen. Die Basis für sichere Internetanwendungen bilden kryptographische Verfahren, die aus heutiger Perspektive als sicher angesehen werden können. Für wie lange aber heutige starke Kryptographieverfahren auch in Zukunft noch sicher sind, ist schwer zu sagen. Ein Zeitraum von zehn Jahren ist sicher ein eher pessimistischer Wert, wenn man einen baldigen Paradigmenwechsel in der Rechnerarchitektur ausschließt. Auf den kryptographischen Verfahren aufbauend bilden die Public-Key-Infrastructures einen hinreichend konsistenten Rahmen, um alle relevanten Sicherheitsziele zu gewährleisten. Besonderes Augenmerk muss hier den Trusted Authorities gelten, bei denen besonders hohe Sicherheitsanforderungen notwendig sind. Die Verwendung von Zertifikaten im Rahmen kommerzieller Anwendungen wirft auch viele rechtliche Probleme auf, die erst noch bewältigt werden müssen. Die existierenden Anwendungen im Bereich der Public-Key-Infrastructures sind weit fortgeschritten. Man sollte aber nicht die mit der Verbreitung dieser Anwendungen zunehmende Energie, die ein potentieller Angreifer zu investieren bereit ist, außer Acht lassen. Wie die oft sehr emotional geführte Diskussion um bspw. DRM-Anwendungen zeigt, sind auch gesellschaftliche und soziale Auswirkungen zu berücksichtigen.

Literaturverzeichnis

[1] “Moderne Betriebssysteme“,Andrew S. Tannenbaum, Pearson Studium, 2. Auflage, 2002

[2] “Informatik-Handbuch“,Rechenberg, Pomberger, Hanser, 2. Auflage, 1999

[3] “Bedrohen Quantencomputer die heutige Verschlüsselung¿‘, Elektronik 20/2003

[4] “Betriebssysteme“, Rüdiger Brause, Springer, 2. Auflage, 2001

[5] “Kerberos, eine Frage des Vertrauens“, Peter Wächtler, Linux Magazin 05/1999

[6] “Digitale Signaturen“, Robert Gehring, Diplomarbeit TU-Berlin, 1998

[7] “Secure Shell“, Holger Trapp, 5. Mai 2004
http://www-user.tu-chemnitz.de/ hot/ssh

[8] “Digital Rights Management“, John Walker, 2004
http://www.heise.de/tp/deutsch/special/ende/16658/1.html

[9] “Neue VPN-Lösungen“, Atnarong Kongnin, Seminararbeit UniBwM, 2004

Kapitel 2

AAA Protokolle – Die Basis für kommerzielle Dienste

Dennis Möller

Dieses Dokument beschäftigt sich mit AAA (Authentication, Authorisation, Accounting) Protokollen, die eingesetzt werden, um die kommerziellen Aspekte und die Sicherheit des Internets zu gewährleisten. Der Schwerpunkt liegt dabei auf den Anforderungen und der Funktionsweise der Protokolle, ferner ihren Stärken und Schwächen und den daraus resultierenden Folgen für Internetunternehmen. Bei den Protokollen werden hauptsächlich RADIUS und DIAMETER als einzige existierende reine AAA-Protokolle behandelt, es werden aber auch alternative Ansätze und Protokolle, die Teile des AAA-Spektrums erfüllen, vorgestellt.

Inhaltsverzeichnis

2.1 Einführung

Der Bedarf und die Nutzung des Internets zur Bereitstellung kommerzieller Dienste, kommerziell genutzten Daten und als Kommunikationsmittel für Geschäftabwicklungen ist in den letzten Jahren stark angestiegen und wird auch weiter steigen. Als Folge erhöht sich der Bedarf die sensitiven Daten zu schützen, Dienste vor Mißbrauch zu sichern und Kosten zu berechnen [17]. Dies führt im wesentlichen zu drei Dingen die gefordert werden:

1. Authentifizierung (Authentication): Bei der Authentifizierung wird die Identität eines Nutzers der einen Dienst nutzen will bestätigt. Dies ist wichtig, da die Nutzung eines Dienstes meistens auf einen bestimmten Personenkreis beschränkt sein soll (meist die, die dafür bezahlt haben oder die zu einem bestimmten Unternehmen gehören) und da der Nutzer jederzeit verantwortbar gemacht werden können muß; sei es bei Mißbrauch oder sei es bei der Bezahlung des geleisteten Dienstes. Authentifizierung wird meist durch ein Geheimnis, das nur zwei Seiten (Dienstleister und Nutzer) kennen oder durch eine vertrauenswürdige dritte Instanz realisiert.

2. Authorisierung (Authorisation): Als Authorisation bezeichnet man die Erlaubnis einen Dienst nutzen zu dürfen. Nachdem sich ein Nutzer authentifiziert hat muß sichergestellt werden, daß er die Dienste für die er zugelassen ist auch Nutzen kann. Hinderlich hierbei wäre wenn der Nutzer sich jedesmal neu authentifizieren müßte. In der Praxis werden meist ACLs (Access Control Lists) oder Policies genutzt.

3. Abrechnung (Accounting): Bei der Abrechnung werden vom Dienstleister Daten zur Nutzung seiner Dienste gesammelt zwecks Auswertung für Rechnungen und Kapazitätsplanungen. Hier kommt der ökonomische Aspekt am meisten zur Geltung, da letztendlich der Dienstleister Geld verdienen möchte und somit den Nutzer zur Zahlung der geleisteten Dienste herangezogen wird. Es werden grundsätzlich meist die transferierten Daten oder die Zeit, die der Dienst genutzt wurde, gemessen und dem Nutzer in Rechnung gestellt.

Als Folge des oben Genannten wurden Protokolle entwickelt um diese Aufgaben zu übernehmen. Die Wichtigesten werden wir später vorstellen.

Für diese nun schon etwas älteren Protokolle ergeben sich allerdings neue Schwierigkeiten, denn die Netzwelt hat sich verändert und Errungenschaften wie WLAN, mobile Nodes und Ad-hoc Netzwerke mit ihren dynamischen Topologien führen zu Problemen mit den alten Protokollen. Auf der anderen Seite ist auch der Bedarf an Ad-hoc Netzwerken gestiegen, um Daten auch auf kurze Entfernung und sehr spontan (nämlich sofort dann, wenn sie plötzlich gebraucht werden) zu transferieren. Trotzdem muß auch hier die Sicherheit und damit die Authentifizierung und Authorisierung gewährleistet bleiben.

Stellen wir uns vor, wir steigen in ein ad-hoc Netzwerk in der Firma, in der wir arbeiten, ein, weil wir Daten einer bestimmten Abteilung brauchen. Es wäre fatal, wenn nun jeder andere im Netz, auch wenn er eine niedrigere Sicherheitsfreigabe von der Firma hat, auf unsere hochsensitiven Daten zugreifen kann. Und sollte es denoch jemand tun, so ist es natürlich wichtig zu wissen, wer es war.

In dem hier vorliegenden Dokument werden wir die Ansätze zur Realisierung der AAA, sowie die Funktionsweise einiger Protokolle erklären. Abschließend gehen wir auf den wirtschaftlichen Bezug ein.

2.2 Übersicht über AAA

2.2.1 Begriffsdefinitionen

AAA steht für Authentication, Authorisation, Accounting. Dies sind die drei Dinge, die am häufigsten im Zusammenhang mit Internetsicherheit gefordert werden und sie beziehen sich auf Dienste. Beispiele für Dienste sind die Einwahl ins Internet, eCommerce, Drucken via Internet, Mobile IP, Dateiserver, Unterhaltungsmedien. Nun wollen wir aber ersteinmal die Begrifflichkeiten im Sinne dieses Dokuments klären da diese sonst häufig anders interpretiert werden.

1. Authentifizierung (Authentication) ist der Vorgang der Verifizierung einer Identität, die ein Name eines allen teilnehmenden Seiten bekannten Namenraums, der Urheber einer Nachricht (message authentication) oder der Endpunkt eines Kanals sein kann [1].

2. Authorisierung (Authorisation) ist die Entscheidung ob ein bestimmtes Recht dem Inhaber einer bestimmten Glaubwürdigkeit (z.B. eine authentifizierte Identität) gestattet wird [1].

3. Abrechnung (Accounting) ist das Sammeln von Resourcenbenutzungs-und verbrauchsdaten zwecks Abrechnung von Gebühren, Prognosen und Kapazitätsplanungen.

2.2.2 Aktuelle Protokolle

Aktuelle Vorschläge für AAA-Protokolle kommen von der AAA Working Group [2], sowie von der AAA Architecture Research Group [3] des IETF, wobei letztere verantwortlich für die Entwickling einer generischen AAA Architektur ist. Die Ideen, die in den nächsten Abschnitten präsentiert werden basieren auf den Vorschlägen der AAA Working Group.

Das Ziel ist es ein Protokoll zu entwickeln das Authentication, Authorisation und Accounting realisiert [2]. Im Moment sind diese Aufgaben meist auf mehrere Protokolle aufgeteilt, was die Effizienz und Kompatibilität verringert. Einige wenige Protokolle, die alle drei Aufgaben alleine implementieren exisitieren. Es sind RADIUS [4] und DIAMETER [5] sowie Erweiterungen zu ihnen. Die Funktionsweise dieser Protokolle werden später vorgestellt.

2.2.3 Allgemeine Struktur

Meist werden die Dienste dem Nutzer in der home organisation oder home domain zur Verfügung gestellt (zum Beispiel ein Firmennetz), wobei die home domain meist dort ist wo sich der Nutzer die meiste Zeit aufhält.

Sollte der Nutzer nicht in der home domain sein, sondern in einer foreign domain und möchte trotzdem auf seine Daten zugreifen, so bekommt die Authentisierung und Authorisierung noch mehr Gewicht [6].

Die Struktur eines AAA-Systems sieht in den meisten Fällen vor, daß im Netz AAA-Server verteilt sind, die untereinander mit AAA-Protokollen kommunizieren. Diese AAA-Server führen die Authentifizerierung durch, geben Authorisierungen und sammeln die erforderlichen Benutzungsdaten. In dem Netz können weiterhin sogenannte Broker vorhanden sein, die als vertrauenswürdige dritte Instanz fungieren, wenn sich zwei Objekte gegenseitig nicht trauen.

2.2.4 Generische AAA-Struktur

Die aktuellsten Ideen beschäftigen sich mit einer generischen AAA-Struktur. Dabei ist angedacht die AAA Funktionalität in zwei Teile aufzuteilen:

- Generischer Teil: Dieser Teil der Struktur ist bei allen gleich
- Applikationsspezifischer Teil: Dieser Teil ist auf die anwendende Applikation zugeschnitten

In der angestrebten Realisisierung ist dann neben dem AAA-Server ein sogenannter ASM (Application Specific Module)-Server im Netz vorhanden. Dieser hat applikations-spezifische Informationen und er managt Resourcen und konfiguriert die Platform des Dienstes so, daß der gewünschte Dienst verfügbar wird. Desweiteren wird der ASM-Server aufgrund seiner applikations-spezifischen Informationen auch zur Authentifizierung, Authorisierung und Abrechnung verwendet.

Ein Event Log wird bei jedem Vorgang mitgeschrieben und kann für Entscheidungen herangezogen werden. Je nachdem ob ein Ereignis in der Vergangenheit stattgefunden hat oder nicht wird der Nutzer authorisiert oder nicht. Das Ereignis kann zum Beispiel eine Freischaltung oder Bezahlung von Gebühren sein in deren Abhängigkeit ein Dienst angeboten oder verweigert wird. Auch ein Fehlverhalten kann zum Beispiel die Sperrung eines Dienstes für einen Nutzer bewirken.

Ein Policy Repository enthält Informationen über verfügbare Dienste und Resourcen und über die Entscheidungsrichtlinien die auf die Authorisierung angewendet werden.

Es folgt ein Überblick über den Authorisierungsvorgang in der generischen AAA-Struktur:

1. Der Nutzer stellt eine formatierte Authorisationsanforderung (request) an den AAA-Server. Die Formatierung ist wichtig damit der AAA-Server die Anfrage sofort bearbeiten kann und nicht erst die Anfrage interpretieren muß, weil sie applikationsspezifisch ist.

2. Der Server prüft die Anforderung, erkennt welches Art der Authorisierung erwünscht ist und holt Richtlinien aus dem policy repository und führt eine der drei folgenden Möglichkeiten aus:
   (a) Die Anfrage wird an den ASM-Server weitergeleitet um ausgewertet zu werden.
   (b) Auf Grundlage der Richtlinien aus dem policy repository wird eine Entscheidung über die Anforderung getroffen (meist Annahme der Ablehnung).
   (c) Die Anforderung wird an einen weiteren AAA-Server weitergeleitet, der dann ebenfalls eine der drei Möglichkeiten ausführt. Dies kann so lange geschehen bis endlich ein AAA-Server eine Entscheidung für oder gegen die gewährung der Anforderung trifft.

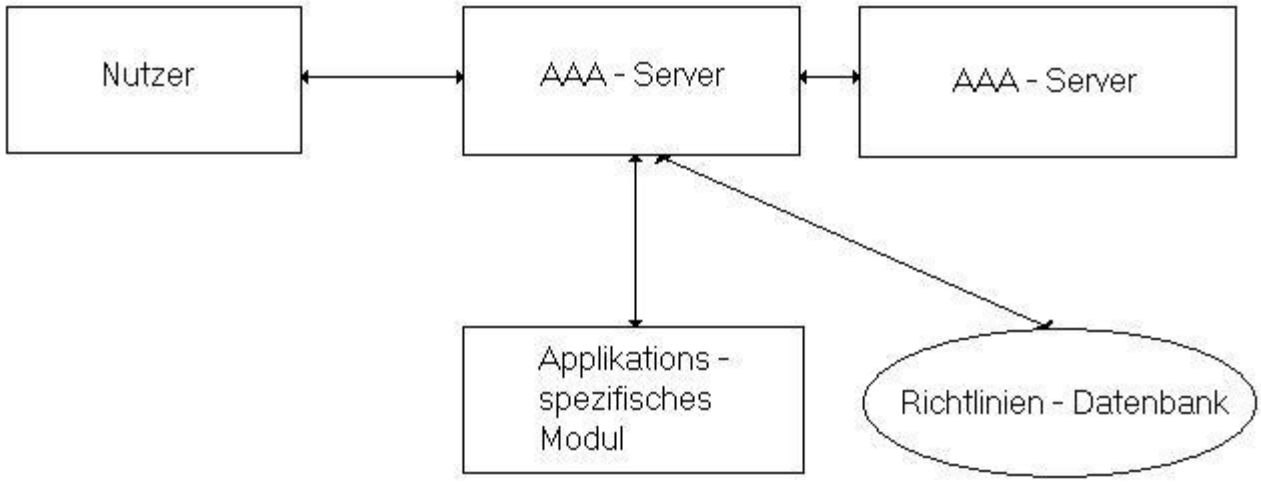

Abbildung 2.1: Funktionsweise einer generischen AAA-Struktur

Die Authentifizierung und Abrechnung sehen bei der generischen AAA-Struktur generell genauso aus.

2.3 Anforderungen an AAA-Protokolle

Dieser Abschnitt beschäftigt sich mit den allgemeingültigen Anforderungen an AAA-Protokolle wie sie auch von der AAA Working Group dargestellt werden.

Ein AAA-Protokoll kann in die drei Teile Authentication, Authorisation und Accounting aufgeteilt werden. Entsprechend werden die Anforderungen an diese Teile im folgenden einzeln präsentiert:

2.3.1 Anforderungen an die Authentifizierung (authentication)

Authentifizierung bedeutet daß eine Identität bestätigt wird und derjenige, der authentifiziert werden soll, derjenige ist, der er vorgibt zu sein. Es darf also keine große Wahrscheinlichkeit bestehen, daß sich jemand als jemand anders ausgeben kann und so die Rechte des anderen übernimmt und die Verantwortung für eventuelle Missetaten dem anderen übergibt. Eine absolute Sicherheit bei der die Wahrscheinlichkeit einem Täuschungsversuch zum Opfer zu fallen 0% beträgt ist praktisch unmöglich, doch sollte die Wahrscheinlichkeit möglichst gering sein.

Ein Authentifizierungsverfahren wird primär danach bewertet wie zuverlässig es bei der Identitätsfeststellung ist. Dazu muß es auch gegen gewollte und durchdachte Attacken (replay-, man in the middle-Angriffe) widerstandsfähig sein.

Es gibt verschiedene Ansätze die Authentifizierung durchzuführen. Die einfachste sind Passwörter. Diese besitzen allerdings den Nachteil, daß sie leicht herauszufinden sind. Größere Sicherheit bringt hier challenge response oder symetrische, sowie asymetrische Verschlüsselungsverfahren.

2.3.2 Anforderungen an die Authorisierung (authorisation)

Bei dem Vorgang der Authorisierung wird eintschieden ob einem Nutzer ein Recht gewährt wird oder nicht. Um diese Entscheidung fundiert treffen zu können und dabei keine Sicherheitsrisiken einzugehen muß vorher der Nutzer authentifiziert werden. Das Recht, das der Benutzer beantragt ist meist ein Zugriff auf bestimmte Daten oder der Zugang zu einem Netz. Die Entscheidung ob das Recht gewährt wird oder nicht wird anhand von Richtlinien (policies) gefällt. Wichtig bei der Authorisierung ist, daß die Richtlinien und die Entscheidung mit der dahinterstehenden Absicht konsistent ist. Es dürfen keine Personen authorisiert werden, die eigentlich keinen Zugang haben sollen. Auf der anderen Seite darf der Zugang auch nicht berechtigten Personen verwehrt werden. Dies ist dann meist auf widersprüchliche oder sich überschneidene Richtlinien zurückzuführen. Die policies müssen also eindeutig und klar gewählt und umgesetzt werden.

Ungewollte Lücken werden meist dadurch verhindert, daß der Ansatz “es ist nur erlaubt, was ausdrücklich erlaubt” ist gewählt wird anstatt der Philosophie “es ist nur erlaubt, was nicht verboten ist”.

Im folgenden wird ein zentralisiertes Modell vorgestellt wie es von der AAA Working Group vertreten wird [2]. Es gibt durchaus auch dezentralisierte Modelle, die sich besonderns für ad-hoc Netzwerke eignen.

In einem normalen Authorisationsvorgang können wir vier Rollen ausmachen:

- Der Nutzer, der den beantragten Dienst nutzen möchte und dafür authorisiert werden muß.
- Die home organisation, in der sich der Nutzer meist aufhält. Diese kennt unter Umständen Informationen die der Dienstanbieter nicht kennt und wird deswegen in den Vorgang eingebunden.
- Der AAA-Server der die Authorisation vornehmen soll.
- Der Dienst als solcher, sowie die darunterliegenden Resourcen. Dies können zum Beispiel Daten und der Server auf dem die Daten liegen sein [7].

Die Authorisationsanforderungen von AAA-Systemen wurden von Vollbrecht und Calhoun wie folgt definiert [8], [9]:

- Ein AAA-Protokoll sollte getrennte und muß kombinierte Authorisationsnachrichten unterstützen.
- Ein AAA-Protokoll muß von einem AAA-Server an einen anderen weitergeleitet werden können.
- Ein AAA-Protokoll muß zwischengeschalteten sogenannten Brokern ermöglichen ihre eigenen Sicherheitsinformation zu den Anforderungen und Antworten hinzuzufügen.
- Wenn Broker zwischengeschaltet sind muß Endpunkt-zu-Endpunkt-Sicherheit gewährleistet sein.
- Brokern muß es möglich sein die Weiterleitungsadresse an dem Anfragesteller mitzuteilen um eine direkte Kommunikation zwischen den beiden Endpunkten zu ermöglichen.

2.3.3 Exkurs: Authorisierungsmethoden

Es gibt verschiedene Methoden für den Authorisationsvorgang: Sie heißen push, pull und agent. Diese werden im folgenden vorgestellt [7]:

1. Agent

   Der AAA-Server fungiert als Vermittler zwischen Nutzer und Dienstserver. Jegliche Kommunikation vor der eigentlichen Dienstbenutzung erfolgt über den AAA-Server.

   (a) Der Nutzer stellt eine Benutzungsanfrage für einen bestimmten Dienst an den AAA-Server.

(b) Der AAA-Server wendet eine policy auf die Anfrage an. Wenn dem Antrag stattgegeben wird leitet der AAA-Server die Anfrage an den Dienstserver weiter. Dieser stellt dann den Dienst zur Verfügung.

(c) Der Dienstserver meldet dem AAA-Server, daß der Dienst bereit gestellt wurde.

(d) Der AAA-Server benachrichtigt den Nutzer, daß der gewünschte Dienst jetzt genutzt werden kann.

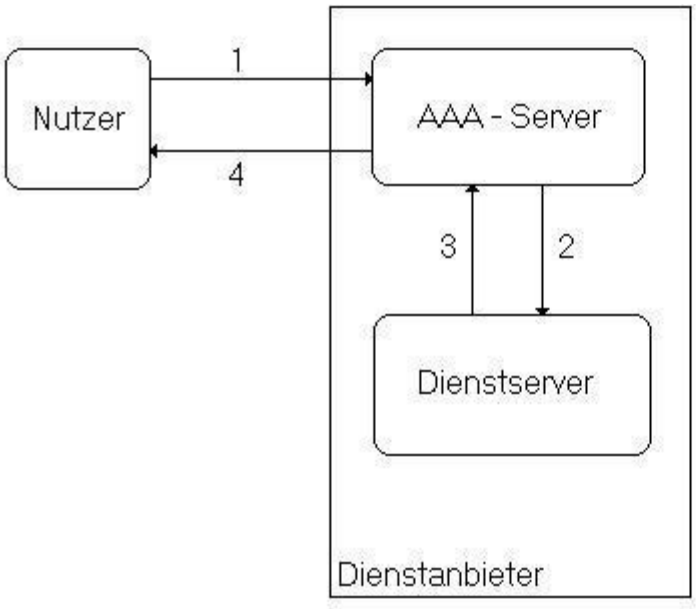

Abbildung 2.2: Agent-Methode

2. Pull

Der AAA-Server fungiert hier als "Berater" des Dienstservers. Der AAA-Server wird im Laufe des Authorisationsvorgangs vom Dienstserver gefragt, ob der Dienst dem Nutzer angeboten werden soll / darf.

Das Bild veranschaulicht die Authorisation mit der pull-Methode:

(a) Der Nutzer stellt seine Anfrage an den Dienstserver

(b) Der Dienstanbieter fragt den AAA-Server, ob der Nutzer für den Dienst zugelassen werden darf indem er die Anfrage an den AAA-Server weiterleitet.

(c) Der AAA-Server entscheidet gemäß seinen Richtlinien, ob der Nutzer zugelassen wird und sendet die Erlaubnis oder das Verbot als Antwort an den Dienstserver.

(d) Der Dienstserver stellt bei positiver Antwort den Dienst zur Verfügung und benachrichtigt den Nutzer, daß er den Dienst nun nutzen kann. Bei negativer Antwort antwortet er dem Nutzer entweder gar nicht oder er teilt ihm mit, daß sein Gesuch abgelehnt wurde.

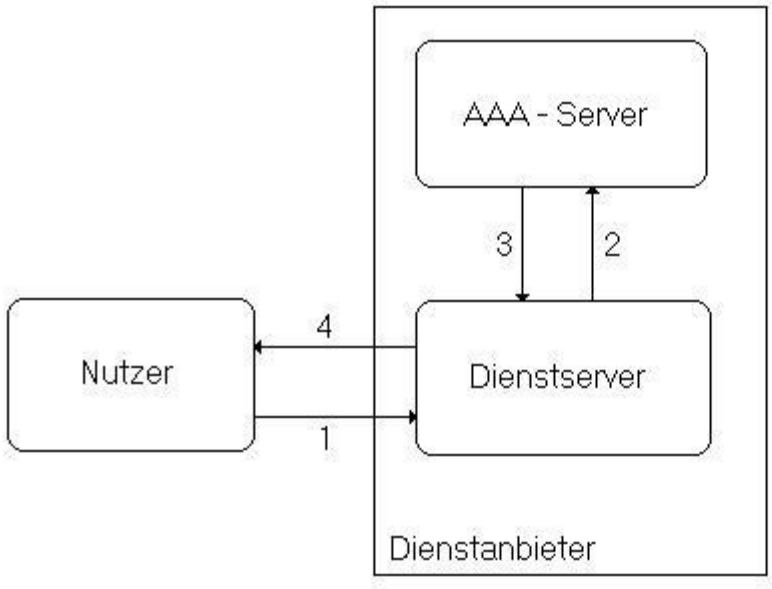

Abbildung 2.3: Pull-Methode

3. Push

Bei der push-Methode fragt der Nutzer den AAA-Server, ob er den Dienst benutzen darf. Wenn es ihm erlaubt wird, so erhält er ein sogenanntes Ticket, daß er dem Dienstserver präsentieren kann und das dem Dienstserver zeigt, daß der Nutzer vom AAA-Server authorisiert wurde. Die Schritte im einzelnen:

(a) Der Nutzer stellt seine Anfrage an den AAA-Server.

(b) Der AAA-Server entscheidet aufgrund einer policy ob dem Gesuch stattgegeben wird. Bei positivem Entscheid erhält der Nutzer ein Ticket, das nur vom AAA-Server erstellt werden kann.

(c) Der Nutzer sendet das Ticket als Beweis seiner Berechtigung den Dienst zu nutzen an den Dienstserver.

(d) Der Dienstserver prüft, ob das Ticket vom AAA-Server ausgestellt wurde. Wird das Ticket als gültig erkannt wird der Dienst bereit gestellt und der Nutzer benachrichtigt, daß der Dienst bereit steht.

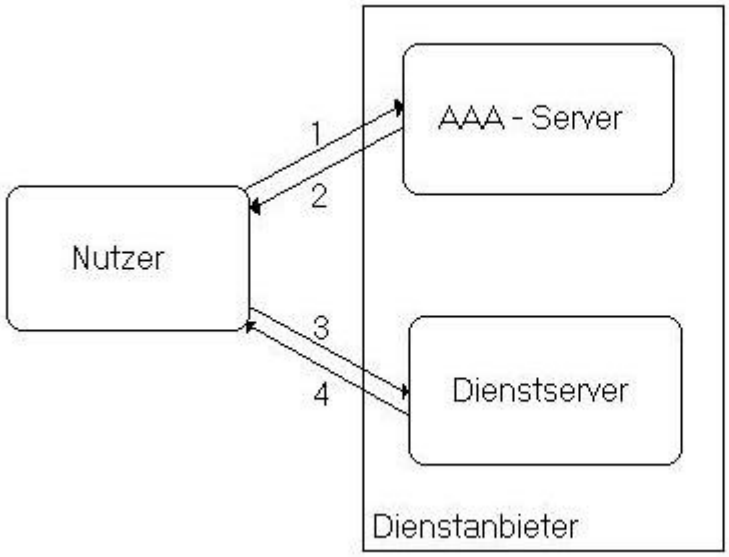

Abbildung 2.4: Push-Methode

2.3.4 Anforderung an Accounting

Beim Accouning werden Daten rund um Transfers und Dienste gesammelt. Diese Daten dienen dazu bei gebührenpflichtigen Diensten den Nutzern korrekte Rechnungen zu stellen, Trends aufzuzeigen und so Vorhersagen zu ermöglichen, um die Kapazitäten zu planen, Fehler und Ausfälle aufzudecken und die eigenen Kosten zu erfahren. Man kann das Accounting in zwei Bereiche unterteilen: Zum einen in das Intra Domain Accounting, das innerhalb administrativer Grenzen liegt und das Inter Domain Accounting, das administrative Grenzen überschreitet, wobei letzteres ein lohnenderes und anfälligeres Ziel für Angriffe und Betrugsversuche ist. Das macht die Sicherheit für das Inter Domain Accounting naturgemäß wichtiger. Daraus erwächst die Forderung nach Schutz gegen replay-Attacken, Integrität der Datenobjekte, Vertraulichkeit und Nicht-Zurückweisbarkeit. Das Inter Domain Accounting muß desweiteren mit der höheren Packetverlustrate umgehen und eventuellen zusätzlichen Sicherheitsmaßnahmen.

2.3.5 Trendanalyse und Kapazitätsplanung

Bei der Trendanalyse und Kapazitätsplanung ist ein gewisser Packetverlust nicht schwerwiegend, da sie nur benutzt werden um die zukünftige Auslastung und Resourcenbereitstellung zu planen. Deshalb ist eine Robustheit gegen eine mittlere Packetverlustrate im intra-domain Fall und gegen eine hohe Packetverlustrate im inter-domain Fall ausreichend. Das liegt daran, daß die Verlässlichkeit beim Datentransfer innerhalb der Domäne größer ist als bei dem außerhalb der Domäne. Der Aufwand die Verlässlichkeit außerhalb der Domäne zu erhöhen steht in keinem Verhältnis zu dem Nutzen, weshalb man sich hier mit der oben genannten Robustheit begnügt [10].

2.3.6 Rechnungserstellung

Die Abrechnung von Diensten läßt sich in zwei Teile Teilen: usage-sensitve und non-usage-sensitve.

Bei letzterem spielen die gesammelten Daten keine Rolle. Die Abrechnung erfolgt hier nicht über das in Anspruch genommenen Datentransfervolumen, sondern meist durch eine logische Unterteilung. Ein Beispiel wäre hier eine Flatrate bei der im Monat ein feststehender Betrag berechnet wird-unabhängig davon wieviel Datenverkehr von dem Nutzer in Anspruch genommen wurde und wie lange oder oft er mit dem Netz verbunden war. Ein weiteres Beispiel wäre video-on-demand oder ähnliches bei dem das Produkt nicht der Datentransfer, sondern der komplette (digitale) Film ist.

Bei usage-sensitive billing wird die Datenmenge oder Zeit gemessen, die der Nutzer transferiert hat beziehungsweise den Dienst in Anspruch genommen hat. Hier ist eine korrekte Messung der Daten naturgemäß äußerst wichtig, um zum einen dem Kunden nicht zu wenig zu berechnen, was dem eigenen Geschäft schaden würde, und zum anderen dem Kunden nicht zu viel zu berechnen, da man somit Kundenzufriedenheit und damit letztendlich die Kunden verliert und sich hier anfällig macht für Klagen von seiten der Kunden [10].

Desweiteren muß hier die Authentifizierung, Vertraulichkeit und besonders die Nicht-Zurückweisbarkeit verläßlich sein, denn hier liegt der wohl wahrscheinlichste Angriffspunkt. Es muß sichergestellt werden, daß die Verbindung zwischen einem Nutzer und seinen Gebühren einwandfrei und eindeutig festgestellt wird. Betrüger könnten hier versuchen, eine Identität vorzutäuschen um auf Kosten eines anderen den Dienst in Anspruch zu nehmen oder die Gebühren mit der Begründung, daß sie den Dienst gar nicht in Anspruch genommen haben zurückweisen oder die Abrechnung durch falsche Daten zu ihren Gunsten fälschen.

In der Praxis wird ein archivierender und verzögender Ansatz gewählt um das finanzielle Risiko zu minimieren, da man hier mehr Reaktionszeit und eine finanzielle Einlage hat [3].

2.3.7 Buchführung

Die Daten werden auch zur Buchhaltung des Dienstleisters genutzt. Sie müssen somit korrekt sein, um tatsächlich entstandene Kosten und Einnahmen beziehungsweise einzufordernde Gebühren zu berechnen. Dies ist für die Führung eines Unternehmens unerlässlich, zum einen, da man gewissen Personen die finanzielle Lage melden muß (Aktionären, Finanzamt), zum anderen, da man langfristig seine Kosten und Einnahmen so ausrichten muß, daß dabei ein Gewinn für einen selbst herauskommt. Eine falsche Abrechnung hier könnte zum Beispiel dazu führen, daß man die Gebühren senkt, obwohl man mehr Kosten hat als es einem die falsche Abrechnung glauben läßt oder daß man glaubt die Gebühren aufgrund zu hoch eingeschätzter Kosten nicht senken zu können und somit gegenüber der Konkurrenz zu teuer ist, was meist zu Kundenverlust führt. Dies führt dazu, daß

diese Daten annährend ebenso korrekt, sicher und zuverlässig sein müssen wie bei der Rechnungserstellung.

2.3.8 Abgleich und Verlässlichkeit der Abrechnung

Bei der Abrechnung gibt es drei Bereiche, die beachtet werden müssen: Fehlertoleranz, Resourcenbedarf und das Modell, nach dem Daten gesammelt werden.

- Fehlertoleranz:

  Aufgrund der Tatsache, daß es bei der Abrechnung meist um Geld geht und ein Versagen der Abrechnung in Verlusten resultiert ist die Fehlertoleranz sehr niedrig auszulegen. Typische Gründe für ein Versagen der Abrechnung sind: Packetverluste, Netzwerk-und Serverausfälle und Neustarts der Hardware. Die Möglichkeiten diesen Fehlerquellen zu begegnen sind die gleichen, die aus den entsprechenden Dokumenten oder Vorlesungen zu Netzwerken allgemein ersichtlich sind. Hinzu kommt, daß man zum Beispiel nicht flüchtige Speicher benutzt und häufige Backups anlegt.

- Resourcenbedarf:

  Die Abrechnung des Resourcenverbrauchs verbraucht selber Resourcen. Die wichtigsten Resourcen, die die Abrechnung benötigt sind Bandbreite, flüchtigen und nicht flüchtigen Speicher und CPU-Zeit. All diese Resourcen haben Einfluss auf die Leistung und Zuverlässigkeit des Gesamtsystems. Damit die Abrechnung selbst möglichst wenig Resourcen verbraucht kann man die Abrechnung optimieren, zum Beispiel in dem man Daten bündelt oder die Daten bei geringer Netzauslastung anstatt bei Stoßzeiten sendet.

- Modelle:

  Es gibt verschiedene Modelle zur Datensammlung. Vier davon sind polling, event driven no batching, event driven with batching und event driven with polling.

  Bei ersterem sendet die Stelle, die die Daten sammelt und verarbeitet in regelmäßigen Abständen oder bei Bedarf Anfragen an die Messtellen und fragt nach Daten an, die dann-wenn vorhanden-geschickt werden.

  Bei event driven no batching schickt die Messtelle sobald sie Daten zum senden hat. Dies hat den Nachteil, daß hier kein batching betrieben wird.

  Quasi als Weiterentwicklung gibt es dann event driven with batching. Bei diesem Verfahren warten die Messtellen bis ein bestimmtes Datenvolumen sich angehäuft hat und schickt es dann gebündelt an den Accounting Manager.

  Bei event driven with polling wartet der Accounting Manager auf ein bestimmtes Ereignis. Tritt dieses ein, so wird eine Sendeaufforderung an die Messtelle bezeihungsweise Messtellen gesendet. Ereignisse können beispielsweise das Eintreten einer günstigen Netzauslastung oder eine Datenaktualisierung beim Accounting Manager sein. Messtellen können ein Ereignis senden, wenn zum Beispiel eine gewisse Zeit überschritten wurde oder ein Packet bereit liegt. Der Accounting Manager entscheidet dann, wann er die Messtelle zum Senden auffordert [10].

2.4 Existierende Protokolle

Im Moment existieren lediglich zwei wirkliche AAA-Protokolle: RADIUS [4] und DIAMETER [5]. Wir werden allerdings auch ISAKMP/IKE und SASL, die lediglich Authentifizierungsverfahren enthalten, und Kerberos, das Authentifizierungs-und Authorisierungsverfahren bereitstellt, vorstellen. Am Ende werden wir noch auf die ungewöhnlichen Ansätze von PolicyMaker und KeyNote eingehen.

2.4.1 RADIUS

Motivation

Network Access Server sind bei großen Providern häufig mit vielen anderen Dingen beschäftigt, wie zum Beispiel Konfigurierung, so, daß eine zusätzliche Belastung durch Authentifizierung, Authorisierung und Abrechnung zu einer Verzögerung des gesamten Netzverkehrs führen kann. Desweiteren haben große Provider Unmengen an Nutzern und damit an Daten zu verwalten. Es ist wünschenswert, daß diese zentralisiert sind, da sich die Daten laufend ändern und trotzdem eine Konsistenz im Netz gewährleistet sein muß. RADIUS entspricht diesen Anforderungen und bietet dazu noch einen gewissen Schutz gegen Sniffing oder aktive Angriffe. Durch seine weite Verbreitung liefert es auch ein weiteres Argument für sich, nämlich, daß Wissen und Ausgereiftheit, sowie Varianten für die meisten Fälle reichlich vorhanden sind.

Allgemeines

Das RADIUS-Protokoll wurde entwurfen um Authentifizierungs-, Authorisierungs- und Accountingdaten zwischen einem NAS (Network Access Server) und einem RADIUS-Server zu transportieren. Typischerweise wird es bei Modempools benutzt, die verwendet werden um einen bestimmten Dienst von außerhalb in Anspruch zu nehmen. Der RADIUS-Server speichert dabei alle relevanten Nutzerdaten in nur einer Datenbank, was hilft die Übersichtlichkeit zu bewahren. RADIUS benutzt das pull-Verfahren. Der NAS agiert als Client und sendet jegliche Nutzerdaten und Anforderungen an den RADIUS-Server und verhält sich dann der Antwort des RADIUS-Servers entsprechend. Der RADIUS-Server übernimmt dann die Aufgabe die Anforderungen zu bearbeiten, den Nutzer zu authentisieren und gegebenenfalls zu authorisieren und den Client so zu konfigurieren, daß der spezielle Nutzer den Dienst nutzen kann.

Die Sicherheit der Mitteilungen zwischen NAS und RADIUS-Server wird durch ein Shared Secret Verfahren gewährleistet bei dem ein Geheimnis, das nur diese beiden kennen, zur Authentifizierung benutzt wird. RADIUS benutzt weiterhin Verschlüsselung, um zum Beispiel Nutzerpasswörter zu übermitteln und es verwendet verschiedene Verfahren zur Authentifierung wie beispielsweise PAP (Password Authentication Protocol) und CHAP (Challenge Handshake Authentication Protocol). Das RADIUS-Protokoll kann leicht erweitert werden, da es 3-tupel, die aus Attribut, Länge und Wert bestehen verwendet und ein Anhängen eines weiteren 3-tupels das Protokoll nicht stört.

Der Vorgang der Authentifzierung sieht bei RADIUS wie folgt aus: Der Nutzer möchte einen Dienst nutzen. Mit der Anfrage schickt er seine Authentifizierungsdaten (Nutzername, Passwort, Client-ID und Port-ID) an der Dienstleister. Möchte dieser den Nutzer authentisieren, so schickt er die Anfrage an den RADIUS-Server. Ist dieser nicht erreichbar kann sich der RADIUS Client auch an einen anderen RADIUS-Server wenden. Ist der Server für den Dienstleister nicht zuständig, so leitet der Server die Anfrage weiter. Wenn der zuständige RADIUS-Server die Anfrage erhält prüft er die Identität der Nutzers anhand eines Vergleichs der Authentifizierungsdaten aus der Anfrage mit denen aus seiner Datenbank. Ist der Nutzer authentisiert, so prüft der RADIUS-Server anhand seiner Datenband welche Rechte dem Nutzer zustehen und ob die angeforderten darin enthalten sind. Ist der Nutzer authorisiert schickt der Server eine Nachricht (Access Accept Resonse), sowie Konfigurationsdaten an den Dienstleister, der dann den Nutzer benachrichtigt sobald der Dienst zur Verfügung steht.

Protokollaufbau

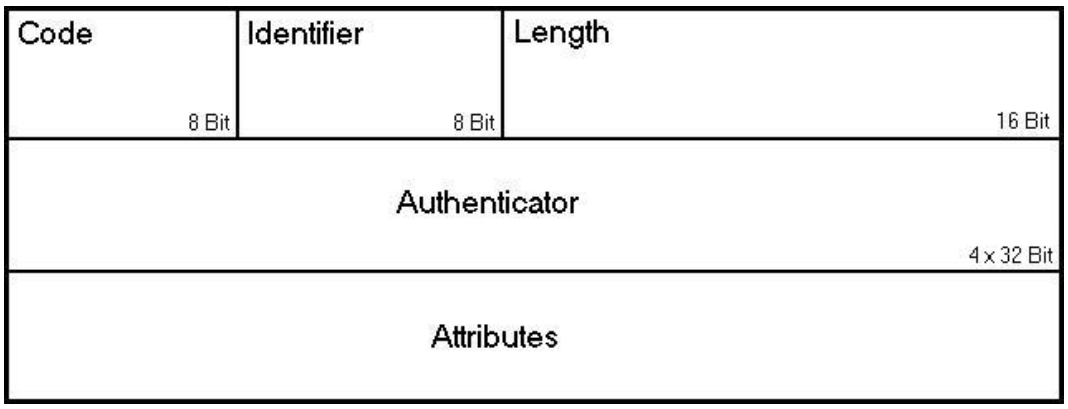

Abbildung 2.5: RADIUS - Protokoll Packet

Zur Erklärung [18]:

- Der Code besteht aus 8 Bit. Diese bestimmen die Art des RADIUS - Packets:
  - 1 : Access-Request
  - 2 : Access-Accept
  - 3 : Access-Reject
  - 4 : Accounting-Request
  - 5 : Accounting-Response
  - 11 : Access-Challenge
  - 12 : Status-Server (experimentell)
  - 13 : Status-Client (experimentell)
  - 255 : *reserved*

- Der Identifier besteht auch aus 8 Bit und dient dazu, dem Packet eine Identität zu geben. Für jedes Verfahren, das mit dem RADIUS-Server durchgeführt wird, wird eine ID vergeben, damit Anfrage und Antwort als zusammengehörig erkannt werden können. Implementiert ist der Identifier meist als ein einfacher Zähler.

- Das Attribut-Feld enthält zusätzlich Attribute. Ihre Anzahl ist freiwählbar, jedoch sind der Name und das Passwort des Nutzers unumgänglich.

- Die Länge gibt an, wie lang das gesamte Packet ist. Dies kann sich durch die Anzahl der Attribute ändern [18].

Vorgang der Authentifizierung

Der Client, der einen Dienst zur Verfügung stellt, erhält vom Nutzer eine Anfrage nach einem bestimmten Dienst. Dem Client werden dabei Passwort und Name des Nutzer übermittelt. Dieser leitet in einem Access-Request-Packet die Daten an den RADIUS-Server weiter. In Authenticator-Feld wird ein zufälliger 16 x 8 Bit String geschrieben. Das Packet bleibt bis auf das Passwort ungesichert. Das Passwort wird durch ein shared secret zwischen Client und RADIUS-Server geschützt. An dieses wird der Request-Authenticator herangehängt und das ganze läuft durch ein MD5-Hash, was ein 16 x Bit Ergebnis liefert. Mit diesem und des von dem Nutzer angegebenen Passworts wird eine XOR-Verknüpfung durchgeführt.

Nachdem das Packet abgeschickt wurde kommt es bei dem RADIUS-Server an und dieser prüft, ob er mit dem Client ein Shared Secret teilt. Wenn dies nicht so ist wird das Packet ohne Benachrichtigung verworfen. Sollte der Server ein Shared Secret haben, so benutzt er dieses um das Passwort in Klartext zu übersetzen. Der Server prüft Passwort und Nutzername mit Übereinstimmungen in seiner Datenbank. Im negativen Fall antwortet der Server dem Client mit einem Access-Reject. Im positiven Fall wird ein Access-Accept Packet an den Client gesendet. Beide Packete unterscheiden sich lediglich im Codefeld. Der Indentifier ist der gleiche wie bei dem Access-Request Packet. Der Response-Authenticator ist bei beiden Antworten das MD5-Hash vom Response-Packet mit dem dazu gehörigen Request-Authenticator konkateniert mit dem Shared Secret. ResponseAuthenticator = MD5(Code+ID+Length+RequestAuth+Attributes+Secret) wobei + eine Konkatenation darstellt. Sobald der Client das Packet erhält vergleicht er den Identifier mit den Identifieren, die zu bis dato ausgebliebenen Packeten gehören. Steht keine Antwort mit dem angegebenen Identifier aus, so wird das Packet stillschweigend verworfen. Der Response-Authenticator wird durch den Client mit demselben Verfahren wie beim Server erstellt und mit dem angekommenen verglichen. Sind diese unterschiedlich, so läßt der Client auch dieses ohne Rückmeldung im – sit venia verbo – "Datennirvana" verschwinden. Bei Gleichheit wird die Antwort ausgewertet und der Nutzer bekommt die Antwort und wird authentifiziert beziehungsweise zurückgewiesen.

2.4.2 DIAMETER

DIAMETER ist eine Entwicklung die auf RADIUS basiert und abwärts- kompatibel ist. Das DIAMETER-Protokoll ist für die Kommunikation zwischen ISPs (Internet Service Provider) und Firmennetzwerken gedacht [5]. DIAMETER besteht aus Basisprotokoll und applikationsspezifische Erweiterungen. Es gibt bereits Erweiterung für Mobile IP, NASREQ, Abrechnung und hohe Sicherheit. DIAMETER ist als peer-to-peer-Protokoll angelegt; das heißt, daß jede Netzwerkknoten eine Anforderung senden kann und alle gleichberechtigt sind. Es werden zwei Weiterleitungsmethoden unterstützt:

- Weiterleiten als Proxy: Hier werden die Daten einfach von einem Knoten zum nächsten weitergeschickt.
- Weiterleiten als Broker: Bei dem Broker-Verfahren schalten sich die Broker so dazwischen, daß eine direkte Verbindung zwischen den Endpunkten möglich wird.

DIAMETER verwendet sogenannte AVPs (Attribute Values Pairs). Ein AVP besteht aus drei Hauptfeldern: AVP Code, AVP Länge und AVP Daten, wobei der Code aussagt, was für Daten das AVP enthält, die Länge, die Länge des Datenfeldes enthält und das Datenfeld die eigentlichen Nutzdaten enthält. Jegliche Daten werden in dieser Form versendet. Jeder Knoten kann eine Anfrage oder Anforderung an einen anderen Knoten senden. Meist besteht der Nachrichtenverkehr aus Anfrage / Antwort. Unter Umständen kann es vorkommen, daß nur eine Anfrage gestellt wird. DIAMETER wird größtenteils dazu verwendet über Resourcen zu verhandeln, zu klären wie Nachrichten gesendet und beantwortet werden und wie einzelne Mitglieder aufgegeben werden [5].

2.4.3 ISAKMP

ISAKMP (Internet Security Association and Key Management Protocol) ist ein Framework für Authentifizierung und für das Managen und Einrichten von SA's (Security Association). Es definiert allerdings kein eigentliches Protokoll, das benutzt wird. Eine SA ist eine Absprache zwischen zwei Kommunikationspartnern welche Verfahren benutzt werden um die Sicherheit beim Datentransfer zu gewährleisten. Desweiteren definiert es die Informationen die für den Einsatz der abgesprochenen Sicherheitsmaßnahmen erforderlich sind.

ISAKMP richtet eine SA in zwei Phasen ein.

- Die erste Phase wird ISAKMP benutzt um den weiteren Datenverkehr, der für das komplette Einrichten nötig ist, zu sichern.
- In der zweiten Phase richtet ISAKMP ein anderes Sicherheitsprotokoll wie zum Beispiel IPSec ein und startet es. Viele Sicherheitsprotokolle sind nicht in der Lage sich selbst einzurichten und zu starten. Das ist der Hauptgrund, wo ISAKMP eingesetzt wird. Eine ISAKMP SA kann mehrere SA's mit Sicherheitsprotokolle einrichten. Das

hat den Vorteil, daß wenn eine ISAKMP SA ersteinmal eingerichtet ist und läuft, dann kann es genutzt werden um viele weitere Verbindungen zu sichern. Phase 1 muß also nur einmal durchlaufen werden, der Überhang der Phase 1 wird minimiert, weil beispielsweise der Authentifizierungsprozess nur einmal abgearbeitet werden muß.

Bei ISAKMP sind die Rollen der Kommunikationspartner nicht festgelegt und können von einer Phase zur anderen wechseln [11].

2.4.4 IKE

IKE steht für Internet Key Exchange und ist eine Implementierung die auf dem ISAKMP-Framework basiert. Die Aufgabe von IKE ist der sichere Austausch von Schlüsseln über das Netz. Es kann eingesetzt werden, um eine IPSec gesicherte Verbindung aufzubauen. Um die Sicherheit beim Transfer von Schlüsseln zu gewährleisten wird der Diffie-Hellmann-Algorithmus verwendet.

IKE übernimmt den Ansatz der zwei Phasen von ISAKMP. In der ersten Phase wird die IKE SA eingerichtet und in der zweiten Phase werden die Schlüssel ausgetauscht und das eigentliche Sicherheitsprotokoll eingerichtet und gestartet [12].

2.4.5 SASL

SASL (Simple Authentication and Security Layer) ist ein Prinzip um Authentifizierungsverfahren in verbindungs-orientierte Protokolle einzufügen. Das Protokoll muß dazu einen Befehl enthalten der den Nutzer auffordert sich beim Server zu authentifizieren und der den Namen des zu verwendenden Authentifizierunsgverfahren enthält. Dieser Name muß bei der IANA (Internet Assigned Number Authority) registriert sein. Wenn dem Server, das Authentifizierungsverfahren bekannt ist startet er die den Authentifizeirungsprozess gemäß des Verfahrens. Der Prozess beinhaltet challenge-response-Paare, die für jedes Authentifizierungsprotokoll spezifisch ist. Als Beispiel für ein solches Authentifizierungsprotokoll werden wir im nächsten Abschnitt Kerberos vorstellen.

SASL hat zusätzlich die Möglichkeit vor Beginn der Authentifizeirung eine zusätzliche Sicherheitsschicht zu benutzen. Der Datenverkehr der Authentifizierung findet dann in der zusätzlichen Sicherheitsschicht statt und aller nachfolgende Kommunikation findet verschlüsselt statt.

Die Spezifikation von SASL definiert vier verschiedene Mechanismen: Kerberos 4, GSS-API, S/Key und externe Mechanismen wie zum Beispiel IPSec oder TLS [13].

2.4.6 Kerberos

Kerberos ist ein dreiköpfiger, blutrünstiger und schlangenumwundener Hund der griechischen Mythologie, der den Eingang des Hades-der Unterwelt-bewacht.

Zum anderen ist Kerberos ein Netzwerkauthentifizierungs-und authorisierungsprotokoll, das an dem MIT (Massachusetts Institute of Technology) entwickelt wurde. Es ist frei und sogar als Quellcode erhältlich; es gibt allerdings auch zahlreiche kommerzielle Varianten.

Kerberos geht von folgenden Annahmen aus: Das Netzwerk ist unsicher, IPs, IDs, etc. werden nicht beachtet und die Schlüssel werden geheimgehalten.

Kerberos schafft im Netz eine dritte vertrauenswürdige Insatz, das Key Distribution Center (KDC). Dieses kann Tickets austellen, welche zur Benutzung eines bestimmten Dienstes berechtigen. Sie gelten aber nur für einen Dienstleister und einen Dienst und sind zeitlich begrenzt.

Das KDC besitzt eine Datenbank, in der sämtliche Daten zu den Nutzern gehalten werden. Schlüssel werden bei Kerberos oft aus Passwörtern des Nutzers generiert.

Der Ablauf einer Authentifizierung und Authorisierung sieht wie folgt aus:

1. Nutzer fordert ein Ticket vom KDC an.
2. KDC schickt eine Antwort. Die Antwort wird mit einem Schlüssel chiffriert. Dieser Schlüssel wird aus dem Passwort des Nutzers generiert. Da dieses Passwort nur dem KDC und dem Nutzer bekannt sind, sind diese die einzigen, die diesen Schlüssel kennen.
3. Nutzer erhält Antwort. Nur wenn er der ist, der er vorgibt zu sein, besitzt er sein geheimes Paßwort und kann die Antwort entschlüsseln. Alle anderen potentiellen Empfänger können mit der Antwort nichts anfangen. Somit authentifizert sich der Nutzer dadurch, daß er entschlüsseln kann. Nur das KDC kann die Antwort verschlüsseln, somit ist das KDC für den Nutzer authentifiziert. Die Antwort erhält einen Sitzungschlüssel, das Ticket und ein Nonce (eine Art Zeitstempel).

4. Ticket ist mit dem Schlüssel des Dienstleisters verschlüsselt. Der Nutzer hat zwar das Ticket, kann dieses aber nicht entschlüsseln. Das Ticket enthält auch den Sitzungschlüssel. Der Nutzer schickt nun das Ticket und einen Authentifizierer (Name, Adresse, Nonce) verschlüsselt mit dem Sitzungsschlüssel an den Dienstleister. Der Server dechriffiert mit seinem Schlüssel das Ticket und kommt somit auch an den Sitzungsschlüssel. Nun kann er den Authentifizerer entschlüsseln und kennt den Nutzer. Dienstleister und Nutzer können nun über eine mit dem Sitzungschlüssel gesichtere Verbindung kommunizieren.

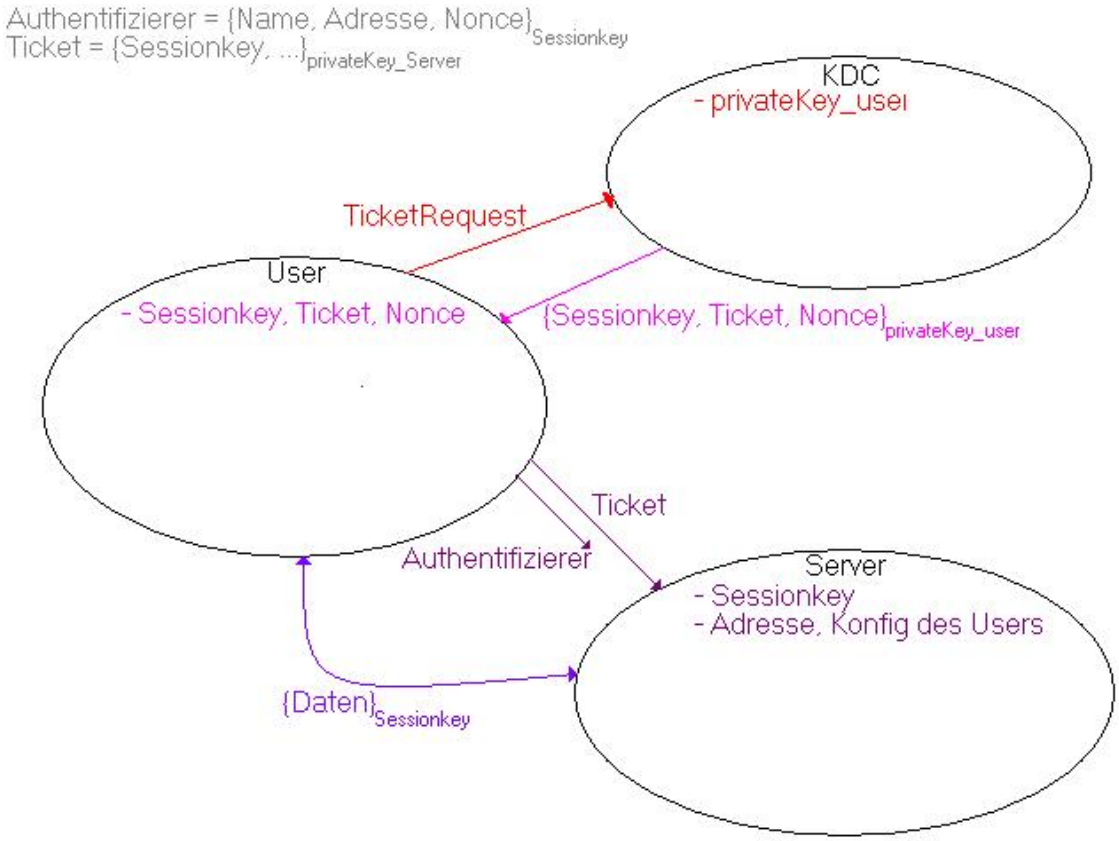

Abbildung 2.6: Kerberos

Optional kann der Server sich noch mit dem Senden des mit dem Sitzungsschlüssel verschlüsselten Nonce an den Nutzer bei diesem authentifizieren.

Mögliche Angriffspunkte sind bei Kerberos, wenn die Schlüssel Unbefugten in die Hände fallen, wenn Passwörter kompromittiert werden oder das KDC in feindliche Hände fällt. Die Kompromittierung von Passwörtern durch Password-Guessing oder ähnliches ist deshalb gefährlich, da die Source von Kerberos frei erhältlich ist und somit jeder weiß, wie aus den Passwörtern die Schlüssel generiert werden. Als Folge dessen sind hier starke Passwörter äußerst wichtig [14].

2.4.7 Vertrauensmanagementsysteme

Ein Vertrauensmanagementsystem bietet die Möglichkeit Richtlinien, Beglaubigungen und Beziehungen im Netz zu spezifizieren und zu interpretieren. Sie verknüpfen Schlüsseln direkt mit der Authorisation für bestimmte Aktionen, im Gegensatz zu den älteren Systemen wie zum Beispiel X.509, die auf Zertifikatbasis arbeiten, und Schlüssel zu Identitäten zuordnen. Ein Nutzer hat also bestimmte Rechte wenn er einen bestimmten Schlüssel vorweisen kann, während bei den Zertifikatsystemen die Rechte erst gebilligt werden, wenn der Nutzer identifiziert ist und seiner Identität das Recht zugeordnet wurde. Das macht die Vertrauensmanagementsysteme unkomplizierter.

Ein Vertrauensmanagementsystem besteht aus fünf Teilen:

- Eine Sprache, die die Aktionen beschreibt, die authorisiert werden sollen.
- Eine Möglichkeit, die Nutzer zu identifizieren, die Rechte verlangen.
- Eine Sprache, mit der die Richtlinien, nach denen die Gewährung von Rechten entschieden wird, beschrieben werden können.
- Ein Modell, daß die "Beglaubigungen" beschreibt, mit denen Vertrauen und Rechte innerhalb des Netzes delegiert werden können.
- Eine Prüfinstanz, die anhand der Richtlinien und den Beglaubigungen entscheidet, ob ein Gesuch angenommen oder abgelehnt wird.

Dieser Ansatz hat dann große Vorteile, wenn die Teilnehmer und die sicherheitsrelevanten Knoten und Daten im Netz stark verteilt sind. Ein Dienstanbieter kann sich mit wenig Aufwand und Datenverkehr an eine Prüfinstanz wenden und anfragen, ob eine bestimmte Aktion erlaubt ist oder nicht [15].

KeyNote2

Ein flexibles und einfaches Vertrauensmanagementsystem ist KeyNote. Seine Stärke liegt in dem einfachen Aufbau, der es ihm ermöglicht aufgrund seiner Geschwindigkeit auch bei Echtzeitanwendungen eingesetzt zu werden. Es kann sowohl in großen und in kleinen Netzwerken arbeiten. Die Verwendeten Sprachen und Modelle sind auch für den Menschen verständlich, was eine Programmierung und Wartung erleichtert. Es codiert Aktionen als Schlüssel-Wert-Paare. Die Applikation erstellt eine Liste von Schlüssel-Wert-Paaren. Nutzer werden mit IDs beschrieben. Ein Nutzer kann ein Mensch, ein Prozess, ein Netzwerkknoten oder eine andere sinnvolle Abstraktion sein.

Die Nutzer können entweder ein bestimmtes Recht anfordern oder sie können andere Nutzer zu Aktionen in ihrem eigenen Bereich ermächtigen. Dazu wird eine Liste an die Prüfinstanzen geschickt, die enthält von wem sie erstellt wurde und welche Nutzer welche Aktionen durchführen dürfen. Wird jetzt eine Anfrage gestellt, so kann der Nutzer, der den Dienst anbietet, die Prüfinstanz fragen, ob die Aktion erlaubt werden soll oder nicht.

Die Prüfinstanz sendet dann im einfachsten Fall eine Ja/Nein-Antwort, kann aber auch andere Werte senden, wie zum Beispiel, die Umstände unter denen die Aktion erlaubt wird oder die Anforderung von zusätzlichen Information.

In dem Netz existiert noch ein spezieller Knoten, der alle Aktionen ausführen darf. Dieser wird "Policy" genannt.

Um diese Listen fälschungssicher zu machen und notfalls auch durch unsichere Netze zu schleusen können die Nutzer ihre Listen mit einem Schlüssel signieren [15].

PolicyMaker

PolicyMaker ist ein weiteres Vertrauensmanagementsystem, Es benutzt zur Definition und Beschreibung von Richtlinien, Vertrauensverhältnissen und Rechten eine sichere Programmiersprache. Es ist flexibel genug um auch in großen Netzwerken und neben anderen Protokollen operieren zu können. Die Netztopologie besteht in der Regel aus mehreren Knoten, die alle lokal Entscheidungsgewalt haben und für ihren Bereich Rechte billigen oder nicht billigen. Die Mechanismen zur Entscheidungsfindung sind von den Richtlinien getrennnt. Somit können für die Entscheidungsfindung mehrere verschiedene Mechanismen zum Einsatz kommen, je nachdem, welcher am besten paßt und trotzdem folgen alle den selben Richtlinien. Von außen erscheint PolicyMaker wie eine Datenbank. Es nimmt als Eingabe Beglaubigungen, die lokalen Entscheidunsgrichtlinien und einen String der die geforderte Handlung definiert. Es entscheidet dann anhand der gegebenen Informationen und gibt entweder eine einfache Ja/Nein-Antwort oder, kann aber auch andere Werte senden, wie zum Beispiel, die Umstände unter denen die Aktion erlaubt wird oder die Anforderung von zusätzlichen Information.

Richtlinien und Beglaubigungen sind als Filter realisiert. Die Richtlinien und Beglaubigungen werden mit einem öffentlichen Schlüssel verbunden. Der Filter erlaubt Aktionen dann in Abhängigkeit davon, ob der Nutzer, der den Dienst anfordert, den entsprechenden privaten Schlüssel besitzt. Zu Verzögerungen kann es kommen, wenn die benötigten Informationen lokal nicht vorhanden sind. Die Informationen müssen dann von einem vertrauenswürdigen Knoten geholt werden. Erst dann kann die Entscheidung gefällt werden.

PolicyMaker kann mit den meisten anderen Authentifizierungs und Sicherheitsprotokollen kollaborieren, da es nicht versteht, was in dem String, der die Aktion bezeichnet, steht. Mit der Entscheidung, ob die Aktion genehmigt wird hat die Aktion selbst nichts zu tun. Die Entscheidung fällt nur durch die Filter, wie oben beschrieben [16].

2.5 Ökonomische Aspekte und Zusammenfassung

Die Anzahl der kommerziellen und monetär-relevanten Dienste im und um das Internet ist stetig steigend. Online-Banking, Online-Shops, Video-on-demand, der Internetzugang per se, Websites, die moralisch fragwürdige Mediendaten anbieten, Singlebörsen, Online-Drucken, Spieleserver um einige Beispiele zu nennen.

Dabei ist es wichtig, daß die Relation Dienstnutzung-Nutzer eindeutig ist. Die Wahrscheinlichkeiten einer Identitätsfälschung, Zurückweisung, Indiskretion oder Fehlberechnung der Kosten muß äußerst gering sein, da das direkt oder über Umwege geschäftsschädigende Folgen haben kann. Dies wird durch die Anonymität des Internets und die unter Umständen wechselnden Geräte, mit denen der Nutzer die Dienste in Anspruch nimmt, erschwert.

Um trotzdem die Sicherheit in den oben genannten Bereichen zu erzielen sind spezielle Konzepte erforderlich. Idealerweise erledigt ein Konzept allein die drei Hauptanforderung Authentication, Authorising und Accounting. Doch obwohl die Anforderungen erkannt wurden sind in der Vergangenheit nur sehr weniges AAA-Protokolle entwickelt worden. Dies ist zum einen darauf zurückzuführen, daß der Internetboom erst vor wenigen Jahren eingesetzt hat, und zum anderen, daß die Entwicklung eines solchen kombinierten Protokolls und seine Realisierung mit der entsprechenden Zuverlässigkeit und Sicherheit eine nicht triviale, sondern schwierige Aufgabe ist. Dazu kommt noch die Tatsache, daß man mit der Entwicklung eines solchen Protokolls nicht unbedingt soviel wirtschaftlichen Gewinn machen kann, wie mit anderen IT-Entwicklungen und die Entwicklung somit auf meist weniger leistungsfähige Institutionen wie Universitäten zurückfällt. Auch braucht ein Protokoll eine gewisse Erfahrungsphase, wie alle neuen Produkte, in der nicht bedachte Schwächen und Fehler in der Praxis aufgedeckt und behoben werden.

Zuletzt wurde die Notwendigkeit aber erkannt und es wurden diverse Arbeitsgruppen eingerichtet, die sich mit dem Ausarbeiten von Konzepten beschäftigen. Aus diesen werden dann hoffentlich in nächster Zukunft brauchbare und vielseitige AAA-Protokolle und Programme entstehen. Bis dahin werden die Interessengruppen, die ein AAA-System benötigen, aber auf RADIUS zurückgreifen müssen oder die Verantwortlichkeit für Authentication, Authorising und Accounting auf mehrere Protokolle aufteilen müssen.

Literaturverzeichnis

[1] Glass, S. & Hiller, T. & Jacobs, S. & Perkins, C. , Mobile IP Authentication, Authorization, and Accounting Requirements, RFC 2977

[2] AAA Working Group des IETF (http://www.ietf.org/html.charters/aaa-charter.html)

[3] AAA Architecture Research Group des IETF (http://www.phys.uu.nl/~wwwfi/aaaarch/charter.html)

[4] Rigney, C., Willats, W., Calhoun, P., RADIUS Extensions, RFC 2869

[5] Calhoun, P. R. & Rubens, A. C. & Akhtar, A. & Guttman, E., DIAMETER Base Protocol, RFC 3588

[6] de Laat, C. & Gross, G. & Gommans, L. & Vollbrecht, J. & Spence, C., Generic AAA architecture, RFC 2903

[7] Vollbrecht, J. & Calhoun, P. & al., AAA Authorization Framework, RFC 2904

[8] Vollbrecht, J. & Calhoun, P. & al., AAA Authorization Application Examples, RFC 2905

[9] Farrell, S. & Vollbrecht, J. & Gommans, L. & Gross, G., AAA Authorization Requirements, RFC 2906

[10] Aboda, B. & Arkko, J. & Harrington, D., Introduction to Accounting Management, RFC 2975

[11] Maughan, D. & Schertler, M. & Schneider,M. & Turner, J., Internet Security Association and Key Management Protocol (ISAKMP), RFC 2048

[12] Harkins, D. & Carrel, D., The Internet Key Exchange(IKE), RFC 2409

[13] Myers, J., Simple Authentication and Security Layer (SASL), RFC 2222

[14] LtzS Möller, D., Kerberos

[15] Blaze, M. & Feigenbaum, J. & Ioannidis, J. & Keromytis, A. ,The KeyNote Trust-Management System Version 2, RFC 2704

[16] Blaze, M. & Feigenbaum, J. & Lacy J. , Decentralized trust management

[17] Andrea Schalk, Der Internethandel boomt, e-business.de 25.05.04

[18] Joshua Hill, An Analysis of the RADIUS Authentication Protocol

Kapitel 3

Definition and Use of Service Level Agreements (SLA)

Witold Jaworski

Bedingt durch die technologische Weiterentwicklung des Internets in den letzten Jahren ergeben sich immer mehr Möglichkeiten der Service Provider ihren Kunden bedarfsgerechte Dienste anzubieten. Wie es in der Geschäftswelt allgemein üblich ist, die benötigten Dienstleistungen von einem Unternehmens vertraglich zu regeln, so hat dieses Konzept nun auch im IT-Dienstleistungsbereich Einzug gehalten. Sogenannte Service Level Agreements (SLA) spezifizieren den Bedarf der Unternehmen und privaten Kunden an IT-Diensten. Der Service Provider ist verpflichtet die Dienste gemäß den verhandelten SLAs auszuliefern. Das setzt voraus, daß die Qualität der gelieferten Dienste in irgendeiner Weise meßbar sind.

In diesem Seminarvortrag wird untersucht wo SLAs eingesetzt werden und welche Arten existieren. Der Inhalt eines SLA wird näher beleuchtet, insbesondere die SLA-Parameter. Sie stellen die einzelnen verhandelten Verbindlichkeiten dar. Weiter wird darauf eingegangen, wie mit Verletzungen der Vereinbarungen bei Nichterfüllung der Dienstleistungen umgegangen wird und wie Service Provider sanktioniert, aber auch motiviert werden können, um den Kunden zufriedenzustellen. Es wird gezeigt, wie ein SLA formuliert wird und wie man ihn mit Hilfe formaler Sprachen ausdrücken kann, um sich der Doppeldeutigkeiten menschlicher Sprache zu entledigen und um SLAs technisch einfacher umsetzen zu können. Zum Abschluß wird erörtert, wie man SLAs monitoren und durchsetzen kann. Durchweg wird an geeigneten Stellen angesprochen, welche Schwierigkeiten es bei der Umsetzung von SLAs geben kann.

Inhaltsverzeichnis

3.1 Einführung

Der Verwalter eines Gebäudekomplexes, z.B. einer Wohnanlage, hat umfangreiche Aufgaben wahrzunehmen. Da er diesen Aufgaben aber nicht alle alleine nachkommen kann, beauftragt er verschiedene Personen und Firmen, die diese wahrnehmen. So stellt er Hausmeister ein, die die Gebäude in Stand halten, setzt Sicherheitspersonal ein, um den Zugang zu regeln und die Sicherheit der Anlage zu gewährleisten. Es werden Firmen zu Reinigung verpflichtet, genauso wie ein Unternehmen, welches die Müllabfuhr erledigt. Betrachtet man so eine Firma genauer, z.B. die Firma, die sich um die Aufzüge in den Gebäuden kümmert, so kann diese eine Vielzahl von Aufgaben haben. Die Aufzugsfirma muß regelmäßig alle Aufzüge überprüfen, instandhalten und nach gesetzlichen Vorschriften kontrollieren lassen. Bei Störungen und Notfällen muß sie jederzeit erreichbar sein, um schnell Abhilfe zu schaffen.
Der Gebäudeverwalter wird, damit er sich nun selber nicht mehr um die Aufzüge zu sorgen braucht, diese besagte Firma verpflichten. Man erkennt leicht, daß die Aufzugsfirma eine Art Dienstleistung erbringt, nämlich das reibungslose Funktionieren der Aufzüge und im Störfall eine schnelle Behebung desselben. Die Firma wird wohl nicht nur diese eine Wohnanlage betreuen, sondern derer vieler, da sie sich ja direkt darauf spezialisiert hat. Natürlich kommt sie auch mit mehr als einer Art Aufzügen zurecht. Der Verwalter wiederum hat sich, bevor er diesen Auftrag an die Firma vergeben hat, am Markt erkundigt, welche Unternehmen diese Aufgabe wahrnehmen können, deren Angebot studiert und die Preise verglichen. Er hat einen Vertrag mit der Aufzugsfirma geschlossen, in dem die Pflichten niedergeschrieben sind, z.B. wie oft die Aufzüge kontrolliert werden sollen, oder wie lange es dauern darf, bis ein Techniker erscheint, wenn eine Störung gemeldet wurde. Der gerade beschriebene Vorgang, daß eine Unternehmen einen Dienstleistungsauftrag erhält, weil der Auftraggeber diese Aufgabe selber nicht wahrnehmen kann oder will, ist in der Geschäftswelt eine reine Selbstverständlichkeit und somit ist es auch nicht verwunderlich, daß dieses in der IT-Welt Einzug gehalten hat. Unternehmen oder Personen, die irgendeine Art von IT-Dienstleistung benötigen, die sie selber nicht in der Lage zu leisten sind, wenden sich an derart spezialisierte IT-Dienstleister, wie sie ihren Ansprüchen genügen und die sie am Markt vergleichen können.

3.1.1 Definition SLA

Zuerst erfolgt eine Defintion des Begriffes SLA, wie sie häufig in abgewandelter Form zu lesen ist:

> „Ein Service Level Agreement ist eine schriftliche Vereinbarung zwischen dem Kunden (Servicenehmer) und dem IT-Dienstleister (Servicegeber) oder zwischen zwei IT-Dienstleister über Qualität und Quantität von IT-Dienstleistungen."

In der Literatur findet man auch oft den Begriff „Vertrag" statt „schriftliche Vereinbarung", was aber genau genommen nicht ganz richtig wäre, da z.B. auch SLAs innerhalb eines Unternehmens zwischen verschieden Abteilungen, eine wäre dann der Servicegeber, zustande kommen. Da ein Vertrag rechtliche Konsequenzen nach sich zieht, wäre so etwas an dieser Situation in den seltensten Fällen sinnvoll.

3.1.2 Verwendung von SLAs

Die Gründe für die Entstehung von SLAs sind vielfältig. So sind sie z.B. die Grundlage um „Quality of Service“ (QoS) Garantien einzuführen. Wie in Abb. 3.1 zu sehen ist, setzt das ITU-T G.1000[1] in gewisser Weise voraus, daß das QoS- Angebot des Providers und die QoS-Anforderungen des Kunden in Einklang zu bringen sind, um anschließend den geleisteten QoS bewerten zu können. Dieser Einklang und die Maßnahmen, wenn dieser verletzt wird, sind in den SLAs wiederzufinden.

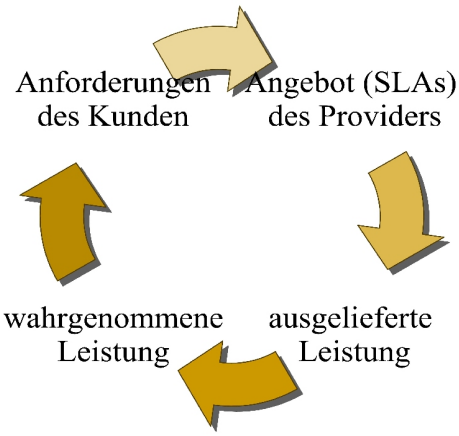

Abbildung 3.1: *ITU-T G.1000 Modell*

Da Unternehmen in immer größerem Maße von IT-Diensten abhängig sind, sind sie immer öfter gezungen, Teile ihrer IT-Infrastruktur an externe Dienstleister auszulagern. Dies erfolgt, weil den Firmen das nötige Know-How fehlt, selbst eine dafür zuständige Abteilung zu unterhalten, externe Dienstleister mehr spezialisiert sind und kostengünstiger arbeiten können. Anderseits, wenn man IT-Dienstleistungen innerhalb eines Unternehmens anbietet, stellt sich die Frage, inwieweit diese Abteilungen mit dem Rest des Unternehmens effizient zusammenarbeiten und sich somit eine interne Leistungsverrechnung erstellen läßt. Durch den Einsatz von SLAs können Unternehmen nun verschiedene IT-Dienstleister miteinander vergleichen, da diese Servicekataloge für Dienstleistungen erstellen. Desweiteren ist der IT-Dienstleister nun zum Nachweis verpflichtet, seine Dienstleistung gemäß den SLAs geliefert zu haben. Somit einsteht ein Wettbewerbs- und Nachweisdruck für die Dienstleister zum Wohle der Kunden. Dieser bekommt seinen auf seinen Bedarf zugeschnittenen Service und weiß, was er von seinem IT-Dienstleister zu erwarten hat, wogegen dieser das Wissen hat, was der Kunde erwartet. Es besteht somit eine eindeutige gegenseitige Erwartungshaltung. Dieses Konzept sollte, wenn es richtig umgesetzt wurde, die Kundenzufriedenheit erhöhen und den Kunden an den IT-Dienstleister binden.

3.2 Arten von SLA

Das ASP Industry Consortium[2] hat vier grobe Einteilungen für verschieden Arten von SLAs vorgenommen, auf die jetzt im einzelnen eingegangen wird.

3.2.1 Network SLA

Ein Network SLA betrifft die Netzwerkverbindung zwischen dem Kunden und seinem Netzwerk Service Provider, das Netzwerk selbst und die Sicherheit des Netzes. In Abb. 3.2 ist zu sehen, daß ein Client über das Netzwerk des Providers eine Applikation auf einem Server in einem LAN in Anspruch nimmt. Diese Inanspruchnahme ist durch ein Application SLA geregelt und wird im Kapitel 1.2.3 näher erläutert. Da der Service Provider des Servers nicht im Zusammenhang mit dem Netzwerk Provider stehen muß, benötigt der Kunde ein Network SLA. Dieser regelt wichtige Fragen der Leistung, Verfügbarkeit, Sicherheit etc. des Netzes. In diesem Beispiel könnte der Rechner des Kunden über ein VPN ins eigene Firmennetzwerk eine Firmenapplikation nutzen. Hier wird auch schon deutlich, daß sich die Möglichkeiten des Network Service Providers z.B. für QoS einschränken, wenn Verbindungen benötigt werden, bei denen sich ein Verbindungspartner nicht mehr im Autonomen System (AS) des Internet Service Provider (ISP) aufhält. Dies liegt am „best effort“ Charakter des Internets.
Im folgenden werden die wichtigsten Aspekte zur Charakterisierung eines Netzwerkes angesprochen.

- **Verfügbarkeit**
  Die Verfügbarkeit ist eine Prozentangabe in Abhänigkeit von der Betriebszeit des Netzwerkes, in dem das Netz dem Kunden funktionierenderweise zu Verfügung steht und der Gesamtzeit, in der der Kunde das Netzwerk laut Vereinbarung nutzen darf. Arbeitete des Netzwerk durchgängig fehlerfrei und gäbe es keine Äusfalle, ergäbe dies eine Uptime von 100%. Heutige Provider geben Betriebszeiten im üblichen von mindestens 99% an. Eine Annäherung an 100% bedarf eines exponentiellen Aufwandes, was die Frage stellt, inwieweit dies sinnvoll wäre. Auch sollte man beachten, daß bei 99% Uptime im Jahr 87 Stunden Ausfallen hinzunehmen sind. Bei einem Unternehmen kann ein Ausfall während der Geschäftszeit viel gravierender nachteilige Folgen haben, als nachts um 2 Uhr. Hier bietet es sich durchaus an im Interesse des Kundens verschieden Level zu definieren.

- **Leistung**
  Die Leistung des Netzwerkes wird vordergründig mit dem Durchsatz (Bit / Sekunde) beschrieben. Verbindungen zu Backboneknoten des Providers und darüber hinaus zu bedeutenden Peeringpunkten im Backbone sind eine Möglichkeit, wie auch Punkt zu Punkt Verbindungen für ein privates Netzwerk im AS des Providers. Umlaufzeiten (Round Trip Time, RTT) findet man mitunter als Perfomancemetrik, die auch Indikatoren für Verfügbarkeit sein können.

- **QoS**
  IP basierte Netzwerke bieten die Möglichkeit, Kunden QoS anzubieten. Da QoS Level durch QoS Parameter beschrieben werden, sind diese hier von Interesse. Neben dem schon erwähnten Durchsatz, gibt es Datenverlust (Data Loss), Latenz, Verzögerung (Delay), Jitter, Fehlerkontrolle (Error Control), Übertragungszuverlässigkeit (Transmission relaibility), Priorität, etc. So hat ein Netzwerk um Video- oder Audiostreams in Echtzeit darstellen zu können z.B. bedeutend höhere Anforderung an Verzögerung und Datenverlust als herkömmliche Webanwendungen. Im SLA werden

die QoS Level mit ihren entsprechenden Parametern in Form von SLA Parametern repräsentiert.

- **Sicherheit**
  Sicherheit betrifft den Grad der Verschlüsselung im Netzwerk für die Daten, definiert die Punkte, wo die Ver- und Entschlüsselung erfolgt und identifiziert Dienste die Verschlüsselung benötigen. Auch die Benutzung von öffentlichen oder privaten Verschlüsselungsverfahren gehört hier dazu. In Abhänigkeit vom Umfang der Sicherheit beeinflußt diese die Performance des Netzwerk, z.B. den Durchsatz oder QoS Level, was dann zu berücksichtigen ist.

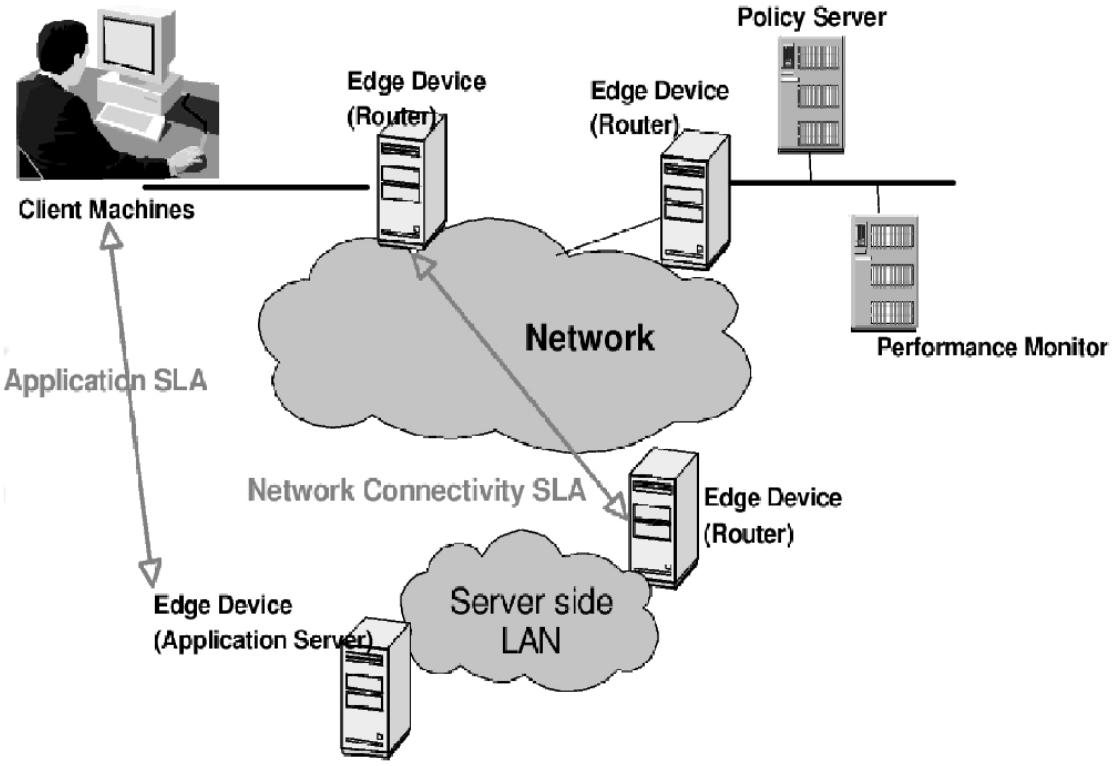

Abbildung 3.2: *Network- und Application SLA im Netzwerk*

3.2.2 Hosting SLA

Ein Hosting SLA behandelt die Verfügbarkeit von serverbasierten Betriebsmitteln. Man versteht gewöhlich darunter die Verfügbarkeit, Administrierung und Backup von Servern. Die Leistung steht eher im Hintergrund. Betrachtet man Abb. 3.2 nochmals, so könnte man sich vorstellen, daß die Applikation auf einem Server läuft, der wiederum von einem dritten Provider verwaltet wieder. Der Server, also die Hardware, beherbergt nun diese Applikation, die Software, und beide Provider schließen ein Hosting SLA miteinander ab.

- **Verfügbarkeit**
  Der Parameter ist analog zu verstehen wie beim Network SLA, nur hier auf einen Server bezogen, anstatt auf das Netzwerk. Daneben gibt es auch „mean-time-to-restore“ (MTTR). Die MTTR gibt die Zeit an, wie lange es zwischen Serverausfall und Wiederherstellung des Service dauert. Zu beachten ist, daß die MTTR nur eine Durchschnittszeit ist. Wenn ein Service Provider 60 Sekunden MTTR innerhalb von 24 Stunden für einen Monat verspricht, so erfüllt er dies auch, wenn der Server an drei Tagen jeweils 9 Minuten und an den restlichen Tagen je nur 2 Sekunden nicht erreichbar ist. Der Kunde heißt so eine große Zeitspanne von 9 Minuten mitunter nicht gut und verlangt Obergrenzen in der Zeitdauer für einzelne Ausfälle.

- **Administrierung**
  Dies beinhaltet Vereinbarungen zur Administrierung des Servers. Dazu gehören z.B. Wartung / Aktualisierung der (Betriebssystem-)Software, Netzwerkkonfiguration, Verwalten von Benutzern, Alarmierung bei Störungen, Auswerten und Präsentieren von Logfiles.

- **Backup**
  Das Anlegen von Backups und deren Wiederherstellung kann man auch als Administrierungsaufgabe ansehen. Es beinhaltet die Anzahl und Frequenz von Backups und die Zeitdauer zur Aufbewahrung der Backupdaten. Auch deren Aufbewahrungsort kann verhandelt werden, z.B. ein besonders abgesicherter Ort gegen physische Einflüße von außerhalb, was ein feuerfester Safe sein kann. Das Entwickeln und Testen von Wiederherstellungsmaßnahmen für besonders wichtige Server im Falle von Totalausfällen infolge von Katastrophen (Erdbeben, Hurrikans) um zumindest Grundfunktionalitäten für den Kunden wieder herzustellen, kann ebensogut im SLA verhandelt werden.

- **Physikalische Serversicherheit**
  Die physikalische Sicherheit von Servern, insbesondere Datenserver, kann mit restriktiven Zugang zu Räumlichkeiten für ausgewiesenes Personal, Überwachung mit Videokameras, abschließbaren Anlagen, etc. gewährleistet werden. Spezieller Schutz vor Umweltbedingungen außerhalb der Betriebsparameter der rundet die physikalische Sicherheit ab. Kunden sollten die Möglichkeit haben, diese Maßnahmen vor Ort zu überprüfen.

3.2.3 Application SLA

Application Service Provider (ASPs) bieten ihren Kunden Applikationen zur Nutzung an. Dafür gibt es verschiedene Möglichkeiten, was das Eigentum an der Applikation betrifft. So kann der Kunde die Software lizenzieren und der ASP betreut diese, oder die Software wird nur gemietet und das Eigentum bleibt beim Provider. Beispiele für solche Applikationen sind z.B. Internetshops, Finanz- oder Voice over IP (VoIP)-Anwendungen.
Es ist nun nicht mehr so einfach den Service und die dazu passenden Garantien zu bestimmen, die in einem SLA hineingehören. Wie auch bei den vorherigen Arten steht die Verfügbarkeit ganz oben auf der Liste. Da die Applikationen zumeist über das Netzwerk

genutzt werden, ist die Performance der Applikation nicht nur von ihrer eigenen Ausführungsgeschwindigkeit abhänig, sondern auch von Intranet des Kunden und eventuell auch dem Internet, über das die Daten übertragen werden. Anderseits sollte der Begriff Leistung im Zusammenhang mit der Applikation stehen. So könnte die Zeit zwischen einer Kundeneingabe und einer Antwort der Applikation genau festgelegt werden, da im Falle des Überschreitens dieser Zeitdauer der Kunde nicht mehr sinnvoll arbeiten kann. Mögen 3 Sekunden für den normalen Anwender von Webapplikationen genügen, so gibt es genauso Anforderungen für Echtzeit, z.B. im Falle eines Finanzhändlers. Prozentangaben von erfolgreichen Benutzerinteraktionen, Downloads und Anfragen in einer bestimmten Zeitspanne sind weitere Beispiele für verhandelbare SLA-Parameter.
Im Vordergrund steht die Zufriedenheit des Kunden, was voraussetzt, daß man all seine Bedürfnisse auch erfaßt und im SLA verhandelt hat. Um beim Beispiel mit den erfolgreichen Downloads zu bleiben, es ist sicher nicht befriedigend, wenn dieser mehrere Stunden dauert, auch wenn er am Ende erfolgreich war.
Die Sicherheit der Applikation, insbesondere der zu übertragenden Daten spielt genauso eine Rolle, wie Authentifikationsmechanismen. Auch die Speicherung und Sicherung von Nutzerkonfigurationen, sowie dessen Herstellung gehören betrachtet. Versionskontrolle und Update-Regelungen sind weitere Punkte.

3.2.4 Costumer Care/Help Desk SLA

	Critical	**Normal**	**Minor**
Initial Investigation	Starts within half an SLA hour of call acceptance.	Starts within one SLA hour of call acceptance.	Starts within eight SLA hours of call acceptance.
Initial Feedback	One SLA hour from Call Acceptance.	Four SLA hours from Call Acceptance.	One business day from Call Acceptance.
Secondary Feedback	Two SLA hours from Call Acceptance. Client's Account Manager notified	One business day from Call Acceptance. Client's Account Manager notified	Two business days from Call Acceptance.

Abbildung 3.3: *Help Desk einer Webhosting Firma (www.onesquared.net)*

Service Provider bieten für ihre Kunden Unterstützung/Betreuung bei auftretenden Problemen z.B. in Form eines Kunden Support Call Centers an. Wenn diese Betreuung in technischer Form erfolgt, so spricht man von „Help Desk“. Nichttechnische Unterstützung wird als „Costumer Care“ bezeichnet. Diese Art von SLA beinhalten Vereinbarungen, wie z.B. die Dauer von entsprechenden Antwortzeiten bei der Meldung des Kunden von Problemen an den Provider oder die Benachrichtigungen des Providers an den Kunden

bei Identifizierung eines Problems seinerseits. Probleme werden oft in Abhänigkeit ihrer Bedeutung priorisiert und mit unterschiedlichen Zeiten behandelt. Dazu gehören weiter auch die Erreichbarkeiten von Ansprechpartnern für die Kunden (per EMail, Telefon, Web-Formular, Fax) und die Zeiträume, in denen diese Ansprechpartner erreichbar sind. Für spezielle Anforderungen werden mitunter Eskalationsprozeduren vereinbart.
In Abhänigkeit von der Benachrichtigungsart und der Priorität des Problems werden zwar oft kurze Zeitangaben für die Registrierung des Problems angegeben, aber wichtiger wäre Zeitangaben, bis wann es gelöst ist. Die Abb. 3.3 zeigt so ein typisches Beispiel mit verschieden Prioritäten. Die Kundenzufriedenheit richtet sich im allgemeinem nach der Richtigkeit der Information zur Problemlösung, Schnelligkeit von Antwort- und Reaktionszeiten und einem angemessenem Umgangston.

3.3 SLA Parameter

Der schon öfter genannte Begriff „SLA Parametert“ soll in diesem Abschnitt näher behandelt werden. Zur Veranschaulichung des Ganzen dient die Abb. 3.4. Die Kunden haben ganz bestimmte Anforderungen an benötigte Dienste. Anderseits können Provider ihr Angebot gut durch die Resourcemetriken, basierend auf Meßgrößen der Layer 2-4 im ISO/OSI Modell, beschreiben, was dem Kunden nicht unbedingt nutzt, da er mit diesen Angaben nichts anfangen kann, da sie zu technisch sind und er keinen unmittelbaren Zusammenhang zu seinen Dienstanforderungen erkennt. Lassen sich Network- und Hosting SLAs noch einigermaßen ausschließlich mit Resourcemetriken als Anforderungen handeln, ist spätestens bei den Application SLAs Schluß. Diese benötigen eigene Applikationsmetriken, die wiederum von Resoucemetriken abhänig sein können. Aus diesem Grunde erstellen Provider Dienstleistungskataloge mit SLA Parametern, die jeweils auf die Bedürfnisse der Kunden abgestimmt sind. Sie bilden sozusagen ihre Metriken auf die Parameter ab, wobei ein Parameter durchaus mit mehreren Metriken beschrieben wird, was für den Kunden aber von weniger Interesse ist. Ihn interessiert dieser Parameter, ob er den eigenen Bedürfnissen, seinen „Service Level Requirements“, entspricht. Ist dies nicht der Fall kann er ausgehandelt werden.
SLA Parameter werden von den Providern jeweils mit verschiedenen Level gewichtet, so z.B. Serveruptime Level 1 99%, Level 2 99,5%, Level 3 99,99%. Desweiteren können die Parameter mit high/low Schwellwerten versehen werden, z.B. Minimum und Maximum der Verzögerung von IP Paketen im Netzwerk hinsichtlich für Multimediakommunikation. Die Priorisierung von SLA Parametern dient dazu die Wichtigkeit derer festzulegen. So kann ein Überschreiten von Schwellwerten bei den einem Parameter für den Kunden wesentlich nachteiligere Auswirkungen haben, als bei einem anderem Parameter. Dies hat unmittelbare Folgen für die Art der Strafen gegen den Provider und dieser kennt die Schwerpunkte seitens der Kundenanforderungen.
Ein weiterer Aspekt ist die Art der Überwachung und Protokollierung der SLA Parameter. Ein Kunde könnte darauf bestehen, daß dieses nach seinen eigenen Wünschen geschieht, inklusive der Berechnungsalgorithmen, was aber in den seltensten Fällen so sein wird. Meistens wird das Angebot des Providers ohne Verhandlung akzeptiert, oder der Kunde wünscht nur ausgewählte Daten zu wissen.

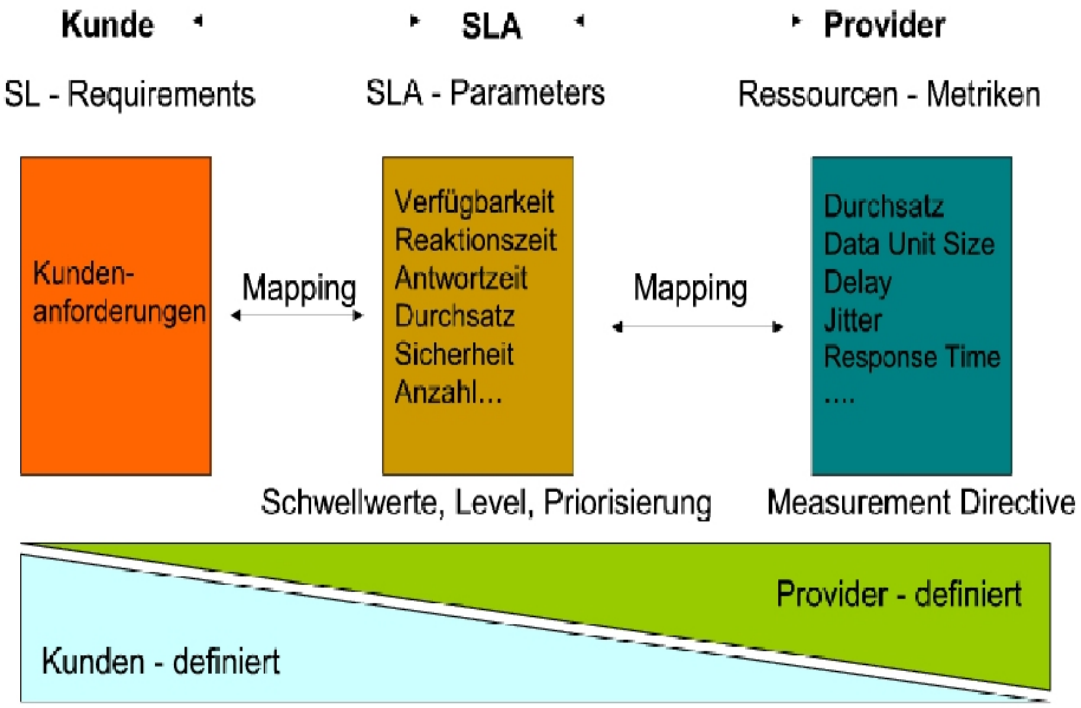

Abbildung 3.4: *SL-Requirements, SLA Parameter, Metriken*

3.4 SLA Gestaltung

Abb. 3.5 stellt die Rahmenbedingungen und den Ablauf beim Zustandekommen eines SLA dar. Bis jetzt wurde die Implementierung behandelt, wobei noch der Aspekt für die Eskalations- und Belohnungsmechanismen fehlt, der natürlich auch vorher verhandelt wird, damit diese Mechanismen während eines bestehenden SLA greifen können. Um das jetzige Wissen kurz zusammenzufassen: Provider stellen Dienstkataloge für ihre potentiellen Kunden auf, die mit den schon vorgearbeiteten SLA Parametern, die Bedürfnisse und Anforderungen der Kunden treffen sollen. Die einzelnen SLA Parameter werden verhandelt und sind im SLA schlußendlich vereinbart. Wie jetzt der SLA formuliert wird und welchen genauen Inhalt er hat, folgt noch. Der Kunde nutzt nun den Service des Providers und durch Monitoring und Reporting können Verstöße gegen den SLA geahndet werden. Stellen Kunden nach gewisser Zeit Verbesserungsmöglichkeiten oder -notwendigkeiten fest, so kann durch Nachverhandlung des SLA das Angebot des Providers für den Kunden verbessert werden. Noch viel besser wäre, wenn der Provider solche Zusammenhänge erkennt und selbst handelt. So kann er z.B. einen günstigeren Service Level anbieten, weil er erkannt hat, daß die jetzigen Vereinbarungen überdimensioniert sind.
Da viele Kunden ihre eigenen Anforderungen haben, gibt es mitunter viele spezielle SLAs, die ein Provider zu berücksichtigen hat. Das kann im Extremfall zu Skalierungsproblemen führen. Genauso besteht dieses Skalierungsproblemen eher noch bei den Kunden, wenn einmal verhandelte SLAs aufgrund von Wachstum nicht mehr die Anforderungen erfüllen können. Dies sind alles gute Gründe die Zeitdauer für SLAs auf eher kürzere und überschaubare Zeiten zu begrenzen.

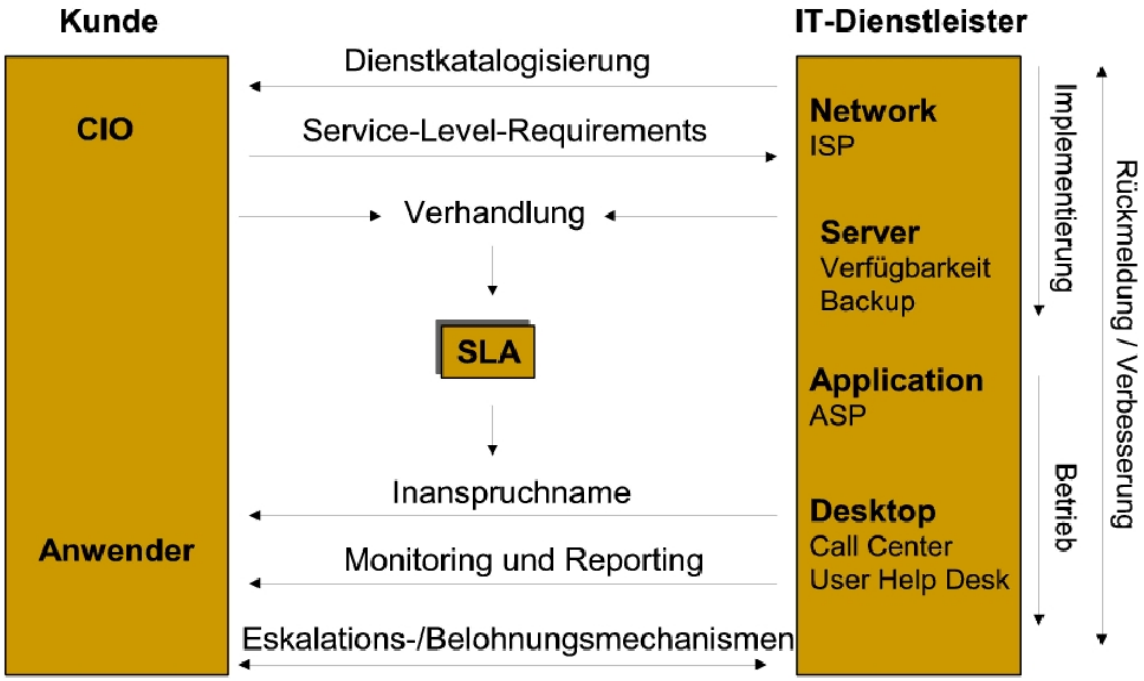

Abbildung 3.5: *SLA-Überblick (CIO = Chief Information Officer)*

3.4.1 Strafen und Motivation

Die Gefahr von Strafen bei Verletzung des SLA sollen den IT-Dienstleister erfolgreich motivieren, den SLA einzuhalten. Diese Strafe steht aber in Konkurrenz zum finanziellen Verlust des Kunden durch die Vertragsverletzung. Soll und kann ein Provider z.B. bei einem größerem Unternehmen für dessen finanziellen Verlusten bei Verletzung gerade stehen? Wie würde dieser Verlust gemessen werden? Inwieweit wären solche Szenarien kalkulierbar? Muß der Provider sich dann für solche Fälle extra absichern? Soll er dann noch zusätzlich bestraft werden, weil er Vereinbarungen im SLA nicht eingehalten hat? Service Provider werden sich deshalb aus gutem Grund wehren bei einem Schaden mit „legal tender“ zu begleichen. Unternehmen haben aber gute Gründe extra darauf zu bestehen, kann doch der Ausfall der IT-Dienstleistung im schlimmsten Fall heute schon in den Ruin führen. Eine Rückerstattung für die Zeitdauer der Verletzung des SLA hilft da wenig.
Anderseits sind finanzielle Strafen aus Sicht des Kunden nicht immer unbedingt vorteilhaft. Bei firmeninternen Dienstleistern z.B. besteht für so etwas wenig Bedarf, außer man richtet seinen Blick auf das Budget der Abteilungen, Prämien der Mitarbeiter, oder deren Gehalt. Eine Budgetkürzung wird aber wohl eher ein Schnitt ins eigene Fleisch sein. IT-Dienstleister könnten wiederum auf den Gedanken kommen, die Strafen zu bezahlen, weil die Kalkulation ergaben hat, daß es billiger ist, zu zahlen, als den SLA zu garantieren.
Somit vergeben Service Provider gerne Credits (Gutschriften) für zukünftige Inanspruchnahme von Diensten an ihre Kunden. Es besteht auch die Möglichkeit den Provider zu belohnen, wenn bestimmte Vorgaben übererfüllt wurden. So etwas ist z.B. in einem Call Center vernünftig, wenn jetzt innerhalb von 30 Sekunden 95% aller Anrufe entgegenge-

nommen werden, statt 90%. Steigert sich aber die Verfügbarkeit der Netzwerks von 99,5% auf 99,99% so sollte sich der Kunde fragen, inwieweit dieser Effekt überhaupt wahrgenommen wird.
Die Abb. 3.6 beschäftigt sich mit der Frage, wie eine geeignete Auswahl von Schwellwerten bei den SLA Parametern den Provider motivieren können. Im linken Diagramm könnte man davon ausgehen, daß der Provider kein Interesse hat, seinen Dienst zu verbessern, solange der Schwellwert nicht erreicht wird. Erst nach Überschreiten dieses Wertes würde er aktiv werden. Das rechte Diagramm dagegen besteht nicht mehr aus einem Schwellwert, sondern derer vieler. Dies könnte den Provider veranlassen seinen Dienst in Richtung der niedrigeren Schwellwerte zu verbessern, um mehr Mehrwert zu erzielen.

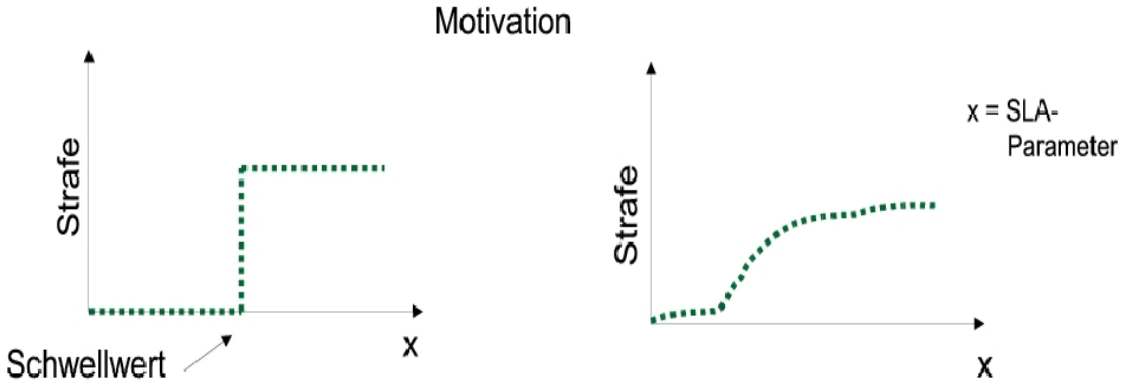

Abbildung 3.6: *Motivation durch Schwellwerte*

Eine ganz andere Möglichkeit der Bestrafung des Providers von psychologischer Natur wäre, wenn Eskalationsmechanismen die Hierarchie des Providers hochklettern und somit untere Verantwortlichkeiten in Verlegenheit bringen.

3.4.2 Inhalt und Formulierung eines SLA

Folgende Punkte sollte jeder SLA mindestens beinhalten:

- Beschreibungen von Dienstleistung, Servicenehmer, Servicegeber
- SLA Parameter
- Meßmethoden (zu messende Metriken, Algorithmen zu Berechnung...)
- Verantwortlichkeiten
- Eskalationsprozeduren
- Sanktionen bei Verstößen gegen die Vereinbarungen im SLA
- (Belohnungen bei Ertragserfüllung)

SLA sind in nichttechnischer Sprache formuliert, da dies den Kunden in den wenigsten Fällen zuzumuten wäre. Der Kunde wäre durch ein Übermaß an technischen Fachbegriffen schnell überfordert. Wenn SLAs die Grundlage einer vertraglichen Vereinbarung sind, so werden Verträge in der Regel von Geschäftsleuten und Anwälten gezeichnet, bzw. betreut. Dies führt zwangsläufig zu Problemen, wenn Begriffe von verschieden Personen unterschiedlich aufgefaßt werden. So kann z.B. die Verfügbarkeit einer Applikation auf einem Server mit dem bestehen einer TCP/IP Verbindung zum Server, dem Zugriff auf die Applikation selber oder der Serverantwort auf einen HTTP-Request einer bestimmten Monitoring Software beschrieben werden.
Deshalb gibt es Ansätze SLAs durch flexible, formale Sprachen zu beschreiben, oder auch nur Teile davon (z.B. SLA-Parameter). Die Vorteile, die man sich davon verspricht, sind zu einem die erwähnten Zweideutigkeiten bei Begriffen ausschließen zu können. Eine einheitliche Sprache könnte weiter für viele verschiedene SLAs angewendet werden. Es gäbe auch die Möglichkeit der Interaktion mit anderen kompatiblen Systemen im eCommerce oder B2B Bereich. Durch genaue Spezifikationen könnten z.B. Monitoringaufgaben leichter an darauf spezialisierte Provider übertragen werden.
Beispiele für solche Sprachen sind das WSLA Framework (IBM)[3] basierend auf XML und MobyDick[4]. Auf das WSLA Framework wird im letzten Kapitel nochmal eingegangen.

3.4.3 Probleme

Hier sollen nochmal bedeutende Probleme, die bei der SLA Gestaltung auftreten, zusammengefaßt werden. Die Ermittelung des bedarfsgerechten Service für den Kunden bei unterschiedlicher Kundenstruktur stellt an den Service Provider größere Herausforderungen. Nicht jede Kundenforderung muß in einem SLA behandelt werden, genauso wie es Basisdienstleistungen für jedermann gibt. Wenn Kunden den bestmöglichen Service haben wollen, stellen sie oft zusätzliche Anforderungen, die der Provider dann leisten muß.
Die Frage der Quantität neben der Qualität ist wichtig. Vereinbarungen bei Überschreitung von Quantitätsgrenzen hinsichtlich der Qualität müssen getroffen werden. Skalierungsprobleme wurde angesprochen.
Das Verhandeln von geeigneten Sanktions- und Belohnungsmaßnahmen kann sich als schwierig erweisen, genauso wie die Definition nachvollziehbarer Messkriterien und Messintervalle von Metriken für die SLA Parameter. Messungen des Providers stehen nicht immer direkt im Bezug zum Kunden. Wenn ihm Monitoringtools fehlen, um selber Meßungen vom Provider verifizieren zu können, oder der Provider Meßdaten nicht, oder in unbrauchbarer Form abliefert, so kann er berechtigt den Sinn eines SLA in diesem Fall in Frage stellen. Ungenügendes QoS-Management seitens des Providers, ist ein Grund für ungenügende Reports an den Kunden.

3.5 SLA Monitoring und Enforcement

In diesem Kapitel wird eine Möglichkeit vorgestellt, wie Monitoring und Enforcement in Bezug auf SLA realisiert werden kann. Dies geschieht in Anlehnung an das WSLA Framework, wie es dort vorgeschlagen wurde. Abb. 3.7 stellt das Konzept grafisch dar. Man

hat einen Kunden, der einen Dienst des Providers in Anspruch nimmt. Beim Provider gibt es wiederum Monitoring und Management Schnittstellen, auf die weitere Dienste aufsetzen können. Diese speziellen Dienste sind für Monitoring und Enforcement zuständig.

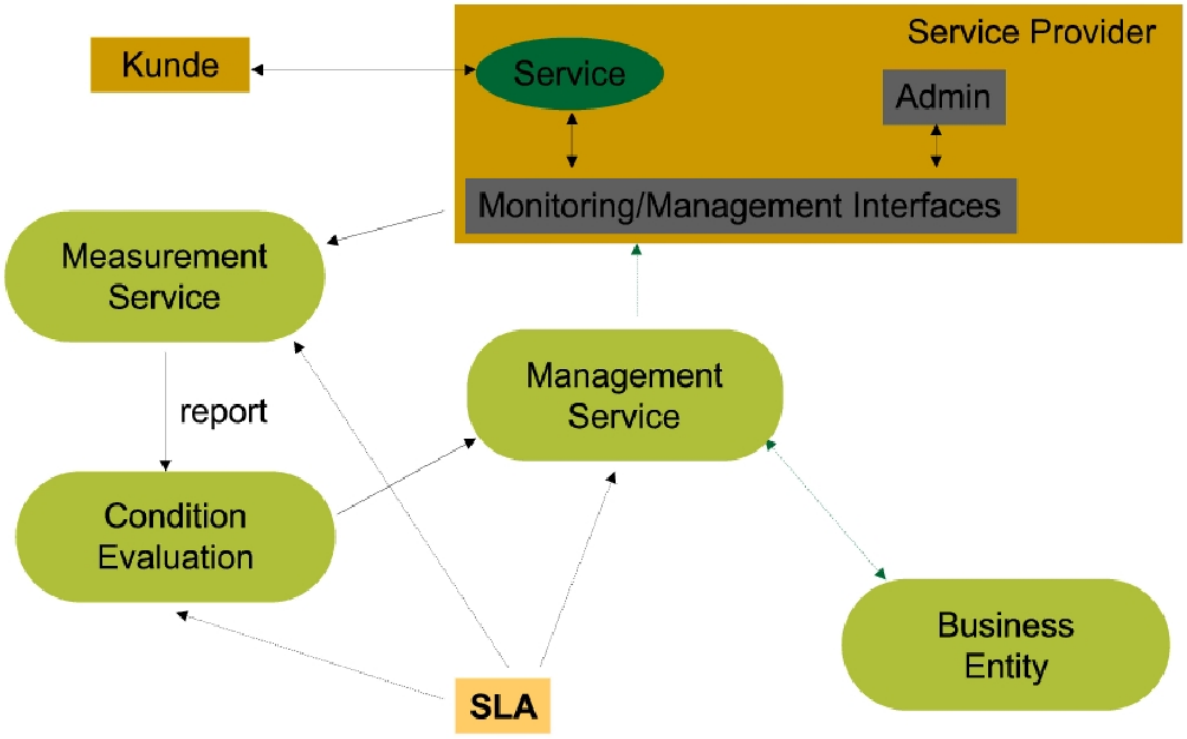

Abbildung 3.7: *Monitoring und Enforcement*

Man erkennt schon, daß diese Aufgaben somit von Dritten vorgenommen werden können, da sie nicht mehr zwangsläufig beim Provider stattfinden müssen. Dies kann durch den Einsatz der angesprochenen formalen Sprachen realisiert werden. Die folgende Bedeutung der einzelnen Dienste sollte das Konzept erklären:

- **Measurement Service**
  Der Measurement Service hält die Laufzeitinformationen der Metriken vor, die für die SLA Parameter relevant sind und muß den Meßort berücksichtigen. Meßmethoden dürfen die Leistungsfähigkeit des Systems nicht beeinflussen und er gibt Messwerte an den Condition Evaluation Service weiter.

- **Condition Evaluation Service**
  Der Condition Evaluation Service ist verantwortlich für den Vergleich der definierten Schwellwerte im SLA mit Meßwerten der Resourcemetriken und benachrichtigt das Management System. Der Vergleich kann periodisch oder ereignisgesteuert, wenn z.B. neue Meßwerte vorliegen, vorgenommen werden

- **Management Service**
  Der Management Service setzt bei Benachrichtigung entsprechende Maßnahmen in Gang, um Probleme zu lösen, speziell bei Verstößen gegen SLA. Er fragt aber vor Aktionen um Erlaubnis bei der Business Entity nach und er arbeitet mit dem Management System des Providers zusammen.

- **Business Entity**
  Die Business Entity repräsentiert die Ziele und Absichten des Providers, die normalerweise den Kunden verborgen sind. Anhand der Vorgaben des Providers werden dem Management Service Anweisungen geben, z.B. die Verweigerung weiterer Beantwortung von Anfragen eines Kunden wegen Kreditüberschreitung bei Inanspruchnahme von pay-per-use Diensten oder die Bevorzugung eines bestimmten Kunden während der Vergabe von Bandbreiten im Falle von Engpässen.

Zum Abschluß noch ein kleine Beispiel, wie man beim WSLA Famework in XML eine Measurement Directive für eine Metrik definiert.

```
<Metric name="ProbedUtilization" type="float" unit="">
    <Source>ACMEProvider</Source>
        <MeasurementDirective xsi:type="Gauge" resultType="float">
                <RequestURL>http://acme.com/SystemUtil</RequestURL>
       </MeasurementDirective>
</Metric>
```

Für die Metrik `ProbedUtilization` vom Typ `float` wurde eine Measurement Directive vom Typ `Gauge` definiert. Die URL `http://acme.com/SystemUtil`wird benutzt, um den Wert von `SystemUtil` zu bestimmen.

3.6 Zusammenfassung

Dieses Papier hat den Begriff „Service Level Agrement“ im Zusammenhang mit IT- Dienstleistungen eingeführt. Es wurde Gründe aufgezeigt, warum in Zeiten steigender Inanspruchnahme und damit verbundener Abhänigkeit von IT-Dienstleistungen, es für den Servicenehmer von Interesse ist, einen auf seinen Bedarf zugeschnittenen Service in Anspruch zu nehmen. Service Provider können mit SLAs dem entgegenkommen und sich damit neue Geschäftsfelder erschließen. Durch das Aufstellen von Dienstleistungkatalogen haben Kunden die Möglichkeit diese Dienstleistungsangebote mit ihren Bedürfnissen zu vergleichen oder gar neu zu entdecken und am Markt nach wirtschaftlichen Gesichtspunkten zu vergleichen.
Die unterschiedlichen Arten von SLA mit Beispielen wurden behandelt, sowie die wichtigsten dazugehörigen SLA-Parameter, die Grundlage der Vereinbarungen im SLA sind. Vertragsstrafen bei Nichteinhaltung der Vereinbarungen im SLA, sowie weitere Maßnahmen zur Motivation der Dienstleister, um dem Kunden den „perfekten“ Dienst zu erbringen sind ein sehr wichtiger Bestandteil eines SLA.
SLA werden in nichttechnischer Sprache formuliert und beinhalten trotz unterschiedlichster Dienstleistung gewisse Mindestanforderungen, die in jedem SLA wiederzufinden sind. Darüber hinaus gibt es Bestrebungen SLA durch formale Sprachen zu beschreiben. Mit deren Hilfe können SLA insbesondere überwacht und durchgesetzt werden. Dies ist ein weiterer wichtiger Bestandteil eines SLA, da damit dem Kunden die Möglichkeit gegeben

wird, Verletzungen des SLA zu erkennen und nachvollziehen zu können, sowie der Provider seinen Kunden die Diensterbringung nachweisen kann und Verletzungen mitunter vor dem Kunden erkennt und damit schnell zu beheben in der Lage ist.

Literaturverzeichnis

[1] Communications Quality of Service: A framework and definitions http://www.itu.int/rec/recommendation.asp?type=items&lang=e&parent=T-REC-G.1000-200111-I

[2] Das ASP Industry Consortium ist jetzt als Computing Technology Industry Association bekannt http://www.comptia.org/

[3] A. Keller, H. Ludewig, „The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services“, IBM Research Report RC22456, May 2002

[4] Hasan, Burkhard Stiller „Auditing of QoS-Supporting Mobile Internet Services“, TIK Report 183, Dezember 2003, ftp://www.tik.ee.ethz.ch/pub/publications/TIK-Report183.pdf

Kapitel 4

How the Internet is Run: A Worldwide Perspective

Christoph Pauls

The evolution of the Internet began in the early 60's. It was designed as a means of communication between a few computers, but quickly became a mesh of research networks and later on a collection of several networks making its commercial facet more appealing.
Due to the fast growth of the Internet and to the fact, that it is almost impossible to control the way it develops, it is almost impossible to deploy a generic end-to-end pricing framework.
The aim of this document is two-fold. First, the document provides an overview of the topology of the Internet and second, it expands on the subject of who is to be held responsible for different developments concerning the Internet and what entities make it possible for the Internet to stay accessible for everyone across the world in order to maintain its international character.

Inhaltsverzeichnis

4.1 Who runs the Internet

The Internet is a huge mesh of different networks around the world, which originated in the USA. However, its exponential growth and self-regulatory nature lead to a lack of control. Currently, it is hard to have a clear picture about all of the different networks that compose the Internet. To understand who/whose entities "run" the Internet, it is first necessary to take a look at who is responsible for maintaining networks on the Internet and who provides access to it.

4.1.1 What are Internet Service Providers

An *Internet Service Provider (ISP)* is usually a commercial entity that provides a customer with a way to access the Internet (usually via access hardware, e.g., modem) possibly coupled with a method of authentication/authorization (usually username and password). This means, that ISPs are in fact holders of Internet pieces and commercially offer the possibility to users of connecting to their network to gain access to the worldwide community named Internet.
This is possible for single users as well as companies, which can be given the opportunity to connect their whole network through the one of a provider ISP, thus gaining Internet access.

4.1.2 The TIER hierarchy

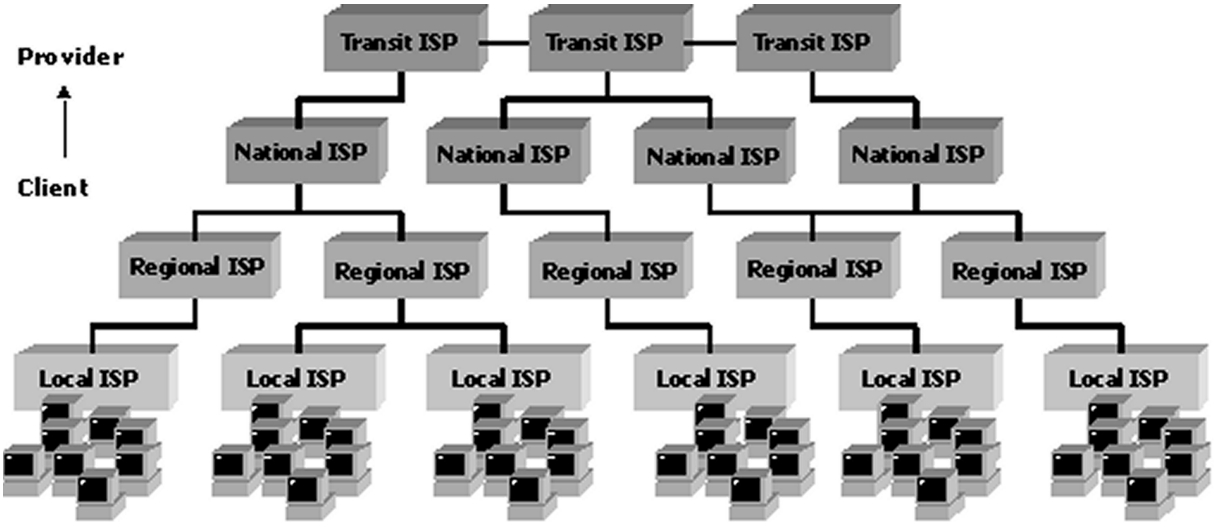

Abbildung 4.1: The TIER Hierarchy

Figure 4.1 shows what is referred to as the *TIER hierarchy*. The Tier hierarchy is simply a model of the relationships between ISPs. Tier-1 ISPs are the largest and peer with each other to provide access to the Internet routing table. Tier-2 ISPs buy connectivity (upstream transit) from one or more Tier-1 ISPs, and can also peer with each other to

minimize the amount of traffic to and from Tier-1 ISPs. Tier-3 ISPs buy upstream transit from Tier-2 ISPs. However, they can also buy upstream transit directly both from a Tier-1 and a Tier-2, and may peer with a Tier-2. This nomenclature is simply a way of differentiating Tier-1 ISPs, which do not buy upstream transit, given that they peer with other Tier-1 ISPs. Connected to Tier-3 ISPs are clients, which range from Local ISPs to individual clients.
In an ideal environment this scheme is quite simply applicable. It all starts at the bottom of the hierarchy: Local ISPs pay Regional ISPs for transmitting their data (more on that in chapter 2 and 4) across their networks, Regional ISPs pay National ISPs and so on. In this picture, every ISP is only connected to an ISP of a higher Level.
Nevertheless, when one talks about the ISP TIER hierarchy, it usually only means Tier-1 and Tier-2 ISPs for there are very many levels below that and it is (in most cases) not possible to differentiate as clearly between the levels that lie deeper in this structure than between the higher levels.
In the real world there may also be several other differences to this picture of the ideal TIER hierarchy. For example it is possible for an ISP to be connected to the networks of several other higher level ISPs. There is also the possibility, that an ISP may be connected to an ISP one level higher as well as an ISP two or more levels higher.
All of this leads to a far more complex view on the logical topology of the Internet, that is presented next.

4.1.3 Major Internet Service Providers and their networks

According to [11], the three largest ISPs in terms of their *connectivity* (meaning the number of networks connected), are

- UUNET/WorldCom/MCI;
- Sprint;
- Cable and Wireless USA.

These three ISPs have their networks spread all over the world. Although they all originated in the USA they are the three largest ISPs worldwide. Their backbones interconnect across different continents as well as across different countries. Their share of connections between Europe and the USA is quite similar, but MCI clearly provides more connections between the USA, Asia and south America as well as connections within Asia itself. This results in a 27.9% share of the worldwide market in contrast to "only" 6.5% that Sprint holds.

4.1.4 Major Internet Entities

Although at first sight it may seem so, the Internet is not a complete chaos in what concerns its different systems and standards as one may suspect. There are several voluntary

entities that have as their primary goal to administer the Internet, all in different areas. The most important of them are:

- The Internet Engineering Task Force;
- The World Wide Web Consortium;
- The Internet Society;
- The Internet Architecture Board;
- The Internet Assigned Numbers Authority;
- The Internet Corporation for Assigned Names and Numbers.

The Internet Engineering Task Force

The *Internet Engineering Task Force (IETF)* [1] is a large open and volunteer international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet. It is open to any interested individual.
The actual technical work of the IETF is done in its *working groups (WG)*, which are organized by topic into several areas (e.g., routing, transport, security, etc.). Much of the work is handled via mailing-lists. The IETF holds meetings three times per year. To become a participant in the IETF, one merely becomes active in one or more working groups by asking the responsible area director to be added to the mailing-list of a specific WG.
The IETF WG are grouped into areas, and managed by Area Directors, or ADs. The ADs are members of the *Internet Engineering Steering Group (IESG)*. Providing architectural oversight is the *Internet Architecture Board, (IAB)*. The IAB also adjudicates appeals when someone complains that the IESG has failed. The IAB and IESG are chartered by the *Internet Society (ISOC)* for these purposes. The General AD also serves as the chair of the IESG and of the IETF, and is an ex-officio member of the IAB.

The World Wide Web Consortium

The]textitWorld Wide Web Consortium (W3C) [2] was created in October 1994 to lead the World Wide Web to its full potential by developing common protocols that promote its evolution and ensure its interoperability. W3C has around 350 Member organizations from all over the world and has earned international recognition for its contributions to the growth of the Web [2].
The W3C has set its own goals for the Web to build the following three items:

- Universal Access - The W3C wants to make the Internet accessible for everyone, no matter what culture, language, education, access devices, or physical limitations there are;

- Semantic Web - the W3C wants to make resources on the Internet easily available for everyone, so the Internet can be used more efficiently;
- Web of Trust - the W3C is aiming at the legal, commercial and social aspects of the World Wide Web, for instance the consideration of different laws in different countries and the building of trust in safe electronic commerce.

Unlike the IETF, the W3C is not accessible to the regular Internet user. It doesn't consist of several members in the sense of people but rather of different organizations which have made theirs the three goals mentioned. Therefore it is open to any organization, providing for it a seat in the *W3C Advisory Committee*, which is a committee consisting of one representative for every member organization.

The Internet Society

The *Internet SOCiety (ISOC)* is a professional membership society with more than 150 organization and 16,000 individual members in over 180 countries. It provides leadership in addressing issues that confront the future of the Internet, and is the organization home for the groups responsible for Internet infrastructure standards, including the IETF and the IAB [3].
The ISOC can be referred to as a mediator between its member organizations. It coordinates the work of its members and tries to solve conflict between them by providing guidelines. The ISOC is lead and governed by its Board of Trustees, which is elected by all of its members around the world (and thus providing the member organizations with a much stronger role than its single members).

The Internet Architecture Board

The IAB is chartered both as a committee of the IETF and as an advisory body of the ISOC. Its responsibilities include architectural oversight of IETF activities, Internet Standards Process oversight and appeal, and the appointment of the RFC Editor. The IAB is also responsible for the management of the IETF protocol parameter registries [4]. Additionally, the IAB also documents the network structure and basic operations of the Internet.

The Internet Assigned Numbers Authority

The *Internet Assigned Numbers Authority (IANA)* [12] was created in 1972 by the U.S. Defence Information Systems Agency in order to regulate and maintain the domain path throughout the network.
IANA was responsible for assigning unique "addresses" to each computer connected to the Internet.
Today IANA's work is mostly done be the *Internet Corporation for Assigned Names and Numbers (ICANN)*, because IANA was built and financed by the US government and

therefore not the independent organization desired by most other organizations concerning the Internet.

The Internet Corporation for Assigned Names and Numbers

ICANN is an internationally organized, non-profit corporation that has responsibility for Internet Protocol (IP) address space allocation, protocol identifier assignment, generic (gTLD) and country code (ccTLD) Top-Level Domain name system management, and DNS root server system management functions. It is essential for these tasks to be performed internationally, so the ICANN charter has to be adjusted in a proper manner. These services were originally performed under U.S. Government contract by IANA and other entities. ICANN now performs some of IANA's functions [5].

ICANN is the replacement of IANA to ensure independency in their decisions concerning IP addressing space and all other tasks previously performed by IANA.

4.2 How is the Internet Organized

This chapter deals with the issue of how the Internet is organized from the routing perspective. It covers Autonomous Systems as well as the aspect of traffic policies between ISPs. Moreover, it deals with the question of how to obtain IP addressing space.

4.2.1 What are Autonomous Systems

Directly connected to the term "Internet" is the one of *Autonomous Systems (ASes)* [13], which are independent systems within the Internet under a local administration. ASes consist of at least one network connected by interior gateways which use the same routing/traffic policies. This is usually known as the *Core network*. Given that these machines are under the same administration, they are considered to be trusted by the ISP that manages the AS. On the boundaries of each AS, *Boundary Routers (BRs)* communicate both with interior gateways and with BRs from neighboring ASes, exchanging traffic policies according to previous established agreements between ISPs.
To exchange such policies, BRs use a so-called *External Gateway Protocol (EGP)*. Currently, the only available EGP is BGP. In its 4th version, BGP-4 became the de-facto EGP on the Internet. BGP is a distance-vector routing protocol: it exchanges information in the form of vectors, i.e., lists of AS numbers (ASNs). The message exchanged is about reachability. For instance, to go from AS 1 to AS 5 one has to cross AS2, AS3, AS4. Hence, the resulting vector is (AS1,AS2,AS3,AS4). The AS nomenclature derived from the use of BGP, and from its way of identifying different networks. As mentioned each AS chooses and manages its own protocols , so there is no common routing strategy in the Internet. The AS-level topology of the Internet provides a macroscopic view of its routing. Moreover, further division is necessary for it is impossible for one single computer or gateway

to maintain the routing information for every other end system in one table. This is the reason why the Internet uses a hierarchical routing structure:

- routing is destination-based, meaning that routers only rely on the IP destination of packets;
- end-systems have only access to specific routing information needed to send datagrams to other end systems or interior gateways in the same (sub)network;
- core routers only exchange information between themselves;
- BRs can exchange routing information with core routers and other BRs of neighboring ASes.

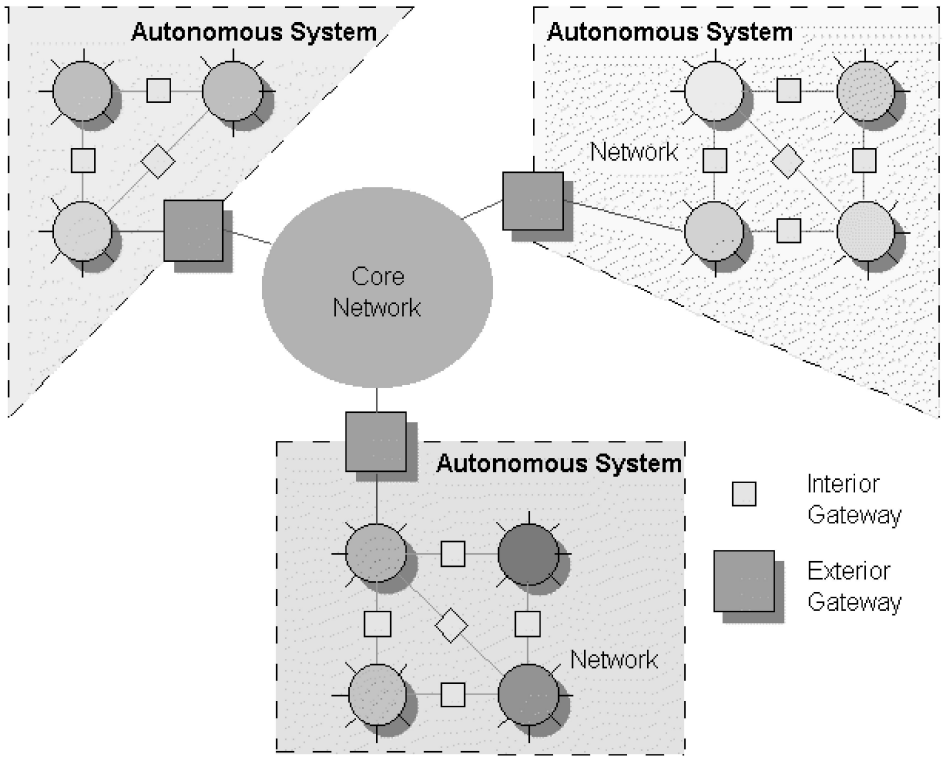

Abbildung 4.2: What are Autonomous Systems ?

4.2.2 Who provides Autonomous Systems

There are several ways to gain access to an Autonomous System, namely:

- through contact to the responsible Internet Registry and filling of the adequate forms. In order to do so one first has to contact two other ASes and agree on peering arrangements with them. This information has to be contained in the forms in addition to what entity will use the AS and the assigned IP addressing space.

- through the connection to a network within an already existing Autonomous System. Then, IP addressing space can be acquired either from the ISP or from the Internet Registry in charge.

4.2.3 Who provides IP addressing space

IP addressing space is provided by Internet Registries. They all work under the supervision of ICANN, but autonomously.
Directly below ICANN there are four *Regional Internet Registries (RIR)*, each providing IP addressing space to a specific world region. ICANN can reassign IP addresses from one RIR to another in case it is needed. The current four RIRs are:

- ARIN (American Registry for Internet Numbers): ARIN provides and administers IP addresses for North America, a portion of the Carribean and sub-Saharan Africa;
- LACNIC (Latin American and Carribean Internet Addresses Registry): LACNIC provides and administers IP addresses for the Latin American and Carribean Region;
- RIPE NCC (Réseaux IP Européens Networkk Coordination Center): RIPE NCC provides and administers IP addresses for Europe, the Middle East, parts of Africa and Asia;
- APNIC (Asia Pacific Network Information Center): APNIC provides and administers IP addresses for the Asia Pacific region (consisting of 62 economies).

Below the RIRs there are *Local Internet Registries (LIR)*. They are the ones, one will usually turn to when trying to obtain IP addressing space. They are divided into groups sorted by size starting with extra small via small and large to extra large.

4.2.4 Traffic Policies between ISPs

As mentioned in chapter 1, the Internet is not the finely structured system it seems to be concerning the TIER hierarchy. In fact, the Internet is a mesh of ASes spread all over the world, each connected to several others either temporarily or over a longer period of time. Therefore, there must be some way to handle traffic between different ISPs owning ASes so constant traffic flow is made possible.
In order to explain traffic policies between ISPs, there are three basic means of interconnection between two ISPs:

- No direct interconnection between two ISPs: In order to get packets from on ISP to the other one, they both have to use intermediaries as transit providers. That way, the mutually exchanged traffic can reach the other side. The interconnection from one of the two to the transit providers can be categorized again;

- One ISP acts as a supplier, another as a customer: In this scenario, one ISP (the supplier) clearly has more traffic coming into and through his system from the customer than going out of it towards the customer. Nevertheless, the supplier in this relationship may as well be the customer or a peer in several other ones. The customer now has to pay the supplier for using its resources, some methods of charging will be introduced in chapter 4;

- Peering arrangement between two ISPs: Each ISP has about the same traffic going into and through his system from the other one as going out of his system towards the other one. If this is the case, the two ISPs often mutually agree on a peering arrangement, each routing the traffic of the other on through their own system without charging the other one.

As mentioned above none of these relationships are static. In an ideal environment...

- Customers want to become peers;

- Peers want to become suppliers.

This is because behind every action on the Internet, one may not forget the commercial aspect, so every customer's goal would be to lower the cost and thus become a peer. Once being a peer, the next step would be to go from having no cost at all to earning money by becoming a supplier.
However, in the real world, a customer may not have any interest in becoming a peer. This might be due to different reasons. For one, the client ISP might not even have the capacity to route traffic through its network. Another one might be, that the ISP follows a strict security policy, letting in as little traffic as possible into its own network, trying to avoid security breaches.

4.2.5 Research Networks

Research networks are networks providing high speed networking to universities, research institutions, schools, cultural entities. Research networks have their own backbone and provide Internet access to their customers. The two most known high speed research networks are

- DANTE (GÉANT): The European Research network;

- ABILENE: The US Research Network.

These two research networks will be introduced shortly for their representative nature for all research networks.

DANTE (GÉANT)

DANTE is a not-for-profit organization whose acronym derives from the name *Delivery of Advanced Network Technology to Europe*. The company was established in 1993 in Cambridge. It is a "Not for Profit" organization and has a special tax-exempt status that has been granted by the UK government [6].
The GÉANT project is a collaboration between 26 *National Research and Education Networks (NRENs)* representing 30 countries across Europe, the European Commission, and DANTE. Its principal purpose has been to develop the GÉANT network - a multi-gigabit pan-European data communications network, reserved specifically for research and education use [7].
GÉANT connects over 3500 research and education institutions in 32 countries. It provides a bandwidth up to 10 Gbps and thus allowing very complex calculations and research activity like DNA Sequencing or Distributed Computing.
A few of the main advantages of GEANT are broadband connections, QoS support, IPv6, and routing know-how.

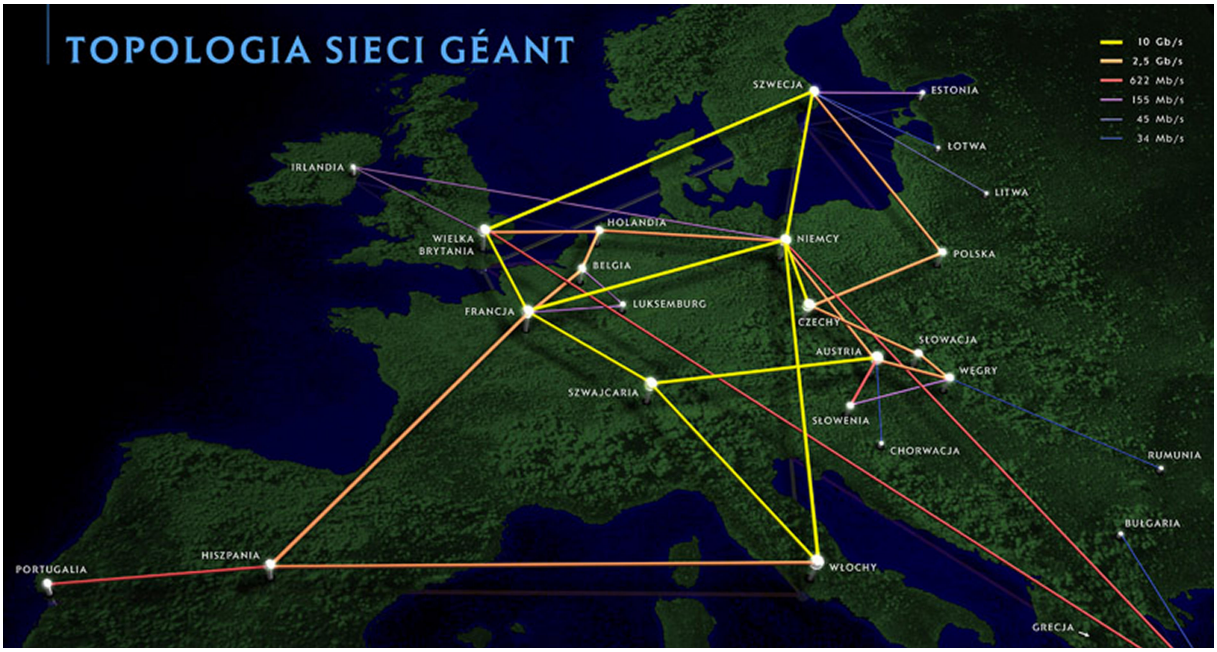

Abbildung 4.3: GEANT Topology Map

ABILENE

The Abilene Network is the Internet2 high-performance backbone network that enables the development of advanced Internet applications and the deployment of leading-edge network services to Internet2 universities and research labs across the USA. The network has become the most advanced native IP backbone network available to universities participating in Internet2 [8].
Thus, Abilene is the US equivalent to DANTE (GÉANT). It provides high speed networking at up to 10 Gbps to universities and research institutions across the USA.

ABILENE is significantly smaller than DANTE (GÉANT) for there are only about 225 entities connected to it.

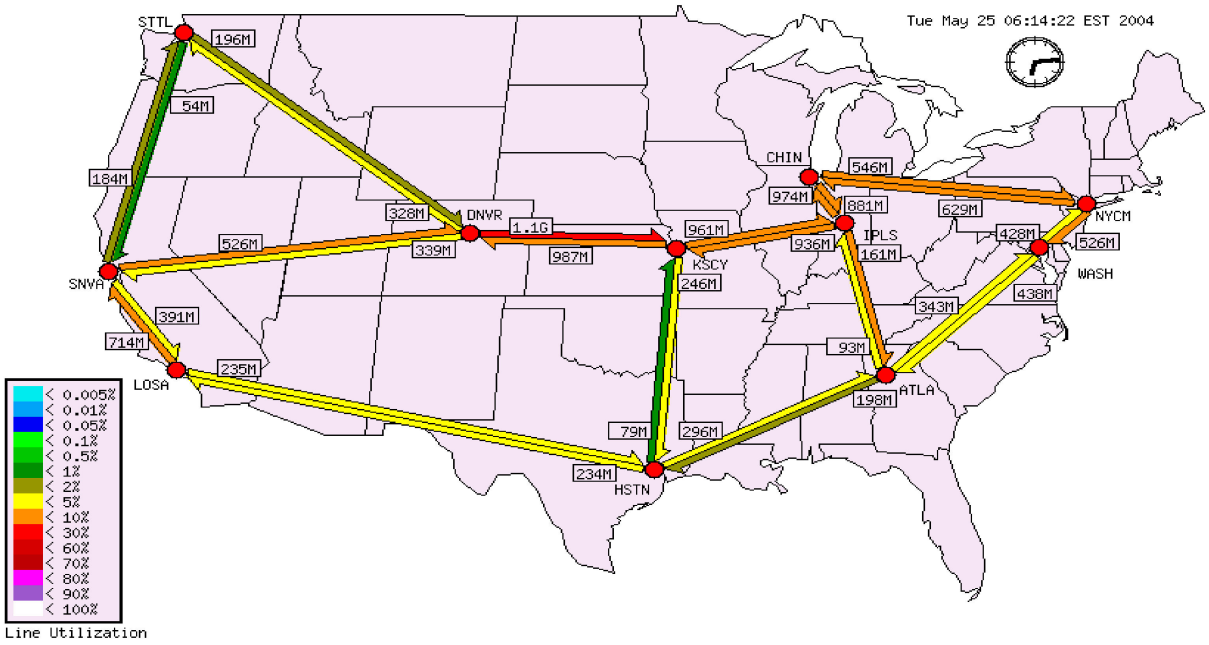

Abbildung 4.4: ABILENE line utilization

Figure 4.4 gives an overview of the most important connections ABILENE provides in the USA. The wide dimensions in connection with the big differences in population density across the USA make it seem as if there are only a few institutions connected to ABILENE, but as mentioned above, there are about 225.

4.3 The Internet in Different Parts of the World

This chapter deals with the international aspect of the Internet. It points out differences in percentage of users, broadband access and worldwide growth of the Internet. Another concern addressed is the aspect of different languages across the Internet, showing which impact English has on it.

4.3.1 Who Uses the Internet

There is a significant difference between different world regions concerning the number of users. Table4.3.1 obtained from [9] clearly shows that Internet access is taken for granted in the USA whereas it is still a luxury to have Internet access in Africa.

In contrast, the worldwide Internet growth shows that world regions with a lower Internet access penetration are catching up to countries like the USA and have a much higher growth rate. This can be explained by just pointing out that developments in Internet accessibility that were made years ago in the USA and Europe are now reaching other parts of the world.

Tabelle 4.1: Internet usage per world region

World Region	Population	Internet Usage	Penetration (percent)
Africa	905,954,600	10,095,200	1.1
Asia	3,654,644,200	236,591,317	6.5
Europe	728,857,380	204,802,658	28.1
Middle East	259,166,000	14,472,500	5.6
North America	326,695,500	226,409,994	69.3
Latin America/Carribean	546,100,900	49,504,287	9.1
Oceania	31,892,487	15,654,781	49.1
World Total	6,453,311,067	757,530,737	11.7

4.3.2 Different Languages Across the Internet

To find out which languages are most commonly used on the Internet, one has to find a method of comparing several web sites and counting the languages they are written in. Fortunately, this can be achieved by a quite simple method (also used by [10]).
The method uses different search engines on the Internet and evaluates the results. [10] used two search engines, Alltheweb [14] and Google [15]. Since it is not possible to enter an empty search string in Google, the search string was set to "pdf" in order to evaluate all pdf documents on the web for most of the have the suffix pdf. Allteweb allows empty search strings and could therefore be used to determine the languages used on web sites themselves. This lead to interesting results:
The most commonly used language on web sites is English with 56.4 %, which is not surprising considering the number of Internet users in English speaking countries. The fact that is more astonishing is that German is the second most used language in Web sites. 7.7% of all web pages are composed in German. Another interesting fact is that Japanese web sites are only on position 4 with 4.9%, Chinese web sites on position 6 with 2.4%.
Another aspect of the Internet usage is the display of different languages and characters. On one hand, there is the possibility to design each web page with a specific character set, having as major disadvantage the need for browsers to support the character set. On the other hand there is the alternative to use Unicode. Unicode [16] is an encoding system providing each character of each character set a unique identifier. Configuring each web browser to support unicode and to publish web pages only using this character encoding would solve the issue of displaying different character sets.

4.3.3 Broadband Internet Access Across the World

Worldwide broadband Internet access statistics show a similar result to statistics related to the number of Internet users around the world. This is not surprising since the number of broadband Internet connections around the world is already contained in the number of Internet connections over all. One thing must be pointed out namely, the fact that although Internet access is considered almost natural in the USA, broadband Internet access is sparsely spread. This can be explained two-fold: the wide USA dimensions, which makes it hard to have a full coverage of all its regions, and the fact that the USA

Tabelle 4.2: taken out of [17]

Country or Region	Broadband Subscribers	Internet Users
Austria	540,000	3,340,000
Canada	3,600,000	16,841,811
Japan	7,806,000	63,884,205
Sweden	693,000	6,913,676
USA	26,200,000	199,096,845

may have been one of the first countries providing Internet access to everyone, but most people are still using old technology such as modems to access the Internet.

4.4 How Internet Access is Priced

This chapter deals with the question of how to price Internet access for end-users as well as for ISPs. Major Pricing schemes are introduced as well as differences in pricing between providers, and end-to-end pricing.

4.4.1 Pricing schemes

Although there are a lot of different pricing schemes for Internet access there are basically three major ones in use, though they may be combined. Those three are:

- Volume-based charging;
- Time-based charging;
- Flatrate.

Volume-based charging as well as time-based charging belong to the so-called parameter-based charging methods, whereas flatrate charging is a group on its own.
Volume-based charging prices Internet access by the volume of data transferred as can be seen in the name. An ISP charges a fixed price per data unit transferred. This pricing scheme is quite popular among DSL users in Germany, for it is quite inexpensive for the user who just uses the Internet for e-mail etc.
Time-based charging makes Internet access dependent on the time being connected to the Internet. This pricing scheme was very common in the beginning of the fast growth of the Internet, since most people used modems to access the Internet.
Flatrate is the most commonly used pricing scheme these days, since it's easy to charge for the supplier and easy to use for the customer, who does not have to worry about how long to stay on the Internet.
Flatrates can be divided into two different types of implementation:
The first one is the real flatrate, which allows unlimited access to the Internet charging

the customer with a fixed monthly fee.
The second one is the flatrate in connection with a limited transfer volume. The customer is still charged a monthly fee, but the data volume transferred is limited to a specified value. Once the customer has exceeded this limit, he is then charged using Volume-based charging.
Another pricing scheme, which has evolved, although it is not used very often is called *Burst-Rate-Charging*. This type of charging can be used when there is the wish not to charge a customer for the amount of data transferred or the time connected, but rather for the bandwidth used.
The ISP periodically measures the volume of data transferred over the connection. For each charging interval, all samples are then sorted by Volume and a fixed percentage from the top of the list is discarded. This is done to eliminate unusual peaks in the data Volume transferred. The highest remaining sample is then taken as the bandwidth used in this charging interval, and based upon a fixed price per bandwith, the connection is then being charged.

4.4.2 Pricing Between Providers

This section tries to answer the question how providers would set their prices when they have to work together to offer a service. One must never forget that each provider acts in his own interest and therefore generally tries to keep its own capacity constraints as private information.
There are several parallels between pricing between providers and traffic policies between ISPs as described in chapter two. As long as two providers have a peering arrangement, there will be no need for either of them to pay the other one for his service. This section deals with the situation where one Provider acts as a supplier and the other on as a customer. Figure 4.5 shows the significance of pricing arrangements between providers in order to be able to offer a certain service to a customer.

Additionally, figure 4.5 also shows the possibility of charging per bandwidth. This requires either the possibility to limit the data transferred through the network to a certain bandwidth or to measure the customer's data transferred by the supplier (e.g., via burst-rate charging).
Although bandwidth prices have been decreasing, another pricing scheme has become popular: flatrate charging is easy for the customer as well as for the supplier. Neither of them to be concerned about the amount of data transferred through the suppliers network, they just agree on a monthly fee and the supplier can route the customers traffic without further thought. This of course isn't quite what happens. The customer may have a strong interest in lowering the fee and therefore presenting the customer with new usage statistics showing a decrease in traffic sent through the suppliers network. The supplier on the other hand profits from presenting the customer statistics showing an increase in traffic flow and therefore trying to raise the fee.

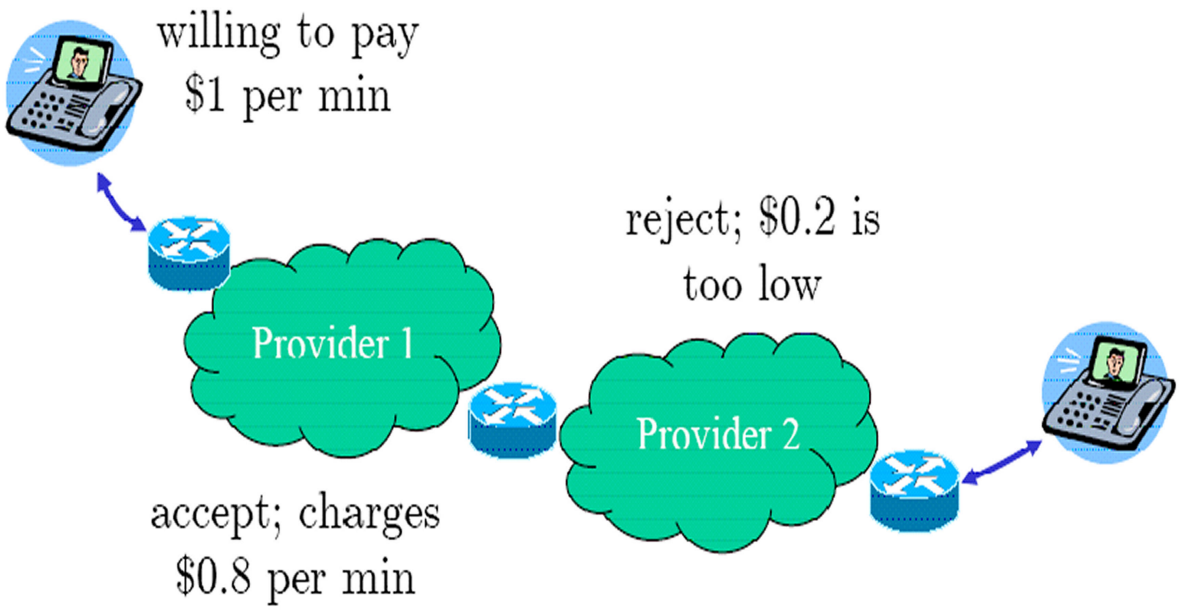

Abbildung 4.5: Pricing between providers

4.4.3 End-to-end Pricing

In end-to-end pricing, all major pricing schemes are used. Nevertheless, flatrates become more and more popular, since they not only are the easiest way of gaining Internet access for the customer but are also an easy (and very profitable) way for the ISP, for though there may be people exceeding the limit, of what is still profitable for the provider, there are still a lot more, who don't nearly reach this limit and thus finance the Volume peaks of others. This clearly show the unfair aspect of flatrate charging concerning fairness among different users. On the other hand, flatrate charging is a very social pricing scheme meaning that every user has to pay the same no matter what social background, income, etc.

Literaturverzeichnis

[1] Homepage of the IETF, http://www.ietf.org

[2] Homepage of the W3C, http://www.w3.org/Consortium

[3] Homepage of the Internet Society, http://www.isoc.org/isoc

[4] Homepage of the IAB, http://www.iab.org

[5] Homepage of the ICANN, http://www.icann.org/general

[6] Homepage of DANTE, http://www.dante.net/server/show/nav.00100g

[7] Homepage of GÉANT, http://www.dante.net/server/show/nav.007

[8] Homepage of ABILENE, http://abilene.internet2.edu/about/

[9] Internet World Stats, http://www.internetworldstats.com/stats.htm

[10] Netz-Tipp.de, http://www.netz-tipp.de/sprachen.html

[11] Russ Haynal, Russ Haynal's ISP Page, http://navigators.com/isp.html

[12] Homepage of IANA, http://www.iana.org

[13] Tom Sheldon, Linctionary.com, http://www.linktionary.com/a/autonomous.html

[14] AlltheWeb Search Engine, http://www.alltheweb.com/

[15] Google Search egnine, http://www.google.de

[16] Unicode Home Page, http://www.unicode.org

[17] International Telecommunication Union, www.itu.int/home/, Dec. 2002

Kapitel 5

XML, Web Services and B2C/B2B: A Technical and Economical Snapshot

Matthias Pitt

Der Kauf bzw. der Handel von Waren über das Internet ist längst für die meisten von uns normal geworden. Im Dritten Jahrtausend kauft man nicht mehr ausschließlich Bücher oder CD's im Laden, sondern ordert sie im Web-Shop von Amazon oder BOL. Neue Kleidung kann man aus dem gesamten Sortiment von Quelle oder Otto online zusammenstellen. Alte oder neue Gegenstände handelt man auf virtuellen Marktplätzen wie Ebay. Doch nicht nur im B2C, sondern auch im B2B sind vernetzte Handels- und Produktionsabläufe längst Standard. So stellt ein Autohändler nach den Wünschen des Käufers ein Auto zusammen und schickt die Anfrage gleich zum Hersteller weiter. Auch Produktionsabläufe in Firmen sind gesteuert, fehlende Teile werden automatisch bei anderen Unternehmen geordert.

Web Service ist eines der Schlagwörter, die erst seit kurzer Zeit die Welt der Wirtschaft beherrschen um die verschiedenen Geschäftsabläufe, die die vernetzte Steuerung von Produktions- und Verkaufsabläufen betreffen, zu ökonomisieren. Die Netzdienste sollen es ermöglichen, standardisierte Daten zu übertragen und entfernte Funktionen durch einen Client auf einem Server auszuführen. Kompatibel soll der Service sein, das heisst mit jeder Programmierschnittstelle soll man einen Dienst konnektieren und nutzen können. Sicher soll der Service ebenfalls sein, das heisst kein Unbefugter kann die übertragenden Daten ausspähen oder verfälschen. Zusätzlich dürfen auch Wartezeiten nicht zu lang werden, um auch die Interaktivität zu wahren. Web Services haben das Potential alle proprietären Lösungen für verteiltes Rechnen abzulösen und damit zu standardisieren. Das ist das erklärte Ziel der größten Softwarehersteller.

Dieser Teil des Seminares wird sich im Folgenden nun mit den technischen Grundlagen von Web Services befassen und anschliessend auf die wirtschaftlichen Konsequenzen der Einführung von Netzdiensten eingehen.

Inhaltsverzeichnis

5.1 Definition von Web Services

Die Technik des Web Service (WS) ist vom World Wide Web Consortium im Jahre 2003 standardisiert worden.

> „A Web service is a software system identified by a URI, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the Web service in a manner prescribed by its definition, using XML based messages conveyed by Internet protocols.“[1]

Ein Web Service ist also eine verteiltes Softwaresystem mit definierten Schnittstellen in XML. Die Kommunikation zwischen den Endsystemen erfolgt dabei, die herkömmlichen Internetprotokolle nutzend, mit XML-basierten Nachrichten. Dieses Softwaresystem kann aber mehr als nur reinen Datenausttausch. Mit Hilfe von XML-basierten Protokollen ist es möglich, beliebige kompatible Web Services zu konnektieren und auch entfernte Funktionen auf anderen Rechnern aufzurufen (RPC - Remote Procedure Call).
Um Web Services (WS) zu erläutern, beschäftigt sich der Text nun im folgenden mit den technischen Grundlagen, der Markupsprache XML [1], anschließend mit dem Aufbau von WS und deren Funktionsweise.

5.2 XML

Die Extensible Markup Language (XML) wurde 1996 entwickelt und im Februar 1998 als Standard vom W3C verabschiedet. Es ist immer noch in der Version 1.0 gültig. Die Version 1.1 steht zur Nutzung bereit, bzw. hat seine Recommendation am 04.02.2004 erhalten[2]. Die praktische Umsetzung für Version 1.1 fehlt allerdings noch.
XML ist eine Metasprache zum Definieren von eigenen Markupsprachen. Dokumente einer Markupsprache enthalten neben der reinen Information, also zum Beispiel einer normalen Textinformation, Zusatzinformationen, die die Textinformation näher attributieren. Dieses Dokument wurde zum Beispiel in der Markupsprache LaTeX geschrieben. Hierbei wird die Textinformation durch eine Escapesequenz vom normalen Text getrennt. Zum Beispiel `\section{Überschrift}` zum Kennzeichnen einer Überschrift.

Man unterscheidet dabei mehrere Stufen einer Markupsprache.

1. **Graphisches Markup**
   Hierbei betrifft die Zusatzinformation nur die Darstellung des Textes. Ein Vergleich mit HTML zeigt, dass das Tag, also die Anweisung `<b> Information </b>` den Text `Information` dementsprechend zur Ausgabe in einer fettgedruckten Schrift vorsieht. Die Anweisungen einer Sprache mit graphischem Markup sind fest vorgegeben und müssen vom Verarbeitungsprogramm richtig interpretiert werden. Dies

[1] http://www.w3.org/TR/ws-gloss, Web Services Architecture, February 2003
[2] Quelle: http://www.w3.org/XML/Core/#Publications

ist der erste augenscheinliche Nachteil des graphischen Markups. Bedeutender ist aber der Verlust an Wissen, die zur Markierung der Information führte. Es ist nicht mehr sofort ersichtlich, warum eine Information fett gedruckt wurde oder warum sie kursiv ist. Vielleicht weil es sich um eine Überschrift handelt, oder um eine wichtige Textstelle ?

2. **Semantisches Markup**
   Die Nachteile des graphischen Markups wurden mit Einführung des semantischen Markups verringert. Semantisches Markup bietet einen Vorrat an Bedeutungen, die verschiedenen Informationen mit gleicher Semantik zugeordnet werden. Überschriften sind als Überschriften erkennbar und haben alle das gleiche Attribut. Mit Hilfe von Stylesheets kann einer Bedeutung eine feste graphische Darstellung zugeordnet werden. Das Dokument ist einheitlich strukturiert. Der Nachteil des festen Pools von Semantiken bleibt aber. Das Verarbeitungsprogramm muss alle Bedeutungen kennen und richtig verarbeiten. Auch die Frage nach der Granulierung der Bedeutungen kann nicht beantwortet werden, wie fein weise ich Rollen zu, wann höre ich auf, den Inhalt zu zerlegen ?

3. **Generisches Markup**
   Aufgeworfene Probleme des Graphischen und des Semantischen Markups versucht das Generische Markup zu eliminieren. Sprachen dieses Markups erlauben die Definition einer festen Syntax zur Strukturierung und von Regeln zur Verarbeitung von Dokumenten. Die Summe von logischen Elementen und von Verarbeitungsanweisungen nennt man Dokumententyp. Ein Dokumententyp kann zum Beispiel ein Buch sein. Dieses habe die logischen Elemente Inhaltsverzeichnis, Kapitel, Überschrift, Abschnitt und darüber hinaus die Regeln zur Strukturierung dieser Elemente. Ein Dokumententyp wird mit Hilfe einer Dokumententypdefinition (DTD) erstellt.

5.2.1 Die Struktur eines XML-Dokuments

Jedes XML-Dokument beginnt mit dem Prolog, der eine Verarbeitungsanweisung enthält. Verarbeitungsanweisungen sind durch die Zeichenfolge <?Liste von Anweisungen?> gekennzeichnet. Der Prolog muss die Angabe **xml version="1.0"** enthalten. Die Angabe **<?xml version="1.0"?>** ist also minimal vorgesehen. Nach der Versionsangabe, kann eine Information über die verwendete Zeichenkodierung folgen. Sie hat die Form **encoding="Kodierungsart"**. XML verarbeitende Software muss mindestens UTF-8 und UTF-16 lesen können.
Nach dem Prolog folgt der eigentliche Inhalt des Dokuments. Dieser Inhalt ist eine Schachtelung von Elementen. Elemente haben wie der Prolog eine fest vorgegebene Struktur. Ein komplettes Elment wird wie folgt dargestellt: `<Elementbezeichner> Inhalt </Elementbezeichner>`. **„Elementbezeichner"** ist dabei ein frei wählbarer Identifikator für ein Element. Der Inhalt eines Elementes kann die reine Textinformation oder eine Liste von weiteren Elementen sein. Elemente ohne Inhalt werden mit `<Elementbezeichner/>` gekennzeichnet. Um ein Element näher zu beschreiben, ist es möglich, dieses Element zu attributieren. Direkt nach dem Elementbezeichner kann eine Liste von Attributen folgen: **<Elementbezeichner attribut 1="wert 1" attribut 2="wert 2" ... attribut**

n=“wert n“> Da die Syntax von XML Groß- und Kleinschreibung beachtet sind die Elemente `<Buch>` und `<buch>` zwei unterschiedliche Entitäten.

Die Schachtelung von Elementen bewirkt die Bildung eines Strukturbaumes, wie das folgende Beispiel zeigt.

```
<?xml version=“1.0“?>
<visitenkarte>
        <person>
                <name> Duck </name>
                <vorname> Donald </vorname>
        </person>
        <anschrift art=“privat“>
                <strasse>
                        <strassenname> Erpelweg </strassenname>
                        <hausnummer> 13 </hausnummer>
                </strasse>
                <plz> 12345 </plz>
                <ort> Entenhausen </ort>
        </anschrift>
        <anschrift art=“geschäftlich“>
        ⋮
        </anschrift>
</visitenkarte>
```

Ein wichtiges Mittel zur eindeutigen Identifikation von Elementen ist das System der **Namensräume**. Man stelle sich ein banales Beispiel vor: Ein Verlag speichert seine Bücher im System als XML-Dokumente. Er verwendet dabei einfache Strukturen wie Buch, Überschrift, Abschnitt, etc. Ein anderer Verlag tut dies ebenso. Weiterhin existiere eine Bibliothek, die diese Bücher im Rechner führt. Bei der Verwendung von Elementnamen kann es nun vorkommen, dass die beiden Verlage gleiche Namen genutzt haben. Um diese Elemente aber eindeutig zu bestimmen, bzw. die Herkunft der Elemente zu identifizieren, verwendet man das System der **Namensräume**. Die XML-Namespace Angabe (`xmlns`) stellt einen eindeutigen Identifikator zur Verfügung, der auf Elemente angewandt die Eindeutigkeit bereitstellt. Ein Namensraum wird im einfachsten Fall im Element selbst durch das `xmlns`-Attribut festgelegt: `<tml:issue xmlns:tml=ẗhttp://www11.in.tum.de/XMLspec/TeachML¨`. Es wird hierbei ein Namenraum „tml “ festgelegt, dessen Herkunft bei der URI `http://www11.in.tum.de/XMLspec/TeachML` liegt. Da URI's (Unique Resource Identifier) eindeutig sind, ist dies auch für alle Elemente des Namensraumes gewährleistet. Das Kürzel `tml` vorangestellt an Elementnamen: z.b. `tml:issue`, fasst diese Elemente zu einer Gruppe zusammen, dessen Herkunft nun bekannt ist. Man kann dann Elemente mit gleichem Namen, aber verschiedenen Namensräumen auseinanderhalten.

Ein XML-Dokument bezeichnet man als **wohlgeformt**, wenn es die XML-Spezifikation einhält. Dazu gehört, dass Elemente richtig gekennzeichnet sind und die äußere Struktur eingehalten wurde. Das bedeutet für das obige Beispiel Visitenkarte, dass das in Person befindliche Element `<name>` auch innerhalb des Elementes `<person>` komplett enthalten

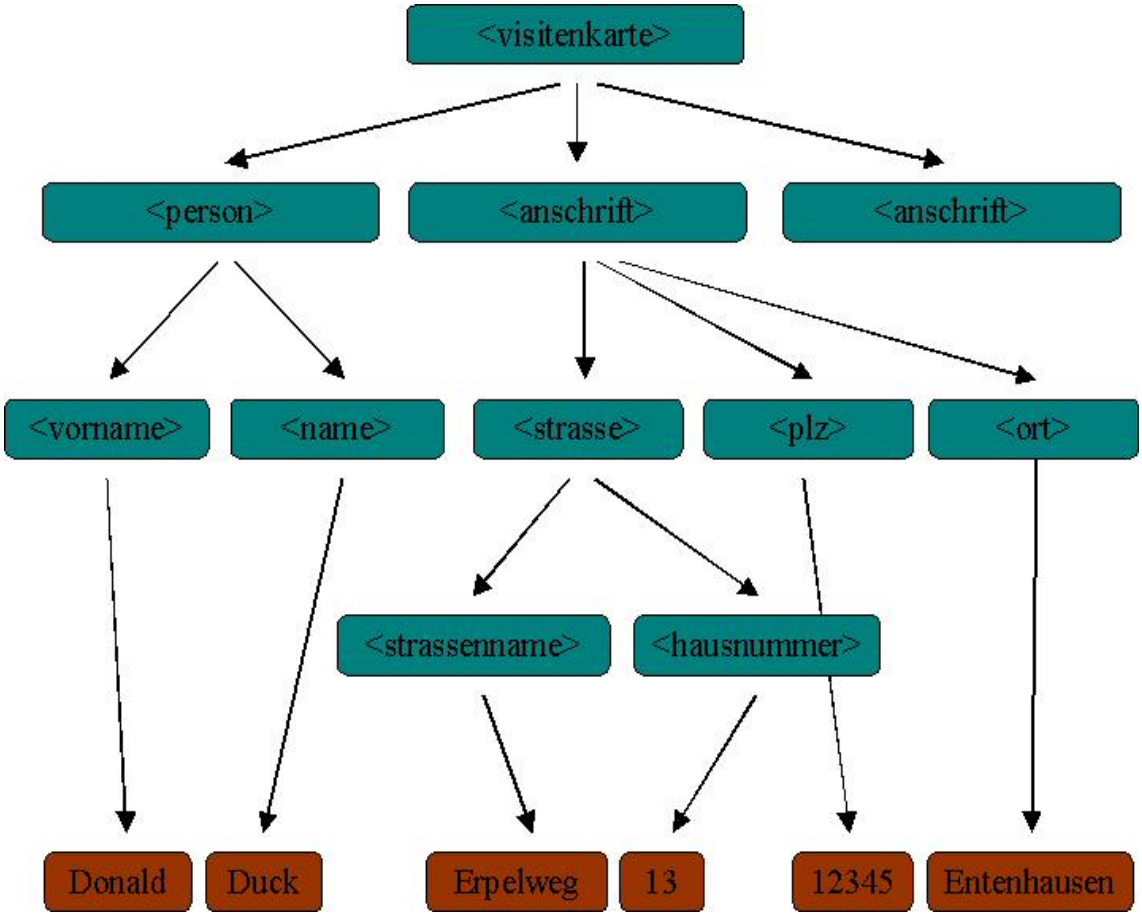

Abbildung 5.1: *Strukturbaum zur Visitenkarte*

sein muss. Also `</name>` muss vor `</person>` stehen.
Als **korrekt** wird das Dokument bezeichnet, wenn die innere Struktur eingehalten wurde. Es dürfen nur fest vorgegebene Elemente verwendet werden und die Schachtelung der Elemente ist ebenfalls zu berücksichtigen. Diese Vorgaben finden sich dann in der Dokumententypdefinition (DTD) wieder.

5.2.2 Die Dokumententypdefinition

Die Dokumententypdefinition (DTD) für XML ist eine kontextfreie Grammatik, die in fest vorgegebener Syntax die Regeln zum Einhalt der Korrektheit eines XML-Dokumentes vorgibt. Für das vorangegangene Beispiel der **Visitenkarte** könnte eine DTD wie folgt aussehen:

```
<!ELEMENT visitenkarte (person, anschrift+, telefon?, freitext?)>
<!ELEMENT person (name, vorname, anrede)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT vorname (#PCDATA)>
<!ELEMENT anrede EMPTY>
```

<!ELEMENT anschrift (ort, strasse, plz)>
<!ELEMENT ort (#PCDATA)>
<!ELEMENT strasse (strassenname, hausnummer)>
<!ELEMENT strassenname (#PCDATA)>
<!ELEMENT hausnummer (#PCDATA)>
<!ELEMENT plz (#PCDATA)>
<!ELEMENT telefon (#PCDATA)>
<!ELEMENT freitext (#PCDATA)>
<!ATTLIST anschrift art (privat, geschäftlich) "privat">
<!ATTLIST anrede type (Herr, Frau) "Herr">

Damit werden dann Elemente, wie z.B. `person` oder `anschrift` definiert und ihre Struktur festgelegt. Darüberhinaus werden Attribute definiert.

Die DTD kann dem XML-Dokument intern oder extern zugeordnet werden. Die folgenden Anweisungen wären dann Bestandteil des XML-Dokuments:

- **Intern:**
  <!DOCTYPE visitenkarte [<!ELEMENT visitenkarte (person, anschrift+)> <!ELEMENT...>...]>
- **Extern:**
  <!DOCTYPE visitenkarte SYSTEM "karte.dtd">
  Diese Angabe sucht eine Datei mit dem Namen `karte.dtd` auf dem Rechner im aktuellen Verzeichnis. Mit der Angabe `PUBLIC` und einer folgenden URI ist auch die Einbindung einer DTD von einer entfernten Quelle im Netz möglich.

Zur Strukturdefinition bietet die DTD im Wesentlichen folgende Befehle [2]:

- Die Angabe `<ELEMENT Elementname Typ-des-Elementes>` definiert ein Element mit dem Namen `Elementname` und schreibt vor, dass der Datentyp des Inhaltes des Elementes `Elementname` beschränkt ist auf die Angabe `Typ-des-Elementes`

  Datentypen:

  (#PCDATA)
  "Parsed Character Data" steht für normalen Fliesstext. Es wird hierbei keine Unterscheidung zwischen Text und Zahl gemacht. Der Wert in `(#PCDATA)` wird hier nicht interpretiert.

  (Elem-1 % Elem-2 % ... % Elem-n)
  Neben Fliesstext kann ein Element auch andere Elemente als Inhaltstyp verwenden. Diese Angabe erfolgt durch eine Listenangabe.
  Als Trennzeichen, an Stelle des %-Zeichens, sind erlaubt:

  , (Das Komma) Die `AND`-Verknüpfung: Das Element enthält eine Menge weiterer Elemente der aufgezählten Typen.

  | (senkrechter Strich) `OR`-Verknüpfung. Das Element enthält ein beliebiges Element der Liste mit einem der aufgeführten Typen.

ANY
Diese Angabe ist zwar erlaubt, sollte aber nicht verwendet werden. Sie steht für eine undefinierte Angabe des Inhaltstypes, basierend auf den beiden obigen Typen. XML-Parser können hier aber keinen exakten Syntaxcheck durchführen.

EMPTY
Es wird ein Element ohne Inhalt definiert.

Beispiele:

- `<!ELEMENT anrede EMPTY>` definiert ein Leerelement mit Namen "anrede"
- `<!ELEMENT ort (#PCDATA)>` definiert ein Element "ort" das beliebige Strings enthalten kann.
- `<!ELEMENT visitenkarte (person, anschrift)>` definiert ein Element "visitenkarte", das als Inhalt die Elemente "person" und "anschrift" jeweils genau einmal enthält.

Für jedes Unterelement kann dessen Häufigkeit vorbestimmt werden, in der Form `Elementname%`. An Stelle des `%` sind erlaubt:

- Kein Zeichen meint genau **ein** Subelement.
- `* (Sternchen)` bedeutet: Das Element "Elementname" kann beliebig oft vorkommen,
- `+ (Plus)` meint beliebig oft, aber mindestens einmal,
- `? (Fragezeichen)` legt eine optionale und einfache Verwendung von "Elementname" fest.

Die Anweisung `<!ELEMENT visitenkarte (person+ | anschrift?)>` bedeutet dann, dass das Element `visitenkarte` mindestens ein Element `person` oder ein optionales Element `anschrift` enthalten muss.

- Zur Festlegung von Attributen dient die Anweisung `<!ATTLIST Elementname Attribut-1 Liste-1 Option-1 Attribut-2 Liste-2 Option-2 ... Attribut-n Liste-n Option-n >`.
  - `Attribut-x` ist eine beliebiger Name zur Beschreibung des Attributes.
  - `Liste-x` definiert eine Menge von Werten, die das Attribut annehmen kann. Erlaubte Werte
    - `(Wert1, Wert2, ..., Wertn) Wert-x` Angabe der möglichen Werte als `String`. Wert-x steht für den Standardwert.
    - `CDATA` Freie Belegung mit einem String möglich.
    - `ID` Ein eindeutiger Identifikator.
    - `IDREF` Ein Zeiger auf eine ID.
    - `NMTOKEN` Ein Token, der auch Sonderzeichen enthalten darf.

    Weitere Werte für Attribute können unter [3] nachgelesen werden.

  - `Option-1`
    - `#REQUIRED`
      Enthält ein Element ein Attribut, so muss es mit einem Wert belegt werden.
    - `#IMPLIED`
      Enthält ein Element ein Attribut, so ist die Angabe optional.
    - `#FIXED`
      Konstante Wertzuweisung
    - `#DEFAULT`
      Ein Standardwert wird festgelegt.

Die Angaben von `ELEMENT` und `ATTLIST` reicht zur Festlegung der Korrektheit von XML-Dokumenten aus. Neben diesen Anweisungen existieren noch:

- `ENTITY`
  Ein XML-Dokument kann aus verschiedenen Speichereinheiten bestehen. Zur Festlegung dieser Einheiten und ihrer Herkunft, dient die Angabe `ENTITY`. Mit der Anweisung können Referenzen auf andere Dokumente bzw. Teile von Dokumenten angelegt werden, z.B. Bilder, die innerhalb eines XML-Dokuments verwendet werden.

- `NOTATION`
  Um die verschiedenen Entities korrekt einzubinden, kann man Notationen vorschreiben.

5.2.3 XML-Schema

Die Bereitstellung von Dokumententypdefinitionen durch das World-Wide-Web-Consortium geschah zusammen mit der XML-Spezifikation schon im Jahre 1998. Daher ist es nicht verwunderlich, dass die Mächtigkeit von DTD's nicht ausreichend ist. Dies liegt zum einen an den fehlenden Datentypen. Es gibt eigentlich nur zwei Datentypen für Elemente: normaler Fliesstext bzw. die Schachtelung von Elementen. Bei Attributen gibt es mehr Typen, aber insgesamt ist die Auswahl ungenügend. Zum anderen fehlt die XML-Konformität. Eine DTD ist kein XML Dokument, dies ist allein schon an fehlenden abschliessenden Elementbegrenzern zu sehen. Dies schränkt die Nutzbarkeit von DTD's stark ein, die fehlenden Namensräume tragen dazu bei. Daher wurde im Jahr 2001 das XML-Schema eingeführt. Es bietet die Mächtigkeit von Programmiersprachen, was die Typenbildung betrifft: Man kann einzelne Datentypen wie `positiveInteger` oder `string` zu Structs (ähnlich C) oder Records (ähnlich PASCAL) zusammenstellen. Die Erstellung von Arrays beispielsweise ist ebenso möglich. Namensräume werden ebenso unterstützt wie auch die XML-Spezifikation. Da man mit einem XML-Schema die Korrektheit von XML-Dokumenten festlegt, kann man einem XML-Schema auch ein anderes Schema zurordnen und dieses damit näher beschreiben, d.h. man könnte XML-Schemata rekursiv aufeinander anwenden.
Auf eine Einführung von XML-Schema wird an dieser Stelle verzichtet und auf [1] und [4] verwiesen.

Hier allerdings noch ein Beispiel für ein XML-Schema, passend zur Visitenkarte:

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:element name="anschrift" type="anschriftTyp"/>
        <complexType name="anschriftType">
                <sequence>
                        <element name="strasse" type="strasseTyp"/>
                        <element name="plz" type="positiveInteger"/>
                        <element name="ort" type="string"/>
                </sequence>
                <attribute name="art" type="string"/>
        </complexType>
        <complexType name="strasseTyp">
        ⋮
        </complexType>
        ⋮
</xsd:schema>
```

5.2.4 Darstellung von XML-Dokumenten

Neben der syntaktischen Korrektheit von XML-Dokumenten spielt auch die Darstellung bzw. Präsentation von XML-Inhalten eine wichtige Rolle. Mittlerweile integrieren viele Hersteller von Software, z.B. Microsoft in die `Office`-Produkte, die Speicherung der Dokumente mit Hilfe von XML. Theoretisch ist es dann möglich, die Informationen, des in Office gespeicherten Dokumentes, mit gleicher Darstellung auch in Konkurrenzprodukten anzuschauen. Der normale Web-Browser ist dabei die zentrale Komponente zur Darstellung, die in der Regel jedem PC-Anwender zur Verfügung steht. Die XML-Komponente **XSL** (Extensible Stylesheet Language) soll genau diese Kompatibilität herstellen. XSL besteht aus den drei Teilen:

- **XSLT**
  Die Transformationssprache `XSLT` kann ein gegebenes XML-Dokument in ein anderes, ähnlich wie XML strukturiertes, Dokument umwandeln.

- **XPath**
  Hiermit ist es möglich mit Hilfe verschiedener Ausdrücke auf Komponenten des XML-Dokuments zuzugreifen. Diese Komponenten sind Elemente, ihre Attribute oder auch definierte Mengen von Elementen mit ihren Inhalten.

- **XSL-FO**
  Diese XML-Sprache stellt Möglichkeiten zur Formatierung von XML-Dokumenten bereit.

XSL bietet zur Darstellung mehrere Möglichkeiten, die Abbildung 5.2 zeigt die verschiedenen Ansätze um das zentrale XML-Dokument zu veranschaulichen. Die Spezifikationen von XSL können hier [13] nachgelesen werden.

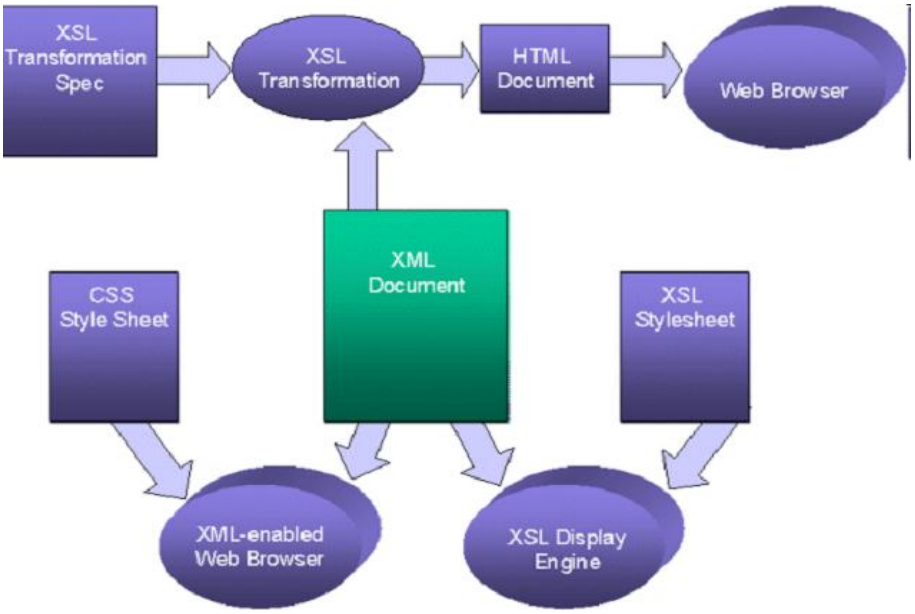

Abbildung 5.2: *Darstellung von Dokumenten mit XSL*

- Die erste Möglichkeit ist die Nutzung eines XSL-Stylesheet. Die besonderen sprachlichen Konstrukte dieser Form können nicht von herkömmlichen Browsern verstanden werden. Es wird daher eine spezielle `XSL-Display-Engine` benötigt.

- Mit Hilfe von bekannten HTML konformen `Kaskadierenden Stylesheets` (CSS) ist es möglich in Browsern, die XML-Dokumente verarbeiten können, auch den Inhalt zu zeigen.

- Die dritte Möglichkeit schliesslich offeriert einen zur älteren HTML-Technik voll kompatiblen Ansatz. Mit Hilfe von Transformationsanweisungen kann man jedes XML-Element und dessen Inhalt mit HTML-Anweisungen sichtbar machen. Eine genauere Beschreibung dieser Technik ist unter [12] nachzulesen. Ein kurzes Beispiel verdeutlicht die Wirkung einer einzelnen Transformationsanweisung:

```
<xsl:template match="Visitenkarte">
    <html>
        <head>
            <title>Anschrift</title>
        </head>
        <body>
        <p>
            <xsl:apply-templates/>
            <–! Wende Inhalt von Visitenkarte
            rekursiv auf Regeln an–>
        </body>
    </html>
</xsl:template>
```

Mit dem Befehl `template match` wird das Element `Visitenkarte` und dessen Inhalt gesucht. Zwischen dieser Anweisung und dem abschliessenden Elementbegrenzer `</template>` folgen diejenigen Befehle, die dann in der HTML-Datei stehen sollen. Das HTML-Ausgabedokument enthält dann also Start- und Endtag, einen Titel und im `<body>` einen neuen Absatz. Genau dieser Absatz muss noch mit Inhalt gefüllt werden. Die Anweisung `<xsl:apply-templates>` schliesslich wendet den Inhalt der `Visitenkarte` rekursiv auf weitere Regeln an, die hier nicht mehr dargestellt sind. Das Ausgabedokument wird also von aussen nach innen aufgebaut.

Diese Transformation kann beim Server selbst stattfinden. Apache-Webserver können hierfür z.B. mit einem XSLT-Modul ausgestattet werden[3]. Diese Server-Variante ist für Clients wie Embedded Systems, Mobiltelefone oder andere Systeme mit geringer Rechenleistung geeignet.
Auf Client-Seite ist die Konvertierung von XML in HTML ebenfalls möglich. Der **Microsoft Internet Explorer** ab Version 5 kann beispielsweise XML-Dokumente mit passendem XSL-Stylesheet darstellen. Die Konvertierung auf Client-Seite hat natürlich Vorteile für den Benutzer, kann er doch mit einstellbaren Standards die Umwandlung nach seinen Bedürfnissen beeinflussen.

5.3 Aufbau von Web Services

Grundsätzlich besteht ein System, das einen Web Service (WS) anbietet (**Application Server**) und ein System, das den Service nutzt (**Client**) aus ähnlichen Softwareschichten, bzw. nutzt die gleichen Protokolle. Das **Simple Object Access Protocol (SOAP)** als Kommunikationsprotokoll, um XML basierte Nachrichten zwischen den Endpunkten des WS auszutauschen, nutzt vorhandene Netzwerkprotokolle und integriert sich in die Applikationsschicht (nach ISO-OSI-Basis Referenzmodell) wie z.B. in `HTTP`. Den Inhalt der Nachrichten und den Web Service in seiner gesamten Funktionsvielfalt definiert das **Web Service Description Language (WSDL)**. Letztendlich ist es eine Applikation die mit

[3]XSLT-Modul „Xalan“, http://xml.apache.org/

standardisierten Schnittstellen auf die über SOAP ausgetauschten und in WSDL definierten Daten zugreift und diese verwaltet. Die Abbildung 5.3 zeigt den Schichtenaufbau eines Application Servers.

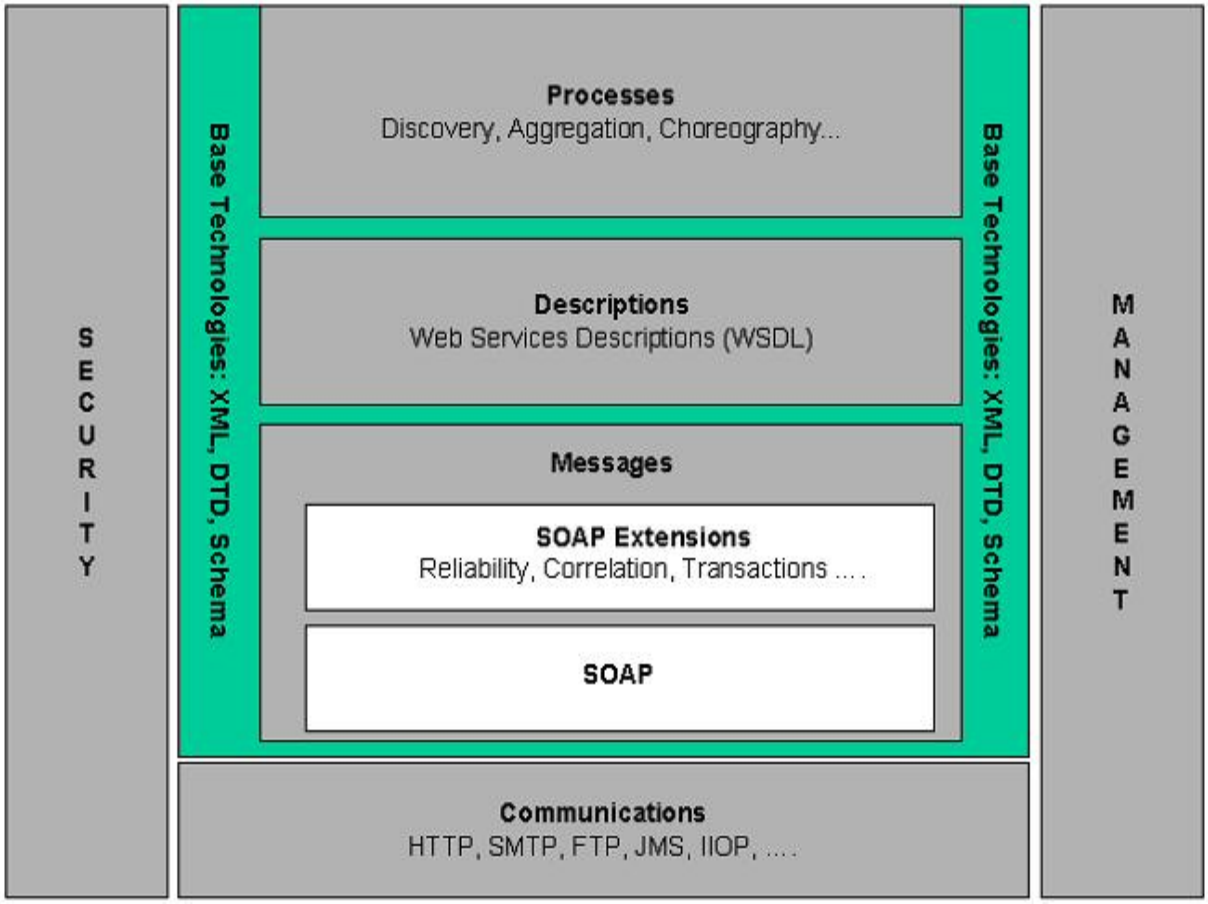

Abbildung 5.3: *Web Service Komponenten*

5.3.1 Die WS-Beschreibungssprache WSDL

WSDL (**W**eb **S**ervice **D**escription **L**anguage) ist eine XML-Sprache. Der Namensraum für WSDL-Elemente ist `http://schemas.xmlsoap.org/wsdl/`. WSDL wurde am 15.03.2001 in der heute gültigen Version 1.1 vom World Wide Web Consortium standardisiert. Version 1.2 und sogar schon Version 2.0 als „Working Drafts“ werden diskutiert. Der folgende Abschnitt beschäftigt sich in einem groben Überblick mit den Spezifikationen der Version 2.0 [5]. Ein Vergleich mit der `Recommendation 1.1` ist mit Hilfe von [6] möglich.

Aufbau von WSDL-Dokumenten

Das Wurzel-Element eines WSDL-Dokuments ist `<definitions>`. Es enthält im Wesentlichen alle Namensraumkürzel, inklusive des oben erwähnten Standard-Namensraumes und des SOAP-Namensraumes. Darüberhinaus wird das Standard WSDL-Schema importiert:

`http://www.w3.org/2004/03/wsdl wsdl20.xsd` Das Wurzelelement enthält die folgenden Unterelemente, hierbei muss man zwischen abstrakten Definitionen und konkreten Definitionen unterscheiden:

- Das `<types>`-Element (abstrakt). Hier werden Datentypen deklariert. Es kommt praktisch immer die XML-Schema Modellierung zum Einsatz.
- Das `<interface>`-Element (abstrakt). Es ist das Kernstück des Web Services. Hier werden alle Nachrichten definiert und ihre Verwendung zwischen Server und Client vorgeschrieben. Das Interface besteht aus einer Menge von Operationen. Die Operationen wiederum bestehen aus einer Menge von Eingabe- und Ausgabenachrichten. Die Vererbung von Interfaces ist möglich. Ein Unterinterface erhält mindestens die Mächtigkeit des Oberinterfaces, es kann erweitert werden. Das `<interface>`-Element muss mindestens das Attribut `name` enthalten, das dem Interface einen eindeutigen Identifikator zuweist. Das `<interface>`-Element enthält die folgenden Elemente:

  1. `<operation>`
     Hier wird die komplette Operation definiert. Sie besteht aus ein oder mehreren der folgenden Elementen:
     - `<input>` Einer einzelnen eingehenden Nachricht werden Identifikator und Datentypen zugeordnet.
     - `<infault>` Fehlerbehandlung
     - `<output>` Einer einzelnen abgehenden Nachricht werden Identifikator und Datentypen zugeordnet.
     - `<outfault>` Fehlerbehandlung
     - `<feature>` und `<property>` Hierfür fehlt eine genaue Spezifikation durch das W3C. Diese Elemente können für nutzerspezifische Daten und Funktionen verwendet werden.

     Die Zuordnung der Nachrichtenrichtung mit den entsprechenden Entitäten `<input>` und `<output>` kann man der Abbildung 5.4 entnehmen.

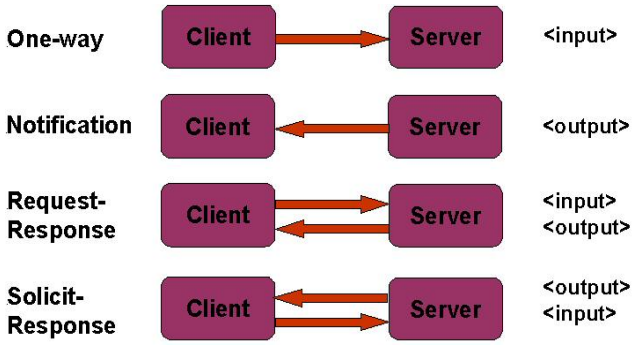

Abbildung 5.4: *Nachrichtenmodelle*

  2. `<feature>` und `<property>` Hier gilt analog die gleiche Nutzbarkeit wie für die Elemente aus der `<operation>`-Ebene. Die beiden Elemente sind in fast jeder Ebene für spätere Erweiterungen vorgesehen.

- `<binding>`-Element (konkret). Es werden passend für jedes Interface mit seinen Operationen konkrete Transport- und Serialisierungsdetails festgelegt.

- `<service>`-Element (konkret). Hier werden die Entry-Points für jedes Interface festgelegt. Sie werden mit den jeweiligen Bindings verknüpft.

5.3.2 Datenübertragung mit SOAP

Das `Simple Object Access Protocol - SOAP` wurde entwickelt, um ein transparentes und multifunktionales Werkzeug zu besitzen, das unabhängig von den verwendeten Netzwerkprotokollen strukturierte Nachrichten zwischen WS-Endpunkten austauschen kann. SOAP ist eine XML-Sprache und wurde vom W3C in der Version 1.2 vom 24.06.2003 herausgegeben, ursprünglich ist es aber eine Entwicklung von Microsoft und etwas später von IBM und SAP.

SOAP ist aufgrund der verwendeten XML-Technologie natürlich unabhängig vom verwendeten Betriebssystem oder anderen technischen Implementierungen des verwendeten Systems.

Aufbau einer SOAP-Nachricht

Jedes SOAP-Dokument besteht aus einem `<envelope>`-Element. Dieses enthält das komplette Regelwerk zur Verarbeitung der Nachricht und auch die erforderlichen Daten. Im `<envelope>`-Element ist ein optionales `<header>`-Element enthalten. Dies hat eine globale Wirkung auf die gesamte Nachricht. Enthalten sind zum Beispiel:

- Verwendete Verschlüsselungen innerhalb der Nachricht

- Transaktionsmanagement: Was passiert beim Verlust von Antworten auf Anfragen, was passiert bei fehlenden Nachrichten, etc.

- Routing der Nachricht: Welchen Weg soll die Nachricht innerhalb von Application Servern nehmen. Teile einer Nachricht könnten für einen als Router genutzten Application Server interessant sein. Ein im `<header>` enthaltenes `<actor>`-Element beschreibt dann das Verhalten des Router's und die Verarbeitung von Teilen der Nachricht. Ob die Verarbeitung optional ist, legt `<mustUnderstand>` innerhalb von `<header>` fest.

Der wichtige Teil der Nachricht ist aber das `<body>`-Element. Im Body werden RPC's ausgelöst oder auch nur einfach Daten übertragen. Die Interpretation der verbundenen Aktion liegt dann beim Empfänger.

5.3.3 RPC mit SOAP

RPC - allgemein

Seit etwa 1984 gibt es den Ansatz von entfernten Funktionsaufrufen zum Ausführen von Berechnungen in verteilten Systemen. Den prinzipiellen Ablauf von RPC (Remote Procedure Call) zeigt die Abbildung 5.5

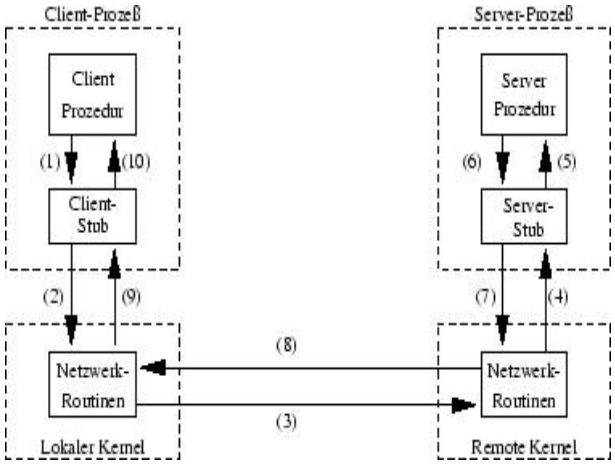

Abbildung 5.5: *RPC - Ablauf*

Ein Programm auf einem Client kann mit RPC auf eine Funktion des Programms auf dem Server zugreifen und eventuelle Parameter mit übergeben. Nach Ablauf der Berechnungen auf dem Server wird ein Ergebnis zum Client übertragen. Zum Übertragen der Parameter packt der Stub des Client's die Daten in ein proprietäres Format. Parameterübergabe ist natürlich nur Call-by-Value möglich, da die Programme der beteiligten Rechner in verschiedenen Speichern oder Adressräumen laufen. Der Server-Stub entpackt die übertragenden Informationen in Nutzdaten.
RPC mit verschiedenen Betriebssystemen oder verschiedenen Hardwareplattformen wirft Kompatibilitätsprobleme bezüglich der Datenformate auf. Zahlenkonvertierungen (wegen verschiedenen Ganzzahl oder Fliesskommatypen) oder Bitshift-Operationen wegen verschiedener Byteordnungen (z.B. Big-Endian und Little-Endian) sind nötig.

Grundsätzlich existieren zwei verschiedene RPC Ansätze:

1. RPC fest integriert in eine Programmiersprache. Hier ist RMI (Remote Method Invokation) in Java zu nennen
2. RPC durch Nutzung eines Object Request Brokers mit Beispiel CORBA.

Die Nachteile beider Architekturen sind auffällig. Erstere ist auf die jeweilige Programmierumgebung beschränkt. Zweitere hat Probleme in der Internetwelt, speziell im Routing von Paketen. Da Internet-Firewalls in der Regel nur HTTP-Pakete durchlassen, könnten CORBA-Daten geblockt werden. Daher wird oft Tunneling der Pakete verwendet. Eine Lösung die in ihrer allgemeinen Verwendung beschränkt ist und Konfigurationsaufwand

bei beiden Kommunikationsendpunkten erfordert.
Daher bietet sich RPC mit SOAP an, da SOAP unter anderem auf dem HTTP-Protokoll aufsetzen kann.

RPC mit SOAP

SOAP kann sich günstig in HTTP-Request/Response einfügen. In einem HTTP-GET wird ein entfernter Methodenaufruf verpackt. Im anschliessenden „HTTP 200 OK“ erfolgt die Rückgabe des Funktionsergebnisses. Die Serialisierung der Funktionsargumente für die SOAP-Nachricht geschieht wie folgt, siehe auch [7]:

- Die komplette Funktion mit ihren Argumenten entspricht einem XML-Schema.
- Der Funktionsname ist gleich dem Namen der ersten Schemaelementfestlegung.
- Argumente der Funktion sind gleichnamige Elemente und deren Typisierung innerhalb des Schemas
- Rückgabewerte der aufgerufenen Funktion werden in einem beliebig benannten Schema untergebracht.
- Sind Argumente oder Rückgaben fehlerhaft oder nicht berechenbar, so wird im jeweiligen `<fault>`-Element eine Fehlerbehandlung übergeben.

Bewertung

Aus diesem Algorithmus wird auch der Nachteil von SOAP schnell deutlich. Es erfolgt zum Übertragen von Daten irgendeiner Softwareschnittstelle (JAVA, C++, etc.) eine Serialisierung und Typkonvertierung. Eine Typkonvertierung deshalb, weil Programm-Typen und XML-Schematypen selten exakt übereinstimmen. Inklusive des Parsen von XML-Dokumenten, welche eine SOAP-Nachricht nunmal darstellt, wird viel Rechenzeit zur Serialiserung verwendet. Diese Rechenzeit ist für heutige Einzelplatzrechner vielleicht vernachlässigbar, ist aber auf stark frequentierten Servern oder `Embedded-System's` mit geringer Rechenleistung zu beachten.

Der Vorteil von SOAP wird aber ebenso schnell offensichtlich, bietet SOAP doch Kompatibilität zu allen Programmierschnittstellen, die auch XML parsen können. Besonders für verschiedenartige Plattformen ist SOAP somit zu empfehlen.
In Verbindung mit HTML ist der Datenfluss durch das Internet problemlos gewährleistet. Auch wenn Firewalls in Zukunft zur absoluten Sicherheit auch SOAP-Nachrichten analysieren müssen, um potentielle Angriffe zu erkennen.

5.4 Die Applikationsschicht eines Web Service

Da WSDL und SOAP nur einen Protokollmechanismus besitzen, der Inhalt und Funktion eines Web Service beschreibt bzw. die Datenübertragung zwischen Endkomponenten standardisiert, fehlt letztendlich eine Softwareschicht zur Nutzung und Verwaltung von Web Services. Da die Schnittstellen mit XML-Sprachen fest definiert sind, ist die Applikationsprogrammierung völlig frei, solange die Programmierumgebung nur XML parsen bzw. verarbeiten kann.

Mittlerweile haben sich viele Anbieter für Programmierumgebungen für WS gefunden. Wobei die beiden größten Anbieter, Microsoft mit der .NET-Technologie und SUN Microsystems mit der ONE-Strategie, den Markt unter sich aufteilen.

5.4.1 Microsoft .NET

Microsoft selbst charakterisiert seine .NET-Technologie wie folgt:

> „Microsoft .NET ist ein Satz von Softwaretechnologien. Mit diesem Paket lassen sich Informationen, Menschen, Systeme und Geräte miteinander verknüpfen.“ `(Quelle: MS Homepage)`

Prinzipiell besteht die Entwicklungsumgebung für .NET aus dem .NET Framework und dem Visual Studio Toolset. Mit diesem lassen sich aber nur „Windows“-kompatible Programme erstellen, also ist man auf Microsoft Betriebssysteme angewiesen. Es bieten sich hier MS Windows 2000, MS Windows Server 2003 oder Biztalk Server mit XLANG, einer XML-Sprache zur Interprozesskommunikation an. Ein Application Server mit .NET-Technologie ist also ein Microsoft Komplettpaket !

5.4.2 SUN Microsystems ONE

Die „Open Net Environment“(ONE)-Technologie ist ein plattformunabhängiges System, das auf Java 2 Enterprise Edition aufsetzt. J2EE liefert Komponenten für eine verteilte geschichtete Software mit: JSP, JDBC, EJB oder XML-Komponenten, wie Parser (JAXP) oder Remote Procedure Call mit JAX-RPC. Für die Entwicklung eines Web Service ist es hierbei egal, ob man mit herkömmlichen Klassen experimentiert oder die Enterprise Java Beans, z.B. als JBOSS-Implementation, benutzt.

5.4.3 RPC mit .NET und J2EE

In der Definition von entfernten Funktionsaufrufen liegt der wesentliche Unterschied [8] zwischen Microsoft's und Sun's Technologie. .NET benutzt zumeist den **Document-Style**, J2EE den **RPC-Style** für entfernte Funktionsaufrufe. Der **RPC-Style** ist die

herkömmliche Technologie mit Funktionsaufrufen in **synchroner** Übertragungsweise und folgt immer dem call/response-Schema. Die Funktionsaufrufe werden mit WSDL beschrieben und sind eng verknüpft mit der eigentlichen SOAP-Nachricht.
Demgegenüber ist der **Document-Style** ein lose Kopplung aus XML-Dokumenten die aber mit XML-Schema eine fest definierte Struktur aufweisen, die der Empfänger richtig interpretieren muss. Die Abarbeitung des entfernten Funktionsausfrufes ist **asynchron**. Eine Antwort der Gegenseite kann erfolgen, muss aber nicht.

5.4.4 Vergleich und Bewertung beider Technologien

Aus diesem großen Unterschied in der RPC-Syntax entstehen natürlich Probleme. Abhilfe schaffen nur die Suche bzw. Frage in einem Forum, wie z.B. bei den SOAPBuilders[4]. Eine Alternative, die die Verhinderung von Kompatibilitätsproblemen unterstützen soll, ist die `Web Services Interoperability` (WS-I)[5]. Die ist ein Firmenzusammenschluss, u.a. mit Intel, Microsoft, Sun Microsystems, IBM und SAP, der es sich zur Aufgabe gemacht hat, die einfache Verbreitung von WS voranzutreiben:

> „WS-I is an open, industry organization chartered to promote Web services interoperability across platforms, operating systems, and programming languages. The organization works across the industry and standards organizations to respond to customer needs by providing guidance, best practices, and resources for developing Web services solutions.“

Die Plattformunabhängigkeit von J2EE, macht Sun's Konzept auch zu einer beliebten Strategie. Eine Umfrage (siehe [9]) zeigte auf, dass das vorhandene Know-How einer Firma (17%) die Wahl der Plattform wesentlich beinflusst, aber die Plattformunabhängigkeit mit 22% der Befragten das wichtigste Kriterium zur Auswahl des Systems ist. Der Kostenfaktor war, ausser für 4% der Befragten, eher nebensächlich.

Aufgrund des Studiums mehrerer Quellen lässt sich ein abschliessendes Urteil über die Vor- und Nachteile der beiden Konzepte nur schwer fällen. Insgesamt stellt sich .NET als das technisch ausgereiftere Produkt dar, konnte Microsoft doch aus den Fehlern von Sun lernen [11] (.NET kam 3 Jahre nach J2EE auf den Markt). Die Plattformunabhängigkeit ist dafür ein klarer Pluspunkt für Sun.
Dementsprechend sind sich Firmen ebenso uneins. Eine andere Umfrage [10] aus dem Jahr 2002 unter 600 Unternehmen in den USA, bescheinigt zwar den Vorteil von SUN in der Anzahl von Projekten: 51% entwickeln unter Java, nur 40% unter .NET . Letztere Plattform konnte aber was die Zukunft angeht, aufholen: so planen 61% der Befragten mit Sun's System, aber ebenso viele mit 63% wollen auch in .NET entwickeln.

[4] http://www.soapbuilders.org
[5] http://www.ws-i.org

5.5 Serviceentdeckung mit UDDI

Sind Dienste erst einmal beschrieben, ihre Verwendung standardisiert und ihr Nachrichtenmodell festgelegt, so sollen sie natürlich benutzt werden. Wenn sich zwei Geschäftspartner schon im Vorfeld auf ein Projekt geeinigt haben, so sind alle Parameter zum Konnektieren des Dienstes bekannt. Was aber, wenn ein frei verfügbarer Dienst angeboten werden soll, bzw. eine Firma ihre Web Services präsentieren möchte.
Hier bietet sich **UDDI (Universal Description, Discovery and Integration)** an. Man kann es als Art von Gelben Seiten (der Post) ansehen, wo Firmen mit ihren Services gespeichert und gesucht werden können. Einträge in eine UDDI-Datenbank sind kostenlos. UDDI wurde von Ariba, Microsoft und IBM als letzte Komponente der Web Services entwickelt. UDDI-Datenbank-Server (`UDDI-Business Registries`) werden neben Microsoft und IBM mittlerweile auch von SAP und NTT-Communications u.a. angeboten. Es existiert eine Umsetzung von UDDI nach Standard 2.0. Wobei der Standard 3.0[6] von Juli 2002 mittlerweile in Testdatenbanken Anwendung findet.

5.5.1 Aufbau von UDDI

Die gespeicherte Information

Durch die dezentrale, aber vernetzte Ansammlung von UDDI-Servern, kann man seine Service-Informationen bei einem beliebigen Server eintragen. Die Information ist im gesamten UDDI-Netz verfügbar. Eine Information über die Angebote einer Firma besteht aus drei Teilen:

Gelbe Seiten
Das Branchenverzeichnis. Es ist kategorisiert nach Art der Unternehmen.

Weisse Seiten
Die Firmeninformationen: Name, Adresse, Kontaktinformationen, Ansprechpartner

Grüne Seiten
Die Services, die eine Firma anbietet, sind hier abgelegt.

Während die Struktur der „Gelben Seiten“ weitgehend festgelegt ist, sind die beiden anderen Daten von der Firma selbst zu erstellen. Dies erfolgt natürlich nach festen Regeln, wie gewohnt definiert duch XML-Schema. Als Tools zum Eintragen der Daten werden von den Firmen Frameworks zum Programmieren einer API oder die Eingabe über ein Web-Interface angeboten, siehe Abbildungen 5.6 und 5.7.

[6] http://www.uddi.org

Abbildung 5.6: *Die drei Teilinformationen (hier im IBM Web-Interface)*

Abbildung 5.7: *Eingabe eines Teiles der Business-Daten*

- **Gelbe und Weisse Seiten**
  Die Informationen über eine Firma und ihre Branchenverknüpfungen werden im **<businessEntity>** abgelegt. Dieses besteht aus den folgenden Teilen:
  - **name** `(benötigt)` - Der Firmenname
  - **businessKey** `(benötigt)` - Ein eindeutiger Identifier
  - **authorizedName** `(optional)` - Name der Person, die den Eintrag erstellt
  - **businessServices** `(optional)` - 0..* <businessService>-Entities, die kompletten Dienstbeschreibungen

  - **categoryBag** `(optional)` - Kategorisierung der Firma
  - **contacts** `(optional)` - Eine Liste von Kontaktpersonen: Name, Email, Telefon, etc.
  - **description** `(optional)` - Eine freie Beschreibung der Firma
  - **descoveryURL's** `(optional)` - Links zum eigenen Unternehmen oder zu anderen <businessEntity>
  - **identifierBag** `(optional)` - Eindeutige Bezeichner in andere Listen: z.B. Handelsregister
  - **operator** `(optional)` - Name des UBR-Operators

- **Grünen Seiten**
  Die Services, die beschrieben werden sollen, sind im Element **<businessService>** abgelegt. Dieses besteht aus den folgenden Teilen:

  - **name** `(benötigt)` - Name des Dienstes
  - **businessKey** `(benötigt)` - Der Link zum Firmeneintrag
  - **bindingTemplates** `(benötigt)` - Die Einstiegspunkte in den Dienst
  - **serviceKey** `(benötigt)` - Ein eindeutiger Identifikator
  - **categoryBag** `(optional)` - Eindeutige Bezeichner in andere Listen: z.B. Handelsregister
  - **description** `(optional)` - Eine Beschreibung des Dienstes im Freitext

  Zusätzlich sind in den Grünen Seiten auch die Einstiegspunkte in den Service beschrieben, dies erfolgt im Element **<bindingTemplate>**.

  - **bindingKey** `(benötigt)` - Name des Einstiegspunktes
  - **tModelInstanceDetails** `(benötigt)` - Technische Spezifikation der Schnittstelle
  - **accessPoint** `(optional)` - Legt die Kommunikationsschnittstelle fest
  - **description** `(optional)` - Ein eindeutiger Identifikator
  - **hostingRedirector** `(optional)` - Verweise auf andere **bindingTemplates**

Den Ablauf eines Eintrages in die UDDI-Datenbank kann man im folgenden Bild 5.8 sehen.

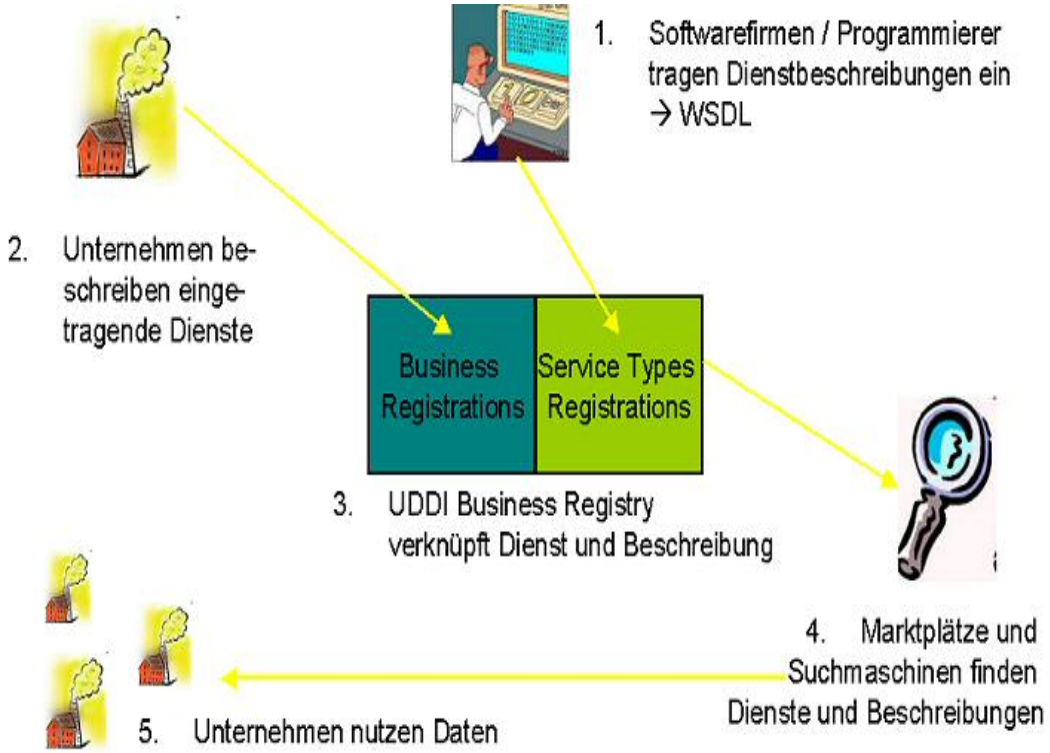

Abbildung 5.8: *Ablauf einer Registrierung im UDDI-System*

5.6 Ökonomische Betrachtung der Gesamtsituation B2B und B2C

Mit B2C, also `Business to Consumer` (oft auch als `Business to Customer` bezeichnet) sind Geschäftsbeziehungen zwischen einem gewerblichen Partner und einem privaten Endverbaucher gemeint. Geschäftsbeziehungen können hierbei die Nutzung bestimmter Dienstleistungen durch den `Consumer` oder auch der Kauf von Waren sein. Man kann es nicht verleugnen: Die Welt des Internetshoppings und damit der Bereich des B2C boomt, zumindest bei den „Großen“ der Liga. Während der Gesamtumsatz des Gewerbes in Deutschland stagnierte (Bruttoinlandsprodukt 2002: +0,2 %, 2003: -0,1%), sind es die Internetportale der Firmen, die zumindest im Bereich B2C Rekordumsätze vermelden. So wuchs der Umsatz des Versandkonzerns **OTTO** im Bereich `Internet` um 24% von 2002 zum Jahr 2003, während der Gesamtumsatz um 2,9% zurückging.[7] Bei OTTO ist diese Umsatzsteigerung des Online-Geschäftes weniger durch den Preisunterschied zwischen Katalog/Ladenware und Onlineshopware zu erklären, denn diese Preise sind bei OTTO identisch. Hier ist wahrscheinlich viel mehr die unkomplizierte Abwicklung des Geschäftes, die jederzeitige Möglichkeit des Online-Shop-Besuches bzw. auch die verbesserten und vermehrten Abbildungen gegenüber den Katalogseiten.
Der eigentliche Vorteil des Onlineshops, nämlich des Einsparens von Ladenflächen und Verkäufern erzeugt bei Internetshops nahezu konstant günstigere Preise. Besonders Spezialartikel wie zum Beispiel Fahrradteile, werden in Onlineshops dauerhaft bis zu 60%

[7] Quelle: www.heise.de

günstiger angeboten. Hier spreche ich aus eigener Erfahrung. Bei Elektronikartikeln sieht es ähnlich aus. Diese günstigeren Preise führten unter anderem dazu, dass die Anzahl der Käufe über Internetportale von Jahr zu Jahr stetig zunimmt: 15,7 Mio. Einkäufe im Zeitraum Oktober 2000 - März 2001, gestiegen auf 20 Mio. im Mai - Oktober 2001.[8] Desweiteren werden Umsatzsteigerungen im gesamten Onlinebereich um 40% pro Jahr bis 2005 vorhergesagt.[9] Diese guten Zahlen sollen allerdings nicht darüber hinwegtäuschen, wo das eigentliche Problem des B2C ist. Der Online-Preiskrieg der Firmen hat einerseits zu vermehrten Firmenpleiten im Internet geführt, z.B. im Jahr 2000: 75% aller Pleiten betrafen den B2C-Sektor und davon über die Hälfte den E-Commerce-Teil[10] anderseits klagt auch der konventionelle Einzelhandel über Umsatzverluste, die nur duch vermehrte und verbesserte Dienstleistungen aufgefangen werden können.

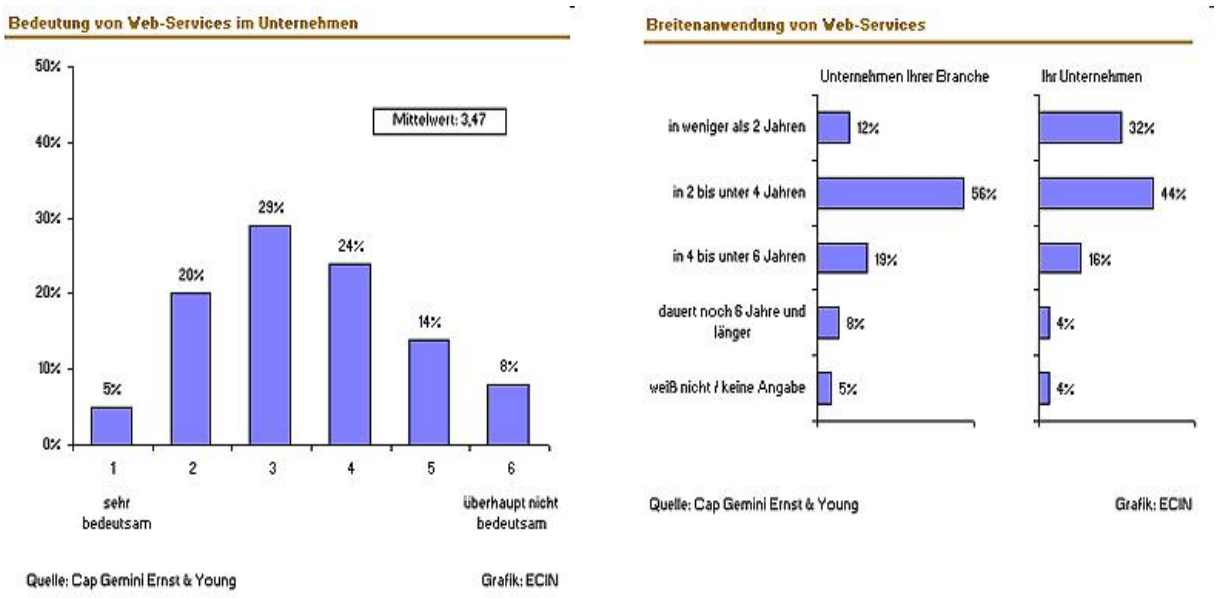

Abbildung 5.9: *Umfrage zur Verbreitung von WS aus dem Jahr 2002*

Der Bereich B2B möchte vom Internet bzw. vernetzten Strukturen natürlich ähnlich profitieren wie der Endverbrauchermarkt. B2B, also `Business to Business` sind wiederum die rein gewerblichen Beziehungen zwischen zwei oder mehreren Unternehmen oder auch Händlern. Der B2B Markt wird statistisch als zehnmal größer als der B2C Markt eingeschätzt. Auch wenn es in der Vergangenheit irrational hohe Überbewertungen des Unternehmenpotentials gab: Im Jahr 2000 betrug der Börsenwert von Yahoo, das 2300 Beschäftigte hatte, etwa 79 Mrd. EUR, Daimler Chrysler mit 467.900 Beschäftigten brachte es nur auf 57 Mrd. EUR. Die Web Services sollen ihren Beitrag zur Steigerung der Gewinnmargen beitragen. Statt teuerer Softwarekonzepte verschiedenster Firmen, sollen sie die günstigere Alternative sein. Die Akzeptanz dieser Dienste ist allerdings noch ausbaufähig. Abbildung 5.9 zeigt die Bedeutung der Web Services in Unternehmen im Jahr 2002. Aufällig ist die fast exakte Gauss-Verteilung mit einem Erwartungswert von 3.5, der exakt getroffen wurde. Allein diese Tatsache zeigt überdeutlich, dass WS zu diesem Zeitpunkt

[8]Quelle: www.onlinekosten.de
[9]Quelle: www.3sat.de
[10]Quelle heise.de

wenig bekannt und verbreitet waren. Die zweite Grafik zur Anwendung von Web Services zeigt zumindest im Unternehmensbereich eine deutlich kurzfristigere Umsetzung der Dienste, diese Tatsache ist allerdings im Hinblick auf die gesamte Branche verständlich, da man dieser als Unternehmen keinesfalls hinterherstehen möchte.

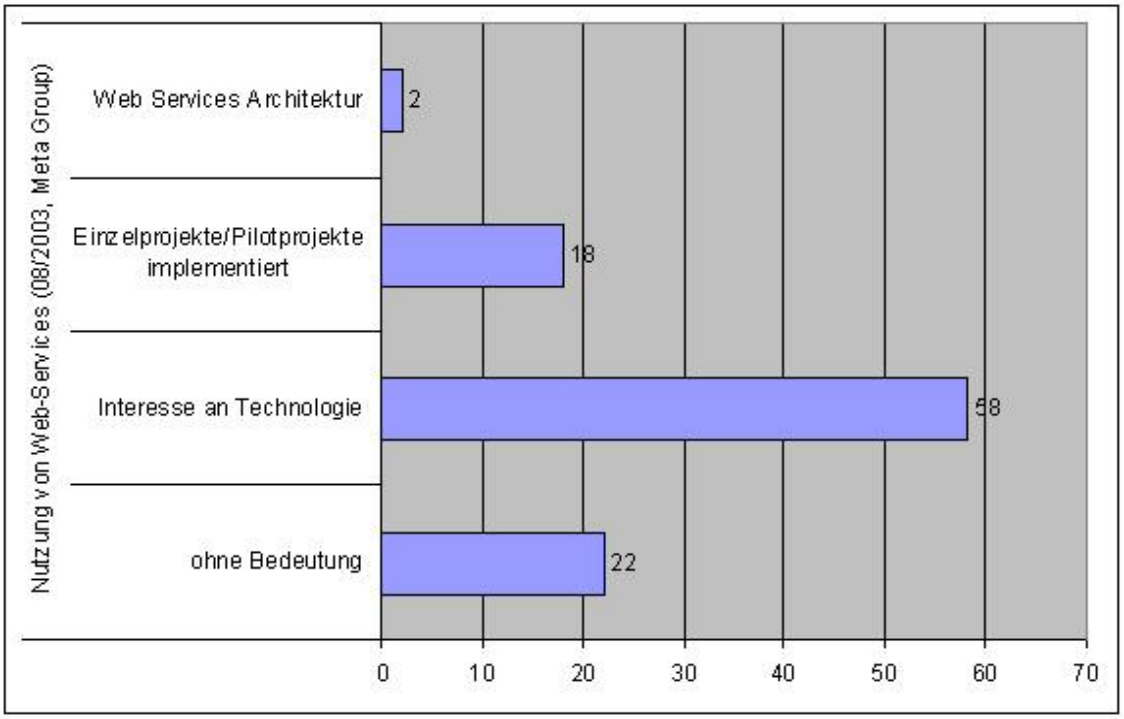

Abbildung 5.10: *Umfrage zur Umsetzung von WS aus dem Jahr 2003*

Eine Umfrage (Abbildung 5.10), die die Inhalte der obigen Statistik zusammenfasst, wurde ein Jahr später angefertigt. Innerhalb eines Jahres ist hier eine deutliche Interessenszunahme an WS zu verzeichnen. Mehr als Drei Viertel der Befragten haben Interesse an der Technologie oder haben Projekte umgesetzt. Diese Umsetzung der Technologie ist aber auch der Schwachpunkt, der in der Umfrage deutlich wird: Erst 2 von 100 Unternehmen haben ein konkretes Projekt im Dienstangebot, etwa 18% testen die Technologie noch aus.

5.7 Zusammenfassung

XML gehört nach Meinung der Softwarehersteller als Datenbeschreibungssprache die Zukunft. Zukünftiges Speicherformat für Dokumente von Programmen wird die Extensible Markup Language sein. Damit wird es zumindest in der Theorie ein standardisiertes und weltweit anerkanntes Format geben. In der Welt der Homepages wird XML nur mittel- oder langfristig HTML ersetzen können, zu verbreitet und optionsreich ist HTML in Verbindung mit Stylesheets. Die kompakte und semantikerhaltende Speicherung von Web-Inhalten in XML-Dokumenten und die anschliessende Transformation von XML zu HTML (mit XSLT) hat größere Chancen zur Durchsetzung.
Die auf XML aufbauenden Web Services sind vom Konzept und von der weltweiten Herstellerunterstützung ein Standard zum verteilten Rechnen, der allerbeste Chancen hat die

etablierten Konkurrenz wie CORBA zu verdrängen. In Entwicklungsumgebungen integrierte Produkte wie z.B. RMI in Java werden aber bestehen bleiben. Es kann hier zu einer Koexistenz von Technologien kommen. Web Services sind ein Schritt zu neuen Möglichkeiten im B2C, aber besonders der B2B-Bereich will damit Produktionszweige ökonomisieren und die Zusammenarbeit im Business vereinfachen und verbessern. Das Interesse an der Technolgie `Web Services` ist mittlerweile vorhanden, aber die nicht vorhandene Vielfalt bereitstehender Dienste zeigt, daß sich die Technologie noch nicht uneingeschränkt durchgesetzt hat.

Literaturverzeichnis

[1] **Skript Hypermedia**, Prof. Dr. Gunnar Teege, Universität der Bundeswehr München, 2003

[2] **Heindl XML Kurs**, www.heindl.de/xml/dtd.html

[3] **DTD-Attributes**, Jan Egil Refsnes, http://www.xmlfiles.com/dtd/dtd_attributes.asp

[4] **W3C XML-Schema**, WWW Consortium, http://www.w3.org/XML/Schema

[5] **W3C WSDL 2.0**, WWW Consortium, http://www.w3.org/TR/wsdl20/

[6] **W3C WSDL 1.1**, WWW Consortium, http://www.w3.org/TR/wsdl

[7] **SOAP**, Lars Stitz, http://www.fh-wedel.de/~si/seminare/ws00/Ausarbeitung/6.soap/soap08.htm

[8] **Oracle**, http://otn.oracle.com/sample_code/tech/java/codesnippet/webservices/docservice/

[9] **Unilog-Umfrage**, http://www.golem.de/0402/29666.html

[10] **Evans Data Corporation auf heise.de**, Heise Newsticker Nr. 31416

[11] **Präsentation, Strategievergleich .NET vs. ONE**, T.Wieland, M.Stal, http://www.cpp-entwicklung.de/downld/Strategievergleich.pdf

[12] **W3C XSLT**,WWW Consortium, http://www.w3.org/TR/xslt

[13] **W3C XSL**,WWW Consortium, http://www.w3.org/TR/xsl/

Kapitel 6

Trade-off between Quality and Cost: QoS vs. Over-provisioning

Ingo Zschoch

The answer to the question of which technique should be used to guarantee quality to a service relies on a critical consideration between complexity and cost. Quality of Service (QoS) concerns the administration of existing resources and their efficient distribution between services. With QoS, some guarantees can be given. But, to provide end-to-end QoS, the operator or provider has to cooperate with other providers along the end-to-end path. Over-provisioning deals with the creation of resources, without guarantees. Over-provisioning is a practical alternative to QoS. It usually attains a higher cost, but also a good performance, being its major drawback a possibly lower performance and higher costs when re-design is needed.
To understand the trade-off between quality and cost, one needs to consider pricing schemes and how they can be applied in current scenarios, namely, either when using QoS or over-provisioning. Currently, there are several pricing schemes. The majority of them are designed for best-effort networks. But some are based on QoS models. With pricing it is possible to influence the users behavior and the efficiency of the network. With pricing, disadvantages of both techniques might be overcome.This work describes the function, problems, costs and use of QoS and over-provisioning as two base strategies. It also provides a summary about the most popular pricing schemes and their impact on QoS and on over-provisioning, as a function of economical efficiency and social fairness.

Inhaltsverzeichnis

6.1 Introduction

The Internet is a best-effort network, initially designed to provide a robust means to exchange data between a small set of computers, through the TCP/IP stack [18, 20, 21]. Best-effort is only a promise of delivery, not being able to guarantee anything, meaning that IP packets (*datagrams*) may get lost. The basic load of the Internet - applications like email and WEB - are not delay sensitive, i.e., they accept longer waiting times for delivery [4]. But today more and more delay sensitive applications, such as real-time video streaming, *Voice over IP (VoIP)*, multimedia or online gaming gain tremendous popularity. A VoIP example is illustrated in Fig. 6.1. These type of services require higher performance and guarantees for faster or assured delivery. Some forms of such guarantees are delay, latency, and packet-lost sensitive, i.e., services require some form of *Quality of Service (QoS)* [4, 10]. To guarantee quality to such applications, the best-effort service is not enough, given that there is no way to choose which traffic has a higher priority [21]. Additionally, the heterogeneous and self-regulation of the Internet makes service differentiation harder.

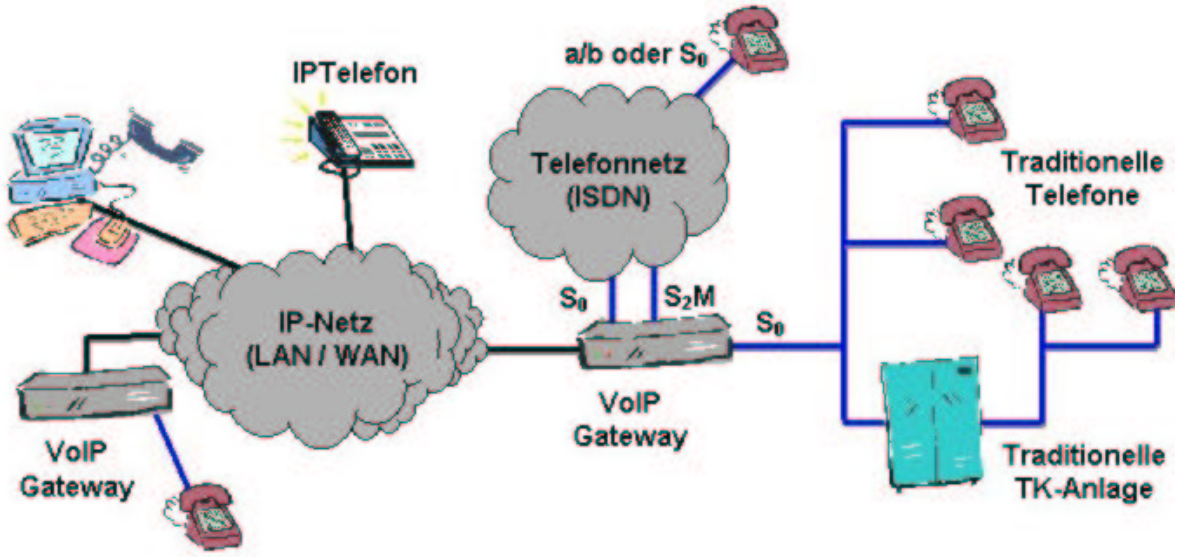

Abbildung 6.1: VoIP example

A practical example of such requirements is the scenario where some Internet user buys a PC to play some (online) games. After some months, or even after some weeks, more workspace, a faster CPU and larger hard-drives are required to speed up the performance of the computer. End-users want to have more "quality" independently of new applications. They wand to make their devices "faster" so that existing applications can also run faster. This new performance is not really needed because the requirements to the performance of the PC do not change but the user want to have more performance to be prepared for future games or load peaks. The behavior of these users is a form of over-provisioning, given that they are enhancing their machines in advance, to achieve a better performance on the long run.

This paper addresses two different strategies that providers use nowadays to provide some

form of "quality", namely, QoS and *Over-provisioning (OP)*. The paper provides a description of these strategies and analyzes the trade-off quality/cost for each of them, being organized as follows. Section I provides some QoS definitions, requirements, how it works, and where is it used or implemented. Section 2 presents the OP strategy including possible definition, requirements, problems, and use cases. The advantages and disadvantages of both strategies are compared in Section 3. Finally, Section 4 describes why we need pricing schemes, providing a brief description on ten of them. In the description of the schemes a variety of criteria have been used, like provision of individual QoS guarantees, impact on social fairness, and degree of network and economic efficiency.

6.2 Quality of Service

6.2.1 Definitions

There are several definitions of QoS, which differ a lot in meaning. For starters, the *International Organization for Standardization (ISO)* [10] mentions QoS as a concept to describe and summarize different control criterion of a network with QoS-parameters. This definition is called *QoS-Syntax*. Other experts [15] claim, that QoS is a form of supplying a guaranteed performance with service differentiation. These guarantees have to be specified in *Service Level Agreements (SLA)* [8, 21] between the involved parties. A third definition of QoS [13, 15] is the ability of a data-network to guarantee some speed to an application. Also, ITU-T [10] describes QoS as the increase of service performance to improve the user satisfaction. This means to give the user more performance as he needs it, and to satisfy him in such way. The different definitions show that QoS is a very complex subject. One speaks about terms such as *service* and *quality*, but in fact, what is a service and what is quality? We will address these issues next.

Services and Quality

The term *service* [2] introduces ambiguity; depending on how an organization or business is structured, service may have several meanings. People generally use the term service to describe something offered to the end-users of any network, such as end-to-end communication or client-server applications. Services can cover a broad range of offerings, from electronic mail to desktop video, from Web browsing to chat rooms. In multiprotocol networks, a service can also have several other definitions. In a Novell NetWare network, each *SAP (Service Advertisement Protocol)* advertisement is considered an individual service. In other cases, service may be categorized according to the various protocol suites, such as SNA, DECnet, AppleTalk, and so forth. In this fashion, one can bring a finer level of granularity to the process of classifying services. It is not too difficult to imagine a more complex service classification scheme in which services might be classified first by protocol type, and then by a finer-grained level within each protocol suite.

Quality [2] can encompass many properties in networking, but people generally use the term quality to describe the process of delivering data in a reliable manner or in a manner

somehow "better" than the normal way. Quality includes the aspect of data loss, minimal (or no) induced delay or latency, consistent delay variation (also known as *jitter*), and the capability to determine the most efficient use of network resources (such as the shortest distance between two endpoints or the maximum efficiency of a circuit). Quality also can mean a distinctive trait or distinguishing property. so people also use quality to define particular characteristics od specific networking applications or protocols.

Types of Service

When we speak about different services, we have to specify some *types of service (ToS)*. ToS is the definition of semantics on a per QoS-parameter basis for a special service. Such specification is called *QoS-Semantics* [10]. For instance, best-effort and *guaranteed services* can be differentiated. Best-effort services [10] have no guarantees in what concerns possible QoS parameters such as delay, jitter, or bandwidth. There is no resource reservation for such services, and no need for any type of monitoring. The specification of QoS-parameters is not necessary. On the other hand, guaranteed services require some form of resource reservation. For such services, every party involved has to keep some form of QoS specification. And, to control the guarantees, one has to enforce monitoring. Additionally, there are two types of guarantees [10]: *deterministic* (strong) and *stochastic* guarantees [10]. Deterministic guarantees ensure that the required quality is always guaranteed. Load peaks are the approximate value and resource reservation is inevitable. The reservation is required for the worst case. On the other hand, stochastic guarantees are statements like "the delay will be lower than 4ms in 99.99 %". There is only a reservation for an average value. In the case of stochastic guarantees it is possible that the quality of the service will be more badly than the guaranteed one in 0.01% of the specified time. The distribution of losses has to be specified in a SLA.

For example, file transfer is a best-effort service, but there is the possibility of a stochastic guarantee for bit-errors. In contrast, there is the example of video streaming. The minimum quality has to be deterministic, but additive achievement could be stochastically guaranteed. These two examples show that services have different guarantee requisites.

QoS-parameters, Metrics, and Views

In the previous sections it was mentioned that the syntax of QoS is defined as a set of parameters which are the requirements to a network. Additionally, there are also some metrics, which hold a different meaning depending on one's perspective. Such different perspectives can be grouped into three major classes: *(network-)provider, service-supplier, and end-user*. The goal of the provider is a high extent of utilization and low costs for operation. The service-offerer, which holds the role of seller in this document, obtains functionality from a provider for user contact and for the delivering of services [13]. For the end-user or the customer of services, guarantees like security, safety, integrity of data, privacy, and delay are very interesting and user-appealing to offer in his services. Other metric discussed is *capacity*. This metric is very interesting for network operations. Capacity refers to the resources available on a network *link* and is usually associated to bandwidth.

Another possible metric is the *error-rate*. This is interesting for the provider. The error-rate could influence the transmitted data so that retransmission is required. Error-rate is an important characteristic for network evaluation.

An important metric for higher layers in the ISO/OSI model is jitter,[13]. Jitter is introduced by networks in data flows due to variable transmission delays. Reasons for jitter are variable service times of intermediate nodes, such as routers or switches. The use of different data paths for data elements of a single flow requires constant jitter [10]. This value is important for real-time applications. If a maximum end-to-end delay is guaranteed and a maximum buffer length at the receiver is available, the jitter can be compensated [10]. Another provider-related metric is response time, which is particularly important for service-offerers e.g., e-commerce. For service-offerers, long response times are very expensive because users tend to cancel their sessions when they have to wait longer than 6 seconds. A larger share of the response time is the latency, resulting from transmission. A final metric for networks that we discuss is the time for connection establishment and preservation time. These metrics are crucial for user and for service-offerer [13, 10].

6.2.2 Goals

The QoS technique has very specific goals. QoS aims at distributing the available resources of a network between different services in a way so that the required QoS-parameters can be guaranteed. In other words, this means that service differentiation and controlled resource reservation is required to apply QoS. The result is an assured quality for the service at a certain time [12]. QoS has not as goal to create new resources, but to use existing resources efficiently. Nevertheless, QoS does not replace the need for capacity planning and network dimensioning [12]. How these goals are achieved is a process described in the next sections.

6.2.3 How it Works

To guarantee efficient QoS to services, one has to provide some form of service differentiation. So every service get the resources he needed and not every service the entire resources. This service differentiation today is implemented with models such as *Integrated Services (IntServ)* or *Differentiated Services (DiffServ)* [19], where different services get different priorities. So, services with "higher" priority, according to some pre-defined metric, get a better quality. DiffServ deals with how data is forwarded (*data plane*), that is, how services can be differentiated. However, there is still the need for a mechanism to manage and check resources, i.e., a QoS *control plane* mechanism. On the control plane, a possible way of performing resource reservation is to use the *Resource Reservation Setup Protocol* (RSVP) [2, 3, 4, 10, 11, 12]. Additionally, there is a separation between the *inter-domain* and the *intra-domain* resource reservation processes. Intra-domain, the provider or operator can reserve and administer resources freely, given that such environment is trusted. Inter-domain the operator is dependent on cooperation with other neighboring operators. Intra-domain the admission control is feasible, but inter-domain other techniques have to used, e.g., a superordinate administration of higher tiers.

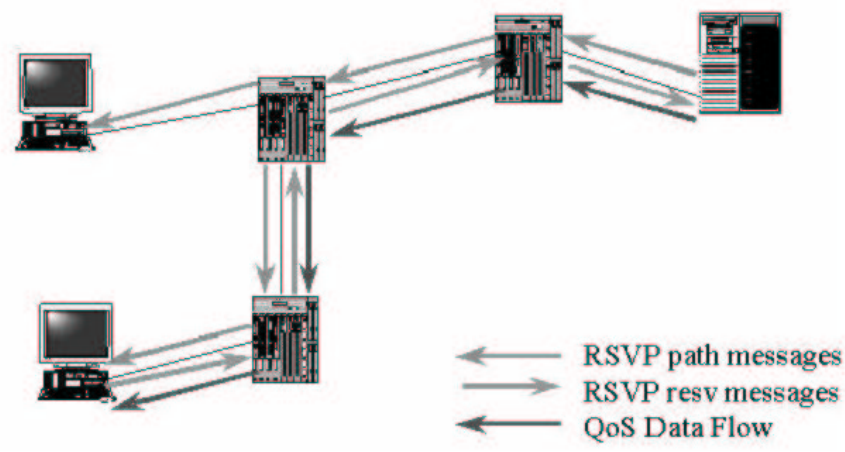

Abbildung 6.2: RSVP principle [21]

The reservation for packets is priority driven. The recognition of a data flow is efficient because the entire flow could be handled instead of every packet. Using IPv4 alone, this flow recognition is very difficult to achieve. One way to do this would be to use different ports for different flows. But this would be very hard to code. A possible future solution is IPv6, which integrates the *Flow Label* [2, 10, 13, 21] field and priorities in the header. So, the entire flow can be managed.

Another possible example of a control-plane, and the one in fact used by default in DiffServ is a static approach, where SLAs are established so that end-to-end QoS can be achieved. Still, to perform QoS, additional management and methods are required, as is explained next.

Management and Methods

As described in section 6.2.1 several professionals have different views on QoS and its different parameters. Due to such nuances, different entities perceive different levels of QoS in different ways. Different components of the communication architecture require distinct parameters [10]. For instance, end-users consider only high, medium, and low quality levels. An application (e.g., video streaming application) has levels of quality related to the media used, such as frames. The communication system specifies communication quality parameters such as delay and error-rates. And, the operating system specifies system-related quality parameters like priority or scheduling and so on. One QoS requirement is to translate these different views into one another, which is quite complicated process.

Besides the different QoS levels, and mapping them to different perspectives, there are some other methods and management required to realize QoS from an application to the network. The methods describe the active task of QoS models, in contrast to static issues, such as QoS parameters or ToS. QoS management entities are divided into QoS parameter, QoS contracts, ToS and service classes. QoS methods are QoS mapping, Traffic Shaping, QoS negotiation and QoS monitoring. Resource management requires resource reservation, association, admission control and QoS enforcement. Finally, pricing and tariffing is required, as described in section 6.5.

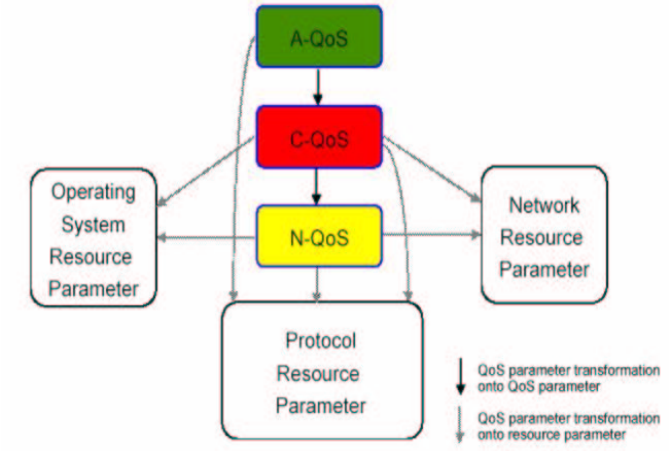

Abbildung 6.3: Mapping overview [10]

QoS mapping, also called QoS transformation, is related to the mapping of different perspectives of QoS. Such perspectives are a performance and functionality oriented performance (P-QoS), application-oriented QoS (A-QoS), communication-relevant QoS (C-QoS), and network-dependent QoS (N-QoS). Fig. 6.3 shows a conceptional overview. It shows that the QoS parameters have to be transformed into resource parameter, too. The target is to provide a comparable view of requirements, to offer the potential to negotiate QoS, and allows processing admission control tests. For mapping, some assumptions are considered, namely, that fixed, operating system, network QoS parameters and protocol resource parameters are specialized parameters.

Monitoring is used to supervise a connection, a service, a task, a protocol, or a program. QoS monitoring requires measurement for diagnostic and to localize errors or congestion, and extracts information on the current behavior of a service. The time scale on which users and machines perceive QoS is, obviously, orders of magnitudes smaller than minutes. One of many possible reasons for degradation of the perceived quality, is congestion on links along the path [12]. The traffic bursts in a small timescale are not significant. Monitoring can be *passive* or ac*t*ive. Monitoring is the control of current values and behavior of negotiated and contracted QoS parameters and important to enforce SLAs. The monitored information has to be stored.

Traffic Shaping is used to shape the user data to be transmitted, to comply with a previously defined and agreed QoS contract.

QoS negotiation is a method to adjust or to align distinct QoS views of different peers to a commonly accepted end-to-end QoS. The negotiation is a very important method between network operators.

The next required management step is the QoS contract. The QoS contract is a template to specify the result of a QoS negotiation and the final resource reservation and admission. It contains certain agreed upon QoS parameter values and traffic patterns for a particular data flow or entire application. It provides the agreement on QoS for services user(s) and the service provider as a direct output of QoS methods.

ToS (see 6.2.1) is the definition of semantics on a per QoS parameter basis for a special service [10].

Service Classes deal with the classification of different services into different classes to simplify the management. Handling of classes is simpler and less costly than handling each individual service.

One piece of resource management is resource reservation. Resource reservation commits resources along a path that has been determined by a routing protocol like it is described in 6.2.3. A difficulty is the reservation of resources within network neighbor systems basis (link-by-link) and peer basis (end-to-end).

Resource association is a mechanism to associate resources within an end-to-end based on a given traffic shape and on the computed QoS of this resource providing by the admission control. The admission control is the guard of the resource management. It is a mechanism to allow for accepting or rejecting new or joining flows for a given architecture including specified resources. Admission control assumed that resources have to be quantifiable and availability have to be stored. QoS Enforcement based on monitored QoS. The enforcement i a mechanism to enforce the QoS contract and to maintain current QoS as close as possible, while controlling and scheduling and-system resources. In the view of the author Pricing is one key mechanism to influence the behavior of users in an economical way.

Pricing is the determination of a scheme that maps delivered services onto monetary units in an open market or rates in a regulated market (tariffs). Pricing is discussed in section 6.5.

To summarize, Fig. 6.4 provides an overview of the QoS management and QoS methods described here, depicting the complexity and difficulty to realize QoS from the application to the network layer.

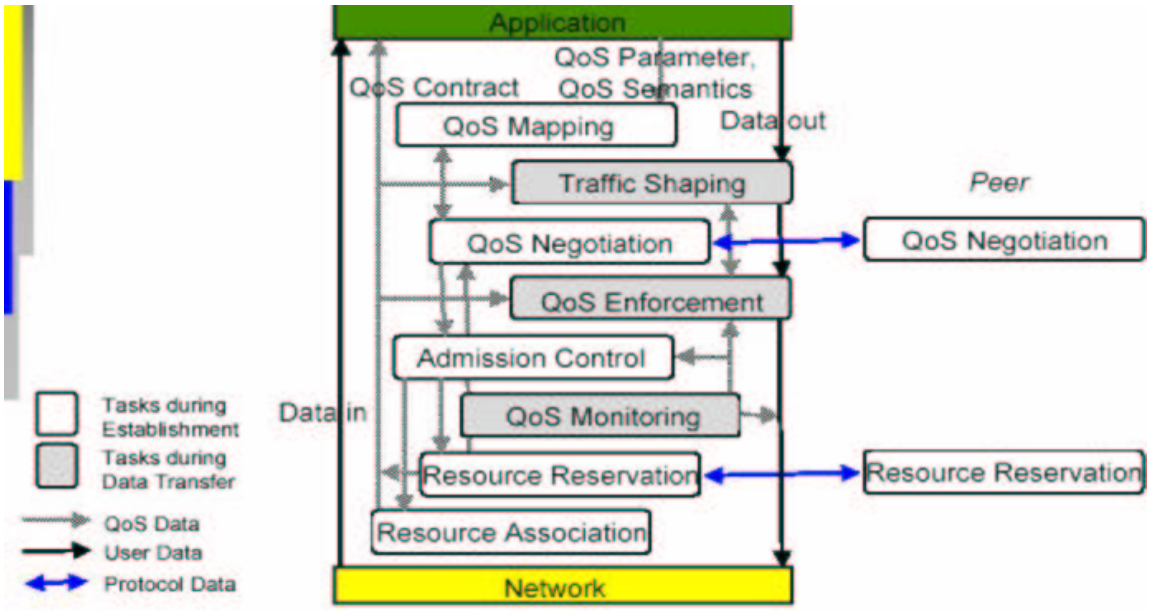

Abbildung 6.4: QoS technical requirements [10]

6.2.4 Requirements and Cost

The previous section provided an operational overview of QoS, its methods and management, patenting the QoS complexity. Often, the requirements for such a technique are in fact the major contributor to its problems. That is the case for QoS. Dynamic change of state and exchange of QoS information is required to deal with application requirements providing the requested service. The QoS methods discussed in the previous section operate in different fields of the end-system, during distinct states of communication protocols, and with control data (QoS information) expressed in QoS parameters or on user data. To support QoS, additional software is required (e.g., the implementation of protocols like RSVP).

Resource reservation encompasses a link-by-link communication between participating end-systems and involved intermediate systems. QoS might require specific hardware modules applied at the boundaries of domains (e.g., monitoring tools which can be applied at ingress or egress routers). Finding the proper user profile is not a trivial process on the provider side. The applications have to specify requisites. Customers or services (users or applications) have to specify QoS parameters. The classification and evaluation of the packets adds some, even though little, processing in edge routers. Additionally, there is the need to have a concept for inter-domain cooperation up to the end systems [12]. The resource reservation requires either the use of dynamic reservation protocols like RSVP, or of a static approach such as the DiffServ *Bandwidth Brokers*.

Cost comes in many forms. Equipment costs, for example, often are less than 20% of the total cost, but then comes the expense of technical support, network operations and service administration. The implementation of QoS introduces complexity and a set of decisions to be made at all business levels that did not exist before [24]. In particular, many users and applications lack the ability, or understanding, or both, to determine the exact level of QoS they need and should require from the network. Without QoS for IP, there is not an effective way for carriers to guarantee a level of performance for each of several different applications and users across many customer locations. Qualitative applications require some level of QoS, but the network administrator must decide how much they need to perform correctly. One of the main disadvantages of QoS is that the costs are not related to equipment and network installation and upgrade, but to management costs. The software costs are minimal, given that they result from a one-time only process. Buffering and analyzing packet header is no overhead and feasible. And, QoS guarantees can be achieved in edge devices with minimal impact [25]. Still, QoS carries the cost of investing in inter-operator contracts and SLAs. The saving of data for tariffing, resource availability or measurement of other providers requires a close cooperation between operators, which is not a trivial task.

6.2.5 Problems

Besides the QoS requirements described in the previous sections, there are additional problems. The most significant of them is the high complexity and the inter/intra-domain operation. The implementation of a QoS model is only a small overhead but because of the

inherent preferences the implementation might not be easily portable. The heterogeneity of the Internet with its different scales, SLAs, applications, and technologies makes it even more complex to apply a single network strategy. But there are some other problems. One relevant problem is related to current technology solutions. IPv4 could not, at its time, foresee the need for guaranteed quality. [23]/,specifies the bit-vector field *Type of Service* (ToS), but this is a field rarely used in IPv4 [10, 21]. It should be noticed that the ToS field is in fact the field used for the DiffServ model. But, this field is related to the packet and not to the 5-tuple (source address, port; destination address, port; protocol number) flow. Consequently, flow-recognition is hard to implement in IPv4. Another problem is the lack of a control plane architecture, which makes the introduction of protocols such as RSVP or its extensions (RSVP-TE) unavoidable. Also, sometimes the applications are not able to determine the QoS parameters or requirements. A completeness of the realization in every level of abstraction is absolutely necessary. The quality cannot be guaranteed if there is one entity that does not support QoS within the end-to-end path. The additional hardware and software, like admission control and resource reservation in boundary routers, is also required, and it is a costly problem.

6.2.6 Use Cases

QoS is already implemented as a production service in some research-oriented networks like Géant [12]. However, there is no use or implementation outside the research networks because of the high complexity involved, specially because of the lack of a control plane.

ATM [10, 21] supports different QoS services, such as Available Bit Rate (ABR), Variable Bit Rate (VBR), Unspecified Bit Rate (UBR), Guaranteed Frame Rate (GFR) and Constant Bit Rate (CBR).

The DFN project "Quasar" [12] is related to the evaluation of QoS, architectures, performance analysis, and IP-QoS accounting. The main aim of the project is end-to-end IP-QoS and QoS over several domains using different QoS technologies.

To provide end-to-end quality, there is another technique currently used by major providers, namely, OP, which is presented next.

6.3 Over-provisioning

6.3.1 Definition

OP is current practice in large IP backbone networks [3]. It is an alternative approach to QoS [5]. It is described as a process to determine the amount of bandwidth needed on each link to support a pre-defined level of performance [4]. Another possible definition is that over-provisioning is a traffic-engineering technique based on capacity planning to make available from sufficient bandwidth around communication also to peak load times to place surely. Fig. 6.5 illustrates this over-designing. *Traffic management* is the task that

maps traffic flow in a existing physical topology [10]. *Capacity planning* is the process of predicting load on the long-run, so that there is always a reasonable percentage of available bandwidth, even in peak rate hours [14]. Capacity planning is the key technology for OP.

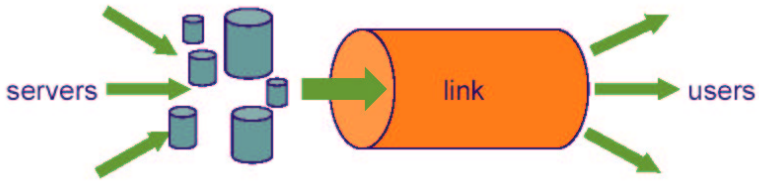

Abbildung 6.5: Over-provisioning [22]

6.3.2 Goals

OP has the main goal of providing enough bandwidth in peak load times, so that a pre-determined performance is met. Therefore, the network and its components will be over-designed as shown on the left-hand chart of Fig. 6.6. The right-hand chart represents congestion, which is used as a signal for expansion in case the capacity gets limited. The strategy is not applicable to ensure guaranteed quality to a congestion sensitive application. OP can only create conditions for quality guarantees, but not guarantee

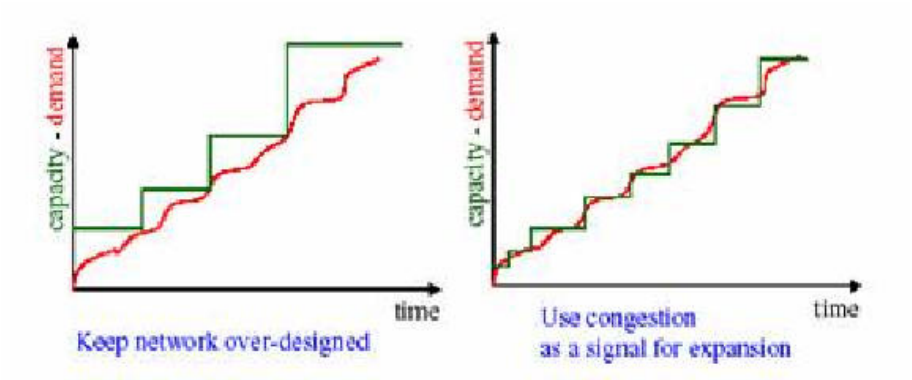

Abbildung 6.6: The principe of OP [9]

quality in itself. An advantage of OP is the design related to the bought bandwidth. In other words, it is possible for users to use more bandwidth for the same money because there is no traffic restriction or traffic control.

6.3.3 Favorable Arguments

Besides the requirements and problems that OP brings, there are several arguments in favor of this technique. One argument is its ability to enhance the network, making it faster and more reliable. The backbone traffic volume is foreseen according to previous

and careful planning. An additional advantage for operators and providers is that the costs are simply related to equipment and network installation and upgrade, thus avoiding management costs. Consequently, costs are derived from a simpler process. OP allows a better management and elimination of error sources [12].

6.3.4 Requirements and Problems

The theoretical principles of OP are simple, but practice is another history. The over-dimensioning and offered increased performance is shared among all users in the network. There is no service differentiation. Every service will be offered, in principle, the same level of quality, which can become inefficient [4] if users only produce traffic like e-mails or WEB services, which do not need have high quality requirements.

Another problem is that the traffic volume (per user) in the network has to be known to quantify an adequate capacity of the network [14]. The provider has to characterize the behavior of its users, e.g., with user-visiting models or user-behavior graphs [14]. Adding to this, empirical evidence is required. To over-provision, load figures are needed on the basis of seconds, or even less [5].

Another drawback that OP attains is its global cost. Over-designing in the form of capacity and achievement in every router of an entire network backbone could be very expensive. The cost of OP is a function of the bandwidth on every link, and every component on the network, e.g., the memory and CPU of routers. OP is an economical mechanism. The change of requirements on the network will be as follows a completely new concept process and capacity planning.

OP cannot guarantee any quality to a service and provides no traffic control.. Since there is no service differentiation, no classification and no prioritization, no resource guarantees could be given. So, there are only a few cases where OP by itself can be used to provide end-to-end quality. Also, there is no natural limit for the use of bandwidth and resources. Consequently, OP does not offer support for real-time services.

The larger share of its cost comes from the re-design of network components, after a change of the initial quality requirements. And, this is not simply related to network capacity. Other network devices involved will require updates also. The re-design of the performance and resources of each component of the network (e.g. routers) is in fact the major cost contribution.

A solution to the missing service differentiation is the combination of OP and *priorities* [3]. Following such approach, the higher capacity or performance provided can be prioritized among services with different priorities.

There are some capacity planning specific problems. Fig. 6.7 shows more than one problem that capacity planning gives rise to.

In Fig. 6.7, the x-coordinates refer to time units and the y-coordinates refers to the extent of utilization. The plot describes the burstiness of Internet traffic, and it shows first that the traffic is not evenly distributed and second, that the time between accesses has a

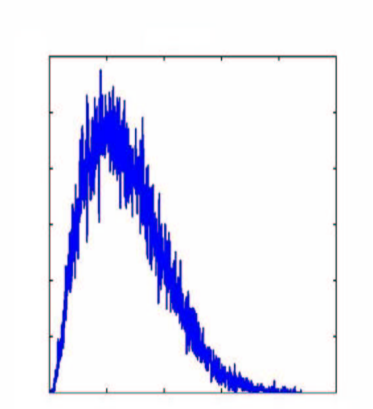

Abbildung 6.7: One picture for more than one problem [4]

high variance [14, 3]. For example, weekend traffic is much lower than the week traffic. Similarly, the traffic at 8am is lower than at 10am. Another problem which is illustrated is the *heavy-tail* distribution of Internet traffic [14, 3]. [26] found that the object-size distribution of Internet traffic mostly shows the heavy-tail property; the distribution decays more slowly than an exponential distribution and has high squared coefficients of variation and very small medians. A heavy-tail distribution describes that small objects are not only more numerous but also that they are requested more often than large objects. It means that the Internet holds more *short-lived* flows (mice) than *long-lived* flows (elephants), given that the majority of Internet traffic is related to applications such as e-mail [15]. Demand for long flows, e.g., due to video streaming, are not so frequent but have to be considered too, given that mice are responsible for 20% of Internet traffic, but elephants are responsible for 80% [14, 15]. This heavy-tail distribution is very important for caching. Capacity planning must be exact, since range and resources are expensive when considering an entire backbone.

Summing up, OP gives rise to high initial costs but low operating costs. For capacity planning five steps are required. The first step is to understand the system and its software and hardware. In the second step the load has to be specified. The last three steps are modelling of load, modelling of performance, and the analysis of costs and performance. The most important problem is change. If the load characteristic changes, the complete process has to restart. The measurement of load in complex systems is very difficult. The load of future applications is hard to predict. Therefore, to predict *adequate capacity* is a hard process, based on modelling and on SLAs [14].

6.3.5 Use Cases

As mentioned, OP is used in large IP backbone networks. A study [6] shows that only 2.6 percent of the total investment is used for OP in backbones, but more than 6 times higher in private networks. OP could be used in connections between ISPs or backbones and local networks of customers because of the easy approximate traffic [5]. One example of the usage of OP is the SPRINT IP [4, 12] network. They use it in their backbone

and claim to require only 5% to 15% over-design in order to achieve a delay lower than 4ms [4]. This example shows that is it possible that no high over-design is required. But in other examples [5], a claim of more than 100% over-design is mentioned. So an over-dimensioning of an university network of about 100% might be required to cater for 99% of the peaks [5].

6.4 Comparison of OP and QoS

Throughout the past sections the functionality, cost, advantages and disadvantages of QoS and OP have been described. This section summarizes and provides a brief comparison on the most important features of both. QoS is complex and requires the ability to differentiate services. Because of its complexity and of current technology evolution, the commercial world is reluctant in providing support for it. Its implementation in production environments is possible with little effort, as proved by GÉANT. QoS does not create resources but allows controlled resource assignment. However, the flip-side of QoS is the additional hardware and software in the form of new protocols, e.g, the implementation of RSVP and DiffServ. An additional problem is, from the author's perspective, the required concepts and the SLAs establishment between providers. As for OP, it cannot hold any quality or resource guarantee. Therefore, it is an *economical mechanism* [10]. The equal treatment of all services is inefficient. The knowledge of traffic volume in the network and every user with its applications is the problem of the capacity planning. So, in the author´s opinion, the greatest disadvantage of OP are the costs associated to the required continuous upgrade. The bottom-line is: OP is a very practicable technique, but offers no guarantees. QoS is very tricky still, but allows to give finer-grained quality guarantees.

The next Section give an overview about different pricing schemes and their impact to the two techniques QoS and OP. With these schemes it is possible to influence the user behavior and to adjust some disadvantages of the techniques described.

6.5 Pricing schemes

6.5.1 Why do We Need Pricing

E-Commerce has become more and more widespread. ISPs start to offer some form of differentiated services. As the Internet moves away from pure "best-effort" services to a more differentiated service network, new schemes for pricing and SLAs will necessary evolve [8]. Why do we need pricing? Providing free network services leads to the *tragedy of the commons* [6]. An example of such economical situation is a scenario where villagers share a common grazing ground (*commons*) for their cattle. With unlimited and free access to the commons, each villager will allow cattle to graze as much as possible, without considering the cost imposed on other villagers (who have less grass available for their cattle as a result). Therefore, the commons will be overgrazed. A second reason is that providers want to cover their costs and make profit. The use of pricing could influence

the user behavior so that the network could be used more efficient and economic. The last point is the social criterion, meaning that Internet and services should not become a luxury commodity, only accessible for richer people.

6.5.2 Definition

Pricing (also known as *Tariffing*) is a (regular) specification of pieces for goods in particular network resources and services [10]. A pricing scheme is, therefore a method of translating the costs and required resources for an offered service to an amount of price. Pricing is affected by the market structure of the network, the cost recovery strategy, network demand, network load, cost of network equipment, ToS, capacity expansions or regulatory environment and has played a prominent role in commercializing the communication services and technologies.

Pricing schemes can be split into *dynamic vs. static*, *flat vs. parameter-based*, and *best-effort vs. QoS-based* [9]. Issues and notions related to pricing are discussed next.

6.5.3 Pricing Schemes and Their Relation to QoS and Economy

A number of pricing schemes have been proposed. The goal of this section is to review these schemes using the following criteria: provision of individual QoS guarantees, impact of social fairness, and degree of network and economic efficiency. The provision of individual QoS guarantees is related to individual user guarantees, as opposed to raising the service level for a typical user [6]. The criterion network efficiency evaluates the expected utilization level in networks. While high utilization levels are desirable for the network, they are not necessarily desirable for the user. Low utilization levels imply availability of service. A highly utilized network, on the other hand, may have to deny service to some customers. The economic efficiency criterion indicates the focus of the chosen scheme overall utility level (or user benefit) of the user community. The social fairness criterion [6] indicates whether the pricing scheme prevents some users from accessing the network during congestion periods purely because of their inability to pay. Note that *fairness* is a vague concept and that there are many definitions. In proportional fairness pricing, for example, a resource allocation is considered fair if it is in proportion to the charge. But this is just one interpretation. Fairness is, in general, hard to define. Should it be defined at the packet or at the call level? How can a large file transfer be handled in a manner that is fair to all users? This section presents briefly several pricing that aim at providing answers to the problems mentioned. In particular, the section will focus on the following schemes:

- Flat pricing;
- Expected Capacity pricing;
- Responsive pricing;

- Effective Bandwidth pricing;
- Proportional Fairness pricing;
- Priority pricing;
- Paris-Metro pricing;
- Smart-market pricing;
- Edge-Pricing.

Flat Pricing

Under a flat pricing scheme [6, 9, 17] the user is charged a fixed fee per time unit (e.g., month), irrespective of usage. This pricing scheme has several desirable advantages. First and foremost, it is simple and convenient. Flat pricing makes no assumptions about the underlying network technology that is already deployed. No measurements are required for billing and accounting. This also leads to social fairness in the sense that no distinction is made between poor and rich users. This scheme does not allow the network to influence the user's transmission behavior and have no incentive to alter their transmission behavior to support the network operation. Flat pricing is therefore unsuitable for congestion control or traffic management. Flat pricing also does not explicitly support individual QoS guarantees to the user. Providers are even discouraged from provisioning QoS, since they cannot recover the associated costs. [6]/,suggests that QoS could be adequately provided by combining a flat pricing scheme with resource OP. Flat pricing does not improve the economic efficiency of the network. Individual user's utility levels are not taken into account.

Paris-Metro-Pricing (PMP)

A PMP-based network [6, 7, 9] is a set of logical networks or channels. The total bandwidth capacity is divided into such networks. Each channel is priced differently. Users choose one of these logical networks for the transmission of their traffic, and this implicitly defines the service and the quality level, which is the same for users sharing the same channel.Network operators set the prices for each logical subnetwork. Users make a selection based on the expected network congestion and their budget. Assuming that prices are kept stable over significant periods of time and that users are price sensitive, higher priced networks will experience lower utilization and hence be able to provide a higher service level.

The main advantage of the scheme is its simplicity. Since each logical network implements flat pricing, the scheme is compliant with existing technologies and similar to the previous presented scheme. Pricing indeed becomes a control for traffic management under PMP. Instead of defining and providing different QoS categories, the network operator provides a set of technically identical networks that are distinguished by their access price. PMP traffic management encourages users to separate themselves into different classes. While the perceived service level in each subnetwork varies with price, PMP still does not

support individual QoS guarantees. Each network still operates on a best-effort basis. A relatively better service level is expected for higher priced networks (these are, in effect, over-provisioned). Economic efficiency in PMP networks improves. The utility level of users is bound to increase by the choice of the different subnetworks. PMP still does not use pricing as a means to optimize the overall utility of the user community. A technical disadvantage of PMP networks is their potential for instability (e.g., caused by fractal traffic). During periods of congestion, price-insensitive users may choose a higher-priced network in expectation of receiving better service. This may lead to congestion in the higher priced networks and thus cause instability. PMP-pricing is currently seen as an initial proposal, far from deployment.

Priority Pricing

Any priority-based [6] network must be accompanied by a pricing scheme to deter each user from using the highest priority class as a default. Users are forced to indicate the value of their traffic by selecting a priority level. During periods of congestion the network can then carry the traffic by the indicated level. Priority pricing requires a priority field in every packet header. Such a field is already provided in IP. Measurements are now required for billing and accounting to keep track of the priority level of each transmitted packet for each user. Priority pricing allows the users to send appropriate signals to the network to facilitate traffic management, even at short time frames. During periods of congestion, traffic is transmitted by priority level. Low-priority traffic is delayed or even dropped. Individual QoS guarantees still cannot be given. In contrast to the previous pricing schemes, priority pricing assumes that resources will be scarce or running at high levels of utilization. Transmission capacities of tens of megabits or more will provide sufficient bandwidth to enable bandwidth-intensive applications to be run by an increasing community of users. Since prices are an indicator for usage, priority pricing raises the economic efficiency of the network. Any dropped packets will have a lower priority level and thus a lower value to the user. Studies find that a network can always set prices such that the overall user satisfaction is higher under priority pricing than under flat pricing. A user's ability to pay may in this case not guarantee transmission by the network. With a large number of priority levels and/or high prices for high priority levels, traffic from poor users (in a relative sense) may be starved.

Smart-Market Pricing

A smart-market pricing [6, 8, 10] scheme focuses on the issues of capacity expansion and the social cost imposed on other users. In addition to a fixed charge to cover the connection costs and a (possibly small) charge per packet to cover the incremental cost of sending a packet, MacKie-Mason and Varian introduce a usage charge when the network is congested. This charge is determined through an auction. The user associates a price with each packet, carried in the packet's header, communicating the user's willingness to pay for transmission. Typically, such a price would be derived from default values, with the option to the user of overriding the default value for special transmissions. The network collects and sorts all the bids. It then determines a threshold value and transmits all the

packets whose bid exceeds the threshold value. The threshold value is determined by the capacity of the network and represents network's capacity and represents the marginal cost of congestion. Each transmitted packet is then charged this marginal congestion cost, not the value of the bid. An auctioning mechanism leads to noncompliance with existing technologies. Smart-market pricing requires significant technical changes to protocols and networking hardware. For each transmitted packet the user's billing records need to be updated. This also implies considerable overhead, even if the auctioning mechanism is only applied during congested periods. The scheme also does not provide the users with service guarantees or even a guarantee of transmission. Like priority pricing, it only ensures that the packets are transmitted according to their relative priority, determined by the bid. A potential problem of social distribution is that prices may deter "poor" users from using the service, thus creating a case for government regulation. On the other hand, smart-market pricing encourages both network and economic efficiency. Using an auctioning mechanism, the network also encourages economic efficiency. All transmitted packets have a positive benefit to the user. The bid provides the benefit to the user of having the packet transmitted, thus representing its social value. Packets with high bids might be routed over shorter paths, whereas packets with low bids may be routed through longer paths.

Edge-Pricing

Edge-Pricing [6, 8] provides a conceptual contribution to shift the focus to locally computed charges based on simple expected values of congestion and route. An example of such an expected congestion cost is time-of-day charges. Basing the charge on the expected distance between source and destination allows the charge to be applied at the edge of the network. Prices may be communicated from any place in the network. Such a pricing scheme is much simpler and facilitates receiver payments. The value of multicasting and receiver-charging is particularly stressed. Multicasting increases network efficiency and should therefore be encouraged. Receiver-charging is important for future multimedia applications, such as video-on-demand. Even though edge pricing is predominantly a conceptual contribution, we view it as compatible with ATM, or RSVP. Since edge pricing is supposed to be applied over short to medium time frames, it also influences the user's transmission decisions. In this way and by association with, for instance, ATM/RSVP, Edge-Pricing can support traffic management. Traffic measurements for billing and accounting may still be required, but these would be at most locally applied at the edge of the network. Individual users could even be given QoS guarantees. Edge-Pricing promotes network efficiency by encouraging users to use multicast-based services. Economic efficiency is particularly de-emphasized in Edge-Pricing.

Expected Capacity Pricing

This scheme [6] centers around the user specifying the required expected capacity. The user is then charged according to the expected capacity that the network provisions, not on actual usage, based on a long-term contract with the network. This indicates the capacity he or she expects to use when the network is congested. Expected-capacity pricing

again seems to be compatible with ATM, RSVP or *Multiprotocol Label Switching (MPLS)*. This implies support for congestion control or traffic management by encouraging users to determine the service level and then charging accordingly. Individual QoS guarantees can therefore be given. One of the main advantages of expected-capacity pricing is that charging is not related to actual traffic volume, but rather to expected traffic volume. Measurements are not required, saving significant overhead for the network. Combined with the concept of Edge-Pricing, a user's bill can easily be computed at the edge. However, even though measurements are not required for billing. The scheme also handles inter-provider traffic. Service providers are assumed to enter into contractual agreements among themselves, thus establishing a user-network relationship that could be handled. The evaluation of this scheme in terms of economics and network efficiency by [6] therefore does not differ from edge pricing. In his view, this scheme is socially fair in the sense that even relatively poor users can negotiate contracts with the network. Note that expected-capacity pricing should be distinguished from resource-based pricing, where the user is charged according to the measured resource usage. Such schemes are typically computationally more demanding.

Responsive Pricing

The concept of responsive pricing [6] describes a dynamic price-setting strategy by the network, illustrating how the network can exploit the adaptive nature of users to increase economic and network efficiency. Price is emphasized as an alternative (and even improved as compared to traditional feedback mechanisms) means for congestion control to ensure proper network operation, and in particular to guarantee different service levels. Similar to smart-market pricing, the charging mechanism only comes into operation during periods of congestion. Responsive pricing is based on the realization that users are adaptive and respond to price signals. In case of high network utilization, resources are stressed and the network increases the prices for the resources. Adaptive users then, by definition, reduce the traffic offered to the network. Similarly, in case of low network utilization, the network decreases the price and the community of adaptive users increases their offered traffic. In this way, adaptive users do not just increase the network efficiency, but also economic efficiency. Adaptive users fall into two classes: *elastic* users and *inelastic* users. Elastic users cannot tolerate losses but are willing to allow transmission delay. Inelastic users require strict delay guarantees, but can tolerate some degree of losses. Both classes of adaptive users value the service level they receive from the network. Both types of users thus respond to the price signals given from the network by altering the amount of traffic they transmit. Responsive pricing was designed for ATM ABR services. [6]classifies it as compatible with existing technologies and as being socially fair.

Effective Bandwidth Pricing

The Effective Bandwidth pricing scheme [6] is designed to induce the user to declare the true values for the mean and the peak cell rates of general traffic sources during *Connection Admission Control (CAC)*. The user is charged according to a linear function tangent to the effective bandwidth curve of the source. The pricing pricing scheme ensures that the

network can deduce the anticipated load generated by a user. The scheme allows the network to infer the actual (parameterized) function from the user's declaration. The user is assumed to act rationally in the sense of wanting to minimize the economic cost of the connection. Note that under effective bandwidth-based CAC, QoS guarantees can be given to individual users, and statistical multiplexing effects can be exploited to obtain high levels of utilization[6]/,considers effective bandwidth pricing socially fair. The user is only required to provide the expected value of the traffic stream. Any additional information about the joint distribution of the mean rate and the peak rate is not used in the scheme. A disadvantage of this scheme is that the functional form of the effective bandwidth is assumed to be known in advance. This pricing scheme can also be extended to allow for time-varying prices and to deter the user from splitting the source traffic [6].

Proportional Fairness Pricing

The concept of Proportional Fairness Pricing (PFP) was motivated by the desire to incorporate the notion of fairness into the allocation of network resources. In this scheme, a resource allocation is fair if it is in proportion to the users willingness to pay. An allocation of resources guarantees economic efficiency, since users utilities are maximized. According to the definition of social fairness presented in [6] , this scheme is also fair. Every user who is willing to pay is allocated some bandwidth. This guarantees access even for "poor" users. Similarly, network efficiency is high since the network is always willing/able to allocate all its resources. Congestion in this scheme is avoided by allocating resources to the users. During periods of high demand, each user would get a proportionally smaller amount of bandwidth, for example, and thus users need to throttle their transmission at the edge of the network. Note that billing and measurements are not required, since each user is charged according to the indicated willingness to pay. The network only needs to keep track of the willingness to pay and allocate resources accordingly.

6.5.4 Summary

In this section a number of pricing schemes for broadband multiservice networks have been briefly described. Most of the schemes attempt to use pricing as a means to control congestion by exploiting the price sensitivity of users. In some cases, this is done by prioritizing user traffic (PMP, Priority Pricing, and Smart-Market-Pricing). Other schemes implicitly support congestion control and traffic management by associating pricing with CAC (Edge-Pricing, Expected-Capacity-Pricing, or Effective-Bandwidth-Pricing). Note that most schemes are designed for pricing best-effort services, and therefore do not support individual QoS guarantees. Fig. 6.8 gives an overview of the discussed pricing schemes.

Priority schemes allow users to purchase a relative advantage over other users, and so users paying for high priority enjoy a better service. In case of badly designed business models, all users might be able to purchase high-priority service. This underlines the importance of well designed prices to make priority schemes effective. QoS guarantees for individual users can typically be given in pricing schemes that are related to CAC.

	Flat	PMP	Priority	Smart market	Edge/exp.cap	Resp.	Efficiency bandwidth	PFP
Compliance	IP	IP, VN	IP		ATM, RSVP	ATM, VN	ATM	ATM, IP
Billing Measures	No	No	Yes	Yes	Yes (local)	Yes	Yes (local)	No
Cong. control traffic management	No	Yes (rel.)	Yes (rel.)	Yes (rel.)	Yes (CAC)	Yes	Yes (CAC)	Yes
Individual QoS	No	No	No	No	Yes	Part	Yes	No
Network efficiency	Low	Var.	High	High	High	High	High	High
Economic efficiency	Low	Var.	High	High	Var.	High	High	High
Social fairness	Yes	Yes	No	No	Yes	Yes	Yes	Yes
Time frame	Long	Long	Short	Short	Medium/long	Short	Short	Short

Table 1. *Summary of pricing scheme features.*

Abbildung 6.8: Summary of pricing schemes [6]

Flat pricing and PMP argue that due to technological advances, congestion can always be avoided by OP. Other pricing schemes take a different point of view. Price changes are used to induce users to alter their transmission behavior. Pricing schemes designed to handle short-term congestion at least approach economic efficiency and also allow for higher network efficiency. In [6], the criterion of social fairness is satisfied by all pricing schemes except priority pricing and smart-market pricing.

6.6 Conclusions

This document dealed with two base techniques, QoS and OP, used by operators to provide quality in IP networks. QoS is the administration of existing resources an the efficient distribution between services. It was shown that QoS is quite complex and requires a lot of methods and management, with the major advantage of being able to guarantee quality to services. As a result, the operator or provider depends on other providers because of the management of resource reservation or the negotiation of QoS parameters in end-to-end connections. OP is a viable alternative, which cannot guarantee quality and has higher costs when upgrades are required. The final part of this document dealed with pricing, and how it can influence users and networks: users, in their economical behavior and networks, in their efficiency. Pricing achieves its goals by motivating the customers to buy prioritized services. The majority of pricing schemes target best-effort networks. But there are also some that can be applied to environments where QoS guarantees are supported. An interesting fact is that pricing can improve OP or QoS. For instance, the missing service differentiation in OP can be repaired with Priority-Pricing. For QoS, the complexity inherent to cooperation between providers can be repaired with Edge-Pricing.

Acknowledgements

The author would like to thank Dr. Rute Sofia and Professor Dr. Burkhard Stiller for technical help, proof reading, and support.

Literaturverzeichnis

[1] S. Casner, "A fine-grained view of high performance networking", NANOG 22, May 1997 http://www.nanog.org/mtg-0105/ppt/casner/index.html

[2] P. Ferguson, G. Huston, "Quality of Service: Delivering QoS in the Internet and in the Corporate Network", Wiley Computer Books, New York, NY. 1998

[3] T. Telkamp, "Traffic Characteristics and Network Planing", NANOG 26, October 2002, http://www.nanog.org

[4] C. Fraleigh, F. Toubagi, C. Diot, "Provisioning IP Backbone Networks to Support Latency Sensitive Traffic", Infocom 2003, San Francisco, USA, April 2003, http://www.ieee-infocom.org/2003/papers/10_01.PDF

[5] R. van de Meent, A. Pras, M. Mendges, H. van den Berg, L. Nieuwenhuis, "Traffic Measurements for Link Dimensioning: a Case Study", University of Twente, Netherlands, August 2003, http://www.ub.utwente.nl/webdocs/ctit/1/000000d2.pdf

[6] M. Falkner, M. Devetsikiotis, I. Lambadaris, "An Overview of Pricing Concepts for Broadband IP Networks", IEEE Communications Surveys, 2000, http://www.comsoc.org/livepubs/surveys/public/2q00issue/falkner.html

[7] R. Kaleelezhicathu, "History of Internet Pricing", HUT, Finland, 2003, http://www.netlab.hut.fi/opetus/s38042/k03/topics/HistoryIntPrice.pdf

[8] P. Reichl, S. Leinen, B. Stiller, "A Practical Review of Pricing and Cost Recovery for Internet Services", 2nd Internet Economics Workshop Berlin (IEW 99), Berlin, Germany, May 28-29, 1999, http://www.tik.ee.ethz.ch/~cati/paper/netnomics00.pdf

[9] Costas Courcoubetis, "Pricing the Internet", Athens University of Economics and Business and ICS-FORTH

[10] Burkhard Stiller, Vorlesung "Protokolle für Multimedia Kommunikation - PMMK", University of German Armed Forces, HT2003

[11] Chen-Nee Chuah, "Providing End-to-end QoS for Real- Time Applications over the Internet", Networking Seminar, April 2000, ICEBERG Research Group Department of Electrical Engineering and Computer Science University of California, Berkeley

[12] Rudolf Roth, "Quality of Service in IP Netzen", 3. BZVD Workshop Dresden, 12. März 2001, GMD FOKUS

[13] Andreas Fischer, Seminar "Quality of Service in Wireless Local Area Networks", UniBwM, WT2004

[14] Rachid El Abdouni Khayari, Vorlesung "capacity planing of web-based systems", UniBwM, WT2004

[15] Boris Jakubaschk, Roland Moos,
Website: `http://www.i-m.de/home/datennetze/index.htm`

[16] Liang Guo, Ibrahim Matta, "The War Between Mice and Elephants", 2001, `http://citeseer.ist.psu.edu/guo01war.html`

[17] Zhen Liu, Laura Wynter, Cathy Xia, "Pricing and QoS of Information Services in a Competitive Market", June 9-12, 2003, San Diego, California, USA

[18] Lars Burgstahler, "Overprovisioning und QoS - simulativer Vergleich", 7. April 2003, Institut für Kommunicationsnetze und Rechensysteme, Universität Stuttgard

[19] Austrian Research Center, Website: `http://www.arcs.ac.at/IT/ITS/MF/mobile.htm`

[20] Glynn Rogers, "Programmable Networks: What is P1520 Going to Do for You", CSIRO Telecommunications and Industrial Physics, Marsfield, NSW

[21] Tibor Dekany, Seminar "Cutting Edge IT-Engineering 2003", "Technische Lösungen für Quality of Service", Universität Zürich, Institut für Informatik, 20.01.2003

[22] J. Roberts, "IP traffic and QoS control", France Telecom R&D, December 2002

[23] The Internet Engineering Task Force (IETF), RFC791 Specification, September 1981, `http://www.ietf.org/`

[24] Richard R. Forberg, "The changing costs of QoS", Burlington, quarry technologies, Massachusetts, 2000, `http://www.quarrytech.com/news/pdf/global_telephony.pdf`

[25] R. Guérin, L.Li, S. Nades, P. Pan, V. Peris, "The Cost of QoS Support in Edge Devices - An Experimental Study"

[26] Rachid El Abdouni Khayari, Ramin Sadre, Boudewijn R. Haverkort, "A Class-Based Least-Recently Used Caching Algorithm for WWW Proxies", Laboratory for Performance Evaluation and Distributed Systems, Department of Computer Science, RWTH Aachen, D-52056 Aachen, July 12, 2002

Kapitel 7

Reputation and Trust - The Key for Business Transaction

Björn Hensel

In der heutigen Zeit stellt sich das Internet als größte Handelsplattform dar, auf der jeder, der einen Zugang dazu besitzt, fast alles bekommen kann, was er möchte. Des Weiteren bietet es die Möglichkeit zum gegenseitigen Beziehungsaufbau von verschiedenen Entitäten, um den virtuellen Einflussbereich auszubauen. Die Gründe hierfür liegen eindeutig in der Globalität und der Unabhängigkeit gegenüber räumlichen und zeitlichen Beschränkungen. Ein derzeitiger Trend ist der Zusammenschluss zu virtuellen Gemeinschaften mit gleichen Interessen zwecks des Erfahrungsaustausches oder auch mit ökonomischen Absichten. Hierbei haben hauptsächlich weniger etablierte Unternehmen die Chance, ihre eigenen Leistungen zu erweitern und Standortprobleme zu überwinden [1]. Allerdings sind diese Voraussetzungen kein Garant für ein erfolgreiches Bestehen in der digitalen Welt, da es andere Grenzen gibt, die ein Konzept schon in der Anfangsphase zum Scheitern bringen können. Diese Grenze ist der Mensch - ob Kunde oder Partner. Dabei spielt es keine Rolle, in welchem Businessbereich man sich befindet. Selbst in der realen Welt ist die Frage des Vertrauens allgegenwärtig. Aber das Hauptaugenmerk richtet sich in Anbetracht des Seminarthemas natürlich auf das Internet. Diese Ausarbeitung wird dabei aber keine explizite Trennung zwischen Business-to-Business und Business-to-Customer vornehmen, sondern einen allgemeinen Ausblick über Vertrauen im Internet geben. Die Betrachtung des Aspektes der Vertrauensbildung, sowie Wege und Mittel sind Gegenstand dieser Ausarbeitung.

Inhaltsverzeichnis

7.1 Vertrauen und Reputation - eine Definition

Als erstes wird dieser Abschnitt die Begriffe Reputation und Vertrauen so definieren, wie sie für den Kontext dieser Arbeit zu verstehen sind. Des Weiteren gibt er eine kurze Erklärung zum Principal-Agent-Problem und wie es sich im Internet auswirkt. Wie bereits Eingangs erwähnt, ist es eine allgemeine Betrachtung; die hier gemachten Aussagen können gleichermaßen auf B2B angewandt werden, wie auch auf B2C, nur dass die beteiligten Parteien eine andere Beziehung zu einander haben.

7.1.1 Vertrauen

Es gibt viele Definitionen für Vertrauen. Wobei es zwei Relationsgruppen gibt, die man betrachten muss. Zum einen geht man bei Vertrauen immer von einer Mensch-Mensch Beziehung aus. Eine erste allgemeine Begriffsbestimmung kann man hierzu der *Wikipedia.org* Seite entnehmen [2]. Sie beschreibt das Vertrauen als "subjektive Überzeugung (oder auch Glaube) von der Richtigkeit, bzw. Wahrheit von Handlungen und Einsichten anderer oder sich selbst. Des Weiteren zählt auch die Überzeugung zur Möglichkeit von Handlungen und die Fähigkeit zu Handlungen zum Vertrauen dazu." Wobei letzteres im allgemeinen auch als Zutrauen bezeichnet wird. Allerdings wird diese Unterscheidung bei dieser Betrachtung nicht vorgenommen und beides als Vertrauen deklariert.

Grundsätzlich handelt es sich hierbei also um eine Relation zwischen zwei Parteien die auf Interaktionen basiert, wobei die eine als Vertrauensgeber und die andere als Vertrauensnehmer bezeichnet wird. Die Voraussetzung für das Vertrauen ist das Vorliegen von Informationen über den Partner, die dem Vertrauensnehmer entweder bekannt oder zumindest zugänglich sind. Hierauf ruht das entscheidende Kriterium für die Stärke des Vertrauens. Gleichzeitig ist es aber auch das Hauptproblem, welches in der virtuellen Welt die Vertrauensbildung erschwert.

Eine weitere Definition, die sich sehr gut auf das E-Commerce beziehen lässt, liefert *Ratnasingham* [3]: "Trust is the calculation of the likelihood of future cooperation and is a defining feature of virtual cooperation. As trust declines, people are increasingly unwilling to take risks and demand greater protections against the probability of betrayal. Trust is only relevant in situations where one must enter into risks and where one cannot control what is to happen in advance." Hierbei weist er auch auf die Konsequenzen des Misstrauens hin, welche allerdings schon als Schlussfolgerung aus der allgemeinen Definition herleitbar sind. Auch wenn diese Beschreibung auf Virtuelle Unternehmen (*Abschnitt 7.2.1*) anspielt, kann sie auch bei jeder anderen Entität angewandt werden, da sie die Folgen für den Vertrauensnehmer erklärt.

Letztendlich liefern *Mayer* und *Schoorman* [4] eine ähnliche Beschreibung, jedoch mit einem entgegengesetzten Gesichtspunkt. "Trust can be defined as the willingness of a party to be vulnerable to the actions of another party, based on the expectation that the other will perform a particular action important to the trustor." Diese Definition setzt Vertrauen mit freiwilliger Verwundbarkeit gleich. Denn mit dem Zugeständnis von Vertrauen gibt man einen Teil seiner Sicherheit einem anderen Preis.

Die zweite Relationsgruppe beschreibt Vertrauen zwischen Mensch und Maschine. Diese spielt in der virtuellen Welt eine wichtige Rolle, da für die Arbeit im Internet immer eine Maschine als Zugangsbasis dient. Auch der Aktionspartner ist in erster Linie ein Rechner, der erst im Nachhinein von einem Menschen gesteuert wird. Allerdings gibt es viele Online-Shops, bei denen ein Kauf fast vollautomatisch abgewickelt wird, da die Möglichkeit besteht auch nachts einzukaufen, wenn kein Mitarbeiter mehr vor Ort ist. Dann werden alle Transaktionen, zumindest die Kaufbestätigung und Weiterleitung, durch einen Serverdienst geleistet.

Auch wenn der Verkäufer immer eine Organisation oder Person ist, findet die Interaktion nicht mehr zwischen zwei Menschen statt. Daher müssen beide Parteien auf die Funktion der Maschine vertrauen. Zum einen können dem Verkäufer durch Fehler im Programm Umsatzeinbußen entstehen und zum anderen kann dem Käufer Zeit und auch Geld verloren gehen. Aber eigentlich werden diese Aspekte normalerweise eher vernachlässigt (vorwiegend von Käufern), da man von einer korrekt arbeitenden Maschine ausgeht, wenn sie im Internet für die Abwicklung von hohen Werten benutzt wird. Wobei wir wieder bei der Mensch-Mensch-Beziehung wären, da im Internet zwar immer Interaktionen zwischen Mensch und Maschine stattfinden, die Vertrauensbildung aber indirekt zu dem Menschen aufgebaut werden, der für diese Maschine verantwortlich ist. Abschließend kann man sagen, dass viele Faktoren für Vertrauen und dessen Bildung notwendig sind. Eine nähere Betrachtung erfolgt dazu im dritten Abschnitt.

7.1.2 Reputation

Ein guter und auch viel verwendeter Indikator für Vertrauen ist die Reputation. Auch hier liefert *Wikipedia.Org* eine erste Definition [5]: “Reputation ist der an eine Einzelperson oder Institution gebundene Ruf höherer Kompetenz und Qualifikation, hinsichtlich der Erbringung von gesellschaftlich relevanten Leistungen. Sie kann in der Terminologie Pierre Bourdieus als kulturelles Kapital verstanden werden, d.h. als eine soziale Ressource, welche sich für den Inhaber in andere Kapitalformen transformieren lässt; beispielsweise im Zusammenhang mit der Verteilung von Forschungsgeldern an verdiente und vielversprechende Wissenschaftler (ökonomisches Kapital).“ Allerdings ist diese Definition nicht vollständig, denn sie erwähnt nicht, für wen diese Reputation relevant ist. Auch kann der Ruf gerade auf schlechtere Kompetenz oder die Uneignung für eine entsprechende Leistung hinweisen.

Dazu hat *Charles J. Fombrum*, Direktor des Instituts für Reputation folgende Definition vorgeschlagen [6]: “A corporate reputation embodies the general estimation in which a company is held by employees, customers, suppliers, distributors, competitors, and the public.“ Nun bezieht sich diese Beschreibung zwar speziell auf Firmen, aber sie lässt sich auch auf einzelne Personen anwenden (in Hinblick auf die Reputation bei eBay [*Abschnitt 7.4.1*]), wenn man die einschätzende Gruppe dementsprechend einschränkt. Zumindest ist Reputation nach *Fombruns* Ansicht die allgemeine Einschätzung aller Gruppen, die in unmittelbarer oder mittelbarer Verbindung zum Einzuschätzenden stehen. Diese Gruppen werden auch als Teilöffentlichkeiten bezeichnet. Nunmehr lässt diese Definition eine Interpretation als Identität oder Image zu, da sie nach *Bazil* folgende Merkmale aufweisen[7]:

- **Ganzheitliche Wahrnehmung:** Kognitive, emotionale und motivierende Ansichten werden durch Wünsche, Wertungen und Erwartungen der Wahrnehmenden verstärkt. Diese Teileindrücke werden zu einem Ganzen zusammengefasst.

- **Kollektiv bzw. intersubjektiv:** Sie entstehen aus sozialen Netzwerken - siehe die erwähnten Teilöffentlichkeiten. Dabei spielt die Homogenität der Gruppe eine große Rolle bei der Annahme von Einstellungen. Wenn zuviele verschiedene Meinungen vorherrschen, ist es schwierig einen gemeinsamen Konsens zu finden.

- **Relativ:** Durch die Verschiedenheit dieser Gruppen treten jeweils unterschiedliche soziale Repräsentationen hervor, so dass theoretisch von mehreren Reputationen gesprochen werden müsste. Bei der Zusammenfassung dieser würden zwangsläufig nicht alle Einstellungen berücksichtigt werden können.

Allerdings wird Reputation durch vier spezielle Attribute von Image und Identität unterschieden. Diese wurden von *Fombrum* spezifiziert als [6]:

- Glaubwürdigkeit
- Zuverlässigkeit
- Vertrauenswürdigkeit
- Verantwortungsbewusstsein

Während Identität die Selbstansicht und Image das Ansehen nach außen darstellen, ist die Reputation ein Hinweis darauf, wie eine Organisation oder Person im Stande war, der Öffentlichkeit klar zu machen, dass sie in einem speziellen sozialen Kontext glaubwürdig, zuverlässig, vertrauenswürdig und verantwortungsbewusst ist. Allerdings kann man daraus auch eine hierarchische Struktur ableiten, wie das *Fombrum* getan hat [6].

Abbildung 7.2 zeigt, dass die Reputation aus dem Image entsteht, daher kann sie nur mittelbar erworben werden. Außerdem ist das ein zeitaufwendiger, lang andauernder Prozess, der niemals abgeschlossen sein wird. Bei konsistentem Verhalten kann sich der Vorgang zumindest auf einen halbwegs festen Zustand verdichten. Wenn eine Firma immer gute Arbeit leistet und immer einen guten Service aufweist, wird sie auch einen guten Ruf erwerben und wenigstens solange behalten, bis sie ihre Leistungen nicht mehr so vertrauensvoll erfüllt.

Auch kann Reputation nur teilweise beeinflusst werden. Allerdings ist die Nachhaltigkeit dieser Beeinflussung relativ abhängig von der Öffentlichkeit, die ja eigenständig in ihrer Meinungsbildung agiert. So kann man zwar mit Hilfe von Werbung versuchen sein Image zu verbessern, letztendlich entscheiden aber alle angesprochenen Zielgruppen selbst über deren Wirkung.

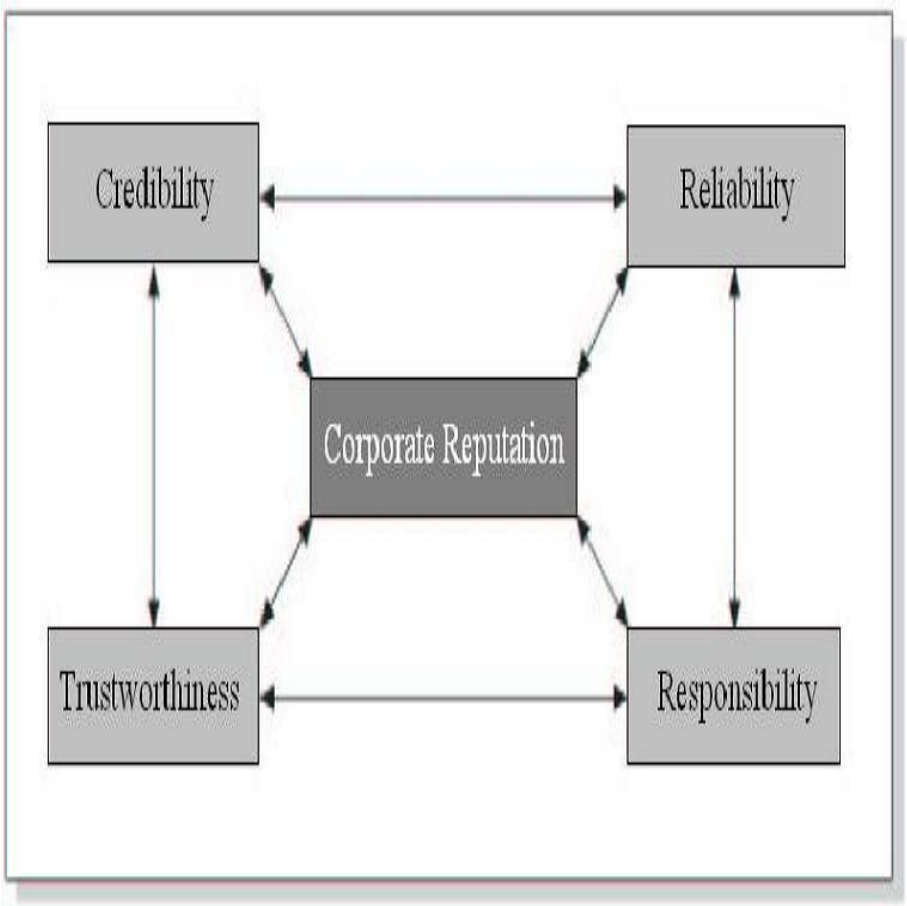

Abbildung 7.1: Die vier Attribute der Reputation

7.1.3 Das Principal-Agent-Problem

Dieses Problem tritt fast immer dann auf, wenn bei Interaktionen zwischen zwei Entitäten eine Informationsasymmetrie vorliegt. Im allgemeinen wird hierbei von einer Person oder Organisation eine Leistung gefordert. Diese wird als Agent bezeichnet, während die Partei, die die Leistung einfordert, als Principal gekennzeichnet ist. Als Beispiel einer solchen Relation könnte man eine Auftragserteilung durch eine Firma an eine andere angeben, wobei letztere der Erteilenden nur bedingt bekannt ist. Dies zeigt schon, dass bei fast allen Geschäften dieses Problem auftritt, bei denen die Akteure noch nie oder nur bedingt zueinander Kontakt hatten, wie das bei vielen im Internet geschlossenen Verträgen (produkt- oder leistungsgebunden) der Fall ist. Meistens wirkt es sich auch noch in beiden Richtungen aus, also von Käufer zu Verkäufer und umgedreht. Die Informationsasymmetrie ist hierbei als unzureichendes Wissen über den Aktionspartner zu verstehen.

Das eigentliche Principal-Agent-Problem entsteht erst daraus, dass der Principal den Agenten nur bedingt oder gar nicht kontrollieren kann, inwieweit dieser seine Aufgabe durchführt. So kann der Agent durch opportunistisches Handeln versuchen sich Vorteile zu verschaffen, seien sie informeller oder ökonomischer Natur. (Ein Beispiel wäre das absichtliche Zurückhalten von Zahlungen zur eigenen Kapitalanreicherung durch Zinsgeschäfte) Dieses Handeln kann den Principal natürlich stark schädigen, da er zum Bei-

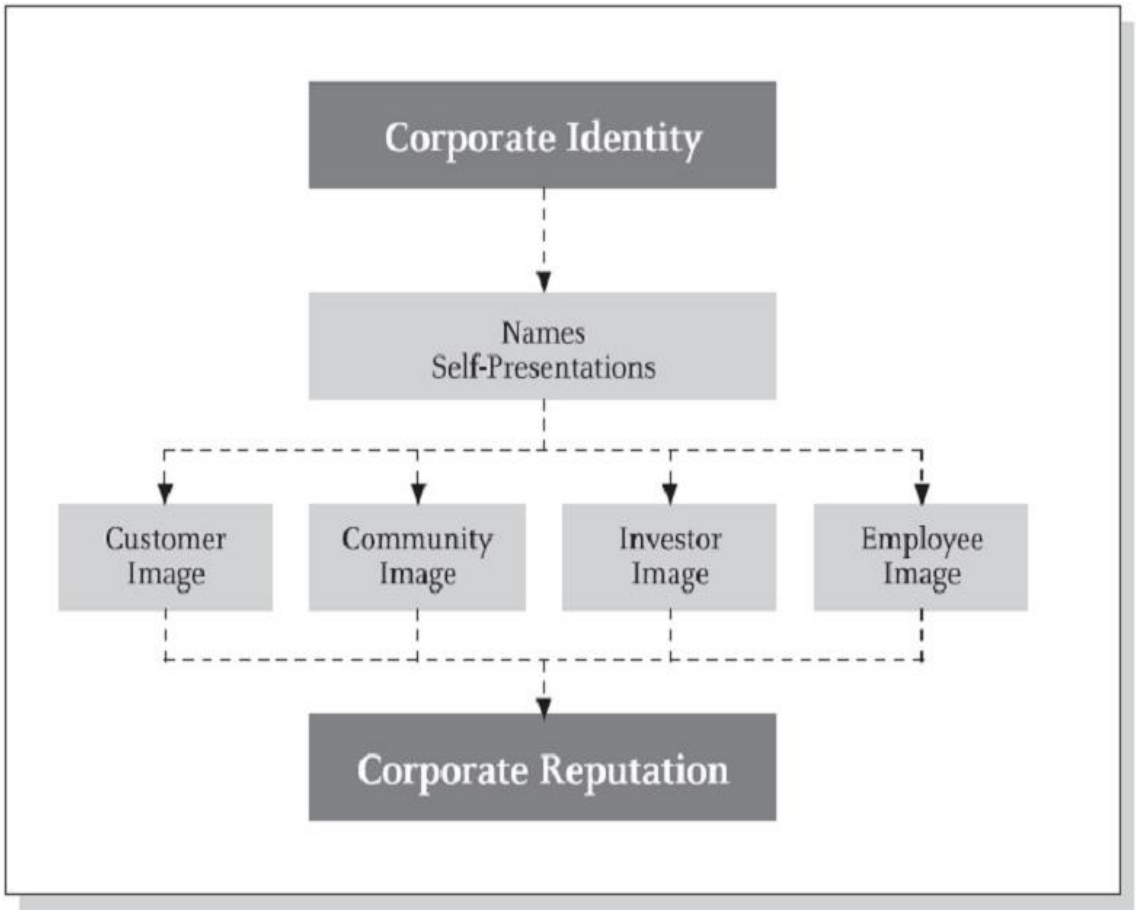

Abbildung 7.2: Hierarchische Struktur der sozialen Repräsentationen einer Organisation

spiel auf die Zahlungen angewiesen ist, um seine eigenen Ausgaben zu decken. Genau diese Problematik tritt, wie schon erwähnt, im Internet ständig auf. Es finden ständig Auftragserteilungen und Leistungseinforderungen zwischen zwei gegenseitig vollkommen unbekannten Parteien statt. Dabei spielt es keine Rolle, ob es sich bei den Parteien um Organisationen oder um Einzelpersonen handelt.

Abbildung 7.3 zeigt noch einmal einige Fälle, die bei einer Transaktion zwischen Käufer und Verkäufer auftreten können. Nicht immer tritt der gewünschte Fall ein und so entsteht einem von beiden ein gewisses Risiko. Da jede Partei sich ihres eigenen Risikos bewusst ist, versucht sie sich selbst abzusichern, indem zum Beispiel der Verkäufer durch Festlegung der Zahlungsmodalitäten Einfluss nimmt. Eine Forit-Studie über Zahlungsformen in Onlinegeschäften hat ergeben, dass derzeit 30% aller Transaktionen über Kreditkarte, 26% per Überweisung, 22% mittels Lastschrift und 13% durch Nachnahme abgeglichen werden[8]. Somit kann der Verkäufer sein Risiko mindern oder diesem sogar entgehen. Der Käufer hingegen versucht sich durch nachhaltige Informationsbeschaffung abzusichern, indem er beispielsweise die Reputation des Verkäufers ermittelt, oder andere Wege sich einräumt, wie zum Beispiel ein Umtausch- oder Rückgaberecht, wie es vom Gesetzgeber vorgeschrieben ist. Allerdings wird ein solches nicht bei Privatverkäufen in Online-Auktionen gewährt, so dass der Käufer dem Verkäufer ein gewisses Vertrauen entgegen bringen muss.

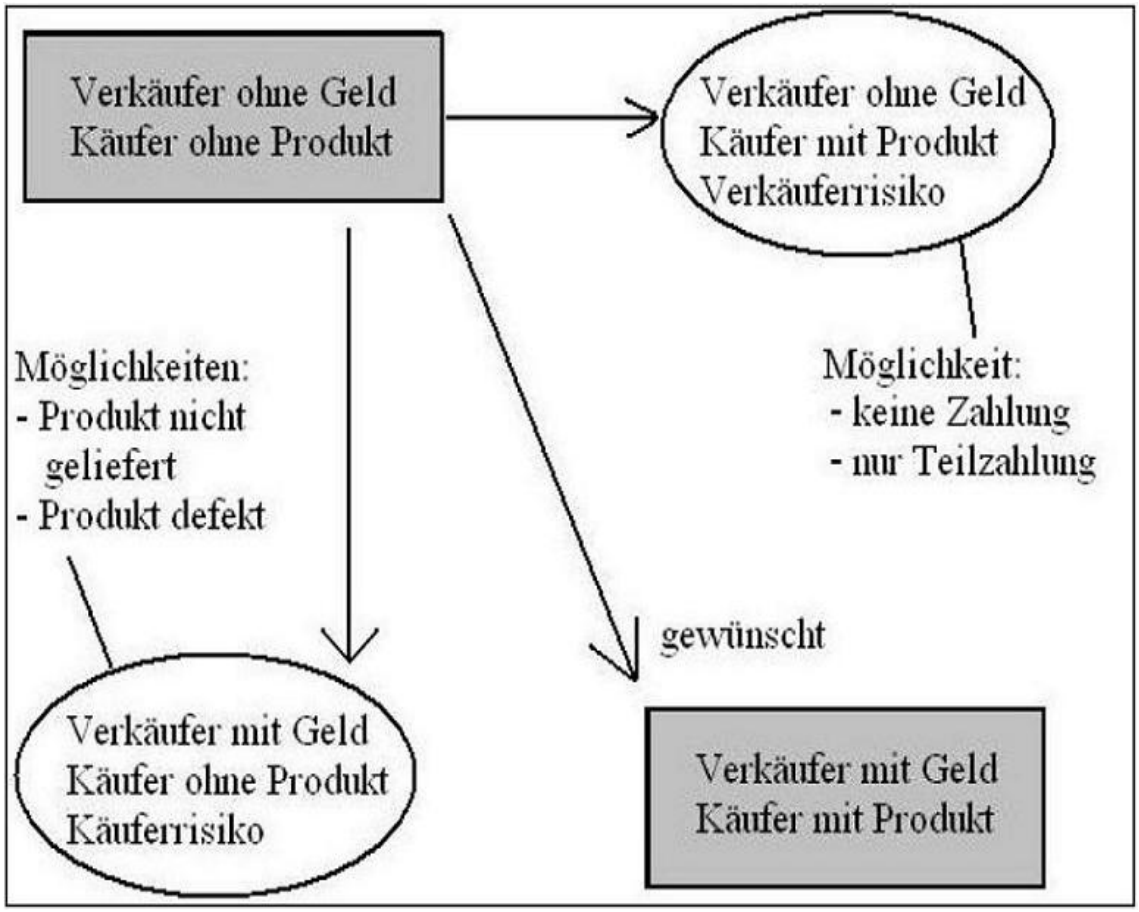

Abbildung 7.3: Gefangenendilemma im E-Commerce

7.2 Anwendungsbereiche in der Welt des Internet

Der erste Abschnitt hat die grundlegenden Begrifflichkeiten erläutert, die Hauptbestandteil dieser Ausarbeitung sind. In diesem Abschnitt werden nun die "Räumlichkeiten" betrachtet, auf die die erwähnten Aspekte ihre Anwendung finden. Sie haben bei Virtuellen Gemeinschaften einen sehr großen Einfluss, ebenso wie bei den Beziehungsnetzwerken.

7.2.1 Virtuelle Gemeinschaften

Der Begriff der Virtual Community (im nachinein als VC bezeichnet) wurde erstmals durch *Howard Rheingold* [9] definiert und beschreibt die Bildung von Gruppen mit gleichen sozialen oder wirtschaftlichen Interessen. Hierbei findet die Kommunikation fast ausschließlich über das Internet statt. Eine Webseite, die nur Chaträume, Diskussionsforen und Downloadbereiche beinhaltet, ist bei weitem noch keine Virtuelle Gemeinschaft. Solche Seiten sind nur Themesites mit interaktiven Elementen. Eine Virtuelle Gemeinschaft muss, um als solche bezeichnet zu werden, fünf Merkmale erfüllen[10]:

1. ein spezifischer Interessenschwerpunkt

2. das Vermögen, Inhalt und Kommunikation zu integrieren
3. die Verwendung von Informationen, die Mitglieder bereitstellen
4. der Zugang zu konkurrierenden Anbietern
5. eine kommerzielle Orientierung

Damit soll folgendes Ziel erreicht werden: Die Besucher werden zu loyalen Mitgliedern durch Aufbau persönlicher Beziehungen in der Community. Diese außergewöhnliche Kundenloyalität macht laut *Hagel* und *Armb* "virtuelle Communities zu einem Magneten für Kunden mit gleichen Kaufprofilen"[10]. Die Bindung an die VC entsteht also in erster Linie durch persönliche Beziehungen der Mitglieder untereinander. Dadurch eröffnet sich jedoch gleichzeitig ein Problem: Es ist eine Mindestanzahl an Mitgliedern und damit Aktivitäten notwendig, damit das nötige Wachstumspotential entwickelt wird. Erreicht man diese Anforderung nicht, besteht die Gefahr, dass sich die Gemeinschaft schnell wieder auflöst. Daher müssen Betreiber und Initiatoren solcher Communities "ein gutes Händchen" im Umgang mit den beteiligten Personen und im Management haben. Sie müssen den Mitgliedern suggerieren können, dass sie vertrauenswürdig sind und deren Interessen vertreten.

Eine besondere Form solcher VC's ist die Virtuelle Organisation. Dabei handelt es sich um einen Zusammenschluss mehrerer Partner zur Bewältigung einer gemeinsamen Aufgabe. Dies kann innerhalb eines Unternehmens oder firmenübergreifend geschehen. Der Zustand ist nur temporär und löst sich normalerweise nach Abschluss des Projektes auf. Diese Organisationen sind immer eine gute Grundlage zum Aufbau von Beziehungsnetzwerken (*siehe folgenden Abschnitt*), speziell wenn es zur interkooperativen Zusammenarbeit mehrerer Unternehmen kommt.

Der Vorteil solcher Organisationen liegt in der Aufteilung der Aufgabenbewältigung. So werden bei komplexen Projekten Teilgebiete an Firmen übergeben, die das notwendige Spezialwissen und Know-How haben. Aber noch wichtiger ist, dass die Partner nicht am selben Standort und zur gleichen Zeit agieren müssen. Durch die Nutzung der Modularisierung und Vernetzung erreicht man einen Innovationsvorsprung gegenüber ortsgebundenen Strukturen. Einen Überblick über räumliche und zeitliche Zusammenarbeit liefert die folgende Abbildung 7.4 der Anytime/Anyplace Matrix, wie sie bei *Picot / Reichwald / Wiegand* [11] verwendet wird.

Auch wenn eine Virtuelle Organisation aus vielen Teilpartnern bestehen kann, so zeigt sie sich dem Kunden doch als ein geschlossenes Ganzes. Er hat das Gefühl, dass es auf seine Bedürfnisse zugeschnitten ist, obwohl bei jedem Auftrag andere Firmen beteiligt sein können, die im Normalfall an unterschiedlichen Orten und zu unterschiedlichen Zeiten arbeiten. Dies bleibt dem Kunden aber verborgen. Daraus ergibt sich natürlich das Problem, dass das Vertrauen des Kunden ausgenutzt werden kann. Als Beispiel sei folgendes Szenario gegeben (Abbildung 7.5).

Kunde "A" hatte der Firma "XYZ" einen Auftrag erteilt, den diese Firma aber nicht zu seiner Zufriedenheit erfüllte. Daher wandte er sich im Internet an ein anderes Unternehmen. Dieses war ein Zusammenschluss mehrerer Unternehmen, zu denen Firma "XYZ" ebenfalls

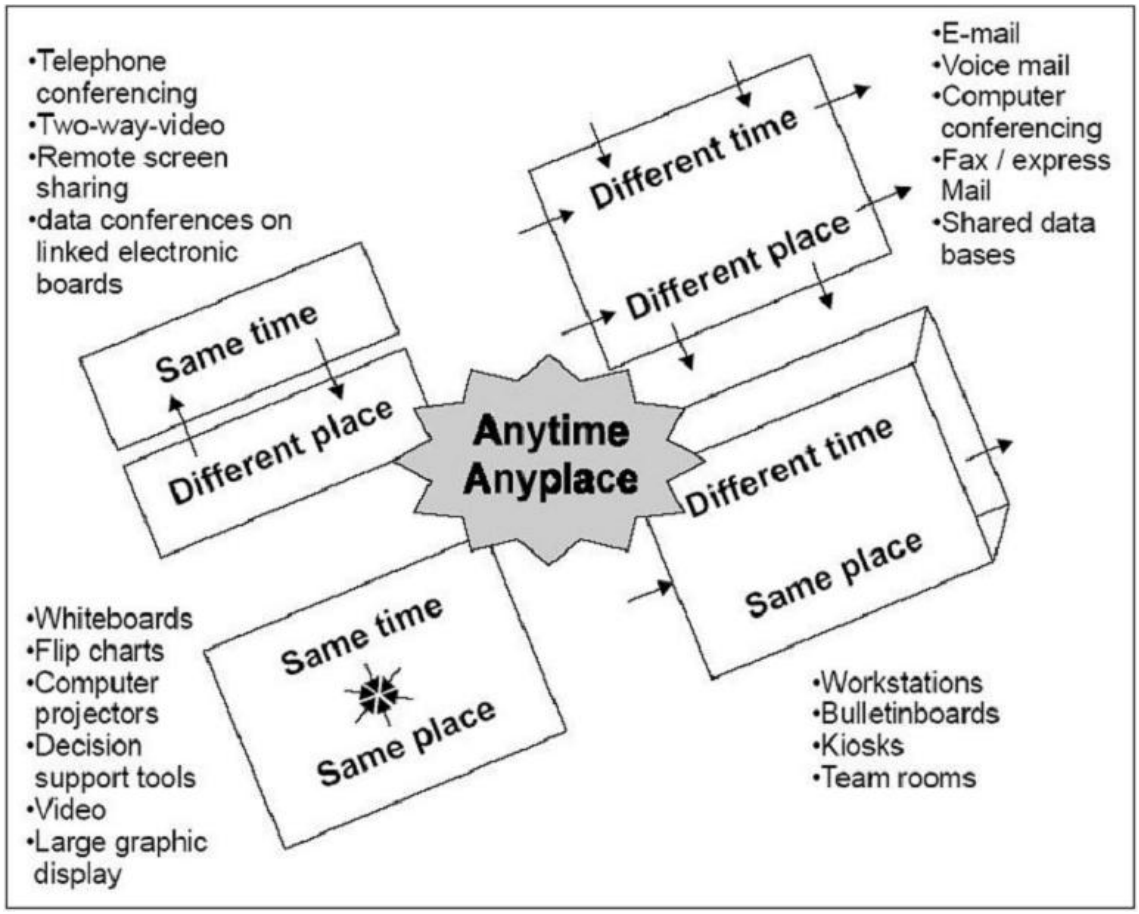

Abbildung 7.4: Anytime/Anyplace Matrix

gehörte. Dies blieb dem Kunden jedoch verborgen, weil die Firma nur spezielle Teile für das Gesamtprojekt lieferte und er sich auch nicht weiter informierte. Es ist offensichtlich, dass dem Kunden daraus ein Nachteil entstehen kann, wenn genau diese speziellen Teile anfänglich zum Scheitern des Auftrages führten. Aber grundsätzlich haben Virtuelle Organisationen große Vorteile in B2B-Beziehungen.

7.2.2 Beziehungsnetzwerke

Wie schon bei den Virtual Communities erwähnt, sind Beziehungsnetzwerke von großer Bedeutung. Bieten VC's doch eine gute Grundlage für die Entstehung solcher Netzwerke. Sobald es zu Interaktionen zwischen zwei Parteien kommt, wird sich daraus eine Beziehung aufbauen. Der Grad und die Tiefe dieser sind abhängig von Art,Umfang und Ausgang der Interaktion. Durch das Internet sind dadurch viele neue Formen von Beziehungen herangewachsen. Chat-Räume, Foren oder E-Mail-Bekanntschaften sind ebenso Teil dieser Formen, wie auch persönliche Beziehungen. Dabei sind Beziehungen zu einzelnen Personen genauso wichtig, wie zu ganzen Unternehmen. Obwohl letztere natürlich ausschlaggebend für eine Integration in eine Virtuelle Organisation sind.

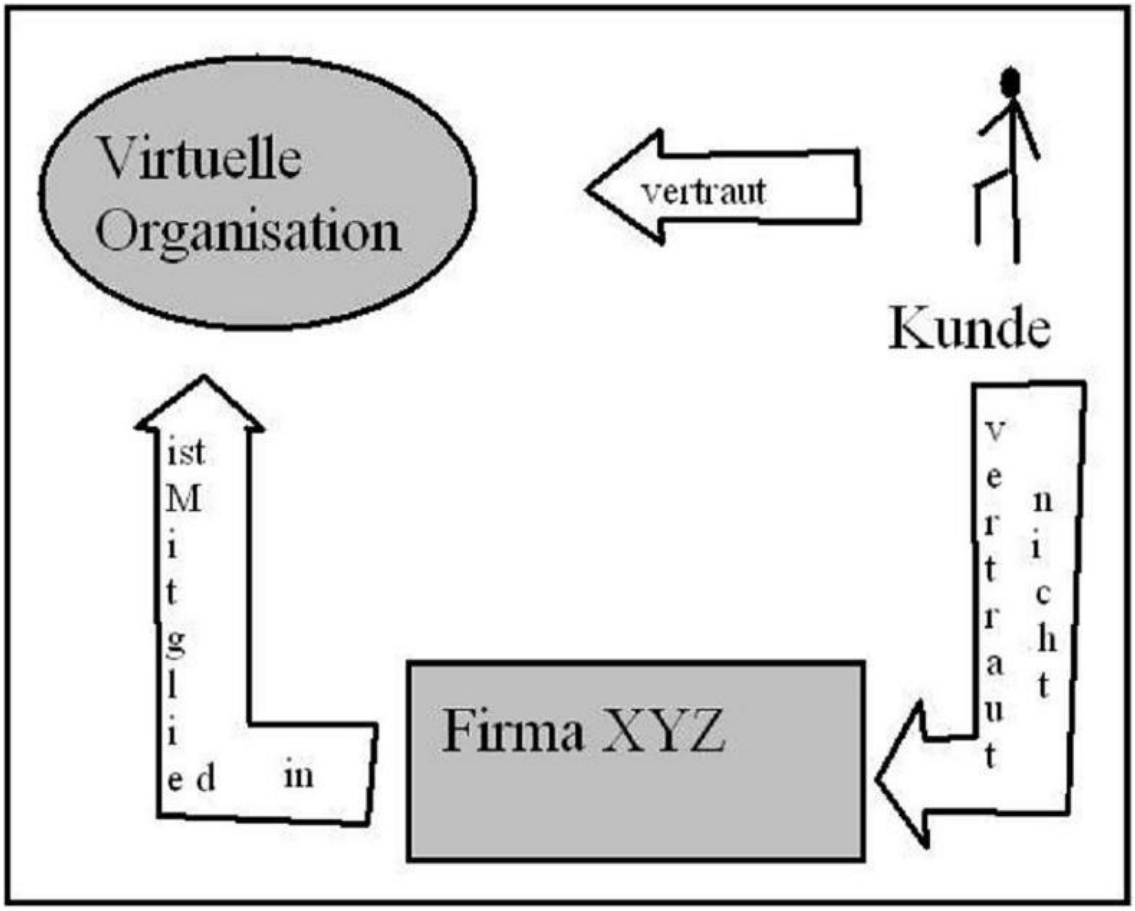

Abbildung 7.5: Problem der Transparenz Virtueller Organisationen

Auch bei der Suche nach potentiellen Partnern erweisen sich Beziehungsnetzwerke als sehr nützlich, braucht man doch nicht erst neue fremde Unternehmen oder Personen suchen, wenn man schon einige kompetente Organisationen kennt, die die Voraussetzungen für eine Zusammenarbeit erfüllen, oder zumindest selbst kompetente Partner vermitteln können. Allerdings gibt es eigens zu diesem Zweck im Internet mittlerweile mehrere Foren und Anbieter solcher Dienste. TiBiD (*Abschnitt 7.4.2*) ist eines davon. Rein formal betrachtet ist ein Beziehungsnetzwerk ein ungerichteter Graph, bei dem die Kanten nach Intensität der Beziehung gewichtet sein können. Abb. 6 stellt so ein Beziehungsnetzwerk dar.

Die Kantengewichtung drückt hier die bisherige Dauer der Beziehung in Jahren aus. Die einzelnen Personen können zum Beispiel eigenständige Experten auf einem Fachgebiet oder vielleicht auch Politiker sein. Daraus lässt sich ableiten, dass Beziehungsnetzwerke nicht nur in der Wirtschaft sondern auch in allen anderen sozialen Bereichen ihre Anwendung finden. Letztendlich hängt auch hier die Intensität wieder von dem Vertrauen ab, welches sich die beiden Beziehungspartner entgegenbringen. Einer weniger starken Beziehung wird man bei einer Entscheidung selbstverständlich auch weniger Beachtung schenken, als einem guten Bekannten.

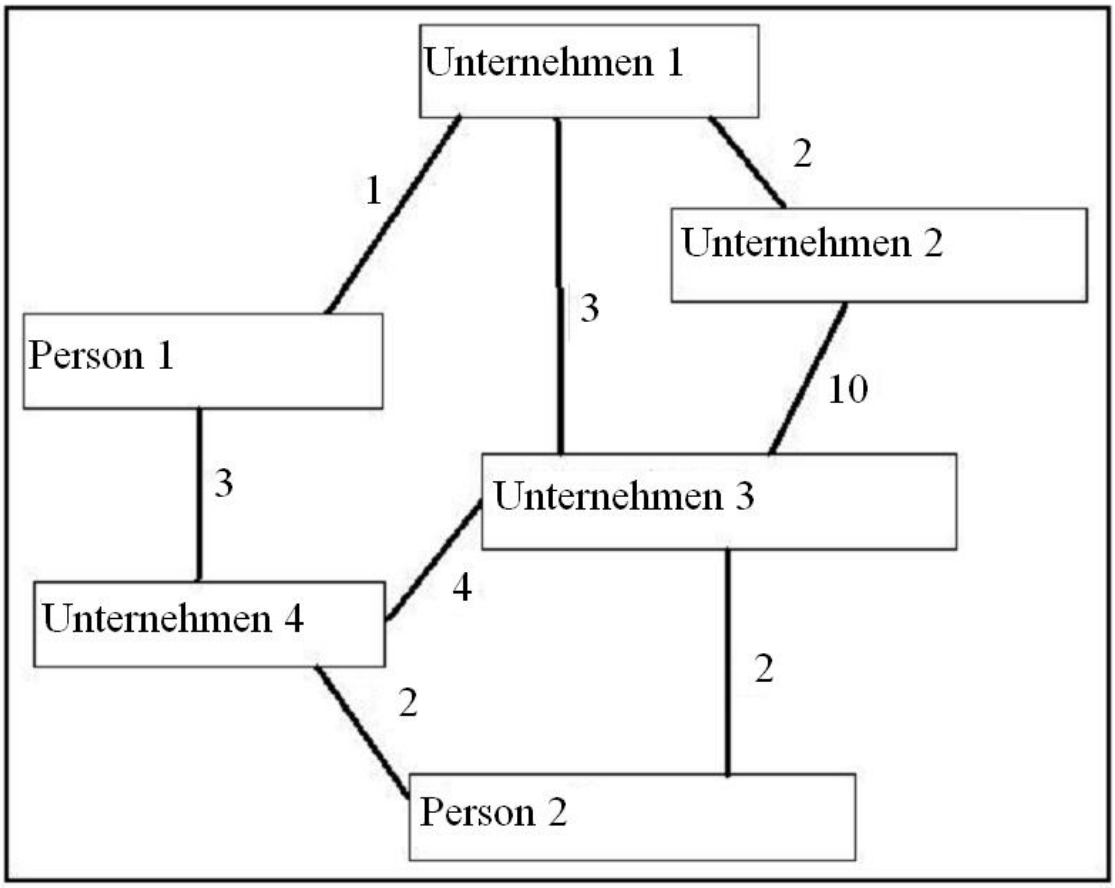

Abbildung 7.6: Visualisiertes Beziehungsnetzwerk

7.3 Methoden der Vertrauensbildung

Nachdem die ersten beiden Kapitel der Festlegung der Begrifflichkeiten und Grundlagen gedient haben, zeigt dieser Abschnitt Wege auf, die zur Bildung von Vertrauen beschritten werden können. Dazu gibt es vier unterschiedliche Hilfsmöglichkeiten. Vertrauensbildung erfolgt

1. aufgrund gemeinsamer Erfahrungen,
2. durch Identifizierung gleicher Interessen,
3. durch die Vermittlung über Dritte und
4. durch die Erwerbung von Reputation.

Alle diese Punkte werden im einzelnen in den nächsten Abschnitten angesprochen und genauer erklärt. Allerdings dienen sie nur zur Feststellung der Vertrauenswürdigkeit des Partners. Der eigentliche Vertrauensbildungsprozess ist umfangreicher und setzt sich aus mehreren Teilschritten zusammen, wobei der Großteil von der eigenen subjektiven Einschätzung und dem Willen zum Vertrauen abhängt.

- **Vertrauensbereitschaft** - Dies ist der wichtigste Punkt und muss zwingend vorhanden sein, da sonst kein Vertrauen zustande kommt.
- **Vertrauenswürdigkeit der Zielperson/Organisation** - Dies ist ebenfalls wichtig, kann aber, wie schon eingangs erwähnt, durch Informationsmangel oft nicht subjektiv eingeschätzt werden. Hierzu wird, wenn möglich, auf die Reputation oder andere Hilfsmittel zurückgegriffen.
- **Erfahrungen** - (*siehe Abschnitt 7.3.1*)
- **Situationswahrnehmung** - Dies beinhaltet die eigene subjektive Einschätzung der Situation oder Zielperson des Vertrauens.
- **Risikoanalyse** - Betrachtung des eigenen Risikos - Dies ist ein sehr wichtiger Punkt für den Vertrauensbildungsprozess, der differenziert geprüft werden muss, da die Gefahr eines Vertrauensmissbrauchs nie ausgeschlossen werden kann.

Daraus resultiert bei positiver Bewertung aller Punkte ein Mindestvertrauen gegenüber der Zielperson/Organisation - immer im Hinblick auf die zu betrachtende Situation. Wird diese Grenze unterschritten, wird es zu keiner Interaktion kommen.

7.3.1 Erfahrung als Ausgangspunkt

Es gibt zwei Arten von Erfahrungen bei der Vertrauensbildung. Zum einen sind das die allgemeinen, das Umfeld betreffend und zum anderen die speziellen, den Vertrauensgeber betreffend. Beide haben durch ihre Gewichtung mehr oder weniger Einfluss beim Vertrauensbildungsprozess, wobei die speziellen Erfahrungen natürlich höher bewertet werden, als die allgemeinen. Wie aber schon in vorangegangen Kapiteln beschrieben, sind in vielen Fällen Erfahrungen mit dem Aktionspartner nicht vorhanden. Dann spielen die allgemeinen Kenntnisse eine wichtige Rolle.

Als Beispiel sei hier ein im Umgang mit dem Internet vollkommen unerfahrener Benutzer genannt. Dieser wird bei seinem ersten Besuch auf einer Webseite wahrscheinlich übervorsichtig handeln (da er es sich selbst nicht zutraut) oder aber leichtsinnig (weil er sich nicht der Konsequenzen bewusst ist). Wohingegen ein erfahrener Internetsurfer bei ihm unbekannten Anbietern schon wissen kann, wie er sich diesen gegenüber zu verhalten hat, oder welches Vertrauen er ihnen zukommen lassen kann, da er früher schon mit ähnlichen Webseiten Erfahrungen gewonnen hat. Allerdings kann es auch sein, dass der unbedarfte Nutzer den Anbieter im speziellen kennt, weil er eventuell in dessen Geschäft schon einkaufen war. Dann hat dieser dem erfahrenenen User gegenüber natürlich einen viel größeren Vorteil, da schon ein gewisses Grundvertrauen vorhanden ist. Dieses Szenario soll Abb.7.7 verdeutlichen.

Was beiden jedoch gleich bleibt, ist die Tatsache, dass sie bei negativen Erfahrungen sicherlich kein Vertrauen aufbauen werden. Allerdings muss es auch bei positiven Eindrücken nicht unbedingt zum Vertrauen und damit zu einer Interaktion kommen, da der Prozess der Vertrauensbildung aus viel mehr als nur einem Schritt besteht. Die Erfahrung

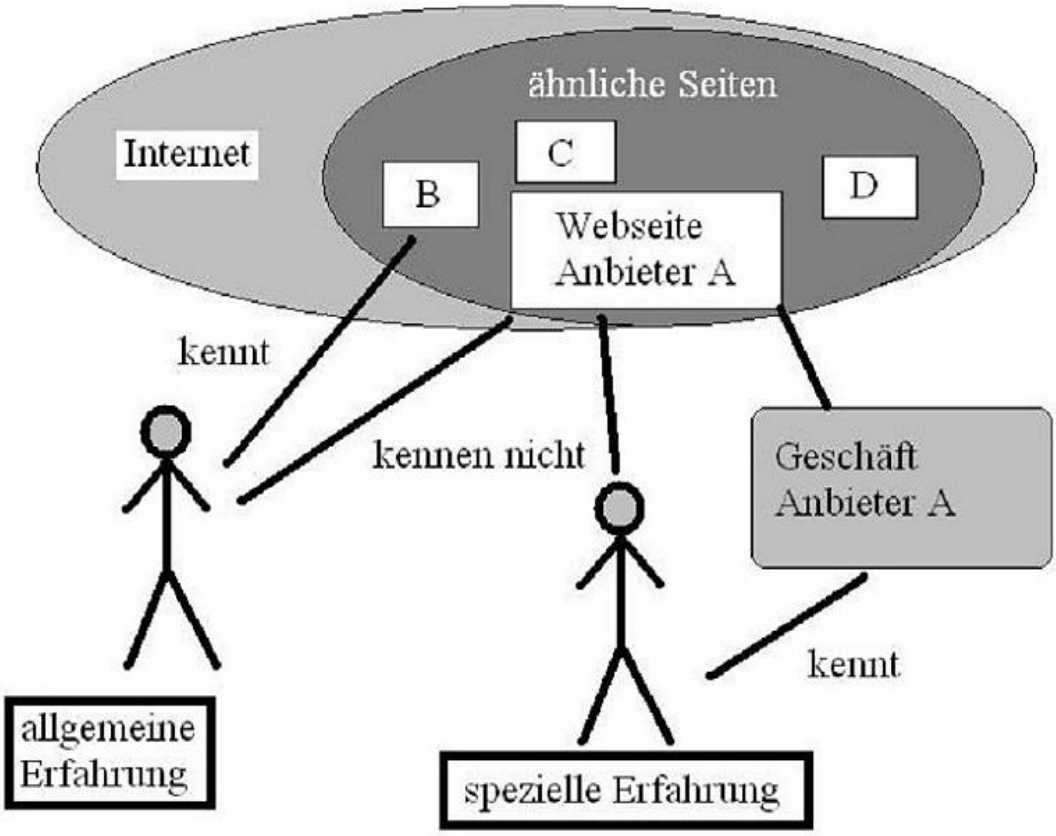

Abbildung 7.7: Unterschied allgemeine, spezielle Erfahrung

kann nur die Entscheidung, ob der Partner vertrauenswürdig ist, in eine Richtung beeinflussen. Jedoch haben eigene Erfahrungen mit der Zielperson den höchsten Stellenwert bei der Vertrauensabschätzung.

7.3.2 Basis gleicher Interessen

'Gleiche Interessen' ist eines der fünf wichtigen Merkmale für die Entstehung einer Virtuellen Gemeinschaft (*Abschnitt 7.2.1*). Durch diese Interessen wird es erst zu einer Beziehungsanbahnung kommen, die wiederum ein Kriterium für die Bildung von Vertrauen ist. Auf diesem Prinzip basieren alle Meinungsbörsen (zum Beispiel *www.dooyoo.com*) und Wissensforen. Hierzu stellt das Mitglied zu einem Thema Informationen bereit, die alle anderen abrufen und beurteilen können. Nun besteht die Möglichkeit sich mit speziellen Teilnehmern intensiver auszutauschen und daraus entwickelt sich nach und nach eine mehr oder minder starke Beziehung. Allerdings kann dies auch bedeuten, dass man gewissen Mitgliedern aufgrund deren gegensätzlichen Meinungen gar nicht mehr vertraut.

Im Bereich der Wirtschaft ist es ähnlich. Durch die Erstellung eines Beziehungsnetzwerkes und die Angabe spezieller Interessen und Bereiche, kann ein Unternehmen schnell potentielle Partner ausfindig machen. Voraussetzung dazu ist natürlich das Vorhanden-

sein entsprechender Datenbestände. Hier sei als Beispiel die Kooperationsbörse der IHK genannt. Man gibt stichwortartig die Belange des Unternehmens ein und ähnlich wie bei den Gelben Seiten findet das System andere Firmen mit dem gewünschten Interessengebiet.

Abschließend sei noch erwähnt, dass es im allgemeinen schneller zu einer Kommunikation kommt, wenn man die gleichen Interessen vertritt, denn darin liegt der Vorteil von Virtual Communities. Allerdings gilt auch hier, wie bei den Erfahrungen schon angesprochen, dass gleiche Interessen nicht gleich Vertrauen bedeutet, da damit noch nicht die Kompetenz der Zielperson im angestrebten Konsens nachgewiesen ist.

7.3.3 Der Vermittlungsweg

Als erstes versucht man seine Geschäfte mit Personen oder Organisationen zu führen, mit denen man schon positive Erfahrungen gesammelt hat. Dies ist in vielen Fällen jedoch nicht möglich, vor allem, wenn man sich auf 'unbekanntem Terrain' befindet. Bei Sachen mit geringem Wert (eine einfache Informationsbeschaffung) handelt man sicher auf gut Glück. Geht es jedoch um hohe Werte, ist die Empfehlung einer vertrauenswürdigen dritten Person ein wichtiges Kriterium bei der Wahl des Geschäftspartners - zum Beispiel beim Hausbau. Auch im Businessbereich überprüft ein Unternehmen zuerst sein Beziehungsnetzwerk, ob potentielle Partner vorhanden sind und wenn nicht, ob Personen oder Unternehmen bekannt sind, die das passende Unternehmen empfehlen oder gar vermitteln können. Bedenkt man, dass man schon als Kind gelernt hat, seinen Verwandten zu vertrauen, besonders bei Unbekanntem, so fällt es auch nicht schwer empfohlenen Personen einen größeren Vertrauensvorschuss als üblich zu gewähren. In diesem Zusammenhang stellt sich natürlich die Frage der Transitivität von Vertrauen[12].

Bei der Initiierung einer Virtuellen Organisation werden sogar extra Vermittler, sogenannte Broker engagiert, die die Erstellung, Koordination und Steuerung übernehmen. Sie zeigen sich auch dem Kunden als Repräsentant des Virtuellen Unternehmens (VU). Am Anfang haben sie die Aufgabe, geeignete Partner für das Unternehmen zu finden, die das nötige Know-How haben und sich in ihren Kompetenzen gegenseitig ergänzen. Auch müssen sie im weiteren Verlauf immer wieder neue Partner vermitteln können, da sich ein Projektziel ändern kann. Dies setzt eine hohe Fachkompetenz und ein gesundes Maß an Objektivität voraus.

Bei besonders vielen verschiedenen Unternehmen kann der Vermittler auch helfen, gegenseitiges Vertrauen aufzubauen und eventuelles opportunistisches Handeln unterbinden[12]. Allerdings hat sich das Problem der Beziehungsanbahnung nur vom Unternehmen auf den Broker übertragen, denn er wird hauptsächlich Firmen empfehlen und einplanen, die er selbst kennt. Er ist für das Finden potentieller Partner spezialisiert und kann ein umfangreicheres Beziehungsnetzwerk vorweisen, als das Unternehmen selbst. Daher bietet dieses Prinzip sehr gute Vorteile bei der Vertrauensbildung im Businessbereich. Im normalen Dienstleistungsbereich könnte man einen Immobilienmakler mit einem Broker vergleichen.

7.3.4 Vertrauen durch Reputation

Wenn keine direkte Empfehlung zur Verfügung steht, sucht man sich einen Partner der einen guten Ruf hat. Letzten Endes ist die Reputation auch nur eine Empfehlung durch Dritte[12], nur dass man diese meist nicht persönlich kennt. Wie schon im Abschnitt *Reputation* erwähnt, handelt es sich dabei um eine Bewertung der Vertrauenswürdigkeit des Partners. Diese ist im Normalfall auch allgemein zugänglich. "Zugleich kann Reputation bzw. die Angst vor einem Reputationsverlust bei opportunistischem Verhalten und einer daraus resultierenden Verringerung künftiger Kooperationsgewinne ein wirksames Sicherungsgut innerhalb einer Vertrauensbeziehung darstellen"[11]. Natürlich muss man beachten, dass es sich um Bewertungen vergangener Aktionen handelt und von wem diese abgegeben wurden.

Des Weiteren ist es wichtig, inwiefern auch der Bewertende Informationen bereitstellt, die seine Vertrauenswürdigkeit ausdrücken. Je nachdem wird nämlich die Glaubwürdigkeit der Aussage gewichtet. Man spricht in diesem Zusammenhang auch von kollaborativem Filtern: "Bei kollaborativem Filtern handelt es sich um die Methode zum Filtern von Datensätzen unter Einbeziehung von Wertungen zu den Datensätzen durch andere Personen. Eine spezielle Form von kollaborativem Filtern versucht, z.B. Personen mit gleichem Interessensprofil ausfindig zu machen und dann hoch bewertete Vorschläge von solchen Personen als Empfehlungen weiterzuleiten[14]."

Ein weiteres Problem der Reputation entsteht, wenn Bewertungen auschließlich von Fremden kommen. Denn auch wenn subjektiv gesehen viele positive Aussagen gemacht wurden, muss dem objektiv nicht zwingend vertraut werden. Schließlich besteht ja auch die Möglichkeit, eine fremde Identität anzunehmen, um sich selbst positive Bewertungen zu schreiben. Deswegen geben viele Communities zusätzlich zu den Bewertungen auch noch Informationen an, die die Dauer einer Mitgliedschaft der Identität beinhalten. Diese Betrachtungen werden ebenso im Abschnitt *eBay* eine wichtige Rolle spielen.

Diekmann/Wyder[15] haben dazu eine Einteilung der Reputationssysteme in fünf Kategorien vorgenommen. Diese sind im Einzelnen:

1. **Informelle Reputation in sozialen Netzwerken:** Diese beschreibt die hauptsächlich persönliche oder auch fernmündliche Bewertung spezieller Personen in einem sozialen Kontext.

2. **Reputation durch Markennamen:** Hierbei handelt es sich um spezielle Konsumprodukte, die ein besonderes Image aufgrund einer Modeerscheinung errungen haben.

3. **Experten-Rating:** Diese Kategorie könnte man ebenso bei der direkten Empfehlung mit eingliedern, da im Normalfall durch die Fachkompetenz der wertenden Person bereits ein großer Vertrauensvorschuss herausgebildet wurde. Ein Beispiel wäre die Stiftung Warentest, der mehr vertraut wird, als der DIN.

4. **Konsumentenrating:** Reputation durch Endverbraucher, die auch Laien sein können. Im Internet existieren dazu viele Meinungsforen.

5. **Reputationsverfahren von Internet-Auktionen**

Der letzte Punkt wird in Kapitel 7.4.1 näher beleuchtet. *Kollock* unterscheidet nur zwischen positiven und negativen Reputationssystemen[16]. Hierbei werden entweder nur gute oder schlechte Erfahrungen ausgewertet und der öffentlichkeit vermittelt. Systeme für negative Reputation können im Internet in drei Formen vorkommen: als Webseiten, als Newsgroups oder als Blacklists. Es gibt aber auch äquvalente Formen in der realen Welt. Jedoch wurden viele dieser Informationsseiten aufgrund vermehrt autretender Probleme wieder abgeschafft. Positive Reputationssysteme haben sich in den letzten Jahren besonders in Online-Auktionshäusern etabliert. Dabei werden verstärkt die erfolgreichen Handlungen eines Teilnehmers hervorgehoben. Dies hat den Vorteil, dass bei einer einzigen Verfehlung nicht sofort die vollständige Vertrauenswürdigkeit des Mitglieds zerstört wird. Es treten kaum Identitätswechsel auf, da die bis dahin errungene Reputation verloren gehen würde.

7.4 Praktische Anwendungsbeispiele

In diesem Abschnitt wird gezeigt, wie einige Aspekte der vorherigen Kapitel in konkreten Bespielen realisiert wurden. Dazu wird als nächstes die eBay-Community betrachtet, die aufgrund ihres großen Erfolges ein Paradebeispiel für das Funktionieren eines Online-Handelsportals ist. Weiterhin wird das Projekt TiBiD näher beleuchtet - eine Applikation zum Auffinden potentieller Partner in Virtuellen Organisationen, als Beispiel der Implementierung eines Beziehungsnetzwerkes.

7.4.1 eBay

Bei eBay handelt es sich um eine Online-Auktionsbörse, bei der Versteigerungen in fast allen Produktbereichen von Unternehmen und Privatpersonen angeboten werden. Hierzu vorerst ein paar allgemeine Informationen [17]:

- gegründet 1995 in Kalifornien durch Pierre Omidyar als Marktplatz für den Austausch von Sammelartikeln
- es sind mittlerweile 104,8 Millionen Mitglieder registriert
- ist weltweit in 28 internationalen Märkten präsent
- mehr als 25 Mio. Artikel in 50000 Kategorien sind ständig vorhanden, täglich kommen 3,5 Mio. Artikel neu dazu
- allein in den USA erwirtschaften über 430000 kleine und mittelständische Unternehmen einen Teil ihres Kapitals bei eBay

eBay ist die weltgrößte Handelsplattform. Der allgemeine Ablauf beginnt mit dem Mitbieten bei einer Auktion. Es besteht auch die Möglichkeit Sofortkäufe zu tätigen. Natürlich sollte man sich als Käufer vorher anhand des Bewertungs-Forums informieren, ob der Verkäufer die gewünschte Vertrauenswürdigkeit aufweist. Nach Abschluss des Bietvorganges setzen sich die beiden Parteien mittels Email in Verbindung. Der Verkäufer teilt dem Käufer dabei die Zahlungsmodalitäten und eventuell seine Bankverbindung mit. Die meist verwendete Zahlungsform ist die Überweisung, aber auch die Nachnahme wird angeboten (hauptsächlich durch Unternehmen). Eine Privatperson benutzt bei Verkäufen meistens nur die Vorauszahlung. Damit möchte sie sich gegen Betrugsversuche absichern. Die dritte Möglichkeit des Austauschens von Geld und Waren bietet der von eBay bereitgestellte Treuhandservice. Dieser teilt sich in fünf Schritten auf[18]:

1. Der Käufer überweist das Geld auf ein von iloxx eingerichtetes Treuhandkonto.
2. Der Treuhandservice meldet den Zahlungseingang dem Verkäufer
3. Der Verkäufer schickt die Ware an den Käufer.
4. Der Käufer inspiziert die Ware und bestätigt dem Treuhandservice den Wareneingang.
5. Der Treuhandservice überweist das Geld an den Verkäufer.

Dies ist zwar die sicherste Form der Transaktion, aber auch die zeitlich aufwendigste. Es kann zwar grundsätzlich jeder Artikel darüber abgewickelt werden, aber eBay empfiehlt ein Mindestwarenwert von 200 Euro. Zusätzlich sollte vor dem Einsteigen in eine Auktion mit dem Verkäufer abgesprochen werden, ob er diesen Dienst unterstützt, da hierfür extra Kosten anfallen (von 2,50 bis 50 Euro je nach Warenwert).

Ist der eigentliche Kauf abgeschlossen, können beide Parteien noch eine Bewertung des jeweils anderen vornehmen. Diese besteht aus einem einfachen Voting (positiv, neutral oder negativ) und einem selbst formulierten Kommentar, welche im Bewertungsprofil des jeweiligen Mitgliedes abgespeichert werden und von allen aufgerufen werden können.

Im Profil wird die Gesamtanzahl von Bewertungen aufgeführt und wie sich diese auf die letzten zwölf und sechs Monate sowie auf den letzten Monat verteilen. Des Weiteren sind Angaben zur Dauer der Mitgliedschaft und eventuell früherer oder noch aktiver Identitäten aufgeführt. Die Punkte berechnen sich als Summe aller positiven abzüglich der negativen Bewertungen. Bei entsprechend großer Punktezahl vergibt ebay als besonderes Kennzeichen einen Stern unterschiedlicher Farbe, gelb ab zehn, türkis ab 100, violett ab 500, rot ab 1000 und gold ab 10000. Außerdem kann ein Handelspartner nur eine Bewertung pro Transaktion abgeben, somit kann man nicht ohne Weiteres positive Bewertungen zusätzlich dazu bekommen.

Ein Kommentar kann bei einer negativen oder auch neutralen Bewertung auf die Umstände der Unzufriedenheit aufmerksam machen, sollte aber immer fair und nachvollziehbar sein. Schließlich kann der Partner darauf durch seine Wertung reagieren. Es besteht auch die Möglichkeit zu juristischen Schritten bei einer mutmaßlichen Verleumdung. Ebenso

Abbildung 7.8: Bewertungsprofil eines ebay-Mitglieds

kann ein Mitglied sein Profil als privat festlegen und die Einsicht durch andere verwehren. Allerdings behindert dies den Vertrauensaufbau zu potentiellen Partnern. Zusätzlich zur Angabe einer Email-Adresse werden neue Mitglieder, aber auch solche die nur ihren Benutzernamen gewechselt haben, gesondert markiert. Dies ist zwar kein sicherer Schutz vor Identitätsmissbrauch, deutet aber für andere Mitglieder darauf hin, dass sie bei dieser Person die nötige Vorsichtig walten lassen sollten.

Abschließend zeigt sich, dass die Vertrauensbildung durch Reputation bei ebay sehr vorteilhaft entwickelt ist, was die anfangs genannten Zahlen unmissverständlich belegen.

7.4.2 TiBiD - Ein Projekt zum Aufbau von Beziehungen im Internet

Bei TiBiD (Telekooperation in Beziehungsnetzwerken für informationsbezogene Dienstleistungen) handelt es sich um die theoretischen Hintergründe und Voraussetzungen einer Community-Plattform, die besonders für junge Unternehmen zum Aufbau von Beziehungsstrukturen durch das Finden geeigneter Partner nützlich sein kann. Dabei wird zusätzlich die Bildung von Vertrauen mit unterstützt. Des Weiteren kann es auch für Kunden als zentrale Anlaufstelle genutzt werden, um kompetente Leistungsträger für spezielle Aufgaben ausfindig zu machen. Es existiert auch schon ein Prototyp der unter der Bezeichnung "TUMmelplatz" auf der Webseite *WWW.UNTERNEHMERTUM.DE* erreichbar ist[19].

Der Ablauf der Zusammenarbeit zwischen Plattform und Unternehmen beginnt mit der Registrierung der Firma bei der Community. Hierbei sollten gesonderte Sicherheitsmaßnahmen getroffen werden, die einen Missbrauch der Daten oder die nachträgliche Veränderung durch nicht authorisierte Nutzer verhindern. Im allgemeinen sind das die passwortgeschützte Anmeldung und bei Bedarf auch eine SSL-Verbindung.

Beim Anmeldevorgang werden unternehmensrelevante Angaben zu allgemeinen Informationen und Kontaktanschrift gemacht. Diese werden nach vorheriger Überprüfung in einer

Alle erhaltenen Bewertungen | Von Käufern | Von Verkäufern | Alle abgegebenen

9 Bewertungen für rayder77 (0 in gegenseitigem Einverständnis zurückgezogen) — Seite 1 von 1

Kommentar	Von	Datum/Uhrzeit	Artikelnummer
*** von *** Danke. Bestens gelaufen. Top Ebayer.	Verkäufer amanda1896 (38 ☆)	24.05.04 02:43	3096103555
Zuverlässig, nett, problemlos! Ware ok! Wer sind die Mädels?	Käufer kaispath (216 ☆)	21.12.03 19:45	3366056720
netter Kontakt! Reibungsloser Ablauf! Gerne wieder!	Verkäufer topbrandproducts (108 ☆)	26.11.03 20:39	2764332948
Danke für die reibungslose und angenehme Transaktion. Exzellenter Käufer. Note 1	Verkäufer hegenloh (10333 ☆)	24.11.03 10:20	2766794004
Korrekte, schnelle Abwicklung. Jederzeit wieder. gsm-welt.de	Verkäufer akkumann01 (41314 ☆)	15.11.03 07:28	2762331127

Abbildung 7.9: Bewertungskommentare

Datenbank im Unternehmensprofil gespeichert. Danach ist es notwendig Informationen zur Unternehmenskultur zu erlangen, da diese später beim Finden von Partnern und zur Vertrauensbildung benötigt werden. Dies kann man mit Hilfe von geeigneten Fragen in einem Fragebogen erreichen. Zumindest müssen aber die Kernkompetenzen der Organisation aufgeführt werden.

Abschließend sollten noch alle Beziehungen des Unternehmens mit aufgenommen werden. Da dies in vielen Fällen aber sehr zeitaufwendig ist, weil jede Beziehung bestätigt werden muss, kann dieser Vorgang auch nach und nach vollzogen werden - bei späteren Besuchen der Community zum Beispiel. Die Plattform soll in diesem Zusammenhang die selbständige Suche anhand des Adressbuches der Firma unterstützen. Wenn das Unternehmen seine Beziehungen nicht bekannt machen will, entfällt dieser Punkt. Dies bringt aber auch Nachteile mit sich, da gleichzeitig zur Angabe eines bereits bekannten Partners dessen Mitgliedschaft geprüft und zum anderen das Beziehungsnetzwerk der Organisation entsprechend erweitert wird.

Nach Bereitstellung der allgemeinen Daten und Kernkompetenzen kann das Unternehmen sofort mit der Suche nach Kooperationspartnern beginnen, vorausgesetzt es liegt ein relevanter Auftrag vor. Bei diesem Vorgang ist es wichtig, die Projektanforderungen so spezifiziert wie möglich anzugeben. Dazu definiert *Leckner* in seiner Ausarbeitung zum Projekt folgende wichtige Punkte[20]:

- Textuelle Kurzbeschreibung des Projekts: *Worum geht es?*
- Auswahl der nötigen Kompetenzen aus der Kompetenzliste bzw. (falls nötig) Definition neuer Kompetenzen
- Terminliche Festlegung für das Projekt: *Bis wann soll es fertig sein? Bis wann kann man sich als Partnerunternehmen bewerben? Etc.*
- Anforderungen an einen potentiellen Partner
- Persönliche Ansprechpartner in den bisherigen VU-Mitgliedsunternehmen

Wenn die Daten innerhalb der Plattform veröffentlicht sind, können sich Mitglieder, die die Anforderungen erfüllen, darauf bewerben. Eine Bestätigung der Teilnahme in der VU kann dabei nur durch diese selbst erfolgen.

Da viele Mitglieder nicht ständig die Plattform besuchen und nach Aufträgen Ausschau halten, muss ein Unternehmen nicht zwangsläufig warten, bis sich ausreichend Bewerber gemeldet haben. Es kann auch selbständig eine Suche initiieren. Hierzu gibt es vier verschiedene Arten der Suche[20]:

1. **Unternehmensprofile durchsuchen:** Eine Suchmaschine unterstützt das Finden von Mitgliedern mit geeigneten Kernkompetenzen.
2. **Beziehungsnetz analysieren:** Suche von Unternehmen mit den passenden Anforderungen im eigenen Beziehungsnetzwerk.
3. **Automatisches Matching:** Die Plattform sucht selbständig nach passenden Partnern - vorzugsweise in Beziehungsnetzwerken der VU-Mitglieder.
4. **Kollaboratives Suchen:** Es wird speziell nach Unternehmen mit ähnlichen Profileigenschaften gesucht und deren Aktionshistorie auf postive Zusammenarbeit mit anderen Organisationen geprüft. So können auch Partnerschaften entstehen, die nicht Teil des eigenen Netzwerkes sind.

Gleichzeitig mit dem Finden von geeigneten Unternehmen werden Informationen über Reputation und Aktionshistorie bereitgestellt, die zusätzlich Aufschluss über Eignung und Vertrauenswürdigkeit eines potentiellen Partners geben. Außerdem wird angezeigt, ob eine Firma zum Zeitpunkt der Suche schon Mitglied in einer anderen VU ist. Dies hat Einfluss auf die Vertrauensbildung, da die Gefahr eines Know-How-Abflusses besteht[21]. Deswegen können Plattform-Mitglieder auch angeben, ob überhaupt Interesse und Ressourcen für die Teilnahme in einer Virtuellen Organisation zur Verfügung stehen. Somit werden Firmen ohne Interesse gar nicht erst informiert.

Es lässt sich daher feststellen, dass diese Community-Plattform einen deutlichen Innovationsvorsprung für alle Mitglieder bietet. Allein der Aufwand an Zeit der durch das Suchen anhand spezieller Kriterien gespart wird, ist ein großer Vorteil. Allerdings darf auch hier nicht vergessen werden, dass es sich bei einer VU nur um eine temporäre Organisation handelt. Dies könnte einige Mitglieder dazu ermutigen, sich nicht so intensiv an die Nutzungsregeln zu halten. Daher ist eine ständige Kontrolle unabdingbar.

7.5 Zusammenfassung und Aussicht

"Am Anfang zwischenmenschlicher Beziehungen steht ein Vertrauensvorschuss. Wir wären gar nicht lebensfähig, wenn wir jedem, der uns auf der Straße begegnet oder neben uns in der U-Bahn sitzt, unterstellen würden, dass er uns möglicherweise verprügeln, ausrauben oder umbringen möchte. Vertrauen wächst durch besseres Kennenlernen, aber es wächst nicht ins Grenzenlose"[22], denn grundsätzlich ist der Prozess der Vertrauensbildung durch subjektive Einschätzungen geprägt. Es gibt zwar Hilfsmittel die eine Entscheidung positiv mitbeeinflussen können, diese werden aber meistens auch nur durch Dritte bereitgestellt, sei es durch einfache Informationen oder innerhalb einer ganzen Plattform. Diesen Personen müssen wir ebenfalls vertrauen, auch wenn sie 'nur' Informationen anbieten, denn inwiefern kann man prüfen, ob gerade diese für uns relevanten Aussagen mit bestem Wissen und Gewissen gemacht wurden. Eine Möglichkeit wäre die digitale Signatur der Person in Verbindung mit einem Zertifikat einer übergeordneten Stelle. Die Frage wäre dann nur noch, wer bereit ist, alle im Internet getätigten Angaben und Informationen zu überprüfen. Dies ließe sich zum derzeitigen Stand nicht realisieren. Daher bleibt nur der Weg der Selbsterfahrung. Sicher wird auch ein Neukunde bei seinem ersten Besuch in der digitalen Einkaufswelt nicht gleich so risikobereit sein, dass er sich seine eigenen Erkenntnisse sammeln kann. Wenn in naher Zukunft durch den steten Anstieg des E-Commerce nicht mehr nur 26% der Internet-Nutzer Online Einkaufen gehen, werden sich auch in diesem Bereich bessere Erkenntnisse ergeben. Die als Beispiel genannten Reputationssysteme und Community-Plattformen bieten dafür die notwendige Grundlage.

Literaturverzeichnis

[1] Galla M., Wagner M., 'Partnersuche im E-Business',
www11.informatik.tu-muenchen.de/publications/pdf/Galla2001.pdf, 2001

[2] http://de.wikipedia.org/wiki/Vertrauen besucht im Mai 2004

[3] Ratnasingham P., 1998, 'The importance of trust in Electronic Commerce', Internet Research: Electronic Networking Applications and Policy 8 4 pp313-321

[4] Mayer RC, Davis JH and Schoorman FD, 1995, 'An Integrative Model of Organisational Trust', Academy of Management Review 20 3 pp709-15

[5] http://de.wikipedia.org/wiki/Reputation besucht im Mai 2004

[6] Fombrun, J. Charles. 'Reputation. Realizing Value from the Corporate Image.' Boston,Massachusetts 1996

[7] Bazil, Vazrik 2001, 'Reputation Management - Die Werte aufrecht erhalten',
www.luchterhand.de/hlv_con.nsf/datei/04906000Leseprobe/
$File/04906000Leseprobe.pdf

[8] Studie: Bezahlen im Internet 25.07.2000
http://www.intern.de/news/751.html besucht im Juni 2004

[9] Rheingold, H. 'The Virtual Community - freie Online-Version des gleichnamigen Buches von 1994', verfügbar unter URL: http://www.rheingold.com/vc/book/

[10] Hagel, John/ Armb, A.G. 'Net.Gain - Profit im Netz. Märkte erobern mit virtuellen Communities', Gabler Verlag, Wiesbaden 1997, S.77

[11] Picot A., Reichwald R., Wigand R. 'Die grenzenlose Unternehmung', Gabler Verlag, Wiesbaden, 1996

[12] Clarke, Roger 'Trust in the Context of e-Business'
http://www.anu.edu.au/people/Roger.Clarke/EC/Trust.html#Srces 2001, besucht im Juni 2004

[13] Faisst W., Birg O. 'Die Rolle des Brokers in Virtuellen Unternehmen und seine Unterstützung durch die Informationsverarbeitung', Universitäten Bern, Leipzig und Erlangen-Nürnberg, Arbeitspapier Nr. 17/97, 1997

[14] Borghoff U., Schlichter J. 'Rechnergestützte Gruppenarbeit: Eine Einführung in Verteilte Anwendungen', 2. Auflage, Springer-Verlag Berlin Heidelberg 1995, 1998

[15] Diekmann, Andreas/Wyder, David 'Vertrauen und Reputationseffekte bei Internet-Auktionen' In: Kölner Zeitschrift für Soziologie und Sozialpsychologie 54, H. 4, S. 674-693, 2002

[16] Kollock, Peter 'The Production of Trust in Online Markets' In:

http://www.sscnet.ucla.edu/soc/faculty/kollock/papers/online_trust.htm besucht im Juni 2004

[17] ebay - Zahlen und Fakten,

http://presse.ebay.de/news.exe?page=index&comp_id=100000

&date1=01012001&date2=04062004&h=11 besucht im Juni 2004

[18] ebay - Treuhandservice,

http://pages.ebay.de/help/buy/buytrust-escrow.html, besucht im Juni 2004

[19] TiBiD Prototyp, http://www.telekooperation.de/tibid/prototyp.html besucht im Juni 2004

[20] Leckner, Thomas 'Kontaktanbahnung in Virtuellen Unternehmungen - Konzeption einer Plattform zur Unterstützung der Kontaktanbahnung in Virtuellen Unternehmungen mit Hilfe von Beziehungsnetzwerken und Community-Konzepten', In: www11.informatik.tu-muenchen.de/publications/ pdf/da-leckner2001.pdf Diplomarbeit TU München Fakultät Informatik 15. Februar 2001

[21] Hermle F., Diplomarbeit: 'Möglichkeiten und Grenzen der Vertrauensbildung in virtuellen Unternehmungen', Universität Konstanz 1998

[22] Berner, Winfried 'Vertrauen: Der steinige Weg zu einer "Vertrauenskultur"

http://www.umsetzungsberatung.de/psychologie/vertrauen.php besucht im Juni 2004

Kapitel 8

Internet Economics: Advanced Pricing Schemes for Content

Christoph Hölger

Dieses Kapitel befasst sich mit der Bepreisung von "online-content", auf welchen Strategien die Preisfestlegung basiert und welche speziellen Faktoren des Online-Handels zu berücksichtigen sind. Die Darlegung von E-Business-Strategien, über die Content vertrieben wird, und Distributionswege werden angesprochen.

Inhaltsverzeichnis

8.1 Einleitung

In dieser Arbeit im Rahmen des Seminars: Internet Economics IV, sollen Abrechnungs- und Verrechnungssysteme und -dienste des Internets in Hinsicht auf ihre wirtschaftliche Bedeutung, ihre Nutzerfreundlichkeit und auch in Bezug auf ihre technische Umsetzbarkeit geprüft werden. Bepreisungstrategien und Klassifizierung von "online-content" erfährt eine Vorstellung. Ebenfalls vermittelt werden ausgewählte "Inhalte des Internets", die auf diese Dienste und Systeme Bezug nehmen. Hintergründig für die Entwicklung solcher Systeme war und ist die Notwendigkeit Services im Internet abzurechnen. Hierbei müssen natürlich Kriterien wie z.B. Kundenfreundlichkeit und Technische Umsetzbarkeit beachtet werden.

8.2 Gegenwärtige Situation

Oberflächlich betrachtet ist die Situation folgende: Personen, die heutzutage privat das Internet nutzen, sind gewohnt, dass nahezu jeder Dienst, jede Website, jede Information kostenlos zur Verfügung steht, denn für den Nutzer direkt kostenpflichtige Angebote und Inhalte sind zur Zeit noch in der Minderheit [1]. Dies ist aber nur der erste Eindruck, auch im Internet gilt der Leitsatz: "Nichts ist umsonst". Wie aber sollen und werden Preise für Inhalte des Internets, seien es Dienste oder Informationen festgelegt. Beim Online-Verkauf fallen oft große Teile von Transaktions- und Zwischenhändlerkosten weg. Diese Kostenersparnis seitens der Anbieter hat Einfluss auf die Preise, die Content-Anbieter von Kunden verlangen. Das Preismanagement ist somit von zentraler Bedeutung im E-Business geworden [23].

8.3 Arten und Klassifizierung von "Online-Content"

Allgemein ist "online-content" alles, was innerhalb einer Web-Seite dargestellt wird. Die Form des Auftretens kann hierbei sehr unterschiedlich sein, sei es Text, Bild/Grafik, Film, Sprache oder Musik. PaidContent ist ein Spezialisierung des normalen Content, für die der Nutzer Geld zu zahlen hat. Die Form der Bezahlung variiert (Kap. 8.5). Eine Einteilung in Arten und Klassen von "online-content" erscheint erstens bei der existierenden Vielfältigkeit, zweitens im Hinblick auf eine Zuordnung: "Content zu Pricing Strategie" sinnvoll.

Das Klassenkonzept erfolgt mit dem Ziel Inhalte so einteilbar zu gestalten, dass eine Auswahl in Hinblick auf kommerzielle Ziele vereinfacht wird [18].

1. Standard-Content

   Darunter fallen typische Informationen, die eigentlich auf jeder Website zu finden sein sollten. Also vor allem ein Impressum, eine Anfahrtsskizze, Allgemeine Geschäftsbedingungen, Öffnungszeiten, Ansprechpartner und ähnliches. Diese Angaben bilden bildlich gesprochen das unabdingbare Content-Gerüst. Ein Blick auf einige

Websites selbst bekannter und ambitionierter Online-Anbieter zeigt jedoch, dass dies keine Selbstverständlichkeit ist. Wer allerdings noch nicht einmal (persönliche) Ansprechpartner oder ein Impressum auf seiner Website nennt, braucht sich über fehlende Anfragen nicht zu wundern.

2. Lange aktueller Content vs. Content mit kurzer 'Halbwertszeit'

   Manche Inhalte sind nahezu ewig "haltbar", weil Sie lange aktuell bleiben (z.B. allgemeine Hintergrundinformationen, Foto-Serien eines bestimmten Ortes). Die Wichtigkeit möglichst oft aktualisierter Inhalte wird immer wieder betont und leuchtet ein. Haupthindernis sind jedoch die hohen Kosten für Inhalte, die mit einer stetigen Aktualisierung verbunden sind. Doch nicht nur auf die zu aktualisierenden Content-Bausteine ist zu achten: Der Site-Betreiber ist gut beraten, wenn er mit Bedacht Inhalte auswählt, die eine lange Zeit nicht ausgetauscht werden müssen. Da bei der digitalen Distribution von Content über die Website nur Fixkosten für die Erstellung (bzw. den Einkauf bei fremdbezogenem Content) relevant sind, verteilt sich dieser Fixkostenblock mit jedem neuen Nutzer auf mehrere Schultern.

3. Exklusiver vs. nicht-exklusiver Content

   Jeder Website-Betreiber sollte einen möglichst hohen Anteil an exklusiven Inhalten anstreben, die nur auf seiner Website zu finden sind. Zwar ist dies teuer, da die Inhalte erst neu geschaffen und bezahlt werden müssen, doch lohnt sich dies, da die Site nunmehr über ein Alleinstellungsmerkmal verfügt. Einige Sites verfolgen bereits heute diesen Ansatz, begehen aber den Fehler, die vorhandenen Exklusiv-Inhalte nicht ausreichend herauszustellen. Wer die Kosten und Mühen umfangreichen Exklusiv-Contents nicht scheut, sollte das so geschaffene Alleinstellungsmerkmal auch nutzen und aggressiv kommunizieren (z.B. mit Hinweisen wie "Diesen Leitfaden finden Sie nur bei uns", "Exklusiv bei uns", "Sonst nirgendwo im Netz" etc.). Manche Site-Verantwortliche mögen hier einwenden, dass die Kosten für selbst produzierte Inhalte im Zuge einer Zweit-, bzw. Mehrfachverwertung ('Content-Syndication') gemindert werden können. Dieser Verkauf des eigenen Contents mag zwar für Einnahmen sorgen, ist aber fragwürdig, da das angesprochene Alleinstellungsmerkmal der Exklusivität der Inhalte dann eben nicht mehr besteht.

4. Text-Content vs. Multimedia-Content

   Bislang verstand man unter Content einer Website hauptsächlich Text-Inhalte. Mit dem Aufkommen von Internet-Pauschaltarifen und höheren Bandbreiten gewinnen multimediale Content-Bausteine ständig an Bedeutung. Darunter verstehen wir ausdrücklich nicht jene zwar gut gemeinten, jedoch völlig nutzlosen Flash-Animationen und ähnliche Gimmicks. Insbesondere informative, zielgruppengerechte Streaming-Media-Angebote sind Content-Bausteine mit konkretem Mehrwert. Experten- (Audio-)Interviews, Produktdarstellungen per Streaming Video oder etwa Live-Übertragungen zielgruppenrelevanter Ereignisse - die Möglichkeiten sind endlos.

5. Auf verschiedene Kaufphasen bezogener Content

   Üblicherweise konzentrieren sich Anbieter bei ihren Content-Konzepten nur auf solche Inhalte, die konkret auf den Kaufzeitpunkt ausgerichtet sind oder den Kauf

möglichst direkt auslösen sollen. In der Regel also platte, nichtssagende Eigenreklame. Dabei wird aber vergessen, daß jedem Kauf eine mehr oder minder intensive Informationsphase vorausgeht und (wichtiger noch angesichts der Bedeutung der Bindung bestehender Kunden) eine Nachkaufphase folgt, die ganz andere Informationsbedürfnisse weckt. Aus diesem Grund sollte jeder weitsichtige Website-Betreiber bei seinen Content-Strategien diese unterschiedlichen Kaufphasen berücksichtigen und entsprechend ausgerichtete Content-Pakete schnüren.

6. Eigenproduzierter vs. fremdbezogener Content

   Bis dato war es nackte Notwendigkeit, sich selbst um die Beschaffung, Auswahl und Produktion der Website-Inhalte zu kümmern. Mittlerweile jedoch gibt es erste Dienstleister, die diese Aufgaben übernehmen und sich entsprechend spezialisiert haben. Vorteil dieser Content-Broker ist das relativ große Content-Angebot und die schnelle Verfügbarkeit. Außerdem hat der Inhalte-Händler bereits die Urheberrechtsfrage geklärt, so daß langwierige Verhandlungen entfallen können. Nachteil der Inanspruchnahme derartiger Content-Dienstleister ist der -trotz des großen Angebots- nicht immer optimal zur eigenen Website passende Content. Deshalb können zugekaufter Inhalte stets nur ergänzend zum Einsatz kommen. Es gilt, hier eine vertretbare Mischung zu finden zwischen Content, um den sich der Site-Betreiber selbst gekümmert hat und solchen, der fremdbezogen ist.

7. Anbieterbezogener vs. nicht anbieterbezogener Content

   Es fällt den meisten nach wie vor schwer, neben üblicher anbieterbezogener Eigendarstellung auch den Sinn von nicht anbieterbezogenen Inhalten zu erkennen. Inhalte, die nicht primär den Abverkauf stimulieren sollen, sondern davon losgelöst einen anbieterunabhängigen Mehrwert bieten, können die Glaubwürdigkeit der eigenen Website deutlich erhöhen.

8. Anbietergenerierter vs. nutzergenerierter Content

   Üblicherweise stellt der Anbieter den Content der Website zusammen. Dabei haben nutzergenerierte Inhalte oft einen höheren Stellenwert! Darunter fallen z.B. Beiträge in Diskussionsforen oder Gästebüchern. Viele Besucher werten rege Nutzerbeteiligung als Indiz für die Beliebtheit und Akzeptanz der Site. Wer zudem Nutzer auf der Website unzensiert zu Wort kommen lässt, zeigt, dass er nichts zu verbergen hat und die (potentiellen) Kunden -nicht sich selbst- in den Mittelpunkt stellt. Aus diesen Gründen sollte jede Website über nutzergenerierte Content-Elemente verfügen. Nachdem wir eine Einteilung verschiedener Content-Arten versucht haben, ist es Sache der Site-Betreiber, einen an ihrer Zielsetzung und Zielgruppe ausgerichteten Content-Mix mittels dieser unterschiedlichen Content-Typen zu schaffen.

Am bedeutsamsten für Bezahlinhalte im Internet sind sicherlich die Kategorien 2, 3, 4 und 5.

Die Arten, in denen Content für den Betracher/Nutzer/Kunden aufbereitet wird, unterscheidet man in 4 Bereiche [17], es wird quasi die Form der Internetpräsenz beschrieben:

1. Dynamische Seiten: Seiten, die automatisiert ihren Inhalt ändern.
2. Statische Seiten: Jedes Hypertext-Angebot ist aber zunächst statisch. Es präsentiert sich dem User so lange in derselben Form, bis der Online-Journalist etwas daran ändert.
3. Semidynamische Informationen: Semidynamische Informationen sind eine Mischung aus statischen und dynamischen Elementen. Sie stehen auf Abruf zur Verfügung, werden aber redaktionell aktualisiert.
4. Personalisiert: Das System muss sich die Vorlieben des Users merken und setzt dynamische Konzepte bei der Benutzerverwaltung voraus.

8.4 Pricing Strategies

Grundsätzlich gesehen gibt es verschiedene Vorgehensweisen Preise für "online-content" zu bestimmen [4], [5]. Die Herkunft dieser Strategien ist in der bisherigen Wirtschaft zu suchen, das heißt mit diesen Vorgehensweisen zur Preisbestimmung, wird der Markt für "Online-Contet" zunächst nur als weiteres Handelsfeld betrachet. Die Besonderheiten des Internets müssen später in der Anwendung der Strategien, beziehungsweise in der konkreten Wertermittung von "Online-Content", Beachtung finden.

8.4.1 Nachfrageorientiert

- Penetration Pricing

  Hier wird das Angebot mittels eines zunächst sehr niedrigen Einstiegspreises gestartet. Im Verlauf der Zeit wird der Preis gesteigert bis er ein mittleres Preisniveau erreicht hat. Ziel dieser Strategie ist es: schnell viele Kunden zu gewinnen, wobei der niedrige Anfangspreis als Lockmittel dient. Diese Strategie wird dann angewendet, wenn Anbieter eine schnelle Durchdringung (engl.: penetration) des Marktes mit seinem Produkt erreichen will. Das Risiko für den Anbieter besteht darin, dass die Kunden eine Preissteigerung eventuell nicht mitgehen und das Produkt nicht mehr kaufen. Auch die Amortisierung der Kosten bei der Herstellung und Bereitstellung des Produktes ist durch den niedrigen Preis erst auf längere Sicht zu erreichen.

- Skimming

  Skimming ist eine Hochpreispolitik im Gegensatz zum Penetration Pricing. Hierbei setzt der Anbieter darauf, dass bereits der Verkauf eine geringeren Stückzahl gewinnbringend ist. Voraussetzung hierfür ist aber eine hohe Qualität des Produktes,

so dass die Kunden den Preis auch bereit sind zu zahlen. Der Preis wird im Verlauf der Zeit gesenkt, um weitere Kunden zu gewinnen. Bei dieser Princing Strategie besteht das Risiko in einen zu hohen Preis, der sich in zu geringen Verkaufszahlen niederschlägt. Auch ist der Anreiz für die Konkurenz gross ein vergleichbares Produkt anzubieten, aber etwas billiger, um die Kunden für sich zu gewinnen.

- Teasing

  Eine Strategie, in der die Angebote sehr billig zu erwerben sind. Der Preis soll kein Hinderiss beim Kauf sein. Grund hierfür ist den Kunden billig eine Produktprobe anzubieten. Im weiteren Verlauf wird der Preis erhöht, um wieder gewinnbringend zu verkaufen. Auch hier ist die Gefahr gegeben, dass die Kunden die Produkte nach Preiserhöhung nicht mehr kaufen.

- Trial and Error

  Eine sehr selten angewandte Methode einen Preis festzulegen, da hier eher schwer vorhersagbare statistische Angaben den Preis bestimmen, was mit einem hohen Risiko für den Anbieter einhergeht.

- Prestige-Pricing

  Bei niedrigpreisigen Angeboten gehen viele Nutzer implizit davon aus, daß das Produkt auch von niedriger Qualität ist. Ein höherer Preis kann dieser Annahme entgegenwirken.

- Odd-even Pricing

  Hier wir ein psychologischer Effekt ausgenutzt, bei dem der Käufer Preise wie zum Beispiel 19,95 Euro als spürbar billiger empfindet als 20 Euro. Entsprechend kauft er Produkte mit solchen ungeraden Preisen häufiger. Für den Anbieter stellt solch ein minimal gesenkter Preis oft einen Schlüssel zu höheren Verkaufszahlen dar.

- Demand-Backward Pricing

  Demand -Backward Pricing beruht vollständig auf der Idee des "Willingness to Pay". Also was ist der Kunde bereit für ein bestimmtes Produkt zu Zahlen. Aufgrund solch eines ermittelten Wertes, zum Beispiel durch Umfragen, kann zurückgerechnet werden wie die Gewinnspanne des Produktes liegt. Mit diesem Wert kann der Hersteller entscheiden, ob sich solch ein Produkt finanziell überhaupt lohnen würde. Durch "willingness to pay" kann der Kunde, mittels seiner Zahlungsbereitschaft, indirekten Einfluss auf den Produktpreis nehmen.

- Bundle-Pricing

  Hinter Bundle-pricing verbirgt sich die Idee 2 oder mehr Produkte gesammelt zu verkaufen. Sowohl der Käufer als auch der Anbieter profitieren davon. Der Anbieter muss nur das Bundle und nicht 2 getrennte Produkte vermarkten und der Kunde erhält mit wenig Mehrkosten viel mehr Ware. Entscheidend ist aber die Zusammenstellung der Bundles in Bezug darauf, dass einige der enthaltenden Produkte einen so großen Anreiz bieten sie zu kaufen, dass der Kunde bereit ist, eventuell auch nicht nachgefragte Waren mit zu erwerben.

Die bis hierhin erläuterten Strategien sind ausschließlich nachfrageorientiert, nun folgen einige kostenorientierte und konkurrenzorientierte Methoden zur Preisbestimmung.

8.4.2 Kostenorientiert

- Standard mark-up Pricing

  Ebenfalls sehr weit verbreitet. Das Konzept hinter mark-up pricing ist einfach. Der Hersteller verkauft sein Produkt um einen bestimmten Betrag über dem Herstellungspreis, ein Zwischenhändler schlägt ebenfalls eine Gewinnspanne auf den Weiterverkaufspreis auf und der Händler auch. Dies ist eine sehr einfache Methode für Hersteller und (Zwischen)Händler, jedoch der Kunde bekommt das Produkt nur zu einen immer höheren Preis je mehr Stationen es durchlaufen hat.

- Experience curve pricing

  Das Produkt wir erstmal hergestellt und angeboten, der Herstellungspreis mit der Zeit durch Erfahrungswerte, technischen Fortschritt oder ähnlichem jedoch gesenkt. Diese Einsparungen können über einen sinkenden Preis an die Kunden vermittelt werden.

8.4.3 Konkurenzorientiert

- Customary Pricing

  Eine Art Festpreis. Der Kunde kann immer und überall sein Produkt erwerben und ist sicher, dass er es für den gleichen Preis bekommt.

- Above-, at-,below- market pricing

  Die Idee hier ist seine Güter/Waren teurer, gleichpreisig oder niedrigpreisiger als die Konkurenz anzubieten. Ein höherer Preis soll auf den Bereich der Luxusware abzielen, ein kleinerer Preis soll durch größerer Absatz gewinnbringend sein und der gleiche Preis soll ein Stück des Marktes für den Hersteller erschließen.

- Loss-leader pricing

  Einige Produkte werden unter Wert verkauft und dienen als Lockmittel für Kunden in der Hoffnung diese Kaufen gleichzeitig andere Produkte. Für Kunden, die nach billigen Waren suchen, sehr vielversprechend, der Händler trägt aber das Risiko zuviel Geld an den unter Wert angebotenen Waren zu verlieren, als im der zusätzliche Verkauf von weiteren Produkten einbringt.

- Flexible-pricing [22]

  Der Preis ist Veränderungen unterworfen. Gleichzeitige Anfragen an das System generieren gleiche Preise, aber wenn über den Zeitverlauf die Systemlast schwankt, ändert sich der Preis. Auch andere externe Faktoren beeinflussen den Preis. Flexible-Pricing findet auch Anwendung um die Kundenakzeptanz gegenüber bestimmten Preisspannen zu bestimmen.

- Static Pricing [22]

  Festpreise, die nur selten geändert werden. Sie sind von Kunden am meisten akzeptiert. Sie ermöglichen ihnen eine Vergleichsmöglichkeit zwischen verschiedenen Anbietern. Jedoch sind Fixpreise schwer bestimmbar, da die Bereitschaft Geld auszugeben von Kunde zu Kunde, zeitlich und in Bezug auf die finanziellen Möglichkeiten schwankt.

- Discriminatory Pricing [22]

  Verschiedenen Kunden das gleiche Produkt zu unterschiedlichen Preisen anbieten. Grundlage des Preises stellt ein Kundenprofil dar. Je nachdem ob wie gut dieses Profil erstellt und ausgewertet wurde, desto besser wird der Kunde den Preis akzeptieren. Kunden die für gleiche Produkte mehr zahlen, können sich jedoch ungerecht behandelt fühlen.

8.5 Wertermittlung von Online Content

Im letzten Abschnitt wurden allgemeine wirtschaftliche Aspekte und Strategien die auf den Warenpreis wirken angesprochen. In diesem Abschnitt werden spezielle Gegebenheiten des Internet-Handels/Marktes betrachtet und Faktoren aufgeführt, die den Preis von "Online-Content" maßgeblich beeinflussen.

Die Preisermittlung beruht im wesentlichen auf fünf Kriterien:

1. nach dem Nutzermehrwert,
2. nach dem Preisniveau der Konkurrenz,
3. aus Controllinggesichtspunkten, d.h. Kosten plus Marge,
4. nach "Bauchgefühl" bzw. "Trial & Error" oder
5. auf Basis von Marktforschung.

Die Orientierung am Nutzermehrwert erfolgt, indem der Preis im Hinblick auf den wahrgenommenen Wert abzüglich der mit der Nutzung verbundenen Kosten ermittelt wird. Die entscheidenden Faktoren dieser Preisfindung ergeben sich aus dem Mehrwertfaktor und den Kosten, die dem Nutzer entstehen. Voraussetzung hierfür ist eine profunde Kenntnis des Nutzerverhaltens: Unterschiedliche Kunden ziehen aus demselben Produkt verschiedene Vorteile und sind daher bereit, unterschiedliche Preise zu bezahlen. Im Gegensatz dazu wird bei der konkurrenz-orientierten Preisermittlung der Fokus auf die Wettbewerber gelegt. Die Preisfindung errechnet sich aus dem Durchschnitt der Bepreisung der jeweiligen Wettbewerber. Das dritthäufigste Verfahren bei der Preisermittlung ist der Kosten-plus-Marge-Ansatz (Cost-Plus). Basis dieses Verfahrens ist die Festlegung der Marge, die addiert zu den Kosten des Produktes, dessen Profitabilität ergibt. "Bauchgefühl" (Trial & Error) und Marktforschung spielen nach Auskunft der Anbieter kostenpflichtiger Angebote eine eher untergeordnete Rolle. Dass Marktforschung bisher bei der Preisermittlung

von Kostenpflichtigen Inhalten nicht häufiger eingesetzt wurde, wie in anderen etablierten Produktkategorien schon lange üblich, liegt vermutlich daran, dass der Markt für kostenpflichtige Inhalte noch im Aufbau ist [5]. In Bereich der Controllinggesichtspunkte fallen die speziellen Gegebenheiten von PaidContent:

- Herstellungskosten für "online-content"
  Kosten für Recherchieren, Erstellen, Vermarktung. Eine Besonderheit im "online-content" stellt dar, dass die erste Ausfertigung den größten Teil des Herstellungspreises ausmacht, jede weitere (Daten)Kopie nur noch relativ geringen finanziellen Aufwand bedeutet. Einträge in Suchmaschinen oder ähnliche Datenbanken sind für Internetauftritte unerlässlich, jedoch meist kostenpflichtig.

- Redaktionelle und Laufende Kosten
  Warten und Aktualisieren des Internetauftrittes und Einpflegen neuer Angebote, sowie Werbung im Internet fallen in diese Kategorie.

- Providerkosten
  Kosten für den Webspace, Content-Management und speziell Traffic wenn der verkaufte Content ein hohes Datentransfervolumen herbeiführt.

- Kosten für Distributionssoftware
  Wenn der Vertrieb über ein eigens entwickeltes Onlineportal oder eine extra Software läuft. Hier wären als Beispiel "iTunes" und "Steam" zu nennen [7].

- Format von Content
  In welchem Dokumenten-, Audio-, Grafik- oder Videoformat die Bereitstellung von Content erfolgt. Zu berücksichtigen ist hier: nicht jedermann kann jedes Format nutzen. Zudem ist Software zum verarbeiten/erstellen oft kostenpflichtig [19].

Alle diese Einzelkriterien müssen bei der Bepreisung von "online-content" berücksichtigt werden. Nun soll hier eine kurze Darstellung über mögliche Zuordungen von Content zu bestimmten Preisstrategien erfolgen (s. Kap. 8.4.1, Kap. 8.3, [15]).

Klassen von "online-content"	Preis-/E-Businessstrategie
Standard-Content	meist kostenfrei, Möglichkeit zur Werbe- oder Sponsorenfinanzierung
exklusiver Content	Skimming, Prestige-Pricing oder Above-market-Pricing
nicht exklusiver Content	Penetration-Pricing
Multimedia Content	Penetration-Pricing, Teasing, Flexible-Pricing
Text	Teasing, Standard mark-up Pricing
lange aktueller Content	Standard mark-up Pricing, Customary Pricing, Skimming
kurz aktueller Content	Below-market-Pricing, Teasing

Diese Übersicht stellt wie gesagt nur eine Möglichkeit dar. Welche Preisstrategie(n) (s. Kap. 8.4.1) der Anbieter letztendlich anwendet ist dann seine Entscheidung, dafür kann zur Zeit keine allgemeingültige Aussage getroffen werden. Hier wäre die Möglichkeit gegeben Studien/Umfragen durchzuführen, um Erkenntnisse zu gewinnen, über welche Preisstrategie(n) sich ganz bestimmter Content am besten verkauft.

8.6 Bedarf und Zahlungsbereitschaft im Bereich Online Content

Welche Güter und Informationen fragen Kunden überhaupt nach? Ist es technisch auch möglich diese Über das Internet anzubieten? Mit diesen Fragen beschäftigt sich der folgende Abschnitt.

Die Bereitschaft, für Internetinhalte zu bezahlen, wächst beständig. Bereits 66 Prozent der regelmäßigen Internetnutzer können sich vorstellen, für bestimmte Online-Inhalte Geld zu bezahlen. Am höchsten ist die Zahlungsbereitschaft bei Online-Spielen bzw. Spieleplattformen (37 Prozent), gefolgt von Musikdiensten (35 Prozent), Videodiensten (32 Prozent) und Informationsdiensten (31 Prozent), so die Ergebnisse einer Befragung des Marktforschungsunternehmens Smart-Research [6].

Dieses deckt sich auch nahezu vollständig mit einer weiteren Umfrage von "w&v online" [9]. Hier erkennt man erstens welche Waren nachgefragt werden und ob Kunden bereit sind für dies zu zahlen. Ob eine Zahlungsbereitschaft für bestimmte Artikel besteht ist sehr wichtig. Denn wenn niemand bereit ist Geld dafür auszugeben, wird solch ein Gut keinen Absatzmarkt finden uns somit auch keinen Gewinn für den Anbieter erwirtschaften.

"Willingness to pay"(Zahlungsbereitschaft) ist von Kunde zu Kunde individuell verschieden und auch der spezifische Nutzen des einzelnen Angebots wird von Fall zu Fall erneut bewertet. Die Grafik 8.1 gibt eine Einblick wie ein potentieller Kunde den Nutzen eines Angebotes im Vergleich zum Preis und finanziellen Aufwand einschätzt [20].

In der zweiten Formel wird die bereits erworbene Erkenntnis zwischen Preis/Kosten/Nutzen in Bezug zum Kauf bei einem Konkurenzanbieter gesetzt [20].

Abbildung 8.1: Nutzen eines Angebotes im Vergleich zum Preis und finanziellen Aufwand

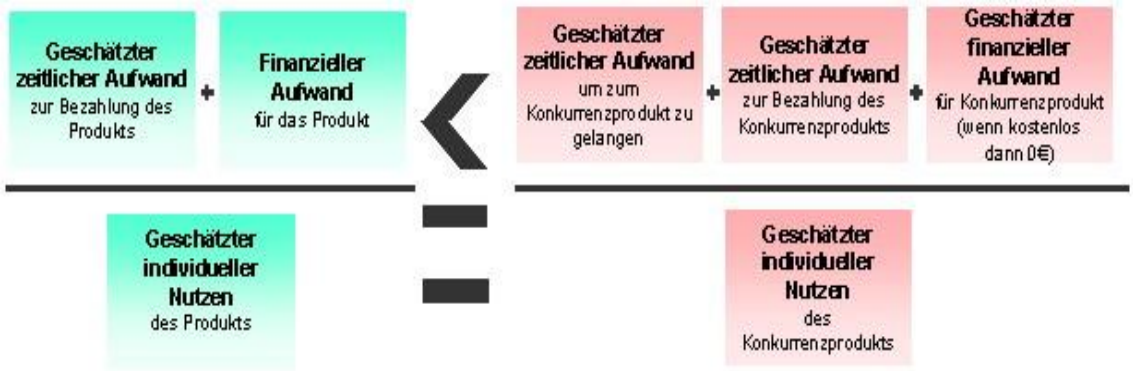

Abbildung 8.2: Bezug von Preis/Kosten/Nutzen bei Kauf von Konkurenzanbieter

Hingewiesen sei hier auf eine Beispiele wie: der aktuelle Wetterbericht, Nachrichtenticker oder Sportereignisse. Diese Beispiele sind in so vielfältiger und kostenlos verfügbar, dass die Bereitschaft Geld dafür zu zahlen, in dem er sie über Internet erwirbt, wahrscheinlich gering ist. Ein Trend jedoch in Sachen: "Was möchte ich über das Internet käuflich erwerben" ist zweifelsohne Multimedia (Musik, Filme) und Unterhaltungssoftware [7] oder kurz der gesamte Bereich Freizeit und Unterhaltung.

Die folgender Grafik beschreibt zwar nur den Bereich des "additional content", also zusätzlich auf einer Webseite untergebrachte Inhalte, aber auch hier ist der Zuspruch für Unterhaltung am größten [21].

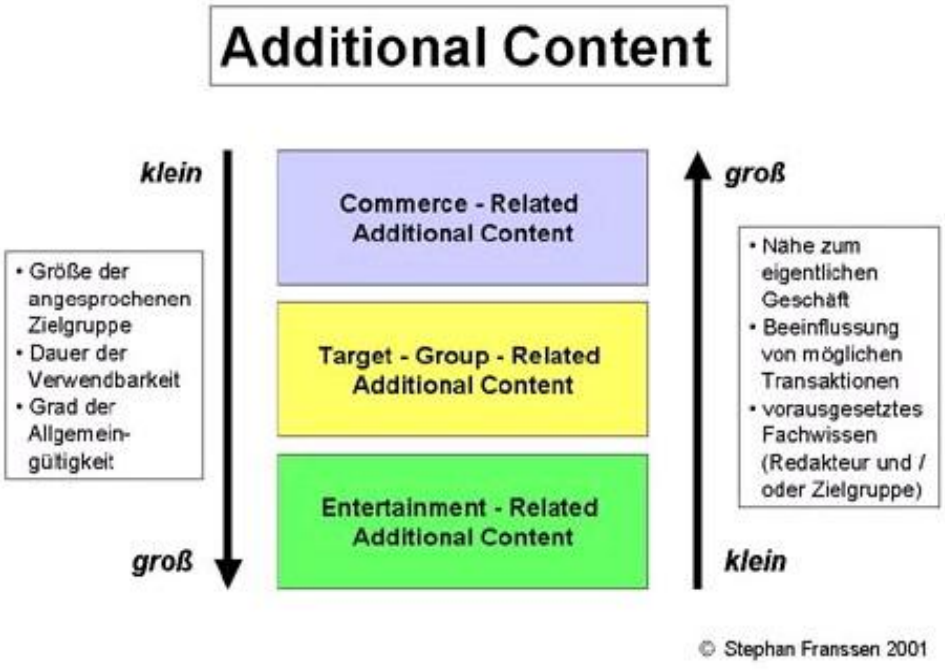

Abbildung 8.3: Eigenschaften von Additional Content

Das ist wie gesagt nur ein Trend, ein anderer geht in Richtung Electonic Libraries, "Content-Aggregators", Content-Makler und elektronischen Diensten, die den bisher größten Teil kostenpflichtiger Internetnutzung ausmachten [8], [10].

Solcherlei Electronic Libraries sind an Universitäten (zum Beispiel "University of Toronto") oder anderen Forschungseinrichtungen sehr gefragt. Die zur Zeit führenden Anbieter beziehungsweise "Content-Aggregators" sind SCRAN, AMICO und JSTOR sowie OnDisC [11], [12]. In den Bereich Content-Makler fallen Portalseiten und Suchmaschinen.

8.7 E-Business und Abrechnungssysteme

Da nicht nur mit Waren oder Dienstleistungen Geld über oder im Internet erwirtschaftet wird gelangen nun Konzepte des E-Business zur Übersicht.

- Brokerage
  Das Konzept beschreibt die Zusammenführung von Käufer und Verkäufer in einer Art Auktionshaus. Einnahmen werden mittels Erhebung einer Auktionsgebühr erwirtschaftet.

- Advertising
  Eine erste Webseite enthält Werbebanner, die bei Anklicken auf die Webseite des Werbenden führen. Hierbei werden pro Klick auf das Werbebanner, und dem somit erfolgenden Aufruf der Webseite des Werbestellers, Gebühren an den Besitzer der ersten Webseite gezahlt. Auch in Form von "pop-ups" vorhanden.

- Infomediary und Data-Mining
  In diesem Modell wird darauf abgezielt, Informationen über den Nutzer zu erlangen, zum Beispiel seine aufgerufenen Webseiten, seine bevorzugten Interessen oder Kaufgewohnheiten. Diese Informationen werden aufgearbeitet und verkauft.

- Mercant
  Hierbei werden Güter oder Informationen über eine Art Webshop beziehungsweise Online-Katalog verkauft.

- Manufacturer
  Ähnlich der Mercant-konzept, aber hier bietet ein Hersteller seine Waren direkt an, im Gegensatz zum Merchant, der einen Zwischenhändler mit meist sehr vielfältiger Warenpalette darstellt.

- Affiliate
  Ein Vertriebsmodell, in dem über eine Weiterleitung oder ein "pop-up" u.ä. der Nutzer von der zuerst aufgerufenen Webseite auf Webshops (siehe Merchant oder Manufacturer) gebracht werden. Vom Kauf von Waren in diesen Webshops erhält der Betreiber der Webseite, von der der Käufer zum Webshop geleitet wurde, einen Teil des Gewinns aus dem Verkauf.

- Community
  Die Webseite der Community (Netzgemeinschaft) wird über Spenden, entweder ihrer Mitglieder oder von Sponsoren, finanziert.

- Subscription
  Hier wird der Zutritt und Zugriff zu Webseiten kostenpflichtig gemacht. Oft verbunden mit freien Inhalten, die eventuell nicht Vollständig sind oder nicht wertvoll genug sind, um verkauft zu werden.

- Utility
  Ein Modell, in dem "micro-payment" [13] eine große Rolle spielt. Der Nutzer zahlt nur für das, was er auch wirklich sehen will. Das heißt er zahlt getrennt für jede Webseite oder Information erst beim Aufruf. Stellt ein Gegenteil zu Bundling dar, in dem vom Kunden schon vorgefertigte Pakete angeboten werden.

Affiliate, Community, Infomediary, Mercant, Manufacturer und Brokerage ist eines gemeinsam. Die Webseite selbst generiert kein Einkommen, sondern dieses wird sekundär erwirtschaftet. Zum einen durch über oder innerhalb der Webseite angebotene Waren, oder über Aufruf der Webseite ermittelte Informationen, beziehungsweise Spenden. Die Kombination von zwei oder mehr E-Business-Konzepten ist die Regel.

Im Wirtfschaftsfeld des E-Business sind technische Aspekte im Bereich Abrechnung, Bereitstellung und Distribution von grundlegendem Charakter. Daher hier der Anriß von Abrechnungs- und Distributionssystemen. Einige kleine Übersichten finden sich auf diesen Webseiten [14], [16]. Natürlich ist auch die Form, in der Content, abgerechnet wird vielfältig: Datengröße, Anzahl, Pay-per-View, Qualität, Zeit [16].

8.8 Zusammenfassung

Als Ergebnis dieser Arbeit ist zu sagen, dass zum Aktuellen Zeitpunkt noch keine allgemeingültige Strategie zur Bepreisung von "online-content" existiert. Als schematische Zusammenfassung dient die nachstehende Grafik.

Abbildung 8.4: Schema zur ständigen Preisermittlung und Überprüfung

Als Trend festzustellen ist jedoch eine Preissenkung gegenüber der Bepreisung bei Vertrieb über herkömmliche Distributionswege, was aber größere Preisschwankungen für gleiche und ähnliche Produkte nicht ausschließt. Eine Untersuchung mit dem Ziel Erkenntnisse zu gewinnen, über welche Preisstrategie(n) sich ganz bestimmter Content am besten verkauft erscheint sinnvoll.

Literaturverzeichnis

[1] Eino Kivisaari, Sakari Lukkainen: Content-Based Pricing in the mobile Internet; 2004, http://www.tml.hut.fi/Opinnot/T-109.551/2004/Content_based_pricing.pdf

[2] Dr. Karl-Michael Henneking: Optimizing Mobile Data and Content Pricing; CeBIT März 2004, http://www.detecon.com/media/pdf/Cebit2004OptimisingMobileDataAndContentPricing.pdf

[3] John Chung-I Chuang, Marvin A. Sirbu: Optimal Strategie for Digital Information Goods: Network Delivery of Articles an Subscriptions; http://www.sims.berkeley.edu/~chuang/pubs/ediip2.pdf

[4] Dr. Ralph F. Wilson: Pricing Strategy as Part of Your Internet Marketing Plan; Mai 2000, http://www.wilsonweb.com/wmt5/plan-pricing.htm

[5] Alexander v.Reibnitz, Arndt Rautenberg, Manfred Schwaiger: Pricing von Paid Content - Vorgehen bei der Preisfindung und gegenwärtiges Preisniveau; September 2003, http://www.contentmanager.de/magazin/artikel_380_pricing_paid_content.html

[6] www.golem.de: Studie: Zahlungsbereitschaft für Online-Content; Februar 2003, http://www.golem.de/0302/23802.html

[7] Valve Corporation, Apple Computer Inc.: Steam und iTunes; http://www.steampowered.com/ und http://www.apple.com/de/itunes/

[8] Verband Deutscher Zeitschriftenverleger e.V: VDZ: Pricing von Paid Content und Paid Services; http://www.wuv.de/daten/studien/062003/755/

[9] Verband Deutscher Zeitschriftenverleger e.V: VDZ: Pricing von Paid Content und Paid Services; http://www.wuv.de/daten/studien/062003/755/2390.html

[10] Verband Deutscher Zeitschriftenverleger e.V: VDZ: Pricing von Paid Content und Paid Services; http://www.wuv.de/daten/studien/062003/755/2389.html

[11] OnDisC Alliance, SCRAN Online, The Art Museum Image Consortium, The Scholarly Journal Storage: Hompages von "Content-Aggregators", http://www.ondisc.ca/, http://www.scran.ac.uk/, http://www.amico.org/, http://www.jstor.org/

[12] Albert W. Darimont: Environmantal Scan of Pricing Models for Online Content; November 2001, http://dlist.sir.arizona.edu/archive/00000204/01/phase1.pdf

[13] WordWideWebConsortium: Micro Payment Transfer Protocol (MPTP) Version 0.1; November 1995, http://www.w3.org/TR/WD-mptp-951122

[14] at-mix.de, Berliner Arbeitskreis Information: Überblick über Zahlunssysteme im Bereich Mobile Data und Online Content; http://www.at-mix.de/zahlungssysteme.htm und http://bak-information.ub.tu-berlin.de/software/e_comm.html

[15] Martin Oelbermann, Alexander von Reibnitz: Paid Content - Der Markt für Online Inhalte, http://www.businessvillage.de/Magazin/mag_detail/mag-92_Paid_Content_%96_Der_Markt_fuer_Online_Inhalte.html

[16] Christian Breunig: Internet: Auf dem Weg zu einem kommerziellen Medium; August 2003, http://www.ecc-handel.de/erkenntnisse/1030694998/

[17] online-journalismus.org: Beispiele und aktuelle Ergänzungen zum Buch "Online-Journalismus", http://www.online-journalismus.org/polyphem.php?nal=nlbecon_art& nao=nobe&hau=./beruf/contorg/contart

[18] explido Webmarketing: Content is King; 2004, http://www.promotionwelt.de/marketingmix_content.htm

[19] Luxem, Redmer: Digital Commerce. Electronic Commerce mit digitalen Produkten; Köln 2000, http://www.ecc-handel.de/branchen_prob/1074781498/?show_id=1074782540

[20] Hannes Fehr: Paid Content erfolgreich verkaufen; http://www.ecin.de/zahlungssysteme/paidcontent/

[21] Stephan Franssen: Klassifizierung von Additional Content; September 2001, http://www.contentmanager.de/magazin/ artikel_82_klassifizierung_von_additional_content.html

[22] B.Stiller, K.Almeroth, J.Altman, L.McKnight, M.Ott: Content Pricing in the Internet; ITCom2002, http://imj.ucsb.edu/papers/COMCOM-04b.pdf.gz

[23] Dr.Wolfgang Semar: 10 Erfolgsfaktoren im E-Business / E-Commerce; http://www.inf-wiss.uni-konstanz.de/CURR/winter0102/li/erfolgsfaktoren.pdf

Kapitel 9

Handoff Efficiency in Mobile IPv6 Scenarios

Robin Cronin

Due to the shortage of IPv4 addresses, it is merely a matter of time until IPv6 will replace its predecessor. This means that mobile node will have to migrate from Mobile IPv4 to Mobile IPv6. Although this new standard brings along some great advantages, users or even network administrators may choose a specific implementation that may not be the best concerning their needs.
The first part of this work provides the reader with general information about wireless networking and common technologies. Because it is in the nature of mobile nodes to move around, handoffs play a decisive role in a mobile world, as it will be shown chapter 2. Subsequently, the third part points out the main characteristics of Mobile IPv6 and its most promising enhancements. Merging the gathered knowledge, chapter 4 compares Mobile IPv6 and its enhancements in terms of handoff efficiency. Finally, some useful tips and ideas, based on observations made in the previous chapter, are offered in the conclusion.

Inhaltsverzeichnis

9.1 Introduction

9.1.1 Motivation

In a world of information and individual mobility, wireless communication becomes more and more important. Because these portable terminals that are used, such as cell phones or notebooks, are limited in transmission range and coverage by their specific network technologies, they need to change connectivity between base stations when moving out of their current cell. Because ongoing and seamless communication is desirable, this process, also called a **handoff**, should be as unobtrusive and **efficient** as possible.

Most wireless networks that identify their nodes by IP addresses still use Mobile IPv4 (MIPv4), which relies on IPv4 with a total of 4 billion addresses. But because of the approaching shortage of available IPv4 addresses, the need for IPv6, which supports up to 3.4 x 10^{38} addresses, becomes obvious. To enable mobility in this new network, **Mobile IPv6** (MIPv6) has been introduced.

The main focus of this work is on the performance of this protocol and its enhancements, especially during handoffs.

9.1.2 Definitions

Some facts and terms relating to wireless networking are assumed to be known. The following table describes some of them as they are defined in [10] and [14]; others will be explained later in this work.

1. General Terms

IP	Internet Protocol
node	A device that implements IP.
router	A node that forwards IP packets not explicitly addressed to itself.
link	A communication facility or medium over which nodes can communicate at the link layer, such as an Ethernet (simple or bridged). A link is the layer immediately below IP.
interface	A node's attachment to a link.
subnet prefix	A bit string that consists of some number of initial bits of an IP address.
interface identifier	A number used to identify a node's interface on a link. The interface identifier is the remaining low-order bits in the node's IP address after the subnet prefix.
packet	An IP header plus payload.

2. Mobile IPv6 Terms

mobile node (MN)	A node that can change its point of attachment from one link to another, while still being reachable via its home address.
correspondent node (CN)	A peer node with which a mobile node is communicating. The correspondent node may be either mobile or stationary.
access router (AR)	The mobile node's default router. The access router aggregates the outbound traffic of mobile nodes.
access point (AP)	A Layer 2 device connected to a subnet that offers wireless connectivity to a mobile node.
home agent (HA)	A router on a mobile node's home link with which the mobile node has registered its current care-of address. While the mobile node is away from home, the home agent intercepts packets on the home link destined to the mobile node's home address, encapsulates them, and tunnels them to the mobile node's registered care-of address.
home address	A unicast routable address assigned to a mobile node, used as the permanent address of the mobile node. This address is within the mobile node's home link. Standard IP routing mechanisms will deliver packets destined for a mobile node's home address to its home link. Mobile nodes can have multiple home addresses, for instance when there are multiple home prefixes on the home link.
home subnet prefix	The IP subnet prefix corresponding to a mobile node's home address.
home link	The link on which a mobile node's home subnet prefix is defined.
foreign subnet prefix	Any IP subnet prefix other than the mobile node's home subnet prefix.
foreign link	Any link other than the mobile node's home link.
care-of address (CoA)	A unicast routable address associated with a mobile node while visiting a foreign link; the subnet prefix of this IP address is a foreign subnet prefix. Among the multiple care-of addresses that a mobile node may have at any given time (e.g., with different subnet prefixes), the one registered with the mobile node's home agent for a given home address is called its „primary“ care-of address.

binding	The association of the home address of a mobile node with a care-of address for that mobile node, along with the remaining lifetime of that association.
movement	A change in a mobile node's point of attachment to the Internet such that it is no longer connected to the same link as it was previously. If a mobile node is not currently attached to its home link, the mobile node is said to be „away from home“.

9.1.3 Wireless Network Technologies

Although this work has its focus on Layer 3 events, it is important to understand what is meant by „wireless network technology“. Because MIPv6 (at Layer 3) is depending on and interacting with its underlying layers, the most important are briefly mentioned in this section.

Figure 9.1 shows application areas and bandwith capacities of different (wired and wireless) network technologies.

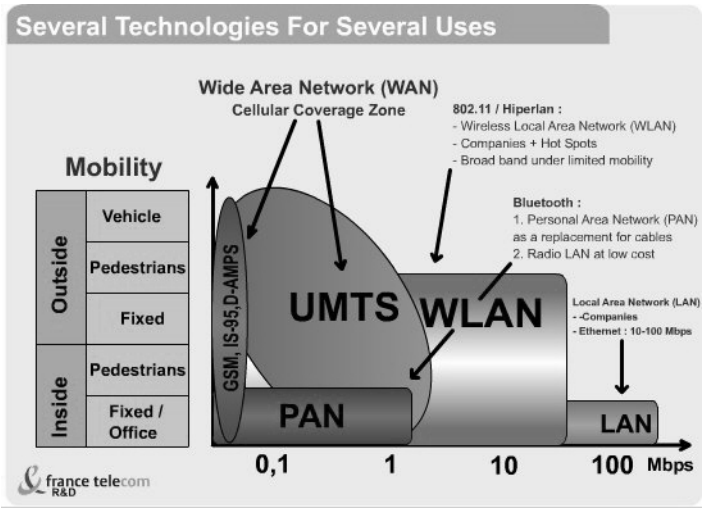

Abbildung 9.1: Several Technologies For Several Uses [6]

The most important wireless network technologies:

1. Wireless Local Area Network (WLAN)

   The WLAN technology is defined by the IEEE 802.11 specifications. The first 802.11 (aka. 802.11y or 802.11legacy) standard introduced physical and Media Access Control (MAC) specifications for wireless networks. By now, three following standards (802.11a, 802.11b, 802.11g) enhanced the physical layer specifications of WLAN.

Other versions (802.11c-f, 802.11h-j, 802.11n) may be considered as service upgrades or corrections. All 802.11 standards implement Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA) mechanisms for media access.

2. Global System for Mobile Communications (GSM)

   GSM is a digital cell phone standard that employs Time Division Multiple Access (TDMA) between stations on a frequency duplex pair of radio channels, with slow frequency hopping between (eight logical) channels. A GSM transmission is either a voice call or a data call. The former is limited to 13 kBit/s while using speech compression codecs (Half Rate, Full Rate, Enhanced Full Rate, Adaptive Multi Rate); the latter has a modem-like 9.6 kBit/s bandwidth (Circuit Switched Data).

3. General Packet Radio Service (GPRS)

   GPRS is a 2.5G standard for digital cell phone networks. It is an advancement of GSM to support packet based communications while bundling the 8 GSM channels. GPRS is packet-switched, which means that the data connection is not used if no data is being transmitted or received. So, it is only compatible with GSM networks.

4. Universal Mobile Telecommunications System (UMTS)

   UMTS is a 3G cell phone technology, based on the Wideband-Code Division Multiple Access (W-CDMA) air interface and some GSM specifications like speech codecs. Data transfer rates in UMTS are about 2 MBit/s, but may be extended to up to 10 MBit/s by using the High Speed Downlink Packet Access (HSDPA) enhancements.

The following table [5],[7] gives an overview of data rates and transmission ranges of the wireless network technologies mentioned above:

Technology	**Data Rate**	**Range**
802.11b	11 MBit/s	150 m
802.11a	54 MBit/s	150 m
GSM	13 kbit/s (voice)	50 km
	9,6 kbit/s (data)	50 km
GPRS	171 kbit/s	50 km
UMTS	144 kbit/s (World Cell)	> 20 km
	144 kbit/s (Macro Cell)	20 km
	384 kbit/s (Micro Cell)	300 m
	2 MBit/s (Pico Cell)	50 m

9.2 Handoffs

In terms of wireless networking, a handoff is the movement of a MN between different points of attachment to a network. If radio transmission is available, a handoff describes the process of a MN changing its connection from one particular base station to another one.

9.2.1 Handoff Types

1. Horizontal Handoffs

   If the handoff takes place between base stations of the same type of network technology, it is called a horizontal handoff.

2. Vertical Handoffs

   Handoffs between base stations that are using different wireless network technologies are referred to as vertical handoff.

   A handoff from a network of small coverage (e.g., WLAN) to a network of higher coverage (e.g., GPRS) is called an upward vertical handoff. In contrast to the upward vertical handoff, the movement to a link of smaller coverage is known as a downward vertical handoff.

9.2.2 Handoff Steps

Handoffs are devided into two main steps: the handoff decision and the handoff execution [1].

Handoff Decision

The decision to whether or not to perform a handoff can be made by the MN, the network, or even in cooperation [18]. In horizontal handoffs, the decision is often determined by signal strength. Whereas in vertical handoffs, several reasons can be the crucial factors. Motives for deciding a handoff (horizontal and vertical) may be bandwidth, cost, security, Quality of Service (QoS), etc [19].

Handoff Execution

When the MN is under coverage of a new base station, and already decided to handoff to, the handoff execution process is going to take place. It is devided into three steps: detection, configuration, and registration [1].

1. Detection

   The first step in the handoff execution process is the detection of the new link. A MN recognizes the existence of a new link when it obtains basic information sent by the corresponding base station. A MN may perform active scans/requests or just wait to receive those technical details.

2. Configuration

   In this step, the MN configures its mobile interface based on the information given by the base station in the previous step, and updates its network setting.

3. Registration

   During the registration step, the MN informs its HA and/or CNs of its current location. Registration is completed when the MN has received all confirmations, thus being able to continue to communicate with its HA/CNs.

9.3 Mobile IPv6

As already mentioned in the introdution, MIPv6 is a network layer protocol that extends IPv6 by mobility support, so MNs remain reachable in IPv6 networks [10]. But before getting any deeper into MIPv6, the protocol it relies on, IPv6, will be introduced first.

9.3.1 Internet Protocol, Version 6

Because of the enormous complexity of IPv6, this work will only highlight aspects that are of particular importance to MIPv6.

1. The IPv6 Address

   While IPv4 addresses were limited by 32 bits, IPv6 introduces 128-bit addresses [4]. The preferred form of textual representation is a hexadecimal format, devided into eight 16-bit pieces that are seperated by colons.
   `87F8:0000:0000:0450:A408:0128:0000`

   Leading zeros can be omitted, but there has to be at least one number in every field.
   `87F8:0:0:450:A408:128:0:0`

   `::` may be used to represent one or more 16-bit fields of zeros.
   `87F8::450:A408:128:0:0`
   `87F8:0:0:450:A408:128::`
   `0:0:0:0:0:0:0:0 -> ::`
   `0:0:0:0:0:0:0:1 -> ::1`

   The `::` must not appear more than once in an address, because of the possibility of misinterpretation [8].

2. The IPv6 Header

   According to the specifications made in [4], the IPv6 header contains the following fields:

Version	4-bit Internet Protocol version number = 6.
Traffic Class	8-bit traffic class field.
Flow Label	20-bit flow label.
Payload Length	16-bit unsigned integer. Length of the IPv6 payload, i.e., the rest of the packet following this IPv6 header, in octets.

Next Header	8-bit selector. Identifies the type of header immediately following the IPv6 header. Uses the same values as the IPv4 Protocol field.
Hop Limit	8-bit unsigned integer. Decremented by 1 by each node that forwards the packet. The packet is discarded if Hop Limit is decremented to zero.
Source Address (Src)	128-bit address of the originator of the packet.
Destination Address (Dst)	128-bit address of the intended recipient of the packet (possibly not the ultimate recipient, if a Routing header is present).

Note that the traffic class field may be used to define packet priority, similar to Type of Service in IPv4. The flow label can be used for flow identification, thus providing QoS.

Two IPv6 extension headers are of greater interest to MIPv6:

- The routing header contains a list of IPv6 addresses. When a node (Dst in the main header) receives this packet, it swaps the Dst with the *next address* in the routing header and reduces *segments left* by 1. This number is used to identify the current *next address* within the address list. Then the packet is forwarded to the new Dst. If the *segments left* is 0, there is no *next address* left, so the final receiver processes the next header.
- The destination options header (DOH) informs the node, to which the packet is destined to, about special options. MIPv6 uses this extension header, as it will be shown in *3.2.3 Moving to a Foreign Link* and *3.2.4 Route Optimization*.

3. Neighbor Discovery

   On multiple access links, nodes can communicate directly with other nodes that are sharing the same link, but only if they know each other's MAC address. Neighbor Discovery [13] is quite similar to the Address Resolution Protocol (ARP) in IPv4, but includes several new functions, and relies on ICMPv6 (Internet Control Message Protocol for IPv6) messaging [2]. To discover at least one default router, a host sends a router solicitation (RtSol) to the all-routers multicast address; when received by a router, it will be answered by a router advertisement (RtAdv). This message not only informs the host about the router's IPv6 and link-layer addresses, but also the subnet prefix, the prefix length, and Maximum Transmission Unit (MTU). The RtAdv may be sent unsolicited to the all-nodes multicast address.

   Neighbor solicitation messages are used for address resolution, Duplicate Address Detection (DAD), and Neighbor Unreachability Detection (NUD). The response messages are called neighbor advertisements (NA) [3],[8],[13].

9.3.2 Overview of Mobile IPv6

MIPv6 introduces methods and specifications that provide (moving) MNs with reachability in IPv6 networks [10]. Because MIPv6 is compatible to standard IPv6, stationary nodes do not need to know of IP mobility of MNs.

The following list [10] highlights the major differences between MIPv4 and MIPv6:

- There is no need to deploy special routers as *foreign agents*, as in MIPv4. MIPv6 operates in any location without any special support required from the local router.
- Support for route optimization is a fundamental part of the protocol, rather than a nonstandard set of extensions.
- MIPv6 route optimization can operate securely even without pre-arranged security associations. It is expected that route optimization can be deployed on a global scale between all MNs and CNs.
- Support is also integrated into MIPv6 for allowing route optimization to coexist efficiently with routers that perform *ingress filtering*.
- The IPv6 NUD assures symmetric reachability between the MN and its default router in the current location.
- Most packets sent to a MN while away from home in MIPv6 are sent using an IPv6 routing header rather than IP encapsulation, reducing the amount of resulting overhead compared to MIPv4.
- MIPv6 is decoupled from any particular link layer, as it uses IPv6 Neighbor Discovery instead of ARP. This also improves the robustness of the protocol.
- The use of IPv6 encapsulation (and the routing header) removes the need in MIPv6 to manage *tunnel soft state*.
- The dynamic home agent address discovery mechanism in MIPv6 returns a single reply to the MN. The directed broadcast approach used in IPv4 returns separate replies from each HA.

Home Link

Like any other node in the IPv6 Internet, a MN is identified by a unique IPv6 address. Because this address is formed on the MN's home link, it is called the home address. On the home link, a MN acts like any other IPv6 node. All packets that are sent or received by a MN are routed just like packets between other IPv6 nodes [4],[10].

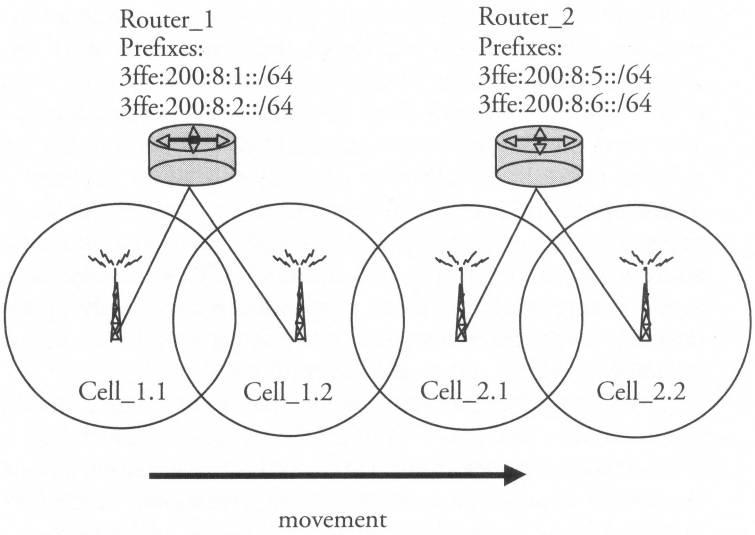

Abbildung 9.2: Layer 2 vs. Layer 3 Handoffs [15]

IP Mobility

In chapter *2 Handoffs*, a change of base stations (i.e., APs) was defined as handoff. This does not necessarily imply that the MN has moved in the IPv6 topology.

A handoff, as shown in Figure 9.2, from Cell_1.1 to Cell_1.2 does not result in a movement in the IPv6 topology. Although the MN has moved, it only changed its AP, not its router. Because both APs are associated with Router_1, the MN's IPv6 address is still valid - the subnet prefix has not changed. The handoff has only taken place on the MN's radio link layer.

IP mobility is only invoked when a MN moves to a new subnet. Therefore, a handoff from Cell_1.2 to Cell_2.1 would not only cause a Layer 2 handoff, but also a Layer 3 handoff because the current IPv6 address is not valid on the new link due to the different subnet prefix of Router_2.

As it was mentioned in *3.2.1 Home Link*, the MN has an IPv6 address that is used for Layer 3 connections. Assuming Router_1 is the MN's HA, the MN's home address would be a valid IPv6 address for communication on its home link. However, on the new link of Router_2 it is topologically incorrect.

Just changing the communicating address, the home address, into a valid IPv6 address would cause severe IP mobility issues:

- On the one hand, the MN cannot use its home address on a foreign link because it is topologically incorrect due to the different subnet prefix. Therefore, the MN has to change its communicating address.
- On the other hand, a stable home address is mandatory. The MN's upper layers identify the MN by its home address and do not accept other ones. Furthermore,

other nodes (i.e., CNs) should be able to reach a MN without knowing its current location/address. So, the home address would be the best address to start communication with.

Because the home address is still needed, but is invalid on foreign links, it cannot be the only IPv6 address of a MN. Therefore, a second address, called CoA, has to be formed and used on a foreign link [10],[15].

Moving to a Foreign Link

When moving to a foreign link, a CoA is formed, based on the prefix of the foreign link combined with the MN's interface identifier. (The CoA can be formed on stateless or stateful mechanisms.)

To inform the HA of the MN's CoA, a binding update (BU) is sent from the MN to the HA, containing the MN's home address and its CoA. The home address is included in the home address option (HAO), which is part of the DOH; the CoA is put into the Src or into the alternate-care-of address option (inside the mobility header). The HA acknowledges this BU by sending a binding acknowledgment (BAck) message to the MN, and stores this binding information in its binding cache, thus being able to forward incoming packets, addressed to the MN's home address, to the MN's CoA.

Whenever a packet is sent to the MN's home address, the HA checks its binding cache for an entry based on this address. If so, an outer header is created (including the HA's address as Src, and the MN's CoA as Dst) on that packet and will be sent to the MN. When the MN receives this packet, the outer header will be removed, and the original packet is handed over to the upper layers.

This tunneling process is bidirectional. So, when sending a packet from the MN to the CN, an (inner) header is formed, containing the MN's home address in the Src and the CN's address in the Dst field; then an outer header is created, which is like the one from the HA, but Src and Dst swapped. After the packet reaches the HA, the outer header is being removed and forwarded to the CN. That way, mobility of the MN is totally transparent to the upper layers and the CN [10].

Henceforth, this procedure is being referred to as **normal mode** because it is the standard technique for opening and, unless otherwise decided by the MN, proceeding communications in MIPv6, which does not require the CN to be aware of the MN's IP mobility (i.e., just „understanding“ IPv6, not MIPv6).

Route Optimization

In many cases, a MN will be connected to foreign links, far away from its HA. But still, all traffic between the MN and its CNs has to be routed by the HA, even if both shared the same link, as shown in Figure 9.3 (continuous line).

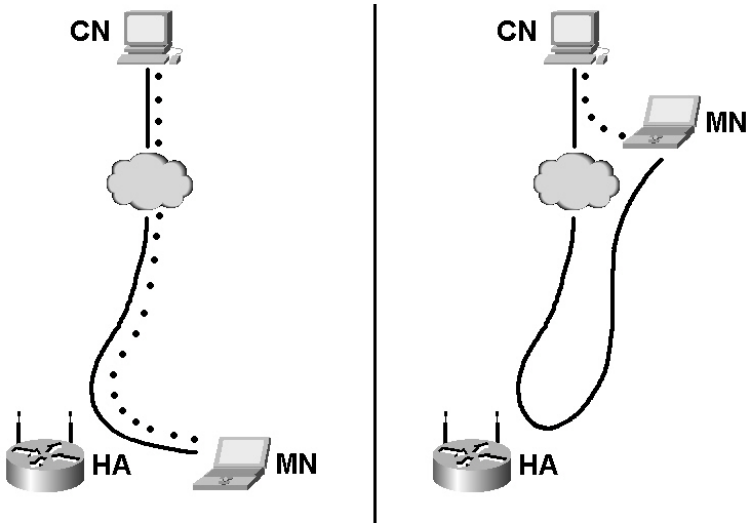

Abbildung 9.3: Route Optimization

But if the CN is aware of the MN's IP mobility, and if the MN decides to establish a direct connection to its CN (bypassing the HA), routing can be optimized by the following steps.

The MN sends a BU to the CN. The MN's home address is included in the HAO, so the CN will know which connection to update/redirect. Like the HA, it stores this information in its binding cache and sends a BAck to the MN. Now the MN can communicate directly with the CN without tunneling all packets through the HA, as shown in Figure 9.3 (dotted line).

When the MN's IP layer receives data from the upper layers, a packet is formed, including a typical IPv6 header (Src: MN's home address, Dst: CN's address). If there is an entry in the binding update list for this Dst, the HAO is added in the DOH, containing the CoA. After processing optional operations (e.g., IPsec), the CoA in the HAO is replaced by the Src and vice versa. Now the packet is sent to the CN, which performs the same operations on that packet in reverse order.

Like the MN, the CN creates packets as expected by the upper layers. If there is a binding cache entry for the Dst (in this case: the MN's home address), the Dst is replaced by the MN's CoA. Furthermore, a routing header is added, including the MN's home address (the former Dst) and a segments left field set to 1. When the packet is received by the MN, it processes the routing header by replacing the Dst with the routing header address. Since this address is the MN's home address, it will forward the packet to itself and finally to its upper layers.

The advantage of using these methods instead of tunneling is that there are only three addresses (Src, Dst and HAO or routing header) in every packet, and therefore saving bandwidth. Tunneling requires four addresses: two outer and two inner ones [10].

A Mobile IPv6 Handoff Scenario

Depending on specifications and implementations, a MN can detect (Layer 3) movement due to several reasons: unreachability of the current AR, link-layer triggers, or even hints from RtAdv with different subnet prefix.

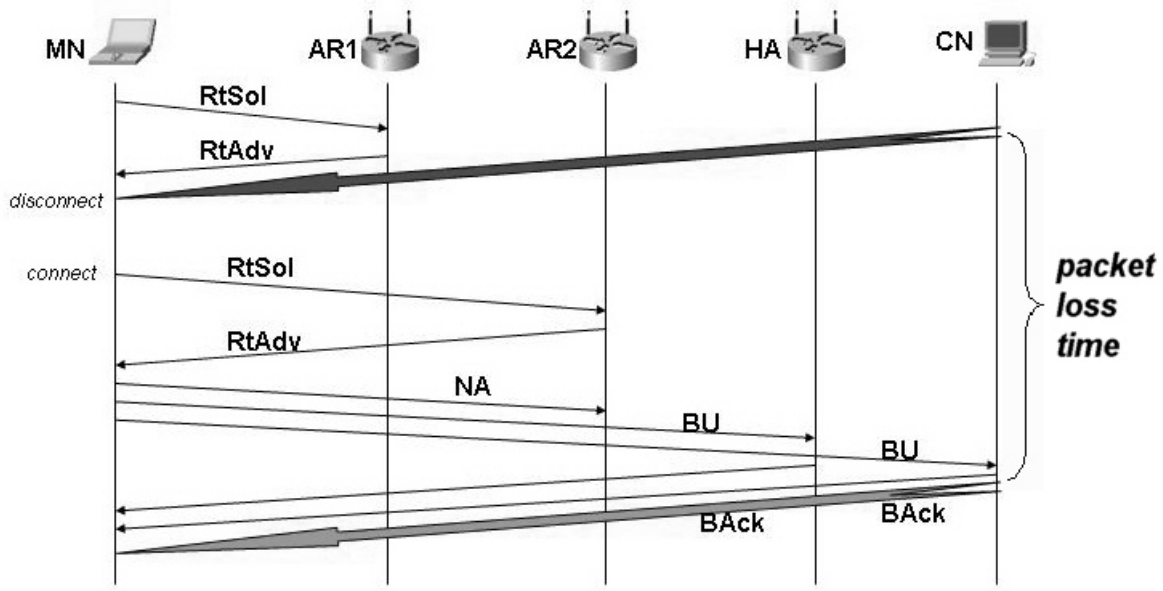

Abbildung 9.4: MIPv6 Signaling Diagram

However, when a MN already *decided* to move to a new link, it disconnects (or gets disconnected) from the current link. Then it has to form a topologically valid CoA. Therefore, the MN sends a RtSol, which will be answered by the new AR, sending a RtAdv. This message may be sent unsolicited.

After *detecting* the new link (i.e., receiving the RtAdv), the MN is able to *configure* its new CoA on the basis of the subnet prefix, derived from the RtAdv message, and its interface identifier. According to [17], the MN has to perform DAD on the new link to verify uniqueness of its new CoA. Beside this stateless mechanism, IPv6 allows nodes to resolve their addresses by stateful address configuration using the Dynamic Host Configuration Protocol for IPv6 (DHCPv6).

Because the MN's HA and CNs keep sending packets to the old CoA, which is no longer in use by the MN and therefore causes the AR/APs on the old link to drop these packets, the MN should inform them about its movement (i.e., let them know about the new CoA). As described in *3.2.3 Moving to a Foreign Link*, the MN updates its bindings. After the HA and CNs received the BUs, they start addressing their packets to the MN's new CoA, beginning with the BAck messages. When all BAcks have been received by the MN, the registration - and consequently the entire handoff, too - is completed [10].

Between disconnection and completion of the registration step, every packet destined to the MN got lost. The next subchapter, *3.3 Mobile IPv6 Enhancements*, will show how to improve handoff efficiency.

9.3.3 Mobile IPv6 Enhancements

Handoff latency and the chance of packet loss may be reduced when using MIPv6 enhancements. Therefore, it should be tried to reduce the time it takes to perform detection (t_d), configuration (t_c), and registration (t_r) during handoffs.

The total time it takes to perform a handoff (T_h) is the sum of these three factors [16]:

$$T_h = t_d + t_c + t_r$$

Since the duration of the configuration step is determined by the MN's hardware, the main focus of these enhancements is to reduce detection and registration times.

Fast Handoffs for Mobile IPv6

To reduce the chance of packet loss while performing a handoff, Fast Handoffs for Mobile IPv6 (FMIPv6) introduces some methods that anticipate a MN's IP layer mobility and tunnels the MN's packets while it updates its connections.

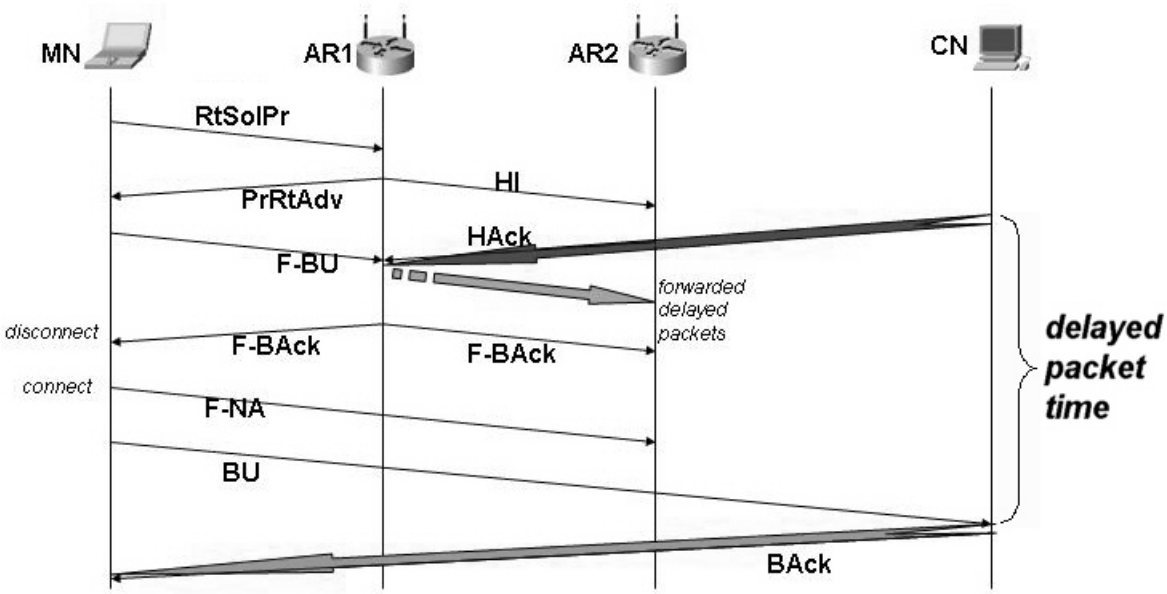

Abbildung 9.5: FMIPv6 Signaling Diagram

If the MN requests information on an AP that is not connected to the MN's current AR, it sends a router solicitation for proxy (RtSolPr) message to its AR including the MAC address of every AP that may be selected in the next handoff. The AR answers with a proxy router advertisement (PrRtAdv). This message contains the RtAdv of the new AR providing the MN with (at least) one prefix option valid for the new link.

With this information, the MN is now able to form a new CoA for the requested link, but it still uses its old one to send a fast-binding update (F-BU) to the current AR. This message informs the AR to tunnel all packets (with Dst: MN's old CoA) to its new AR.

In the meantime, the AR checks the validity of the MN's new CoA by sending a handoff initiate (HI) message to the new AR. This also sets up the (bidirectional) tunnel between these ARs. The new AR answer is a handoff acknowledgment (HAck) message, which may contain the validity of the new CoA, too. (If the validity check failed, the MN has to perform DAD on the new link.) After receiving the F-BU and HAck messages, the AR sends a fast-binding acknowledgment (F-BAck) to the MN, which informs it to whether or not to use its previously formed new CoA on the new link.

On the new link, the MN sends a RtSol message including a fast neighbor advertisement (F-NA) option, which contains the MN's old CoA and its MAC address. If the MN has not received a F-BAck before, this message is also used to confirm the MN's new CoA. The new AR's response is a RtAdv with neighbor advertisement acknowledgment (NAAck) option that indicates the validity of the MN's new CoA. Now the MN is ready to receive the tunneled and buffered packets, and update its bindings [12].

An Alternative Implementation of FMIPv6

A different approach to FMIPv6, as it was intended in [9], attempts to reduce packet delays caused by fast handoff tunneling/caching and suboptimal routing.

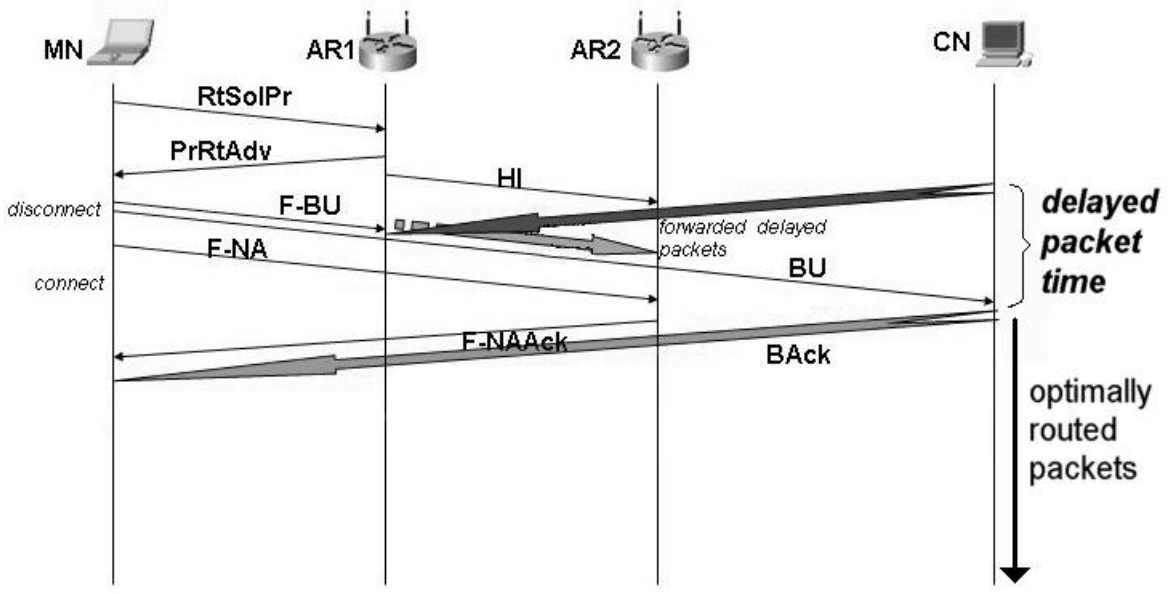

Abbildung 9.6: Alternative FMIPv6 Signaling Diagram

In standard FMIPv6, the MN updates its bindings after it has moved to a new link. But the CN keeps sending packets to the MN's old CoA until it receives a BU. So, the previous AR tunnels not only packets that were sent during disconnection, but also packets that

were sent by the CN while the MN has already established connection to the new AR and sent a BU to the CN.

The number of these delayed (tunneled and cached) packets can be reduced when sending all necessary BUs before link disconnection. Because the MN (on the old link) should already know its new CoA, an alternative design of FMIPv6 signaling (Figure 9.6) could enable the MN to inform all CNs (and its HA) of its next location. So, the BUs should be sent just after gaining knowledge of the new CoA. As it is supposed in [9], this is done by the old AR after the Layer 2 *handoff warning*.

Hierarchical Mobile IPv6

Everytime a MN moves, it has to update its bindings to its HA and CNs. In many cases, they are a long way away from each other. During that period of time, the MN is unable to receive any packets sent by them. So, the registration time (last step of the handoff execution process) depends on the distance between the MN and its CN/HA.

The basic idea of Hierarchical Mobile IPv6 (HMIPv6) is to „shorten“ distances of necessary BUs, thus reducing the registration time. This is done by giving the MN one address for a domain of several ARs, the regional care-of address (RCoA). This is the visible CoA of the MN for CNs outside this specific domain. The RCoA is assigned by a mobility anchor point (MAP) placed within a domain of several ARs, which will work as a local HA for the MN. The MN also has a second - topologically valid - CoA, the on-link care-of address (LCoA), which is derived from the AR it is connected to.

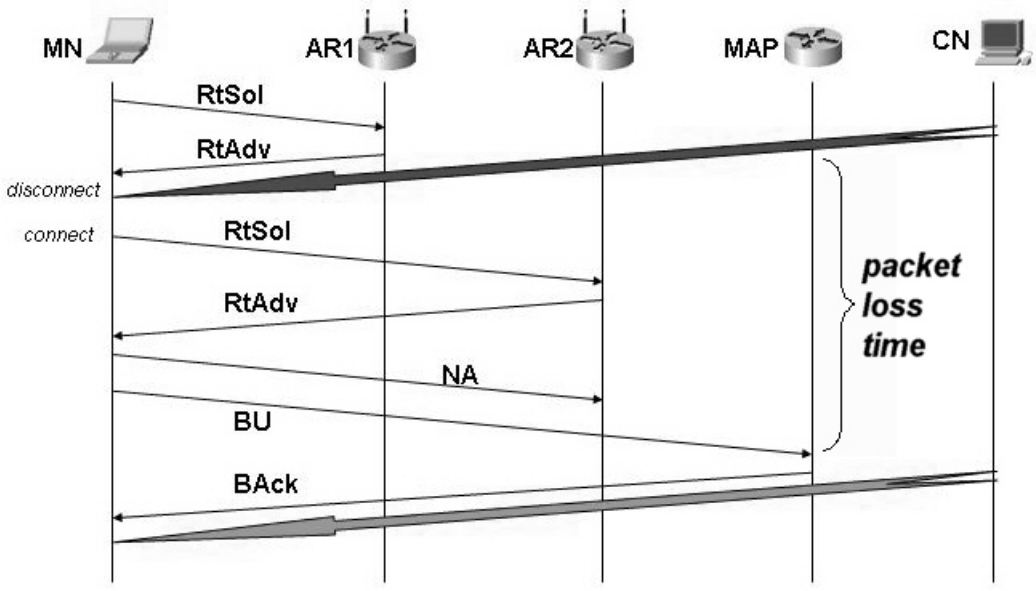

Abbildung 9.7: HMIPv6 Signaling Diagram

When entering a MAP domain, the MN receives a RtAdv message that includes a MAP option, containing the MAP's IPv6 address. The MN can now form a RCoA based on

its interface identifier and the MAP's subnet prefix, and sends a BU to the MAP. Like a BU to the HA, the MN informs the MAP of its current CoA (i.e., the LCoA) and uses the RCoA as home address (inside the HAO). But instead of setting the *H* flag, HMIPv6 introduces a new *M* flag, indicating that the MAP should operate as a temporary and local HA.

After receiving a BAck, the MN informs its HA about its new CoA. Not the real CoA, the LCoA, is being sent, but its RCoA. That way, the HA will forward packets, which are destined to the MN, to the RCoA. Because the MAP associates this address with the MN's LCoA, it will tunnel incoming packets to the MN's true CoA.

Like in normal MIPv6 mode, the MN may also decide if it updates bindings to its CNs for route optimization. If so, they will be informed about the MN's RCoA, not the LCoA. When moving between two ARs within this domain, the MN only changes its LCoA, just like it would change its CoA in standard MIPv6. Then it informs the MAP of this by sending a local BU. Because the CNs only know about the MN's RCoA (which stays the same), no BUs to the CNs or the HA are needed. That way, the critical time of packet loss during the registration step is „shortened" to the distance between the MN's new AR and the MAP.

Distances only increase if the CN is located within the same MAP domain as the MN because packets will always have to travel to the MAP first. But if a CN is on the same link as the MN, BUs may contain the LCoA because the MN knows about this circumstance due to the same subnet prefix. Nevertheless, HMIPv6 reduces registration time in the majority of cases [14].

Fast Handoffs for HMIPv6

It is obvious that a combination of the advantages of FMIPv6 and HIMPv6 would reduce (almost eliminate) the chance of packet loss within a MAP domain.

The simplest way to combine them is to implement FMIPv6 directly into a HMIPv6 network. The fast handoff ability is given to the ARs, setting up a bidirectional tunnel between them when a MN starts a fast handoff.

Unfortunatly, this approach could possibly cause higher load on the ARs (re-tunneling/caching) and higher/inefficient bandwith usage:

- The previous AR has to perform several new tasks, such as: setting up tunnels for fast handoffs, intercepting packets from the MAP to the MN, managing/ caching packets for the MN.
- In most cases, ARs are not directly connected to each other; so the actual data path of the bidirectional tunnel between the ARs often includes the MAP, and the traffic doubles between the old AR and the MAP because the packets are sent twice on this route.

- The MAP is not aware of a handoff until the MN is connected to its new AR. Only now, the MAP can redirect traffic destined to the MN's RCoA to its new LCoA. This way, the gains of anticipation become marginal.

Another approach for combining FMIPv6 and HMIPv6 is based on fast handoff tunneling by the MAP (instead of the ARs). As defined in HMIPv6, the MAP forwards packets it received for the MN's RCoA to the MN's LCoA (by forming an outer header). Just like in FMIPv6 networks, the MN (triggered by L2 events) starts requesting information on new ARs if needed. But the control messages that are sent and received are not addressed to the MN's current AR. In Fast Handoffs for Hierarchical Mobile IPv6 (F-HMIPv6), the MAP is responsible for fast handoffs.

Accordingly, the bidirectional fast handoff tunnel is set up between the MAP and the MN's new AR, which caches the incoming packet and eventually forwards them to the MN when connected. The MN then sends a HMIPv6-regular local BU, but not to inform the MAP of its new LCoA; the MAP is of course already aware of that because it managed to resolve a valid new LCoA for the MN (included in the PrRtAdv message). The MAP takes this BU as a signal that the fast handoff tunnel is no longer needed and terminates it. As a response, the local BAck indicates the end of fast handoff networking, switching back to HMIPv6 mode [11].

9.4 Roaming Scenarios

Every MIPv6 enhancement mentioned in *3.3 Mobile IPv6 Enhancements* provides improvements to regular MIPv6, reducing handoff times and packet loss. But in some cases, they can be pointless or even destructive in terms of handoff efficiency and bandwidth usage.

9.4.1 The Value of Route Optimization

Four simple scenarios shall demonstrate how close **pointless**, **very useful**, **worsening**, and **unspecified** really are when evaluating the value of a MIPv6 enhancement/improvement like route optimization.

The main focus is on how the value of a certain MIPv6 implementation can vary.

1. As already shown in chapter *3.2.4 Route Optimization*, packet delays can be reduced by direct communication between MN and CN, instead of routing the traffic through the HA. If the MN and its CN are relatively close, route optimization may be considered as **very useful**.

2. But if the MN is very near (but not on) the home link, route optimization would cause nearly the same traffic flow as in normal mode. Hence, route optimization is **pointless** in this case.

3. To make things even **worse**, a handoff between the HA and a nearby AR results in an extremely long registration time if the CN is far away because the traveling times of the BUs and the returning BAcks are determined by the distance between the MN's and its CN. (Without route optimization: MN-AR-HA.)

4. According to a scenario shown in Figure 9.8, the value of route optimization is not so clear (i.e., **unspecified**).

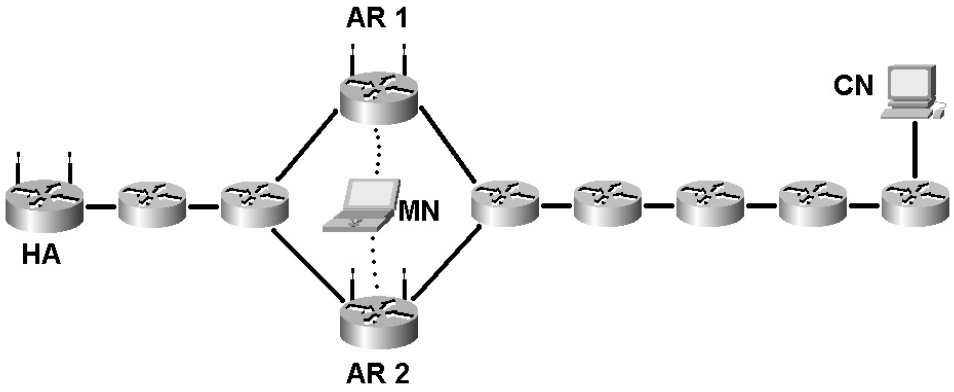

Abbildung 9.8: Unspecified Value of Route Optimization

A MN between its HA and its CN profits from route optimization when communicating directly with the CN. Packets travel 7 hops (MN-AR1-CN), instead of 13 hops (MN-AR1-HA-AR1-CN) in normal mode. But when the MN moves from AR1 to AR2, registration time is almost two times longer (14 hops: MN-AR2-CN-AR2-MN) than in normal mode (8 hops: MN-AR2-HA-AR2-MN) because the MN is closer to its HA than to the CN. In this case, neither the one implementation nor the other can be declared **winner**.

9.4.2 Impact of Distance

Most MIPv6 enhancements are designed for special network setups.

In a typical handoff scenario, there is:

- a MN, performing a handoff between
- 2 ARs, while keeping connection alive to its
- CN and
- the HA.

Because a MAP acts as a local HA, in roaming scenarios in (F-)HMIPv6 networks, the MN just needs to update bindings with the MAP, not with its real HA.

These roaming scenarios reflect the impact of increasing the distance between one operating unit and the rest. That way, it can be shown that placing just one component **wrong** can reverse the effects of improvement causing packet loss and delays.

1. AR <–> HA/MAP

   This roaming scenario display the dependency on updating the HA. In (F-)HMIPv6, this represents a poorly configured MAP domain.

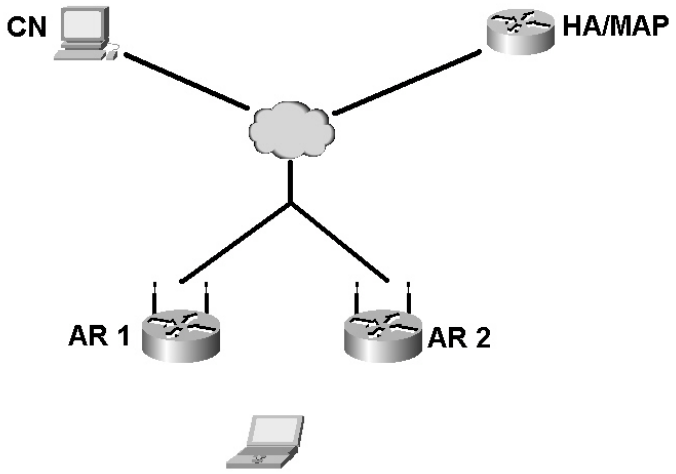

Abbildung 9.9: Distance AR <–> HA/MAP

   Increasing the distance between HA/MAP and the ARs does not affect communication between the MN and its CN during the handoff process of route-optimized MIPv6 and FMIPv6.

   But in HMIPv6 this prolongs the registration process; during that period of time every packet between the MN and its CN is lost because the MAP keeps sending all packets (destined to the MN's RCoA, which is still valid) to the MN's previous LCoA, until the MAP receives a BU, and starts forwarding packets to the MN's new LCoA.

   In comparison to MIPv6, FMIPv6 offers low detection times, and therefore should be used when far from home link, or being located in poorly configured MAP domains.

   Although F-HMIPv6 reduces packet loss in a MAP domain, it is not an improvement to pure FMIPv6 in this case because the MAP has to manage fast handoff signaling and tunneling. (In FMIPv6, the previous AR manages fast handoff signaling and tunneling while being located at close range to the new AR.)

2. Home Link <–> CN

   A MN is often located in its home network (or somewhere nearby) while the CN is far away.

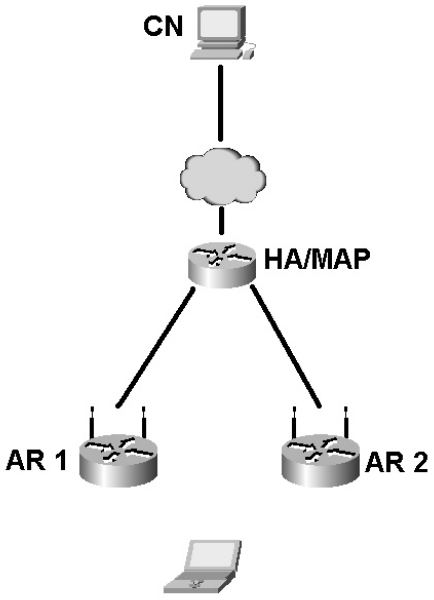

Abbildung 9.10: Distance Home Link <–> CN

Because no mobility is invoked outside a MAP domain, it doesn't matter where the CN is located (outside the MAP domain) when performing a handoff within a HMIPv6 enhanced network. That applies to MIPv6 too, but only if no route optimization is invoked.

Route-„optimized" BUs to the CN have to travel longer than BUs to the HA. Also, the distance between MN and CN increases the registration time directly proportional. FMIPv6 suffers from the same problem because of its route optimization. But in contrast to HMIPv6 and MIPv6, there is no packet loss during a handoff. However, because of long delayed communication, the CN continues sending packets to the MN's old CoA, causing the ARs to tunnel and cache many packets, and thus increasing the delay in communication more and more. Running real time applications (e.g., VoIP) has to be found impossible.

The best solution to this scenario would be a packet loss reducing hybrid of HMIPv6 and FMIPv6, like F-HIMPv6.

9.4.3 Rapid Movement

When moving with high speed (e.g., in a car, train, or airplane) within a well-covered mobile area, network design and deployment become questions of immense importance. The faster movement is, the lesser bandwith there is. This connection will be shown in this scenario.

Considering this scenario in a network setup as shown in Figure 9.11, it is obvious that route optimization should not be activated because traffic between the MN and its CN always passes through the HA. Due to the similarity to the roaming scenario in *4.2 Impact*

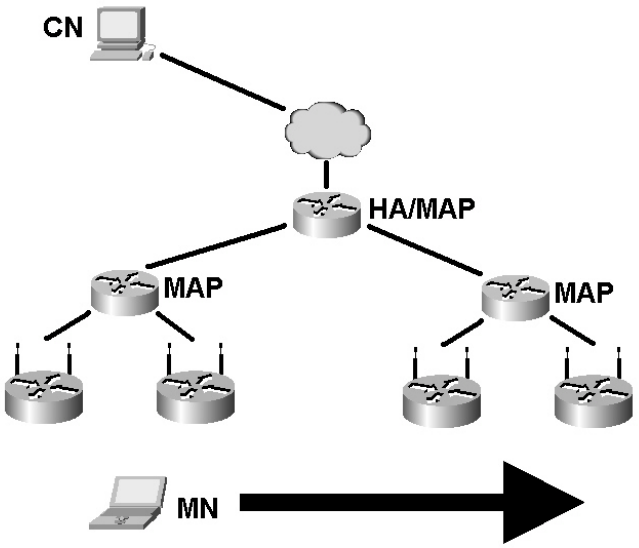

Abbildung 9.11: Rapid Movement

of Distance: 2. Home Link <–> CN, it is obvious that sending BUs directly to the CN is not an option!

When running MIPv6 in normal mode in this roaming scenario, the MN may not be able to update its bindings with its HA in time. It may receive packets on the new link; but at a certain velocity, the handoff execution process may take an unreasonably amount of time.

It should be noted that there is no benefit from a HMIPv6 implementation that deploys a MAP at the HA/MAP position shown in Figure 9.11. Although it may seem enticing to cover the whole area of movement by a sigle MAP, this will result in nearly the same handoff times as in MIPv6 (normal mode). But „shortening“ distances of necessary BUs by bringing a local HA (MAP) closer to the MN may be realized by deploying a MAP near an AR. Because this domain only covers a section of the entire movement, there have to be several adjacent/overlapping MAP domains. Unfortunately, handoffs between MAP domains cause even longer registration times.

But detection and registration steps can be reduced by anticipation, and therefore solve problems relating to the time it takes to perform the handoff execution process. With a list of neighboring ARs (and maybe even recommended addresses), the MN can form its new CoA before spending valuable time on the new link, trying to get a valid CoA. It is also thinkable to forward packets to the next link, it will move to. This way, no packet gets lost. So, FMIPv6 may solve some problems of rapid movement. But when movement becomes even faster, the MN may not be able to receive all the forwarded and cached packets. This will result in even more forwarded packets. But when reducing the amount of data (e.g., decreasing the quality of VoIP), the MN may be able to receive all buffered packets.

Again, a combination of FMIPv6 and HMIPv6 gives the best results with the benefits of anticipation, forwarded/cached packets, and lesser signaling. But still, there are more

problems. The faster movement is, the further the MAP must be from its associated ARs, to „prevent“ BUs between the MN and its CN. But a MAP too far away from its ARs degrades the handoff efficiency of (F-)HMIPv6.

This brings in bi- or multi-casting. Because the MAP is not aware of the direction that the MN is moving to, the MAP has to broadcast every packet on its domain. Assuming the ARs know, how to handle and forward/cache these packets, data loss may be eliminated. But that would cause an extremely high bandwidth usage.

Another approach is to move to a different wireless network technology. An upward vertical handoff (e.g., WLAN to UMTS) reduces the number of handoffs that are needed because of the higher network coverage. An upward vertical handoff often takes longer and may cause packet loss, but as a result, the connection is stable and enduring. Furthermore, possible horizontal handoffs on the new link may be as fast as on the old network.

On the downside, bandwidth is often lower than on networks of smaller coverage. But as it was shown, the MN is unable to use it because of permanent handoffs. Some other aspects may also be considered before performing an upward vertical handoff, such as cost, security, and QoS. In contrast to horizontal handoffs, the handoff decision step (mentioned in *2.2.1 Handoff Decision*) becomes more important because signal strength may not always be the crucial factor.

9.5 Conclusion

As it was shown in chapter *4 Roaming Scenarios*, handoff efficiency greatly improves in most cases when using MIPv6 enhancements. But the central question is: Which enhancement should be implemented?

When talking about implementations on the client side, the MN should „understand“ all the standards and enhancements of MIPv6. The decision to whether or not to activate a specific enhancement can be made by (not) sending the required signals (i.e., messages). Unfortunately, the MN does not know which mode would be best in a certain situation. A solution may be to rely on experience-based results that may be gathered by (automatic) connection tests or user preferences. But when visiting a new link for a short time, it would be unwise to test every available MIPv6 implementation with all CNs. So the user has to trust in the network design after all.

In general, there are four basic rules for MNs:

1. If there is a CN in the same MAP domain, but not on the same link as the MN, direct route optimization via BU should be considered. (Don’t use HMIPv6!)
2. Enable route optimization for communication with near CNs.
3. Disable route optimization if the HA is significantly closer than the CN.
4. When routing is optimized, use FMIPv6.

(These rules are not universally valid in terms of improving handoff efficiency!)

But when designing and setting up a wireless network, it is of fundamental importance to choose the right MIPv6 implementation because, as mentioned above, most MNs trust in the network administrator's decision.

Since route optimization is managed by MNs, there are only two options of improvement left that can be provided by the network administrator: FMIPv6 and HMIPv6. Because the latter can be extended by fast handoff mechanisms that decrease detection time and reduce the chance of packet loss, F-HMIPv6 should be considered worthy of implementation when setting up a MAP domain. Since it is yet another question of route optimization, one simple clue could indicate which enhancement should be implemented:

- If MNs communicate mainly with CNs that are located outside an intended MAP domain, choose (F-)HMIPv6.
- Otherwise, implement FMIPv6 to improve handoff efficiency of directly communicating MNs.

In closing, there is no perfect MIPv6 implementation. But because of different functionality, it is possible to determine the most efficient approach when considering all necessary circumstances.

Literaturverzeichnis

[1] Charkravorty, R., P. Vidales, L. Patanapongpibul, K. Subramanian, I. Pratt, J. Crowcroft, „On Inter-network Handover Performance using Mobile IPv6“, University of Cambridge Computer Laboratory, Technical Report, June 2003.

[2] Conta, A., S. Deering, „Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification“, RFC 2463, December 1998.

[3] Crawford, M., „Transmission of IPv6 Packets over Ethernet Networks“, RFC 2464, December 1998.

[4] Deering, S., R. Hinden, „Internet Protocol, Version 6 (IPv6) Specification“, RFC 2460, December 1998.

[5] Elektronik-Kompendium, „UMTS-Systemarchitektur“, http://www.elektronik-kompendium.de/sites/kom/0910221.htm.

[6] France Telecom, „Wireless Networks“, http://www.rd.francetelecom.com/en/technologies/ddm200207/print_index1.htm, July 2002.

[7] Herzog, U., „Wireless Access Technologies“, http://www.eurescom.de/message/messageMar2002/wirelessaccesstutorial.asp, March 2002.

[8] Hinden, R., S. Deering, „Internet Protocol Version 6 (IPv6) Addressing Architecture“, RFC 3513, April 2003.

[9] Howie, D., J. Sun, A. Koivisto, „A hybrid model for wireless mobility management using IPv6“, University Oulu, Finland, 2001.

[10] Johnson, D., C. Perkins, J. Arkko, „Mobility Support in IPv6“, draft-ietf-mobileip-ipv6-24, work in progress, June 2003.

[11] Jung, H. Y., S. J. Koh, H. Soliman, K. El-Malki, B. Hartwell, „Fast Handover for Hierarchical MIPv6 (F-HMIPv6)“, draft-jung-mobileip-fastho-hmipv6-04, work in progress, June 2004.

[12] Koodli, R. (Ed.), „Fast Handovers for Mobile IPv6“, draft-ietf-mobileip-fast-mipv6-08, work in progress, October 2003.

[13] Narten, T., E. Nordmark, W. Simpson, „Neighbor Discovery for IP Version 6 (IPv6)“, RFC 2461, December 1998.

[14] Soliman, H., C. Castelluccia, K. El-Malki, L. Bellier, „Hierarchical Mobile IPv6 Mobility Management (HMIPv6)“, draft-ietf-mobileip-hmipv6-08, work in progress, June 2003.

[15] Soliman, H., „Mobile IPv6: Mobility in a Wireless Internet“, 1st Ed., Boston, April 2004.

[16] Stemm, M., „Vertical Handoffs in Wireless Overlay Networks“, ACM Journal of Mobile Networs and Applications (MONET), Vol. 3, No. 4, 1998.

[17] Thomson, S., T. Narten, „IPv6 Stateless Address Autoconfiguration“, RFC 2462, December 1998.

[18] Tripathi, N. D., J. H. Reed, H. F. Vanlandingham, „Handoff in Cellular Systems“, IEEE Personal Communications, December 1998.

[19] Wang, H. J., J. Giese, R. Katz, „Policy-Enabled Handoffs Across Heterogenous Wireless Networks“, in Proceeding of 2nd IEEE Workshop on Mobile Computing and Applications (WMCSA 1999), New Orleans, LA, February 1999.

Kapitel 9

Sichere Authentifizierung mit 802.1x im WLAN

Robert Schultz

Authentifizierung war schon ein Thema, das es lange vor der großflächigen Verbreitung des Internets gab. Über eine sichere Art und Weise die Identität einer anderen Person darzustellen dachte die Menschheit schon nach, lange bevor es IEEE oder IETF gab, die solche Standards für das Internet ausarbeitet. Das Internet ist ein weitgehend anonymes und allumfassend zugängliches Medium. Der Schutz persönlicher Daten ist eine Herausforderung, der bisherige Standards mehr oder weniger effektiv gewachsen sind.

Mit der Verbreitung von drahtlosen Netzwerkumgebungen ist diese Herausforderung größer geworden, und nicht alle Schritte, die zu ihrer Bewältigung vorgenommen wurden, hatten lange Bestand.

Mit der Anpassung des Protokolls 802.1x an drahtlose Netzwerkumgebungen wurde ein großer Schritt in zuverlässige Sicherheit von drahtlosen Netzwerken getan. In dieser Arbeit sollen die für den Nutzer so transparente, aber für den Betreiber und das System sehr komplexe Arbeitsweise, aber auch der praktische Einsatz dieser Technologie und einem dahinter arbeitenden RADIUS Server beleuchtet werden.

Inhaltsverzeichnis

1 Einleitung

1.1 Motivation

Computernetzwerke sind aus dem heutigen Leben kaum mehr wegzudenken. Jede größere Firma betreibt ihr eigenes Intranet, praktisch jede Universität betreibt ein Campus-Netz. Innerhalb dieser Netze bedarf es mittlerweile immer größerer Mobilität, verbunden mit einer großen Zahl an Zugangsmöglichkeiten zu diesen Netzwerken. Professoren halten ihre Vorlesungen mit Notebook und Beamer ab, Präsentationen für einen Kunden laufen nach dem selben Stil oder ein Mitarbeiter möchte seine Arbeit einfach im Freien erledigen, wie uns die Werbung für das kabellose Notebook-Gesamtpaket eines führenden Herstellers für Computerplattformen in seiner Werbung suggeriert.

Hierbei tritt das Problem auf, dass an jedem dieser Zugriffspunkte, seien sie drahtgebunden oder nicht, eine Zugangskontrolle durchgeführt werden muss. Eine zentrale Verwaltung solcher Zugangsdaten ist hierbei obligatorisch, da der Zugang meist über eigene Endgeräte geschieht, die nicht permanent an diesem Zugangspunkt stehen: der Mitarbeiter steckt einfach sein Notebook in eine Ethernet-Dose, der Student schaltet sein WLAN ein.

Gegebenenfalls ändern sich auch die Rechte, die ein einzelner Benutzer hat, um auf verschiedene Resourcen eines Netzwerks zuzugreifen. Zum Beispiel kann ein Student als wissenschaftlicher Mitarbeiter beschäftigt werden, was ihm größere Zugriffsrechte einräumen könnte, Mitarbeiter wechseln in andere Abteilungen, die jeweils unterschiedliche Resourcen zu Verfügung haben. Die Möglichkeit eines einfachen Wechsels aller Attribute eines Nutzers, ohne großen Verwaltungsaufwand, wird zur Pflicht.

Mit dem wachsenden Aufkommen solcher Architekturen wurde also schnell klar, dass ein Standard zur Authentifizierung geschaffen werden musste, der die Bedürfnisse der Netzwerke unterstützte. Da Zugangskontrolle auf Ebene 2 schon länger ein Thema war, wenn es um Zugang zu Einwahlservern war, wurde ein Authentifizierungsprotokoll (EAP aus PPP) von dort adaptiert, das Resultat ist 802.1x.

1.2 Verwendete Protokolle

In der Arbeit werden verschiedene Basisprotokolle genannt, die an der jeweiligen Stelle nicht weiter erläutert werden. Dies geschieht nun vorab hier:

MD5 ist ein Algorithmus, der dafür entwickelt wurde, digitale Signaturen zu erzeugen. Er erzeugt eine Prüfsumme aus einem String, auf eine Art und Weise, dass es sehr unwahrscheinlich ist, dass zwei verschiedene Strings die selbe Prüfsumme ergeben.

PAP (Password Authentication Protocol) ist eine Methode, um Zugangsdaten zu prüfen. Dabei wird dem Server vom Client unverschlüsselt Benutzerkennung und Passwort übergeben, die der Server dann auf Gültigkeit prüft.

CHAP (Challenge Handshake Authentication Protocol) ist ebenfalls zur Prüfung von Zugangsdaten gedacht. Hierbei bekommt der Client vom Server einen ID-String sowie einen zufällig generierten String. Diese beiden verbindet er mit seinem Passwort (das der Server kennt), erzeugt einen MD5-

Hash des ganzen und gibt diesen an den Server zurück. Stimmt dieser mit dem vom Server nach gleichem Prinzip erzeugten Hash überein, ist die Authentifizierung erfolgreich.

TKIP (Temporal Key Integrity Protocol) ist ein Protokoll zur Erzeugung von Schlüsseln für die Datenübertragung in drahtlosen Netzen. Hierbei wird mittels eines temporären Session Keys, den MAC-Adressen der beteiligten Systeme und einem sogenannten MIC-Schlüssel (Message Integrity Code) ein WEP-Schlüssel erzeugt, der nur begrenzt gültig ist.

2 Konkurrierende Technologien

2.1 VPN mit IPSec

Schon vorne weg eine grundsätzliche Bemerkung zum *Virtual Private Network* (VPN): es verfolgt im Grundsatz andere Ziele als 802.1x. Steht bei 802.1x die Sicherung von Netzwerkresourcen deutlich im Vordergrund, so konzentriert sich ein VPN auch konkret auf die Sicherung von Netzwerkverkehr. 802.1x bietet hierfür nur Automatismen, um bestehende Sicherheitsmechanismen effektiver zu machen (mit periodisch erzwungener Neuanmeldung entstehen pseudodynamische WEP-Schlüssel).

Es gibt mehrere Tunneling-Protokolle, die für ein VPN verwendet werden könnten, dennoch ist *IPSec* (Secure IP) vom Sicherheitsaspekt das stärkste, darum wird auf eine Betrachtung anderer Protokolle an dieser Stelle verzichtet (Tunneling bedeutet, dass der IP-Verkehr zusätzlich in ein anderes Transportprotokoll verpackt wird).

Die Idee hinter einem VPN ist die, einen entfernten Zugriff auf ein (Firmen-)Netzwerk so zu behandeln, dass einmal die Sicherheit der Daten, die zwischen den beteiligten Rechnern (also dem VPN-Server im Firmennetz und dem Remote-Client) versendet werden so sicher zu machen, als würden sie nur innerhalb des LAN versendet, nicht etwa über jede beliebige Datenleitung im Internet, deren Sicherheit niemals eingeschätzt werden kann. Speziell bei drahtlosen Abschnitten in solchen Verbindungen ist die Sicherheit eher gering einzuschätzen, da hier zusätzlich zum logischen Zugriff auch der physikalische Zugriff auf das Trägermedium für Aussenstehende möglich ist. Ein Client nimmt zunächst Kontakt zu einem VPN-Server auf, der in dem zu erreichenden Netzwerk steht. Dort authentifiziert, baut der VPN-Server eine getunnelte Verbindung zum Client auf. Dem Client stehen nun die Resourcen des Netzwerks so zur Verfügung, als ob er physikalisch Teil dieses Netzwerks wäre.

IPSec arbeitet in 2 verschiedenen Modi, im *Transport Mode* und im *Tunnel Mode*. Im Transport Modus wird der IP-Header ersetzt, um die kryptographischen Funktionen auf das Paket anwenden zu können, im Tunnel Modus wird das gesamte IP-Paket in ein neues Paket gekapselt. Der Tunnel Modus wird für VPN Verbindungen verwendet.

Das IPSec-Protokoll verwendet im Tunnel-Modus 2 weitere Protokolle für die Absicherung des gesendeten Daten: *Authenticated Header* und *Encapsulated Security Payload.* AH sorgt dafür, dass der Empfänger eines Pakets sicher sein kann, dass der Absender auch der ist, der er zu sein scheint, gleichzeitig garantiert AH die Integrität eines Pakets. Damit wird einer Veränderung der Nutzdaten durch dritte vorgebeugt. ESP übernimmt die Absicherung der Nutzdaten gegen einfaches mitlesen, indem diese in verschlüsselter Form übertragen werden. Kombiniert man beide Methoden erreicht

man ein hohes Maß an Sicherheit, sowohl für die Daten als auch für den Empfänger.

Zusammen betrachtet, ist eine VPN/IPSec-Lösung zur einfachen Absicherung des Zugriffs auf ein Netzwerk zu groß. Sofern nicht die Vorgabe, gesicherten Zugriff auf das Netzwerk aus allen Teilen des Internets erlangen zu können im Lastenheft steht, reicht eine Absicherung eines Netzwerks an seinen WLAN-Accesspoints mittels port-basierter Authentifizierung völlig aus. Zumal auch dort eine Datenverschlüsselung realisiert werden kann.

2.2 WPA

Wi-Fi Protected Access (WPA) ist eine Entwicklung der Wi-Fi Alliance, einem Zusammenschluß verschiedener Hardwarehersteller. Es baut auf 802.1x und EAP auf. In zwei verschiedenen Modi (*Pre-Shared Key* und *Managed Key*) soll es für den Benutzer auf einfachste Art größtmögliche Sicherheit ermöglichen, ohne neue Hardware zu benötigen. Es ist auf jeder Wi-Fi zertifizierten Hardware einsetzbar, ein Zertifikat, das heutzutage auf fast allen WLAN-Produkten zu finden ist. Im PSK-Modus kennt der Accesspoint ein Passwort, mit dessen Hilfe sich jeder Benutzer an einem Netzwerk anmelden kann. Die Sicherheit dieses Betriebsmodus ist für Heimnetzwerke meist ausreichend, da jedes Paket mit einem dynamischen Schlüssel verschlüsselt übertragen wird. Hierfür kommt das *Temporal Key Integity Protocol* (TKIP) zum Einsatz.

Im Managed Key Modus wird die Authentifizierung über das Extensible Authentication Protocol abgewickelt (siehe dazu Kapitel 4: EAP). Die Verschlüsselung wird ebenfalls über TKIP realisiert.

Der PSK-Modus ist für größere Netzwerke nicht geeignet, da mit jedem regulären Benutzer die Gefahr steigt, dass das einzige Passwort an unbefugte Dritte weitergegeben wird. Der Managed-Modus erfordert die Einrichtung eines Authentication-Servers, ganz so, wie es auch in 802.1x vorgesehen ist. WPA-Managed bringt Vorteile gegenüber allen 802.1x-Implementierungen, die keine dynamische Schlüsselvergabe verwenden, ansonsten ist sie technisch gleichwertig. Allerdings erfreut sich WPA einer größeren Verbreitung als 802.1x mit EAP, im Sommer 2004 wurde der Standard 802.11i verabschiedet, der WPA2 (eine Weiterentwicklung von WPA) beinhaltet. Damit dürfte 802.1x als Authentifizierungs-mechanismus für WLANs ausserhalb von WPA und WPA2 aussterben.

3 Port-basierte Authentifizierung (802.1x)

3.1 802.1x Architektur und Namensgebung

In der Definition von 802.1x existieren drei interagierende Systeme. Zum einen auf der Benutzerseite der sogenannte *Supplicant*, das System des zu authentifizierenden Benutzers. Auf der anderen Seite der *Authentication Server,* das System, dass entscheiden kann, ob ein Benutzer berechtigt ist, den angefragten Zugriff auf Netzwerkresourcen zu erhalten. Zwischen diesen steht der sogenannte *Authenticator*, dieses System ist der Zugangspunkt zu einem Netzwerk für den Supplicant, so zum Beispiel ein Einwahlknoten eines ISP, aber auch ein Accesspoint in einer drahtlosen Netzwerkumgebung.

Supplicant und Authenticatior werden nach Spezifikation jeweils auch als *Port Authentication Entities* (PAE) bezeichnet.

Der Authenticator selbst markiert nur einen Zugang zu einem bestimmten Netzwerk, respektive das Ende eines bestimmten Netzwerks. Er selbst hat keinerlei Informationen über erlaubte Benutzer und deren konkrete Zugriffsrechte. Jeglichen eingehenden Verkehr prüft er lediglich auf Rechtmäßigkeit und leitet dann gegebenenfalls eingehende Authentifizierungsanfragen an einen oder mehrere ihm bekannte Authentication Server weiter. Verkehr von bereits authentifizierten Systemen lässt er in das hinter ihm liegende Netzwerk passieren.

Diese Überprüfung geschieht mittels Zuweisung eines bestimmten Status für jeden logischen Port, *authorized* und *unauthorized*. Letzterer sorgt dafür, dass über einen Port kein Benutzerverkehr zugelassen wird. Die Spezifikation erlaubt es allerdings, Verkehr zum Management des Clients zuzulassen, wie zum Beispiel DHCP oder eine Webseite, die als Informationsseite dient. Solche Ausnahmen werden in der Konfiguration des Authenticator festgelegt.

Ein Authentifizierungsvorgang findet, logisch gesehen, zwischen Supplicant und Authentication Server statt, der Authenticator dient ausschliesslich als Vermittler. Um die Authentifizierungsdaten zwischen den relevanten Systemen zu transportieren, verwendet der Authenticator ein von EAP abgeleitetes Protokoll, wahlweise EAP over LAN bzw. EAP over Wireless (EAPOL/EAPOW, sie unterscheiden sich nur im Format ihrer zugehörigen Frames), um die Kommunikation mit dem Supplicant zu bearbeiten, für die Kommunikation mit dem Authentication Server kommt das RADIUS-Protokoll zum Einsatz.

802.1x selbst implementiert keine Authentifizierungsmethoden. Es stellt lediglich ein Framework zur Verfügung, um verschiedene Authentifizierungsalgorithmen zu transportieren. Es stellt Mechanismen zur Verfügung, um Authentifizierungsanfragen zu stellen und deren Resultat zu präsentieren. Damit lösen Änderungen der Authentifizierungsmechanismen keine gravierenden Anpassungen auf dem Benutzersystem aus.

3.2 802.1x in WLANs

802.1x bietet auch ein Framework zur Benutzerauthentifizierung für drahtlose Netzwerke. Die einzige Anpassung gegenüber kabelgestützten Netzwerken ist die Definition eines „Ports“ (nicht zu verwechseln mit dem Port einer TCP/UDP-Adresse). IEEE hat festgelegt, dass die Zuordnung eines mobilen Endgeräts an einen Accesspoint als logischer Port zur Verwendung von 802.1x interpretiert werden darf. Die erfolgte Verbindung zwischen einem AP und einem WLAN-NIC wird der State Engine in der Clientsoftware als Aktivierung des Link Layers gemeldet. Übertragen auf das verdrahtete Netzwerk käme dies wahlweise dem Einstecken eines Netzwerkkabels oder der Aktivierung der Netzwerkschnittstelle gleich. Diese 802.11-Zuordnung muss erfolgreich durchgeführt worden sein, damit 802.1x aktiv wird. Eine zweite Anpassung ist die Verwendung des EAPOL-Key-Frames um dynamisch WEP-Schlüssel an den Supplicant zu übermitteln.

Ein weiteres Merkmal drahtloser Netzwerke im Vergleich zu kabelgestützten Varianten ist das Roaming. Hierbei bewegt sich ein Client von einem AP zum nächsten. Dies war in der ursprünglichen Definition von 802.1x nicht vorgesehen und wird deshalb von diesem Protokoll auch nicht unterstützt. Das Roaming muss von den beteiligten APs geregelt werden. Hierzu verwenden sie das Inter-Acces Point Protocol (IAPP), das von der IEEE unter dem Standard 802.11f entwickelt wurde.

4 Das Extensible Authentication Protocol (EAP)

802.1x verwendet auf Clientseite das *Extensible Authentication Protocol*. EAP ist in RFC 2284 definiert und wurde ursprünglich für die Benutzung mit dem *Point-to-Point Protocol* (PPP) entwickelt. Mit der Einführung von PPP bedurfte jedes Authentifizierungsprotokoll einer „PPP protocol number“. Um zu vermeiden, dass für PPP eine große Zahl solcher „protocol numbers“ definiert werden (es musste flexibel gehalten werden, da Authentifizierung auf verschiedenste Bedürfnisse zugeschnitten sein muss), die später vielleicht nutzlos werden würden, legte die IETF fest, EAP als Standardprotokoll zu verwenden. EAP kam mit einer einzigen solchen Nummer aus (0xC227), während es in sich eine große Zahl Authentifizierungsmechanismen vereinte.

EAP ist im eigentlichen nur eine Kapselung eines Authentifizierungsverfahrens und kann prinzipiell über jedes Protokoll auf der Ebene der Sicherungsschicht im ISO/OSI-Modell verwendet werden, bisher wurde es aber hauptsächlich mit PPP benutzt. Die Besonderheiten von drahtlosen Netzwerken in Bezug auf Abhörsicherheit laden aber dazu ein, EAP auch dort zu verwenden.

4.1 EAP Paket Format

Ein EAP-Paket kommt in der Praxis zwischen dem Client (Supplicant) und dem Zugriffspunkt des Netzwerks (Authenticator) vor. Ein typischer Paketaustauch ist in Abb.2 zu sehen.

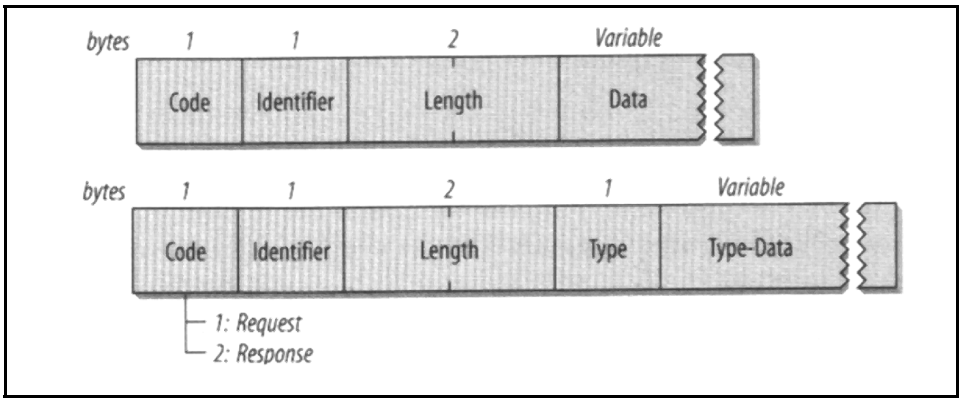

Abb.1 Das Format eines EAP-Pakets.

Das *Code*-Feld des EAP-Pakets ist ein Byte lang und identifiziert die Art des Pakets. Nach ihm richtet sich die Art und Weise, wie der Inhalt des *Data*-Feldes des Pakets interpretiert wird. Das *Identifier*-Feld ist ebenfalls ein Byte lang und dient dazu, Antworten (*responses*) einer vorhergegangenen Anfrage (*request*) zuzuordnen. Das *Length*-Feld ist 2 Byte lange und definiert die Länge des gesamten Pakets, also inklusive *Code*, *Identifier*, *Length* und *Data*. Das letzte Feld, *Data*, hat eine variable Länge. Abhängig von der Art des Pakets kann es null Bytes lang sein, oder es ist noch einmal zweigeteilt (siehe 4.2).

4.2 EAP Codes

Es gibt 4 verschiedene Werte im *Code*-Feld eines EAP-Pakets: *request*, *response*, *success* und *failure*. Sie werden von einem Byte mit den Werten 1 bis 4 repräsentiert. EAP Anfragen bestehen stets aus einem *request*/*response*-Paar. Der Authenticator sendet dem Client des Benutzersystems (Supplicant) eine Anfrage, Zugang zum System wird dann anhand der Auswertung der Antwort, die der Client zurücksendet, erteilt oder verwehrt. Während einer Transaktion werden mehrere EAP Anfragen abgewickelt. Eine Transaktion bezeichnet hier den gesamten Datenaustausch zwischen den beteiligten Systemen, beginnend mit der initialen Clientanfrage, beendet von der finalen Annahme oder Ablehnung durch den Zugangspunkt.

Nach einer abgeschlossenen EAP Transaktion wurde der Client also entweder erfolgreich oder erfolgslos authentifiziert. Sobald der Authenticator festlegt, dass eine solche Transaktion beendet ist, sendet er entweder ein *success*-Paket für erfolgreiches authentifizieren, oder eben ein *failure*-Paket für einen Misserfolg. Es sind mehrere Anfragen erlaubt bevor eine Authentifizierung fehlschlägt, um sicherzustellen, dass ein Client korrekt authentifiziert wird.

Das *Data*-Feld eines *success* oder *failure*-Pakets bleibt leer, jenes eines *requests* oder einer *response* besteht aus 2 Teilen, *Type* und *Type-Data*. Das *Type*-Feld ist ein hierbei ein Byte lang und bezeichnet die Art der Anfrage, respektive der Antwort. Das *Type-Data* Feld ist von variabler Länge, sein Inhalt wird über das *Type*-Byte identifiziert und interpretiert.

Type-Code 1 (*Identity*) signalisiert dem Supplicant die Aufforderung, die Identität des zu authentifizierenden Nutzers preiszugeben. Die Antwort beinhaltet im *Data*-Feld ausschliesslich den Login-Namen des Benutzers. Passwortinformationen werden hier noch nicht übertragen.

Type-Code 2 (*Notification*) wird verwendet, um dem Benutzer vom Authentication Server Nachrichten zukommen zu lassen. Diese sind rein informativ und sollen auf dem System des Clients dargestellt werden. Ein Beispiel für eine solche Information wäre das bevorstehende Ablaufdatum eines Passworts oder einfach nur eine personalisierte Begrüßung. Ein *request-notification* ist vom Supplicant stets zu beantworten, *Type-Data* bleibt hier allerdings leer, das Paket ist nur eine Empfangsbestätigung.

Der Type-Wert ist in zusammengehörigen *request/response*-Paaren immer identisch, mit einer Ausnahme. Für den Fall, dass ein *request*-Type (> 3) für den Client nicht akzeptabel ist (weil er diese Form der Authentifizierung nicht kennt oder unterstützt), kann er an Stelle dieses Types einen Type 3 - *NAK* - zurücksenden um einen oder mehrere alternativen Authentifizierungsmethoden vorzuschlagen. Dieser Vorschlag verbirgt sich in diesem Fall im *Type-Data*-Feld, so, wie er in einem regulären request im *Type*-Feld dargestellt würde: mit je einem Byte, das den einer Authentifizierungsmethode zugewiesenen Wert annimmt, für jede Methode, die der Client anbieten will.

Beginnend mit Type Code 4 werden Authentifizierungsmethoden dargestellt.

4.3 EAP-Authentifizierungsmethoden

Grundsätzlich lässt sich im EAP jedes beliebige Authentifizierungsverfahren unterbringen. RFC 3748 definiert in ihrer initialen Fassung nur wenige, nämlich *MD5-Challenge* (Code 4), *One-Time Password* (OTP, 5), *Generic Token Card* (GTC, 6), sowie *Expanded Types* (254). Der Vollständigkeit halber sei noch Code 255 für experimentellen Gebrauch erwähnt.

Bei der *MD5-Challenge* prüft der Authentication Server den Benutzernamen und einen MD5-Hash des Benutzerpassworts. Dies ist für kleine, drahtgebundene Netzwerke meist ausreichend, für drahtlose Netze jedoch nicht zu empfehlen. Der Benutzername und der Passworthash können dort von dritten leicht abgefangen und missbräuchlich genutzt werden. Alternativ könnte ein Accesspoint die Identität des eigentlichen Authenticators vortäuschen und so an die Daten gelangen (eine Art eines Man-in-the-middle Angriffs). Jede EAP-Implementation muss MD5-Challenge unterstützen, wobei es aber gestattet ist, eine Anfrage nach *MD5* mit *NAK* zu beantworten.

Für *OTP* wird zur Authentifizierung des Clients der OTP-Algorithmus verwendet, wie er in RFC 1938 beschrieben ist.

Bei *GTC* sind die Authentifizierungsdaten auf einer an das System angeschlossenen SmartCard gespeichert und werden von dort an den Authentication Server übermittelt. Ein Vorteil dieser Methode ist eine erhöhte Sicherheit gegen einen Keylogger, der das Passwort direkt auf dem Benutzersystem abgreift.

Der Expanded Type wurde eingeführt, um Methoden zu unterstützen, die nicht zentral festgelegt werden. Jeder Supplicant müsste für alle unbekannten EAP-Typen eine Vorgehensweise kennen, falls er mit einem solchen konfrontiert wird, also legt die RFC 3748 fest, unbekannte Typen unter dem Code 254 zusammenzufassen. Da diese logischerweise nur auf bestimmten hardwarespezifischen Konfigurationen vorkommen, ist die Implementierung dann in entsprechender Software auf dem Authenticator und dem Supplicant vorhanden. Es erfolgt eine Kapselung des herstellerspezifischen Codes nach dem Type-Code 254 im Feld Type-Data nach dem Muster HerstellerID (1 Byte) und Herstellermethode (1 Byte).

Weitere verbreitete Authentifizierungsmethoden sind *EAP-TLS* (Transport Layer Security), *EAP-TTLS* (Tunneled TLS), *LEAP* (Lightweight EAP) und *PEAP* (Protected EAP). Sie unterscheiden sich im Detail voneinander, dennoch haben sie alle eines gemein: Sie erweitern die ursprünglich vorgesehene (und von MD5 und GTC verwendete) Client-Authentifizierung auf eine beidseitige Authentifizierung, um besseren Schutz vor sogenannten Rouge Peers bieten zu können. Als solche bezeichnet man Rechensysteme, die sich fälschlicherweise als der eigentlich gewünschte Accesspoint ausgeben, um so an Zugangsdaten zu gelangen. Desweiteren unterstützen sie eine dynamische Schlüsselvergabe für drahtlose Umgebungen.

Im Detail arbeitet Lightweight EAP serverseitig mit einem Password Hash. Es wurde von Cisco entwickelt und eignet sich darum am ehesten für Netzwerke, in denen Ciso Hardware verwendet wird, und die den Sicherheitsstandard gegenüber MD5 nur ein wenig erhöhen müssen, eben durch Serverauthentifizierung und dynamische Schlüsselverwaltung (bei drahtlosen Netzwerken).

EAP-TLS verwendet auf beiden Seiten Zertifikate, um innerhalb eines geschützten TLS-Tunnels die Authentifizierung vorzunehmen. Hierbei wird ein TLS-Tunnel errichtet, innerhalb dessen der Austausch beider Identitäten geschützt stattfindet. Der Server kann den Client entweder durch Prüfung der im Zertifikat festgelegten Identität oder durch Nachschlagen des Zertifikats in einem Directory Service (z.b. LDAP) authentifizieren. Der Client kann die Identität des Servers im

Zertifikat kontrollieren, nachdem er die Gültigkeit des Zertifikats geprüft hat. Um EAP-TLS allerdings einsetzen zu können, muss eine eigene *Certificate Authority* (CA) betrieben werden, um alle Rechner mit Zertifikaten auszustatten.

Diese Vorgehensweise bietet einen starken Schutz gegen unerwünschte Zuhörer, wird für größere Netzwerke aber sehr schnell unpraktikabel, da für jeden Client ein Zertifikat erstellt werden muss.

EAP-TTLS schliesst dort die Lücke. Bei dieser Art der Authentifizierung besitzt nur der Server ein Zertifikat. Der Austausch der Identitäten erfolgt wie auch bei EAP-TLS in einem sicheren TLS-Tunnel, die Client-Authentifizierung wird aber nicht durch ein Zertifikat erreicht, sondern durch eine weitere EAP-Transaktion, die innerhalb des TLS Tunnels abläuft. Alternativ können auch einfachere Authentifizierungsmechanismen getunnelt werden wie PAP (Password Authentication Protokoll), CHAP (Challenge Handshake Authentication Protocol) oder MSCHAP(v2) (Microsoft CHAP, auch Version 2).

PEAP unterstützt zusätzlich noch Public Keys (Zertifikat, SmartCard), ist aber im großen und ganzen identisch mit EAP-TTLS.

Diese jetzt benannten Authentifizierungsmethoden bieten einen guten Schutz vor den meisten denkbaren Angriffen (insbesondere auf drahtlose Netzwerke), dennoch bleibt ihnen allen ein Nachteil: die Identität des Clients wird ungeschützt auf dem Netzwerk übertragen. Eine korrekte Benutzer-ID zu kennen ist aber eine Grundvoraussetzung, um einen Angriff mittels Brute-Force oder Dictionary auf passwortgestützte Zugriffspunkte zu führen, darum ist die unverschlüsselte Übertragung nicht optimal. Es müsste also ein Ansatz gefunden werden, der auch die Identität des Clients schützt.

Diese Lücke versucht *EAP-Double-TLS* zu schliessen. Die Authentifizierung wird hier in 2 Phasen geteilt. In Phase 1 authentifizieren sich beide Seiten über einen TLS-PSK Handshake und generieren eine Verschlüsselung. In Phase zwei wird dann ein kompletter TLS-Handshake durchgeführt, um beide Seiten sicher zu authentifizieren. Hier wird auch die Identität des Clients übermittelt, die somit vor unberechtigtem Abhören geschützt ist.

Nokia arbeitet an weiteren EAP-Methoden, *EAP-AKA*, *EAP-SIM* und *EAP-SIM6*, die für Einsatzszenarien in mobilen Endgeräten mit SIM-Karten gedacht sind. Hierbei zielt EAP-AKA auf UMTS und GSM-Geräte, EAP-SIM auf GSM alleine und EAP-SIM6 auf GSM-Geräte in IPv6 Umgebungen. Sie verwenden die SIM-Karten in den mobilen Endgeräten, um einen Client zu authentifizieren. Hierbei kommen bei UMTS zwei 128bit-Schlüssel zum Einsatz, bei GSM ein 56bit Schlüssel.

4.4 Beispiel einer EAP Transaktion

Abbildung 2 bildet eine EAP-Transaktion ab, die serverseitig als EAP-MD5 durchgeführt werden soll, aber clientseitig auf EAP-GTC geändert wird. Eines ist jeder Transaktion gemein, sie beginnt mit einem EAP Request und endet, je nach Ausgang, mit einem EAP Success oder einem EAP Failure.

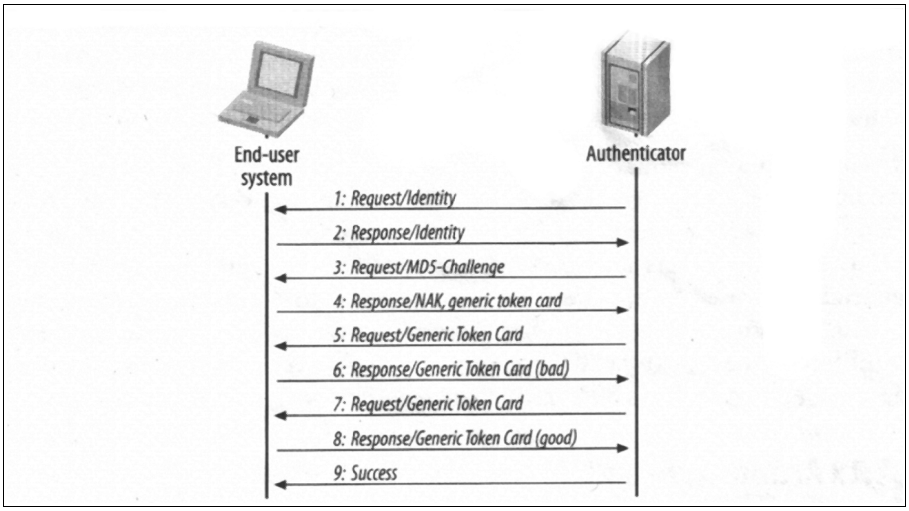

Abb.2 Eine EAP Transaktion

1. Der Authenticator erfragt die Identität eines Benutzersystems: Request/Identity
2. Der Supplicant antwortet mit einer Response/Identity
3. Mit der Indentität des Benutzers kann der Authenticator jetzt eine Authentifizierung erfragen. In diesem Fall fordert der Authenticator eine MD5-challenge: Request/MD5-Challenge
4. Das Benutzersystem ist zur Authentifizierung mittels Token Card konfiguriert. Es sendet also ein NAK mit dem Vorschlag, GTC zu verwenden: Response/NAK, GTC
5. Nun erfragt der Authenticator eine Authentifizierung mittels Token Card.
6. Der Client antwortet, hier aber zunächst mit einer falschen Sequenz: Response/GTC (falsch)
7. Der Authenticator fordert aufgrund des Fehlers eine erneute Authentifizierung an, da seine Konfiguration mehrfache Requests erlaubt: Request/GTC
8. Erneut sendet der Client seine Daten, dieses Mal korrekt
9. Nachdem die korrekten Daten beim Authenticator angelangt sind, antwortet dieser mit einem Success.

4.5 EAPOL

802.1x verwendet für die Kommunikation zwischen den PAEs EAP over (W)LAN. Die grundsätzliche Struktur eines EAPOL-Frames (in einem verdrahteten Netzwerk) zeigt Abbildung 3.

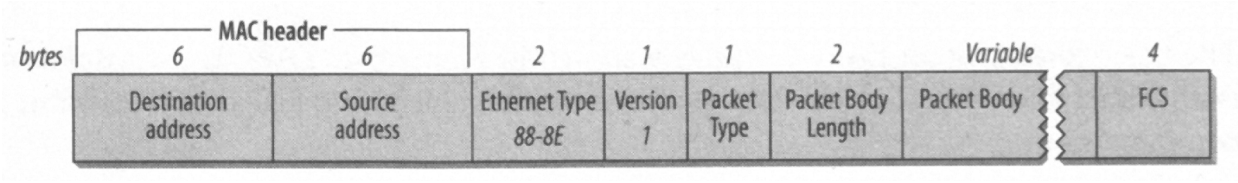

Abb.3 Ein EAPOL-Frame

Der MAC-Header enthält Absender- und Zieladresse, in einer drahtlosen Umgebung besteht er aus 24 bis 30 Bytes und enthält neben 4 Adressfeldern noch Informationen über die Protokollversion, Steuerdaten und Sequenzkontrolldaten.

Ethernet Type beinhaltet den Bytecode für die Art des Frames. In diesem Fall wäre das 88-8e, der EAPOL zugewiesen ist.

Das Feld *Version* ist momentan immer 1.

EAPOL ist eine Erweiterung von EAP. Zusätzlich zu den 4 EAP Typen definiert EAPOL noch eine kleine Zahl weitere Typen, die durch das *Packet-Type*-Feld gekennzeichnet werden.

Zunächst wäre da Typ mit dem Code 0, *EAP-Packet*. Dieses Frame beinhaltet ein EAP-Paket, diese Art macht den Großteil des EAPOL-Verkehrs aus. *EAPOL-Start* (Code 1) wird von Clients verwendet, um einen Authenticator dazu zu bringen, eine EAP-Transaktion zu beginnen, anstatt darauf zu warten, dass der Authenticator beginnt. *EAPOL-Logoff* (2) kann dazu verwendet werden, einem Authenticator das Ende der Nutzung des Zugangsports anzuzeigen, um diesen wieder in den Status „*unauthorized*“ zu versetzen. *EAPOL-Key* (3) kann verwendet werden, um Schlüsselinformationen auszutauschen, und zuletzt *EAPOL-Encapsulated-ASF-Alerts* (4) um sogenannte *management alerts* an das System zu senden. Sie haben aber für den Benutzer keine Bedeutung.

Packet Body Length gibt die Länge der Nutzdaten an, die in *Packet Body* enthalten sind. Es ist 2 Byte lang und wird auf null gesetzt, wenn keine Nutzlast im Frame enthalten ist. *Packet Body* ist von variabler Länge und in allen EAPOL Nachrichten vorhanden, mit Ausnahme von *EAPOL-Start* und *EAPOL-Logoff*.

Der Unterschied zwischen EAP und EAPOL ist lediglich der, dass eine EAPOL-Transaktion Clientseitig mit einem EAPOL-Start initiiert werden kann.

5 Remote Acces Dial-In User Service (RADIUS)

Ein RADIUS Server hat im eigentlichen drei Aufgaben zu bewältigen. Er soll Benutzer zunächst authentifizieren, im nächsten Schritt ihnen einen Zugriff auf vorhandene Netzwerkresourcen erteilen und in einem kommerziellen Umfeld auch die Abrechnung der zur Verfügung gestellten Dienste übernehmen. Zusammen nennt man das dann das AAA-Modell, AAA steht für Authentication, Authorization und Accounting (in dieser Arbeit wird das Accounting vernachlässigt). Das klassische Anwendungsszenarion eines RADIUS Servers mit AAA wäre zum Beispiel die Abwicklung des Kundenverkehrs bei einem Internet Service Provider, aber auch die Bewältigung des Kundenaufkommens an einem öffentlichen Access Point, für dessen Nutzung Gebühren anfallen sollen.

5.1 Das RADIUS-Protokoll

Der Back-End Verkehr, also der Verkehr zwischen Authenticator und Authentication Server, läuft in einer typischen 802.1x-Umgebung über das RADIUS Protokoll ab. RADIUS ist ein UDP basiertes, verbindungsloses Protokoll. Diverse RFCs befassen sich mit RADIUS, RFC 2865 beinhaltet die grundlegenden Festlegungen zu diesem Protokoll.

Die Kommunikation erfolgt an Port 1812, was eine Abweichung zur ursprünglichen Festlegung in den RFC darstellt. Der eigentlich vorgesehene Port 1645 wurde geändert, da eine Kollision mit dem „datametrics“ Dienst vorlag.

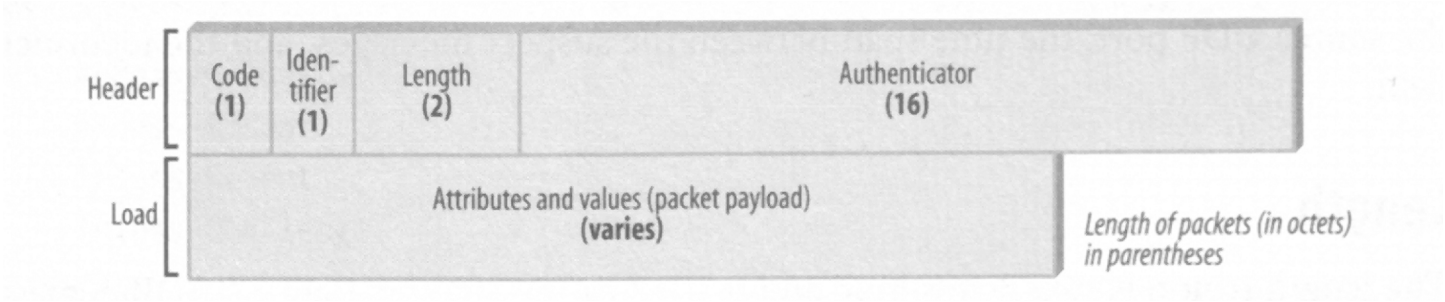

Abb.4 Ein RADIUS Paket

Das Code-Feld beinhaltet, wie schon bei den EAP und EAPOL-Paketen, die Information über die Art des RADIUS-Pakets. Es gibt derer neun, Pakete mit einer falscher Codezahl werden ohne weitere Reaktion gelöscht. Die einzelnen Codes sind:

Access-Request	1
Access-Accept	2
Access-Reject	3
Accounting-Request	4
Accounting-Response	5
Access-Challenge	11
Status-Server	12
Status-Client	13
Reserviert	255

Für diese Arbeit sind nur die Codes 1, 2, 3 und 11 relevant. Das *Identifier*-Byte wird dazu verwendet, Anfragen und Antworten einander zuzuordnen. Es dient unter anderem auch dazu, doppelte Nachrichten auszufiltern, zusammen mit Daten wie der IP-Adresse des Absenders, dessen Port sowie dem Zeitabstand zwischen zwei verdächtigen Paketen. Das *Length*-Feld beträgt 2 Byte. Es gibt die Länge des gesamten RADIUS-Pakets an. Erlaubt sind Längen zwischen 20 und 4096 Bytes. Erhält der Server ein Paket, das länger ist als im Kopf angegeben, so verwirft er alle überzähligen Bytes. Erhält er hingegen ein nach Headerangaben zu kurzes Paket, so verwirft er es komplett. Das *Authenticator*-Feld beinhaltet zwei verschiedene Typen von Daten. Der „request Authenticator“ beinhaltet 16 Byte zufällig generierten Code, während der „response Authenticator“ einen MD5-Hash über das gesamte Paket plus ein Passwort beinhaltet. Der zufällig generierte Bytecode im request Authenticator soll das ausspähen erschweren.

Ein Paket mit Code 1, Access-Request, wird vom Authenticator verwendet, um, im Kontext dieser Arbeit, einem Client Zugriff auf ein Netzwerk zu ermöglichen. Die Payload sollte den

Benutzernamen dessen beinhalten, der Zugang erlangen will, weiterhin die IP-Adresse des Authenticators, über den der Client Zugang haben will, entweder ein MD5-gehashtes oder ein CHAP-basiertes Passwort des Benutzers, wahlweise eine Zustandsmeldung. Der Identifier ist vom Typ request. Es gilt der Grundsatz, dass alle eingehenden, validen Access-Request-Pakete beantwortet werden müssen, egal ob für positiv oder negativ befunden.

Ein Paket mit Code 2, ein Access-Accept, wird vom RADIUS-Server an den Authenticator zurückgesendet um ihm mitzuteilen, dass der Zugriff auf das Netzwerk für einen Client akzeptiert wurde. Die Zugehörigkeit zu einem bestimmten Client wird hierbei seitens des Authenticators mittler des Identifiers realisiert. Das Access-Accept Paket kann so wenig Information wie möglich enthalten, darum kann die Payload dieses Pakettyps auch leer sein.

Code-3-Pakete, Access-Reject, werden im umgekehrten Fall vom Server an den Authenticator gesendet, nämlich dann, wenn eine Authentifizierung des Clients mit den gegebenen Daten nicht möglich war. Ein Reject-Paket kann zu jeder Zeit gesendet werden, auch wenn die Verbindung vorher bereits hergestellt war. Dies ermöglicht in RADIUS-Umgebungen mit aktiviertem Accounting zum Beispiel die Abschaltung des Zugriffs nach Ablauf eines bestimmten Zeitkontos, muss aber von der verwendeten Hardware seitens des Authenticators unterstützt werden.

Das Access-Challenge-Paket, Code 4, wird vom Server an den Authenticator gesendet, um eine erneute Authentifizierung zu erzwingen. Sie wird verwendet, wenn der Benutzer widersprüchliche Daten gesendet hat, der Server mehr Daten vom Client haben will oder einfach um verdächtige Authentifizierungsvorgänge zu wiederholen. Die Antwort des Clients auf dieses Paket ist immer ein neuer Access-Request, mit der erwünschten Information in der Payload.

Die Payload eines RADIUS-Pakets besteht ausschliesslich aus *Attribute-Value-Pairs* (AVP). Um die Sicherheit des gesamten Systems zu erhöhen, ist manchen Attributen der Zeitpunkt vorgegeben, an denen sie eingesetzt werden können. Beispielsweise wird von der RFC 2868 festgelegt, dass das Passwort eines Benutzers niemals in einem Response-Paket vom Server an den *Authenticator* übertragen werden darf, um zu vermeiden, dass ein solches Passwort während einer einzigen Transaktion mehrfach übertragen wird. Ebenfalls wird nie der Name eines Attributs übertragen. Alle Attribute sind mit einer Nummer versehen, die stattdessen übertragen wird. Die Nummer liegt zwischen 1 und 255, die Kennzahl 26 ist dabei als Träger für herstellerspezifische Attribute reserviert. Da die Namen eines Attributs innerhalb des Datenverkehrs nicht verwendet werden, kann es vorkommen, dass gewisse Hersteller diese Namen in ihren Implementierungen abändern. Nichtsdestotrotz behalten sie die Bedeutung, die ihrer Nummer zugewiesen sind. Jedes AVP ist nach dem selben Muster aufgebaut: Nummer, Länge, Wert. Die Länge bezeichnet hierbei die Länge des gesamten AVPs in Bytes und ist damit minimal 3. Herstellerspezifische Attribute, für die kein Platz mehr in den 255 vorhandenen Attributnummern war, werden in ein Attribut vom Typ 26 gekapselt. Dem gekapselten Attribut wird dann noch eine VendorID vorangestellt, die 4 Bytes lang ist und in der RFC 1700 festgelegt ist, der Rest ist wieder wie bei jedem anderen Attribut, die Länge des gesamten Wertpaars dann mindestens 7 Bytes.

Jedes Attribut ist festgelegt auf einen von 6 bestimmten Datentypen

Integer (*INT*):	Diese Werte werden wie sie sind übernommen.
Kennziffern (*ENUM*):	Diesem durch Integer-Werte repräsentierten Typ werden anhand der Kennziffer bestimmte Bedeutungen zugemessen, wie zum Beispiel 26=VendorSpecific.
IP-Adressen (*IPADDR*):	Eine 32-bit-Zahl, die eine IP-Adresse repräsentiert.
Zeichenketten (*STRING*):	Als UTF-8 printable codierte Zeichenketten.
Datum (*DATE*):	eine 32-bit-Zahl, die die verstrichenen Sekunden seit dem 1.1.1970 repräsentiert.
Binärdaten (*BINARY*):	Sie werden wörtlich interpretiert.

Die Informationen, welches Attribut zu welcher Kennziffer gehört und welchen Datentyp es haben kann, sind in sogenannten Dictionaries auf dem RADIUS-Server gespeichert. Ein Server muss alle Standardattribute kennen, ebenso muss er alle herstellerspezifischen Attribute verzeichnet haben. Weiterhin muss für jedes einzelne als ENUM gekennzeichnete Attribut ein weiteres Dictionary haben, in dem die Zuordnung zwischen Kennziffer und Bedeutung abgelegt ist. Durch das Prinzip der Dictionaries ist eine Erweiterung mit eigenen Attributen sehr einfach realisierbar.

Zurück zu den Attributen, für die Aufgabe als Authentication Server sind die Attribute Nr. 1-39 und 60-63 von Relevanz. Die Beschreibung aller relevanten Attribute würde den Rahmen dieser Arbeit sprengen. Darum sind in Tabelle 1 nur ausgewählte Attribute aufgeführt, die am ehesten verwendet werden und in jeder Transaktion vorhanden sind.

Bezeichnung	***ID***	***Länge***	***Datentyp***	***Erlaubt in***
UserPassword	2	18-130	STRING	Access-Request
CHAP-Password	3	19	STRING	Access-Request
NAS-IP-Address	4	6	IPADDR	Access-Request
NAS-Port	5	6	INT	Access-Request
NAS-Identifier	32	3	STRING	Access-Request
NAS-Port-Type	61	6	ENUM	Access-Request

Tabelle 1: einige RADIUS-Attribute

Eine vollständige Liste der Attribute findet sich in [1].

6 RADIUS in der Praxis: freeRADIUS

6.1 freeRADIUS Konfiguration

Es gibt viele Implementierungen eines RADIUS-Servers, die meisten sind jedoch komerziell. Frei erhältliche Implementierungen sind rar, wie zum Beispiel das GNU-Projekt RADIUS[1] oder aber freeRADIUS[2], ein Projekt, dass sich selbst als den „ersten Open Source RADIUS Server" bezeichnet und sich laut eigenen Angaben unter den Top 5 der weltweit eingesetzten RADIUS Server befindet. Alle weiteren Angaben in diesem Kapitel beziehen sich auf freeRADIUS.

Die Konfiguration eines RADIUS Servers kann sehr komplex sein, je nach Aufgabengebiet. Zur Erinnerung, der typische Einsatzzweck eines RADIUS Servers war die Aufzeichnung und Abwicklung von Kundenverkehr bei einem ISP, von daher gibt es eine Vielzahl von Attributen, die einem einzelnen Benutzer zugeordnet werden können. Wohlgemerkt: können, nicht müssen. Denn er kann von einem externen LDAP- oder SQL-Server mit Benutzerdaten gefüttert werden, so dass die Portierung eines bestehenden Nutzerverzeichnisses auf den RADIUS-Server nicht erforderlich ist.

Die Konfiguration ist nicht nur komplex, sondern gerade dadurch auch hochflexibel. So kann einer einzelnen Benutzeridentität ein bestimmtes Protokoll zugeordnet werden, über das, je nach dem, in welcher Art RADIUS-Packet das AVP „*Framed-Protocol*" platziert wird, der Benutzer an ein bestimmtes Übertragungsprotokoll gebunden werden (PPP, SLIP, ARAP und weitere) oder aber selbst eines vorschlagen kann. Es ist serverseitig möglich, die Routingtabelle des Clients zu manipulieren („*Framed-Routing*"), um bestimmten Benutzern bestimmte Gateways vorzuschreiben, die Vergabe einer IP-Adresse kann auch über den RADIUS-Server erfolgen („*Framed-IP-Address*"). Mit selbigem Attribut kann ein Client auch eine bestimmte IP-Adresse „beantragen".

Schon die Installation von freeRADIUS kann Stolperfallen beinhalten. So ist zum Beispiel unbedingt darauf zu achten, beim Konfigurieren des makefile den korrekten Pfad zu einer obligatorischen OpenSSL-Installation[3] (mindestens Version 0.97d bei freeRADIUS 1.0.0) mit einzubeziehen, da ansonsten die Verwendung von kryptobasierten Authentifizierungsmethoden fehlschlägt. Es sollte auch während des Kompilierens die Bildschirmausgabe überwacht werden, um sicherzustellen, dass sämtliche TLS-Bibliotheken auch übersetzt werden. Ein Fehler in diesem Fall kann später nur schwer erkannt werden und kann zu langwieriger Fehlersuche an falscher Stelle führen, da der Server zwar funktioniert, er aber eine Authentifizierung einfach nur ablehnt, wenn TLS zum Einsatz kommen soll, ohne mitzuteilen, dass er nicht in der Lage ist, diese Authentifizierung vorzunehmen.

Das Standard-Konfigurationsverzeichnis des Servers ist */etc/raddb*. Die dortige Datei *radiusd.conf* steuert grundlegende Serverfunktionen wie z.B. den Benutzer, unter dem der Server ausgeführt wird, der Ort der PID-Datei, die die Process-ID des Servers beinhaltet, oder auch die IP-Adresse und Port, an denen der Server arbeiten soll, falls das System, dass ihn beherbergt, mehrere Netzwerkschnittstellen hat (und nicht alle genutzt werden dürfen). Die Datei enthält auch Angaben zur Benutzerdatenspeicherung, ob also zum Beispiel gegen einen SQL-Server oder einen LDAP-Server authentifiziert werden soll. Sie im ursprünglichen Zustand sehr gut dokumentiert und damit auch recht gut konfigurierbar, wenn auch sehr umfangreich.

1 http://www.gnu.org/software/radius/radius.html
2 http://www.freeradius.org
3 http://www.openssl.org

Die Datei *clients.conf* beinhaltet muss all die Rechensysteme beinhalten, die mit dem RADIUS-Server als Authenticator zusammenarbeiten dürfen. Konkret handelt es sich hier (in einer drahtlosen Netzwerkumgebung) um alle Access Points, die ihre Zugriffe über diesen Server abwickeln wollen. Beispiel 1 zeigt eine beispielhafte Eintragung für einen solchen Serverclient:

```
client 127.0.01 {
      secret      = changeme
      shortname   = localhost
      nastype     = other
}
```

Bsp. 1 Auszug aus der Datei *clients.conf*

secret ist hier das Passwort, dass dieser Accesspoint bei der Anmeldung am RADIUS-Server zu verwenden hat, *shortname* ist eine Bezeichnung für den entsprechenden AP und *nastype* legt fest, um welche Art von Gerät sich hinter dem Client verbirgt. Da manche Hersteller nicht immer RFC-konform sind, können hier verschiedene Herstellernamen angegeben werden. Diese wären *cisco*, *computone*, *livingston*, *max40xx*, *multitech*, *netserver*, *pathras*, *patton*, *portslave*, *tc*, *ushriper* oder *other*.

Die Datei *eap.conf* befasst sich – wie der Name schon nahelegt – mit der Konfiguration der EAP-Optionen. Es muss zunächst ein *default*-Typ festgelegt werden, da beim Server ankommende Nachrichten nichts darüber aussagen, welchen EAP-Type sie benutzen. Wird in einem anderen Modul das EAP-Type-Attribut gesetzt, so erhält es den Vorrang über diesen Wert.

Im folgenden werden die einzelnen Optionen für die unterstützten EAP-Typen *MD5*, *LEAP*, *GTC*, *TLS* und *TTLS* gesetzt, so zum Beispiel Pfade zu Zertifikatdateien, den eigenen und denen der Root-CA.

Die Datei *users* dient als Sammelpunkt der notwendigen und optionalen Attribute für alle Benutzer. In ihr werden einzelne Benutzer aufgeführt, deren festgelegte Attribute (wie angesprochene *Framed-IP-Address*, *Framed-Routing* etc.) angeführt und *Fall-Through*-Werte gesetzt. War es früher so, dass ein Benutzer erst gegen das *users*-File und dann gegen das Unix-System authentifiziert wurde (es war nur ein default-Wert erlaubt) und ein Misserfolg zur Ablehnung des Benutzers führte, so ist es mit diesen Werten möglich, mehrere default-Einträge in die Datei zu schreiben und das System bei einem Misserfolg auf den nächsten „herunterfallen“ zu lassen. Dies ermöglicht eine differenziertere Benutzerauthentifizierung, zum Beispiel kann eine große Menge von Benutzern, die alle gleiche Attribute haben, durch einen default-Benutzer vertreten werden, einzeln abgeänderte Attribute von einzelnen Nutzern können explizit unter dem Benutzernamen aufgeführt werden. Dieser default-User fängt alle Benutzer auf, deren Attributinformationen unvollständig sind oder für die es keinen expliziten Eintrag gibt. Durch das *Fall-Through*-Prinzip ist es möglich, mehrere default-Benutzer zu haben. Da die Datei von oben nach unten abgearbeitet wird, beendet der Server normalerweise die Bearbeitung, sofern er einen zutreffenden Eintrag findet. Ist das *Fall-Through*-Attribut gesetzt, wandert er weiter durch die Datei, und verwendet letztendlich den Eintrag, der am besten passend ist.

Ein Eintrag im *users*-File ist immer nach dem gleichen Prinzip aufgebaut.

```
Meier            Auth-Type := Local, User-Password == „password“
                 Reply-Message = „Hello, %u“
                 Service-Type = Framed-User
                 Framed-Protocol = PPP,
                 Framed-IP-Adress = 137.193.111.222,
                 Framed-IP-Netmask = 255.255.255.0,
                 Framed-MTU = 1500
DEFAULT     Auth-Type := System
```

Bsp. 2 Auszug aus der Datei *users*

In diesem Fall handelt es sich um einen Benutzer *Meier*, der im users File (*local*) mit dem Passwort *password* authentifiziert wird. Diese Informationen stehen in der ersten Zeile der jeweiligen Benutzersektion. Darunter folgen nutzerspezifische Attribute, in diesem Fall eine einfache Textnachricht („Hello, Meier“), die Information über das *Protokoll*, dass er benutzen muss (PPP), die ihm zugewiesene *IP-Adresse* mit *Subnetz-Maske* und einem Wert für seine *Maximum Transfer Unit* (MTU). Wählt sich ein anderer Benutzer ein, so wird auf ihn der DEFAULT-Benutzer angewendet, er wird gegen das Unix-System authentifiziert.

Genauso wie die users-Datei Zugriffsrechte gibt, kann sie sie auch verbieten. Wird hinter den Benutzernamen das Attribut *Auth-Type := Reject* gesetzt, so wird dem Benutzer der Zugang verweigert.

Ein weiteres Feature sind Prefixe und Suffixe zu den Benutzernamen. Es kann damit festgelegt werden, dass ein Benutzer auf eine ganz spezielle Art angemeldet werden kann, wenn er seinen Benutzernamen entsprechend ergänzt. So kann zum Beispiel mit dem nächsten Beispiel Benutzer Meier durch Verwendung des Strings *meier.konsole* einen telnet-Zugriff auf den Host *meierhost.unibw-muenchen.de* bekommen:

```
DEFAULT     Suffix == „.konsole“, Auth-Type := System
                 Service-Type = Login-User,
                 Login-Service = telnet,
                 Login-IP-Host = anyhost.unibw-muenchen.de
```

Bsp. 3 Auszug aus der Datei *users*

Durch ersetzen des Keywords „Suffix“ durch „Prefix“ erreicht man dasselbe mit einer Vorsilbe (*konsole-meier* zum Beispiel).

Auf weitere Konfigurationsdetails wird an dieser Stelle nicht eingegangen, da diese zu umfangreich sind. Eine gute Dokumentation zu dieser Thematik findet sich in [1].

6.2 Accesspoints

Bedingt durch den technischen Fortschritt von Zugangskontrolle mit SSID und WEP-Schlüssel über WPA zu WPA2 existieren heute hauptsächlich Hardware-Accesspoints, die 802.1x in Verbindung mit WPA anbieten. Ist ein Accesspoint gewünscht, der den Front-End Verkehr angepasst abwickelt, so hilft häufig nur der Griff zu einem Eigenbau. Unter Unix/Linux Betriebssystemen ist hier die erste Anlaufstelle der hostAP-Daemon (*hostapd*) von Jouni Malinen[4]. Diese Software versetzt einen WLAN-NIC, der im *master*-Mode (die Bezeichnung für den Betrieb einer WLAN-Karte als Accesspoint) betrieben wird, in die Lage, als Authenticator zu fungieren. Er arbeitet momentan mit dem zugehörigen hostAP-Treiber für Prism2/2.5/3 Chipsätze zusammen, sowie mit dem Prism54.org-Treiber (für Prism54-Chipsätze) und dem madwifi-Treiber für Atheros ar521x-Chipsätze.

Der *hostapd* wird bei regulärer Installation über die Datei */etc/hostapd.conf* konfiguriert. In ihr befinden sich alle notwendigen Einstellungen zum Betrieb als Accesspoint mit 802.1x/WPA Authentifizierung, also nach programmspezifischen Einstellungen die Konfigurationen für das zu nutzende WLAN (SSID, Zugangskontrolllisten), 802.1x, den RADIUS-Server und zuletzt WPA.

Ein gutes Tutorial für die Einrichtung eines Linux-Accesspoints befindet sich unter [12].

6.3 Clients

Auch die Clientseite ist dem technologischen Fortschritt unterworfen. So unterstützte beispielsweise Windows XP mit Service Pack 1 noch die Authentifizierung über EAP-TLS oder GTC, nach erscheinen des Service Pack 2 ist Microsoft dazu übergegangen, WPA zu unterstützen.

Intel geht einen anderen Weg, sie liefern zu ihren Karten das Tool PROSet mit, das zumindest EAP-MD5, -TLS, -TTLS und -PEAP unterstützt, zusätzlich zu WPA. Dies ist, nachdem bereits die Treiberunterstützung von Centrino-Karten (Intel PRO/Wireless 2100 und 2200 Chipsätze) für Linux schon sehr dürftig ausfiel, nur Windows-Benutzern vorbehalten.

Unter Linux/Unix bleiben dem Benutzer nicht viele Alternativen. Zum einen wäre dies der Client von Open1x.org[5], *xsupplicant*. Dieser unterstützt zusätzlich zu den von Intel unterstützten Modi auch LEAP, -OTP, -AKA, -MSCHAPv2, zukünftig soll auch EAP-FAST unterstützt werden. WPA-Unterstützung ist in diesem Client zukünftig auch zu erwarten, in der aktuellen Version 1.0.1 ist bereits Code für eine solche Unterstützung enthalten, dieser ist aber noch nicht stabil. Weiterhin ist der mit dem hostAP-Paket erhältliche *wpa_supplicant* von Interesse. Er unterstützt alle wichtigen EAP-Typen, sowie, je nach verwendeter Hardware und Treiber, WPA und WPA2.

Eine weitere Alternative auf Unix-Betriebssystemen wäre der 802.1x Client von Aegis Meetinghouse[6], der ähnliche Unterstützung bietet das Intel Tool, aber dieser ist nicht kostenlos verfügbar, darum eher uninteressant.

4 http://hostap.epitest.fi/
5 http://www.open1x.org/
6 http://www.mtghouse.com/

7 Zusammenfassung

802.1x ist tot – es lebe 802.1x.

Mit der Verabschiedung von 802.11i und WPA2 dürften die Lösungsansätze, die 802.1x mit EAP gebracht haben, immer weiter ins Hintertreffen geraten. Die Flexibilität, sich zwischen verschiedenen Authentifizierungsverfahren zu entscheiden, um ein Zugangskontrollsystem optimal auf die eigenen Bedürfnisse anpassen zu können, gerät damit auch in Gefahr. Es dürfte zu erwarten sein, dass die Unterstützung für eine breite Anzahl von EAP-Modi nur noch gering vorangetrieben werden wird, da ein Großteil der Hardwarehersteller sich in der Wi-Fi-Alliance wiederfinden und sie somit WPA/WPA2 favorisieren dürften[7]. Ich rechne nicht damit, dass sich Firmen wie Cisco zusätzlich weiter um ihre Entwicklungen (in dem Fall LEAP) kümmern werden. Auf dem Consumer Markt wird also – bis zur Aufdeckung erster Sicherheitsprobleme – wohl WPA dominieren.

Nichtsdestotrotz baut auch WPA, zumindest im Managed-Mode, auf 802.1x auf. Ein RADIUS-Server bleibt nach wie vor notwendig, um die Authentifizierung vorzunehmen. Die Flexibilität, die dieses System dabei bietet (insbesondere die Möglichkeit, verschiedene Serverarten mit Authentifizierungsdaten anzubinden), ist ungemein hoch, so dass ein solches System meist sehr leicht in vorhandene Strukturen integriert werden kann. Gepaart mit den Sicherheitsmechanismen, die WPA und WPA2 mitbringen sollte es jedem möglich sein, sein Netz effektiv vor unbefugten Zugriffen zu schützen, wenn das von Nöten ist. Ältere Hardware ohne WPA-Unterstützung bleibt in der Regel noch so kompatibel dazu, dass ein Firmwareupgrade diesen Missstand beseitigen dürfte, sofern er von Herstellerseite angeboten wird.

7 http://www.wi-fi.org/OpenSection/members.asp?TID=2

Literatur

01: RADIUS, Johnathan Hassell, O'Reilly Verlag 2002
02: 802.11 Security, B. Potter & B. Fleck, O'Reilly Verlag 2002
03: 802.11 Wireless Networks, M.S. Gast, O'Reilly Verlag, 2002
04: RFC 2284, PPP Extensible Authentication Protocol, http://www.ietf.org/rfc/rfc2284.txt, 1998
05: RFC 3748, EAP, http://www.ietf.org/rfc/rfc3748.txt, 2004
06: RFC 1938, A One-Time Password System, http://www.ietf.org/rfc/rfc1938.txt, 1996
07: RFC 2865, RADIUS, http://www.ietf.org/rfc/rfc2865.txt, 2000
08: RFC 2548, MS Specific RADIUS Attributes, http://www.ietf.org/rfc/rfc2548.txt, 1999
09: RFC 2869, RADIUS Extensions, http://www.ietf.org/rfc/rfc2869.txt, 2000
10: RFC 1700, Assigned Numbers, http://www.ietf.org/rfc/rfc1700.txt, 1994
11: Deploying 802.1X for WLANs: EAP Types, Lisa Phifer, http://www.wi-fiplanet.com/tutorials/article.php/3075481, 2003
12: Simon Anderson, Linux Wireless Access Point HOWTO, http://oob.freeshell.org/nzwireless/LWAP-HOWTO.html, 2003
13: EAP-Double-TLS Authentication Protocol, IETF Internet Draft, http://www.ietf.org/internet-drafts/draft-badra-eap-double-tls-02.txt, 2004
14: WPA Web Page, http://www.wi-fi.org/OpenSection/protected_access_archive.asp, 2003
15: WPA2 Web Page, http://www.wi-fi.org/OpenSection/protected_access.asp, 2004