Andreas Engel • Andreas G. Lessig

Elektronische Zahlungsmittel im Internet

Übersicht und Bewertung aktueller Verfahren unter Berücksichtigung
von Kriterien der Sicherheit und Funktionalität

Studienarbeit
Vorgelegt zur Begutachtung durch
Kathrin Schier

Dezember 1997

UNIVERSITÄT HAMBURG

FACHBEREICH INFORMATIK

ARBEITSBEREICH ANWENDUNGEN DER INFORMATIK
IN GEISTES- UND NATURWISSENSCHAFTEN

Inhaltsverzeichnis

Abbildungsverzeichnis

Kapitel 1

Einleitung

Als die Autoren der vorliegenden Arbeit ihr Studium begannen, war das Internet in Deutschland allenfalls durch Clifford Stolls Buch „Das Kuckucksei“ aus dem Jahre 1989 bekannt. Der damit verbundene Presserummel um den „KGB - Hack“ weckte zwar das Bewußtsein, daß es ein weltumspannendes Netz von Computern gab, daß dieses aber schon bald auch außerhalb der Welt der Militärs und Wissenschaftler eine Bedeutung für Privatpersonen haben würde, war damals nicht abzusehen.

Dies änderte sich mit der Freigabe des World-Wide Web durch das CERN im Jahre 1992[1] und der Entwicklung von Mosaic, des ersten graphischen Browsers, im Jahre 1993 durch Marc Andreesen, die NCSA und die University of Illinois. Diese Entwicklungen bewirkten einen Boom im Internet, da es nun auch für Personen außerhalb der Universitäten interessant wurde.

Im Jahre 1992 durchbrach die Anzahl der ans Internet angeschlossenen Rechner die 1.000.000-Marke und 1993 verzeichnete das WWW einen Zuwachs von 341.634%. Die Zahl der Rechner stieg in diesem Jahr auf 3.864.000.

Mit der zunehmenden Popularität des Internet begann sofort auch eine Kommerzialisierung. 1994 nahm Pizza Hut Bestellungen über seine Web-Seiten entgegen und mit First Virtual eröffnete die erste Internet-Bank. Auch der erste Blumenladen und diverse Internet - Einkaufscenter wurden gesichtet. Heutzutage ist es schwierig, in einer Illustrierten noch eine Anzeige zu finden, in der nicht auf Seiten im Internet hingewiesen wird.

Obwohl dieser Boom sicherlich viele positive Aspekte hat (diese Arbeit wäre ohne die Informationen aus dem Internet so gar nicht möglich gewesen),

[1] Die Zahlen und Daten stammen aus [Zakon 94] und [Kristula 97] mit Ausnahme des Erscheinungsdatums des „Kuckucksei“, welches den Autoren vorliegt.

so muß aber auch gesehen werden, daß im Goldrausch des Internet-Booms allzuoft vergessen wurde, auf die Sicherheit der verwendeten Zahlungssysteme zu achten.

Auch wenn man es den Firmen nicht verwehren möchte, im Internet Geschäfte zu machen, so gilt es doch darauf zu achten, daß auch der Kunde berechtigte Anforderungen an die Art und Weise hat, wie die Bezahlung erfolgt. Bedenkt man, daß die erste Zahlungsweise darin bestand, seine Kreditkartendaten ohne jeden Schutz über das Internet zu senden, so fallen auf Anhieb mehrere Probleme ins Auge:

1. Die Daten können ohne Probleme abgehört und von Dritten zur Bezahlung benutzt werden.

2. Der Händler kann die erhaltenen Daten dazu benutzen, mehr Waren abzurechnen, als er verkauft hat.

3. Der Kunde kann bestreiten, einen Kauf getätigt zu haben und so den Händler schädigen.

4. Die Kenntnis des Kaufes verfänglicher Artikel könnte dazu benutzt werden, den Käufer zu erpressen oder in der Öffentlichkeit bloßzustellen.

5. Die Erstellung von Bewegungs- oder Kaufprofilen durch Händler, Banken, Netzbetreiber oder Lauscher ist problemlos möglich.

Diese Liste ließe sich sicherlich noch erweitern. Es wird aber deutlich, daß es gute Gründe gibt, sowohl über die Sicherheit als auch über den Datenschutz bei Zahlungen im Internet nachzudenken. Es ist das Ziel dieser Arbeit aufzuzeigen, inwieweit derartige Überlegungen Anteil an der Entwicklung diverser Zahlungsverfahren hatten.

Auch wenn es unmöglich ist, alle Verfahren zu betrachten, so hoffen die Autoren doch einen breiten Querschnitt der Verfahren geben zu können, die Ende 1996 im Gespräch waren. Neue Entwicklungen nach dieser Zeit konnten nicht in die Arbeit mit aufgenommen werden, da eine Fertigstellung derselben ansonsten in Zweifel gestanden hätte.

Kapitel 2

Grundlegende Begriffe

2.1 Grundlegende Begriffe

Die notwendigen Begriffe zum Thema Sicherheit werden in [ITSEC] recht gut definiert:

Sicherheit Unter Sicherheit von Systemen versteht man die Kombination aus Vertraulichkeit, Integrität und Verfügbarkeit.

> **Vertraulichkeit** Die Vertraulichkeit gibt an, inwieweit Schutzmaßnahmen gegen die unbefugte oder unbeabsichtigte Offenlegung von Informationen getroffen worden sind.
>
> **Integrität** Unter Integrität versteht man den Schutz gegen die unbefugte oder unbeabsichtigte Veränderung von Informationen.
>
> **Verfügbarkeit** Die Verfügbarkeit soll gewährleisten, daß Maßnahmen gegen die unbeabsichtigte oder unbefugte Vorenthaltung von Informationen oder Betriebsmitteln getroffen worden sind.

Zur Charakterisierung der unterschiedlichen Systeme sind jedoch noch einige weitere Definitionen notwendig:

Verläßlichkeit Die Verläßlichkeit gibt an, wie sicher ein System gegenüber Ausfällen und Störungen ist.

Zurechenbarkeit Unter Zurechenbarkeit versteht man die Zuordnung einer verantwortlichen Instanz zu einem Vorfall.

Identifikation Unter Identifikation versteht man die Preisgabe seiner Identität gegenüber einer Instanz.

Authentisierung Unter Authentisierung versteht man den Vorgang des Nachprüfens der behaupteten Identität einer Instanz.

2.1.1 Die vier Geldtypen

Um die digitalen Verfahren besser klassifizieren zu können, teilen wir die Art der Bezahlung mit konventionellem Geld in unterschiedliche Zahlungsverfahren, d.h. in vier verschiedene Geldtypen ein:

Koupons Koupons sind nur für die Bezahlung geringwertiger Waren konzipiert. Sie werden im Voraus erworben, haben einen festen Wert und sind im allgemeinen nur an bestimmten Stellen als Zahlungsmittel gültig. Sie können, müssen aber nicht anonym sein.

Kreditkarten Obwohl es auch bei Kreditkarten ein Limit gibt, sind sie Gegensatz zu den Koupons auch für die Bezahlung größerer Summen geeignet. Kreditkarten entsprechen dem in Abschnitt 3.1 erläutertem einfachen Grundmodell , d.h. der Verkäufer muß erst zur Bank, bevor er das vom Kunden per Kreditkarte erhaltene Geld selber weiterverwenden kann. Kreditkarten kann man, wie der Name schon vermuten läßt, im Gegensatz zum eigenen Portemonnaie überziehen, denn der Verkäufer erhält das Geld zunächst von der Kreditkartengesellschaft, die wiederum später (meistens einmal im Monat) das Geld vom Konto des Kunden abbucht. Das Bezahlen mit Kreditkarte ist nicht anonym. Ein Vorteil der Kreditkarte ist die Möglichkeit der Währungsumwandlung.

Schecks Schecks sind natürlich auch nicht anonym. Ihre Betragshöhe kann begrenzt sein, sie sind aber durchaus für die Bezahlung größerer Beträge geeignet. Mit einem Scheck stellt man dem Verkäufer eine Art Schuldschein aus und gibt ihm die Erlaubnis, genau den auf dem Scheck genannten Betrag bei der Bank einzulösen.

Münzen und Geldscheine Die wichtigste Eigenschaft von Münzen und Scheinen ist die Anonymität. Diese Form des Geldes ist nicht orts- oder artikelgebunden und hat keine obere oder untere Schranke bezüglich der Betragshöhe. Außerdem hat diese Art der Bezahlung den Vorteil, daß ein Verkäufer im Gegensatz zu Kreditkarten und Koupons das erhaltene Geld sofort weiterverwenden kann, ohne vorher zur Bank gehen zu müssen.

Was bei der Umsetzung dieser vier Geldtypen in die entsprechenden digitalen Verfahren zu beachten ist, wird später im Abschnitt 3.1.4 erläutert.

2.1.2 Kredit/Debit

Die Verfahren lassen sich in Kredit- und Debitsysteme einteilen, die sich folgendermaßen definieren lassen:

Kreditsystem Der Käufer erteilt dem Verkäufer das Recht, den zu zahlenden Betrag von seinem Konto bei der Bank einzufordern.

Debitsystem Der Käufer erhält von der Bank digitales Geld im Gegenzug zu einer Abbuchung von seinem Konto.

Diese Einteilung wird im folgenden eine Grundlage für unsere Zuordnung der Verfahren zu bestimmten Geldtypen.

2.1.3 Online/Offline

Ein wichtiger Gesichtspunkt bei der Beurteilung der Kosten, die ein Verfahren verursacht ist die Frage, ob es sich um ein Online- oder Offline-Verfahren handelt. Diese Klassen können wie folgt definiert werden:

Online-Verfahren Der Verkäufer leitet die eingehende Zahlung direkt an seine Bank weiter. Die Transaktion wird erst abgeschlossen, wenn von dieser die Bestätigung der Gültigkeit des digitalen Geldes kommt.

Offline-Verfahren Der Verkäufer nimmt die Zahlung entgegen und wendet sich erst nach Ablauf der Transaktion an die Bank, um sich den Betrag gutschreiben zu lassen.

Online Verfahren erzeugen natürlich höhere Verbindungskosten für den Händler, der sich ja zumindest bei jedem Geschäftsvorgang an die Bank wenden muß, während er beim Offline Verfahren alle Transaktionen zu einem einzigen Zeitpunkt bevorzugt nachts abwickeln könnte.

Dem gegenüber muß allerdings auch berücksichtigt werden, daß bei Online-Verfahren ein Schutz vor Falschgeld einfacher ist. Da die Banken dieses Risiko normalerweise entweder auf den Händler abwälzen oder ihm ihre Versicherungskosten in Rechnung stellen, sollte man sich die Beurteilung nicht zu einfach machen.

Kapitel 3

Kriterien

3.1 Das Grundmodell

Das Grundmodell stellt den Geldfluß und die Beziehungen zwischen Kunde, Verkäufer und Bank dar.

3.1.1 Konventionelles und digitales Geld

Um das Grundmodell beschreiben zu können, bedarf es zunächst der Klärung der zwei Begriffe "Konventionelles Geld" und "Digitales Geld" sowie deren Unterschied.

Konventionelles Geld Unter konventionellem Geld verstehen wir Geld im herkömmlichen Sinne, d.h. in Form von Münzen und Scheinen, wie es vor Generationen eingeführt und von uns täglich benutzt wird. Dieses Geld hat mehrere Eigenschaften:

1. Im Gebiet des Ausgeberstaates ist es ein allgemein anerkanntes und verwendetes Zahlungsmittel
2. Es gibt Münzen und Scheine mit unterschiedlichen Nominalwerten
3. Das Geld ist anonym
4. Es sind Vorkehrungen gegen Fälschung[1] getroffen worden, z.B. sind größere Nominalwerte (d.h. meist Scheine) mit einer Seriennummer versehen, so daß jeder Schein ein Unikat ist

[1]Nachahmung von Papier- oder Münzgeld, um es als echt, oder Abänderung von echtem Geld, um es zu einem höheren Wert in Umlauf zu setzen [Lexikon].

Digitales Geld Unter digitalem Geld verstehen wir die Repräsentation von konventionellem Geld in elektronischer Form. Digitales Geld sollte also möglichst alle Eigenschaften des konventionellen Geldes haben und je nach Art der Implementation noch einige zusätzliche.

3.1.2 Die drei Rollen im einfachen Grundmodell

Bank Die Bank hat die Aufgabe, konventionelles in digitales Geld zu tauschen und umgekehrt.

Kunde Der Kunde wendet sich an die Bank, um sein konventionelles Geld in digitales unzutauschen. Anschließend tätigt er seine Transaktionen mit dem soeben getauschtem digitalen Geld.

Verkäufer Der Verkäufer erhält als Gegenleistung für seine Waren vom Kunden digitales Geld. Bei der Bank kann der Verkäufer jederzeit das digitale Geld in konventionelles umtauschen.

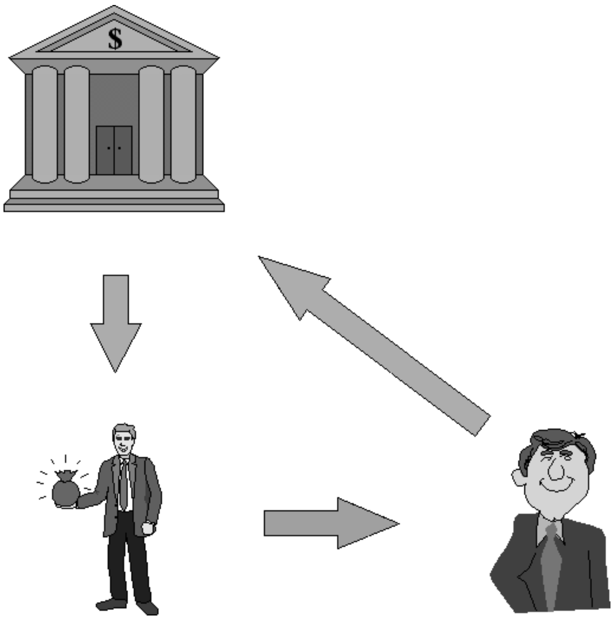

Abbildung 3.1: Das einfache Grundmodell

3.1.3 Die drei Rollen im erweiterten Grundmodell

Die Erweiterung besteht darin, daß der Verkäufer nach Erhalt des digitalen Geldes von seinem Kunden selbst in die Rolle eines Kunden schlüpfen kann, d.h. er bezahlt mit dem soeben erhaltenen digitalen Geld bei einem anderen Anbieter. Die wäre im einfachen Grundmodell nicht möglich gewesen, der Verkäufer hätte dort das vom Kunden erhaltene Geld nur direkt bei der Bank einlösen können. Das erweiterten Grundmodell entspricht also schon eher unserer Vorstellung von Geld, wirft jedoch die Frage auf, welche Geldtypen es überhaupt gibt. Hierzu siehe Abschnitt 2.1.1.

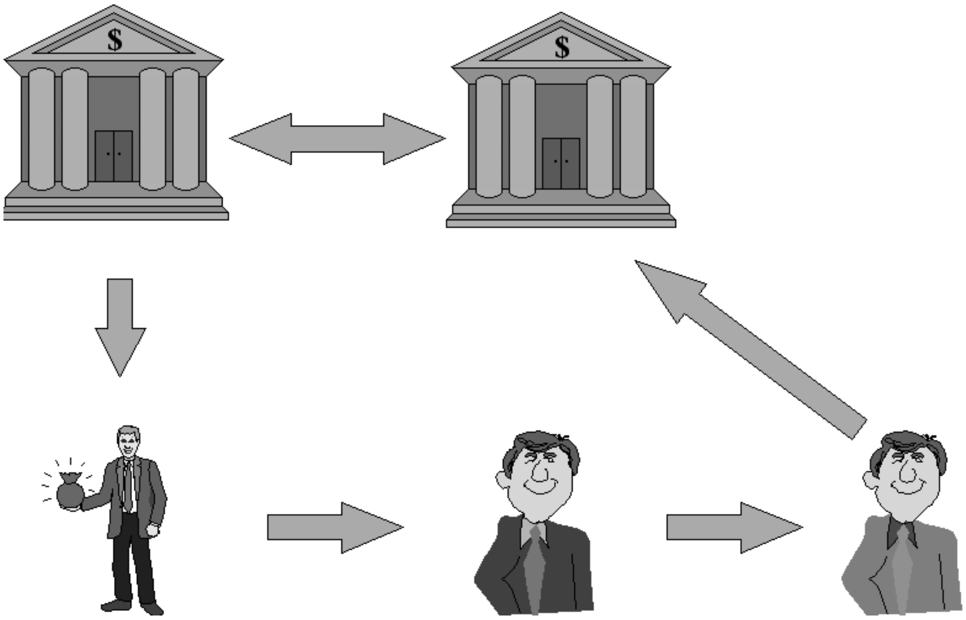

Abbildung 3.2: Das erweiterte Grundmodell

3.1.4 Die digitalen Geldtypen

Da die verschiedenen Systeme sich teilweise doch stark in der Funktionalität unterscheiden, werden wir sie gemäß ihrer Ähnlichkeit zu konventionellen Systemen einteilen.

Wie bei konventionellem Geld werden wir auch hier eine Einteilung in Koupons, Kreditkarten, Schecks und Bargeld vornehmen:

Digitale Koupons Unter „digitalen Koupons“ werden wir ein Debit- System verstehen, das für die Zahlung kleiner Beträge gedacht ist und bei dem digitale Münzen benutzt werden, die nur bei einem bestimmten Händler eingelöst werden können.

Digitale Kreditkarten Wie bei seinem konventionellen Gegenstück handelt es sich hierbei um ein Kreditsystem. Genauer handelt es sich um jene Unterklasse, bei der zum Kauf nur identifizierende Information übermittelt wird, eine Authorisierung des zu zahlenden Betrages durch den Kunden aber nicht erfolgt. Dies entspricht dem Bezahlen beim Teleshopping, wo es genügt, Kreditkartennummer und Ablaufdatum der Karte zu nennen, um dem Händler zu ermöglichen an sein Geld zu gelangen.

Digitale Schecks Bei „digitalen Schecks“ wird dem Händler nur die Erlaubnis erteilt, einen bestimmten Betrag von der Bank einzufordern. Dies entspricht dem konventionellen Scheck, auf dem ja auch der Betrag durch den Kunden eingetragen wird.

Digitales Bargeld Als „digitales Bargeld“ werden wir Debit-Systeme bezeichnen, die prinzipiell als universales Zahlungsmittel konzipiert wurden. Insbesondere dürfen digitale Münzen nicht händlerspezifisch sein.

3.2 Sicherheitskriterien

3.2.1 Vertraulichkeit

Vertraulichkeit des Transaktionsinhalts Normalerweise dürften alle beteiligten Parteien daran interessiert sein, den Inhalt ihrer Transaktionen geheim zu halten. Da das Internet per se aber keine Vertraulichkeit gewährleistet, müssen zusätzliche Maßnahmen getroffen werden um Sniffing zu vereiteln. Unterbleibt dies, besteht die Gefahr, daß abgehörte Daten zu

1. Insidergeschäften,
2. Erpressung, oder
3. Angriffen auf das System

mißbraucht werden.

Vertraulichkeit der Daten in den Endgeräten Ein eventueller Angreifer braucht aber nicht unbedingt den Verkehr auf den Netzen zu belauschen. Er kann auch direkt einen der beteiligten Rechner angreifen. Daß diesem Punkt praktische Relevanz zukommt, zeigt sich schon darin, daß dies 1994 tatsächlich geschah. Damals wurde in einen Rechner des Internetproviders Netcom eingebrochen und 20 000 Kundendatensätze mit Kreditkartennummern gestohlen.

Vertraulichkeit der Kundenidentität Im Gegensatz zur Bank oder dem Verkäufer, deren Identitäten normalerweise bekannt sind, hat der Kunde ein Interesse daran, anonym zu bleiben. Er hat zu Recht den Wunsch, nicht auf Grund seiner Transaktionen zu einem Persönlichkeitsprofil verarbeitet zu werden, auf Grund dessen er entweder mit Werbung belästigt wird oder als schlechtes Risiko betrachtet und von bestimmten Geschäften ausgeschlossen wird.

Schon heute sammeln Kreditkartenfirmen enorme Datenmengen über ihre Kunden. Dabei werden nicht nur Profile des Konsumverhaltens einzelner Benutzer gesammelt, sondern auch Auswertungen auf geographischer Ebene durchgeführt. In [Schröder 95] ist eine Graphik abgebildet, auf der die Zahlungsmoral Hamburger Kunden nach Wohngegenden dargestellt ist. Es gibt Versandhäuser, die bei neuen Kunden deren Kreditwürdigkeit nach der Adresse entscheiden.

Folgende Daten sollten also nach Möglichkeit geheim bleiben:

1. Netzadresse des Kundenrechners
2. Benutzerkennung
3. Realname
4. Adresse des Kunden

Natürlich wird in realen Systemen oft nur eine Untermenge dieser Daten auch tatsächlich geschützt. Daher ist festzustellen, in welchem Maße dieses Kriterium erfüllt wird, anstatt nur die Erfüllung oder Nichterfüllung zu konstatieren.

3.2.2 Integrität

Schutz vor Gutschriften ohne gleichzeitige Belastung
„Keine Buchung ohne Gegenbuchung“ ist ein elementares Gesetz jeder ordnungsgemäßen Buchführung. Gelänge es einem Käufer einem Verkäufer Geld zu überweisen, ohne daß dieses von seinem Konto abgebucht wird, hätte er Falschgeld erzeugt. Eine sichere Implementation elektronischen Geldes muß also

1. das Erzeugen mehrerer Kopien der selben elektronischen Münze,
2. das Verändern des Wertes einer zu übermittelnden Münze und
3. Manipulationen am eigenen Kontostand (z.B. auf einer Chipkarte)

effektiv verhindern.

Schutz vor unberechtigter Belastung Einem Händler sollte es nur erlaubt sein, von einem Kunden Geld anzunehmen, wenn die Bestätigung vom Kunden vorliegt, daß dafür ein Gegenwert geliefert wurde. So ist es beim Bezahlen mit Kreditkartendaten dem Händler möglich Beträge abzubuchen, die ihm nicht zustehen. Dies sollte entweder verhindert werden oder nachträglich behebbar sein.

Schutz vor der Umleitung und Manipulation von Zahlungsströmen
Ist die Übertragung nicht genügend gesichert, so ist es möglich, z.B. die Angabe des Empfängers einer Zahlung zu ändern. Daher muß auch nach einer gegenseitigen Authentisierung sichergestellt werden, daß kein Zwischenknoten sich in die Transaktion einschalten und als einer der Beteiligten maskieren kann. Dies kann z.B. durch Chiffrierung oder Signierung des Datenstroms geschehen.

3.2.3 Verfügbarkeit

Dieser Punkt kann von keinem der untersuchten Verfahren gewährleistet werden. Dies liegt in der Tatsache begründet, daß das Internet als Medium verwendet werden soll. Es ist trivial, einen Server durch das Senden einer ausreichenden Menge von Paketen oder besser durch das Öffnen einer ausreichenden Menge von TCP - Verbindungen praktisch lahmzulegen.

Dies setzt noch nicht einmal einen wirklichen Angriff, d.h. kriminelle Absichten voraus. Auch im Normalbetrieb kommt es regelmäßig zu Staus auf der Datenautobahn, wie auch unregelmäßige Besucher des World Wide Web aus eigener Erfahrung berichten können.

Aus diesen Gründen werden wir auch darauf verzichten, diesen Punkt in Zukunft noch einmal zu erwähnen.

Der Fairness halber soll aber auch noch darauf hingewiesen werden, daß auch konventionelle Verfahren nicht immer eine hohe Verfügbarkeit gewährleisten. Beim Zahlen mit Kreditkarte kam es schon einmal vor, daß die Kasse keine Verbindung zur Kreditkartengesellschaft aufbauen kann wodurch das Bezahlen unmöglich wurde. Auch ist es unmöglich einen Scheck oder eine Überweisung außerhalb der Öffnungszeiten der Bank abzugeben[2].

3.2.4 Verläßlichkeit

Transportverläßlichkeit Da das Internet keinen Schutz vor Störungen bietet (Vernichtung, Verdopplung oder falsche Reihenfolge von Paketen, zeitweiliger Ausfall der Verbindung), muß ein darüber liegendes Protokoll verhindern, daß dadurch Inkonsistenzen Auftreten. Die ACID[3]-Eigenschaften einer Transaktion müssen gewährleistet sein.

Wirkung einer Beschädigung der Endgeräte Gibt es nach der Beschädigung z.B. einer Smartcard oder einem Plattencrash eine Möglichkeit an das Geld zu kommen oder ist es verloren?

Anfälligkeit von Token gegen physikalische Einflüsse Falls zu dem System ein Token gehört (z.B. Smart Card), wie empfindlich ist es?

3.2.5 Zurechenbarkeit

Entlarvung von Fälschern Die Bank und der Händler haben ein Interesse, den Schuldigen einer Fälschung dingfest zu machen. Schließlich wird mindestens einer von beiden für den Schaden haften. Daher stellt sich die Frage, inwieweit der Fälscher im Nachhinein festgestellt werden kann.

---

[2] Auch Online-Banking bietet hier nur scheinbar einen Ausweg, immerhin kann auch die Telefonleitung besetzt sein.

[3] Acronym für **A**tomicity (Transaktion wird ganz oder gar nicht ausgeführt), **C**onsistency (Ein konsistenter Zustand wird in einen ebensolchen überführt), **I**solation (Gleichzeitig ablaufende Transaktionen dürfen sich nicht beeinflussen) und **D**urability (Wenn ein beteiligter Rechner abstürzt, muß es möglich sein, auf einen konsistenten Zustand zurückzufallen). (s.a. [CaSiTy])

Beweis einer erfolgten Zahlung Ist es möglich, daß der Händler leugnet eine Zahlung erhalten zu haben?

Autorisierung von Zahlungen Muß der Kunde eine Zahlung authorisieren oder reicht der Besitz eines Tokens oder die Kenntnis von identifizierenden Daten.

Gegenseitige Identifizierung und Authentisierung Alle Parteien haben ein Interesse daran zu wissen, ob sie mit einer authentischen Gegenstelle kommunizieren. Ansonsten bestünde die Gefahr,

1. Geld an jemanden zu geben, der sich als der rechtmäßige Empfänger maskiert, oder
2. Geld von jemandem anzunehmen, der gar keins besitzt.

Der erste Fall ist wohl einsichtig. Der zweite Fall kommt zum einen bei Kreditverfahren, zum anderen aber bei Verfahren zum Tragen, die auf einem Token z.B. einer Smart Card beruhen.

Mißbrauch durch das organisierte Verbrechen In der Diskussion um anonymes digitales Geld wird immer wieder die Frage laut, ob dies genau wie heute das Bargeld erlaubt, Transaktionen zu tätigen, ohne daß Finanzamt oder Polizei dies nachverfolgen können.

3.3 Funktionalitätskriterien

Im Gegensatz zu den Sicherheitskriterien fällt es bei der Funktionalität sehr viel schwerer zu sagen, was besser oder schlechter ist. Es kommt vielmehr auf den Kontext an, in dem das Verfahren eingesetzt werden soll. Ein Händler, der teure Waren über das Internet verkauft, braucht eher ein System, das diverse Währungen und Wertabstufungen der Münzen erlaubt, als eine Presseagentur, die Meldungen für Pfennigbeträge verkauft.

3.3.1 Einfache Benutzbarkeit

Eine wichtiger Punkt wäre natürlich die Forderung, daß ein Benutzer ohne Kentnisse, die über die Bedienung eines Webbrowsers hinausgehen, mit dem System zurechtkommen sollte. Leider übersteigt die Überprüfung dieses Kriteriums unsere Möglichkeiten im Rahmen dieser Studienarbeit.

3.3.2 Transferrierbarkeit

Erlaubt das Verfahren Zahlungen

1. nur direkt über die Bank (z.B. Überweisung)
2. zwischen Normalmensch und Geschäft (z.B. Kreditkarten)
3. zwischen zwei natürlichen Personen[4], die ihr elektronisches Geld dann aber auf der Bank einlösen müssen (z.B. Schecks)
4. zwischen bliebigen Parteien beliebig oft. (z.B. Bargeld)

Zu Punkt 2 muß noch gesagt werden, daß die Beurteilung aus Kundensicht erfolgt. Wenn er sich an den Händler wendet, dieser aber gewissermaßen nur als Fassade für die Bank dient (Online-Verfahren), so spielt das für den Kunden keine Rolle.

3.3.3 Skalierbarkeit

Unter Skalierbarkeit wird in der Literatur die Anpaßbarkeit eines Systems an eine drastische Steigerung der Anzahl der beteiligten Parteien verstanden.

Limits Für Zahlungen welcher Größenordnung ist das System gedacht?

Wertabstufungen Haben die vom Hersteller herausgegebenen elektronischen Münzen alle den gleichen Wert oder gibt es da Abstufungen?

Währungen Ist der Gebrauch verschiedener Währungen vorgesehen?

Anzahl der Banken Muß die Umwandlung konventionelles Geld $\leftrightarrow$ digitales Geld bei einer zentralen Bank erfolgen oder gibt es ein System mit dem digitales Geld einer Bank auch von anderen angenommen und verrechnet werden kann.

3.4 Organisatorisches und Rechtliches

Die folgenden Fragen beschäftigen sich nicht mit dem Verfahren an sich, sondern ergeben sich aus dem Vertrag, den man abschließen muß, um das digitale Geld benutzen zu dürfen.

[4]D.h. Privatpersonen, keine Händler im engeren Sinne oder Kreditkartenakzeptanzstellen.

3.4.1 Haftungsfragen

Es stellt sich die Frage, wer haftet wenn Falschgeld auftaucht. Haftet die Bank, der Verkäufer oder gibt es eine Versicherung für solche Fälle?

3.4.2 Rücktauschbarkeit

Kann das digitale Geld in konventionelles zurückgetauscht werden? Diese Frage stellt sich natürlich nur bei Debit-Systemen. Wünschenswert wäre natürlich eine Rücktauschbarkeit, bei Systemen zur Bezahlung geringer Summen könnte es aber unter Umständen unrentabel sein.

3.4.3 Erstattung verlorenen Geldes

Gibt es Regelungen falls ein Token verlorengeht, gestohlen wird oder der Rechner eines der Beteiligten irreperabel zerstört wird und das darauf gespeicherte Geld nicht mehr vorhanden ist?

Hierzu wäre eine Kulanz- oder Versicherungsregelung wünschenswert.

Kapitel 4

Digitale Koupons

4.1 Millicent

4.1.1 Motivation

Beim Transferieren von digital repräsentiertem Geld ergibt sich das Problem, daß es, um den Transfer möglichst sicher zu machen, bestimmter Verschlüsselungsverfahren bedarf. Diese kosten jedoch Zeit und Rechenleistung. Durch die benötigte Zeit entsteht eine maximale Obergrenze von Transaktionen pro Zeiteinheit und durch die benötigte Rechenleistung entstehen Kosten pro Transaktion, die eine Untergrenze für den zu transferierenden Betrag festlegen, da ab einem bestimmten Betrag die Kosten für die Transaktion einen zu beträchtlichen oder sogar grösserer Anteil als der Kaufpreis darstellen.

Gegenwärtige Maschinen können etwa bis zu vier Dutzend Transaktionen mit aufwendigeren Verschlüsselungsverfahren pro Sekunde durchführen. Bei Transaktionen im Dollarbereich stellt dies kein Problem dar, anders verhält ist es sich jedoch, wenn die Beträge der Transaktionen unter die Centgrenze fallen.

Für diesen Fall hat das System Research Center der Digital Equipment Corporation erstmals auf der Fourth International WWW Conference (Dec. 1995 - Boston) ihr Millicent Protocol for Inexpensive Electronic Commerce vorgestellt [Millicent 96].

Nun mag man sich vielleicht fragen, wozu man überhaupt Transaktionen von beispielsweise eines Hundertstel- oder Tausendstelcents benötigt. Online-Anbieter von Zeitungen, Magazinen oder Börsenkursen haben alle individuelle Artikel, die recht günstig sein würden, wenn sie einzeln verkauft würden. Die Möglichkeit, diese Artikel einzeln zu erwerben, würde diesen Service für eine breitere Masse von Benutzern attraktiv machen. Der Durchschnittsbenutzer, der wahrscheinlich nicht dazu bereit ist, bei einem unbekannten Verleger ein Konto von zehn Dollar zu eröffnen, wäre aber vielleicht eher dazu bereit, ein par Cents für einen interessant aussehenden Artikel auszugeben.

Spielt es nun eine Rolle, ob die kleinste Einheit fünf Cent oder ein Tausendstelcent ist? Bei fünf Cent könnte man die tägliche Zeitung in einige wenige Sparten aufteilen (Sport, Wirtschaft, weltweite Nachrichten, Lokalteil), bei einem Tausendstelcent könnte man die Zeitung jedoch in einzelne Artikel, Horoskop, Wetterbericht, Comics, usw. aufteilen.

Eine weiteres Einsatzgebiet wäre die Möglichkeit, einzelne Seiten eines Buches zu kaufen. Für Personen, die den kostengünstigeren Weg bevorzugen, also Studenten, wäre es dann billiger, ihre Informationen auf elektronischem Wege zu kaufen und auf Magnetdatenträgern zu speichern. Außerdem bekommen dann nicht die Hersteller von Papier und Toner, sondern der Autor selbst das Geld.

Dies mag zwar noch sehr utopisch klingen, aber in nicht zu ferner Zukunft wird man für die meisten Seiten im World Wide Web bezahlen müssen, und für diesen Zweck ist ein Protokoll wie Millicent entwickelt worden.

Im System Research Center der Digital Equipment Corporation hat man zur Zeit sowohl testweise eine Millicent-Anwendung für deren Netzwerk-Firewall-Dienste implementiert, man arbeitet aber auch an einem Millicent-Dienst für das World Wide Web.

Die Entwickler von Millicent sehen ein Ansteigen der Internet-Dienste, bei denen für den Kunden nur geringe Kosten anfallen. In Zukunft wird man für Internet-Dienste wie e-mail, file transfer, Internet telephone und tele-conferencing bezahlen müssen. Diese Dienste benötigen ein passendes Zahlungsprotokoll, und ihrer Meinung nach sei das Millicent-Protokoll ein geeigneter Kandidat.

4.1.2 Grundidee

Wie schon erwähnt, ist das Millicent-Protokoll hauptsächlich für Transaktionen geringen Betrages gedacht. Da es bei diesen geringen Beträgen nicht so stark auf die Sicherheit ankommt, wie bei großen Beträgen, können einfachere Verschlüsselungsverfahren verwendet werden. Dies verschafft einen Zeitvorteil, man kann also viel mehr Transaktionen pro Zeiteinheit durchfüren, was aufgrund der kleineren Beträge auch notwendig ist, um rentabel zu bleiben.

Die Millicentidee beruht auf der Einführung einer neuen Währung namens Scrip. Diese Währung läßt sich am ehesten vergleichen mit Fahrkarten, Telefonkarten oder Essensmarken. Scrip gilt also nur bei einem bestimmten Händler oder für ein bestimmtes Produkt.

Um zu verhindern, daß der Kunde bei allen Händlern, bei denen er etwas erwerben möchte, ein Konto in der Währung des jeweiligen Händler-Scrips einrichten muß, werden sogenannte Makler eingeführt. Jeder Makler hat seine eigene Währung, das sogenannte Makler-Scrip.

Ein neuer Kunde muß also einmalig bei einem Makler ein Konto mit Makler-Scrip einrichten. Da es sich hier um etwas grössere Beträge handelt, verwendet er hierzu eine andere Methode als das Millicent-Protokoll, zum Beispiel ein sicheres Protokoll zu Übermittlung von Kreditkartendaten. Möchte dieser Kunde jetzt bei einem Händler etwas erwerben, so wendet er sich an seinen Makler und kauft bei ihm mit dem Makler-Scrip das entsprechende Händler-Scrip. Das Wechselgeld erhält er in Makler-Scrip. Jetzt kann der Kunde wie gewünscht beim Händler einkaufen und erhält sein Wechselgeld im jeweiligen Händler-Scrip.

Die Idee des Maklers bewahrt auch den Händler davor, für jeden Kunden ein eigenes Konto führen zu müssen. Makler und Händler haben eine Langzeitgeschäftsbeziehung. Der Makler verkauft das Händler-Scrip an die Kunden und bezahlt den Händler. Es kann mehrere Möglichkeiten geben, wie der Makler das Händler-Scrip bezahlt, zum Beispiel Bezahlung im Voraus, Bezahlung auf Kommissionsbasis oder Erstellen des Händler-Scrip in Lizenz. In allen Fällen kann der Makler dadurch Profit machen, daß er beim Händler Mengenrabatt für größere Händler-Scrip-Beträge erhält, die er dann einzeln an seine Kunden weiterverkauft.

Es gibt drei unterschiedliche Möglichkeiten für den Makler, Händler-Scrip zu erlangen, je nachdem wie häufig das entsprechende Händler-Scrip verlangt wird.

Scrip Warenhaus Sind die Anfragen für das Scrip eines bestimmten Händlers nicht übermäßig hoch aber doch konstant, so handelt der Makler wie ein Warenhaus, daß heißt, er kauft regelmäßig größere Mengen des Händler-Scrip zu Discountpreisen, die er dann seperat an seine Kunden verkauft.

Scrip-Fertigung in Lizenz Ist der Bedarf der Kunden für ein bestimmtes Händler-Scrip besonders hoch, so kann es für Händler und Makler günstiger sein, wenn der Makler das Scrip des Händlers in Lizenz produziert. Konkret bedeutet dies, daß der Händler dem Makler einen bestimmten Bereich von Seriennummern zur Verfügung stellt, die für einen vereinbarten Zeitraum gültig sind. Der Makler produziert daraufhin nach Bedarf das entsprechende Händler-Scrip und erhält dafür Geld von seinen Kunden. Der Händler registriert den Gesamtbetrag des bei ihm eingehenden in Lizenz produzierten Script des Maklers, und wenn der Gültigkeitszeitraum dieser Serie vorüber ist, wird abgerechnet.

Ein Händler kann Lizenzen an mehrere Makler ausgeben, indem er ihnen unterschiedliche Bereiche von Seriennummern zuteilt, und selbstverständlich kann er auch weiterhin sein eigenes Scrip produzieren.

Günstiger ist das Lizenzverfahren also deshalb, weil weniger Kommunikation stattfindet. Die Seriennummern erfordern weniger Übermittlungzeit als das gesamte Scrip. Außerdem spart der Händler Rechenzeit, da er das Scrip nicht selbst erzeugen muß. Der Vorteil für den Makler ist, daß er nicht große Mengen des Scrip lagen muß, sondern es je nach Bedarf produziert.

Mehrere Makler In einer Umgebung mit mehreren Maklern kann es vorkommen, daß ein Händler der Einfachheit halber nur mit einem Makler, seinem „Hausmakler“ eine Geschäftsbeziehung eingehen möchte.

Ein Kunde dagegen möchte nicht jedesmal bei einem neuen Makler ein Konto eröffnen, nur weil der Händler, bei dem er einkaufen möchte, mit dem Makler des Kunden keine Geschäftsbeziehung unterhält.

Dieser Konflikt läßt sich mit folgender Idee lösen: Angenommen, ein Makler bekommt eine Anfrage für das Scrip eines Händlers, der nicht mit ihm, sondern nur mit seinem Hausmakler eine Geschäftsbeziehung unterhält. Auf die Bemühungen des Maklers, mit dem Händler eine Geschäftsbeziehung aufzubauen, erhält er als Antwort den Namen von dem Hausmakler des Händlers. Der Makler kauft jetzt das Scrip des Hausmaklers und verkauft es danach an seinen Kunden weiter. Daraufhin kann der Kunde mit dem soeben erhaltenen Scrip bei dem Hausmakler des Händlers das entsprechende Händler-Scrip erwerben und danach wie gewünscht bei dem Händler einkaufen.

Diese Idee kann mit dem Lizenzverfahren kombiniert werden, so daß ein Makler das Scrip eines anderen Maklers erzeugen kann.

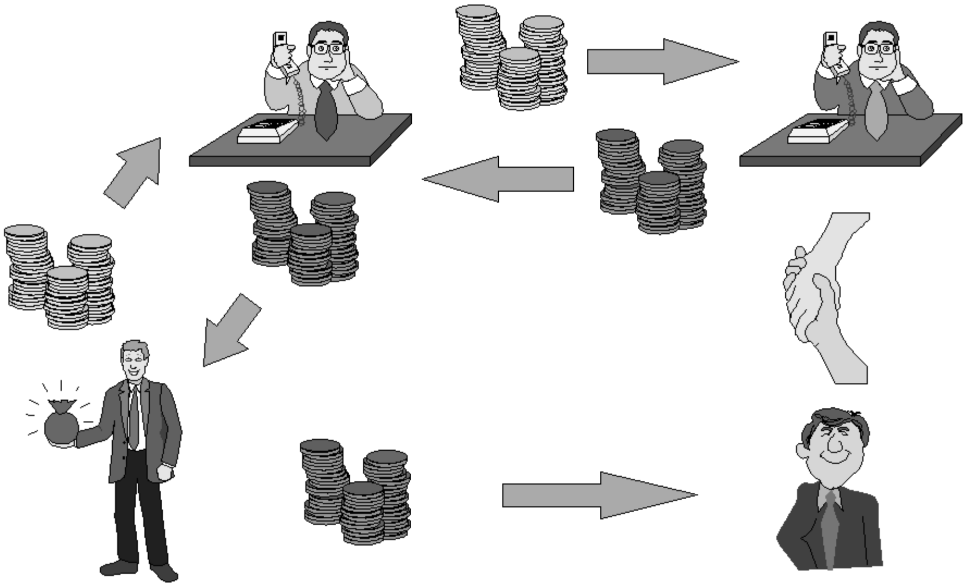

Abbildung 4.1: Millicent

4.1.3 Realisierung

4.1.3.1 Sicherheit und Vertrauen

Die Sicherheitsüberlegungen für die Millicent-Idee basieren auf der Annahme, daß das Protokoll nur für Beträge geringen Wertes eingesetzt wird. Man erwartet von den Benutzern, das sie nur ein paar Dollar in Scrip-Währung zur Zeit besitzen und nicht etwa Hunderte oder auch nur mehrere Zehner. Aus diesem Grunde lohnt sich der Diebstahl nicht, es sei denn man stiehlt größere Mengen; Da dies jedoch viel zu auffällig wäre, würde es mit Sicherheit aufgedeckt werden. Die im Besitz des Kunden befindliche Scripmenge läßt sich am ehesten mit der Kleingeldmenge in einem Portemonnaie vergleichen. Die meisten Menschen behandeln Kleingeld anders als große Scheine, und wenn ab und an mal eine Münze verlorengeht, ist man nicht allzusehr betroffen.

4.1.3.2 Vertrauensbeziehungen zwischen Kunde, Makler und Verkäufer

Der Makler wird als am vertrauenswürdigsten angenommen, dann kommt der Verkäufer und ganz zum Schluß der Kunde. Man geht davon aus, daß die Makler große, in weitem Unfeld bekannte und etablierte Institutionen sein werden, so wie Kreditkartengesellschaften, Banken oder die großen Internet Service Provider.

Betrug seitens der Makler würde sich nicht lohnen, da Kunden- und Händlersoftware unabhängig voneinander das Scrip prüfen und den Kontostand verwalten, so daß ein Betrugsversuch des Maklers sofort aufgedeckt werden würde. Da die Kunden wie schon erwähnt keine größeren Mengen an Scrip zur Zeit besitzen, müßte der Makler außerdem sehr viele betrügerische Transaktionen durchführen, damit sich der Betrug auszahlt. Dies erhöht jedoch die Wahrscheinlichkeit der Aufdeckung, welche eine drastische Verschlechterung des Rufes eines Maklers zur Folge hätte. Ein guter Ruf zieht mehr Kunden an und die regulären Transaktionen mit seinen Kunden sind für den Makler weit profitabler als die Menge an Scrip, die er sich durch Betrug verschaffen könnte.

Betrug seitens der Händler würde darin bestehen, keine Waren für gültiges Scrip auszugeben. Falls dies eintritt, wird sich der Kunde bei seinem Makler beschweren, der daraufhin, falls sich die Beschwerden häufen, die Geschäftsbeziehung mit dem Verkäufer beenden wird. Im Millicent-Modell ist der Händler jedoch auf den Makler angewiesen, um seine Geschäfte einfach abzuwickeln, aus diesem Grunde wird er von Betrug absehen.

Betrug seitens der Kunden würde in Fälschung und mehrmaligem Ausgeben desselben Scrip bestehen. Die Millicent-Protokollmechanismen verhindern dies jedoch und unterstützen gleichzeitig sogar indirekte Aufdeckung von Maler- und Händlerbetrug.

4.1.3.3 Sicherheit

Der Sicherheit von Millicent-Transaktionen liegen mehrere Tatsachen zugrunde.

1. Alle Transaktionen können (auf unterschiedliche Arten, siehe /refmillifam) geschützt werden
2. Geringwertige Transaktionen beschränken den Gewinn durch Betrug
3. Betrug ist aufdeckbar und gegebenenfalls rückverfolgbar

4.1.3.4 Die Eigenschaften von Scrip ...

- Scrip ist händlerbezogen
- Scrip kann nur einmal ausgegeben werden
- Scrip ist abgesichert gegenüber Manipulationen und schwer zu fälschen
- Scrip kann nur von seinem rechtmäßigen Benutzer ausgegeben werden
- Scrip kann effizient erzeugt und geprüft werden

4.1.3.5 ... und die grundlegenden Techniken zum Erzielen dieser Eigenschaften

- Der Text von Scrip liefert seinen Wert und identifiziert den Händler
- Das Srip hat eine Seriennummer, um Mehrfachausgaben zu verhindern
- Es existiert eine digitale Signatur, um Manipulationen und Fälschungen zu verhindern

- Der Kunde signiert jedes Scrip bei Benutzung mit einem auf das jeweilige Scrip bezogenen geheimen Schlüssel
- Durch Benutzung einer schnellen Einweg-Hashfunktion (z.B. MD5[1] oder SHA[2]) kann die Signatur effizient erzeugt und geprüft werden

4.1.3.6 Die Struktur von Scrip

Um Scrip erzeugen, prüfen und benutzen zu können, werden drei geheime Schlüssel benötigt.

`Customer_Secret` Jedem Kunden wird ein Kundenschlüssel zugeteilt, der beweißt, daß das jeweilige Scrip sein Eigentum ist.

`Master_Customer_Secret` Dieser Kundengeneralschlüssel wird vom Verkäufer benötigt, um den Kundenschlüssel aus den Kundendaten des Scrip zu erhalten.

`Master_Scrip_Secret` Um die Manipulation und Fälschung von Scrip zu verhindern, wird vom Verkäufer dieser Scripgeneralschlüssel verwendet.

All diese Schlüssel werden nach dem Zero-Knowledge-Verfahren ausgetauscht, d.h. man beweißt, daß man den geheimen Schlüssel besitzt, ohne ihn jedoch herauszugeben.

Um eine Nachricht zu signieren, wird der geheime Schlüssel als Zeichenkette an den Nachrichtentext angehängt. Auf das Resultat wird eine Hash-Funktion angewendet, die die gewünschte Signatur erzeugt. Die ursprüngliche Nachricht (ohne den angefügten geheimen Schlüssel) zusammen mit der erzeugten Signatur beweißt den Besitz des geheimen Schlüssels, da man die korrekte Signatur nur erzeugen kann, wenn mann im Besitz des richtigen geheimen Schlüssels ist.

4.1.3.7 Die Felder von Scrip im Einzelnen

`Vendor` identifiziert den Händler, bei dem das Scrip Gültigkeit besitzt

`Value` gibt die Betragshöhe des Scrip an

[1] Message Digest 5 von Rivest
[2] Secure Hash Algorithm vom amerikanischen Accredited Standard Committee X9

`ID#` ist die Seriennummer des Scrip. Ein Teil der Nummer wird dazu verwendet, den Scripgeneralschlüssel für die Zertifizierung auszuwählen.

`Cust_ID#` wird benötigt, um den Kundenschlüssel zu generieren. Ein Teil der `Cust_ID#` wird benutzt, um den Kundengeneralschlüssel auszuwählen, der ebenfalls zur Generierung des Kundenschlüssels notwendig ist.

`Expires` gibt das Ende der Gültigkeit des Scrip an.

`Props` enthält personenbezogene Daten des Kunden (Alter, Wohnort, usw.).

`Certificate` ist die Signatur des Scrip.

4.1.3.8 Überprüfung des Script

Sobald ein Händler das Scrip von einem Kunden erhält, führt er zwei Schritte zur Überprüfung des Scrip durch. Zunächst berechnet er mit Hilfe des Eintrages im Feld `ID#` selbst nocheinmal die Signatur des Scrip und vergleicht sie mit der mitgeschickten Signatur im Feld `Certificate`. Sind die beiden Signaturen identisch, handelt es sich um gültiges Scrip.

Als Zweites kontrolliert er, ob die im Feld `ID#` stehende Seriennummer bereits in einer Liste des bereits von ihm angenommen Scrip auftaucht. Ist dies der Fall, verweigert der Händler die Annahme des Scrip, da der Kunde offensichtlich ein und dasselbe Scrip mehrfach ausgeben wollte. Um zu verhindern, daß der Händler umfangreiche Listen mit Seriennummern führen muß, hat jedes Scrip im Feld `Expires` eine Art „Verfallsdatum“. Ist dieses Verfallsdatum überschritten, ist das Scrip ungültig und der Händler kann die entsprechende Seriennummer aus seiner Liste streichen. Der Kunde sollte also das in seinem Besitz befindliche Scrip rechtzeitig bei dem entsprechenden Händler in Scrip mit einem späteren Verfallsdatum (und neuer Seriennummer) eintauschen. Hierfür könnte der Händler eine kleine Gebühr verlangen, um so den Kunden davon abzuhalten, mehr Scrip einzutauschen, als er in näherer Zukunft auszugeben beabsichtigt.

Dem Feld `Props` kann der Verkäufer personenbezogene Daten wie z.B. Alter oder Wohnort des Kunden entnehmen. Der Alterseintrag könnte beispielsweise verhindern, daß der Verkäufer seine nur für Erwachsene geeigneten Artikel an Minderjährige abgibt. Außerdem läßt sich durch die Kenntnis des Wohnortes die in vielen Bundesstaaten der USA unterschiedliche Mehrwertsteuer korrekt berechnen.

Der genaue Aufbau des `Props`-Feldes wird zwischen Makler und Verkäufer vereinbart. Bei jedem neuen Erstellen von Scrip werden von jeweiligen Erzeuger die entsprechenden Daten in das Feld eingetragen. Der Händler kann in das Feld eintragen, was immer er für nützlich hält, der Makler trägt jedoch in das von ihm erzeugte Scrip die bei der Kontoeröffnung vom Kunden erhaltenen Daten ein.

4.1.3.9 Die Familie der Millicent-Protokolle

Die Millicentidee beinhaltet eine Familie mehrerer Protokolle mit unterschiedlichen Eigenschaften in Bezug auf Sicherheit, Geheimhaltung und Komplexität.

Das einfachste Protokoll dieser Familie ist das sogenannte **Scrip in the clear**. Der Aufbau des Protokolls ist einfach und sehr effizient, da keinerlei Verschlüsselungsverfahren verwendet werden. Der Kunde schickt einfach sein Scrip unverschlüsselt an den Händler, und dieser schickt wiederum die bestellte Ware und das Wechselgeld in unverschlüsselter Form zurück. Diese einfache Form des Protokolls enthält natürlich so gut wie keine Sicherheitsmaßnahmen. Ein Angreifer kann das Wechselgeld auf dem Weg vom Verkäufer zum Kunden abfangen und selber ausgeben. Versucht der eigentliche Besitzer später, dieses Geld auszugeben, so ist es bei dem Händler als bereits schon ausgegeben gespeichert, und der Händler wird die Annahme verweigern.

Ist Sicherheit und Privatsphäre erwünscht, empfiehlt sich die Benutzung des **Private and secure**-Protokolls. Dieses beinhaltet die Einführung eines nur den beiden Parteien Kunde und Verkaufer bekannten geheimen Schlüssels. Unter Verwendung eines effizienten, symmetrischen Verschlüsselungsverfahrens (z.B. DES, RC4 oder IDEA) und der Benutzung des geheimen Schlüssels entsteht dann ein sicherer Übertragungskanal.

Kauft nun ein Kunde sein erstes Stück Scrip für einen bestimmten Händler, so wird unter Verwendung der Information des Feldes `Cust_ID#` der geheime Schlüssel generiert und auf sicherem Wege mit dem Scrip an den Kunden geschickt. Ein sicherer Weg bedeutet in diesem Fall die Verwendung eines Millicent-Protokolls aus der Familie der sicheren Protokolle oder eines anderen sicheren Nicht-Millicent-Protokolls.

Der Inhalt des Feldes `Cust_ID#` muß für jeden neuen Kunden eindeutig sein, um die Privatsphäre zu waren, sollte er jedoch in keinem Zusammenhang mit der Identität des Kunden stehen. Mit den Informationen des Feldes `Cust_ID#` läßt sich auf schnellem Wege der geheime Schlüssel wiederberechnen, der Händler speichert ihn deshalb nicht direkt in Zusammenhang mit dem entsprechenden Stück Scrip.

Bekommt der Händler nun eine Anfrage seitens des Kunden, errechnet er unter Verwendung des Scrip-Feldes `Cust_ID#` wie schon erwähnt den geheimen Schlüssel, aus diesem erhält er wiederum den zum Entschlüsseln der eigentlichen Anfrage notwendigen Nachrichtenschlüssel. Das Wechselgeld in Form von Scrip kann bedenkenlos unverschlüsselt an den Kunden zurückgeschickt werden, während die Antwort auf die Anfrage und alle neuen Schlüssel mit dem Nachrichtenschlüssel verschlüsselt werden.

Bei Verwendung dieses Protokolls werden Anfrage und Antwort völlig privat gehalten. Ohne Kenntnis des Kundenschlüssels kann ein Angreifer die Nachrichten nicht entschlüsseln. Außerdem kann er das übertragene Scrip nicht stehlen, da er es ohne die Kenntnis des Kundenschlüssels nicht ausgeben kann.

Die dritte Protokollvariante trägt den Titel **Secure without encryption** und verzichtet auf den Einsatz von Kryptoverfahren, was den Verlust der Privatsphäre im Bezug auf Anfrage und Antwort zu Folge hat.

Auch bei dieser Variante erhält der Kunde beim erstmaligen Kauf von Scrip eines bestimmten Händlers auf sicherem Weg einen geheimen Kundenschlüssel. Beim Kauf sendet der Kunde jedoch zusätzlich zu seiner Anfrage und dem Scrip eine Signatur der Anfrage an den Händler. Diese Signatur wird auf dieselbe Art wie die weiter oben beschriebene Signatur des Scrip erzeugt, d.h. der Kunde wendet auf die Zeichenkette bestehend aus dem Scrip, der Anfrage und seinen Kundenschlüssel eine Einweg-Hashfunktion an. Das Resultat bildet die mitzuschickende Signatur.

Der Händler verfährt ähnlich wie bei den vorherigen Protokoll, d.h. er erhält aus den Scripinformationen den Kundenschlüssel und berechnet damit die seiner Meinung nach korrekte Signatur der Anfrage. Stimmt diese Signatur nicht mit der mitgeschickten überein, so liegt eine Manipulation des Script oder der Anfrage vor.

Das zurückgeschickte Wechselgeld enthält dieselbe `Cust_ID#` wie das von Kunden and den Verkäufer geschickte Scrip, der ursprüngliche Kundenschlüssel kann also auch beim Ausgeben des Wechselgeldes verwendet werden. Es besteht keine Notwendigkeit, irgendeine der Antworten zu verschlüsseln, da es für einen potentiellen Angreifer keinen Sinn machen würde, das Scrip zu stehlen, da er ohne Kenntnis des Kundenschlüssels die Signatur einer Anfrage nicht berechnen könnte. Dem Händler steht es offen, seine Antworten mit dem Kundenschlüssel zu signieren, um sich so gegenüber dem Kunden zu authentisieren.

Im Gegesatz zu dem vorherigen Protokoll reichen hier also ein paar Hashfunktionen, um recht sichere Transaktionen relativ effizient durchzuführen. Da für viele Millicent-Anwendungen ein voll verschlüsselter Kommunikationskanal gar nicht notwendig ist, bildet das **Secure without encryption**-Protokoll ein gutes Mittelmaß.

Kapitel 5

Digitale Kreditkarten

5.1 S-HTTP von Enterprise Integration Technologies

5.1.1 Motivation

Für viele Anwendungen ist die gegenseitige Authentikation von Client und Server und der vertrauliche Austausch sensitiver Daten wichtig. Für die meisten Anwendungen im World Wide Web (WWW) wird jedoch hauptsächlich das HyperText Transfer Protokoll (HTTP) verwendet, dessen ursprüngliche Spezifikation die Verwendung von Kryptoverfahren, die für oben genannte Anwendungen geeignet wären, nur in geringem Maße unterstützt. Deshalb arbeitet man an einer Erweiterung für HTTP, dem sogenannten Secure HyperText Tranfer Protokoll (S-HTTP), dessen gegenwärtiger Status als Internet-Draft[1] verfügbar ist.

S-HTTP ist so konzipiert, daß es zusammen mit HTTP benutzt wird und ohne größeren Aufwand in bestehende HTTP-Anwendungen integriert werden kann. Deshalb ähnelt S-HTTP sehr dem Stil und der Syntax von HTTP. S-HTTP ist kompatibel zu HTTP, d.h. S-HTTP-Clients können mit einem Server kommunizieren, der die S-HTTP-Erweiterung nicht kennt und umgekehrt, wobei solche Transaktionen natürlich nicht die Sicherheitserweiterungen von S-HTTP nutzen können.

S-HTTP soll eine Reihe individuell anpaßbare Mechanismen für vertrauenswürdige Transaktionen, Prüfung der Echtheit, Gewährleistung der In-

[1]Entwürfe der Internet Engeneering Task Force (IETF) mit einer Gültigkeit von sechs Monaten, die jederzeit überarbeitet oder durch neue Dokumente ersetzt werden können

tegrität und zur Verhinderung des Leugnens der Herkunft zur Verfügung stellen. Hiermit würde es die nötigen Sicherheitsoptionen für einen großen Kreis möglicher Anwendungen im World Wide Web liefern, zum Beispiel das Durchführen spontaner kommerzieller Transaktionen.

5.1.2 Grundidee

Da es viele unterschiedliche Verfahren zum Erreichen oben angesprochener Punkte wie Intergrität, Prüfung der Echtheit usw. gibt, man mit S-HTTP jedoch maximale Flexibilität erreichen möchte, handeln die Parteien für jede Transaktion eine Reihen von Optionen aus. Diese beinhalten zum Beispiel die Wahl der Sicherheitsvorkehrungen, der zu verwendenden Verschlüsselungsverfahren und die Art der Beglaubigung.

So könnte man beispielsweise festlegen, ob man die Anfrage signiert oder verschlüsselt oder sogar beides, und ob dies auch für die Antwort gelten soll. Auerdem könnte man die Wahl zwischen RSA oder DSA für das Signieren und DES oder RC2 für die Verschlüsselung haben. Für die Signatur der Anfrage könnte man vom Anbieter gebeten werden, seine vorhandenen Mitgliedsnummer zu benutzen.

S-HTTP kann die Vorteile einer vorhandenen Public-Key-Infrastruktur nutzen, ist aber nicht darauf angewiesen, d.h. es setzt nicht die Existenz eines Klientenschlüssels mit öffentlichem und geheimem Teil voraus, es unterstützt auch die Verwendung ausschließlich symmetrischer Schlüssel. Dies ist ein wichtiger Punkt, denn er bedeutet, daß spontane private Transaktionen stattfinden können, ohne daß sich vorher beide Parteien einen öffenlichen (und einen geheimen) Schlüssel zulegen müssen.

Um eine S-HTTP-Nachricht zu erstellen, benötigt der Versender drei Eingaben:

1. Die Nachricht im Klartext. Dies kann entweder ein HTTP-Text oder ein anderes Datenobjekt sein.

2. Die kryptographischen Präferenzen und das Schlüsselmaterial des Empfängers. Wird hier vom Empfänger nichts explizit festgelegt, so werden seine Standard-Vorgabewerte verwendet.

3. Die kryptographischen Präferenzen und das Schlüsselmaterial des Versenders.

Die Präferenzen des Versenders und des Empfängers werden verglichen und es entsteht eine Liste möglicher anwendbarer Verfahren und zu benutzender Schlüssel. An dieser Stelle bedarf es eventuell einer Entscheidung seitens des Versenders, z.B. wenn es mehrere Schlüssel zur Auswahl gibt. Nachdem alle Maßnahmen festgelegt worden sind, werden diese Verfahren auf den Klartext angewendet und es entsteht eine S-HTTP-Nachricht.

Um die S-HTTP-Nachricht wieder zu entschlüsseln, benötigt der Empfänger vier Eingaben:

1. Die S-HTTP-Nachricht.
2. Die vom Empfänger an den Versender mitgeteilten kryptographischen Präferenzen und sein Schlüsselmaterial.
3. Die momentanen kryptographischen Präferenzen und das Schlüsselmaterial des Empfängers.
4. Die vom Versender verwendeten Kryptoverfahren und das entsprechende Schlüsselmaterial.

Aus der Kopfzeile der S-HTTP-Nachricht erfährt der Empfänger, welche Kryptoverfahren und welche Schlüssel der Versender auf die Nachricht angewendet hat. Mit Hilfe dieser Informationen und einer Kombination des Schlüsselmaterials des Versenders und des Empfängers transformiert er die Nachricht zurück in den ursprünglichen Text.

Zum Schutz der Nachricht stehen einem Verschlüsselungsverfahren, eine Signatur und die Authentikation zur Verfügung. Man kann diese Verfahren einzeln, als Kombination oder gar nicht auf die Nachricht anwenden.

Authentikation und Integrität kann man durch das Berechnen eines Message-Authentication-Codes (MAC) erreichen. Hierzu wird auf den Nachrichtentext und einen als Zeichenkette angehängten geheimen Schlüssel eine Hashfunktion angewendet.

Beim Schlüsselmanagement hat man mehrere Möglichkeiten zur Auswahl, z.B. ein über das Public-Key-Verfahren ausgetauschter Schlüssel, ein Kerberos-Ticket oder ein manuell ausgetauschter paßwortähnlicher Schlüssel.

Außerdem hat jede Partei die Möglichkeit, ein Challenge-Response-Verfahren durchzuführen, um so die Aktualität der Transaktionen zu überprüfen.

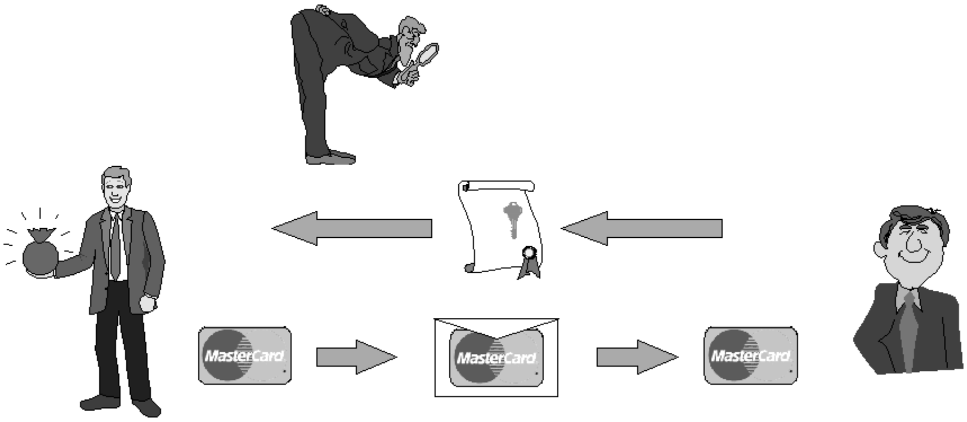

Abbildung 5.1: SHTTP

5.1.3 Realisierung

Aus Kompatibilitätsgründen zu bestehenden HTTP-Implementationen werden alle S-HTTP-Nachrichten mit einen Bezeichner gekennzeichnet (zur Zeit "Secure-HTTP/1.2"). Alle zur Vereinbahrung der Optionen notwendigen Befehle sind in einer einzigen Kopfzeile der Nachricht zusammengefaßt.

Aushandeln der Optionen

Wie schon erwähnt, haben beide Parteien die Möglichkeit, die gewünschten S-HTTP-Sicherheitsoptionen in der Kopfzeile der Nachricht anzugeben. Welche Optionen man wählt, hängt von den Möglichkeiten des Systems und von den Anforderungen der Anwendung ab. Die Kopfzeile zum Aushandeln der Optionen ist eine Sequenz einzelner Spezifikationen, die sich wiederum in vier Teile gliedert:

1. `Property` - Die Option, um die es geht, z.B. den Block-Verschlüsselungsalgorithmus

2. `Value` - Der Wert für die Option, z.B. DES-CBC

3. `Direction` - Die betroffene Richtung (`orig, recv`), also beim Verschicken oder beim Beantworten (vom Standpunkt des Versenders)

4. `Strength` - Die Notwendigkeit der Option, also gefordert, wahlweise oder abgelehnt (`required, optional, refused`)

Für ein Paar aus `Direction` und `Strength`, z.B. "`recv-required`" würde dies bedeuten, daß der Empfänger keine Nachrichten bearbeitet, auf die nicht die in den Feldern `Property` und `Value` beschriebenen Verfahren angewendet wurden. Hingegen würde "`orig-refused`" bedeuten, daß der Versender keine Nachrichten mit dem entsprechenden Verfahren generiert.

Für alle Optionen existieren Standard-Vorgabewerte, die automatisch eingesetzt werden, wenn sie nicht explizit vom Benutzer überschrieben wurden.

Stellt man fest, daß man mit einem inkompatiblen Partner kommuniziert, sendet man die Meldung "nicht implementiert" zurück oder bricht einfach die Verbindung ab.

Ein Server kann aber auch Fehlermeldungen zurücksenden, aus denen hervorgeht, daß die Anfrage nicht völlig fehlgeschlagen ist, und daß der Klient mit etwas anderen kryptographischen Optionen doch noch Erfolg haben könnte. So gibt es z.B. eine spezielle Kopfzeile, die benutzt wird, wenn eine HTTP-Anfrage gesendet worden ist, aber eine S-HTTP-Anfrage hätte gemacht werden sollen.

S-HTTP bietet auch die Möglichkeit, Nachrichten zu verschicken, die schon im Voraus signiert oder verschlüsselt worden sind.

Ein Beispiel hierzu wäre ein Händler, der ständig seinen Produktkatalog oder seine Preisliste verschickt. Der Server signiert das Dokument also einmalig und verschickt jedesmal, wenn eine Anfrage kommt, das Dokument aus dem Zwischenspeicher und spart so jedesmal eine Operation mit geheimem Schlüssel. Dies funktioniert, da der Kunde durch die Signatur sicher ist, daß es sich um die gültigen Preise handelt, Vertraulichkeit jedoch nicht gefordert ist, da alle Kunden die gleichen Preise erhalten. Diese Dokumente können sogar im Zwischenspeicher anderer Server abgelegt werden, die den Schlüssel zum Signieren nicht kennen. Auf diese Weise kann man Dokumente mit Echtheitsbeweis sogar über nicht vertrauenswürdige Server verteilen.

5.2 Netscape SSL

5.2.1 Motivation

Angesichts der Tatsache, daß Nachrichten im Internet mit den heute benutzten Protokollen im Klartext übertragen werden, ist das Senden von Kreditkarteninformationen gleichbedeutend mit dem Senden eines Blankoschecks als Postkarte.

Wohl als erster hat Philip Zimmermann in [Zimmerman 94] auf diese Gefahren hingewiesen, auch wenn es ihm eher allgemein um elektronische Privatsphäre als speziell um die Sicherheit elektronischer Transaktionen ging. Sein Verschlüsselungsprogramm PGP hatte aber den Nachteil, nicht einfach genug bedienbar zu sein, um wirkliche Beachtung außerhalb der akademischen Welt zu finden. Es ist einem Durchschnittsbenutzer einfach nicht zuzumuten, jede Nachricht, die vertraulich verschickt werden muß, vorher mit einem eigenen Programm zu verschlüsseln.

Netscape gebührt wohl die Ehre, die Problematik erkannt und auf eine Weise gelöst zu haben, die einfache Bedienbarkeit mit einer brauchbaren Sicherheit verbindet. Ihr „Secure Socket Layer“ [SSL 96] ist ein Protokoll zur Verschlüsselung von Übertragungen, das im Gegensatz zu PGP leicht in die Übertragung eingeschaltet werden kann, ohne daß der Benutzer viel davon bemerkt.

Es soll

- kryptographische Sicherheit
- Interoperabilität[2]
- leichte Erweiterbarkeit
- Effizienz, insbesondere im Hinblick auf die hohen Rechenzeiten kryptographischer Verfahren

bieten.

Da Netscape auch den marktbeherrschenden Browser „Netscape Communicator" herstellt, war der Erfolg vorprogrammiert. Es scheint nicht weit hergeholt, daß SSL noch eine ganze Weile als quasi - Standard für sichere Übertragungen gelten wird. Im Moment wird es von American Express für Geldgeschäfte empfohlen, bis ausgefeiltere Konzepte (z.B. SET s. Kapitel 6.3) zur Verfügung stehen, die spezieller auf die Bedürfnisse der Geldwirtschaft eingehen.

5.2.2 Grundidee

Die Grundidee bei der Entwicklung von SSL war es, ein Protokoll zu schaffen, das, im Gegensatz zu PEM (Privacy Enhanced Mail), Secure Shell oder SHTTP (s. Kap. 5.1), nicht ein bestehendes Protokoll abändert um es sicherer zu machen, sondern zwischen einem Transportprotokoll (z.B. TCP) und einem Anwendungsprotokoll(z.B. HTTP) angesiedelt ist, ohne daß besagte Protokolle dafür geändert werden müssen.

Kryptographische Sicherheit definiert Netscape als

- eine private Verbindung, die nach dem Aushandeln eines geheimen Schlüssels durch symmetrische Verfahren wie den DES oder RC4 geschützt werden.

[2]Interoperabilität bedeutet, daß wenn zwei Programmierer zwei unabhängige Implementationen einer Protokollspezifikation schreiben, diese kommunizieren können, ohne daß dazu die Kenntnis des Codes der jeweils anderen Implementation nötig ist. Dies bedeutet aber in diesem Zusammenhang nicht, daß eine Verbindung in jedem Fall zu Stande kommt. Es ist völlig legal, wenn z.B. ein Server eine Verbindung verweigert, weil der Klient nicht über eine Hardware verfügt, die der Server zwingend vorschreibt. Dies könnte eine Chipkarte oder ein ähnliches Zugangstoken sein, für das der Klient keinen Anschluß besitzt.

- eine Authentisierung des Gegenüber mit Hilfe von Public Key Verfahren wie z.B. RSA oder DSS.

- eine verläßliche Verbindung, bei der die Integrität der Nachricht durch eine MAC geschützt wird.

Um auch Erweiterbarkeit zu erreichen, ist SSL als Rahmen konzipiert, in den neue kryptographische Verfahren leicht eingebunden werden können. Im Moment können neue Verfahren definiert werden, wenn zwei unabhängige Parteien notariell beglaubigte Briefe an Netscape Communications Corp. schicken. Es ist aber geplant, daß diese Aufgabe abgegeben wird, sobald ein öffentliches Standardisierungsorgan die Kontrolle über SSL übernimmt.

Effizienz wird angestrebt, indem die Parameter einer Sitzung gespeichert werden, um später weiter verwendet werden zu können. Auf diese Weise kann der Aufwand beim Aushandeln dieser Parameter zu Beginn jeder Sitzung deutlich eigeschränkt werden. In gleicher Weise können auch während einer Sitzung neue Sitzungen angelegt werden, gleichfalls ohne neue Verhandlungen zu erfordern. Um auch den Netzverkehr während des Haupteils der Sitzung zu verringern, unterstützt SSL Kompression der übertragenen Daten.

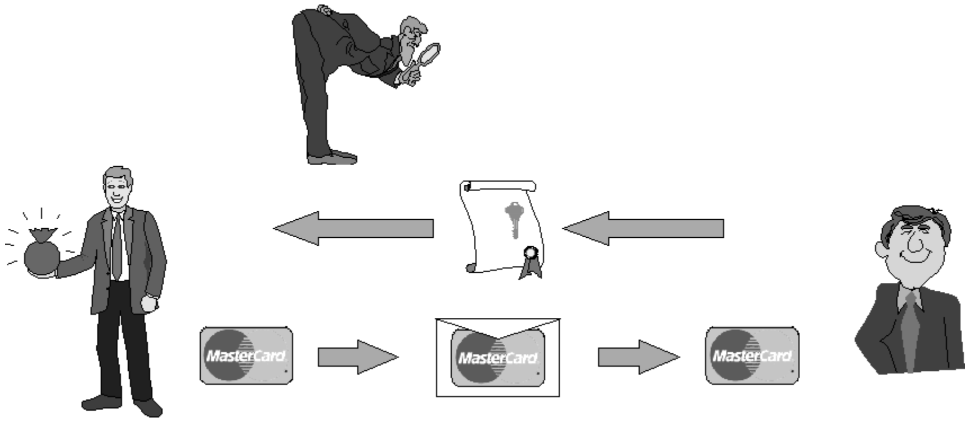

Abbildung 5.2: SSL

5.2.3 Realisierung

5.2.3.1 Grundlagen

Das SSL Protokoll ist in zwei Schichten aufgebaut. Dabei wird die untere Ebene von SSL Record Protokoll eingenommen, das dafür verantwortlich ist, Nachrichten der oberen Schicht in Blöcke aufzuteilen, sie optional zu komprimieren, eine MAC zu bilden, das Resultat zu verschlüsseln und zu übertragen.

5.2.3.2 Record Schicht

Sitzungszustände

Um die Daten der höherliegenden Schichten verarbeiten zu können, muß die Record Schicht den Zustand der jeweiligen Sitzung kennen. Zu diesem Zweck gibt es einen Zustand für jede Sitzung und jede Verbindung (eine Sitzung kann mehrere Verbindungen beinhalten).

Ein Sitzungsstatus enthält dabei

- eine Sitzungskennzahl, diese wurde willkürlich vom Server gewählt, um aktive oder fortführbare Sitzungen identifizieren zu können
- ein X509.v3 Zertifikat des Gegenübers
- eine Kompressionsmethode
- eine Chiffrenspezifikation (Angabe des Verfahrens, um die Daten zu verschlüsseln, sowie eines Verfahrens, um eine MAC zu bilden)
- ein Hauptgeheimnis (ein 48 Byte großes Datum, das zur Generierung von Schlüsseln zur Verschlüsselung, zur Bildung einer MAC und des Initialisierungsvektors für CBC Verschlüsselungen gebraucht wird)
- Ein Flag, das anzeigt, ob diese Sitzung weitere Verbindungen initiieren kann

Der Verbindungsstatus besteht aus

- Zufallszahlenfolgen, die von Server und Klient für jede Verbindung gewählt werden
- je ein Schlüssel zur Generierung von MACs für Server und Klient

- je ein Schlüssel zur Chiffrierung für Server und Klient
- Initialisierungsvektoren
- Paketfolgenummern

Es ist die Aufgabe des Verhandlungsprotokolls, diese Zustände in Klient und Server konsistent zu halten. Dabei existiert jeder Status während der Verhandlungsphase zweimal; einmal als momentan aktiver Status, und einmal als zukünftiger Status, der in den aktiven Status kopiert wird, wenn die Verhandlungen abgeschlossen sind.

Pakete

Die Record Schicht teilt die Daten der höherliegenden Schichten zuerst in Blöcke von maximal 2^{14} Bytes auf.

Danach werden die Daten mit der gerade aktiven Komprimierungsmethode komprimiert. Es ist immer eine Komprimierungsmethode aktiv, auch wenn dies zu Anfang die Methode „keine Komprimierung" ist. Komprimierungsmethoden müssen verlustfrei[3] arbeiten und dürfen das Paket nicht um mehr als 1024 Bytes verlängern[4].

Als nächstes wird eine MAC gebildet. Danach wird der gesamte Block inklusive MAC verschlüsselt. Es gibt drei Typen von Verschlüsselungen

- keine
- Stromchiffren [5]
- Blockchiffren

Dabei wird die MAC als

$$\begin{aligned}&\mathrm{hash}(k_1 + pad_2 + \\ &\quad \mathrm{hash}(k_1 + pad_1 + seq\_num + length + content))\end{aligned}$$

[3]Es gibt in der Bildverarbeitung Kompressionsverfahren (z.B. JPEG), die unwichtige Details auslassen. Dies wird dem Betrachter im Normalfall nicht auffallen. Für ein Protokoll mit dem aber u.U. Finanzdaten übertragen werden ist dies natürlich nicht zuläßsig.

[4]Um eine Datei zu komprimieren muß oft ein Header mit gespeichert werden. Dies verlängert die Datei, falls die Kompression gering ausfällt (weil die Datei z.B. schon vorher mit einem anderen Programm komprimiert worden war).

[5]eine Methode erzeugt eine fortlaufende Schlüsselfolge, die mit dem Eingabestrom xor'ed wird

berechnet. Dabei steht '+' für Stringkonkatenation, k_1 ist der Schlüssel aus dem Verbindungszustand, pad_1 das Zeichen 0x36 48 mal für MD5 bzw. 40 mal für SHA, pad_2 das Zeichen 0x5c genauso oft wiederholt und seq_num schließlich die Folgenummer dieser Nachricht. Bei der Verwendung von Blockchiffren (z.B. RC2, DES) müssen schließlich in der Regel noch Zeichen angehängt werden, damit die Länge des Textes ein Vielfaches des Blocklänge ist. Blockchiffren werden im CBC Modus verwendet. Dabei wird der erste Initialisierungsvektor vom Verhandlungsprotokoll festgelegt, die nachfolgenden Nachrichten nehmen jeweils den letzten Block der vorhergehenden Nachricht als Initialisierungsvektor.

5.2.3.3 Das Chiffrenänderungsprotokoll

Das Chiffrenänderungsprotokoll besteht aus einem einzigen Byte mit dem Wert 1. Es dient dazu, den Übergang zu einer neuen Chiffrenspezifikation anzuzeigen. Es wird von beiden Parteien gesendet um anzuzeigen, daß die folgenden Nachrichten mit der neuen Spezifikation geschützt werden. Nach dem Senden bzw. dem Empfang einer solchen Nachricht wird der zukünftige Zustand in den aktiven überführt.

5.2.3.4 Alarm Protokoll

Um auf das Auftreten von Fehlern reagieren zu können, unterstützt SSL Alarm - Nachrichten. Diese werden genauso komprimiert und verschlüsselt wie andere Nachrichten auch, bestehen aber nur aus zwei Werten, von denen einer angibt, ob es sich um eine Warnung oder einen terminierenden Fehler handelt, während der zweite den Grund angibt.

Terminierende Fehler bewirken das Beenden der Verbindung. Alle zu dieser Verbindung gehörenden Schlüssel und Geheimnisse sind aus dem Speicher zu löschen.

Andere Verbindungen der selben Sitzung dürfen offen bleiben. Damit sie aber nicht dazu benutzt werden können, ihrerseits neue Verbindungen zu eröffnen, ist die zugehörige Sitzungskennzahl ebenfalls zu löschen.

Zu den terminierenden Fehlern gehören der Empfang einer unerwarteten Nachricht, eine falsche MAC, Nachrichten, die nach der Dekompression zu groß werden, die Unmöglichkeit während der Verhandlung eine ausreichende Sicherheit zu vereinbaren, sowie das Autreten von Feldern in einer Verhandlungsnachricht, die keinen Sinn ergeben.

Ob das Fehlen eines gültigen Zertifikats eines Typs, den der Gegenüber kennt, die Verbindung abbricht, schreibt das Protokoll nicht vor.

Zuletzt gibt es noch eine Meldung zur Mitteilung des bevorstehenden Endes einer Verbindung. Unterbleibt sie, kann die Sitzung nicht mehr fortgeführt werden.

5.2.3.5 Das Verhandlungsprotokoll

Die Verhandlung beginnt mit dem Senden einer „Hallo“ Nachricht durch den Klienten. Diese enthält

- benutzte SSL Version
- gegenwärtige Zeit und Datum im UNIX Format
- 28 Bytes generiert von einem sicheren Zufallszahlengenerator
- eine Sitzungskenzahl (leer, wenn nicht eine alte Sitzung wiederaufgenommen werden soll)
- eine Liste unterstützter Chiffrierungsverfahren
- eine Liste unterstützter Komprimierungsverfahren

Ist kein Fehler aufgetreten, wird der Server mit einer eigenen „Hallo“ - Nachricht antworten. Sie wird die höchste SSL Versionsnummer enthalten, die beide unterstützen.

Wenn der Server einverstanden ist, eine alte Sitzung fortzuführen, wird er die vorgeschlagene Sitzungskennzahl zurücksenden, andernfalls eine neue vorschlagen. Sitzungskennzahlen dürfen dabei keine geheimen Informationen enthalten.

Die zurückgesendeten Listen von Chiffrierungs- und Komprimierungsverfahren enthalten jeweils nur ein Element.

In der Regel wird der Server als nächstes sein Zertifikat senden. Darauf kann allerdings verzichtet werden, wenn der Server nicht authentisiert werden soll.

Wurde kein Zertifikat gesendet oder enthielt das Zertifikat keinen Schlüssel[6], so folgt nun noch eine Nachricht mit den nötigen Informationen zu den zu verwendenden Schlüsseln (selbstverständlich digital signiert und public key verschlüsselt).

[6]Es gibt Zertifikate, die nur einen Schlüssel zum Signieren zertifizieren, aber keinen Schlüssel zum Chiffrieren. Diese Unterscheidung, die bei Public Key Verfahren ziemlich willkürlich ist, liegt ebenfalls an der amerikanischen Exportpolitik. Public Key Verfahren

Wurde der Server authentisiert, kann er auch noch ein Zertifikat vom Klienten anfordern.

Nachdem er all diese Nachrichten gesendet hat, wird er eine „Hallo beendet“ Nachricht senden und auf die Antworten des Klienten warten.

Dieser wird nun, falls verlangt, sein Zertifikat oder eine entsprechende Fehlermeldung senden, seine Schlüsselinformation und, falls er ein Zertifikat geschickt hat, eine digitale Unterschrift über alle vorhergehenden Nachrichten beginnend mit der ersten „Hallo“ Nachricht.

An diesem Punkt angelangt wird er nun eine Chiffrenänderungsnachricht schicken, so daß die nun folgende „Ende“ Meldung, die noch einmal je einen MD5 und einen SHA Hash über das Hauptgeheimnis[7], die vorhergehenden Meldungen und die Identifikatoren der beteiligten Parteien enthält, schon verschlüsselt gesendet wird.

Der Server sendet nun seinerseits wieder eine Chiffrenänderungs - und eine „Ende“- Nachricht, womit die Verhandlung beendet ist und die Übertragung der Anwendungsdaten beginnen kann.

Wurde allerdings eine alte Sitzung wiederaufgenommen, so sendet der Server gleich nach seiner „Hallo“ - Meldung eine Chiffrenänderungs - und eine „Ende“ - Nachricht. Die Klient antwortet in gleicher Weise.

5.2.3.6 Sicherheitshinweise der Entwickler

Im Anhang von [SSL 96] befinden sich zwei Kapitel, die sich mit Hinweisen zur sicheren Implementation von SSL und einer Sicherheitsanalyse befassen.

Exportversionen

Exportversionen dürfen nur mit Public Key Verfahren verschlüsseln, deren Schlüssellänge 512 Bits beträgt. Diese seien relativ unsicher und nicht für Transaktionen mit hohen Beträgen oder Anwendungen, die Sicherheit über einen langen Zeitraum verlangen.

---

zum Chiffrieren dürfen maximal eine Schlüssellänge von 512 Bit verwenden, was vom kryptographischen Standpunkt nicht sehr viel ist. Verfahren für digitale Unterschriften unterliegen diesen Einschränkungen nicht.

[7]Dieses wird aus dem mit der Schlüsselinformation gesendeten Initialisierungsgeheimnis sowie den Zufallszahlen aus den „Hallo“ - Nachrichten berechnet. Bei der Wiederaufnahme einer Sitzung wird das alte Hauptgeheimnis zum Initialisierungsgeheimnis, womit das Hauptgeheimnis auch bei der Wiederaufnahme einer Sitzung einen neuen Wert erhält. Aus ihm werden die eigentlichen Schlüssel für die Verbindung berechnet.

Da der Schlüssel zum reinen Signieren aber beliebig lang sein darf, schlagen sie vor einen temporären 512 Bit Schlüssel zu beutzen, der mit einem längeren Schlüssel signiert wird.

Dieser temporäre Schlüssel müßte dann jeden Tag oder alle 500 Transaktionen gewechselt werden, wobei die Generierung des neuen Schlüssels ein ziemlich zeitaufwendiger Prozess sei, der am besten durch einen Hintergrundprozeß erledigt würde. So könne nach jeder Erzeugung eines neuen Schlüssels gewechselt werden.

Auch die 40 Bit Begrenzung für die symmetrischen Verfahren wird von den Autoren immer wieder als potentielles Sicherheitsrisiko genannt.

> „Die 40 Bit Verschlüsselung läßt sich zum Beispiel recht einfach knacken, daher sollten Anwendungen, die eine starke Sicherheit verlangen, keine 40 Bit Schlüssel erlauben.“[SSL 96]

Pseudozufallszahlengeneratoren

Nicht nur im Protokoll selber, sondern in der Regel auch bei der Generierung von Schlüsseln spielen Zufallszahlen eine große Rolle. Da normalerweise keine wirklich zufälligen Zahlen zur Verfügung stehen, muß man sich mit algorithmischen Verfahren zufrieden geben, die eine scheinbar zufällige Folge von Zahlen generieren. Obwohl solche Verfahren mittlerweile recht sicher sind, sind sie wertlos, wenn der Anfangswert der Berechnung bekannt ist. Wird dieser zu sorglos gewählt, ist die gesammte Kommunikation kompromittiert. Auch gilt es zu berücksichtigen, daß so ein Verfahren nur eine endliche Anzahl von Zuständen hat, daß sich die Folge also nach einer gewissen Zeit wiederholt. Dem ist mit regelmäßigem Neuinitialisieren vorzubeugen.

„Mann in der Mitte“ Angriffe

Läßt man die Probleme mit der Schlüssellänge einmal außer Acht, so kann man sich vor passiven Lauschern relativ sicher fühlen. Hat man es aber mit einem Angriff zu tun, bei dem sich der Angreifer auf der Transportebene in die Kommunikation einschalten und Pakete unterdrücken, umsortieren, änndern oder austauschen kann, so entstehen neue Risiken.

So erlaubt SSL z.B. den Aufbau einer anonymen Sitzung, bei der nicht einmal der Server authentisiert wird. In diesem Fall könnte der Angreifer ohne Probleme den beiden Parteien seinen öffentlichen Schlüssel als den der jeweiligen Gegenpartei unterschieben. Da die öffentlichen Schlüssel auch dazu dienen, den symmetrischen Schlüssel für die spätere Übertragung zu schützen, hat er es geschafft: er kann nun nach Belieben in die Kommunikation eingreifen. Als einzigen Ausweg bietet sich nur, anonyme Sitzungen grundsätzlich zu verbieten und zu mindest einen authentisierten Server zu verlangen.

Eine andere Gefahr wird darin gesehen, daß ein Angreifer die beteiligten Parteien dazu bringen könnte, statt Version 3 des Protokolls die alte Version 2 zu benutzen. Dem wird aber vorgebeugt, indem bei der Übermittlung der Schlüsseldetails ein Feld, das normalerweise Zufallsbits enthält, von einem Version-3-Klienten, der Version 2 benutzt, in der zweiten Hälfte auf 0x03 gesetzt wird. Ein Version-3-Server, der dies bemerkt, muß die Verbindung abbrechen. Da dieses Feld durch einen symmetrischen Schlüssel geschützt ist, muß der Angreifer, um eine Entdeckung zu vermeiden, diesen Schlüssel vor dem Timeout des Servers knacken und das Feld entsprechend verändern.

Bei diesem wie auch bei allen anderen Angriffen gegen das Verhandlungsprotokoll, dessen Nachrichten ja zum Teil im Klartext gesendet werden, muß der Angreifer allerdings das Hauptgeheimnis kennen, das am Ende für den Hash über die gesammte Kommunikation benutzt wird. Da dieses von dem Initialisierungsgeheimnis abhängt, welches wiederum von einem Public Key Verfahren geschützt wird, muß ihm auch der geheime Schlüssel desjenigen, der der Empfänger der manipulierten Nachricht sein soll, bekannt sein.

Wiederaufnahme von Sitzungen

Bei der Wiederaufnahme einer Sitzung wird eine neues Hauptgeheimnis aus Hashoperationen über dem alten Hauptgeheimnis und den Zufallszahlen aus den „Hallo“ - Nachrichten gebildet. Aus dem neuen Hauptgeheimnis werden nun wiederum mittels mehrerer Hashoperationen neue Schlüssel gebildet, so daß die Kenntnis der alten Schlüssel dem Angreifer keinen Anhaltspunkt zur Berechnung der neuen Schlüssel geben sollte.

Um die Gefahr der Kompromittierung des Hauptgeheimnisses gering zu halten, wird vorgeschlagen, daß eine Sitzungskennzahl nur eine Lebensdauer von 24 Stunden haben sollte.

Schutz der Anwendungsdaten

Da die Anwendungsdaten mit einer MAC geschützt sind, die auch über die Nachrichtenfolgenummer und die Nachrichtenlänge gebildet wird, in jeder Richtung ein anderer Schlüssel benutzt wird, und die MAC innerhalb des verschlüsselten Bereichs liegt, ist zum aktiven Eingreifen in den Nachrichtenstrom sowohl die Kenntnis des MAC-Schlüssels als auch des Schlüssels für die eigentliche Verschlüsselung nötig.

Zum Lesen der Nachrichten reicht es, den Schlüssel für die eigentliche Verschlüsselung zu knacken.

Da MAC-Schlüssel länger als Verschlüsselungsschlüssel sein dürfen, kann die Situation entstehen, daß ein Angreifer zwar über die Mittel verfügt, Nachrichten mitzulesen, aber nicht in der Lage ist, sie zu manipulieren.

5.2.3.7 Aufbauende Standards und Implementation

https, smtp und snntp

Für HHTP hat man „https“ als eigenes Protokoll eingeführt. Wobei dies aber nur bedeutet, den Port 443 im Gegensatz zu 80 beim normalen HTTP zu benutzen. Funktionell bedeutet „https“ die Ausführung von HTTP über SSL. Wie [Netscape 96a] ausführt, hat dies den Vorteil, daß man für jede HTML - Seite entscheiden kann, ob sie gesichert übertragen werden soll, oder ob ihr Inhalt keinerlei Schutz benötigt. So könne ein Händler seine Homepage und Kataloge ungeschützt vertreiben, für die Übertragung der Formulare mit der Kreditkartennummer aber eine URL[8] verwenden, die mit „https“ beginnt und so das SSL Protokoll aufruft. Da ein Browser, der SSL nicht unterstützt, auf „https“ URLs nicht zugreifen kann, wird er nicht versuchen, Formulare unsicher zu übertragen, deren Inhalt eine sichere Übertragung erfordert.

Auf gleiche Weise hat Netscape beantragt, 465 als Port für ssmtp und 563 als Port für snntp zu reservieren. Dabei wäre ssmtp SMTP (Simple Mail Transfer Protocol) mit SSL, während snntp für NNTP (Network News Transfer Protocol) mit SSL steht.

Netscape Communicator

Der Communicator unterstützt das SSL Protokoll seit der Version 0.93. Um eine Exportgenehmigung der Vereinigten Staaten zu bekommen, benutzt er als Verschlüsselungsmethode RC4 mit 40 Bit Schlüssellänge. Netscape gibt allerdings an, es brauche im Mittel 64 MIPS - Jahre, eine Nachricht zu

[8]**URL**: **U**niformous **R**esource **L**ocator, eine Angabe bestehend aus Protokoll, Rechner, sowie weiteren optionalen Informationen, die es erlauben, eine Anfrage an einen Rechner zu schicken. Eigentlich ist sie dazu gedacht, eindeutig anzugeben, wo ein Dokument auf einem Rechner liegt, aber die Definition einer URL ist allgemein genug, daß auch Suchanfragen oder ausgefüllte Zahlungsformulare auf diesem Weg verschickt werden können.

knacken, die mit dieser Methode verschlüsselt sei. Dies sei zwar keine Sicherheit nach den Anforderungen des Militärs, aber für normale Anwendungen durchaus ausreichend.

In den USA gibt es schon eine Version, die RC4 mit 128 Bit Schlüssellänge unterstützt. Durch die schon erwähnten Exportbeschränkungen wird sie aber nicht in absehbarer Zeit außerhalb der USA verfügbar sein.

Zum gegenwärtigen Zeitpunkt werden für die Authentisierung nur Serverzertifikate unterstützt. Zertfikate für Klienten gibt es noch nicht. Zertifikate verschiedener Zertifizierungsstellen werden mit dem Navigator mitgeliefert, so daß die Zertifikate des jeweiligen Servers auch effektiv überprüft werden können.

Der Benutzer kann an einem „Sicherheits - Farbbalken" am oberen Bildrand und einem Icon in der unteren linken Ecke erkennen, ob das gerade angezeigte Dokument verschlüsselt übertragen wurde oder nicht. Will er ausführlichere Informationen, so kann er über das Datei - Menü ein Informationsfenster öffnen, das ihm die verwendete Übertragungsmethode sowie die relevanten Angaben aus dem Zertifikat des Servers anzeigt. Zusätzlich erscheinen Dialogboxen, wenn

- eine gesicherter Raum betreten wird,
- ein gesicherter Raum Verlassen wird,
- ein Formular unsicher übermittelt wird oder
- ein sicher übertragenes Dokument unsicher übertragene Graphiken enthält.

Diese Dialogboxen können aber auf Wunsch auch abgestellt werden.

Bekanntgewordene Angriffe

Bereits zweimal zeigten Mitglieder der Mailing-Liste der „Cypherpunks", daß SSL zum einen durch die Exportbeschränkung auf 40 Bit Schlüssellänge, zum anderen durch Implementationsschwächen im Netscape Navigator, keine perfekte Sicherheit vor ambitionierten Amateuren bietet.

Hal's Herausforderungen

Am 14.Juli 1995 veröffentlichte Hal Finney eine Netscape Sitzung, die unter Benutzung der RC4 Verschlüsselung mit 40 Bit Schlüssellänge „gesichert" war. Er forderte die Cypherpunks auf, diese zu knacken.

Am 15.August meldete Damien Doligez vom Institut National de Recherche en Informatique et en Automatique (INRIA) er habe mit Hilfe von 120 Workstations und drei Parallelcomputern die Sitzung in acht Tagen entschlüsselt. Dabei wurde etwa die Hälfte des Schlüsselraumes durchsucht.

Bei der Sitzung handelte es sich um die Bestellung eines gewissen Herrn Cosmic Kumquat, SSL Trusters Inc bei Netscape, die aber leider nicht zustande kam, weil er vergaß eine Telefonnummer anzugeben.

Er zieht daraus den Schluß, daß erbestimmt nicht der einzige sei, der Zugang zu einer Rechenleistung habe, erfülle das Protokoll nicht den Anspruch, schwach genug zu sein, um von Regierungen geknackt werden zu können, Amateuren jedoch zu wiederstehen. [Doligez 95]

Am 17.August stellte sich dann heraus, daß noch ein zweites Team die Sitzung geknackt hatte. Sie hatten ihre Erfolgsmeldung schon zwei Stunden vor Doligez an einen der Organisatoren geleitet, von diesem aber nicht schnell genug weiter geleitet worden. [Back 95]

Am selben Tag kam auch die offzielle Reaktion von Netscape [Netscape 96b]. Sie wiesen darauf hin, daß

1. nur eine Sitzung geknackt wurde[9], jede weitere Sitzung aber wieder neu geknackt werden müßte, was noch einmal acht Tage bedeute.

2. die Kosten für acht Tage Rechenzeit an zwei Paralelcomputern und 120 Workstations etwa 10.000 $ betrügen, jede Nachricht die weniger wert sei, also durchaus sicher sei.

3. innerhalb der USA ja 128 Bit lange Schlüssel zu Verfügung stünden.

In Kreisen der Cypherpunks wurde die Zahl von 10.000 $ stark bezweifelt. Außerdem wurden die Angriffe ja von Operateuren gemacht, die die freien Zeiten auf ihren Rechnern „zu Testzwecken“ nutzten, also die Rechenzeit nicht bezahlen mußten.

Um zu sehen wie schnell es wirklich ginge, organisierten die Cypherpunks einen „key cracking ring“. Dabei spendeten Besitzer von Computern im ganzen Internet Rechenzeit. Sie installierten auf ihrem Rechner ein Programm, das es erlaubte, koordiniert einen Teil des Schlüsselraumes zu durchsuchen.

Am 19. August schickte Hal Finney eine zweite Herausforderung. Der Ring begann am 24. August $18^{\underline{00}}$ GMT mit der Arbeit und brauchte weniger als 32 Stunden.

[9]Eigentlich ist die Rede von einer einzelnen verschlüsselten Nachricht. Aber das ist nachweislich falsch, es waren mehrere auf einander folgende Pakete.

Ein schlecht implementierter Pseudozufallszahlengenerator

Während das Problem mit den 40 Bit Schlüsseln nicht in die Verantwortung von Netscape fällt, sondern eindeutig der US Regierung zuzurechnen ist, stellte sich am 17. September 1995 heraus, daß sich auch Netscape selber Versäumnisse eingestehen muß.

Wie zwei Studenten, Ian Goldberg und David Wagner, bei dem Versuch feststellten, Teile des Programmcodes des Netscapenavigators zu rekonstruieren, wurde bei der Realisierung des Pseudozufallszahlengenerators (PRNG) ziemlich nachlässig gearbeitet.

Es ist nicht schwierig lange Folgen von Zahlen zu erzeugen, zwischen denen so einfach kein Zusammenhang zu erkennen ist. Das Verfahren dazu ist jedoch deterministisch, d.h. bei Kenntnis des Anfangswertes kann die ganze Folge einfach rekonstruiert werden. Werden die Zahlen für kryptographische Verfahren verwendet, so ist es daher unbedingt nötig, den Anfangswert so unratbar wie möglich zu wählen.

Genau hier lag das Problem des Navigators. Als Anfangswert wurden unter UNIX

- die Prozeß ID des Navigators (PID)
- die Prozeß ID des aufrufenden Prozesses (PPID)
- die Systemzeit inclusive der Mikrosekunden

Hinzu kommt, daß nach dem ersten Paket der PRNG nicht wieder mit einem neuen Anfangswert gestartet wird, so daß nach dem Knacken des ersten Paketes auch die Werte für neue Verbindungen (und damit Challenge-Response, Schlüssel…) vorhersagbar sind.

Nun aber zum konkreten Angriff. Die Systemzeit kann mit einer gewissen Genauigkeit geschätzt werden, nur die Mikrosekunden müssen durchprobiert werden. PID und PPID kann sich ein Angreifer anzeigen lassen, falls er Zugriff auf die Maschine seines Opfer hat.

Aber auch Angriffe von außerhalb sind möglich. Dazu muß man wissen, daß sendmail seine eigene PID als Teil der Message ID von Emails verwendet. Man braucht also nur sendmail direkt anzusprechen und eine Email bouncen zu lassen, um diese PID zu erfahren. Da PID's sequentiell vergeben werden, wird die Differenz zur PID von sendmail nicht allzu groß sein.

Um schließlich die PPID zu raten, kann man annehmen sie sei 1, was immer dann der Fall ist, wenn man Netscape von einem X-Windows Menü startet. Funktioniert dies nicht, hat man gute Chancen mit der Annahme, die PPID sei nur wenig kleiner als die PID.

Alles in allem dauert es nur wenige Minuten um alle Informationen zu bekommen, die es braucht, um alle geheimen Schlüssel des Opfers zu erfahren.

Wie sich später herausstellte, waren nicht nur UNIX-Versionen des Navigators betroffen. Auf PC- und Mac-Systemen wurde offenbar die Systemzeit als Startwert benutzt, eine Angabe die ähnlich unzufällig ist. Auch war nicht nur der Navigator betroffen, sondern auch der kommerzielle Server.

Netscape hat inzwischen den fraglichen Code von Sicherheitsexperten der RSA Inc. überprüfen lassen, eine Maßnahme, die zuvor aus Gründen der Geheimhaltung abgelehnt wurde. Auch wurde der strittige Teil der Quellen offengelegt. Ein Patch wurde umgehend zur Verfügung gestellt und die neuen Versionen werden diesen Fehler wohl nicht mehr aufweisen.

Bleibt noch anzumerken, daß Netscapes Versäumnis nicht in der laienhaften Implementation des PRNG lag, sondern in dem Glauben, Geheimhaltung bringe Sicherheit. Nur die gewissenhafte Überprüfung durch unabhängige Experten kann die Sicherheit eines kryptographischen Produktes garantieren. Übertriebene Geheimhaltung spielt dagegen demjenigen in die Hände, der das nötige Wissen oder die Mittel hat, sich die Informationen trotzdem zu beschaffen.

5.3 First Virtual

5.3.1 Motivation

Ziel von First Virtual war es, einer möglichst großen Anzahl von Benutzern einen sicheren Mechanismus für den Kauf und Verkauf im Internet zur Verfügung zu stellen. Damit praktisch jeder dieses Verfahren nutzen kann, sollte es relativ einfach zu bedienen sein. Man wollte deshalb auf spezielle Software und zusätzliche Hardware verzichten. Außerdem sollte kein Einsatz von Kryptoverfahren erfolgen, da er für manche Benutzer zu kompliziert erscheinen könnte und immer einen zusätzlichen Arbeitsschritt bedeutet. Zusätzlich sind diese Kryptoverfahren meist durch Patente, Urheberrecht oder Exportbestimmungen eingeschränkt.

Das nach diesen Anforderungen entwickelte System der First Virtual Holdings Inc. ist seit Oktober 1994 offiziell im Einsatz.

5.3.2 Grundidee

Das von First Virtual entwickelte Verfahren basiert auf dem bestehenden Kreditkartensystem und dem e-mail Mechanismus. Damit die Kreditkartennummern der Kunden nicht in das Internet gelangen, teilt First Virtual eine Transaktion in zwei Bereiche auf. Der eine Teil davon findet im "unsicheren" Internet statt, der andere Teil, der sensitive Daten wie z.B. die Kreditkartennummer verwendet, außerhalb des Netzes. Die Kunden sind bei First Virtual mit ihren Kreditkartendaten[10] und ihrer e-mail Adresse registriert und haben eine sogenannte "VirtualPin" erhalten. Möchte ein Kunde bei einem ebenfalls bei First Virtual registrierten Händler etwas kaufen, so schickt er ihm per e-mail zusammen mit seiner Bestellung anstatt seiner Kreditkartennummer seine VirtualPin (unverschlüsselt). Der Verkäufer schickt diese VirtualPin zusammen mit den Angaben über die Höhe des zu erhaltenden Betrages an First Virtual. First Virtual schickt jetzt wiederum dem Kunden eine e-mail, die den Namen des Händlers und die Höhe des geforderten Betrages enthält, und verlangt von dem Kunden eine Bestätigung. Der Kunde kann mit "Ja", "Nein" oder "Betrug" antworten, und nur, wenn der Kunde mit "Ja" geantwortet hat, wird die Kreditkarte des Kunden unter Verwendung des bestehenden Kreditkartensystems außerhalb des Internets durch First Virtual belastet. Der von First Virtual vom Kunden eingeforderte Betrag wird dann auf das Konto des Händlers überwiesen.

5.3.3 Realisierung

5.3.3.1 Registrierung

Um das Verfahren von First Virtual nutzen zu können, muß man sich zunächst registrieren lassen. Hierzu fordert man ein Anmeldeformular über das World-Wide-Web, per e-mail oder mit Telnet an. In dieses Formular trägt man unter anderem seine e-mail Adresse und den Namen ein, unter dem man seine Käufe bzw. Verkäufe tätigen will. Da es sich bei diesem Namen um öffentliche Daten handelt, ist auch ein Firmenname, Spitzname oder ein Pseudonym zugelassen. Außerdem trägt man in dieses Formular eine selbstgewählte Pin ein, die aus mindestens 8 Zeichen und/oder Zahlen bestehen muß. Diese Pin wird später zur VirtualPin, indem von First Virtual ein Präfix bestehend aus 4 Zeichen und/oder Zahlen und ein Bindestrich hinzugefügt wird. Durch das Hinzufügen dieses Präfix will man sicherstellen,

[10] zur Zeit werden Visa- und Mastercard akzeptiert

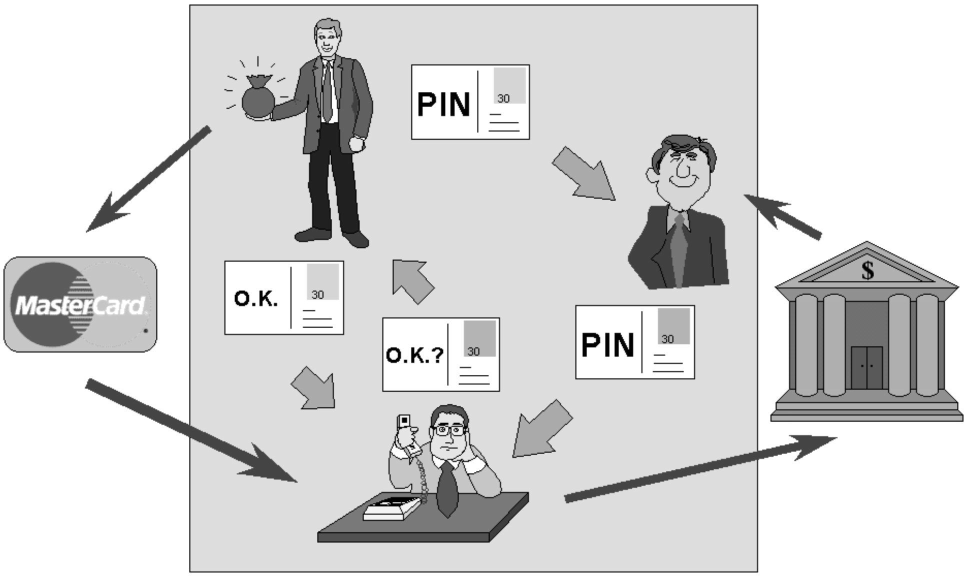

Abbildung 5.3: First Virtual

daß die VirtualPin eindeutig ist. Nachdem man dieses Anmeldeformular an First Virtual zurückgeschickt hat, erhält man kurze Zeit später eine e-mail, die ein zwölfstellige Bearbeitungsnummer und eine Telefonnummer enthält.

Möchte man sich als Käufer registrieren lassen, kann man jetzt mit Hilfe dieser Bearbeitungsnummer First Virtual seine Kreditkartendaten telefonisch mitteilen. First Virtual prüft daraufhin die Gültigkeit der Kreditkarte, erhebt eine Registrierungsgebühr (zur Zeit US $2.00) und schickt einem die aktivierte VirtualPin, die man jetzt im Internet anstatt seiner Kreditkartennummer verwendet, per e-mail.

Wenn man sich dagegen als Händler registrieren lassen möchte, schickt man einen Scheck über die Höhe der Registrierungsgebühr für Händler (zur Zeit US $10.00) per Post an First Virtual. Diesem Scheck wird die Nummer des Kontos entnommen, auf die später von First Virtual die Zahlungen der Kunden überwiesen werden. Auch als Händler erhält man seine aktivierte VirtualPin, die in diesem Fall zur Identifikation des Händlers und des zugehörigen Bankkontos dient, per e-mail.

In beiden Fällen ist also die VirtualPin, die man im Internet verwendet, der Bezug zur den außerhalb des Internets gepeicherten Daten, die für die Abwicklung des außerhalb des Netzes stattfindenden Zahlungsverkehrs notwendig sind.

5.3.4 Geldfluß

Im Gegensatz zu anderen Ansätzen für Zahlungsverfahrem ist bei First Virtual zu keinem Zeitpunkt "richtiges" Geld im Internet im Umlauf. Man benutzt das Internet lediglich, um autorisierte Zahlungsanweisungen zu verschicken, die außerhalb des Netzes in reale Geldtransaktionen umgesetzt werden.

Wie schon erwähnt, schickt ein Käufer zur Bezahlung seine VirtualPin an den Händler. Dieser schickt die Pin zusammen mit den Angaben über die Höhe des vom Käufer zu erhaltenden Betrages an First Virtual und fordert die Überweisung dieses Betrages auf sein Konto. Daraufhin wird von First Virtual zunächst automatisch eine e-mail an den Käufer geschickt, mit der Bitte, die Zahlung zu autorisieren. Der Käufer muß dann lediglich die "`reply`"-Funktion seines Mail-Reader-Programms zur Beantwortung benutzen. Diese Funktion sollte den Inhalt des "`subject field`" der erhaltenden Nachricht automatisch in das entsprechende Feld der Antwort kopieren, da dieses Feld einen eindeutigen Bezeichner für diese spezielle Autorisierungsanfrage enthält. Als Antwort muß der Käufer nur ein einziges Wort zurückschicken, entweder "`yes`", wenn die Zahlung erfolgen soll, "`no`", wenn er die Zahlung verweigert, oder "`fraud`", um einen Betrug zu signalisieren. Nur wenn die Antwort "`yes`" lautet, der richtige Bezeichner im "`subject field`" der Antwort steht, und die Nachricht die richtige e-mail Adresse als Absender hat, hat der Käufer die Zahlung korrekt autorisiert, und es kann eine Überweisung auf das Konto des Händlers stattfinden.

Diese Überweisung findet allerdings erst statt, wenn die Summe der vom Käufer zu zahlenden Beträge eine bestimmte Grenze (zur Zeit US $10.00) überschreitet, oder die letzte Transaktion länger als ein bestimmter Zeitraum (zur Zeit 30 Tage) zurückliegt. Auf diese Weise wird verhindert, daß die Kreditkartengesellschaft, deren Karte der Kunde zur Bezahlung seiner Käufe benutzt, ständig eine größe Anzahl kleinerer Beträge verrechnen muß.

Hat First Virtual das Geld über die Kreditkarte des Kunden erhalten, so wird zunächst ein Zeitraum von 91 Tagen abgewartet, bevor der Betrag auf das Konto des Händlers überwiesen wird. Die ist notwendig, da First Virtual das Risiko eines sogenannten "Chargeback"[11] nicht tragen will. Erfolgt ein

[11] Unter bestimmten Umständen erlaubt es die Kreditkartengesellschaft dem Kunden, seine Zahlung innerhalb von 90 Tagen zu widerufen.

solcher "Chargeback", beendet First Virtual sofort die Beziehung mit diesem Kunden, und seine VirtualPin wird deaktiviert.

In jedem Fall wird immer, wenn First Virtual eine Transaktion durchführt, d.h. die Kreditkarte eines Käufers belastet oder Geld auf das Konto eines Händlers überweist, an die entprechende Partei eine e-mail geschickt, die ihr eben dieses mitteilt.

Außerdem kann ein Kunde jederzeit Informationen über zurückliegende Transaktionen abfragen, indem er eine e-mail mit seiner VirtualPin im "`subject field`" an `history@card.com` sendet. Daraufhin erhält er eine Zusammenfassung darüber, was er wann zu welchem Preis gekauft hat. Diese Daten beziehen sich auf die letzten 60 Tage, nach Ablauf dieses Zeitraums werden die Daten gelöscht.

5.3.4.1 Schutz vor Betrug

Wird eine e-mail, die die VirtualPin eines Kunden enthält, von einer anderen Person abgefangen, so kann diese damit im Internet keine Waren kaufen, da alle Transaktionen von dem rechtmäßigen Besitzer der VirtualPin per e-mail autorisiert werden müssen, bevor die Kreditkarte belastet wird. Da der rechtmäßige Besitzer der VirtualPin die Transaktion nicht getätigt hat, die er autorisieren soll, wird er die Bitte um Autorisierung mit dem Wort "`fraud`" beantworten. Daraufhin deaktiviert First Virtual sofort die entsprechende VirtualPin, und wenn der Kunde First Virtual weiterhin nutzen möchte, muß er eine neu Pin beantragen. Außerdem erhält der Händler eine e-mail, die ihm mitteilt, daß keine Zahlung erfolgen wird, da die Transaktion vom Käufer nicht autorisiert wurde.

Das System von First Virtual war ursprünglich nur für den Verkauf von Informationen im Internet gedacht. Der Vorteil für den Käufer hierbei ist, daß er die Ware (Dokumente) sieht, bevor er sie bezahlt. Ist der Kunde mit der Ware unzufrieden, d.h. die Informationen sind für ihn unbrauchbar oder nicht vollständig (aufgrund eines Abbruchs der Kommunikationsverbindung), so antwortet er auf die Autorisierungsanfrage von First Virtual für die betreffende Zahlung mit "`no`". Daraufhin bekommt der Händler von First Virtual eine e-mail, aus der er entnehmen kann, daß First Virtual den geforderten Betrag nicht auf das Konto des Händlers überweisen kann, da der Käufer die Zahlung verweigert. Beim Verkauf von Informationen ist der Schaden für den Händler nicht zu groß, da z.B die Kosten für das Anfertigen einer Kopie eines Dokuments relativ gering sind. Beim Verkauf "physikalischer" Waren ergibt sich dieses Problem für den Händler nicht, da er seine Waren erst abschicken kann, wenn er von First Virtual benachrichtigt worden ist, das der Käufer seine Zustimmung für die Transaktion gegeben hat.

Versucht jedoch ein Käufer zu betrügen, indem er viele Autorisierungsanfragen mit "`no`" beantwortet, so wird First Virtual bei einer Häufung dieser Vorfälle die VirtualPin des betreffenden Käufers sperren und die Geschäftsbeziehung mit ihm beenden. Was in diesem Fall "zu oft" bedeutet, liegt im Ermessen von First Virtual. Beispielsweise kann schon die einmalige Verweigerung einer Zahlung zu oft sein, wenn es sich um einen hohen Betrag handelt.

Um sich vor Betrug zu schützen, kann der Händler vor dem Verkauf an den Kunden eine Anfrage an First Virtual schicken, mit der Bitte auf eine Gültigkeitsprüfung der VirtualPin des Käufers. Die Antwort von First Virtual, daß es sich um eine gültige Pin handelt, sagt dem Händler, daß für diese Pin ein Kunde mit einer gültigen Kreditkarte registriert ist, und die VirtualPin noch nicht aufgrund eines Betrugs durch "`fraud`" gesperrt worden ist. Diese Anfrage bietet jedoch keinen Schutz davor, daß die VirtualPin gerade jetzt von einem anderen als dem rechtmäßigen Besitzer der Pin verwendet wird. Außerdem gibt es auch keine Garantie, daß die Kreditkatengesellschaft First Virtual für die Käufe des Kunden bezahlt.

5.3.4.2 Gebühren

Zur Zeit belaufen sich die Gebühren für einen Käufer auf US $2.00, die er zu zahlen hat, wenn er sich das erste Mal registrieren läßt, und jedes Mal, wenn sich seine Kreditkartendaten ändern. Sonst entstehen für den Käufer keine Kosten.

Ein Händler zahlt bei der Registrierung einmalig US $10.00 und für jede stattfindende Transaktion US $0.29 plus 2% des Verkaufspreises. Außerden entsteht eine Gebühr von US $1.00, wenn die bei First Virtual gesammelten Einkünfte des Händlers auf dessen Konto überwiesen werden.

5.3.4.3 Schutz vor Angriffen von außen

Die an das Internet angeschlossenen Computer sind durch eine extreme Form einer Firewall von dem Netzwerk getrennt, das die finanziellen Transaktionen bearbeitet, und auf dem die sensitiven Daten (Kreditkartendaten der Käufer und Kontonummern der Händler) gespeichert sind. Es existiert eine klare Grenze zwischen dem Internet und dem Finanz-Netzwerk. Die Verbindung der beiden Netze ist so minimal wie möglich gehalten und wird streng überwacht. Eine Kommunikation findet jeweils immer nur zwischen einer einzelnen an das Internet angeschlossenen Maschine und einer einzelnen Maschine auf der anderen Seite der Grenze statt, wobei die an das Internet angeschlossene Maschine hierbei der einzige Host in ihrem Subnetz ist. Als zusätzlicher Schutz ist die Kommunikation ausschließlich batch-orientiert und benutzt keine Standard-Internetdienste.

5.3.4.4 Ausblick

Das System wird ständig weiterentwickelt, um den Verkauf "physikalischer" Waren und Dienste besser zu unterstützen, da das System ursprünglich nur für den Verkauf von Informationen konzipiert worden ist. Diese Erweiterung beinhaltet die Verwendung kryptographischer Verfahren für die Authentikation bestimmter kritischer Nachrichten, die von First Virtual an die Händler geschickt werden.

Außerdem soll das System besser international angepaßt werden, d.h. es soll unterschiedliche Währungen und Sprachen unterstützen. Weiterhin soll es mehr Möglichkeiten für die Kunden geben, ihre Käufe zu bezahlen, und zusätzliche Möglichkeiten für die Händler, diese Zahlungen zu erhalten.

Geplant ist auch eine bessere Unterstützung von extrem kleinen Transaktionen, sogenannten "Micropayments".

Kapitel 6

Digitale Schecks

6.1 MPTP

6.1.1 Motivation

Wie viele andere zuvor macht sich auch das World Wide Web Consortium (W3C) Gedanken über einen Standard für ein Verfahren, um Informationen über das Internet zu verkaufen.

Während bei materiellen Gütern die Kosten für den Transfer des zu bezahlenden Betrages vernachlässigbar ist, spielen sie bei Informationen, die in Pfenningen pro Seite berechnet werden, eine entscheidende Rolle. Hinzu kommt, daß auch die Abwicklung der Bezahlung schnell erfolgen muß, wenn man z.B. abgerufene WWW - Seiten einzeln berechnet. Eine zusätzliche zeitliche Verzögerung beim Abruf der Seiten würde den Profit deutlich mindern, da ja weniger Seiten pro Zeit umgesetzt werden könnten.

Aus diesen Überlegungen folgt, daß

1. Online - Berechnungen minimert werden müssen,
2. Offline - Berechnungen nach Möglichkeit durch den Kunden geschehen sollten, da dieser den geringsten Durchsatz hat,
3. Zugriffe auf externe Speicher nach Möglichkeit vermieden werden sollten,
4. die Abrechnung mit der Bank Offline geschehen muß.

Das Ergebnis dieser Forderungen ist das Micro Payment Transfer Protocol [MPTP 95] oder MPTP, das am 22.November 1995 erstmals als W3C

Working Draft veröffentlicht wurde. Wie die Bezeichnung „Working Draft“ schon sagt, handelt es sich hierbei um einen Entwurf, der sich noch ändern kann. Die grundlegenden Eigenschaften sind aber schon erkennbar und sollen im folgenden besprochen werden.

6.1.2 Grundidee

Bevor ein Käufer eine Geschäftsbeziehung mit einem Händler eingeht, druckt er eine Ticketrolle und schreibt eine Zahlungsvollmacht. Die Ticketrolle kann man sich so vorstellen wie die Rollen mit Eintrittskarten, die früher im Kino, Zirkus oder touristischen Attraktionen verwendet wurden. Jedes dieser Tickets ist mit einer fortlaufenden Nummer, und der Nummer der Zahlungsvollmacht bedruckt.

Will der Käufer nun bezahlen, sendet er dem Händler das erste Ticket der Rolle zu, sowie die Zahlungsvollmacht, die alle nötigen Angaben über sein Konto enthält, außerdem den Namen des Händlers, den Namen des Kunden und einen Betrag, den ein einzelnes Ticket wert ist. Nachdem sich der Händler überzeugt hat, daß alles seine Richtigkeit hat, beginnt die eigentliche Kaufsitzung.

Für jeden zu kaufenden Artikel schickt der Käufer dem Händler eine Anzahl Tickets in der Reihenfolge der Nummerierung zu. Dieser braucht bei jedem Ticket nur noch darauf zu achten, daß es die Nummer hat, die derjenigen des letzten Tickets folgt.

Da die Tickets fortlaufend nummeriert sind, braucht er auch nur das jeweils letzte aufheben. Da die Tickets nur bei ihm verwendet werden, bedeutet der Besitz eines Tickets, daß er auch alle vorhergehenden erhalten hat.

Nach dem Ablauf einer Sitzung wird der Händler das letzte Ticket an die Bank schicken, die in aller Ruhe dessen Gültigkeit und Wert überprüfen kann, bevor sie den Betrag transferiert.

Um der Tatsache Rechnung zu tragen, daß die Ware beim Transport beschädigt werden könnte, schlägt der Entwurf die Benutzung von zwei Rollen vor. Eine Rolle enthält Tickets, die versprechen, einen bestimmten Betrag zu zahlen, falls der Kunde zufrieden ist, die andere Tickets, die eine Bezahlung bindend zusichern. Bei einem Kauf würde der Händler also zuerst ein vorläufiges Ticket erhalten, nach Prüfung der Ware dann ein bindendes.

Auf diese Weise wäre es möglich, eine Politik des „bezahle nur, wenn Du zufrieden bist“ zu realisieren. Hierbei könnte die Bank als Schiedsrichter in Streitfällen fungieren. Dabei trägt der Händler natürlich prinzipiell das

Risiko für denjenigen Betrag, für den er nur vorläufige Tickets hat.

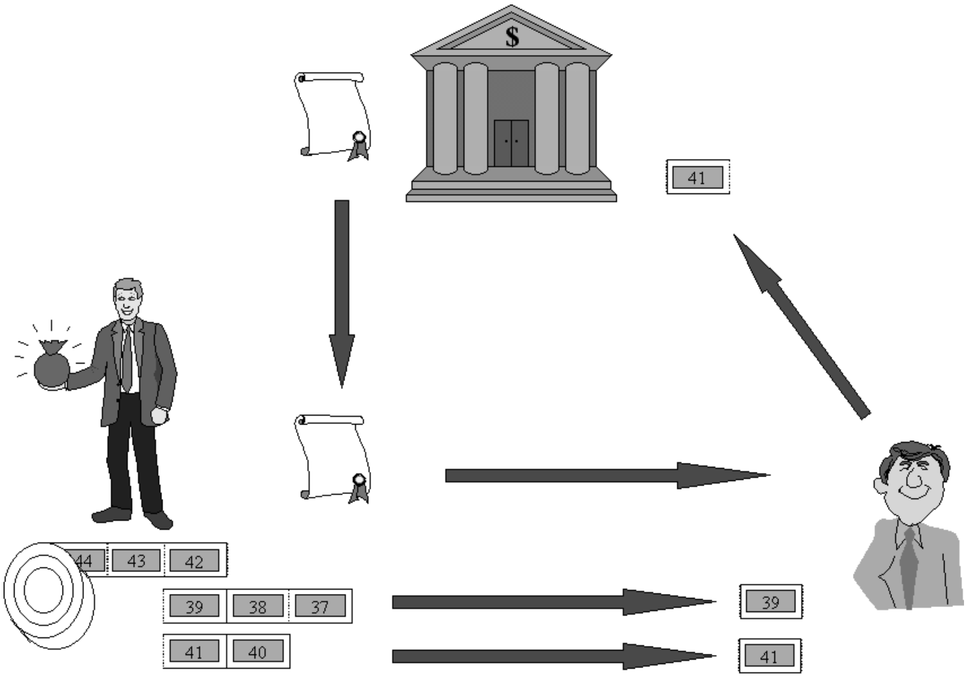

Abbildung 6.1: MPTP

6.1.3 Realisierung

6.1.3.1 Aufbau einer Sitzung

Wie oben beschrieben, gibt es eine Zahlungsvollmacht und eine Ticketrolle.

Die Zahlungsvollmacht enthält diejenigen Daten, die der Händler braucht um die Kreditwürdigkeit des Kunden in Erfahrung zu bringen und die Tickets zu überprüfen.

Diese Vollmacht enthält

1. eine Versionsangabe
2. eine eindeutigen Bezeichner für diese Vollmacht
3. einen Bezeichner für den Käufer
4. einen Bezeichner für den Verkäufer
5. ein Datum (um die Frischheit zu garantieren)
6. eine Liste von Ticketrollenköpfen
7. eine Beschreibung des Kontos
8. einen Signaturalgorithmus
9. eine digitale Signatur über die vorgenannten Punkte unter Benutzung des obigen Algorithmus

Ein Ticketrollenkopf ist gewissermaßen der Anfang einer Ticketrolle. Er gibt die Nummer des ersten Tickets, den Wert eines Tickets und die Tatsache, ob es eine Rolle von vorläufigen oder bindenden Tickets ist, an.

Bei der Beschreibung des Kontos handelt es sich um eine digitale Urkunde der Bank, die folgende Angaben enthält:

1. Kontonummer
2. eine Menge von Flags, die angeben, ob das Konto
   - Zahlungen nur akzeptieren,
   - nur ausgehende Transfers erlauben,
   - zu kleine Einzahlungen ablehnen,
   - oder sich ganz normal verhalten soll

3. Kreditrahmen

4. eine Liste von Servern der Bank

5. eine Liste von Haftungsbeschränkungen (optional):

   - nur von bestimmten IP - Adressen bis zu einem bestimmten Betrag
   - Betrag, bei dessen Überschreitung die Zahlung nicht garantiert wird
   - Betrag, bis zu dem ohne Einzelgenehmigung durch die Bank die Zahlung garantiert wird
   - Betrag, dessen Überschreitung eine Einzelgenehmigung durch die Bank zwingend erfordert

6. Beginn der Gültigkeit

7. Ende der Gültigkeit

Nachdem der Kunde dies geschickt hat, überprüft der Händler das Datum, um sicherzugehen, daß es sich nicht um eine alte Vollmacht handelt. Nun sieht er nach, ob er schon einmal eine Vollmacht mit dem selben Bezeichner erhalten hat. Hierzu muß er eine Liste führen, in der die Bezeichner stehen, die innerhalb eines bestimmten Zeitintervalls bei ihm eingetroffen sind. Ältere Bezeichner kann er löschen, da die zugehörigen Vollmachten schon anhand des Datums zurückgewiesen würden.

Da der Standard zum einen den Begriff der Signatur recht weit faßt und auch MAC's zuläßt, zum anderen aber auch die Garantien für den Zweck nicht ausreichend sein könnten, kann es nötig werden, bei der Bank die Kreditwürdigkeit des Kunden zu erfragen. Bei der Verwendung von Public Key Verfahren und ausreichenden Garantien sollte dies im Regelfall aber nicht nötig sein. Die Anfrage bei der Bank wird mittels MAC geschützt.

Nachdem der Händler nun hinreichendes Vetrauen in den Kunden hat, wird er eine Sitzung eröffnen und die Ankunft der Tickets erwarten.

6.1.3.2 Eine Zahlung

Eine Zahlung besteht aus dem Bezeichner der zugehörigen Vollmacht, einem oder mehrerer Tickets und der Angabe, ob das die letzte Zahlung der Sitzung war.

Eine Ticketrolle besteht aus einer Folge der Art $w_0, w_1, \ldots w_n$, wobei w_n ein frei gewählter Zufallswert ist. Die einzelnen Tickets können relativ einfach durch

$$w_i = h(w_{i+1})$$

berechnet werden, wobei $h()$ eine kryptographisch sichere Einweg-Funktion ist. D.h. es sollte recht einfach sein, $w_i = h(w_{i+1})$ zu berechnen, dagegen darf die Umkehrfunktion $w_{i+1} = h^{-1}(w_i)$ nicht mit vertretbarem Aufwand berechenbar sein.

Dies bedeutet, daß der Käufer in Kenntnis von w_n die ganze Folge berechnen kann, der Händler aber das jeweils nächste w_i erst erfährt, wenn er bezahlt wird. Gleichwohl kann er schnell überprüfen, ob es gültig ist. Er braucht nur $h()$ darauf anwenden und das Ergebnis mit w_{i-1} vergleichen. Dazu hat er als Ausgangspunkt w_0 mit der Vollmacht erhalten.

Als mögliche Implentationen von $h()$ werden Hash - Algorithmen wie der MD5 (Message Digest 5) von Rivest oder der SHA (Secure Hash Algorithm) des amerikanischen Accredited Standards Committee X9 vorgeschlagen. Bei beiden handelt es sich um bekannte Verfahren aus dem Bereich der digitalen Unterschriften.

Dieses Verfahren verketteter Hashwerte wurde zuerst von Leslie Lamport in [Lamport 81] als Verfahren zur Generierung von Einweg - Passworten beschrieben. Dies wurde dann in [RFC 1760] in eine Implementation namens S/KEY umgesetzt. Rivest und Shamir gebührt wohl die Ehre, es zuerst als elektronisches Zahlungsmittel vorgeschlagen zu haben [RiSha 96][1].

6.1.3.3 Einlösung bei der Bank

Diesen Punkt handelt das Papier recht kurz ab. Der Händler schickt der Bank eine Liste von Vollmachten, sowie die dazugehörigen letzten Tickets. Er erhält als Antwort einen Statuswert, eine Liste der Ticketrollen, deren Bezahlung abgelehnt wird, sowie jeweils einen Parameter, der den Grund der Ablehnung angibt.

In der gegenwärtigen Version des Entwurfes wird nur von einer Bank ausgegangen, bei der sowohl Händler als auch Käufer ein Konto haben. Es wird aber eingeräumt, daß in Zukunft wohl das Clearing zwischen den Banken mit berücksichtigt werden muß.

[1] Dieser Artikel wird in [MPTP 95] als „noch nicht erschienen“ geführt. Die engültige Version wurde wohl erst recht spät veröffentlicht.

6.2 NetCheque

6.2.1 Motivation

In jüngster Zeit ist die Anzahl der Benutzer und Unternehmen, die man über das Internet erreichen kann, drastisch angestiegen. Für viele Unternehmen bietet das Internet daher einen effizienten Weg, potentielle Kunden zu erreichen. Bisher werben die Unternehmen jedoch nur für ihre Produkte, z.B. auf den Seiten des World Wide Web. Der eigentliche Kauf der Produkte findet außerhalb des Netzes statt. Grund dafür ist das Fehlen einer geeigneten Bezahlungsmethode, die einerseits sicher, aber auch benutzerfreundlich und einfach integrierbar sein sollte. Das an der USC/ISI[2] entwickelte NetCheque-Verfahren soll hier Abhilfe schaffen, indem es einen verteilten Buchungsservice zur Verfügung stellt.

6.2.2 Grundidee

Das NetCheque-Verfahren ist ein verteilter Buchungsdienst und basiert auf dem Credit-Debit-Modell[3].

Um dieses Verfahren nutzen zu können, muß sich der Kunde bei einem sogenannten Buchungsserver seiner Wahl als NetCheque-Benutzer registrieren lassen und dort ein Konto eröffnen. Ist der Kunde erst einmal bei einem Server registriert, kann ein von diesem Kunden ausgestellter Scheck bei jedem beliebigen Buchungsserver eingelöst werden.

Die Bezahlung mit NetCheque funktioniert im Prinzip wie das Bezahlen mit herkömmlichen Schecks, d.h. im Falle einer Bezahlung stellt der Kunde ein elektronisches Dokument aus, daß seinen Namen, seine Kontonummer, den Namen des Finanzinstitutes, bei dem dieses Konto eingerichtet ist, den Namen des Empfängers und die Betragshöhe enthält. Wie ein herkömmlicher Scheck trägt auch ein NetCheque die (in diesem Fall elektronische) Unterschrift des Kunden. Bevor dieser Scheck ausgezahlt wird, hat auch der Zahlungsempfänger mit seiner digitalen Unterschrift die Zustimmung zu

[2] **I**nformation **S**ciences **I**nstitute der **U**niversity of **S**outhern **C**alifornia

[3] Die Kunden sind mit einem Konto bei Servern, die die Bezahlung abwickeln, registriert. Einen Bezahlung erfolgt dadurch, daß der Kunde einer Abbuchung von seinem Konto zustimmt. Handelt es sich um ein Debit- oder Scheckverfahren, so hat der Kunde ein positives Guthaben, von dem die geforderten Beträge abgebucht werden. Beim Credit-Verfahren dagegen gehen die Kosten zunächst zu Lasten des Credit-Anbieters, und dieser fordert dann in regelmäßigen Abständen die in diesem Zeitraum entstandenen gesammelten Kosten von dem Kunden ein.

geben. Die Unterschriften auf den Schecks werden durch Verwendung des Kerberos-Systems authentisiert.

Verläßlichkeit und Skalierbarkeit werden dadurch erreicht, daß NetCheque nicht nur auf einem, sondern auf einer Vielzahl von Buchungsservern basiert. Da NetCheque ein verteilter Buchungsservice ist, können korrekt unterzeichnete Schecks zwischen den einzelnen Buchungsservern ausgetauscht werden, um so die Kontostände zwischen den einzelnen Servern auszugleichen.

NetCheque ist entwickelt worden, um Bezahlungen zwischen NetCheque-Buchungsservern abzuwickeln, es ist jedoch auch geeignet, Zahlungsvorgänge mit Servern einen anderen Typs zu berbeiten.

Wie bei der Bezahlung mit herkömmlichen Schecks ist auch die Bezahlung mit NetCheque nicht anonym, man kann das Verfahren jedoch dazu verwenden, Beträge zwischen Diensten zu transferieren, die eben diese Anonymität gewährleisten (z.B. NetCash).

NetCheque ist auch für die Bezahlung von sehr geringen Beträgen geeignet, da die NetCheque-Idee ursprünglich dafür entwickelt wurde, bestimmte Quoten für verteilte Systemressourcen zu verwalten, was ständige Transaktionen für kleine Mengen bedeutete. Die Abwicklung vieler kleiner Transaktionen verlangt jedoch nach einer relativ hohen Leistung. Diese wird dadurch erreicht, daß man herkömmliche[4] statt Public-Key Kryptographie verwendet. Hierdurch macht man jedoch kleine Einbußen im Bezug auf die unabhängige Überprüfung der Dokumente an jeder Stelle der Bezahlungskette.

[4]in diesem Fall sind mit herkömmlicher Kryptogrphie die symmetrischen Verfahren gemeint

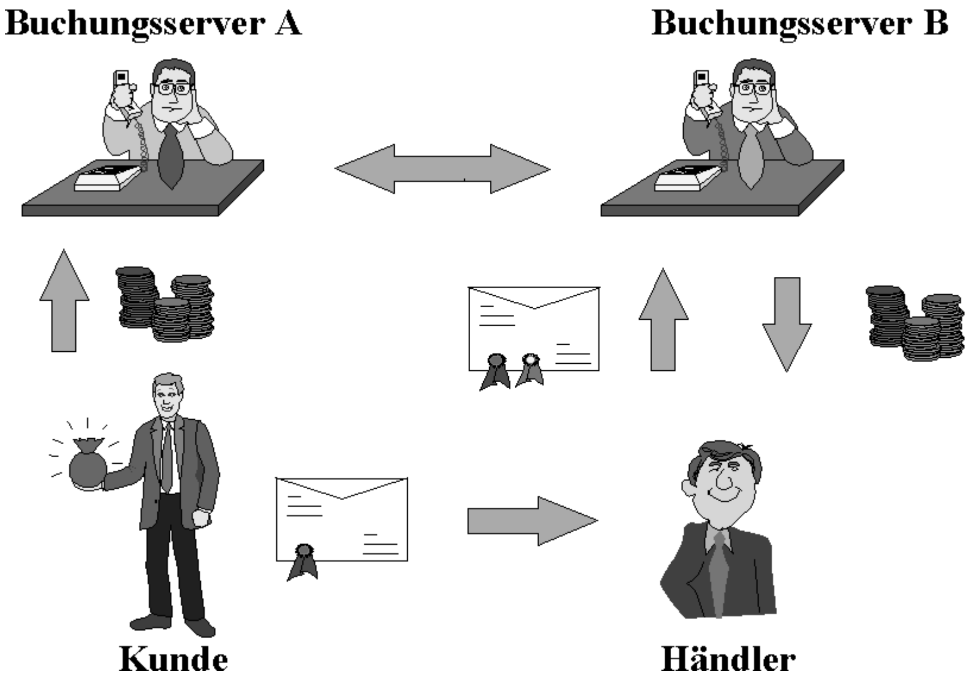

Abbildung 6.2: NetCheque

6.2.3 Realisierung

Die elektronische Unterschrift, die man beim Ausstellen und Bestätigen eines NetCheques verwendet, ist eine spezielle Art eines Kerberos-Tickets, genannt Proxy. Der Scheck selbst enthält folgende Informationen über:

- Die Höhe des Betrages
- Die Währungseinheit
- Das belastete Konto
- Den bzw. die Zahlungsempfänger

All diese Informationen sind von dem Aussteller des Schecks frei lesbar. Zusätzlich enthält der Scheck noch alle während des Prozesses entstandenen Unterschriften und Bestätigungen, die von dem Buchungsserver, der durch diesen Scheck belastet wird, überprüfbar sind. Aus Leistungsgründen basiert das verwendete Kerberos-Proxy auf herkömmlicher Kryptographie, kann jedoch durch eine Signatur ersetzt werden, die einen Public-Key verwendet (mit dem Nachteil, daß man an Leistung verliert).

6.2.3.1 Ausstellen eines NetCheques

Um einen Scheck auszustellen, ruft der Kunde eine Funktion namens `write_cheque` auf. Als Parameter werden der Funktion übergeben:

- Das zu belastende Konto
- Der Zahlungsempfänger
- Die Höhe des Betrages
- Die Währungseinheit

Die Standardvorgabewerte für das Konto und die Währungseinheit werden aus der vom Kunden angelegten `.chequebook`-Datei entnommen. Die `write_cheque`-Funktion generiert den Klartextteil des Schecks und erwirbt ein Kerberos-Ticket, daß benutzt wird, um den Kunden gegenüber dem Buchungsserver zu authentisieren. Danach wird ein Authentikator mit einer eingebetteten Prüfsumme über die Informationen des Schecks generiert, und dieser Authentikator mitsamt dem Kerberos-Ticket im Signaturfeld des Schecks vermerkt. Der Scheck wird jetzt mit dem Base-64-Verfahren kodiert und kann dann per e-mail oder über eine bestehende Online-Verbindung an den Empfänger geschickt werden.

6.2.3.2 Einlösen eines NetCheques

Beim Einlösen eines Schecks wird die `deposit_cheque`-Funktion aufgerufen. Diese liest den Klartextteil des Schecks und erwirbt ein Kerberos-Ticket, daß zur Kommunikation mit dem Buchungsserver des Kunden benutzt wird. Danach generiert die Funktion einen Authentikator, der im Namen des Zahlungsempfängers bestätigt, daß dieser Scheck nur auf das Konto des Zahlungsempfängers gutgeschrieben werden kann. Diese Bestätigung wird an den Scheck angehängt. Daraufhin wird eine verschlüsselte Verbindung zum Buchungsserver des Kunden aufgebaut, um dort den Scheck einzulösen.

Haben Kunde und Zahlungsempfänger den selben Buchungsserver, so kann der entsprechende Betrag sofort vom Konto des Kunden abgezogen und auf das Konto des Zahlungsempfängers gutgeschrieben werden.

Sind die Buchungserver des Kunden und des Zahlungsempfängers nicht identisch, teilt der Buchungsserver des Zahlungsempfängers diesem zunächst nur mit, daß der Scheck akzeptiert worden ist und er den entsprechenden Betrag erhalten wird, sobald die beiden beteiligten Buchungsserver sich gegenseitig abgeglichen haben. Der Empfänger hat auch die Möglichkeit, auf einen sofortigen Ausgleich (d.h. in Echtzeit) mit dem Konto des Kunden zu bestehen, hierfür wird der Buchungsserver jedoch wahrscheinlich eine zusätzliche Gebühr verlangen.

Wird ein Scheck zurückgewiesen, so geht er zurück an den Zahlungsempfänger, der dann entsprechende Maßnahmen einleiten kann.

Um seinen aktuellen Kontostand und Informationen über eingelöste Schecks zu bekommen, können authorisierte Benutzer die sogenannte `statement`-Funktion aufrufen, die eine verschlüsselte Verbindung zum Buchungsserver aufbaut und daraufhin den aktuellen Kontostand für jede Währungseinheit und eine Liste aller Schecks, die in letzter Zeit auf dieses Konto eingelöst oder ausgestellt worden sind, erhält. Hierbei wird der gesamte Scheck als Antwort geschickt, so daß das Anwendungsprogramm des Benutzers auch wirklich alle Daten anzeigen kann, die der Benutzer selbst oder eine andere Anwendung benötigt.

6.3 SET

6.3.1 Motivation

SET wurde von den Kreditkartenfirmen VISA und Mastercard entwickelt. Nachdem sie zuerst getrennte Ansätze verfolgten, einigten sie sich am 1.

Februar 1996 ein gemeinsames Verfahren zum Bezahlen mit Kreditkarte im Internet zu entwickeln[SET 96].

Angesichts der geringen Sicherheit des Internets per se und der bekannten Risiken bei der Übertragung konventioneller Konzepte in eine digitale Umgebung soll das Verfahren sicherstellen, daß

- sich Kunde und Händler gegenseitig authentisieren können,
- eine Zahlung nicht durch eine der Parteien oder einen Außenstehenden manipuliert werden kann,
- der Inhalt einer Transaktion vertraulich bleibt[5].

Inzwischen ist SET in der Testphase und liegt als dreibändige Beschreibung mit insgesammt 622 Seiten vor.

6.3.2 Grundidee

Um sich gegenseitig authentisieren zu können, haben Händler und Kunde von einer vertrauenswürdigen Instanz eine Art Ausweis bekommen. Bevor sie nun miteinander Geschäfte machen, überprüfen sie gegenseitig ihre Ausweise um sicherzustellen, daß sie auch wirklich mit demjenigen sprechen, den sie glauben vor sich zu haben.

Nachdem die geschehen ist, schreibt der Kunde auf einen Zettel was er kaufen will, wieviel er dafür zu zahlen bereit ist und von welcher Kreditkarte dies abgebucht werden soll. Diesen Zettel verschließt er in einem Umschlag. Nun gibt er den Umschlag dem Händler zusammen mit der Angabe, was er zu kaufen wünscht.

Der Händler schreibt nun ebenfalls die Artikel und den Betrag auf einen zweiten Zettel und gibt diesen zusammen mit dem Umschlag auf der Bank ab. Die Bank öffnet den Umschlag, vergleicht die Angaben und leitet die Zahlung ein, falls beide die selben Artikel und den selben Preis angegeben haben und der Kunde liquide ist.

[5] Dabei wird insbesondere daran gedacht, daß es das Ausspähen von Kreditkarteniformationen erlaubt, Betrug zu begehen und auf anderer Leute Kosten einzukaufen.

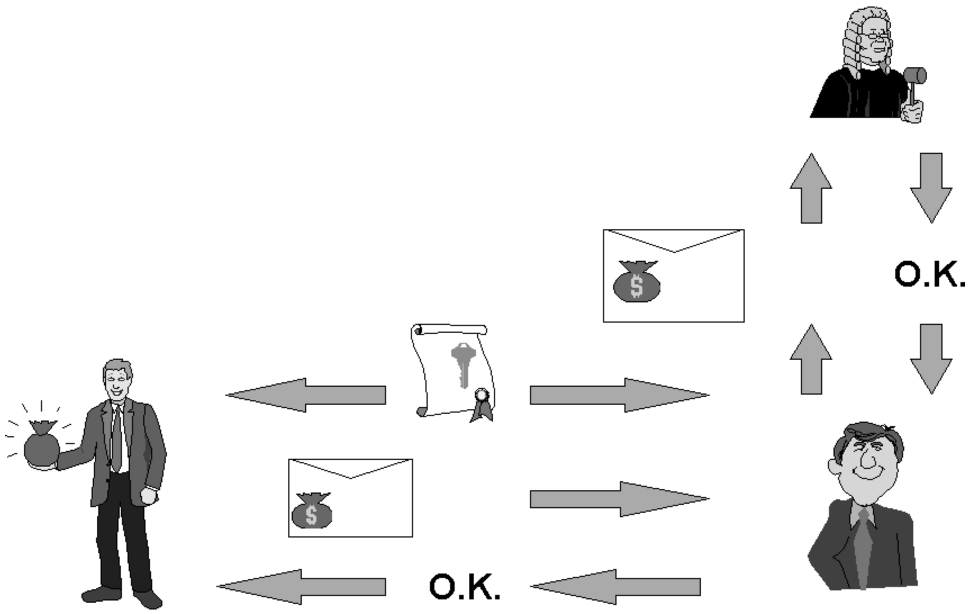

Abbildung 6.3: SET

6.3.3 Realisierung

Das folgende entstammt in erster Linie der „Business Description“, dem ersten Band der SET Beschreibung gedacht für Anwender. Beim Durchlesen des zweiten und dritten Bandes wird man schnell feststellen, daß das Protokoll viel komplizierter ist als hier dargestellt, insbesondere was die Anwendung von Hashfunktionen angeht. Die 120 Seiten Protokolldefinition sowie das 431 seitige Handbuch für Programmierer hier aufzunehmen hätte den Rahmen aber leider gesprengt.

6.3.3.1 Die Beteiligten einer Transaktion

Ohne zu weit vorgreifen zu wollen erscheint es doch nötig, die Beteiligten einer Transaktion aufzuführen, damit es später zu keinen Mißverständnissen kommt.

Am Anfang stehen natürlich **Kunde** und **Händler**. Um Geld auszutauschen, benötigen sie ein **Payment Gateway**. Dieses ist Teil der **Bank des Händlers** und dient dazu dem Händler mitzuteilen, ob eine Zahlung gedeckt ist. Dazu muß es Kontakt zur **Bank des Kunden** aufnehmen.

6.3.3.2 Doppelte Unterschriften

Bei einem Kauf per Kreditkarte entstehen zwei Sorten von Information. Da ist auf der einen Seite die Kontoinformation (z.B. die Kreditkartennummer, Verfallsdatum, etc), auf der anderen die Information über das Produkt und und die Bedingungen der Transaktion (Zahlungs- und Liefermodalitäten).

Der Kunde hat ein Interesse, daß der Händler die Kontoinformation nicht erhält, während die Bank die Produktinformation oder die Modalitäten nicht benötigt. Andererseits möchte er, daß die Bank das Geld nur dann auszahlt, wenn der Händler mit allen Bedingungen einverstanden ist.

Um nun zu erreichen, daß jeder nur die Informationen erhält, die er benötigt, führt SET das Konzept der doppelten Unterschriften ein. Bei einer normalen Unterschrift[6] wird ein Hash über die zu signierende Nachricht gebildet und dieser dann mit dem geheimen Schlüssel eines Public - Key - Verfahrens verschlüsselt.

Hier werden nun erst einmal die Hashwerte der Einzelnachrichten gebildet. Diese werden aneinandergehängt. Von dem Ergebnis wird wieder ein Hash gebildet, der dann verschlüsselt wird.

[6] „Unterschrift“ bedeutet in diesem Kontext immer digitale Signatur.

Nun kann der Kunde seine Kontoinformation, den Hash der Modalitäten, sowie das Endergebnis obiger Berechnung der Bank übermitteln, während er dem Händler nur die Modalitäten schickt. Nachdem nun der Händler seinerseits den Hash der Modalitäten berechnet und an die Bank gesendet hat, kann diese überprüfen, ob tatsächlich beide die selben Modalitäten wünschen, ohne die Modalitäten tatsächlich zu kennen.

6.3.3.3 Export Probleme

Einige Staaten beschränken den Export kryptographischer Verfahren. Normalerweise wird er aber genehmigt, falls

- der Inhalt finanzieller Natur ist,
- er wohldefiniert ist,
- seine Länge bregenzt ist, und
- das Verfahren nicht einfach zu anderen Zwecken zu benutzen ist.

Da SET diese Anforderungen erfüllt, sollte jede Implementation, die
ihre kryptographischen Verfahren ausschließlich zu diesem Zweck einsetzt, und die nicht leicht für andere Zwecke benutzt werden kann, auch eine Exportlizenz bekommen.

6.3.3.4 Zertifikate

Zertifizierungshierarchie

Alle Beteiligten an einer Transaktion müssen einander vertrauen. Um diesem Vertrauen eine Basis zu schaffen, wurde eine Hierarchie von Zertifizierungsinstanzen geschaffen, die der jeweils nächstniedriegeren Ebene ihre Authentizität bescheinigen.

Dazu unterschreibt jede Zertifizierungsinstanz den öffentlichen Schlüssel der nachfolgenden Instanz. Die Zertifikate hängen nun wie eine Kette zusammen. Um sie zu überprüfen, ist es nötig bei jedem Zertifikat festzustellen ob es wirklich mit dem Schlüssel aus dem nächsthöheren Zertifikat unterschrieben wurde. Selbstverständlich wird man dabei irgendwann zum Zertifikat der obersten Instanz gelangen. Dieses ist mit dem eigenen Schlüssel signiert.

Da nun alles von der Authentizität dieses einen Zertifikats abhängt, ist es nötig, daß der Benutzer des Systems den öffentlichen Schlüssel der obersten

Instanz auf einem sicheren Wege erhält. Dies geschieht, indem das oberste Zertifikat zusammen mit dem Hashwert eines Ersatzschlüssels der SET Software beiliegt.

Die Software kann überprüfen, ob sie den aktuellen Schlüssel hat, indem sie den Hashwert des obersten Zertifikats an eine Zertifizerungsinstanz sendet. Ist dieses nicht mehr gültig, sendet die Instanz das aktuelle Zertifikat zurück. Um sicherzustellen, daß dieses auch authentisch ist, muß der Benutzer der Software einen String mitteilen, der dem Hashwert entspricht. Diesen muß er sich vorher von einer verläßlichen Quelle (z.B. seiner Bank) besorgen.

Der normale Weg ein Zertifikat zu ersetzen ist allerdings, eine Nachricht mit einem neuen Zertifikat zu senden, dessen Schlüssel der oben erwähnte Ersatzschlüssel ist, dessen Hashwert die Software bereits besitzt und den sie damit überprüfen kann. Mit dem neuen Zertifikat kommt selbstverständlich ein neuer Hash eines Ersatzschlüssels, so daß das Auswechseln der Schlüssel in regelmäßigen Abständen vom Benutzer unbemerkt ablaufen kann.

Zertifizierung des Kartenbesitzers

Es beginnt damit, daß Benutzer eine Anfrage an eine Zertifizierungsinstanz sendet.

Die Antwort der Instanz enthält ihre Kette von Zertifikaten. Damit ist es dem Kunden möglich, die Authentizität der Instanz zu überprüfen und alle zukünftigen Anfragen mit dem öffentlichen Schlüssel der Instanz zu verschlüsseln.

Um das eigentliche Formular zur Erteilung eine Zertifikats zu erhalten, muß der Kunde der Instanz seine Kontonummer mitteilen. Diese Nachricht wird selbstverständlich verschlüsselt übertragen[7].

Die Instanz wählt das geeignete Formular und schickt es an den Kunden. Ist sie nicht zuständig, lehnt sie ab und sendet die Information, wo das Formular erhalten werden kann.

Der Kunde generiert einen öffentlichen und einen privaten Schlüssel und füllt das Formular aus. Formular, öffentlicher Schlüssel, sowie ein neu generierter symmetrischer Schlüssel k_1 und eine Zufallszahl r_1 werden dann mit dem geheimen Schlüssel signiert und an die Instanz gesendet.

[7] Im folgenden werden wir nur noch darauf hinweisen, wenn eine Nachricht im Klartext übertragen werden sollte. Nachrichten der Zertifizierungsinstanz sind grundsätzlich signiert und enthalten das Zertifikat der Instanz. Auch werden in jedem Schritt gesendete Zertifikate und Signaturen vom Empfänger geprüft.

Die Instanz überprüft das Formular und generiert eine Zufallszahl r_2 diese wird mit r_1 kombiniert um einen geheimen Wert *secret* zu erhalten. Dieser wird dann mit der Kontonummer und dem Verfallsdatum kombiniert und einem Hash - Algorithmus unterworfen. Das Ergebnis wird im Zertfikat eingetragen. Die Kenntnis von Kontonummer, Verfallsdatum und *secret* erlauben es später, die eigene Verbindung zu einem Zertifikat zu beweisen.

Die Instanz verschlüsselt *secret* nun mit dem symmetrischen Schlüssel k_1 und sendet das Ergebnis zusammen mit dem neuen Zertifikat an den Kunden.

Zertifizierung des Händlers

Die Zertifizierung des Händlers läuft ähnlich ab, allerdings benötigt er zusätzlich ein Schlüsselpaar, um symmetrische Schlüssel auszutauschen. Dafür braucht er keinen geheimen Wert *secret*.

6.3.3.5 Der Kauf

Bestellung

Um eine Transaktion zu beginnen, sendet der Käufer an den Händler eine Bitte um dessen Zertifikate und die des für ihn zuständigen Payment Gateways.

Nach dem obligatorischen Überprüfen der Zertifikatketten werden die Zertifikate der Zertifizierungsinstanzen für das Überprüfen der in den folgenden Schritte immer wieder durch das Payment Gateway und den Händler gesendeten Zertifikate gespeichert.

Die Software generiert nun Bestellinformation und Zahlungsinformation. Dabei enthält die Bestellinformation nicht die Beschreibung der zu erwerbenden Güter oder der Zahlungsmodalitäten[8]. All dies wurde in der Einkaufsphase vor der ersten SET - Nachricht festgelegt. Vielmehr werden hier ein Transaktionsbezeichner[9] und ein Hash über die Beschreibung des Auftrages und den Betrag festgehalten.

Über Bestell- und Zahlungsinformation wird die doppelte Signatur gebildet und die signierte Zahlungsinformation mit dem öffentlichen Schlüssel des Payment Gateways verschlüsselt, worauf die verschlüsselte Zahlungsin-

[8] Hier detaillierte Informationen einzufügen, hätte die Gefahr des Exportverbots mit sich gebracht.

[9] enthält Kunde, Händler, Datum

formation, die signierte Bestellinformation und das Zertifikat des Kunden[10] an den Händler gesendet werden.

Der Händler kann nun oder später, wenn er die Gültigkeit der Zahlung überprüft hat, dem Kunden die Bestellung bestätigen.

Die Waren wird der Händler liefern, wenn die Zahlung überprüft wurde.

Nach dem Erhalt der Bestätigung kann der Kunde jederzeit den Status seiner Bestellung (wurde die Zahlung für gültig erklärt, ist schon gezahlt worden?) mittels einer mit einer „order enquiry message“ feststellen.

Überprüfung der Zahlung

Im folgenden werden die Überprüfung der Zahlung und die tatsächliche Zahlung als zwei getrennte Schritte dargestellt. Tatsächlich erlaubt das Protokoll es auch, dies in einem Schritt zu tun.

Der Händler generiert eine Bitte um Authorisierung, diese enthält die zu autoriserende Summe, die Transaktionsnummer aus der Bestellinformation sowie einige zusätzliche Daten über die Transaktion [11].

Diese Nachricht wird mit dem öffentlichen Schlüssel des Payment Gateways verschlüsselt und zusammen mit der verschlüsselten Zahlungsinformation an das Payment Gateway geschickt.

Nachdem dieses die Signaturen und Zertifikate geprüft hat und sichergestellt wurde, daß auch der Händler denselben Hash über Auftragsbeschreibung und zu zahlende Summe gesendet hat, bittet es die Bank des Kunden um Authorisierung und leitet die Antwort an den Händler weiter, wobei diese verschlüsselt übertragen wird. Optional kann ein ebenfalls verschlüsseltes $Capture\_Token$ mitgeschickt werden. Dies geschieht aber nur, wenn dies von der Bank des Händlers verlangt wird.

Vollzug der Zahlung

Dieser Schritt erlaubt es, die eigentlichen Zahlungen mehrerer Transaktionen gebündelt abzuwickeln. Wir werden im folgenden aber nur eine Transaktion betrachten. Sollen mehrere Transaktionen gleichzeitig behandelt werden, so werden die Daten hintereinander in einer Nachricht geschickt.

[10] Im folgenden wird auch weiterhin mit jeder Nachricht das Zertifikat des Sendenden mitgeschickt. Auch werden alle Nachrichten durch den Sender signiert. Dies werden wir daher nicht mehr explizit erwähnen. Wie auch bei der Zertifizierung gilt im folgenden implizit, daß Zertifikate und Signaturen grundsätzlich geprüft werden.

[11] u.a. einen eigenen Hash über Auftragsbeschreibung und Betrag der Transaktion und ob er mißtrauisch bezüglich des Kunden ist

Um das ihm versprochene Geld auch zu erhalten, sendet der Händler eine Zahlungsaufforderung an das Payment Gateway. Diese Aufforderung enthält die verlangte Summe, Transaktionsbezeichner, usw. und - falls vorhanden - das *Capture_Token*.

Das Payment Gateway sendet eine Aufforderung an die Bank des Kunden zu zahlen und schickt dem Händler eine Bestätigung[12]

6.4 Cybercash

6.4.1 Motivation

CyberCash Inc. [Cybercash 97] [Cybercash 95] wurde 1994 in Reston im US-Bundesstaat Virginia gegründet. Ziel war es, eine bequeme, preiswerte und vertrauenswürdige Verbindung zwischen der neuartigen Welt des Cyberspace und und der konventionellen Bankwelt zu schaffen.

Das System sollte das Bezahlen mit Kreditkarte sicherer machen. Dabei sollte es möglich sein alle gängigen Transportprotokolle zu verwenden. (Insbesondere HTTP[13] und SMTP[14]). Um den Kunden zu schützen und Mißbrauch vorzubeugen, sollten Kreditkartennummern weder dem Händler bekannt, noch dauerhaft auf dem Server von CyberCash gelagert werden, der ja damit zu einem Angriffsziel würde.

Realisiert wurde dies durch eine Verschlüsselungssoftware, die Kunden als CyberCash Wallet (Checkfree Wallet, Compuserve Wallet[15]) kostenlos zur Verfügung steht. Für Händler gibt es gegen eine kleine Gebühr das Cash Register, ein Programm, das seit Oktober 1996 zusammen mit Netscape weiterentwickelt wird.

Nach der Einführung des Systems 1995 wird es mittlerweile durch mehr als 300 Händler genutzt.

Im September 1996 wurde auch ein Verfahren zur Bezahlung kleiner Beträge, „CyberCoin“ genannt, in das System integriert. Obwohl dieses eigentlich in die Kategorie „elektronisches Bargeld“ fällt, werden wir es trotzdem hier mit erklären, da es zu eng mit dem Kreditkartenprotokoll verwoben ist, um ein eigenes Kapitel zu rechtfertigen.

[12] Dies hat nichts damit zu tun, daß der Betrag auch tatsächlich gutgeschrieben wurde. Die Antwort erfolgt sofort, ohne eine Reaktion der Bank abzuwarten.

[13] Das Protokoll des World Wide Web.

[14] d.h. Email

[15] Prinzipiell kann jede finanzielle Institution kostenlos eine Wallet mit ihrem Namen vertreiben.

Um den direkten Transfer zwischen Girokonten und damit direkte Zahlungen zwischen Privatkunden zu erlauben, ist es geplant, zusätzlich ein System namens „PayNow Secure Electronic Check Service“ einzuführen.

Das hier vorgestellte Protokoll zur sicheren Übertragung von Kreditkartendaten wird mit der Einführung von SET durch dieses ersetzt werden. CyberCash plant eines der ersten Unternehmen zu sein, das ein SET kompatibles System anbietet. Da die Unterschiede zwischen SET und dem CyberCash Kreditkartenprotokoll aber nur in den Details liegen, und da SET auch noch nicht endgültig freigegeben ist, werden wir uns in diesem Kapitel auf die Cybercash Protokolldefinition in [RFC 1898] beziehen.

Am 6. März 1997 verkündeten CyberCash, die Dresdner Bank und die Sachsen LB das CyberCash-System auch in Deutschland einführen zu wollen. Damit wäre es erstmalig möglich, in einer anderen Währung als US-Dollar zu bezahlen. Als erster Schritt soll das Kreditkartensystem eingeführt werden. Im zweiten Schritt soll auch CyberCoin realisiert werden. Dabei soll es möglich sein, Währungskonvertierung „on-the-fly“ zu realisieren, so daß ein Kunde problemlos in D-Mark bezahlen kann, während der Händler sein Geld in US-Dollar erhält.

6.4.2 Grundidee

6.4.2.1 Kreditkartenzahlungen

Die Grundidee für das Zahlen mit Kreditkarte ist dasselbe wie bei SET. Auch hier sind die drei Hauptakteure der Kunde, der Händler und der CyberCash Server.

Nachdem sich Händler und Kunde auf die zu kaufenden Waren und die zu zahlende Summe geeinigt haben, schreibt der Kunde die nötigen Informationen (Inhalt der Transaktion, Kreditkartennummer) auf einen Zettel, den er in einem Umschlag verschließt und an den Händler weiterleitet.

Der Händler wird nun seinerseits ebenfalls die Transaktionsdaten aufschreiben und zusammen mit dem Brief des Kunden an den Server schicken.

Der Server wird nun die Angaben vergleichen und die Zahlung in die Wege leiten, indem er Kontakt zu den Banken der Beteiligten aufnimmt.

Auch Bestellungen per Telephon oder das Benutzen von unverschlüsselten Webformularen zur Übermittlung von Kreditkartendaten ist möglich. Dazu sendet der Händler die Kreditkartendaten selber an den Server, der daraufhin wie gehabt die Zahlung in die Wege leiten wird.

Um im Streitfalle direkt mit einer Kreditkartengesellschaft in Kontakt treten zu können, ist es aber möglich den Server nach der Kreditkartennummer des Kunden zu fragen. Der Server wird diese mitteilen, darüber aber einen Aktenvermerk machen, um einen Mißbrauch dieser Möglichkeit erkennen zu können.

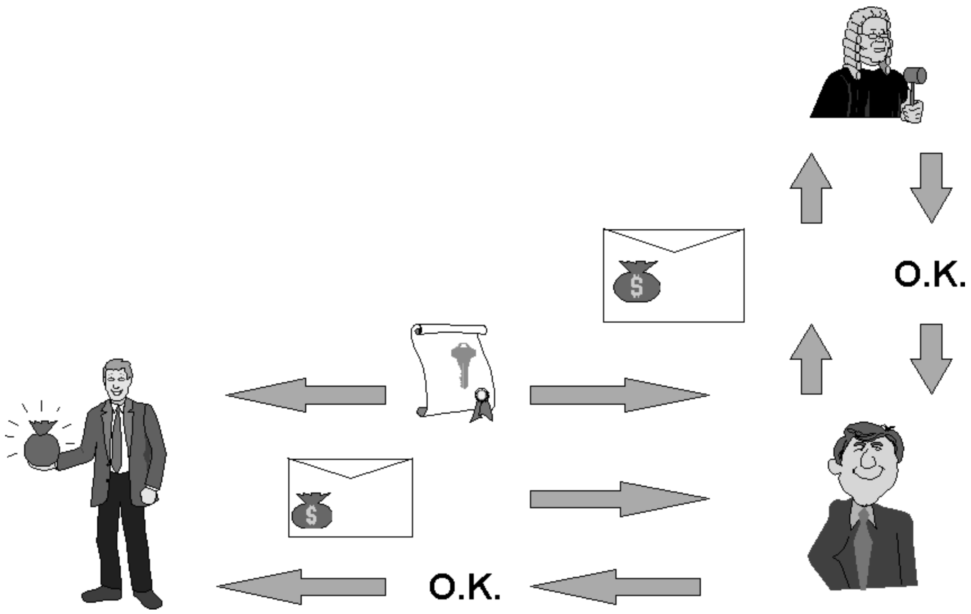

Abbildung 6.4: Cybercash

6.4.2.2 CyberCoin

Um Zahlungen leisten zu können, müssen sich Händler und Kunde registrieren lassen. Dabei wird bei einer CyberCash nahestehenden Bank ein Konto eröffnet[16]. Der Kunde bekommt dabei eine Art Sparbuch.

Bevor er aber nun bezahlen kann, muß er Geld auf sein Sparbuch einzahlen. Dieses wird üblicherweise von einer Kreditkarte oder einem Girokonto abgebucht.

Nachdem dies geschehen ist, steht einer Zahlung nichts mehr im Wege. Hierzu wird das Geld zwischen den Verrechnungskonten transferiert und im Sparbuch die Transaktion und der neue Kontostand vermerkt.

[16]Dies ist kein normales Konto, sondern dient nur der Verrechnung.

Für die Zukunft ist geplant unterschiedliche Währungen zu unterstützen, die während einer Transaktion konvertiert werden können.

6.4.2.3 Transaktionen zwischen Privatpersonen

Dieses System von CyberCash „Electronic Cheque“ genannt wird erst 1997 eingeführt werden. Informationen über die Realisierung liegen uns daher nicht vor. Die Darstellung an dieser Stelle bezieht sich daher nur auf [Cybercash 95]. Diese Quelle stammt leider noch aus der Zeit vor der Einführung von CyberCoin. Die darin enthaltenen Angaben könnten daher z.T. überholt sein.

Um diese Möglichkeit in Anspruch zu nehmen, eröffnen die Parteien jeweils ein Konto bei CyberCash[17]. Auf dieses kann nun Geld von einem normalen Girokonto geladen werden. Das bedeutet, CyberCash bucht das Geld vom Girokonto ab und schreibt es auf einem internen Konto gut.

Wenn nun ein Kunde eine Zahlung zu machen wünscht, sendet er einen Überweisungsauftrag an CyberCash, wo das Geld – falls vorhanden – dem Konto des Empfängers gutgeschrieben wird. Um die Übertragung zu schützen wird die Anweisung digital vom Kunden unterschrieben.

Nachdem der Server die Überweisung ausgeführt hat, schickt er dem Kunden eine Quittung, die ebenfalls digital unterschrieben ist, und die dazu dienen kann, dem Gläubiger die erfolgte Zahlung zu beweisen.

Zusätzlich ist es natürlich auch möglich seinen Kontostand abzurufen und Geld vom Konto bei CyberCash zurück auf das eigene Girokonto zu bewegen.

Prinzipiell soll es auch möglich sein, Zahlungen an Personen zu leisten, die kein Konto bei CyberCash besitzen. Details liegen uns dazu aber nicht vor.

6.4.3 Realisierung

6.4.3.1 Allgemeines

Technische Details

Um es CyberCash Nachrichten zu erlauben, auf beliebigen Protokollen aufzusetzen, sind nur reine ASCII Zeichen in ihnen erlaubt, d.h. der Zah-

[17] Vermutlich wird es sich hier wie bei CyberCoin um ein Konto bei einer CyberCash nahestehenden Bank handeln.

lenwert eines Zeichens darf weder Null noch größer als 127 sein. Daten, die ausserhalb dieses Bereiches liegen, müssen Base64[18] kodiert werden.

Die eigentliche Nachricht hat zwei Teile, einen öffentlichen, der als Klartext übertragen wird und einen privaten, der mit DES verschlüsselt und Base64 kodiert ist.

Antwortnachrichten enthalten in der Regel im privaten Teil ein Textfeld, in dem Meldungen übermittelt werden können, die dann von der Software des Kunden am Bildschirm angezeigt werden.

Zur Verschlüsselung wird ein auf DES und RSA basierendes Public Key Verfahren verwendet. Die im privaten Teil zum Teil vorhandene Signatur des öffentlichen Teils benutzt MD5 und RSA.

CyberCash hat die Genehmigung der US Regierung Schlüssellängen bis 1024 Bit für RSA zu verwenden. Im Moment werden allerdings nur 768 Bit benutzt, was sich allerdings 1997 ändern soll[19].

Registrierung bei CyberCash

CyberCash identifiziert (und authentifiziert) Kunden über eine Persona. Diese besteht in erster Linie aus

`Persona ID` Ein Identifikationsstring auf den sich Kunde und Server geeinigt haben

`öffentlicher Schlüssel` für RSA

`Email Adresse`

`Kreditkarten Fingerabdrücke` Diese bestehen jeweils aus dem Anfang der Kreditkartennummer und einem Hashwert aus Kreditkartennummer und „Salz“[20]

`sonstige Kreditkartendaten` z.B. Name, Ablaufdatum, Adresse

---

[18] Eine gängige Methode im Internet, Zeichen außerhalb des erwähnten Bereiches durch normale Zeichen (Buchstaben, Interpunktionszeichen) auszudrücken.

[19] Vielleicht sollte man dabei im Auge behalten, daß 1024 Bit RSA Schlüssel zwar ganz ordentlich sind, die 56 Bit DES-Schlüssel aber z.T. als nicht mehr wirklich sicher angesehen werden. (s. dazu den Anhang)

[20] Ein Zufallswert um die Umkehrung des Verfahrens zu erschweren. Da Kreditkartennummern eine zu regelmäßige Struktur haben, wäre ein Angriff sonst zu einfach.

Wer eine bessere Übersetzung des Wortes “salt” weiß, möge sich bitte melden.

Prinzipiell ist es zulässig, daß ein Kunde mehrere Personas besitzt oder das eine Persona von mehreren Kunden benutzt wird. Es kann aber sein, daß im letzteren Fall die Kreditkartenfirma zusätzliche Daten zur Authentifikation verlangt.

Benutzen kann so eine Persona normalerweise nur, wer sowohl die Persona ID als auch den dazu gehörigen geheimen Schlüssel kennt. Um in Notfällen oder bei Verlust dieser Daten aber nicht völlig verloren zu sein, gibt es noch eine Notfallpassphrase[21]. Wird diese von CyberCash empfangen, wird die Persona sofort gesperrt. Geld, das auf ein Konto bei CyberCash geladen wurde, wird zurück auf das Girokonto des Besitzers überwiesen.

Diese Passphrase braucht nicht mittels eines speziellen Protokolls gesendet werden, auch Email und Telefon sind zulässig.

Um eine Persona zu registrieren, lädt der Käufer[22] die CyberCash-Software über das Internet, startet sie und füllt einige Angaben zu seiner Person aus. Nach dem dies geschehen ist, sendet die Software eine Nachricht, die

- ihre eigene Versionsnummer
- eine erbetene Persona ID
- die Emailadresse des Kunden

enthält.

Ist alles in Ordnung, dann enthält die Antwort die Persona ID, die mit zusätzlichen Prüfzeichen verlängert wurde.

Gab es die Persona ID schon, wird ein Vorschlag für eine andere ID zurückgesendet.

Ist die Software veraltet, so kann die Software automatisch ein Update von Cybercash anforden. Dafür sind spezielle Protokollnachrichten vorgesehen. Dies geschieht aber wohl nicht ohne das Wissen des Kunden. Im Protokoll ist vorgesehen, das eine Mitteilung mitgesendet wird, die von der CyberCash Software am Bildschirm angezeigt wird.

[21]Eine Passphrase unterscheidet sich von einem Password dadurch, daß sie deutlich länger sein kann. Während ein Password oft maximal 8 Zeichen lang ist, kann es sich bei einer Passphrase schon einmal um einen ganzen Satz handeln, was die Sicherheit deutlich erhöht. Wie dies allerdings bei CyberCash gehandhabt wird, ist uns nicht bekannt.

[22]Die eigentlich Registrierung eines Händlers wird ähnlich ablaufen, ist aber nicht dokumentiert. Da es für ihn aber auch mehr Arbeit bedeutet, da er ja erst eine Bank finden, eine Webside aufsetzen und die CyberCash-Software in sein System integrieren muß, würde eine detaillierte Beschreibung den Rahmen dieser Arbeit sprengen.

Nachdem nun eine Persona registriert wurde, wird als nächstes eine oder mehrere Kreditkarten an die Persona binden. Dazu werden neben der Persona ID, die immer im Klartext übertragen wird, im privaten Teil der Nachricht

- Kartennummer,
- Kartenaussteller,
- Salz (s.o.),
- Verfallsdatum,
- Name und Adresse des Kartenbesitzers

übertragen. Wie schon erwähnt wird der Server die Kartennummer aber nicht speichern, sondern aus ihr nur einen Fingerabdruck generieren.

In der Antwort werden im privaten Teil neben Erfolg oder Fehlschlag auch die Kreditkartendaten zurückgesandt. Diese können im Erfolgsfalle nun von der Software gespeichert werden, um später im aktuellen Einkauf wieder benutzt zu werden.

Kosten

Bemerkenswert ist vielleicht noch, daß CyberCash für Kunden und Händler kostenlos ist. Die Kosten werden vielmehr der Kreditkartenfirma berechnet.

Auch die nötige Software kann vom Kunden kostenlos von der CyberCash Webside geladen werden. Die Software, die ein Händler benötigt kostet allerdings „eine kleine Gebühr“[Cybercash 97].

6.4.3.2 Das Kreditkartenprotokoll

Kommunikation zwischen Käufer und Händler

Das Protokoll beginnt, nachdem der Kunde dem Händler mitgeteilt hat, was er zu kaufen wünscht.

Die erste Nachricht ist die Zahlungsaufforderung durch den Händler. Diese kann als MIME-Typ `application/cybercash` in Web-Seiten eingebettet werden. Dies hat den Vorteil, daß die gängigen Browser so konfiguriert werden können, daß die Nachricht automatisch an die CyberCash-Software weitergeleitet wird.

Die Nachricht enthält eine Freitext-Beschreibung der Transaktion z.B.

ACME Products

Purchase of 4 pairs "Rocket Shoes" at $39.95 ea.
Shipping and handling $5.00

Ship to:
Wily Coyote
1234 South St.
Somewhere, VA 12345

[RFC 1898]

Außerdem enthält sie ein Feld für den Betrag und eine Liste akzeptierten Kreditkarten sowie Schlüsselbezeichner die für die jeweilige Kreditkarte spezifisch sind.

Alle erwähnten Angaben werden im Klartext übertragen und durch eine digitale Signatur des Händlers geschützt. Diese kann der Käufer allerdings nicht selber nachprüfen. Er berechnet statt dessen selber eine digitale Signatur der Felder, die mit der Antwortnachricht zurückgeschickt wird und später an den Server weitergeleitet und dort verglichen wird. Beziehen sich die Signaturen des Händlers und des Kunden nicht auf die gleichen Daten, wird die Transaktion nicht genehmigt.

Neben der Signatur der Transaktionsdaten, die im Klartext übertragen wird, enthält die Antwortnachricht im privaten Teil, der bei dieser Nachricht mit dem Schlüssel des CyberCash-Servers chiffriert wurde, den Betrag und die Kreditkartendaten, wie sie bei der Registrierung zurückgekommen waren.

Der Händler kann den privaten Teil zwar nicht lesen, kann mit seiner Hilfe aber bei Bedarf (wenn der Kunde z.B. später die Zahlung bei seiner Kreditkartenfirma rückgängig macht) die Kreditkartendaten von CyberCash erfragen. Dazu kommen wir aber noch.

Nach der Authorisierung der Zahlung durch den Server beendet der Händler die Transaktion mit einer Antwortnachricht. Diese enthält im Klartext:

- die ID's der Beteiligten
- die Transaktionsnummer und das Datum
- Erfolg/Fehlschlag
- die Signatur des Händlers (aus der ersten Nachricht)

- die Signatur des Kunden (aus der zweiten Nachricht)
- eine Mitteilung des Händlers an den Kunden

Der private Teil wird direkt aus der Authorisierungsmitteilung des Servers übernommen. Der Händler kann ihn nicht lesen. Er enthält noch einmal den Betrag, die Transaktionsnummer, die Kartendaten und eine Angabe ob die Transaktion ein Erfolg oder Fehlschlag war.

Die Protokollbeschreibung erwähnt noch, daß prinzipiell neben dieser Meldung vom Händler auch noch eine Nachricht direkt vom Server kommen kann.

Authorisierung und Vollzug der Zahlung

Authorisierung und Vollzug der Zahlung können in einem oder auch in zwei voneinander getrennten Schritten erfolgen. Hierzu sendet der Händler die verschlüsselten Kundendaten. Im privaten Teil enthält die Nachricht

- den Typ der Nachricht (nur Authorisierung/Authorisierung und Zahlung)
- die Transaktionsnummer
- den Betrag
- die beiden Signaturen des Transaktionsinhaltes
- das Datum

Als Antwort schickt der Server eine Nachricht, die neben der verschlüsselten Antwort an den Kunden (s.o.) im privaten Teil

- Erfolg/Fehlschlag
- die Transaktionsnummer
- die mehrfach erwähnten Signaturen
- eine Nummer, um eine noch nicht endgültig erfolgte Zahlung rückgängig zu machen
- eine Nummer, die später dazu benutzt werden kann, eine bereits authorisierte Zahlung endgültig zu machen

- Ein Hash der Kartennummer des Kunden und des Salzwertes.
- ein Kartenprefix oder eine Kartennummer, das hängt von der ausstellenden Bank ab[23]
- Verfallsdatum
- eine Nachricht an den Händler

Außerdem definiert das Protokoll noch zwei Nachrichten, um noch nicht endgültige Transaktionen rückgängig zu machen, bzw. um einen bereits eingezogenen Betrag zurückzuerstatten.

Zahlungen über das Telephon oder per Post

Um es dem Händler zu erlauben, auch Zahlungen per Post oder über das Telephon entgegen zu nehmen, wurden Protokollnachrichten definiert, die nicht voraussetzen, daß der Käufer eine Cybercash Persona besitzt.

Die Nachrichten sehen im Prinzip so ähnlich aus wie die im vorigen Abschnitt vorgestellten, sie enthalten aber kein Feld für die Persona ID des Käufers und jeder verschlüsselte Bereich wurde mit dem Schlüssel des Händlers chiffriert.

Erfragen der Kreditkartennummer des Kunden

Verweigert der Kunde bei seiner Kreditkartenfirma die Zahlung, so braucht der Händler die Kartennummer des Kunden, um sich direkt an besagte Firma wenden zu können. Daher wurde eine Nachricht implementiert, mit der der Händler die Kreditkartendaten eine Kunden erfragen kann.

Er sendet dazu die Daten der strittigen Transaktion zusammen mit einem Password an den Server, der ihm darauf hin

- die Kreditkartenfirma
- den Namen des Karteninhabers
- das Verfallsdatum der Kreditkarte

mitteilt. Der Server wird diesen ungewöhnlichen Vorfall im Logfile vermerken, so daß Händler, die diese Möglichkeit dazu nutzen, Kreditkartennummern zu sammeln, erkannt werden können.

[23]Es ist unwarscheinlich, das es sich hierbei um die Kreditkartennummer des Kunden handelt. Dann wäre das ganze Protokoll sinnlos. Wahrscheinlich identifiziert diese Nummer eher den Händler.

6.4.3.3 Aspekte von CyberCoin

Da uns zu CyberCoin keine technische Dokumentation vorliegt, bezieht sich dieser Abschnitt nur auf die FAQ's[24] *die unter [Cybercash 97] zu finden sind.*

CyberCoin wurde speziell entwickelt, um Zahlungen im Bereich von 0,25 - 10$ abzuwickeln. Dies äußert sich auch darin, daß nur 80$ pro Monat auf das Verechnungskonto eingezahlt werden dürfen.

In diesem Bereich währe ein reines Kreditkartensystem zu teuer, da die bei jeder Transaktion anfallenden Gebühren die zu bezahlenden Summen sogar übersteigen könnten.

Da das Geld nicht auf dem PC gelagert wird, sondern bei den Banken verbleibt, ist das Geld sogar gegen den Bankrott der jeweiligen Bank bei der FDIC versichert, so daß die amerikanische Regierung für den Betrag geradesteht[25]

Um sicherzustellen, daß der Kunde seine Ware auch erhält, unterstützt das System für virtuelle Waren z.B. Web-Seiten eine verschlüsselte Übertragung, bei der der Betrag erst bezahlt wird, wenn die Ware fehlerfrei angekommen ist. Nachdem bezahlt wurde, erhält der Kunde dann den Schlüssel, ohne den seine „Ware“ ein wertloser Byte-Salat ist.

Für physische Waren verlangt CyberCash, daß die Waren erst in Rechnung gestellt werden dürfen, nachdem sie auf den Weg geschickt wurden.

[24] **F**requently **A**sked **Q**uestions, eine Liste häufig gestellter Fragen und der dazu gehörigen Antworten.

[25] Es bleibt zu hoffen, daß dies mit der Einführung in Deutschland hier ähnlich gehandhabt wird.

Kapitel 7

Digitales Bargeld

7.1 NetCash

7.1.1 Motivation

Viele der zur Zeit eingesetzten Protokolle für Zahlungen in der Umgebung eines unsicheren Netzwerkes sind zwar durchaus für einem großen Teil von Transaktionen geeignet, die wenigsten schützen jedoch hierbei die Identität der beteiligten Parteien. Es gibt zwar auch viele Vorschläge für Protokolle, die anonyme Transaktionen unterstützen (z.B. DigiCash), diese haben jedoch meist den Nachteil, daß sie eine Zentralbank voraussetzen, die an allen Transaktionen beteiligt ist. Die Notwendigkeit einer solchen Zentralbank schränkt jedoch die Skalierbarkeit sehr stark ein. Die nichtanonymen Verfahren, vergleichbar z.B. mit Schecks, sind zwar wesentlich skalierbarer, legen jedoch bei jeder Transaktion die Identität der Beteiligten offen.

Das von der USC/ISI[1] GOST Group entwickelte NetCash kombiniert die Vorzüge anonymer Transaktionen mit der Skalierbarkeit der nichtanonymem Online-Zahlungsprotokolle. NetCash stellt eine Struktur zur Verfügung, die es erlaubt, anonyme elektronische Zahlungsvorgänge über ein unsicheres Netzwerk in Echtzeit abzuwickeln, ohne daß hierfür spezielle, gegen Manipulationen abgesicherte Hardware benötigt wird.

7.1.2 Grundidee

Die drei Parteien im NetCash-Protokoll sind der Kunde, der Verkäufer und sogenannte Währungs-Server. Die Infrastruktur von NetCash basiert auf ei-

[1] **I**nformation **S**ciences **I**nstitute der **U**niversity of **S**outhern **C**alifornia

nem Netz vieler dieser Währungs-Server, die eine Wechselstelle zwischen anonymem elektronischem Geld und nichtanonymen Zahlungsmitteln wie z.B. Schecks darstellen. Die Währungs-Server sind weitläufig verteilt, werden unabhängig voneinander verwaltet und bieten folgende Dienste an:

1. Münzüberprüfung, d.h. Aufdeckung von Mehrfachausgabe und Fälschungen

2. Münzaustausch, um das Verfolgen der Münzen zu verhindern

3. Kauf von Münzen gegen Schecks

4. Rücktausch von Münzen in Schecks

Der Kunde kann seinen Währung-Server frei wählen, z.B. aufgrund geographischer Nähe oder Höhe des Vertrauens. Vom Währungs-Server kauft der Kunde mit einem nichtanonymen Zahlungsmittel elektronische Münzen, die mit der Identität des Währungs-Servers signiert sind. Als gute Kombination bietet sich als nichtanonymes Zahlungsmittel z.B. das NetCheque-Protokoll an.

Aufgrund der Existenz vieler unabhängiger Währungs-Server ist NetCash skalierbarer als andere Verfahren. Die Währungs-Server akzeptieren neben den nichtanonymen Zahlungsmitteln wie z.B. Schecks auch die Münzen anderer Währungs-Server. Zum gegenseitigen Ausgleich zwischen den verschiedenen Währungs-Servern wird ein skalierbares aber nichtanonymes Verfahren verwendet (Auch hier bietet sich NetCheque an). Die Anonymität der Kunden ist in diesem Fall nicht gefährdet, da sich bei diesen Transaktionen nur die Währungs-Server gegenseitig identifizieren und authentisieren.

Einen absolute Anonymität wie z.B. bei DigiCash bietet NetCash jedoch nicht, denn wenn ein Kunde mit einem Scheck Münzen kauft oder Münzen zurücktauscht, kann der Währungs-Server aufzeichnen, welche Münzen er an welche Kunden ausgegeben hat. Es ist jedoch sehr wahrscheinlich, daß eine Vereinbarung mit den Kunden diese Aufzeichnungen ausschließt, und bei Nichteinhalten durch den Währungs-Server steht es dem Kunden frei, sich einen neuen vertrauenswürdigeren Server zu suchen.

Wenn die Münzen erst einmal gekauft sind, können sie eine ganze Weile im Umlauf sein, ohne daß die Identität der Beteiligten offengelegt wird. Obwohl der Währungs-Server jedesmal involviert ist, wenn eine Münze ihren Besitzer wechselt, so kann er nur aufzeichnen, welche Münze gegen welche eingetauscht wurde (auch wenn es ihm untersagt wurde), die Identität seines

Besitzers erfährt er jedoch nur, wenn dieser sich entschließt, die Münzen gegen einen Scheck zurückzutauschen. Beim einfachen Austausch der Münzen kann es sich um den Kunden selbst handeln, oder um einen anonymen Händler, der die Münzen soeben akzeptiert hat und sie jetzt überprüfen (und damit austauschen) lassen möchte, um sicherzugehen, daß sie nicht schon mehrfach ausgegeben wurden. Je länger die Kette der Beteiligten vor dem endgültigen Rücktausch ist, desto weniger Informationen lassen sich darüber gewinnen, wer was wo gekauft hat.

Man kann bei einer Transaktion auch ganz auf die Beteiligung des Währungs-Servers verzichten, dann kann die Münze allerdings nicht auf Echtheit oder Mehrfachausgabe überprüft werden, d.h. solche Transaktionen werden nur zwischen Parteien stattfinden, die sich gegenseitig uneingeschränkt vertrauen.

Die NetCash-Transaktionen können sogar teilweise offline erfolgen, d.h. beide Parteien sind während der engültigen Übergabe der Münzen nicht online mit dem Währungs-Server verbunden. Eine sichere Transaktion setzt jedoch voraus, daß zumindest eine der Parteien zu irgendeinem Zeitpunkt während einer Transaktion mit dem Währungs-Server Kontakt hatte. Wird absolute Anonymität oder ein kompletter Offline-Betrieb gewünscht, so kann die Struktur von NetCash für Transaktionen, die dies benötigen, so erweitert werden, daß sie einen Austausch mit anderen Verfahren unterstützt.

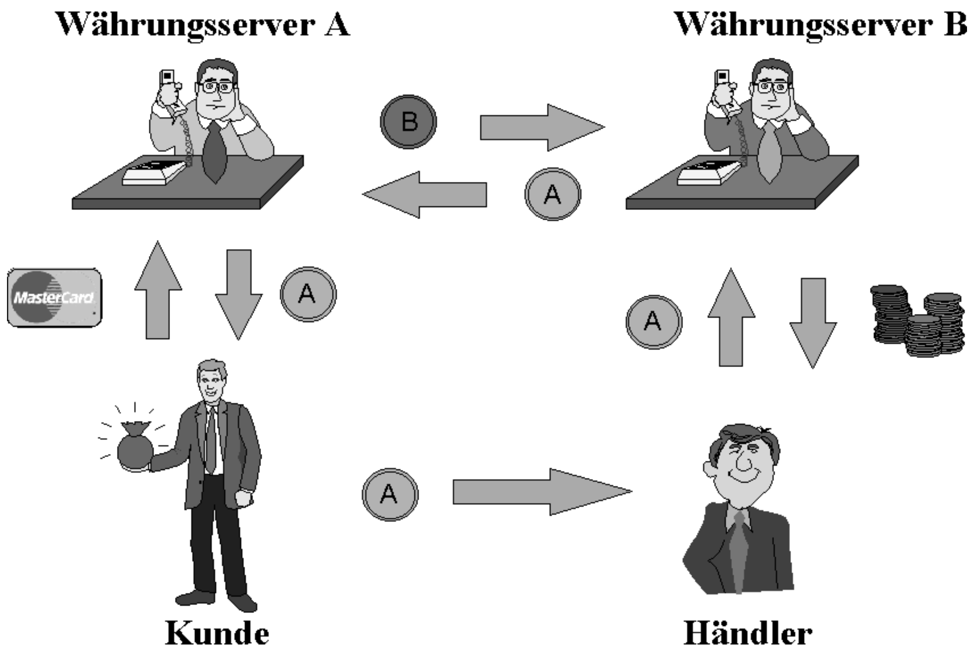

Abbildung 7.1: NetCash

7.1.3 Realisierung

7.1.3.1 Erwerben eines Zertifikats

Um einen Währungs-Server zu betreiben, muß man sich bei einer staatlichen Stelle[2] ein Zertifikat besorgen. Hierzu erzeugt der Währungs-Server ein Public-Key-Paar und schickt seinen öffentlichen Schlüssel über einen sicheren Kanal an die Zertifizierungsstelle. (Der entsprechende private Schlüssel wird zum Signieren der Münzen gebraucht). Die Zertifizierungsstelle stellt sicher, daß als Absicherung eine reale Geldmenge in Form von nichtanonymen Zahlungsmitteln auf den Namen des Währungs-Servers existiert und sendet daraufhin ein Zertifikat zurück, daß mit dem privaten Schlüssel der Zertifizierungsstelle verschlüsselt ist. Mit diesem Zertifikat wird für andere Währungs-Server und Finanzinstitutionen sichergestellt, daß es sich bei der neuen Währung um ein gültiges Zahlungsmittel handelt. Eine Kompromittierung des privaten Zertifizierungsschlüssels wäre verheerend. Das Zertifikat beinhaltet:

1. `Certif_id` - Eine einmalige ID, die den Währungs-Server identifiziert
2. `CS_name` - Den Namen des Währungs-Servers
3. `CS_key` - Den öffentlichen Schlüssel des Währungs-Servers
4. `issue_date` - Das Ausstellungsdatum des Zertifikats
5. `exp_date` - Das Ablaufdatum des Zertifikats

Das neu erhaltene Zertifikat ermächtigt den Währungs-Server, eigene digitale Münzen zu erzeugen und zu verwalten.

7.1.3.2 Struktur der digitalen Münzen

Eine digitale Münze besteht aus einem mit dem privaten Schlüssel des Währungs-Servers verschlüsselten und einem unverschlüsselten Teil. Der verschlüsselte Teil enthält:

1. `CS_name` - Den Namen des Währungs-Servers
2. `s_addr` - Die Internetadresse des Währungs-Servers
3. `exp_date` - Das Ablaufdatum der Gültigkeit der Münze

[2] In den USA z.B. die Federal Deposit and Insurance Corporation

4. `serial_num` - Die Seriennummer der Münze

5. `coin_val` - Den Wert der Münze

Der unverschlüsselte Teil besteht nur aus der ID des Zertifikates, der `Certif_id`. Diese ID verweist einen Kunden, der die Münze überprüfen lassen will, auf das entsprechende Zertifikat, so daß der Kunde den öffentlichen Schlüssel des betreffenden Währungs-Servers erhält. Die Gültigkeit einer Münze ist durch erfolgreiches Entschlüsseln der Münze mit dem öffentlichen Schlüssel des Währungs-Servers bewiesen, aber die Tatsache, daß diese Münze noch nicht mehrmals ausgegeben wurde, ist solange nicht sichergestellt, bis die Münze direkt beim Währungs-Server gegen neue Münzen eingetauscht worden ist.

Die Internetadresse `s_addr` des Währungs-Servers ist Teil der Münze, so daß die Münze direkt zu dem Server zurückgesendet werden kann, der sie ausgegeben hat. Ein Währungs-Server ist implementiert als ein Verbund von Servern, die an ein Netzwerk angeschlossen sind und unter einer kollektiven Adresse im Internet erreichbar sind. Wenn der Währungs-Server unter dieser Adresse nicht erreichbar ist, so kann der Name des Servers `CS_name` dazu benutzt werden, den Server durch eine Anfrage an einen Directory-Server zu finden.

Die Seriennummer `serial_num` ordnet die Münze eindeutig dem Währungs-Server zu, der sie ausgegeben hat. Der Währungs-Server führt eine Liste aller Seriennummern ausstehender Münzen. Wird während einer Transaktion eine Münze zur Überprüfung an den Währungs-Server geschickt, so vergleicht der Server die Seriennummer mit seiner Liste. Wenn die Seriennummer gefunden wird, handelt es sich um eine gültige Münze, die bisher noch nicht ausgegeben wurde. Die entsprechende Seriennummer wird nun aus der Liste gestrichen und eine neue Münze mit neuer Seriennummer, die natülich sofort einen Eintrag in der Liste erhält, an den Kunden ausgegeben. Wird die Seriennummer jedoch nicht in der Liste gefunden, so handelt es sich vermutlich um einen Versuch der Mehrfachausgabe einer Münze, also wird der Umtausch verweigert. Die Münze enthält ein Ablaufdatum `exp_date` ihrer Gültigkeit, um den Umfang der von den Währungs-Servern geführten Listen einzuschränken.

7.1.3.3 Austausch mit dem Währungs-Server

Eine Anfrage an den Währungs-Server seitens des Kunden enthält folgende Informationen:

1. `transaction` - Den Grund seiner Anfrage, z.B. ob er neue Münzen oder einen Scheck will, und wenn er einen Scheck will, den Namen, auf den der Scheck ausgestellt werden soll

2. `instrument` - Den Gegenstand der Anfrage (entsprechend Münzen oder einen Scheck)

3. `Sym_Key` - Einen neuen für diese Transaktion zu benutzenden symmetrischen geheimen Schlüssel

All diese Informationen werden mit dem öffentlichen Schlüssel des Währungs-Servers verschlüsselt an diesen geschickt.

Enthält die Anfrage des Kunden Münzen, die der Währungs-Server selbst ausgegeben hat, so werden diese Münzen auf Mehrfachausgabe überprüft, indem man ihre Seriennummern in der Liste der noch ausstehenden Münzen sucht. Handelt es sich um Münzen eines fremden Währungs-Servers, so kontaktiert der örtliche Server den fremden Server und bittet ihn, die Münzen gegen einen auf den örtlichen Währungs-Server ausgestellten Scheck auszutauschen. Dieser Scheck wird dann später über das bereits existierende globale nichtanonyme Verrechnungssystem ausgeglichen. Erhält der Währungs-Server vom Kunden einen an sich ausgestellten Scheck, so schreibt er den entsprechenden Betrag seinem Guthaben zu. Je nachdem, was der Kunde verlangt hat, erhält er vom Währungs-Server jetzt neue Münzen oder einen auf die in der Anfrage genannte Partei ausgestellten Scheck. Die Verschlüsselung der Antwort mit dem vom Kunden in seiner Anfrage mitgeschichten Schlüssel beweist die Identität des Währungs-Servers (da nur ihm dieser Schlüssel mitgeteilt wurde) und schützt außerdem vor Manipulationen potentieller Angreifer.

7.1.3.4 Transaktion zwischen Kunde und Verkäufer

Es wird angenommen, das der Kunde die Adresse des Händlers kennt. Falls der Kunde den Public-Key des Händlers noch nicht kennt, so kann er einen den Händler indentifizierenden Public-Key von einem Public-Key-Server erhalten.

Falls der Händler anonym bleiben will, wird ein anonymes Public-Key-Paar nur für diese Transaktion generiert. Ist dies der Fall, so bietet das Protokoll Schutz gegen passive Abhörangriffe, da die Anonymität beider Parteien gewahrt bleibt. Es besteht jedoch kein Schutz gegenüber einem aktiven

Angriff, bei dem ein unbekannter Dritter vorgibt, der anonyme Händler zu sein.

Mit Kenntnis des Public-Keys des Händlers kann an ihn eine Anfrage gesendet werden, die folgende Daten enthält:

1. `coins` - Die zur Bezahlung dienenden Münzen
2. `S_id` - Einen Bezeichner für den gewünschten Dienst
3. `Sym_Key` - Einen neuen für diese Transaktion zu benutzenden symmetrischen geheimen Schlüssel
4. `Ses_Key` - Einen Sitzungsschlüssel

All diese Daten werden mit dem öffentlichen Schlüssel des Händlers verschlüsselt. Optional kann noch das Zertifikat des Währungs-Servers mitgeschickt werden, der die zur Bezahlung dienenden Münzen ausgegeben hat. Dieses Zertifikat kann sich der Händler aber auch direkt von dem Währungs-Server oder von einem Verteiler beschaffen.

Um einen Zusammenhang zwischen den Münzen und der zugehörigen Transaktion zu schaffen, extrahiert der Händler den Sitzungsschlüssel. Zu dem Zeitpunkt, an dem der gewünschte Dienst stattfinden soll, stellt der Händler sicher, daß der Kunde den Sitzungsschlüssel `Ses_Key` kennt. Diese Schritte sind wichtig, falls die Transaktion teilweise offline stattfinden soll.

Der Händler benutzt den öffentlichen Schlüssel des Währungs-Servers, um zu überprüfen, ob ein zertifizierter Währungs-Server die Münzen ausgegeben hat, d.h. ob es sich um gültige Münzen handelt. Will der Händler auch noch sicherstellen, daß die Münzen nicht bereits ausgegeben worden sind, so stellt er jetzt eine Anfrage zur Überprüfung an den entsprechenden Währungs-Server.

Sind die Münzen überprüft, schickt der Händler eine mit dem nur für diese Transaktion generierten symmetrischen geheimen Schlüssel `Sym_Key` verschlüsselte und mit seinem privaten Schlüssel signierte Bestätigung an den Kunden. Diese Bestätigung enthält folgende Daten:

1. `amount` - Die Höhe des bezahlten Betrages
2. `date` - Das aktuelle Datum
3. `T_id` - Einen einmaligen Bezeichner, der zusammen mit dem Sitzzungsschlüssel dazu benutzt wird, den gewünschten Dienst zu leisten.

Dieser Transaktionsablauf schützt zwar den Händler vor Betrug seitens des Kunden (Mehrfachausgabe von Münzen), er verhindert jedoch nicht, daß der Händler die vom Kunden erhaltenen Münzen einfach ausgibt, ohne eine gültige Bestätigung an den Kunden geschickt zu haben. Aus diesem Grund gibt es folgende Erweiterung des Protokolls:

7.1.3.5 Gegenseitiger Schutz vor Betrug

Um den Händler vor Mehrfachausgabe der Münzen zu schützen und dem Kunden eine Garantie zu geben, daß er entweder eine gültige Bestätigung oder sein Geld zurückbekommt, muß die Struktur der Münze derart erweitert werden, daß sie an eine bestimmte Partei und an ein festgelegtes Zeitintervall gebunden ist. Der Kunde fordert für diesen Fall spezielle Münzen von seinem Währungs-Server an. Dabei handelt es sich um Münz-Tripel bestehend aus `coin_client`, `coin_vendor` und `coin_extra`. Alle Teilmünzen des Tripels haben die gleiche Seriennummer und den gleichen Nennwert.

Die Teilmünzen enthalten unterschiedliche Zeitstempel, so daß sie nur in bestimmten aufeinanderfolgenden Zeitintervallen Gültigkeit haben. Im ersten Zeitintervall hat nur die `coin_vendor` Gültigkeit, im zweiten Intervall die `coin_client` und im letzten die `coin_extra`. Außerdem enthalten die ersten beide Teilmünzen `coin_client` und `coin_vendor`, die für den Kunden und den Händler gedacht sind, bestimmte Schlüsselinformationen. Will eine der Parteien seine Münze benutzen, so muß sie die Kenntnis des entsprechenden Schlüssels beweisen.

So enthält `coin_vendor` beispielsweise den öffentlichen Schlüssel des Händlers, und wenn der Händler diese Münze bei einer Transaktion mit dem Währungs-Server benutzen will, muß er die Kenntnis seines privaten Schlüssels beweisen. `coin_client` enthält zusätzlich zum öffentlichen Schlüssel des Kunden noch den öffentlichen Schlüssel des Händlers, um die Menge der Informationen zu reduzieren, die der Währungs-Server verwalten muß, um dem Kunden eine Bestätigung auszustellen. Weiterhin benötigt man ein zusätzliches Bit an Information, daß man mit der Seriennummer der Münze in der Datenbank des Währungs-Servers assoziiert, um festzustellen, ob der Kunde oder der Verkäufer die Münze ausgegeben hat.

Bei einer Transaktion behält der Kunde zunächst `coin_extra` und `coin_client` und gibt `coin_vendor` an den Händler weiter. Der Händler kann diese Münze jetzt während des ersten nur für die `coin_vendor` gültigen Zeitintervalls einlösen. Erhält der Kunde vom Händler keine Bestätigung, kann er nach Ablauf des für die `coin_vendor` gültigen Zeitintervalls eine Anfrage an den Währungs-Server schicken und feststellen lassen, ob der Händler die Münze bereits eingelöst hat. Ist dies der Fall, erhält der Kunde vom Währungs-Server eine Bestätigung, die den Nennwert der Münze und den Public-Key des Händlers enthält. War der Händler jedoch anonym, wird diese Information dem Kunden wenig nützen.

Wurde die Münze aber noch nicht eingelöst, so kann der Kunde in dem Zeitintervall, in dem seine Münze Gültigkeit hat, diese gegen eine neue, gegebenenfalls nicht aus einem Tripel bestehende Münze eintauschen. Der Händler muß seinerseits `coin_vendor` im Auge behalten, bis das Gültigkeitsintervall abgelaufen ist, für den Fall, daß der Kunde `coin_vendor` beim Händler doppelt ausgeben will. `coin_extra`, die keine Schlüsselinformationen enthält und von jedem ausgegeben werden kann, bietet zusätzliche Flexibilität, falls sich der Kunde letztendlich dazu entschließt, die Münze doch nicht bei dem entsprechenden Händler auszugeben.

7.1.4 Offline-Protokolle

Bei einer Offline-Transaktion möchte man sowohl die Mehrfachausgabe von Münzen verhindern, als auch die Anonymität der beteiligten Parteien gewährleisten. Das NetCash-Protokoll mit der in 7.1.3.5 beschriebenen Erweiterung läßt sich hierzu wie folgt verwenden: Wenn der Kunde im Voraus weiß, daß er bei einem bestimmten Händler etwas erwerben möchte, so beschafft er sich rechtzeitig die entsprechend markierten Münzen bei seinem Währungs-Server. Zu einem späteren Zeitpunkt kann er diese Münzen dann zur Bezahlung bei dem entsprechenden Händler benutzen. Bis zum endgültigen Abschluß der Transaktion wird ein mehrfaches Ausgeben der Münzen verhindert und die Anonymität des Kunden gewahrt. Der Nachteil dieses Verfahrens ist natürlich, daß man im Voraus wissen muß, mit wem man seine Geschäfte tätigen will.

Eine andere Möglichkeit von Offline-Transaktionen ist die Verwendung des einfacheren Protokolls aus 7.1.3.4 in Verbindung mit manipulationssicheren elektronischen Geldbörsen. Hier wird eine Mehrfachausgabe der Münzen durch Eigenschaften der Hardware verhindert.

7.1.5 Eine Implementierung von NetCash bei der NetBank

Von den Software Agents, Inc. wurde die NetCash-Idee mit einigen Abweichungen bei der NetBank [NetBank 97] implementiert.

Um sogenannte NetCash Koupons von der Netbank zu erwerben, braucht ein Kunde ein Konto bei einer US-Bank. Besitzt der Kunde ein solches Konto nicht, so kann er auch US-Währung in Form von Schecks, Überweisungen, American Express Travelers Cheques oder Bargeld auf dem Postweg der NetBank zukommen lassen, um so seine NetCash Coupons zu erhalten.

Der einfachste Weg ist jedoch das Anfordern der NetCash Koupons von der NetBank per e-mail. Hierzu schickt der Kunde ein vorgefertigtes Anforderungsformular an die NetBank und erhält daraufhin unverzüglich seine NetCash Coupons.

Ein NetCash Koupon setzt sich aus dem Schlüsselwort "`NetCash US$`", der Betragshöhe und einer Seriennummer zusammen, also z.B.:

```
NetCash US$ 25.00 A123456B789012C
```

Da NetBank zwar die entsprechenden Koupons sofort an den Kunden ausgiebt, die von dem Kunden angegebene Bankverbindung aber nicht unverzüglich überprüfen kann, haben die Koupons zunächst den Status "`pending`". Dieser Status bedeutet, daß eine Händler, der diesen Koupon als Bezahlung erhält, ihn bei der NetBank erst einlösen kann, wenn diese das Geld von dem Kunden erhalten hat, der mit diesem Koupon bezahlt hat. Diese Verzögerung beträgt in etwa zwischen einer und zwei Wochen.

Ist seit dem Zeitpunkt, an dem ein Kunde seine NetCash Koupons von der Netbank erhalten hat, und dem Zeitpunkt der Bezahlung beim Händler schon eine kurze Zeit vergangen, so ist es sehr wahrscheinlich, daß die NetBank inzwischen das Geld über die Bankverbindung des Kunden erhalten hat. Der Status des Koupons wechselt in diesem Fall von "`pending`" zu "`valid`", d.h. er kann vom Händler unmittelbar nach Erhalt bei der NetBank eingelöst werden.

Um festzustellen, welchen Status ein NetCash Koupon hat, schickt der Händler nach Erhalt des Koupons vom Kunden sofort eine e-mail an die NetBank. Diese Nachrichten werden bei der NetBank automatisch abgearbeitet und haben eine festgelegte Syntax, d.h. sie beginnen immer mit dem Schlüsselwort "`NetCash US$`", dann folgen Betragshöhe, Seriennummer und am Ende ein Kommando, in diesem Fall der Befehl "`accept`".

Als Antwort auf diese Anfrage erfährt der Händler den Status dieses Koupons, also entweder "`pending`", "`accepted`" oder "`rejected`", falls es sich um einen versuchten Betrug, z.B. eine Mehrfachausgabe handelt. Im Falle von "`pending`" oder "`accepted`" wird bei dieser Anfrage der NetCash Coupon gegen einen neuen Koupon gleichen Wertes aber unterschiedlicher Seriennummer ausgetauscht. Der alte Koupon wird aus dem Verkehr genommen, um so Mehrfachausgaben zu verhindern. Mit Hilfe des "`accept`"-Kommandos werden auch die einzelnen Koupons einer e-mail zu einem einzigen Koupon mit neuer Seriennummer zusammengefaßt. Auf diese Weise braucht man nur eine geringe Anzahl von Seriennummern "im Auge zu behalten".

Eine Händler hat natürlich die Möglichkeit, nur Koupons zu akzeptieren, bei denen er als Antwort "`accepted`" erhält, da er nur diese bei der NetBank sofort einlösen kann. Hierduch verhindert er aber, daß Kunden bei ihm spontane Käufe tätigen, d.h. daß die Kunden sich spontan NetCash Koupons besorgen um so sofort bei ihm einkaufen zu können.

Außer "`accept`" kenn die NetBank noch weitere Kommandos, und zwar "`verify`", "`change`" und "`deposit`". Eine e-mail an die NetBank kann mehrere dieser Kommandos beinhalten.

Das "`verify`"-Kommando dient wie "`accept`" auch zur Überprüfung der Gültigkeit eines Koupons, allerdings mit dem Unterschied, das bei "`verify`" keinen neue Seriennummer ausgegeben wird.

Mit "`change`" hat der Kunde die Möglichkeit, seine vorhandenen NetCash Koupons in andere Einheiten umzutauschen, also z.B. einen 5$ Koupon in fünf einzelne Koupons a 1$. Ein Kunde kann seine Koupons so oft einwechseln, wie er möchte, es entstehen ihm dadurch keine Kosten.

Das "`deposit`"-Kommando dient dem Händler dazu, seine gesammelten NetCash-Koupons bei der NetBank wieder in reales Geld einzutauschen.

7.1.5.1 Sicherheit der NetBank Transaktionen

Die e-mail Nachrichten mit den NetCash-Transaktionen liegen im Klartext vor, so daß ein Angreifer die versendeten Koupons einfach abfangen und selbst ausgeben kann. Um dieses Problem zu beheben, akzeptiert die NetBank neben den unverschlüsselten Nachrichten auch solche, die mit PGP verschlüsselt sind. NetBank empfiehlt jedem Benutzer, diese Möglichkeit der Verschlüsselung zu nutzen, unterstützt jedoch auch weiterhin Nachrichten im Klartext, da sie nicht davon ausgeht, das alle Benutzer mit dem Umgang dieser Verschlüsselungsverfahren vertraut sind.

7.1.5.2 Organisatorisches im Bezug auf die Netbank

NetBank berechnet dem Kunden keine Transaktionsgebühren, so daß die NetCash Koupons auch für die Bezahlung relativ kleiner Beträge eingesetzt werden können. Dem Händler wird beim Einlösen seiner gesammelten Net-

Cash Koupons bei der NetBank eine Gebühr berechnet, die sich nicht pro Transaktion, sondern prozentual nach der eingelösten Summe berechnet.

Zur Zeit existiert eine Obergrenze von 100$, die der Verkaufspreis einer vom Händler angebotenen Ware nicht überschreiten darf, wenn mit Hilfe von NetCash Koupons bezahlt werden soll.

Software Agent, Inc., das BankNet Zahlungssystem und alle seine Angestellten übernehmen keine Haftung für eventuelle Schäden und Verluste, die durch die Benutzung von NetCash Koupons entstanden sind.

7.2 Die Geldkarte

Obwohl die Geldkarte nicht direkt in die Kategorie "Zahlungverfahren im Internet" einzuordnen ist, so sei sie jedoch an dieser Stelle der Vollständigkeit halber und aufgrund ihrer zunehmenden Verbreitung erwähnt.

7.2.1 Motivation

Laut [Heirig ??] ist das Bestreben der Anfang 1996 ins Leben gerufenen Idee der Geldkarte, zusätzliche Handelsbereiche, die bislang noch nicht für die Bezahlung mit Kartenzahlungsverfahren gewonnen werden konnten, für das bargeldlose Zahlen zu erschließen.

Im Gegensatz zu dem bekannten electronic-cash-Verfahren mit der Euroscheckkarte, das aufgrund höherer Investitionen und laufender Kommunikationskosten für die Bezahlung von kleineren Beträgen nicht so geeignet ist, ist die Geldkarte für die häufig anfallenden Zahlungen von Kleinbeträgen zwischen 5,- DM und 25,- DM gedacht.

Das geplante Anwendungsgebiet erstreckt sich vom Automatenverkauf, also zum Beispiel Fahrkarten, Zigaretten und Parkscheine, über die Verwendung in Telefonzellen, Taxen und an Tankstellen bis hin zum Einsatz in Supermärkten, Kaufhäusern und im Einzelhandel, also Branchen mit niedrigen Kaufbeträgen und hoher Kundenfrequenz, und soll Kunden und Händler von der Handhabung größerer Mengen an Kleingeld entlasten.

7.2.2 Grundidee

Die Idee der Geldkarte ist die Erweiterung der weit verbreiteten EC- oder Kundenkarten der Banken um die Funktion einer sogenannten "elektronischen Geldbörse". Diese Erweiterung geschieht mit Hilfe eines zusätzlichen Microchips auf den Karten, die bisher größtenteils nur mit einem Magnetstreifen bestückt waren.

Diese elektronische Geldbörse kann von seinem Inhaber bei der Bank "aufgeladen" werden, aus Gründen der Risikominimierung jedoch nur bis zu einem Höchstbetrag von 400,- DM. Gleichzeitig wird das Girokonto des Inhabers um die Höhe des soeben auf die Karte übertragenen Betrages belastet.

Das Geldkartenverfahren wird von der gesamten deutschen Kreditwirtschaft getragen, die Geldkarten werden ausschließlich von Kreditinstituten ausgegeben, denen auch die Verwaltung der Aufladungsgegenwerte obliegt.

Für Personen, die über kein Konto verfügen, zum Beispiel Minderjährige oder Touristen, werden kontounabhängige Geldkarten angeboten, die gegen Barzahlung oder gegen Zahlung über andere Karten aufgeladen werden können. Die nach dem Aufladevorgang im Chip der Karte gespeicherten Geldeinheiten können nun zum Bezahlen an entsprechend ausgerüsteten Terminals verwendet werden.

Das Kaufterminal zeigt dem Kunden zunächst den zu zahlenden Betrag an, woraufhin der Kunde den Betrag bestätigen kann und seine Geldkarte in das Kaufterminal einführt. Die Bezahlung erfolgt nun ohne Leistung einer Unterschrift oder Eingabe einer Geheimzahl, der zu zahlende Betrag wird lediglich von der Karte des Kunden abgebucht und auf die Karte des Händlers bzw. des Kaufterminals übertragen.

Aufgrund der niedrigen Beträge pro Transaktion wird der Zahlungsverkehr zur Verrechnung der Geldbörsenumsätze nicht auf Einzeltransaktionsebene abgewickelt, sondern die bei den Kaufterminals erfaßten Einzeltransaktionen werden gesammelt und an eigens dafür eingerichtete Evidenzstellen übergeben, die die Beträge dann mit Hilfe von Verrechnungsbanken austauschen.

Das Risiko beim Verlust der Geldkarte entspricht dem Verlust von Bargeld (da es sich bei der Geldkarter um eine Vorausbezahlung handelt) und ist deshalb von Karteninhaber zu tragen. Der Verlust des Karteninhaltes aufgrund einer Funktionsuntüchtigkeit der Karte, soweit diese nicht mutwillig herbeigeführt wurde, wird jedoch von den Kreditinstituten getragen.

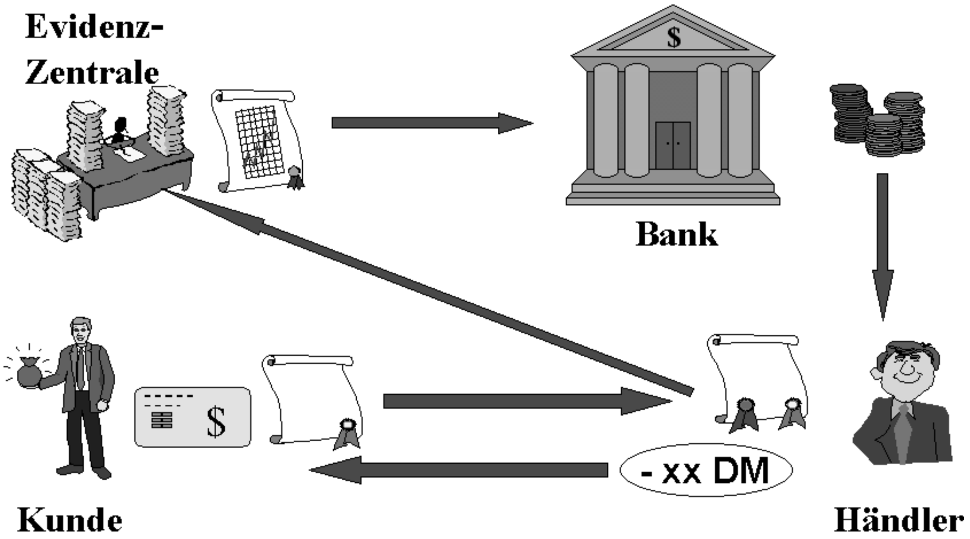

Abbildung 7.2: Geldkarte

7.2.3 Realisierung

7.2.3.1 Aufladen der Geldkarte

Das Aufladen der Geldkarte ist bei der Hausbank kostenlos und kann ausschließlich an entsprechend ausgestatteten Terminals erfolgen, die über eine Online-Verbindung mit dem Autorisierungssystem der Bank des Kunden verfügen.

Der Kunde schiebt seine Karte in ein solches Terminal und gibt seine PIN ein. Nach der Überprüfung der PIN und der übrigen Kartendaten zeigt das Terminal dem Kunden den aktuellen Restbetrag und den maximal möglichen Ladebetrag an. Nach der Eingabe des gewünschten Ladebetrages wird eine Autorisierungsanfrage an das für die Geldkarte zuständige Autorisierungssystem aufgebaut, die die mit einem Kryptogramm der Karte verschlüsselten Daten der Geldkarte enthält. Diese Anfrage wird durch das Autorisierungssystem geprüft, welches daraufhin eine durch ein Kryptogramm des Autorisierungssystems gesicherte Antwort mit dem neuen Betrag der Geldkarte zurücksendet. Diese Daten werden vom Terminal an die Karte des Kunden weitergeleitet, die dann eine Echtheitsprüfung der Daten durchführt, bevor die Kartendaten endgültig aktualisiert werden.

Gleichzeitig erfolgt eine Belastung des Kundenkontos und eine Gutschrift über die Höhe des auf die Karte überschriebenen Betrages auf ein eigens für diese Zwecke von der Bank eingerichtetes Verrechnungskonto.

7.2.3.2 Bezahlung mit der Geldkarte

Im Gegensatz zum Aufladevorgang wird der Bezahlvorgang immer offline durchgeführt. Auch die die Zahlung empfangende Partei, sei es Mensch oder Maschine, verfügt über eine Karte, auf der die erhaltenden Beträge gespeichert werden. Als Schnittstelle zwischen den beiden Parteien dient ein Terminal, das dem Kunden zunächst die Höhe des zu zahlenden Betrages anzeigt.

Nachdem der Kunde den Betrag bestätigt hat, steckt er seine Geldkarte in das Terminal, welches daraufhin die Karte identifiziert und eine Plausibilitätsprüfung der auf der Karte gespeicherten Daten durchführt. Diese Daten werden dann einschließlich der Höhe des zu zahlenden Betrages an die Karte der die Zahlung empfangenden Partei gesendet.

Diese Karte führt ihrerseits wiederum eine Plausibilitätsprüfung der ihr übergebenden Daten durch. Ist der vom Kunden zu zahlende Betrag nicht höher als die auf der Kundenkarte gespeicherte Geldmenge, wird ein Kommando generiert, daß die Kundenkarte zur Abbuchung veranlaßt.

Die Geldkarte des Kunden reduziert daraufhin die auf ihr gespeicherte Geldmenge um den zu zahlenden Betrag, aktualisiert ihr Transaktionslog und übergibt einen mit einem Echtheitszertifikat der Kundenkarte gesicherten Datensatz, der die Transaktionsdaten beinhaltet, an die Karte der die Zahlung empfangenden Partei. Diese Karte addiert nun den empfangenden Betrag zu der auf ihr gespeicherten Geldmenge und generiert ihrerseits ein Zertifikat, mit dem die Transaktionsdaten verschlüsselt und im Terminal gespeichert werden (Die Transaktionsdaten beinhalten das Kaufdatum und den gezahlten Betrag sowie ein weiteres Datum, aus dem sich die Kartennummer der Kunden- und der Empfängerkarte erzeugen läßt). Der Abschluß der Transaktion wird dem Kunden vom Terminal mitgeteilt. Auf der Chipkarte werden die letzten 15 Transaktionen gespeichert und können, z.B. durch einen sogenannten "Taschenleser", jederzeit überprüft werden.

Da die über die Geldkarte abgewickelten Umsätze grundsätzlich gegenüber dem Kartenakzeptanten garantiert sind, wird ihm hierfür ein Entgeld (zur Zeit 0,3 Prozent der Kaufsumme, mindestens aber 2 Pfennig) berechnet, daß den kartenausgebenden Instituten zufließt.

7.2.4 Verrechnung der Geldkartenumsätze

Wie schon erwähnt, werden aufgrund der niedrigen Umsätze pro Transaktion die einzelnen Geldkartenumsätze beim Händler zunächst gesammelt und später auf der Basis von Aggregaten abgewickelt.

Zum Zwecke der Reklamationsbearbeitung, der Echtheitsprüfung von Einzeltransaktionen und der Erkennung von Mehrfacheinreichungen werden von den Kreditinstituten sogenannte Evidenzstellen eingerichtet. Die bei den Händlern erfaßten Einzeltransaktionen werden von diesen Evidenzstellen entgegengenommen, kontrolliert, gesammelt und für die Einleitung in den Zahlungsverkehr vorbereitet.

Dieser Zwischenschritt ist notwendig, da für die Kreditinstitute auf der Basis von Aggregaten keine Möglichkeit der Einzelkontrolle besteht. Durch Zusammenwirken von Kundenbank und Evidenzstellen ist es jedoch möglich, aus den von den Evidenzstellen entgegengenommenen Einzeltransaktionsdatensätzen den Personenbezug der Transaktionsdaten herzustellen um so eine Reklamation zu bearbeiten und gegebenenfalls Beträge zu erstatten.

Die anschließende Abwicklung des Zahlungsverkehrs erfolgt über von den Evidenzstellen zwischengeschaltete Verrechnungsbanken. Evidenzstellen übergreifende Transaktionen werden hierbei zunächst nur unter den von den Evidenzstellen eingeschalteten Verrechnungsbanken ausgetauscht, um die bei den kartenausgebenden Kreditinstituten geführten Verrechnungskonten nur einmal pro Abrechnungsperiode zu belasten.

7.2.5 Ausblick

Mehrere Feldversuche haben gezeigt, daß es der Geldkarte noch an einer breiten Akzeptanz mangelt, da der Großteil der ausgegebenden Karten nur einmalig aufgeladen wurde. Beispiel hiefür ist der im März 1996 durchgeführte Feldversuch in Ravensburg/Weingarten. Der Stand[3] dieses Feldversuches war Ende September 1996:

- Aktive Karten: 30.000
- Anzahl Aufladungen (kumuliert): 50.000
- Anzahl Bezahlvorgänge (kumuliert): 155.000
- durchschnittlicher Aufladebetrag: 115,-
- durchschnittlicher Bezahlbetrag: 27,-

Gegen einen großflächigen Einsatz sprechen auch die für die Händler zusätzlich entstehenden Investitionen und die geringere Verdienstmöglichkeit aufgrund der zu zahlenden Gebühren. Da jedoch jede neu asgegebene EC-Karte mit einem Microchip ausgestattet ist, trägt der Kunde die Funktionalität der "elektronischen Geldbörse" mit sich herum, ob er will oder nicht.

7.3 Mondex

7.3.1 Motivation

Auch die Britischen Banken NatWest und British Midland haben das Potential des Internet als Marktplatz erkannt. Sie haben daher begonnen ein System zu entwickeln, das im Gegensatz zu denen vieler Mitbewerber ein elektronisches Äquivalent für Bargeld darstellen soll.

Da Bargeld keines Clearings bedarf, kann Geld sofort transferiert werden, ohne daß erst auf die Genehmigung durch einen Bankserver gewartet werden muß.

Auch die Bankgebühren entfallen für die meisten Transaktionen solange das Geld nicht wieder auf ein Konto eingezahlt wird. Vergleicht man dies

[3] Angaben aus [Martin 97]

mit Kreditkarten, bei denen der Händler bei jeder Transaktionn eine Gebühr zahlen muß, so dürfte dies doch eine Erleichterung insbesondere für kleine Geschäfte darstellen.

Da mit diesem System nur Geld ausgegeben werden kann, das man auch tatsächlich besitzt, können auch Kinder oder Jugendliche, die mangels Kreditwürdigkeit oder aus juristischen Gründen keine Kreditkarte erwerben können, im Internet einkaufen.

Um aber Bargeld wirklich effektiv zu realisieren, soll Bargeld auch zwischen Privatpersonen transferierbar sein.

> „'Direkte Transferierbarkeit' ist einer der Reize des Bargeldes, der so offensichtlich ist, daß man ihn allzuleicht übersieht – bis man versucht, seinen Kindern ihr Taschengeld zu zahlen, indem man ihnen seine Kreditkarte zeigt. Es gibt viele Privatpersonen die weder Visa noch AMEX 'nehmen' – oder in der Tat jede andere Karte außer Geburtstagskarten.“ [Mondex 96a]

In der Tat geht das Konzept über ein reines Internetzahlungsmittel hinaus. Durch die Verwendung von Chipkarten soll auch die Bezahlung von Waren in der physikalischen Welt, z.B. im Tante Emma Laden um die Ecke möglich sein.

Am 18.Juli 1996 verkündete Natwest die Gründung von Mondex International einem Konsortium von 17 Großbanken weltweit, die mit 250 Produzenten in 20 Ländern zusammenarbeiten. Die Einführung des Systems ist in Großbrittanien, Kanada, Hongkong, Australien, Neuseeland und den USA geplant. Den aktuellen Stand findet man auf [Mondex 96a].

7.3.2 Grundidee

Geld wird auf einer Chipkarte gespeichert. Dabei können bis zu fünf Währungen gleichzeitig auf die Karte geladen werden. Das Währungskonzept ist so aufgebaut, daß es jederzeit möglich ist neue Währungen (wie z.B. den Euro) zu definieren.

Verbindet man nun zwei Karten mit einem geeigneten Gerät, so ist es möglich, Geld von einer Karte auf die andere zu transferieren. Dies geschieht, indem die Karten die jeweilige Gegenkarte authentisieren und und dann den jeweils neuen Wert annehmen.

Neben ihren jeweiligen Beträgen in den verschiedenen Währungen enthält die Karte noch eine 16 Bit ID Nummer, eine Bezeichnung des Kunden (z.B. „Al's Shoes"), sowie einen Kontoauszug der letzten Transaktionen mit Datum, Summe und der Bezeichnung des Gegenüber.

Eine PIN wird einzig und allein dazu benutzt, die Karte zwischen Transaktionen auf Wunsch des Besitzers zu sperren. Ist die Karte geöffnet, findet keine weitere Authentisierung statt.

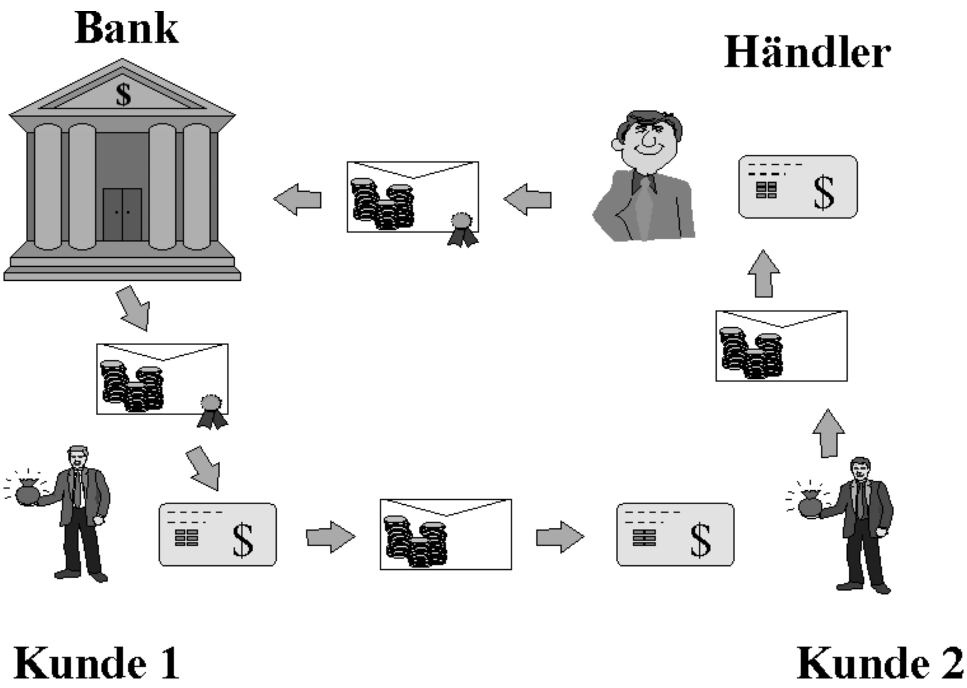

Abbildung 7.3: Ecash

7.3.3 Realisierung

Viele konkrete Details der Umsetzung der Grundidee werden leider nicht offengelegt. Es gibt zwar viele Seiten Information unter [Mondex 96a], aber das meiste erinnert an Hochglanzprojekte der Banken, viele schöne Worte aber wenig konkretes. Am detailliertesten wurde noch Tim Jones, der Erfinder von Mondex, in einer Vorlage für das Repräsentantenhaus der Vereinigten Staaten ([Jones 96]).

7.3.3.1 Hardware

Der Chip, der für Mondex benutzt wird, ist eine speziell maßgeschneiderte Anwendung des Hitachi H8/310. Neue und schnellere Chips werden verwendet werden, wenn sie zur Verfügung stehen um stets auf dem Stand der Technik zu bleiben. Die Chipkarte genügt den ISO - und GSM - Normen bezüglich Widerstandsfähigkeit gegen Hitze, Kälte, Feuchtigkeit, Röntgenstrahlen und elektronischer Interferenz.

Um jederzeit die Kartensalden kontrollieren zu können, gibt es ein spezielles Gerät, um diese auszulesen. Ein weiteres Gerät „Brieftasche“ genannt, dient dazu, zwei Karten zu verbinden und Geld zwischen ihnen zu transferieren. Auch der Geldtransfer zwischen speziellen Telefonen ist vorgesehen. Für den Transfer über das Internet wurde schließlich ein Kartenleser entwickelt, der an die serielle Schnittstelle des Computers angeschlossen werden kann.

Selbstverständlich wird es auch Geldausgabeautomaten für diese Karten geben.

7.3.3.2 Währungsvielfalt

Für jede Währung wird es eine ausstellende Bank geben, die alleine das Recht zur Erzeugung von Geld dieser Währung hat. Sie kontrolliert die in Umlauf befindliche Geldmenge sowohl im eigenen Land als auch im Ausland und ist auch für die Zerstörung des Geldes dieser Währung zuständig.

Die ausstellende Bank bestimmt die maximale Summe in ihrer Währung, die in jeder Stufe[4] einer Mondexkarte, in jedem Land, zu jedem Zeitpunkt enthalten sein darf.

[4] Es scheint ein Stufenkonzept der Art Normalsterblicher, Händler, Großkunde, ... zu geben.

7.3.3.3 Sicherheitskonzepte

Die erste Stufe der Sicherheit besteht in der Konstruktion der Chipkarte. Jeder Versuch sie zu öffnen würde zur Zerstörung des Chips und der auf ihm gespeicherten Daten führen.

Als zweite Stufe stellen kryptographische Verfahren sicher, daß zwei autorisierte Chipkarten miteinander in Verbindung stehen und daß das Geld auch die Karte erreicht, für die es bestimmt ist. Eine Umleitung des Zahlungsstromes auf z.B. eine dritte Karte wird somit ausgeschlossen[5].

Um einem Dieb es im Verlustfall nicht zu ermöglichen, das auf der Karte gespeicherte Geld zu benutzen, kann der Besitzer der Karte sie „abschließen". Da der Dieb nicht die nötige PIN kennt um sie wieder „aufzuschließen", kann er mit der Karte nichts anfangen. Seine einzige Genugtuung besteht darin, daß auch der Besitzer das auf der Karte gespeicherte Geld verloren hat. In dieser Hinsicht verhält sich die Karte wie Bargeld.

Anders sieht es bei einem normalen Verlust der Karte aus. Sollte sie gefunden werden, kann der Finder sie nicht benutzen, falls sie gesperrt ist. Die zuständige Bank kann aber Anhand der ID der Karte den Besitzer ausfindig machen und ihm sein Geld zurückerstatten.

Um ein regelmäßiges Wechseln der Sicherheitssysteme auf der Karte zu ermöglichen, ohne alle Karten an einem bestimmten Tag aus dem Verkehr ziehen und ersetzen zu müssen, hat jede Karte zwei Systeme eingebaut, ein aktives und ein schlafendes. Um das Verfahren zu wechseln, braucht der Karte nur mitgeteilt zu werden, auf das schlafende System umzuschalten. Da nach diesem Zeitpunkt alle neu ausgestellten Karten dieses System als aktives System und ein neues schlafendes System enthalten, wird der Kunde in der Regel von dem Wechsel nichts bemerken.

Neben physikalischer Sicherheit und kryptographischen Verfahren gibt es noch einen dritten Punkt. Neben dem Versuch Betrug zu verhindern, wird es als genauso wichtig angesehen, Angriffe zu erkennen und die Schuldigen festzustellen. Terminals in Geschäften und Geldausgabeautomaten werden als günstige Stellen angegeben, Transaktionsdaten und Daten über typische Verhaltensmuster zu erheben.

Ein einfaches Beispiel ist die Tatsache, daß jede echte Transaktion eine eindeutige Nummer zugeordnet bekommt. Das doppelte Auftreten einer Nummer würde auf Betrug hindeuten. Inkonsistenzen der Aufeinanderfolgender Nummern könnten auf eine gefälschte Karte hindeuten.

[5] Dies ist der Standpunkt von Mondex, nicht der einer unabhängigen dritten Instanz.

Auch können für Transaktionen Grenzwerte gesetzt werden, deren Überschreitung eine Warnung auslöst und es der Bank erlaubt gegebenenfalls die Karte zu sperren.

Man ist der Meinung damit leicht Versuche erkennen zu können, Geld aus kriminellen Unternehmungen im Banksystem zu waschen. Einmalige Abhebungen von ungewöhnlichen Orten aus, regelmäßige Abhebungen aus bestimmten „ungeklärten" Quellen, sowie ein hoher Durchschnitt von Abhebungen nahe am Limit der Karte [6] können alle können alle den Verdacht erregen, hier gehe etwas kriminelles vor.

Natürlich sei es notwendig dabei nicht über das Ziel hinauszuschießen und auch das Recht des Individuums auf Privatsphäre zu beachten, sowie ihn darüber aufzuklären, welche Informationen auf der Karte stehen.

Als letztes sei noch erwähnt, daß jedes Gebiet in Übereinstimmung mit den lokalen Gesetzen und Sicherheitsbedürfnissen eine Struktur von Kartenklassen und dazugehörigen Kartenlimits festlegt. Dabei hat jede Karte eine festgelegte Position in einer Hierarchie, die festlegt, von welchen Karten sie Geld empfangen oder senden kann.

So ist es z.B. möglich zu verlangen, daß Karten mit einem hohen Limit bei jeder Transaktion mit einem Rechner einer Mitgliedsbank Verbindung aufnehmen müssen, wobei die Transaktionsdaten gespeichert werden. Damit wären sie zur Geldwäsche unbrauchbar.

7.3.3.4 Pilotprojekte

Eine ausführliche Übersicht findet sich im Mondex International Prospekt, der uns als Beilage zu [Mondex 96b] geschickt wurde.

Der Byte Trial

Der Byte Trial war der erste Versuch, Mondex in einem realen Umfeld einzusetzen. Die 6000 Mitarbeiter eines Rechenzentrums von NatWest in London erhielten Karten, mit denen sie in Restaurants, Cafe's und Läden des Zentrums (insgesammt 12 Stellen) bezahlen konnten. Die Karten konnten in einer NatWest Filiale im Gebäude aufgeladen werden.

Der Test war ein Erfolg. Ende 1994 waren 1 Million Verkäufe getätigt worden und die Kunden fanden das System generell recht bequem zu benutzen.

[6] zum Beispiel eine Karte einer Normalperson, über die Summen bewegt werden, die eigentlich nur auf den Karten von Ladenbesitzern auftreten

Als Folge der Kundenbefragung wurden die Möglichkeiten geschaffen, die Karte abzuschließen, sowie sie mit einem kleinen handlichen Gerät auslesen zu können.

Anscheinend läuft der Pilot immer noch. Mondex gibt an, ungefähr 4000 Karten würden zur Zeit benutzt, mit denen jährlich für etwa 1,8 Millionen Pfund eingekauft würde. Seit Beginn seien nunmehr schon mehr als 2 Millionen Einkäufe getätigt worden.

San Francisco, USA

Auch die Wells Fargo Bank testet Mondex seit August 1995. 900 Angestellte besitzen Mondexkarten, die sie in mehr als 20 Geschäften benutzen können.

Swindon

Am 3.Juli 1995 wurde Mondex in Swindon, einer Stadt mit 190.000 Einwohnern 70 Meilen außerhalb von London eingeführt.

Im November sollen sollen 700 Geschäfte und über 8000 Kunden an dem Versuch teilgenommen haben. Ob dieser Versuch als Erfolg gewertet wird ist uns nicht bekannt, da uns keine neueren Daten vorliegen. Auf den Homepages von Mondex wird nur von dem Start des Versuches berichtet, der bis Januar '96 dauern sollte und von dem erwartet wurde, daß 40.000 Kunden und und 1.000 Geschäfte daran teinehmen würden. Obige Zahlen stammen aus einer Pressemitteilung von Bell Canada vom 1. November 1995, der von dem Start eines ähnlichen Versuches in Guelph in Kanada berichtet.

Universitätskarten

Wie in [Lockstone 96] angekündigt, wurden laut [Mondex 96b] an der Universität von Exeter in Devon, England, 12000 Karten an Beschäftigte und Studenten ausgegeben.

Außer zum Bezahlen dient die Karte auch als Studenten- und Bibliotheksausweis, zur Zugangskontrolle zu Gebäuden[7] und als Wahlkarte für studentische Wahlen.

Ein ähnlicher Versuch läuft in York, wo im Oktober 1996 1600 Karten an Erstsemesterstudenten ausgegeben wurden.

[7]Zumindest der Zugang zu den Geldautomaten wird aber noch über einen zusätzlichen Magnetstreifen auf der Rückseite der Karte geregelt

Hong Kong

In Hong Kong begannen die HongkongBank und die Hang Seng Bank einen Pilotversuch in zwei belebten Geschäfts- und Einkaufszentren, TaiKooShing auf der Insel Hong Kong und Sha Tin in den New Territories.

Nach einer Testphase, die im September 1996 endete, wird den Kunden nun für eine gewisse Zeit ein Grundpaket angeboten, das kostenlos eine Karte und einen Saldenleser beinhaltet. Auch die „Brieftasche“ wird verbilligt abgegeben.

Für Ende 1997 ist die Einführung in ganz Hong Kong geplant.

Guelph, Canada

In Guelph in Ontario war ein Versuch für Anfang 1997 geplant. Der Pilot wird von der Canada Imperial Bank of Commerce und der Royal Bank of Canada durchgeführt. Es ist geplant, daß bis zur Mitte November 1997 50 Händler Mondexkarten akzeptieren werden mit einem anvisierten Ziel von insgesamt über 350 Händlern.

7.3.3.5 Eine Beschwerde

Der folgende unter [PI 95] nachzulesende Vorgang brachte einige technische Details von Mondex zum Vorschein, die in allen anderen Quellen – wenn überhaupt – nur sehr vage dargestellt werden.

Am 16.September 1995 reichte Simon Davies, Generaldirektor von Privacy International[8], Beschwerde wegen der Aussage von Mondex International auf ihrer Homepage, im täglichen Gebrauch seien Mondex Transaktionen *anonym*[9], genau wie Bargeld.

In der Antwort ließ die zuständige Behörde für Umweltschutz und Handelsstandarts[10] wissen, daß Mondex inzwischen das Wort „anonym“ durch „privat“ ersetzt hat, welches dem Zweck angemessener sei. Da die Aussage nur in einer Pressemitteilung gemacht worden sei, nicht aber nicht in den Unterlagen, die in Banken und Geschäften auslagen, sei ein Verstoß gegen das Gesetz über Handelsbeschreibungen nicht nachzuweisen.

[8]Ein Zusammenschluß mehrerer Menschenrechtsorganisationen, darunter The Stern Foundation (Washington D.C.), The German Marshall Fund (Washington D.C.), Computer Professionals for Social Responsibility (CPSR, Washington D.C.), The University of New South Wales (Sydney). Insgesammt sollen Experten und Organisationen aus vierzig Ländern vertreten sein.

[9]Hervorhebung durch die Autoren

[10]Es war eine Beschwerde wegen irreführender Werbung

Inhaltlich wird in dem Schreiben zugegeben, daß

- auf der Karte die letzten 10 Transaktionen mit Betrag, Händler und Datum gespeichert werden und im Falle eines Verlustes einer nicht abgeschlossenen Karte ausgelesen werden können.
- die Kasse des Händlers die letzten 300 Transaktionen mit Kartennummer, Betrag und Datum speichert. Sie können von der Bank gelesen und für Marketingzwecke benutzt werden. Der Händler kann die Kartennummer nicht dem Kunden zuordnen, da die nötigen Informationen dem Bankgeheimnis unterliegen.
- diese Daten prinzipiell zu Marketing Zwecken an Dritte verkauft werden könnten. Ihm[11] sei aber versichert worden, daß dies nicht geschehe und der Datenschutz beachtet werde.

7.4 Digicash

7.4.1 Motivation

Auch in den Niederlanden machte man sich seit Mitte der achtziger Jahre am CWI (ein niederländisches Zentrum für Forschung in Mathematik und Informatik) Gedanken über elektronische Zahlungsmittel. Anders als bei anderen Firmen sah man aber nicht nur die Probleme in Bezug auf die Fälschungssicherheit der elektronischen Zahlungsmittel, sondern hatte auch die Probleme im Auge, die daraus entstehen, wenn jede Transaktion elektronische Spuren hinterläßt.

Einer der führenden Köpfe war David Chaum, der in mehreren Artikeln [Chaum 85] [Chaum 87] [Chaum 92] vorschlug, einen neuen Weg zu gehen und das Recht des Individuums auf seine Privatsphäre in den Vordergrund zu stellen.

Er entwickelte eine neue Methode, mit der es möglich ist, elektronisches Geld von einer Bank abzuheben und bei einem Händler auszugeben, ohne daß Bank und Händler, selbst wenn sie zusammen arbeiten dem Kunden den Kauf zuordnen können.

Um dies auch praktisch umsetzen zu können, gründete er die Firma DigiCash, die das von ihm entwickelte System nun unter dem Namen „ecash" vermarktet. Inzwischen haben sich vier Banken gefunden, die elektronisches

[11] Robert Gilham, der zuständige Beamte für Handelsstandarts

Geld nach diesem System herausgeben. Auch gibt es schon ausreichend viele Geschäfte, die DigiCash akzeptieren (allein die Auflistung der jenigen Internet Shops, die die Testwährung des Cyberbuck Trials akzeptieren umfaßt 91 Händler in 10 Kategorien).

Ein einziger Schönheitsfehler ist vielleicht noch, daß zwischen den Währungen der verschiedenen Banken nicht elektronisch konvertiert werden kann. Ein Händler, der EUnet ecash akzeptiert, wird daher in der Regel kein Mark Twain ecash nehmen.

7.4.2 Grundidee

Die Idee ist recht einfach, wenn man akzeptiert, daß es elektronische Verfahren gibt, die es erlauben, beliebige elektronische Daten so zu markieren, daß eindeutig ist, wer dies getan hat, und die sicherstellen, daß diese Daten nicht verändert werden können ohne daß dies bemerkt würde. Diesen Vorgang nennt man eine elektronische Unterschrift.

Die einfachste Form elektronischen Geldes wäre das digitale Gegenstück eines Blattes Papier mit der Aufschrift

> *Dieses Papier ist 10 DM wert,*
> *Ihre Bank*

Eine derartige Banknote kann nur von der Bank erzeugt werden und auch der Wertaufdruck ist unveränderbar.

Leider ist es extrem einfach, eine derartige Banknote zu kopieren. In der Welt der Computer handelt es sich ja nur um eine Folge von Einsen und Nullen. Ein Fälscher weiß nicht, wie er eine solche Folge erzeugen kann, aber sie zu kopieren ist einfacher, als einen Geldschein auf einen Fotokopierer zu legen. In der physikalischen Welt erkennt man einen falschen Geldschein vielleicht noch an der Papierqualität oder den schlechten Farben. In der digitalen Welt dagegen besteht alles aus Zahlen. Diese mehrfach an verschiedene Empfänger zu senden, ist technisch weder schwierig noch zeitaufwendig.

Aber es gibt eine einfache Lösung für dieses Problem. Der Käufer schreibt auf ein Blatt Papier eine einmalige Seriennummer. Die Bank braucht dann nur noch mit ihrer digitalen Unterschrift den Wert des neuen Geldscheines zu bestätigen.

Gibt nun der Kunde seinen so erzeugten Geldschein einem Händler, der ihn sofort an die Bank weiterleitet. Wurde noch kein Schein mit dieser Seriennummer vorgelegt, so bekommt der Händler sein Geld und der Kunde seine Ware. Gab es die Nummer dagegen schon einmal, so handelt es sich bei dem Schein um eine wertlose Kopie.

Leider bietet das beschriebene Verfahren noch keine Anonymität für den Kunden. Die Bank weiß, welche Seriennummer von welchem Kunden benutzt wurde und kann so Kunden und Händler, sowie den Betrag der Transaktion in Verbindung bringen. Kooperiert sie sogar mit dem Händler, liegen alle Daten beiden offen, was eine lukrative Erstellung von Kundenprofilen erlaubt.

Um dies zu verhindern schickt der Kunde seine vorbereitete Banknote in einem Briefumschlag an die Bank. Der Umschlag verdeckt die Seriennummer, hat aber ein Fenster, durch den die Bank ihre Unterschrift aufbringen und das Geld somit gültig machen kann. Nach dem Entfernen des Umschlages besitzt der Kunde einen Geldschein, der alle Eigenschaften des vorigen Modells besitzt, der aber zusätzlich nicht zu ihm zurückverfolgt werden kann.

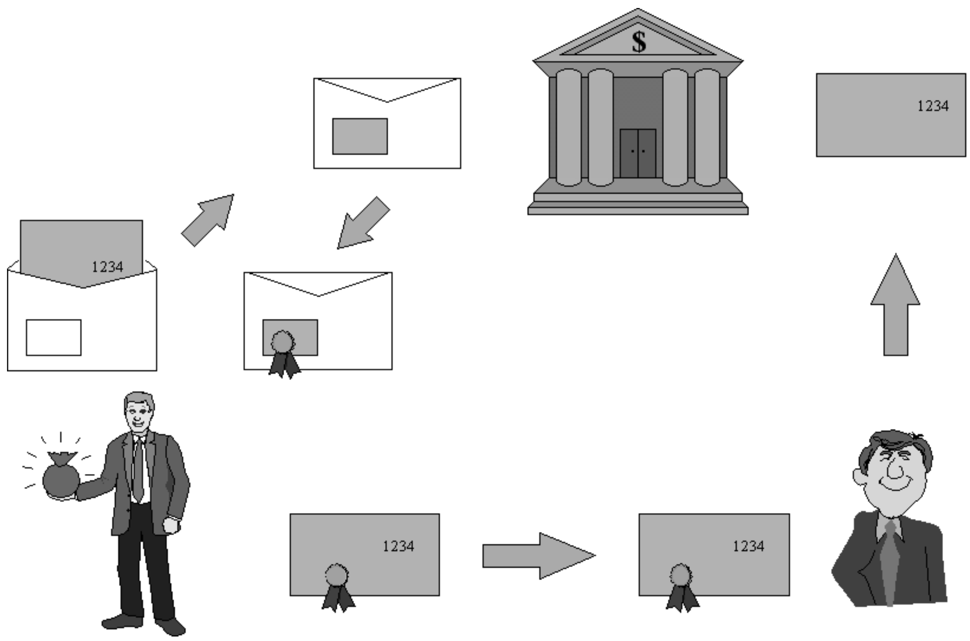

Abbildung 7.4: Ecash

7.4.3 Realisierung

7.4.3.1 Mathematische Grundlagen

Es ist das Verdienst von David Chaum, das Verfahren der blinden Signaturen erfunden zu haben. Um es zu verstehen, ist es aber leider nötig, die Theorie der digitalen Signaturen zu bemühen.

Kennt man $n = pq$, wobei p und q Primzahlen sind, so ist es einfach,

$$y = x^s \bmod n$$

zu berechnen. Es ist aber extrem schwierig, ohne Kenntnis der Faktoren p und q die Funktion

$$z = \sqrt[s]{x} \bmod n$$

zu berechnen. In der Tat kann diese Aufgabe bei ausreichend großem n als praktisch undurchführbar angesehen werden.

Daher kann die Bank z berechnen (sie kennt ja p und q), einem Außenstehenden ist dies aber nicht möglich (er hat ja nur n und s). Es ist ihm aber leicht möglich ein Paar (x_1, z) zu prüfen. Er braucht nur

$$x_2 = z^s \bmod n$$

zu berechnen. Ist $x_1 = x_2$, so ist das Paar gültig.

Ein kleines Problem besteht noch in der Tatsache, daß $(x^s \bmod n, x)$ ebenfalls ein gültiges Paar wäre. Dies kann man aber lösen, indem man

$$z_2 = \sqrt[s]{f(x)} \bmod n$$

verwendet. $f()$ ist hierbei eine Einwegfunktion, die zwar einfach zu berechnen, aber deren Umkehrfunktion $f^{-1}()$ nicht mit vertretbarem Aufwand berechenbar ist.

Kommen wir aber nun zu den blinden Signaturen. Um nun sein $f(x)$ in einem „Umschlag“ zu verpacken, wählt der Kunde eine Zufallszahl r. Es fällt ihm auch nicht schwer,

$$b_{blinded} = f(x) * r^s \bmod n$$

zu berechnen. Die Bank erzeugt nun

$$\begin{aligned} z_{blinded} &= \sqrt[s]{b_{blinded}} \bmod n \\ &= \sqrt[s]{f(x) * r^s} \bmod n \\ &= (\sqrt[s]{f(x)}) * r \bmod n \\ &= z * r \bmod n \end{aligned} \tag{7.1}$$

aus dem der Kunde nun nur noch r herausdividieren muß um z zu erhalten. Das Paar (x, z) dient ihm dann als „digitaler Geldschein“, wobei verschiedene s für verschiedene Beträge verwendet werden können.

Die Bank kann keine Verbindung zwischen einem blind signierten Schein und einem bei dem der „Umschlag“ entfernt wurde, herstellen, solange ihr r unbekannt ist. Sollte der Kunde aber jemals in die Verlegenheit kommen, beweisen zu müssen, daß er eine Zahlung getätigt hat, kann er r bekannt geben und so den Zusammenhang herstellen zwischen einem $b_{blinded}$, das er der Bank geschickt hat, und einem (x, z), das ein Händler eingelöst hat. Damit ist r gewissermaßen seine „Quittung“.

(Nach[Finney 93a])

Dieses r ist auch das Gegenargument von DigiCash gegen den Vorwurf, die Anonymität ihres Geldes könne von Steuerflüchtlingen und Drogenhändlern ausgenutzt werden. Die Firma weist auf ihren Seiten[Digicash 96] darauf hin, daß zwar der Käufer anonym sei, der Verkäufer aber nicht. Dies bedeutet, daß ein Drogenhändler jederzeit damit rechnen muß, das ein unzufriedener Kunde mit seinen Unterlagen zur Polizei geht und der damit hieb- und stichfeste Beweise liefert. Außerdem könne man nicht vor der Bank verbergen, daß man Geld erhalte, was auch Steuervergehen leicht feststellbar mache.

7.4.3.2 Der Schutz der übertragenen Nachrichten

Die folgenden Informationen stammen aus der Protokollbeschreibung, die - leider noch unvollständig - unter [Digicash 96] zu finden ist.

Nachrichten zur Bank können grundsätzlich mit deren Public Key verschlüsselt werden. Die momentane Software verschlüsselt nur die Anforderung ein Konto zu eröffnen[12] und Abhebungen. Abhebungen müssen darüber hinaus von ihm signiert werden.

Als konventionelles Verfahren, dessen Schlüssel dann mit RSA kodiert wird, dient zur Zeit Triple DES. Hashwerte für Signaturen werden generiert, indem zuerst die Nachricht mit SHA gehasht wird. An die resultierenden 20 Byte wird eine 1-Byte-Zufallszahl *hashID* angehängt. Das Ergebnis wird solange mit der *hashID* konkateniert und gehasht, bis das Ergebnis so lang

[12] Dabei ist nicht ganz klar, ob es um das Eröffnen eines Kontos geht oder um das Öffnen (wie beim Öffnen einer Datei, bevor man diese lesen kann).

ist wie der RSA Modulus, z.B. für einen Modul von 41...60 Bytes:

$$
\begin{aligned}
&(SHA(m).hashID.\\
&\quad SHA(SHA(m).hashID).\\
&\quad\quad SHA(SHA(SHA(m).hashID).hashID)) \bmod n
\end{aligned}
$$

('.'bedeutet hier Konkatenation, n ist der RSA modulus)

Nachrichten von der Bank sind grundsätzlich RSA verschlüsselt und enthalten eine Folgenummer, an der der Kunde erkennen kann, ob er alle Nachrichten erhalten hat.

Um auch Zahlungen gegen Veränderungen zu schützen, generiert der Kunde einen Hash der Zahlungsinformationen:

- Bank ID
- Protokoll
- Betrag
- Währung
- Anzahl der Münzen
- Zeitpunkt der Zahlung
- Gültigkeit
- Bank des Händlers
- Konto des Händlers
- Hashwert einer geheimen Zufallszahl (Relikt des Offline - Protokolls)
- Hashwert der Beschreibung der Zahlung

Dieser Hashwert wird selber nicht übertragen, aber mit der Signatur jeder Münze xor'ed, worauf der entstehende Wert noch mit dem öffentlichen Schlüssel der Bank chiffriert wird. Die Bank wird kann daher nur Münzen annehmen, wenn die Zahlungsinformationen nicht nachträglich manipuliert wurden.

7.4.3.3 Die Software

Zur Benutzung von ecash gibt es eine Software, die auf den Seiten der Mark Twain Bank [MarkTwain 96] ausführlich beschrieben wird.

Zu Anfang wird nur ein kleines Fenster angezeigt, daß den momentanen Betrag digitalen Geldes auf der Festplatte und vier Knöpfe präsentiert. drückt man die entsprechenden Knöpfe, so erscheinen kurze Formulare, in denen man die nötigen Angaben macht, um Geld abzuheben oder einzuzahlen. Die oben erklärten komplizierten mathematischen Vorgänge bleiben dabei selbstverständlich transparent. Der Benutzer setzt den Betrag ein und die Software macht den Rest.

Sendet nun ein Händler eine Zahlungsanforderung an den Kunden, so öffnet sich bei diesem ein Fenster, welches ihm erlaubt per Knopfdruck zu bezahlen oder abzulehnen. Ein weiterer Knopf öffnet ein Fenster, daß es ihm erlaubt festzulegen, daß bei

- gleicher Händler ID
- gleichem Betrag
- gleichem Zweck

automatisch bezahlt werden soll, wobei Gesamtbetrag, der Betrag einer einzelnen Zahlung und die Anzahl der Zahlungen begrenzt werden können.

Die Software bietet auch die Möglichkeit selber Zahlungen zu initieren oder zu verlangen. Dazu gibt es ein Fenster, in dem die eigene Konten ID, der Betrag und eine Beschreibung des Zwecks angegeben werden.

Die Konten ID wird bei dem Verschicken von Geld hoffentlich nicht mit übertragen, das Fenster zur Annahme einer Zahlung weist diese Angabe jedenfalls nicht aus.[13]

Als letztes führt die Software noch Buch über alle stattgefundenen Transaktionen. In einem weiteren Fenster können diese Angaben in Augenschein genommen werden. Wird es nötig die Münzen wegen eines Plattencrashes neu zu generieren, so kann dies von der Software übernommen werden. Dazu ist nur eine Zahl nötig, die von der Software bei der Installation ausgegeben

[13] In der (bisher unvollständigen) Protokollbeschreibung werden bei einer Zahlung nur Name und Kontonummer des Händlers übertragen. Alles andere würde dem Sinn des Verfahrens direkt zuwiderlaufen. Das Fenster wird aber sowohl für Sendung als auch Empfang des Geldes verwendet.

wurde und die als Startwert für den Zufallszahlengenerator dient. [14]

Auch die nötigen Angaben, um im Bedarfsfall den Empfänger einer Zahlung preiszugeben, werden vom Programm automatisch verwaltet.

7.4.3.4 Banken

Ecash wird mittlerweile von vier Organisationen weltweit zum Einsatz gebracht. Auch wenn die Frage der Konvertierbarkeit zwischen den digitalen Währungen noch nicht geklärt ist, so scheint es doch als ob ecash eine Chance habe, sich einen gewissen Marktanteil am zukünftigen digitalen Geldmarkt zu erobern.

Einen Überblick über die partizipierenden Banken und Links zu deren Webseiten findet man auf den Seiten von Digicash [Digicash 96].

Mark Twain Bank (USA)

Die Mark Twain Bank in St. Louis Missouri war die erste, die Digicash einführte. Beeindruckt von dem CyberBuck Experiment wo im ganzen Internet 60.000 Menschen ein Jahr lang kostenlos eine fiktive digitale Währung auf ecash Basis benutzen durften, führte sie am 23. Oktober 1996 ecash auf Dollar Basis ein.

Die nötige Software wird kostenlos zur Verfügung gestellt, die Einrichtung eines ecash accounts kostet 11 - 25$[15]. Daneben wird eine monatliche Grundgebühr von 1 - 5$ fällig, die 1 - 5 monatliche Transaktionen zum ecash System erlaubt. Die monatliche Gebühr wird nicht fällig, wenn mindestens 500 - 1.500$ auf dem zugehörigen Bankkonto sind. Es kostet 3 - 1$[16] Geld in ecash umzuwandeln. Will man sein Geld von ecash wieder zurück auf das Bankkonto transferieren, so werden 5 - 4 % Gebühren fällig.

EUnet (Finnland)

Am 13. März 1996 gesellte sich EUnet, ein führender Internet Provider, als zweiter Anbieter von ecash dazu. Zusammen mit Merita, Finnlands größter Bank, erlauben sie es nun im Netz in Finnmark einzukaufen.

[14] Es gibt im Protokoll die Möglichkeit die Bank um die Nachrichten der letzten n (im Moment 16) Abhebungen zu bitten. Die Clientensoftware wird daraus die Münzen berechnen und alle Münzen auf der Platte sofort auf der Bank deponieren. Die Münzen, die noch nicht von Händlern eingelöst wurden, werden dem Konto gutgeschrieben.

[15] Für Privatkunden gibt es drei verschiedene Schemata: „Ich probier das mal“, „Häufiger Benutzer“ und „Der extrem häufige Benutzer“

[16] Je höher die Grundgebühr, umso geringer die Transaktionskosten

Leider war ansonsten keine Information erhältlich[17]

Deutsche Bank (Deutschland)

Die Deutsche Bank wird zum Ende des Jahres mit einem sechsmonatigen Pilotversuch beginnen. Teilnehmen können nur Kunden der Deutschen Bank, Tochtergesellschaften wie die Bank 24 werden nicht teilnehmen.

Die Deutsche Bank führt als Vorteile des Verfahrens auf

1. Die Finalität eines Kaufes. Nach einer Online Prüfung wird die Einlösung der digitalen Münzen durch die Bank garantiert. Für diese Prüfung entstehen keine Kosten, so daß sich der Einsatz von ecash auch bei kleinen Beträgen lohnt.

2. Die Anonymität der Zahlungen (s.o.)

3. Die Möglichkeit des Geldtransfers zwischen Privatpersonen. (Im Gegensatz zu Kreditkarten)

Während des Pilotversuches wird die Deutsche Bank keinen Teilnehmerpreis für die Benutzung von ecash erheben.

Der maximale Betrag, den man auf seinen Rechner transferieren oder dort lagern kann ist auf 400DM beschränkt.

Advanced Bank (Australien)

Die Advanced Bank kündigte am 24.Oktober 1996 an, sie wolle ecash in australischen Dollars herausgeben. Die Bank wolle damit ihre Position als führende Internet Bank Australiens weiter ausbauen. Schon heute könnten Benutzer per Internet Banking Kontoauszüge ansehen, Überweisungen tätigen und Rechnungen bezahlen. (Pressemitteilung Digicash, 24.Oktober 1996)

7.4.4 Erweiterungsmöglichkeiten

Der größte Nachteil dieses Verfahrens ist die Tatsache, daß es, um Sicherheit zu gewährleisten, online ablaufen muß. Um diesem Mangel abzuhelfen entwickelte Chaum eine offline Variante, die im DigiCash Protokoll auch unterstützt wird, die aber über einige Testversuche bisher nicht hinausgekommen ist. In naher Zukunft wird sie wohl nicht praktisch eingesetzt. Ich

[17] Zumindest nicht in Englisch, des Finnischen sind wir leider beide nicht mächtig.

werde daher hier nicht auf die mathematischen Details eingehen, sie können aber in [Finney 93a] nachgelesen werden.

Die eigentliche Idee ist aber schnell erklärt. Der Kunde schickt diesmal nicht einen sondern k ($k \geq 2$) Geldscheine zur Bank zum blinden Signieren. Auf diesen steht nun nicht eine Zufallszahl, sondern sein Name verschlüsselt mit einem Schlüssel a_i ($1 \leq i \leq k$). Dabei sind die a_i alle verschieden. Die Bank signiert die Hälfte der Geldscheine, wobei diese Auswahl nach dem Zufallsprinzip erfolgt.

Die andere Hälfte der Geldscheine wird nicht signiert. Stattdessen fragt die Bank nach den „Umschlägen" r_i und den Schlüsseln a_i. Sie stellt dann sicher, daß in den von ihr geöffneten Umschlägen auch tatsächlich der Name des Käufers steht. Die ungeöffneten Umschläge gehen zurück an den Kunden, dem die darin enthaltenen Scheine als eine Zahlungseinheit dienen.

Will er nun bezahlen, sendet ihm der Händler eine Folge von $\frac{k}{2}$ Nullen und Einsen, die er zufällig gewählt hat. Für jede Null sendet der Kunde einen Schein mit seinem verschlüsselten Namen, für jede Eins ein a_i[18].

Da er jedesmal entweder seinen verschlüsselten Namen oder einen Schlüssel gesendet hat, liegt dem Händler kein Paar von Name und Schlüssel vor, mit dem er die Identität des Kunden herausfinden könnte. Sendet der Kunde aber die selbe Folge von Scheinen an einen anderen Händler, so besteht eine gute Chance, daß dieser einen Schlüssel verlangt, wo sein Vorgänger nach einem Namen gefragt hatte. In diesem Fall kann die Bank die Informationen der Händler zusammenfügen und den Schuldigen herausfinden.

Der Faktor k bestimmt dabei, wie groß diese Chance tatsächlich ist. Bei seiner Wahl gilt es Sicherheit und Schnelligkeit der nötigen Berechnungen gegeneinander abzuwiegen.

7.5 CAFE Wallet

7.5.1 Motivation

CAFE
(Conditional Access for Europe) [CAFE 96][Div 94][Waidner 94][MjMi 97] ist ein Projekt der Europäischen Gemeinschaft im Rahmen ihres ESPRIT - Programmes. Die Arbeiten wurden im Dezember 1992 begonnen und hatten

[18] Plus zusätzliche Informationen, anhand derer der Händler verifizieren kann, daß er auch tatsächlich das bekommen hat, wonach er verlangt hatte.

zum Ziel, ein Offline-Debitsystem zu entwickeln, das

- verschiedene Währungen unterstützt,
- es erlaubt, dem Benutzer sein Geld zurückzuerstatten, wenn er seine elektronische Geldbörse
  - verliert,
  - beschädigt, oder
  - sie ihm gestohlen wird,
- offen konstruiert ist, d.h. verschiedene Banken und Plattformen zuläßt,
- kein Vertrauen zwischen den beteiligten Parteien voraussetzt (Dies schließt die Bank mit ein).

Laut [SchuWe] war für Anfang Juli 1995 ein Feldversuch bei der Europäischen Union in Brüssel geplant.

Das Projekt endete am 29. Februar 1996. Die Ergebnisse des Tests wurden der Europäischen Kommisssion vorgelegt und werden wohl in absehbarer Zeit veröffentlicht.

Als Nachfolger wurde das Projekt Opera ins Leben gerufen, das die Erprobung des CAFE - Systems sowohl auf dem Gelände der Europäischen Kommission als auch international auf den Geländen der Sponsoren fortsetzt. Opera wird primär von den Banken unterstützt, die schon an CAFE teilgenommen hatten.

Das Sekretariat von OPERA kann unter folgender Adresse erreicht werden:

CardWare Ltd.
19 Roundwood Lane
Harpenden Herts AL5 3BW
UK
Tel: +44 1582 760664
Fax: +44 1582 764518

7.5.2 Grundidee

Wie das Offline - Verfahren von Chaum (s. 7.4.4) benutzt auch CAFE ein Protokoll, das anonym ist, solange der Kunde nicht versucht, dieselbe Münze mehrfach auszugeben. Betrügt er dagegen, so ist es der Bank möglich, im Nachhinein festzustellen, wer der Täter war.

Diese Eigenschaft allein reicht aber nicht aus, da der Schuldige zu diesem Zeitpunkt schon außerhalb der Reichweite der Strafverfolgungsbehörden sein könnte. Aus diesem Grunde wurde ein „Beobachter“ in das System eingebaut. Dabei handelt es sich um eine manipulationsgesicherte Chipkarte, die verhindert, daß Zahlungen stattfinden, ohne daß vorher entsprechend hohe Summen digitalen Geldes erworben wurden.

Dies schwächt aber nicht die Eigenschaften des Zahlungsprotokolls. Sollte der Beobachter kompromittiert worden, und die in ihm gespeicherte geheime Information bekannt geworden sein, so kann zwar dasselbe Geld mehrfach ausgegeben werden, aber der Täter ist weiterhin im Nachinein feststellbar. Anders herum gilt aber auch, daß die Sicherheit hier nicht auf Kosten der Anonymität des Kunden geht. Solange dieser nicht dieselben elektronischen Münzen mehrfach verwendet, kann seine Identität nicht ermittelt werden.

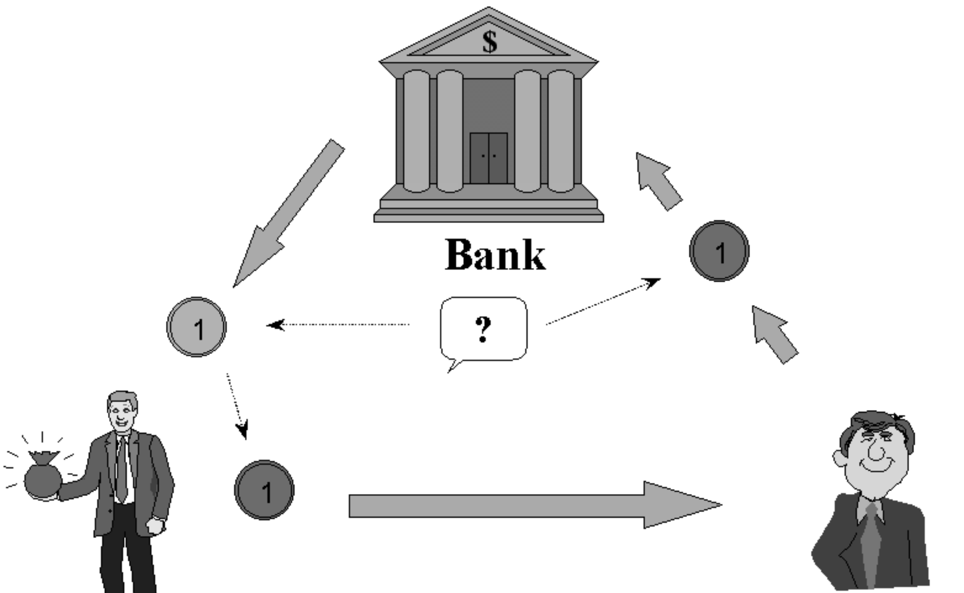

Abbildung 7.5: CAFE: normale Zahlung

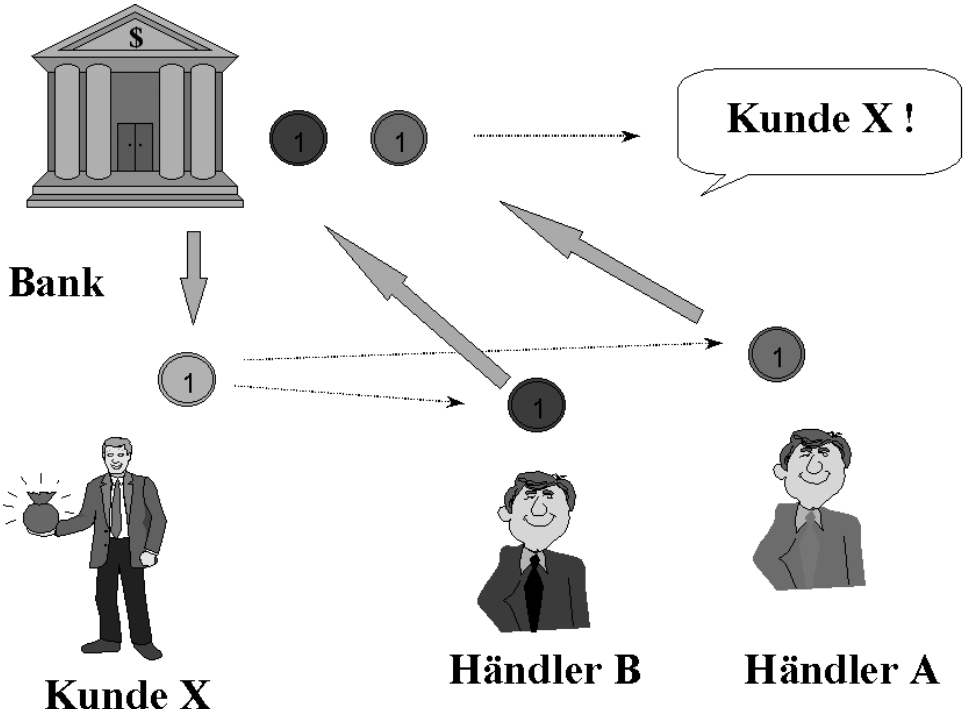

Abbildung 7.6: CAFE: versuchter Betrug

7.5.3 Realisierung

Bei der Realisierung obiger Idee drängt es sich natürlich auf, dem Beobachter als „großem Bruder in der eigenen Brieftasche“ zu mißtrauen. Um solche Probleme, wie sie z.B. in [Finney 93b] formuliert wurden, von vorneherein aus der Welt zu schaffen, wurde das System so entwickelt, daß der Beobachter niemals direkt mit der Bank kommuniziert. Jegliche Kommunikation mit der Außenwelt findet über eine elektronische Geldbörse als vertrauenswürdiges Interface des Kunden statt.

Der Kunde vertraut seiner elektronischen Gelbörse, die für ihn mit der Außenwelt kommuniziert und alle wichtigen Berechnungen durchführt. Dazu braucht sie zwar für jede Transaktion eine bestimmte Zahl, die sie jeweils vom Beobachter erhält, sie selber steht aber nicht zwangsläufig unter der Kontrolle der Bank. Vielmehr ist es ihre einzige Aufgabe die Interessen des Kunden zu vertreten und verdeckte Kanäle zwischen Beobachter und Bank zu unterbinden.

Wie es dabei möglich ist, sicherzustellen, daß der Beobachter in das Protokoll eingeschaltet ist, er aber trotzdem der Bank die Identität des Kunden nicht melden kann, ist unter [Brands 93] nachzulesen. Die dortigen Ausführungen sind zwar zu kompliziert, um hier wiederholt zu werden, es reicht aber sich vorzustellen, daß ein Protokoll ähnlich dem vom Chaum verwendet wird, bei dem der Beobachter gewissermaßen jede Meldung der Geldbörse signiert, ohne dabei mitzuteilen, welcher spezielle Observer er ist, oder zu welchem Kunden er gehört.

Die Implementation realisiert die Trennung von Beobachter und Geldbörse auch physisch. Mittlerweile existiert eine Realisierung der Gelbörse in Form eines Gerätes, das wie ein kleine Taschenrechner aussieht, mittels Infrarot zur Datenübertragung zu seinen Partnern im Zahlungsverkehr fähig ist, und das durch eigene Tasten[19] effektiv die Gefahren durch manipulierte Terminals vermindert. Der Beobachter hingegen ist in einer Chipkarte realisiert, die in die eigentliche elektronische Geldbörse eingeschoben wird.

Obwohl es sich um ein vorausbezahltes System handelt, bei dem zuerst Geld in die elektronische Brieftasche geladen wird, das dann Stück für Stück ausgegeben wird, wollte man eine Möglichkeit schaffen, dem Kunden dieses Geld zurückzuerstatten, wenn besagte Brieftasche beschädigt, verloren oder gestohlen wird. Dazu ist es nötig, daß der Kunde ein Backup seines Brieftascheninhaltes auf eine spezielle Chipkarte macht. Im Verlustfalle erlauben

[19] wahlweise je eine „Ja“- und eine „Nein“-Taste oder ein numerisches Zahlenfeld

es die Daten auf der Karte der Bank, die Münzen zu rekonstruieren, die sich zu diesem Zeitpunkt auf der Karte befanden. Ohne besagtes Backup wäre dies nicht möglich, da in der Brieftasche Berechnungen stattfinden, die die Bank nicht nachvollziehen kann[20]. Die rekonstruierten Münzen können nun mit einer Liste der bei der Bank eingelösten Münzen verglichen werden, woraufhin dem Kunden die Differenz ersetzt wird.

Allerdings ist es beim Abhandenkommen der Brieftasche nötig zu warten, bis die von ihr verwendeten Schlüssel ungültig werden. Ansonsten könnte auch ein unehrlicher Besitzer eine Wiederherstellung der Münzen in die Wege leiten, um dann doch noch die „wiedergefundene“ Brieftasche zum Bezahlen zu verwenden. Um die Wahrscheinlichkeit zu erhöhen, daß zum Zeitpunkt der Wiederherstellung nicht schon das ganze Geld durch eine dritte Person ausgegeben wurde, welche die Brieftasche gefunden oder gestohlen hat, ist eine optionale PIN für das Bezahlen vorgesehen[21]. Ebenfalls ist es möglich, einen bestimmten Betrag im Voraus für eine erwartete Zahlung freizugeben.

CAFE ist in vielerlei Hinsicht als offenes System ausgelegt, dessen hierarchisches Verechnungssystem es erlaubt, jederzeit neue Teilnehmer aufzunehmen. Transaktionen zwischen Kunden verschiedener Banken sind genauso vorgesehen wie die Verwendung unterschiedlicher Währungen.

[20]Dies ist die Grundvoraussetzung für die zugesicherte Anonymität.

[21]Abhebungen erfordern in jedem Fall eine PIN.

Kapitel 8

Bewertung

Um die unterschiedlichen Verfahren besser bewerten zu können, weicht die Einteilung der Verfahren gegenüber den Kategorien bei der Beschreibung der einzelnen Verfahren geringfügig ab.

Um das Millicent-Verfaren, daß bei der Beschreibung das einzige in seiner Gruppe (Digitale Koupons) war, mit anderen Verfahren vergleichen zu können, fällt es in diesem Abschnitt in die neue Kategorie "Verfahren für kleine Beträge", zusammen mit den Verfahren MPTP und CyberCoin.

Die Gruppe "Digitale Koupons" entfällt daher ganz, die restlichen drei Gruppen "Digitale Kreditkarten", "Digitale Schecks" und "Digitales Bargeld" entsprechen wie gewohnt den Gruppen, in die die Verfahren auch bei ihrer Vorstellung eingeteilt wurden.

Die Verfahren MPTP und CyberCoin im Zusammenhang mit den Verfahren ihrer ursprünglichen Gruppe zu betrachten, erschien uns aufgrund der Verwendung dieser Verfahren für wesentlich höhere Beträge (und deshalb auch größerer Sicherheitsanforderungen) weniger sinnvoll.

8.1 Verfahren für kleine Beträge

8.1.1 Gewichtung

Die in diesem Abschnitt betrachteten Verfahren sind für die Bezahlung geringwertiger Beträge konzipiert. Aus diesem Grund sind die Sicherheitsanforderungen an diese Verfahren nicht so hoch wie bei den Verfahren zur Bezahlung höherer Summen. Man geht davon aus, daß man die hier erwähnten Verfahren wie sein Kleingeld in der Geldbörse handhabt, d.h. man sieht es als nicht so tragisch an, wenn ab und zu mal der eine oder andere Groschen verlorengeht.

Der Wahrung der Privatsphäre gebührt bei der Betrachtung dieser Verfahren besondere Aufmerksamkeit, da ein Grund für die Einführung von Verfahren zum Bezahlen geringer Beträge war, daß man so dem Kunden ermöglicht, nicht eine ganze Zeitung, sondern nur einzelne Artikel zu erwerben. Das Lesen eines einzelnen Artikels sagt aber unter Umständen wesentlich mehr über einen Menschen aus. Als Vergleich nehme man eine Person, die ein Herrenmagazin abonniert hat, tatsächlich aber hauptsächlich nur die Interviews liest, oder jemanden, der sich von seiner Tageszeitung immer nur "das Mädchen von Seite 3" anschaut.

8.1.2 Vertraulichkeit

8.1.2.1 Vertraulichkeit des Transaktionsinhaltes und der Kundenidentität

Bei Millicent ist der Kunde gegenüber dem Makler, der im Millicent-Modell im Prinzip die Bank darstellt, nicht anonym. Aufgrund der Tatsache, daß sich die ID des Kunden im eigentlichen Geldstück (Scrip) befindet, ist der Kunde auch gegenüber dem Händler nicht anonym, der Händler kann sogar zusätzliche Informationen über den Kunden (wie z.B. Alter und Wohnort[1]) aus dem im Scipt enthaltenen "`props`-Feld" erlangen. Es wird von Millicent empfohlen, zur Wahrung der Privatsphäre als Kunden-ID ein Pseudonym zu verwenden, doch auch diese Maßnahme kann ein Auswerten der Transaktionen seitens des Händlers nicht verhindern.

Der Transaktionsinhalt ist dem Händler natürlich bekannt, aber auch der Makler hat eine ungefähre Vorstellung von Betragshöhe und Datum, da der Kunde schließlich bei ihm für die geplante Transaktion eine entsprechende Menge an Geld eintauscht. Die Identität des Handlers kennt der Makler aufgrund der Händlerbezogenheit des Geldes natürlich auch.

Inwieweit ein Angreifer Kundenidentität und Transaktionsinhalt erfährt, hängt ganz von dem verwendeten Protokoll aus der Millicent-Familie ab. Im Gegensatz zum "Scrip in the clear"-Protokoll, bei dem die Daten unverschlüsselt übertragen werden, bietet z.B. das "Private and secure"-Protokoll Schutzmaßnahmen durch die Verwendung von symmetrischen Verschlüsselungsverfahren, die ein Angreifer zunächst einmal brechen müßte, um an

[1] Diese Daten werden z.B. aus Gründen gesetzlicher Altersbeschränkungen für bestimmte Artikel und unterschiedlicher gebietsspezifischer Verkaufssteuersätze erhoben.

die gewünschten Informationen zu gelangen. Aus Effizienzgründen wird jedoch anstatt dieses rechenintensiveren Protokolls meist die Verwendung des "Secure without encryption"-Protokolls vorgeschlagen. Dieses Protokoll verzichtet auf die Verwendung eines verschlüsselten Kommunikationskanals und benutzt lediglich ein paar Hashfunktionen, um den Transfer des Script sicherzustellen. Diese Einschränkung hat jedoch den Verlust der Privatsphäre im Bezug auf Transaktionsanfrage und Antwort zur Folge.

Bei MPTP kennt zusätzlich zum Händler auch die Bank das Datum und den Betrag der Transaktion, da ihr diese Daten aus der von Kunden ausgefüllten Zahlungsvollmacht und den an sie zwecks Betragstransfers geschickten Tickets bekannt sind.

Da die übertragenen Daten nur signiert, nicht aber chiffriert werden, sind auch für einen potentiellen Angreifer die Transaktionsdaten zugänglich. Gleiches gilt auch für die Kundenidentität. Da es sich bei MPTP um ein nicht anonymes Verfahren handelt, daß keine Verschlüsselung verwendet, ist die Identität des Kunden im Prinzip jedem bekannt.

Der Inhalt einer Transation bei CyberCoin ist gegenüber einem Angreifer durch Verschlüsselungsverfahren geschützt, inwieweit diese Daten jedoch der Bank bekannt sind, ist nicht bekannt. Dem Händler ist die Kundenidentität nur als Pseudonym bekannt.

Da das CyberCoin-Verfahren bei einer Obergrenze des Zahlungsbetrages von 10 US$ wohl hauptsächlich für die Bezahlung einzelner Web-Seiten konzipiert ist, stellt eine Zuordnung des verwendeten Pseudonyms zur eigentlichen Lieferadresse im Gegensatz zu realen Warenlieferungen keine Gefahr da.

8.1.2.2 Vertraulichkeit der Daten in den Endgeräten

Über den Schutz der Daten in den Endgeräten ist bei Millicent nichts bekannt. Mögliche Angriffspunkte wären die Kundendatenbank des Maklers und die Listen, die der Händler über das bereits angenommene Geld führt. Über die Art der Speicherung (Verschlüsselung?) liegen jedoch keine Angaben vor.

Auch bei MPTP fällt der Schutz der Daten (z.B. der Ticketrolle auf dem Rechner des Kunden) nicht in den offiziellen Betrachtungsrahmen des Verfahrens, obwohl dieser Punkt auch bei geringen Beträgen durchaus nicht unwichtig zu sein scheint und einige Angriffsmöglichkeiten bieten würde.

Bei dem CyberCash-Verfahren, aus dem CyberCoin als eine Art Erweiterung entstanden ist, wird immerhin großen Wert darauf gelegt, nicht die vollständige Kreditkartennummer, sondern nur deren Fingerabdruck auf den hauseigenen Servern zu speichern. Deshalb ist davon auszugehen, daß auch im Bezug auf CyberCoin entsprechende Überlegungen angestellt wurden, aber aufgrund der Tatsache, daß uns als Informationsgrundlage zu CyberCoin nur FAQs vorlagen, sind auch hierzu keine näheren Angaben bekannt.

8.1.2.3 Zusammenfassung

Obwohl aufgrund der geringeren Beträge, für deren Zahlung diese Verfahren konzipiert sind, andere Sicherheitsmaßstäbe anzulegen sind, so unterscheiden sich die Sicherheitsanforderungen im Bezug auf Vertraulichkeit nicht allzusehr gegenüber den Verfahren, die mit höheren Betragen arbeiten. Auch bei geringwertigen Beträgen ist die Frage der Anonymität oder die Tatsache, ob ein Angreifer alle Daten abhören kann, nicht unwichtig.

Da MPTP nicht anonym ist und auch auf den Einsatz von Verschlüsselungsverfahren verzichtet, ist es bei einer Bewertung der Vertraulichkeit weit unten einzustufen.

Millicent und CyberCoin bieten beide die Möglichkeit der Verwendung von Pseudonymen, die natürlich in keinster Weise mit den Vorteilen absoluter Anonymität vergleichbar ist, jedoch auf jeden Fall besser als überhaupt keine Anonymität wie bei MPTP.

Bei Millicent ist eine Bewertung abhängig von dem verwendeten Protokoll. Wird das "Private and secure"-Protokoll verwendet, das mit Hilfe der Verwendung symetrischer Verschlüsselungsverfahren durchaus Schutzmaßnahmen bietet, so ist Millicent in den oberen Bereich einzustufen.

Bei CyberCoin werden zum Schutz des Transaktioninhaltes Verschlüsselungsverfahren eingesetzt, so daß CyberCoin natürlich auch besser als MPTP einzuordnen ist.

Eine Rangliste sieht also wie folgt aus:

1. Millicent, CyberCoin

2. MPTP

8.1.3 Integrität

8.1.3.1 Schutz vor Gutschriften ohne gleichzeitige Belastung

Beim CyberCoin-Verfahren finden die Transaktionen auf der Bank statt, daher ist die Erzeugung von Falschgeld wohl kaum möglich.

Bei MPTP handelt es sich um ein Kreditverfahren, d.h. der Kunde kann in betrügerischer Absicht theoretisch so viele Tickets generieren, wie er will. Da die Tickets jedoch händlerspezifisch sind, wird ein Händler die Duplikate sofort erkennen.

Der Makler im Millicent-Modell hätte prinzipiell die Möglichkeit, Falschgeld zu erzeugen, d.h. ein und dasselbe Händler-Scrip mehrfach auszugeben. Da das Scrip jedoch händlerbezogen ist, und der Händler Listen über bereits eingelöstes Geld führt, würde eine Mehrfachausgabe ein und desselben Scrips sofort bemerkt werden (natürlich auch, wenn ein Kunde diese Art des Betruges versucht). Außerdem existiert ein indirekter Schutz vor Betrug seitens der Makler, da diese ihren "guten Ruf" zu verlieren haben.

Alle Verfahren haben gemeinsam, daß sie nur mit sehr geringen Beträgen arbeiten, so daß ein Betrug sich prinzipiell nicht lohnt, es sei denn, man führt diesen "im großen Stil" durch. Dieser Betrug würde aber aufgrund der dafür notwendigen relativ hohen Zahl von Transaktionen sofort auffallen.

8.1.3.2 Schutz vor unberechtigter Belastung

Bei Millicent hat der Makler, falls er eine Einzugsermächtigung oder die Kreditkartendaten des Kunden besitzt, immer die Möglichkeit, das Konto des Kunden ungerechtfertigt zu belasten. Prinzipiell existiert auch hier wieder der Schutz durch den zu verlierenden "guten Namen". Außerdem ist der Kunde durch Bankmechanismen (Widerruf bzw. "Chargeback" bei Kreditkarten) geschützt, d.h. unter Umständen ist die Bank der Geschädigte, meistens müßte jedoch der Händler greifbar sein.

Der Händler im Millicent-Modell hat natürlich die Möglichkeit, das Kundengeld für ungültig zu erklären oder das Geld anzunehmen, ohne Waren zu liefern. Ob hier der Schutz durch den Verlust des "guten Namens" ausreicht, ist fraglich. Der Makler wird natürlich bei Bekanntwerden dieses Verhaltens die Geschäftsbeziehungen mit dem Händler beenden, und dieser ist im Millicent-Modell zwar abhängig vom Makler, ein betrügerischer Bankrott wäre jedoch immer denkbar.

Bei all diesen Überlegungen ist natürlich immer wieder zu berücksichtigen, daß Millicent nur zum Bezahlen kleiner Beträge gedacht ist.

Bei MPTP stellt sich das Problem der unberechtigten Belastung seitens des Händlers nicht, da der Händler selbst keine neuen Tickets generieren kann, und ein doppeltes Senden eines Tickets an die Bank würde dort sofort bemerkt werden. Auch ein Angreifer kann weder Tickets noch Signaturen generieren.

Um den Kunden bei MPTP dagegen zu schützen, daß ein Händler zwar das Geld entgegennimmt, jedoch keine Waren liefert, gibt es sogenannte vorläufige Tickets, die lediglich die Zahlung eines bestimmten Betrages versprechen. Hat der Kunde die Ware erhalten, bekommt der Händler das endgültige Ticket, daß die Zahlung des Betrages bindend zusichert.

CyberCoin versucht diesen Betrug seitens des Händlers dadurch auszuschließen, daß die Transaktion erst endgültig ist, wenn die Ware beim Kunden angekommen ist. Für digitale Waren bedeutet das, daß diese verschlüsselt übertragen werden und die Bezahlung (und die darauf folgende Entschlüsselung) erst dann erfolgt, wenn die Ware fehlerfrei angekommen ist. Für physische Waren verlangt CyberCash, daß diese erst in Rechnung gestellt werden dürfen, nachdem sie abgeschickt wurden.

Da CyberCoin digitale Signaturen als Zahlungsbestätigung verwendet, ist eine gefälschte Belastung technisch nur schwer möglich. Juristisch ist eine Anfechtung dieser Belastung jedoch jederzeit möglich, da es sich hier um beleglose Zahlungen handelt.

8.1.3.3 Schutz vor Umleitung und Manipulation von Zahlungsströmen

Wie gerade erwäht, verwendet CyberCoin digitale Signaturen und Verschlüsselung über die Transaktionsdaten, so daß eine Manipulation oder deren Umleitung weitgehend ausgeschlossen werden kann.

Bei MPTP stellt es für einen potentiellen Angreifer ein Problem dar, daß die verwendeten Tickets händlerspezifisch sind. Ein Angreifer könnte sich zwar in den Transaktionsverkehr zwischen Kunde und Händler einschalten und dafür sorgen, daß falsche Waren gesendet werden, aber wegen deren geringen Wertes (z.B. WWW-Seiten) und der Möglichkeit des Kunden, vorläufige Tickets zu senden, macht es für ihn wenig Sinn. Überdies hat ein Angreifer kaum eine Möglichkeit, die Menge der gesendeten Tickets zu beeinflussen.

Auch bei Millicent ist das Scrip händlerbezogen und außerdem vom Kunden signiert, so daß nur er die entsprechenden Münzen verwenden kann. Eine Manipulation bzw. Umleitung ist deshalb kaum denkbar, es sei denn, man verwendet das “Scrip in the clear“-Protokoll, welches aufgrund des Verzichtes auf Verschlüsselungsverfahren einem Angreifer genügend Möglichkeiten zur Manipulation bietet.

8.1.3.4 Zusammenfassung

Bei der Bewertung dieser Verfahren muß man natürlich immer wieder berücksichtigen, daß es sich hier um die Zahlung geringwertiger Beträge handelt. Ein Betrug ist aufgrund dieser Tatsache meistens nicht lohnend oder würde wahrscheinlich wegen der dazu benötigten relativ hohen Zahl an zu manipulierenden Transaktionen sofort auffallen.

Trotzdem sind auch bei diesen Verfahren einige Sicherheitsmaßnahmen implementiert.

Den besten Schutz vor Falschgeld bietet wohl CyberCoin, da die Transaktionen direkt bei der Bank stattfinden. Auch MPTP und Millicent haben Mechanismen, um Duplikate als solche zu entlarven, der Schutz bei Millicent vor der Erstellung von Falschgeld seitens des Maklers erscheint jedoch nicht ausreichend, da er lediglich in der Gefahr des Verlustes seines "guten Namens" besteht.

Gleiches gilt für Millicents Schutzmaßnahmen gegen ungerechtfertigte Belastung, wo auch wieder der "gute Name" ein entscheidender (aber nicht ausreichender) Schutzfaktor ist. Hier liegt das Problem darin, daß die Rolle der allgemein als vertrauenswürdig angesehenen Institution "Bank" im Millicent-Modell durch die Makler ausgefüllt wird.

Die Verwendung digitaler Signaturen bei CyberCoin bzw. die Möglichkeit vorläufiger Ticket bei MPTP schütz hier wesentlich besser vor unberechtigter Belastung.

Auch im Bezug auf Manipulation und Umleitung sichert die Verwendung digitaler Signaturen und Verschlüsselungsverfahren CyberCoin eine obere Plazierung in der Rangliste an Ende dieses Abschnittes zu.

Um eine Umleitung bzw. Manipulation zu erschweren, verwenden auch Millicent und MPTP signierte Münzen, die zudem noch händlerbezogen sind. Bei Millicent handelt es sich jedoch um keine "echte" digitale Signatur, es wird lediglich ein Hash über die Daten, aus denen sich die Münze zusammensetzt, gebildet.

Aufgrund des schlechten Abschneidens von Millicent in den ersten beiden Punkten würde eine Rangliste jedoch wie folgt aussehen:

1. CyberCoin

2. MPTP

3. Millicent

8.1.4 Verläßlichkeit

8.1.4.1 Transportverläßlichkeit

Bei CyberCoin verwendet man ein mehrstufiges Protokoll, daß bei digitalen Waren die Zahlung erst endgültig macht, wenn die Ware beim Kunden in verschlüsseltem Zustand vorliegt. Auf diese Art und Weise wird verhindert, daß der Kunde für eine Ware bezahlt, die aufgrund von Netzwerkfehlern nur unvollständig oder gar nicht angekommen ist. Hat die (noch verschlüsselte) Ware den Kunden fehlerfrei erreicht, so wird dieser seine endgültige Zahlung an den Händler machen, woraufhin er den entsprechenden Schlüssel zum Entschlüsseln seiner Ware erhält. Denkbar wäre es natürlich, daß es gerade zu einem Netzausfall während der Übertragung des Schlüssels kommt, was aber aufgrund der im Vergleich zu der eigentlichen Ware relativ geringen Datenmenge des Schlüssels wenig wahrscheinlich ist.

Auch bei MPTP gibt es eine Schutzmaßnahme gegen das Bezahlen defekter Waren, und zwar die vorläufigen Tickets, die dazu dienen, die Bezahlung erst zu komplettieren, wenn die Ware den Kunden ordnungsgemäß erreicht hat. Der Händler kann natürlich prinzipiell das Geld für seine Waren verlieren, wenn er vorläufige Ticket akzeptiert und keine endgültige Zahlung seitens des Kunden erfolgt. Er hat jedoch mit dem vorläufigen Ticket etwas in den Händen, mit dem er bei der Bank zumindest nachweisen kann, daß ihm Geld versprochen worden ist. Eine weitere Eigenschaft von MPTP ist es auch noch, daß es sich hier um ein Kreditverfahren handelt, so daß ein Kunde sein Geld nicht aufgrund eines Festplattencrashes oder eines Netzwerkausfalls verlieren kann.

Bei Millicent liegen leider nicht genug Angaben über das Transportprotokoll vor, es ist jedoch davon auszugehen, daß Münzen, die ihren Bestimmungsort aufgrund von Netzwerkstörungen nicht erreicht haben, verloren sind.

8.1.4.2 Wirkung einer Beschädigung der Endgeräte

Bei Millicent wird auch zu diesem Punkt kein Mechanismus zum Schutz erwähnt. Es ist jedoch zu vermuten, daß bei einer Beschädigung der Endgeräte beim Kunden und beim Makler die betroffenen Münzen verloren sind. Problematisch wird es natürlich auch bei einer Beschädigung des Händlerendgerätes, da dieser dann eventuell keine Listen mehr hat, anhand derer er feststellen kann, ob es sich bei neu eintreffendem Scrip um gültige Münzen handelt oder nicht.

Eine Beschädigung des Kundenendgerätes ist bei MPTP wie schon erwähnt, kein Problem, da es sich hier um ein Kreditverfahren handelt. Händler und Bank benötigen im Beschädigungsfall jedoch ein Backup.

CyberCoin schützt den Kunden im Falle einer Beschädigung seines Endgerätes dadurch, daß sich sein Geld nicht bei ihm selbst auf der Festplatte, sondern auf einem Konto bei CyberCash bzw. deren Partnerbanken befindet. Von dort kann das Geld dann auf ein normales Konto zurücküberwiesen werden.

8.1.4.3 Zusammenfassung

MPTP besitzt aufgrund der Tatsache, daß es sich hier um ein Kreditverfahren handelt, und außerdem die Möglichkeit des Sendens vorläufiger Tickets besteht, eine ausreichende Verläßlichkeit.

Auch CyberCoin ist durch die Methode der endgültigen Zahlung erst bei korrektem Erhalt der Ware und der Tatsache, daß das eigentliche Geld auf CyberCoin-Konten verwaltet wird, im Punkt Verläßlichkeit als gut einzustufen.

Lediglich Millicent bietet wenig Schutzmaßnahmen, so daß eine Rangvergabe folgendermaßen aussehen würde:

1. CyberCoin, MPTP
2. Millicent

8.1.5 Zurechenbarkeit

8.1.5.1 Entlarvung von Fälschern

Sowohl MPTP als auch CyberCoin sind so konzipiert, daß sie keine Fälschungen zulassen, d.h. die Frage der Entlarvung stellt sich hier nicht.

Bei Millicent würde ein Abgleich der Logs und Signaturen eine Entlarvung prinzipell erlauben, vorausgesetzt, man hat nicht das ohne Aufwand zu fälschende "Scrip in the clear"-Protokoll benutzt. Eine Gerichtsverwertbarkeit dieses Abgleichs sei jedoch dahingestellt.

8.1.5.2 Beweis einer erfolgten Zahlung

Bei MPTP handelt es sich um ein Kreditverfahren, d.h. der Kunde kann seine erfolgten Zahlungen anhand der Kontoauszüge beweisen.

Auch bei CyberCoin ließe sich die Zahlung wahrscheinlich über die Abrechnung beweisen, es sind hierzu aber keine näheren Angaben bekannt.

Schwer beweisbar ist eine Zahlung dagegen bei Millicent. Im Grunde ist ein Beweis nicht garantiert, da Geld unter Umständen auch "gestohlen" werden kann. Abgesehen davon würde hier Aussage gegen Aussage stehen.

8.1.5.3 Autorisierung von Zahlungen

Nur der wahre Kunde hat bei MPTP genügend Informationen, um gültige Tickets zu generieren. Indem er diese mitsamt der Zahlungsvollmacht an den Händler schickt, autorisiert er seine Zahlung.

Bei CyberCoin autorisiert der Kunde seine Zahlungen mit Hilfe seiner digitalen Signatur.

Auch bei Millicent autorisiert der Kunde die Zahlung durch seine Signatur, wobei auch hier zu beachten ist, daß die Verwendung des "Scrip in the clear"-Protokolls wieder die Ausnahme bildet.

8.1.5.4 Gegenseitige Authentisierung / Identifizierung

Nur der Kunde wird bei MPTP durch ein Bankzertifikat authentisiert.

Bei CyberCoin findet dagegen nur eine indirekte Authentisierung bzw. Identifizierung über die sogenannte "Persona" statt. Es ist jedoch auch zulässig, daß eine Persona von mehreren Kunden benutzt wird, oder ein Kunde mehrere Personas besitzt.

Millicent bietet einen Mechanismus, den man allenfalls als indirekte Identifizierung bezeichnen kann. Der Händler berechnet nämlich bei Erhalt der Zahlung die Signatur des Scrips erneut und weiß so, daß es sich um gültige Münzen eines bestimmten Kunden handelt. Außerdem kann der Händler optional seine Antwort, als daß Wechselgeld, mit dem Kundenschlüssel signieren, um sich so gegenüber dem Kunden zu identifizieren.

8.1.5.5 Mißbrauch durch das organisierte Verbrechen

Alle drei hier besprochenen Verfahren sind nicht anonym, somit stellt sich die Frages des Mißbrauchs durch das organisierte Verbrechen nicht. Außerdem handelt es sich hier ja auch um Verfahren zur Zahlung geringwertiger Beträge, so daß sich ein Mißbrauch gar nicht lohnen würde bzw. mit viel zu großem Aufwand erfolgen müßte.

8.1.5.6 Zusammenfassung

Im Bezug auf den Mißbrauch durch das organisierte Verbrechen weisen alle drei Verfahren die gleichen Eigenschaften auf. Betrachtet man den Punkt der Entlarvung von Fälschern, so ist es natürlich immer besser, wenn es gar nicht erst zu Fälschungen kommen kann, so wie bei MPTP und CyberCoin. Alle drei Verfahren bieten eine Autorisierung von Zahlungen, der Beweis dieser Zahlungen ist im Gegensatz zu MPTP und CyberCoin bei Millicent im Grunde nicht wirklich möglich. Auch eine gegenseitige Authentisierung findet bei Millicent nicht statt, allenfalls eine Identifizierung. Bei CyberCoin kann man immerhin noch von einer indirekten Authentifizierung sprechen, bei MPTP findet diese zwar direkt statt, dafür aber nur für den Kunden durch ein Bankzertifikat.

Eine Einordnung der Verfahren in eine Rangliste würde folgendermaßen aussehen:

1. MPTP, CyberCoin
2. Millicent

8.1.6 Transferierbarkeit

Alle drei Verfahren sind nur dafür konzipiert, Zahlungen zwischen Mensch und Geschäft durchzuführen. Eine Transfer von Geld zwischen Menschen untereinander ist nicht vorgesehen.

8.1.7 Skalierbarkeit

Die hier besprochenen drei Verfahren sind natürlich alle nur für kleine Beträge konzipiert, über genaue Limits sind jedoch keine näheren Angaben bekannt, lediglich bei CyberCoin spricht man von Beträgen zwischen 0.25$ und 10$.

Millicent läßt beliebige Wertabstufungen zu, der entsprechende Wert ist im Scrip gespeichet. Pro Partei (Händler / Makler) gibt es eine eigene Währung. Eine Ausweitung von Millicent auf einen großen Teilnehmerkreis stellt kein Problem dar, da mehrere dezentrale Makler möglich sind, und die Überprüfung des Scrip beim jeweiligen Händler stattfindet.

Gleiches gilt für MPTP, auch hier sind Wertabstufungen möglich, und ein weite Ausbreitung des Systems stößt nicht auf Schwierigkeiten, da sich MPTP ansonsten wie eine normale Kreditkartenzahlung verhält.

Auch CyberCoin unterstützt mehrere Banken, die Unterstützung unterschiedlicher Währungen mit automatischer Konvertierung ist jedoch erst in Planung.

8.1.7.1 Zusammenfassung

Keines der hier besprochenen Verfahren hat im Bezug auf Skalierbarkeit die Probleme eines "Bottleneck", die sich z.B. bei einer zentralen Bank ergeben würden. Auch bei der Betrachtung von Limits, Wertabstufungen und Währungen unterscheiden sich die Verfahren kaum, so daß bei einer Einordnung in eine Rangliste alle Verfahren denselben Platz belegen würden.

8.1.8 Organisatorisches

Bei CyberCoin kann der Kunde jederzeit sein auf den CyberCoin-Konten befindliches Geld auf sein eigenes Konto zurücküberweisen lassen, bei Millicent liegen keine Angaben zur Rücktauschbarkeit vor, und bei MPTP stellt sich die Frage der Rücktauschbarkeit erst gar nicht.

Der Fall eines Verlustes von Geld kann bei MPTP und CyberCoin nicht eintreten, bei Millicent trägt vermutlich der Kunde das Risiko.

Generell liegen bei allen Verfahren wenig Angaben zu Haftungsfragen vor, nur bei CyberCoin wird wohl im Zweifelsfall der Händler haften, da es sich hier um ein belegloses Verfahren handelt.

8.2 Digitale Kreditkarten

8.2.1 Gewichtung

Beim Bezahlen mit Hilfe von Kreditkarten ergeben sich allgemeine Probleme und Risiken, die auch die hier besprochenen Verfahren größtenteils nicht berücksichtigen bzw. verhindern können.

8.2.1.1 Die Risiken aus Sicht des Kunden

Das größte Problem bei Kreditkarten ist, daß die Person, die im Besitz der Kreditkartendaten ist, mit diesen Informationen (in den meisten Fällen genügen Nummer und Ablaufdatum) beliebig, z.B. per Telefon, einkaufen kann. Diese Person muß nicht einmal ein Dritter sein, der diese Daten auf unrechtmäßige Weise erlangt hat, z.B. aus einem Abfallbehälter im Kaufhaus, in den die Kunden nach dem Bezahlen achtlos ihre Kaufquittungen werfen. Auch der Händler selbst, der diese Daten von Kunden freiwillig erhalten hat, kann diese mißbrauchen. Er könnte dem Kunden z.B. zusätzliche Artikel in Rechnung stellen.

Natürlich sollten diese Artikel möglichst unauffällige Bezeichnungen tragen und den eigentlich zu zahlenden Betrag nicht übermäßig vergrößern, da der Kunde dies sonst bemerken könnte. Tritt dieser Fall ein, so wird der Kunde die Zahlung verweigern, wobei wir bei der Betrachtung der Risiken für den Händler wären:

8.2.1.2 Die Risiken für den Händler

Selbst wenn kein Betrug seitens des Händlers vorliegt, hat der Kunde grundsätzlich die Möglichkeit einer Zahlungswiderrufung. Dieses Verfahren wird "Chargeback" genannt und bedeutet, daß dem Kunden von seiner Kreditkartengesellschaft unter bestimmten Umständen die Möglichkeit eingeräumt wird, seine Zahlung innerhalb einer bestimmten Frist, meistens 90 Tage, zu widerrufen. Dies ist auch der Grund dafür, daß First Virtual den Händlern ihr Geld erst nach Ablauf von 91 Tagen auf deren Konten überweist und sich so gegen ein Chargeback seitens des Kunden absichert.

Der Händler trägt somit das größte Risiko, da er im Falle einer späteren Zahlungsverweigerung des Kunden sein Geld nicht erhält.

8.2.2 Vertraulichkeit

8.2.2.1 Vertraulichkeit des Transaktionsinhaltes und der Kundenidentität

Bei SSL werden die Transaktionen mit Hilfe von RC4 verschlüsselt übertragen, um Transaktionsinhalt und Kundenidentität zu erfahren, müßte ein Angreifer also den RC4 Verschlüsselungsalgorithmus brechen, wobei die Sicherheit bei der Exportversion des Netscape-Browsers, die nur eine Schlüssellänge von 40 Bit verwendet, umstritten ist. Da es sich um ein Kreditkartenverfahren handelt, sind dem Händler Identität und Kreditkartennummer des Kunden bekannt, inwieweit er jedoch die gekauften Posten gegenüber der Bank meldet, bleibt ihm überlassen, er wird dies im Falle einer großen Anzahl von Einzelposten sicher nicht detailliert aufschlüsseln.

Auch bei S-HTTP hat man optional die Möglichkeit der Verschlüsselung der Transaktionen durch symmetrische oder asymmetrische Verfahren, welche Verfahren im einzelnen verwendet werden, ist von der entsprechenden Implementation auf beiden Seiten abhängig. Genügen die Verfahren der

Gegenseite nicht den eigenen Sicherheitsvorstellungen, kommt keine Verbindung zu stande. Ein Angreifer müßte also auch hier das verwendete Verschlüsselungsverfahren brechen, um Kundenidentität und Transaktionsinhalt zu erfahren. S-HTTP ist kein konkretes Zahlungsprotokoll, sondern nur ein Übertragungsprotokoll, z.B. für die Übermittlung von Kreditkartendaten. In diesem Fall kennt der Händler natürlich auch wieder Kreditkartennummer und Identität des Kunden, und für den Wissensstand der Bank über den Transaktionsinhalt gilt das gleiche wie bei SSL.

Bei First Virtual werden die Transaktionsdaten im Klartext per e-mail übertragen, d.h. neben Kunde und Händler sind auch jeder außenstehenden Person, die die e-mail ließt, Transaktionsinhalt und Identität des Kunden bekannt. Der Kunde hat zwar die Möglichkeit, Pseudonyme zu verwenden, die Sicherheit wird dadurch jedoch nicht erhöht, lediglich die konkrete Profilbildung wird etwas erschwert. Auch bei diesem Verfahren bleibt es dem Händler überlassen, inwieweit er die gekauften Posten detailliert aufschlüsselt, in diesem Fall zuerst gegenüber First Virtual, die es dann (außerhalb des Internets) an die entsprechende Kreditkartengesellschaft weiterleitet.

8.2.2.2 Vertraulichkeit der Daten in den Endgeräten

Über die Sicherheit der Daten in den Endgeräten liegen lediglich bei First Virtual entsprechende Angaben vor. Hier schützt man sich vor äußeren Angriffen durch Firewall-Techniken, bei denen immer nur einzelne im Batch-Betrieb laufende Rechner die Verbindung zum Internet bilden, die außerdem keine Standard-Internetdienste verwenden. Bei SSL und S-HTTP bleibt es den Händlern überlassen, inwieweit sie die bei sich gelagerten Daten schützen, generell kann also davon ausgegangen werden, daß sich einem Angreifer dort potentiell gute Möglichkeiten bieten.

8.2.2.3 Zusammenfassung

Da alle hier beschriebenen Verfahren auf der Abrechnung über Kreditkartengesellschaften beruhen, ist eine Anonymität des Kunden gegenüber Bank und Händler natürlich nicht gegeben. Inwieweit die einzelnen Transaktionsposten der Bank bekannt werden, hängt auch nicht vom Verfahren sondern nur vom Händler ab, ein Vergleich der einzelnen Verfahren bezüglich dieses Kriteriums wäre deshalb nicht sinnvoll. Der Transaktionsinhalt wird wohl bei S-HTTP am besten geschützt, da man hier einen Einfluß auf die Wahl des verwendeten Verschlüsselungsverfahrens hat und so immer ein den aktuellen Sicherheitsstandards genügendes Verfahren auswählen kann (Es sei denn, man legt dies nicht in seinen Voreinstellungen fest und überläßt die Wahl dem verwendeten Browser). Außerdem wird natürlich vorausgesetzt, daß das entsprechende Verfahren auf beiden Seiten implementiert ist, durch die hohe Flexibilität von S-HTTP dürfte die jedoch kein größeres Problem darstellen.

Auch bei SSL ist man nicht auf RC4 als Verfahren festgelegt, dessen Sicherheit wie schon erwähnt bei den Exportversionen umstritten ist. Es nützt jedoch nichts, wenn der SSL-Standard zwar andere Verfahren vorsieht, diese in der Praxis jedoch nicht benutzt werden.

Bei First Virtual verzichtet man dagegen ganz auf die Verschlüsselung der Transaktionsdaten, der Inhalt der Transaktion ist also für jedermann lesbar, man muß dem Verfahren jedoch zu gute halten, daß die eigentliche Kreditkartennummer, die ein Angreifer auch außerhalb des Netzes mißbrauchen könnte, nie über das Internet übertragen wird. Eine Rangliste bezüglich der Vertraulichkeit ergibt sich wie folgt:

1. S-HTTP
2. SSL
3. First Virtual

8.2.3 Integrität

8.2.3.1 Schutz vor Gutschriften ohne gleichzeitige Belastung

Die Möglichkeit einer Gutschrift ohne gleichzeitige Belastung ergibt sich bei den hier betrachteten Verfahren nicht, da es sich um Kreditkartenverfahren handelt, bei denen die eigentliche Abbuchung bei der Bank stattfindet.

Unter Umständen könnte ein Kunde sich höchstens Waren liefern lassen und dann später behaupten, jemand anderes hätte seine Kreditkartennummer benutzt. Der Geschädigte wäre in diesem Fall der Händler (siehe Abschnitt 8.2.1).

8.2.3.2 Schutz vor unberechtigter Belastung

Um eine unberechtigte Belastung zu erreichen, müßte ein Angreifer in den Besitz der Kreditkartendaten des Kunden gelangen.

Bei SSL und S-HTTP hätte er hierzu entweder die entsprechend verwendeten Verschlüsselungsverfahren zu brechen, oder er könnte sich als Händler maskieren, da dieser schließlich die benötigten Daten vom Kunden freiwillig erhält. Da der Kunde (wie schon in Abschnitt 8.2.1 erwähnt) immer die Möglichkeit hat, im Falle eines vermuteten Betruges seine Zahlung zu widerrufen, wäre es für den Händler nicht sinnvoll, einfach überhöhte oder zusätzliche Forderungen an den Kunden zu stellen, er müßte mit den erhaltenen Kreditkartendaten schon woanders einkaufen und die Waren an eine Deckadresse liefern lassen.

Bei First Virtual wird die eigentliche Kreditkartennummer nicht über das Netz geschickt, ein Angreifer kann also nicht in ihren Besitz gelangen, um damit (außerhalb des Netzes) unberechtigte Belastungen durchzuführen. Die anstelle der Kreditkartennummer verwendete VirtualPin wird jedoch im Klartext übertragen, ein Angreifer könte also mit Hilfe dieser Pin unberechtigte Einkäufe tätigen. Hierzu müßte er jedoch in der Lage sein, die (ein- und ausgehende) e-mail des ursprünglichen Kunden abzufangen und zu fälschen. Eine von First Virtual per e-mail vom Kunden geforderte Autorisierung für einen von einem Angreifer getätigten Kauf würde der ursprüngliche Kunde natürlich mit “`no`“ oder “`fraud`“ beantworten, der Angreifer müßte diese e-mail also so manipulieren, daß eine positive Autorisierung entsteht. Eine andere Möglichkeit wäre, die e-mail Leitung des Kundenrechners so auszulasten, daß dieser keine e-mail mehr empfangen oder verschicken kann und der Angreifer so anstelle des wahren Kunden die e-mail beantworten kann. Der Geschädigte wäre am Ende in jedem Fall wieder der Händler, da der rechtmäßige Besitzer der VirtualPin spätestens beim Erhalt seiner Kreditkartenabrechnung die Zahlung widerrufen wird.

8.2.3.3 Schutz vor Umleitung und Manipulation von Zahlungsströmen

Da bei diesen kreditkartenbasierten Verfahren die eigentlichen Zahlungsströme außerhalb des Netzes stattfinden, kann ein Angreifer höchstens den Warenstrom umleiten und an eine Deckadresse liefern lassen. Hierzu müßte er bei SSL und S-HTTP die Verschlüsselungsverfahren brechen bzw. bei First Virtual die e-mail entsprechend manipulieren. Ein Angreifer könnte sich außerdem als Händler maskieren, um so die Zahlungen in seine Richtung zu leiten, er handelt sich jedoch so auch wieder alle mit der Zahlungswiderufung durch den Kunden verbundenen Probleme ein, so daß er am Ende wohl nur auf die Kreditkartennummer des Kunden aus wäre.

Zur Manipulation von Zahlungsströmen sei erwähnt, daß der Händler prinzipiell die vereinbarte Summe oder andere Transaktionsdaten verfälschen kann, da der Kunde sie nicht signiert, bei First Virtual würde dies dem Kunden schon bei der per e-mail geschickten Autorisierungsanfrage auffallen, bei SSL und S-HTTP spätestens beim Erhalt der Kreditkartenabrechnung, wobei wir wieder bei den allgemeinen Problemen für den Händler bei Kreditkartenzahlungen wären (siehe Abschnitt 8.2.1).

8.2.3.4 Zusammenfassung

Generell ist der Kunde bei allen Verfahren durch die von den Kreditkartengesellschaften eingeräumte Möglichkeit des "Chargeback" gegenüber unberechtigter Belastung und Manipulation der Zahlungsströme geschützt, so daß der Händler am Ende immer der Geschädigte ist.

Bei First Virtual erkennt der Kunde eine unberechtigte Belastung oder Manipulation nicht erst nach dem Eintreffen der Kreditkartenabrechnung, sondern schon beim Erhalt der von First Virtual per e-mail geschickten Autorisierungsanfrage (vorausgesetzt, sie wurde nicht kompromittiert). Bei S-HTTP und SSL soll ein Ausspähen der Kreditkartennummer und eine Manipulation der Zahlung durch den Einsatz von Verschlüsselungsverfahren schon von vornherein verhindert bzw. erschwert werden. Obwohl die Schlüssellänge von 40 Bit bei dem von den Exportversionen von SSL verwendeten Verschlüsselungsverfahren umstritten ist, ist dies kein Grund, SSL in einer Rangliste schlechter einzustufen als S-HTTP, denn man hat schließlich auch bei SSL die Möglichkeit, ein anderes Verschlüsselungsverfahren zu verwenden, auch wenn dies niemand wirklich nutzt. In jedem Fall ist der Einsatz eines Verschlüsselungsverfahrens als Schutzmaßnahme besser, als wenn man, wie First Virtual, ganz darauf verzichtet.

Eine Rangvergabe würde deshalb wie folgt aussehen:

1. S-HTTP, SSL

2. First Virtual

8.2.4 Verläßlichkeit

8.2.4.1 Transportverläßlichkeit

Natürlich kann es immer vorkommen, daß eine S-HTTP oder SSL Nachricht, bzw. eine First Virtual e-mail aufgrund von Netzwerkfehlern seinen Adressaten nicht erreicht. Da es sich bei S-HTTP und SSL um die einfache Übertragung einer Kreditkartennummer handelt, kommt die Nummer beim

Empfänger entweder an oder nicht, der Vorgang ist in jedem Fall atomar und erfüllt die ACID-Eigenschaften.

Auch bei First Virtual können in den meisten Fällen keine inkonsistenten Zustände entstehen. Erreicht die VirtualPin den Händler nicht, wird er keine Autorisierungsanfrage an First Virtual stellen, erreicht hingegen die Autorisierungsanfrage des Händlers First Virtual nicht nicht, wird der Händler nie eine positive Nachricht empfangen und es wird deshalb keine Transaktion zustande kommen. Schlägt die Weiterleitung der Autorisierungsanfrage von First Virtual an den Kunden fehl, wird auch keine ungewollte Zahlung stattfinden, da der Kunde keine Zahlung autorisert hat. Das gleiche gilt, wenn die Antwort des Kunden verloren geht, da auch hier keine Zahlung ohne Autorisierung erfolgt. Probleme könnte es höchstens geben, wenn die vom Kunden positiv beantwortete Autorisierungsanfrage auf dem Weg von First Virtual zurück an den Händler verloren geht. First Virtual würde aufgrund der positiven Antwort die Kreditkarte des Kunden belasten und nach 91 Tagen das Geld auf das Konto des Händlers überweisen wollen. Da der Händler aber keine positive Nachricht empfangen hat, wird er keine Waren an den Kunden weitergeleitet haben. Vorausgesetzt, der Kunde bemerkt die fehlerhafte Belastung durch First Virtual und wiederruft die Zahlung, wird auch in diesem Fall "kein Geld in falsche Hände geraten".

8.2.4.2 Wirkung einer Beschädigung der Endgeräte

Auf der Seite des Kunden hätte eine Beschädigung der Endgeräte keine schwerwiegenden Folgen, da alle Verfahren auf dem Kreditprinzip basieren, der Kunde also kein eigentliches Geld bei sich lagert. Bei den Händlern wären die Folgen einer solchen Beschädigung schon schwerwiegender, falls er die Waren schon geliefert hat, aber noch nicht über die Bank abgerechnet hat und die Daten nicht als Backup hat. Die Bank, insbesondere auch First Virtual, sollte interne Sicherheitsmaßnahmen besitzen, da im Falle einer Beschädigung bei First Virtual die gesamte Kundendatenbank verloren gehen könnte. Außerdem zahlt First Virtual erst nach Ablauf von 91 Tagen die entsprechenden Beträge an die Händler aus, bei einem Datenverlust dürfte die Zuordnung der Zahlungen jedoch Probleme bereiten.

8.2.4.3 Zusammenfassung

Aufgrund der Ähnlichkeit der Verfahren im Bezug auf Verläßlichkeit kann eigentlich keine echte Rangeinteilung stattfinden, deshalb belegen die Verfahren alle denselben Rang.

1. S-HTTP, SSL, First Virtual

8.2.5 Zurechenbarkeit

8.2.5.1 Entlarvung von Fälschern

Fälschung im Bezug auf Kreditkartenzahlungsverfahren würde in diesem Fall die unerlaubte Benutzung der Kreditkartennummer bedeuten. Wer jedoch am Ende die Kreditkartendaten zum Warenerwerb benutzt hat, ist im Prinzip nicht feststellbar, vor allem da ein Angreifer die Waren wahrscheinlich an eine Deckadresse schicken lassen wird. Eine Ausnahme wäre es, wenn ein Händler die Kreditkarte eines Kunden zusätzlich oder widerrechtlich belastet. Wenn der Kunde seine Kreditkartenabrechnung kontrolliert, wird er diese Tatsache feststellen und seine Zahlung widerrufen. Der Händler wäre somit entlarvt und würde sein Geld auch nicht erhalten.

Das Problem bei First Virtual ist, daß eine e-mail zwar verhältnismäßig einfach zu fälschen ist, der Urheber der falschen e-mail jedoch praktisch nicht zu ermitteln ist, vorausgesetzt der Angreifer versteht sein Handwerk.

Um die Nachrichten bei SSL oder S-HTTP zu fälschen, müßte ein Angreifer die verwendeten Verschlüsselungsverfahren brechen. Wenn ihm dies gelingen sollte, wird er auch in der Lage sein, seine Identität so zu schützen, daß er nicht entlarvbar ist.

8.2.5.2 Beweis einer erfolgten Zahlung

Da es sich hier um Kreditkartenzahlungsverfahren handelt, wird ein Kunde eine erfolgte Zahlung spätestens anhand der Kreditkartenabrechnung beweisen können.

Bei Verwendung geeigneter Verfahren (MACs, digitale Signaturen, ...) ließe sich bei S-HTTP zumindest der Erhalt einer Nachricht (und somit der Kreditkartennummer) beweisen. Gleiches gilt auch für SSL, vorausgesetzt man würde ein anderes als das zur Zeit standardmäßig verwendete Verfahren RC4 verwenden.

Den Erhalt oder das Abschicken einer einfachen e-mail zu beweisen ist jedoch bei dem von First Virtual verwendeten Verfahren nicht möglich bzw. vorgesehen.

8.2.5.3 Autorisierung von Zahlungen

Da im Prinzip jeder, der im Besitz der Kreditkartendaten ist, die Karte belasten kann, ohne dabei eine Unterschrift oder Ähnliches zu leisten, findet eine Autorisierung durch die Person, auf die die Karte ausgestellt ist, bei Kreditkartenzahlungen generell nicht statt.

Im Gegensatz zu SSL und S-HTTP versucht First Virtual dieses Problem zu verringern, indem es die durch den Händler geforderte Belastung der Kreditkarte durch eine e-mail des Kunden bestätigen läßt, bevor die eigentliche Abbuchung erfolgt. Da eine e-mail wie schon beschrieben jedoch nicht fälschungssicher ist, hat man keine vollständige Kontrolle über seine e-mail und kann dieses Verfahren im eigentlichen Sinne nicht als Autorisierung bezeichnen.

8.2.5.4 Gegenseitige Authentisierung / Identifizierung

Für S-HTTP gilt, daß auch in diesem Punkt durch die Wahl geeigneter Verfahren (MACs, digitale Signaturen, ...) eine Authentisierung bzw. Identifizierung vorgesehen und optional möglich ist.

Bei SSL wird zumindest der Händler über ein Zertifikat gegenüber dem Kunden authentifiziert. Dieses Verfahren ist jedoch leicht angreifbar, da die Existenz des Zertifikates dem Kunden lediglich durch ein kleines Symbol in der unteren Eckes des Browser-Fensters angezeigt wird. Den Inhalt des Zertifikats muß sich der Kunde jedoch über einen extra Menüpunkt anzeigen lassen. Diese Tatsache bietet einem Angreifer die Möglichkeit, sich mit Hilfe eines beliebigen Zertifikates als ein bestimmter Händler zu maskieren und so ahnungslose und vor allem nachlässige Kunden (da nur wenige den extra Menüpunkt aufrufen werden) zu betrügen.

Da First Virtual nur das einfache e-mail Verfahren verwendet, ist eine gegenseitige Authentisierung natürlich nicht vorgesehen. Der Händler kennt in bestimmten Fällen nicht einmal die genaue Identität des Kunden, da First Virtual auch die Verwendung von Pseudonymen unterstützt.

8.2.5.5 Mißbrauch durch das organisierte Verbrechen

Die in diesem Abschnitt besprochenen Verfahren basieren alle auf der Zahlung per Kreditkarte, ein Mißbrauch durch das organisierte Verbrechen ist deshalb auszuschließen, da diese Zahlungsart nicht anonym ist und man deshalb zu viele Spuren hinterlassen würde.

8.2.5.6 Zusammenfassung

Die Entlarvung von Fälscher bzw. von Personen, die die Kreditkartennummer widerrechtlich benutzen, ist bei allen drei Verfahren ähnlich, d.h. bemerkt der Kunde den Betrug anhand seiner Kreditkartenabrechnung, trägt der Händler den Verlust, da der Kunde von seiner Möglichkeit der Zahlungsverweigerung gebrauch macht. Bleibt der Betrug jedoch unbemerkt, trägt der Kunde den entstandenen Verlust. Auch wenn der Händler selbst nicht die Kreditkartendaten mißbraucht hat, so trägt er doch immer die Verluste, da ein Angreifer, der mit Hilfe von geklauten Kreditkartennummern bezahlt, seine Identität vermutlich immer so verbergen wird (z.B. durch Verwendung von Deckadressen), daß er nicht entlarvbar ist.

Das Fälschen der eigentlichen Nachricht dürfte bei First Virtual am einfachsten sein, da es sich hier nur um eine einfache e-mail im Klartext handelt. Gelingt es einem Angreifer auch, die bei SSL und S-HTTP verwendeten Verschlüsselungsverfahren zu brechen, so ist bei allen drei Verfahren die Fälschung einer Nachricht in der Art möglich, daß der Urheber nicht feststellbar wäre.

Der Beweis einer erfolgten Zahlung ist bei allen drei Verfahren im eigentlichen Protokoll bisher nicht vorgesehen, zumindest der Beweis des Erhaltens einer Nachricht könnte jedoch im Gegensatz zu First Virtual bei S-HTTP und SSL durch Verwendung geeigneter Verfahren implementiert werden. Wie schon erwähnt, ist ein Beweis jedoch bei allen 3 Verfahren spätestens anhand der Kreditkartenabrechnung möglich.

Die Autorisierung der Zahlung per e-mail, die bei First Virtual verwendet wird, bietet aufgrund des fehlenden Einsatzes von Verschlüsselungsverfahren auch nur wenig Sicherheit und deshalb nur einen geringen Vorteil gegenüber SSL und S-HTTP, bei denen dieser Punkt unberücksichtigt bleibt.

Gegenseitige Authentisierung bzw. Identifizierung ist bei S-HTTP am besten möglich, bei SSL findet diese zur Zeit nur von der Händlerseite aus statt (Zertifikat) und bei First Virtual fehlt sie ganz, der Kunde kann sogar Pseudonyme benutzen.

Zusammenfassend betrachtet würde eine Rangliste bezüglich Zurechenbarkeit wie folgt aussehen:

1. S-HTTP, SSL

2. First Virtual

8.2.6 Transferierbarkeit

Bei allen drei Verfahren ist ein Transfer der Beträge nur einmalig vom Kunden zum Händler möglich, da dies bei Kreditkartenzahlungsverfahren im allgemeinen nur so vorgesehen ist.

8.2.7 Skalierbarkeit

8.2.7.1 Limits

Über eine maximale Betragshöhe liegen bei allen drei Verfahren keine speziellen Angaben vor, da es sich um Kreditkartenzahlungen handelt, werden im Zweifelsfall die vom kartenausgebenden Institut vorgeschriebenen Limits gelten, falls es welche gibt. Eine Untergrenze ergibt sich rein rechnerisch für die Beträge, bei denen die eigentliche Betragshöhe geringer ist als die für die Transaktion an den Kreditkartenausgeber zu zahlende Gebühr.

8.2.7.2 Währungen

Das es beim Bezahlen mit Kreditkarte im Prinzip immer möglich ist, in unterschiedlichen Währungen abzurechnen, sind alle drei Verfahren in diesem Punkt nicht eingeschränkt.

8.2.7.3 Anzahl der Banken

Alle hier erwähnten Verfahren basieren auf dem bestehenden Kreditkartensystem, und da es eine Vielzahl unterschiedlicher Anbieter gibt, und die Zahlung per Kreditkarte offline möglich ist, werden hier keine Engpässe entstehen.

Ein Problem ergibt sich lediglich bei First Virtual, da dort alle Transaktionen über einen zentralen First Virtual Server laufen, der in der Vergangenheit schon mal für einen längeren Zeitraum ausgefallen ist, was erhebliche Probleme verursachte.

8.2.7.4 Zusammenfassung

Im Prinzip entsprechen sich die Verfahren in jedem hier betrachteten Punkt, nur bei First Virtual ist die Skalierbarkeit aufgrund des einen zentralen Servers eingeschränkt.

8.2.8 Organisatorisches

8.2.8.1 Haftungsfragen

Wie im Abschnitt 8.2.1 erwähnt, haftet bei kreditkartenbasierten Zahlungen immer der Händler, falls der Kunde die widerrechtlich erfolgten Belastungen seiner Karte bemerkt und die Möglichkeit des "Chargeback" wahrnimmt. Tritt der Fall ein, daß der Kunde den Mißbrauch seiner Kreditkartendaten nicht bemerkt, haftet er natürlich selbst für die ihm entstandenen Schäden.

Die Frage der Rücktauschbarkeit und der Erstattung verlorenen Geldes stellt sich bei diesen auf Kreditkartenzahlung basierenden Verfahren nicht.

8.3 Digitale Schecks

8.3.1 Gewichtung

Da die in diesem Abschnitt bewerteten Verfahren auch zur Bezahlung von nicht geringfügigen Beträgen gedacht sind, sind hohe Anforderungen an ihre Sicherheit zu stellen.

Im Gegensatz zu Debitverfahren ist es bei einem Kreditverfahren kaum möglich, die Identität des Kunden völlig geheimzuhalten. Zumindest die Bank wird in der Regel die Identität des Kunden kennen, da sie ihm schließlich einen Kredit gewährt.

Es ist aber nicht zwangsläufig nötig, daß der Händler diese Information ebenfalls erhält. Es genügt, wenn er vor der Lieferung der Ware erfährt, daß sein Kunde kreditwürdig ist. Die Bank dagegen braucht nicht zu erfahren, welche Waren der Kunde gekauft hat. Hält man die erwähnten Informationen (Identität, Transaktionsinhalt) getrennt, so ist eine sinnvolle Erstellung von Profilen nur eingeschränkt möglich. Eine reine Pseudonymisierung reicht dazu aber nicht aus, da hier u.U. zusätzliche Informationen (z.B. die Lieferadresse) dazu genutzt werden können, einem Profil das eigentlich nur über ein Pseudonym gebildet wurde, doch noch eine Person zuzuordnen.

Einem außenstehenden Beobachter sollte es schließlich ebenfalls nicht möglich sein, die Identität oder Transaktionsdaten zu erfahren. Leider wird der eigentliche Vorgang der Aushandlung der Transaktionsdaten oft aus juristischen Gründen ausgeklammert.

Ein wichtiger Unterschied zu den digitalen Kreditkarten besteht in der Tatsache, daß es hier dem Händler nicht möglich sein darf, dem Kunden Beträge abzubuchen, die dieser nicht authorisiert hat. Das bedeutet auch, daß eventuell benutzte Kreditkartendaten hier nicht nur vor Lauschern, sondern auch vor dem Händler geheimzuhalten sind, da dieser mit ihnen auch außerhalb der Schutzmechanismen des Systemes einkaufen kann.

8.3.2 Vertraulichkeit

8.3.2.1 Vertraulichkeit des Transaktionsinhaltes

Bei Cybercash wird der Transaktionsaktionsinhalt in der ersten Nachricht im Klartext übertragen. Er kann daher problemlos durch Dritte abgehört werden. Die Bank dagegen erfährt vom Transaktionsinhalt nur Transaktionsnummer, Betrag und Datum.

SET lagert die eigentliche Verhandlung des Transaktionsinhaltes aus juristischen Gründen aus dem Protokoll aus. Wird dieser Vorgang nicht extra mit einem eigenen Verfahren geschützt, so kann er von einem potentiellen Angreifer beobachtet werden. Die Bank erfährt aber nur Datum und Preis, da ihr sogar von der Transaktionsnummer nur ein Hashwert mitgeteilt wird.

Bei Netcheque wird zwar eine MAC gebildet, eine Verschlüsselung findet aber nicht statt.

8.3.2.2 Vertraulichkeit der Daten in den Endgeräten

Dieser Punkt ist bei keinem der Verfahren Gegenstand der Betrachtung.

8.3.2.3 Schutz der Kundenidentität

CyberCash verwendet Pseudonyme. Die Kreditkartendaten werden mit dem Schlüssel des CyberCash-Servers DES chiffriert. Bei der Registrierung eines Kunden bei CyberCash wird der Name des Kunden mit DES verschlüsselt. Eine Erstellung von Profilen lohnt sich daher wohl nur, wenn der Kunde seine Identität freiwillig preisgibt (z.B. durch die Angabe einer Lieferadresse). In diesem Fall können sowohl ein Angreifer als auch der Händler entsprechende Auswertungen vornehmen.

Eine andere Gefahr bildet die Tatsache, daß Kreditkartendaten als Teil der Identität übertragen werden. Sie werden zwar mittels DES chiffriert, aber die Header am Beginn des verschlüsselten Teiles sind bekannt:

`swversion:` $< Softwareversion >$
`amount:` $< Betrag >$

in der Zahlungsnachricht, bzw.

```
type: bind-credit-card
swversion: < Softwareversion >
```

wodurch ein Brute-Force-Angriff auf den DES stark vereinfacht wird.

Wie in Anhang A.2.2 angeführt, sind sich namhafte Kryptologen darüber einig, daß ein derartiger Angriff für größere Organisationen auch praktisch durchführbar ist. Es wäre daher durchaus denkbar, daß das organisierte Verbrechen dies ausnutzen und damit die Kreditkartendaten einer größeren Anzahl Kunden in seinen Besitz bringen könnte. Diese Überlegungen werden aber vermutlich keine praktischen Auswirkungen haben, da das alte CyberCash-Protokoll in naher Zukunft durch SET ersetzt werden soll.

SET macht kein Geheimnis aus der Kundenidentität. Diese dürfte in den reichlich übermittelten Zertifikaten stehen, womit sie sowohl dem Händler als auch einem Lauscher bekannt wäre. Einer Bildung von Profilen steht damit nichts im Wege. Ob die Verschlüsselung der Kreditkartendaten sicherer ist als beim CyberCash-Protokoll, kann zum gegenwärtigen Zeitpunkt nicht entschieden werden. Allerdings wird an einigen Stellen der Protokollbeschreibung ersichtlich, daß Probleme wie die oben diskutierten durchaus eine Rolle beim Entwurf spielten.

Da NetCheque keine Verschlüsselung benutzt, ist ein Schutz der Kundenidentität nicht gegeben. Immerhin basiert das Verfahren nicht auf Kreditkartendaten, womit diese auch keines Schutzes bedürfen.

8.3.2.4 Zusammenfassung

Während SET und Cybercash den Transaktionsinhalt vor der Bank geheimhalten, geschiet dies bei NetCheque nicht. Einem Angreifer stehen diese Daten in jedem Fall zur Verfügung. Da CyberCash als einziges Protokoll Pseudonyme verwendet, stehen dem Händler und dem Angreifer bei den anderen Verfahren alle nötigen Daten zur Profilbildung zur Verfügung. Bei CyberCash muß dazu erst noch die Zuordnung Pseudonym zu Realname geschehen. Dies ist aber kaum ein Problem, wenn physische Güter eingekauft werden. In diesem Fall enthält die Lieferadresse in der Regel die nötigen Angaben.

Da bei SET und CyberCash Kreditkartendaten übertragen werden, wäre hier ein besonders starker Schutz nötig. Bei SET scheint dies gegeben zu sein, falls für die eigentliche Verschlüsselung ein ausreichend starkes Verfahren eingesetzt wird. Bei CyberCash dagegen scheint das verwendete Protokoll Schwächen zu enthalten, die die begrenzte Stärke des verwendeten DES weiter schwächen.

Für die Erstellung der Rangliste waren die folgenden Überlegungen ausschlaggebend: CyberCash hat zwar gewisse Vorteile beim Datenschutz, fällt aber durch seinen mangelhaften Schutz der Kreditkartendaten auf. NetCheque und SET bieten dagegen praktisch keinen Datenschutz, übertragen aber entweder keine Kreditkartendaten oder verfügen über komplizierte kryptographische Verfahren zu deren Schutz. Da Sicherheit in der Regel vor Datenschutz geht, folgt:

- SET, NetCheque
- CyberCash

8.3.3 Integrität

8.3.3.1 Schutz vor Gutschriften ohne gleichzeitige Belastung

Dies kann so bei keinem der Verfahren geschehen, da der eigentliche Geldtransfer auf der Bank stattfindet. Es könnte allerdings möglich sein, „ungedeckte Schecks“ auszustellen[2] und sich damit Waren zu erschwindeln.

Wer in so einem Fall das Risiko trägt, ist unklar. Alle Verfahren verwenden digitale Signaturen oder MAC's unterschiedlicher Güte. Mit dem neuen Telekomunikationsgesetz werden nun zum ersten Mal digitale Signaturen als rechtskräftige Unterschrift zugelassen. Es ist aber noch nicht klar, ob die bei den einzelnen Verfahren verwendeten Signaturen auch den vom Gesetz gestellten Anforderungen genügen.

Zumindest die bei CyberCash und SET verwandten Signaturen könnten es dann durchaus erlauben, einen Schuldigen vor Gericht zu bringen, sofern er sich nicht durch Flucht seiner Strafe entzogen hat.

Das bei NetCheque verwendete konventionelle Verfahren wird dagegen den Anforderungen an eine digitale Signatur wohl kaum genügen. Der Händler müßte daher das Risiko selber tragen.

8.3.3.2 Schutz vor unberechtigter Belastung

Der zu bezahlende Betrag wird in allen drei Verfahren vom Kunden durch digitale Signaturen authorisiert. Innerhalb des Systems ist eine Manipulation daher unwahrscheinlich, sofern die Sicherheit der verwendeten Schlüssel gewährleistet ist.

[2] Ob das geht, hängt von den Vertragsbedingungen ab. Liegt dem Verfahren ein Kreditkartensystem zu Grunde, so könnte der Kunde z.B. der Kreditkartenrechnung widersprechen (Chargeback).

Eine andere Frage ist die Verwendung von übertragenen Daten außerhalb des Systems. Sowohl SET als auch CyberCash übertragen Kreditkartendaten, deren Besitz ausreicht, auf fremder Kosten einzukaufen. Sie werden zwar durch Chiffrierung geschützt, ein Brute-Force-Angriff auf die verwendete Chiffre erscheint aber zumindest bei Cybercash nicht unmöglich.

8.3.3.3 Schutz vor der Umleitung und Manipulation von Zahlungsströmen

Auch hier dürfte die Verwendung digitaler Signaturen die übermittelten Nachrichten gegen Manipulation hinreichend schützen.

8.3.3.4 Zusammenfassung

Alle Verfahren schützen die Integrität der übertragenen Daten durch digitale Signaturen. Es gibt zwar Unterschiede in der Güte der verwendeten Verfahren, daran aber eine Rangliste aufstellen zu wollen, dazu fehlen uns die exakten Spezifikationen. Auch hat sich in der Vergangenheit gezeigt, daß die Bewertung der konkreten Implementation eines Verfahrens Tücken in sich birgt, an denen selbst erfahrene Kryptographen scheitern.

8.3.4 Verläßlichkeit

8.3.4.1 Transportverläßlichkeit

Da hier Kreditverfahren verwendet werden, kann Geld nicht durch Übertragungsfehler vernichtet oder vervielfältigt werden.

Einzig die Gefahr einer Beschädigung der Ware auf dem Transport könnte ein Problem darstellen. Dieses Problem liegt aber außerhalb des Blickwinkels der betrachteten Protokolle.

8.3.4.2 Wirkung einer Beschädigung der Endgeräte

Da das Geld des Kunden nie wirklich auf seinem Rechner liegt, ist eine Beschädigung seines Endgerätes kein Problem. Bank und Händler dagegen brauchen ein Backup ihrer Daten. Zumindest bei den Banken kann dies als gegeben angesehen werden, da sie in der Regel über das nötige Sicherheitsbewußtsein verfügen.

Bei den Händlern kann dies nicht zwangsläufig vorausgestzt werden, weshalb bei ihnen prinzipiell die Gefahr eines Verlustes der Tageseinnahmen besteht falls ihr Rechner kurz vor dem Abgleich mit der Clearingstelle seine Daten verliert. Dies ist aber ein generelles Risiko, das ein Händler in seine Kalkulation einbeziehen muß, wenn er sich auf eine Offline - Bearbeitung von Kreditkartentransaktionen einläßt.

8.3.4.3 Zusammenfassung

Die betrachteten Verfahren haben keine eingebauten Maßnahmen, um Verläßlichkeit sicherzustellen. Da es sich hier aber um Kreditsysteme handelt, besteht bei einem ordnungsgemäßen Backup seitens Bank und Händler auch keine Notwendigkeit dazu.

8.3.5 Zurechenbarkeit

8.3.5.1 Entlarvung von Fälschern

Im Gegensatz zur Verwendung elektronischer Münzen ist es hier nicht möglich, durch mehrfaches Senden der selben Nachricht die Geldmenge im System zu verändern. Selbst das unentdeckte Senden eines doppelten Schecks würde nur dazu führen, daß Geld vom Konto des Opfers auf das des Angreifers transferiert würde.

Da in allen Verfahren der Empfänger auf dem Scheck eingetragen ist, wäre der Angreifer jemand, der schon einmal vom Opfer Geld erhalten hätte. Diese doppelte Einzahlung des selben Betrages könnte ihm nachgewiesen werden, womit der Kunde in der Regel eine gute Chance hätte, seine Ansprüche geltend zu machen.

Hinzu kommt, daß die Schecks auch ein Datum oder eine Seriennummer enthält, das es erlaubt, doppelte Schecks bei der Einlösung zu erkennen und ihre Einlösung zu verweigern, womit die Transaktion nicht zustande käme.

Alles vorher gesagte gilt selbstverständlich nur, wenn die verwendeten kryptographischen Verfahren, die den Inhalt der Nachricht gegen Manipulation schützen, sicher sind. Der bei CyberCash verwendete DES steht zwar unter harscher Kritik, es ist aber nicht damit zu rechnen, daß die amerikanische Regierung längere Schlüssel in absehbarer Zeit für den Export freigibt.

SET wird vermutlich die gleichen Verfahren auf eine deutlich kompliziertere Art und Weise benutzen, während NetCheque auschließlich symmetrische Verfahren benutzt. Diese bieten zwar nicht die Möglichkeit wirkliche digitale Signaturen zu generieren, sie sind aber ausreichend die Anforderungen aus den obigen Szenarien zu erfüllen, sofern die Bank als vertrauenswürdig gilt und die Schlüssellänge ausreichend ist. Da uns das verwendete Verfahren nicht bekannt ist, können wir über seine Sicherheit keine Aussage machen.

Aber auch wenn ein Angreifer im Besitz der Schlüssel ist, muß er schließlich immer noch den Scheck auf der Bank einlösen. Der Vorgang hinterläßt grundsätzlich Spuren, die nachvollziehbar sind.

8.3.5.2 Beweis einer erfolgten Zahlung

Bei Scheckverfahren generiert das System derartige Nachweise. Dies ist bei CyberCash und SET die Kreditkartenrechnung, bei NetCheque kann der Kunde die `statement`-Funktion des Buchungsservers aufrufen.

8.3.5.3 Authorisierung von Zahlungen

Bei CyberCash und SET werden Zahlungen durch digitale Signaturen authorisiert, während NetCheque ein konventionelles Verfahren verwendet.

8.3.5.4 Gegenseitige Authentisierung

Bei Cybercash findet nur eine indirekte Authentisierung statt. Hier überprüft die Bank ob beide Parteien sich im klaren über die Identität des Gegenüber sind.

Auch bei NetCheque authentisieren sich die Parteien nicht untereinander, sondern nur gegenüber dem Buchungsserver mittels Kerberos-Tickets.

Bei SET überprüfen die beteiligten Parteien gegenseitig ihre Zertifikate.

8.3.5.5 Mißbrauch durch das organisierte Verbrechen

Die genannten Verfahren sind nicht anonym. Dieses Kriterium ist daher nicht anwendbar.

8.3.5.6 Zusammenfassung

Die Verfahren sind nicht anonym. Jede Transaktion hinterläßt deutliche Spuren.

Dazu ist bei allen drei Verfahren der Schutz vor Fälschungen gegeben, soweit die verwendeten kryptographischen Verfahren nicht überwunden werden können.

Somit ist sowohl ein hinreichender Schutz vor Geldwäsche, Verleugnung des Erhaltes einer Zahlung, als auch Fälschung gegeben. Schuldige müßten immer damit rechnen, identifiziert und haftbar gemacht zu werden.

8.3.6 Funktionalität

8.3.6.1 Transferierbarkeit

Alle drei Verfahren sind ausschließlich dazu gedacht eine einfache Kund - Händler - Beziehung abzubilden.

8.3.6.2 Skalierbarkeit

Bei allen drei Verfahren sind unterschiedliche Banken und Währungen vorgesehen. Auch eine Anpassung an erhöhte Verkehrsaufkommen dürfte generell leicht zu erreichen sein.

8.3.7 Organisatorisches und Rechtliches

8.3.7.1 Haftung beim Auftauchen von Falschgeld

Obwohl uns bis auf CyberCash keine Vertragsbedingungen vorliegen, kann davon ausgegangen werden, daß bis zum Inkraftreten eines Gesetzes, das digitale Signaturen regelt, der Kunde das Recht hat, Transaktionen für nicht von ihm getätigt zu erklären.

Wie gängige Kreditkartenverfahren zeigen, trägt in so einem Fall der Händler das Risiko.

8.3.7.2 Rücktauschbarkeit

Die Rücktauschbarkeit ist bei allen drei Verfahren gegeben.

8.3.7.3 Erstattung verlorenen Geldes

Nicht anwendbar, da Geld nicht verloren gehen kann.

8.4 Digitales Bargeld

8.4.1 Gewichtung

Da es sich bei den hier betrachteten Systemen um Verfahren zur Bezahlung von nicht geringfügigen Beträgen handelt, sind gegenüber den Verfahren zur Bezahlung kleinerer Beträge höhere Anforderungen an die Sicherheit der Verfahren zu stellen. Mißbrauch und Verlust digitaler Werte hätte hier stärkere Auswirkungen.

Es soll aber auch nicht außer acht gelassen werden, daß es sich hier im Gegensatz zu den digitalen Kreditkarten und Schecks um Debit - Verfahren handelt. Damit böte dieser Bereich eine echte Chance, anonyme Zahlungsverfahren zu realisieren. Dies wäre im Kredit-Bereich nur durch zugegebenermaßen akademische Verfahren wie dem in [Chaum 87] vorgestellten möglich.

Während es bei den digitalen Schecks und Kreditkarten vor allem darauf ankommt, die Kreditkartennummern vor dem Händler und eventuellen Angreifern geheim zu halten, so sollten Verfahren in dieser Kategorie es prinzipiell erlauben, die Identität des Käufers vor jedem zu verbergen, solange er sie nicht von sich aus preisgibt, um z.B. bei Streitigkeiten bezüglich einer Transaktion seinen Standpunkt zu untermauern.

Verfahren, die es erlauben, einen Kunden zu entlarven, der dieselbe elektronische Münze mehrfach ausgibt, verletzen diese Regel nicht, da hier der Kunde sein Einverständnis implizit durch die Verletzung der Regeln gab. Es darf aber nicht möglich sein einen Kunden zu reidentifizieren, der sich korrekt verhält.

Möchte man die Vorteile des Bargeldes, so muß man auch damit rechnen, einige seiner Nachteile zu übernehmen. So kann physisches Bargeld verloren gehen oder gestohlen werden, während z.B. unausgefüllte Schecks noch keinen Wert repräsentieren und daher ihr Verlust keinen Verlust von Geld zur Folge hat. Auch ist es bei Kredit - Systemen möglich, Transaktionen rückgängig zu machen, während bei Bargeld gilt: „Was weg ist, ist weg“

Des weiteren ergeben sich Probleme mit der Konstantheit der Geldmenge. Während sich bei Kreditkarten und Schecks die Geldmenge im System immer feststellen läßt, bedeutet Bargeld die Möglichkeit, Falschgeld zu erzeugen. Betrachtet man digitales Bargeld als eine eigene Währung, so bedeutet das Erzeugen von Falschgeld, die Aushebelung der direkten Korrespondenz zwischen dem digitalen Bargeld und dem physischen.

D.h. es könnte die Situation entstehen, daß im System mehr digitales Bargeld existiert, als physisches in die Bank eingezahlt wurde. Da für so einen Fall keine „Inflation“ vorgesehen ist, würde dies entweder eine Destabilisation der Bank oder der physischen Währung bedeuten, je nachdem, ob das neu erzeugte Geld in großen Mengen als physisches Geld abgehoben wird, und so die die Bank zahlungsunfähig macht, oder ob es schleichend in die physische Währung umgesetzt wird und nun dort die Geldmenge erhöht.

Dies alles legt die Vermutung nahe, daß ein höherer Schutz der Privatsphäre immer auch einen Verlust in der Transaktionalität (ACID) der Zahlungen zur Folge hat. Dies gilt es in der Bewertung zu berücksichtigen.

8.4.2 Vertraulichkeit

8.4.2.1 Vertraulichkeit des Transaktionsinhaltes

Bei allen betrachteten Verfahren liegt das Aushandeln der zu kaufenden Waren außerhalb des Blickfeldes. Dieser Vorgang könnte also beobachtet werden, sollte er über unsichere Kanäle erfolgen. Da zu diesem Zeitpunkt die Identität des Kunden aber noch nicht bekannt ist, können wir diesen Punkt im Rahmen der Betrachtung vernachlässigen, solange besagter Angreifer diese Zuordnung nicht zu einem späteren Zeitpunkt nachholen kann.

Aber auch während des Zahlungsprotokolls muß der Betrag der Zahlung übertragen werden. Diese Übertragung erfolgt bei NetCash, der Geldkarte und Mondex verschlüsselt und ist somit einem Angreifer, der das verwendete Verfahren nicht brechen kann, unzugänglich. Es muß allerdings gesagt werden, daß die konkrete Implementation von NetCash durch die NetBank Verschlüsselung mittels PGP zwar vorsieht, dies aber nicht mandatorisch macht.

Bei ecash und CAFE ist eine Verschlüsselung im Prinzip unnötig, da eine Zuordnung zum Käufer durch die Anonymität der Verfahren nicht erfolgen kann.

Da bei Mondex auf der Karte ein Log der letzten 10 Transaktionen geführt wird, die Karte für das Bezahlen freigeschaltet werden muß, und das Log einer freigeschalteten Karte mittels eines frei erhältlichen Gerätes ausgelesen werden kann, ist es dem Händler prinzipiell möglich, dieses auszulesen und festzustellen, ob der Kunde gerade bei der Konkurenz eingekauft hat[3]. Da das Verfahren nur pseudonym ist, kann er diese Daten außerdem mit früheren Transaktionen verknüpfen und so einen Eindruck vom Kaufverhalten des Kunden gewinnen. (Dies gilt natürlich vorerst nur für POS-Transaktionen[4]. Inwieweit dies bei der geplanten Übertragung des Systemes

[3]Der unauffälligste Weg dies zu tun, wäre eine kleine Schaltung zu benutzen, die die Karte zuerst mit dem Gerät zum Auslesen des Transaktionslogs und dann mit dem eigentlichen Terminal verbindet. Da prinzipiell verschiedene Hersteller Mondex - kompatible Geräte liefern, würde dies dem Kunden bei einer professionellen Gestaltung des Gerätes nicht auffallen.

[4]**POS P**oint **O**f **S**ale = „Ort der Zahlung" steht für Systeme, mit denen die Transaktion direkt an der Ladenkasse über ein Terminal stattfindet. Dieses muß dazu aber nicht online mit der Bank verbunden sein. Dieser Ausdruck wurde eigentlich dazu benutzt zwischen dem elektronischen Auslesen von Karten und den altmodischen Kreditkarten - Systemen zu unterscheiden, die mit Durchpausen der Kreditkartendaten und Unterschrift arbeiteten. Hier benutzen wir die Bezeichnung um zwischen dem Bezahlen an einer „wirklichen" Ladenkasse in einem physisch vorhandenen Laden und einer virtuellen Ladenkasse

auf das Internet noch möglich ist, kann jetzt noch nicht entschieden werden.)

Da auch bei der Geldkarte die letzten 15 Transaktionen auf der Karte gespeichert werden, ist hier prinzipiell das gleiche Risiko anzunehmen. Auch hier gibt es ein einfaches Gerät, um diese Daten auszulesen. Da dort aber nur die Salden der Transaktionen und der letzten 3 Aufladungen angezeigt werden, nicht aber der jeweilige Transaktionspartner, nützen dem Händler die erhaltenen Daten nicht zur Profilbildung.

Im letzten Schritt werden die Transaktionsdaten an die Bank übertragen, die damit zwangsläufig den Betrag und das Datum der Transaktion, sowie die Identität des Händlers kennt.

8.4.2.2 Vertraulichkeit der Daten in den Endgeräten

Dieser Punkt wird bei Mondex, CAFE und der Geldkarte dadurch gelöst, daß Chipkarten als relativ sichere Engeräte eingesetzt werden. Die Gefahr eines Einbruches in diese ist zwar nicht auszuschließen (s. [Spiegel 96], [AnKu], [Kuhn 96]), ist aber sicherlich deutlich geringer als die Gefahr eines Einbruches in ein Computersystem, das mit dem Internet verbunden ist. Man sollte aber nicht vergessen, daß diese Systeme in erster Linie für POS - Zahlungen gedacht waren und daher die Warscheinlichkeit, daß auf sie physisch zugegriffen werden kann, größer ist.

Bei NetCash und ecash kommt diese Gefahr besonders zu tragen, da im Protokoll Public - Key - Verfahren verwendet werden. Gelänge ein Einbruch in den Rechner einer der Parteien, so könnte deren geheimer Schlüssel für Maskeradenangriffe benutzt werden. Auch auf dem Rechner gefundene Transaktionsdaten würden dem Angreifer dadurch bekannt.

Betrachten wir zuerst den Rechner des Kunden. Einen physischen Zugriff wollen wir nicht annehmen, da der Rechner vermutlich in einer Privatwohnung steht und ein direkter Zugriff für einen Angreifer vermutlich zu aufwendig im Verhältnis zum erzielten Gewinn wäre. Auch ein Angriff über das Internet ist unwahrscheinlich, da normale Anwender heutzutage in der Regel keine Server für Internetdienste betreiben. Mit dem Aufkommen von Javascript und Active-X, kann so ein Angriff aber nicht mehr völlig ausgeschlossen werden.

Gelänge es einem Angreifer, auf den Rechner des Kunden zuzugreifen, so muß davon ausgegangen werden, daß er hinterher in der Lage ist, auf Kosten des Kunden einzukaufen. Da zumindest bei ecash auch ein Transaktionslog geführt wird, wäre ihm auch der Inhalt der letzten Transaktionen bekannt.

---

im Internet zu unterscheiden. (POS bedeutet hier also: „beim Kaufmann um die Ecke“.)

Der Rechner des Händlers dagegen ist stärker gefährdet, da er per definitionem einen Internetdienst anbietet. Die Gefahr eines Angriffes über das Internet kann daher als gegeben angesehen werden. Leider kann nicht davon ausgegangen werden, daß auch entsprechende Schutzmaßnahmen (Firewalls, starke Authentifizierungsmaßnahmen) getroffen wurden, da es sich bei den Händlern oft um Geschäfte kleiner und mittlerer Größe handelt, die nicht über das Sicherheitsbewußtsein und Know - How verfügen, wie man es z.B. von einer Bank erwarten würde.

Ein Angriff auf den Rechner des Händlers wird dem Angreifer die Informationen liefern, die dieser benötigt, um dessen Konten zu leeren. Da diese Gefahr aber prinzipiell bei jedem Internetzahlungssystem besteht und nicht Teil der Zahlungsverfahren ist, wollen wir in dieser Arbeit nicht näher darauf eingehen.

Über die Transaktionsdaten wird ein Angreifer auf diesem Wege aber nicht mehr erfahren, als er auch auf anderem Wege herausfinden könnte. Der Kunde bleibt sowohl bei NetCash als auch bei ecash dem Händler gegenüber anonym.

Angriffe auf Bankrechner werden wir hier nicht betrachten, da man davon ausgehen sollte, daß Banken in der Regel ausreichend gesichert sind.

8.4.2.3 Schutz der Kundenidentität

Bei ecash ist der Kunde grundsätzlich anonym.

Das CAFE - Verfahren garantiert Anonymität solange der Kunde nicht versucht, dieselben Münzen mehrfach auszugeben.

Mondex ist prinzipiell pseudonym gegenüber dem Händler, es sollte aber nicht vergessen werden, daß im POS - Geschäft der Händler den aufgeprägten Namen lesen und auch (z.B. durch eine kleine Videokamera) aufzeichnen kann. Gegenüber der Bank ist die Kundenidentität aber nicht geschützt.

Ähnlich verhält es sich mit der Geldkarte. Diese ist laut Aussage der Banken zwar anonym gegenüber dem Händler, die Evidenzstellen erfahren aber im Transaktionsdatensatz die Kontonummern der beteiligten Parteien. In Zusammenarbeit mit den Banken ist eine Zuordnung aber möglich und auch für Reklamationsfälle vorgesehen.

Auch bei NetCash ist der Kunde dem Händler gegenüber anonym. Der Währungsserver, bei dem er physisches Geld in digitales (oder umgekehrt) umwandelt, kann darüber Buch führen und ist somit in der Position die Transaktionen des Kunden nachzuvollziehen. In [MeNe 93a] heißt es dazu

> *Es wird von Währungsservern erwartet dies nicht zu tun, und es ist wahrscheinlich daß die Verträge mit ihren Klienten dies ausdrücklich ausschließen. Hinzu kommt, daß der Klient sich seinen eigenen Währungsserver wählen kann, und den wählen wird, dem er vertraut.*

Bei der konkreten Implementation durch die NetBank identifiziert sich der Kunde gegenüber der Bank über seine Email-Adresse. Diese kann zwar leicht geschützt werden (Remailer, Benutzen temporärer Email-Accounts auf CD's von Online Diensten, etc.), einem normalen Kunden ohne Betrugsabsicht oder Paranoia ist dies aber wohl kaum zuzumuten.

8.4.2.4 Zusammenfassung

Bei allen Verfahren kann ein Angreifer, der eine Transaktion mithört maximal Datum, Betrag und Empfänger einer Zahlung feststellen.

Angriffe auf die Endgeräte der Kunden wären denkbar, scheinen aber unwahrscheinlich. Angriffe auf die Rechner der Händler könnten diesen finanzielle Verluste bringen, was aber keine Schwäche der bewerteten Verfahren, sondern ein generelles Problem ist.

Die hauptsächliche Problematik liegt bei diesen Verfahren in der Gefahr der Profilbildung durch Banken und Händler.

Während der Händler bei NetCash, ecash, CAFE und der Geldkarte wahrscheinlich keine Möglichkeit hat, die Transaktionen dem Kunden zuzuordnen, ist es ihm bei Mondex im POS - Betrieb prinzipiell möglich, sogar Transaktionen bei anderen Händlern in seine Profile einzubeziehen. Sehen wir davon einmal ab, so ist der Kunde ihm gegenüber nur pseudonym, womit eine Profilbildung immer noch möglich ist. Hat der Händler Glück, so gibt ihm der Kunde sogar noch Name und Adresse, wenn physische Waren gekauft werden.

Die Banken haben da schon mehr Möglichkeiten. Nur ecash und CAFE erlauben es dem Kunden anonym zu bleiben. Bei NetCash, Mondex und der Geldkarte muß der Kunde in dieser Hinsicht seiner Bank vertrauen.

Eine Rangliste bezüglich der Vertraulichkeit würde daher etwa so aussehen:

1. CAFE, ecash

2. Geldkarte, NetCash

3. Mondex

8.4.3 Integrität

8.4.3.1 Schutz vor Gutschriften ohne gleichzeitige Belastung

Das mehrfache Ausgeben von Münzen wird bei der Geldkarte schon im Vorfeld durch relativ sichere Hardware und Plausibilitätsprüfungen der Kartendaten bei jeder Transaktion angegangen. Da hier die eigentliche Transaktion auf der Bank stattfindet und sich das Geld nicht wirklich auf der Karte befindet, sollte eine Buchung ohne Gegenbuchung praktisch ausgeschlossen sein.

Während die eigentliche Geldmenge außerhalb des Systems auf diese Weise nicht vermehrt werden kann, besteht allerdings die theoretische Gefahr, daß es jemandem gelänge, eine Karte so zu manipulieren, daß sie der anderen vorspiegelt, der zu zahlende Betrag wäre von ihr abgebucht, während er in der nächsten Transaktion wieder als Guthaben erscheint. Dies hätte eine Schädigung der Bank zur Folge, da diese dem Händler seine Umsätze garantiert.

Auch Mondex und CAFE verlassen sich auf ihre sichere Hardware. Da sich bei Ihnen aber das Geld wirklich auf der Karte befindet, wäre es theoretisch denkbar, Falschgeld zu erzeugen und in Umlauf zu bringen. Dies wird aber durch physische Schutzmaßnahmen und kryptographische Protokolle praktisch wohl verhindert.

Da bei Mondex nicht auf der Bank kontrolliert werden kann, ob wirklich zu jeder Buchung eine Gegenbuchung existiert, wäre es sogar möglich, die Geldmenge zu erhöhen und so Banken oder Währungen zu destabilisieren[5]. Bei CAFE müßte dem Angreifer dazu der geheime Schlüssel der Bank bekannt sein, wovon wohl in der Regel nicht auszugehen ist.

Bei NetCash und ecash wird das Erzeugen von Falschgeld durch kryptographische Verfahren und Seriennummern verhindert, solange die Sicherheit der verwendeten Schlüssel gewährleistet ist.

Da es sich bei ihnen um Online - Systeme handelt, kann der Händler sicher sein, daß er sein Geld erhält, bevor er seine Waren liefert.

Gelänge es aber einem Angreifer falsche Münzen zu erzeugen, so hätte das bei NetCash zur Folge, daß die Münzen eines anderen Kunden plötzlich zu Falschgeld würden, da bei diesem Verfahren sichergestellt wird, daß eine einzuwechselnde Münze auch tatsächlich erzeugt wurde.

Dies bedeutet, das bei diesem Verfahren der Kunde geschädigt wird, die Geldmenge aber nicht erhöht werden kann.

---

[5] Die Warscheinlichkeit dafür dürfte aber eher gering sein

Bei ecash dagegen wird geprüft, ob eine Seriennummer noch nicht eingelöst wurde. Hier könnte besagter Angreifer Münzen erzeugen, die es vorher so nicht gab. Dabei würde effektiv die Geldmenge erhöht und die Bank direkt geschädigt.

Es soll hier aber noch einmal betont werden, daß die verwendeten Verfahren, soweit sie dokumentiert sind, es einem Angreifer nicht einfach machen, Falschgeld zu erzeugen. Geht man davon aus, daß ein Angreifer im Besitz der Schlüssel ist und die Hardware manipulieren kann, so ist wohl kein System als sicher zu betrachten.

Allerdings erlaubt es die konkrete Implementation von NetCash durch die NetBank, unverschlüsselt mit der Bank zu kommunizieren. Dabei bestünde die Gefahr, daß Geld gestohlen werden kann.

8.4.3.2 Schutz vor unberechtigter Belastung

Gehen wir davon aus, daß ein Angreifer nicht im Besitz der Kundenidentität (geheimer Schlüssel, Chipkarte) ist, so kann er keine Zahlung ohne dessen Wissen initiieren.

Damit bleiben nur noch die Gefahren, daß der Kunde über die Höhe des abgebuchten Betrages getäuscht wird, oder, daß ihm Geld abgebucht wird, ohne daß er dafür einen Gegenwert erhält.

Bei Mondex und der Geldkarte ist es prinzipiell möglich, das Händlerterminal so zu manipulieren, daß es einen niedrigeren Betrag als den tatsächlich abgebuchten anzeigt.

Dies ist bei CAFE und ecash nicht möglich, da das eigene Gerät vom Kunden nicht aus der Hand gegeben wird und selber den Betrag anzeigt, der dann vom Kunden per Tastendruck oder PIN bestätigt werden muß.

Das NetCash - Protokoll sieht vor, daß der Kunde die Zahlung initiiert. Wird dies auch in der Praxis so implementiert, ist eine Täuschung des Kunden eigentlich ausgeschlossen.

NetCash sieht auch vor, daß optional ein Verfahren benutzt werden kann, daß die Ausstellung einer Quittung durch den Händler erzwingt. Bei der Bezahlung physischer Waren ist dies das Maximum an Sicherheit, das ein Kunde realistischerweise erwarten kann. Bei der Bezahlung von Informationen kann die Quittung die Ware selber sein, da ihre Realisierung für das Verfahren transparent ist.

Die aktuelle Implementation erlaubt dies aber nicht, vielmehr ist hier eher die Variante „anonymer Händler“ realisiert worden.

Bei der Geldkarte, Mondex und CAFE handelt es sich um Systeme die vor allem zur Bezahlung in der physischen Welt konzipiert wurden. Dort erhält der Kunde entweder seine Ware oder eine Quittung, die er vor Gericht als Beweis vorlegen kann. Für Zahlungen über das Internet dagegen fehlen Konzepte, um die Lieferung sicherzustellen.

Der Fairness halber sollte man hier aber festhalten, daß die Transaktionsdaten bei der Geldkarte reidentifiziert werden können, womit prinzipiell bewiesen werden kann, daß ein Händler eine Summe erhalten hat. Dies ist aber aufwendiger als bei ecash, da die Kreditinstitute von den Evidenzstellen keine Mitteilung der Einzeltransaktionen erhalten. Ob sich die Banken daher in der Praxis darauf einlassen erscheint uns fraglich.

Bei ecash schließlich führt die Klientensoftware Buch über die getätigten Transaktionen und kann im Streitfalle durch Preisgabe der Blinding-Faktoren der benutzten Münzen dem Händler den Erhalt des Geldes beweisen.

8.4.3.3 Schutz vor der Umleitung und Manipulation von Zahlungsströmen

Da bei allen Verfahren eine gegenseitige Authentifizierung der Zahlungsinstanzen (Chipkarte, bzw. Software-Klient) und Verschlüsselung bzw. Signaturen vorgesehen sind, ist ein Eingriff in den Zahlungsstrom durch einen Dritten ziemlich ausgeschlossen.

Dies schützt aber nicht davor, daß ein Angreifer sich als der Händler ausgibt, und die Zahlungen seinem eigenen Konto gutschreiben läßt. Dies kann prinzipiell sogar bei der Geldkarte geschehen. Ist dort ein Angreifer im Besitz einer Händlerkarte, so kann er sich bei einem Händler anstellen lassen und dort seine eigene Chipkarte und u.U. sein eigenes Terminal benutzen. Wird er nicht auf frischer Tat ertappt, hat es der Händler im Nachhinein schwer, den Schuldigen haftbar zu machen.

Dem könnte man nur vorbeugen, wenn Zertifikate verwendet würden, die neben dem öffentlichen Schlüssel des Gegenüber auch dessen Identität als beschreibenden Text enthielten (z.B. „Kaufhaus XYZ Mönckebergstr. 4711, Hamburg, Deutschland“). Dieser Text müßte dem Kunden vor jeder Zahlung angezeigt werden, so daß dieser die Möglichkeit hat, zu entscheiden, ob dies wirklich derjenige ist, den er vor sich zu haben glaubt.

Da dies bisher so nirgendwo realisiert ist, bleiben nur die selben Mechanismen, die einen Kunden davor schützen, daß ein Händler nicht liefert. Schließlich ist er der einzige, der die nötigen Daten hat, um den Schuldigen zu überführen. Auch wird er vermutlich als erster den Angriff bemerken, da der Angreifer in der Regel seinen Gewinn nicht durch die tatsächliche Lieferung von Waren schmälern wird.

8.4.3.4 Zusammenfassung

Prinzipiell bieten alle Verfahren einen guten Schutz vor der Erzeugung von digitalem Falschgeld.

Sollte es aber doch gelingen, so wird bei NetCash sichergestellt, daß sich dadurch die Geldmenge im System nicht erhöht. Bei der Geldkarte wird immerhin noch sichergestellt, daß eine Erhöhung der Geldmenge im System nicht direkt eine Erhöhung der Geldmenge in der physischen Welt erlaubt.

Während alle Verfahren verhindern, daß eine Zahlung ohne das prinzipielle Einverständnis des Kunden stattfindet, ist es dem Händler bei Mondex und der Geldkarte prinzipiell möglich, den Kunden über die Höhe der Zahlung zu täuschen.

Um den Kunden zu schützen, falls eine Ware nicht geliefert wurde, erlaubt es ecash, die Anonymität einer Zahlung im Nachhinein aufzuheben. NetCash kennt ein optionales Protokoll, um die Ausstellung einer Quittung zu erzwingen, das in der betrachteten Implementation aber nicht zum Einsatz kommt. Bei der Geldkarte besteht prinzipiell die Möglichkeit, Transaktionsdaten zu reidentifizieren, auch wenn es bisher keine Stellungnahme der Kreditinstitute gibt, die die Bereitschaft signalisiert, dies auch tatsächlich zu tun.

Ebenfalls gemeinsam ist allen Verfahren, daß sie guten Schutz gegen Eingriffe Dritter in den Datenstrom bieten, einzig die betrachtete Implementation von NetCash macht diesen Schutz optional.

Leider wurde dabei nicht bedacht, daß ein Kunde im Internet nicht sicher sein kann mit wem er spricht. Wenn man dies auch CAFE und der Geldkarte nicht anlasten kann, da sie in erster Linie für POS - Geschäfte gedacht sind, muß man dies bei Mondex schon bedauern, da diese Firma Werbung damit macht, daß ihr System in naher Zukunft auch für das Internet verfügbar sei. Bei NetCash und ecash stellt diese Schwäche schließlich einen echten Mangel dar.

Um zum Schluß noch eine Rangliste zu geben, möchten wir darauf hinweisen, daß diese nur grob differenziert sein kann, will man nicht mit expliziten Prozentpunkten für die Erfüllung obiger Kriterien arbeiten. Es ließe sich z.B. trefflich darüber streiten, ob NetCash (mit Erzwingung von Quittungen) oder ecash der erste Rang gebührt. Auch die Tatsache, daß NetCash ohne Verschlüsselung immer noch der Geldkarte gleichgestellt wird, wird nur dann verständlich, daß für die Integrität der prinzipielle Schutz vor der Erzeugung von echtem Falschgeld[6] und der daraus resultierende Schutz vor einer Erhöhung der Geldmenge im System als genauso wichtig angesehen wurde, wie der Schutz des Kunden durch die gesicherte Übertragung bei den anderen Verfahren.

1. NetCash (mit PGP), ecash
2. Geldkarte, CAFE, NetCash (ohne PGP)
3. Mondex

8.4.4 Verläßlichkeit

8.4.4.1 Transportverläßlichkeit

Dieses Kriterium wird bei der Geldkarte gegenstandslos, da sie direkt mit dem Terminal verbunden ist. Bei Mondex und CAFE ist eine Aussage nicht möglich, da die entsprechenden Protokolle nicht veröffentlicht wurden.

Bei NetCash und ecash besteht prinzipiell die Gefahr, daß auf Grund eines Netzwerkfehlers Unklarheit darüber besteht, wer tatsächlich im Besitz einer Münze ist.

Betrachten wir zuerst ecash. Glauben beide Parteien im Besitz der Münze zu sein, kommt es darauf an, wer sie zuerst einlöst. Geschieht dies durch den Händler, so wird der Kunde bei seinem nächsten Einkauf bemerken, daß seine Münze ungültig ist. Hat der Händler die Ware versand, so ist kein Schaden entstanden. Anderenfalls bleiben dem Kunden nur die oben besprochenen Maßnahmen gegen betrügerische Händler.

Löst andererseits der Kunde die Münze zuerst ein, so wird der Händler beim Einlösen durch die Bank abgewiesen werden. Da er zu diesem Zeitpunkt noch keine Waren versand hat, ist ihm kein Schaden entstanden.

Bliebe noch der Fall, daß beide glauben, der jeweils andere hätte die Münze. In diesem Fall käme es zu einem Disput, der vermutlich in einer Aufdeckung der Blinding-Faktoren resultieren würde. Anschließend wäre es möglich, wieder einen konsistenten Zustand herzustellen.

Eine andere Möglichkeit wäre die Rekonstruktion der letzten 16 Abhebungen und die Einlösung der resultierenden Münzen, die für Ausfälle des Kundenrechners vorgesehen ist. Da der Händler die Münze nicht einglöst

[6]Geld kann zwar gestohlen, aber nicht neu erzeugt werden, da dem Währungsserver die Nummern aller ausgegebenen Scheine bekannt sind.

hat, ist der der Münze zugeordnete Wert hinterher wieder im Besitz des Kunden.

Tatsache ist, daß ein Netzausfall eine inkonsistente Situation herstellen kann, deren Auflösung dem Kunden obliegt. Damit ist es prinzipiell möglich, daß digitales Geld vernichtet wird.

Auch bei NetCash sieht die Problematik ähnlich aus. Hier ist der einzige Schutz des Kunden die optionale Möglichkeit, den Händler zur Ausgabe einer Quittung zu zwingen. Hierbei wird eine dreigeteilte Münze verwendet, deren erster Teil dem Händler geschickt wird und von diesem innerhalb einer bestimmten Zeit einzulösen ist. Erhält der Kunde in dieser Zeit keine Quittung, so sendet er den zweiten Teil an den Server und erhält von diesem entweder sein Geld zurück oder eine Quittung, die auch den öfentlichen Schlüssel des Händlers enthält.

Dieses Verfahren erlaubt es sicherzustellen, daß ein konsistenter Endzustand hergestellt wird. Voraussetzung ist allerdings, daß es dem Kunden möglich ist, durch Vorlage der Quittung seine Ware zu erhalten. Leider ist das bei der betrachteten Implementation nicht realisiert worden.

Da dieses Protokoll nur optional ist, ist es denkbar, daß es nicht eingesetzt wird. In diesem Fall ist es nicht möglich, eine Vernichtung digitalen Geldes zu verhindern. Ansonsten gilt wie bei ecash, daß es vom Benutzer abhängt, ob er dafür sorgt, daß auch tatsächlich ein konsistenter Endzustand erreicht wird.

Die Gültigkeit der ACID - Eigenschaften muß für beide Verfahren verneint werden. Es können inkonsistente Zustände auftreten, die auch die Vernichtung elektronischen Geldes zur Folge haben können.

8.4.4.2 Wirkung einer Beschädigung der Endgeräte

Wir werden hier nur die Folgen einer Beschädigung des Kunden- bzw. Händlerendgerätes besprechen. Wir gehen davon aus, daß die Bankgeräte ausreichend gesichert sind.

Die Kreditinstitute versprechen bei unverschuldeter Beschädigung der Chipkarte die Erstattung des ihr zugeordneten Betrages. Da es sich um ein Verrechnungssystem handelt, sollten dabei keine Probleme auftreten. Was allerdings geschieht, wenn ein Terminal versagt, bevor es die in ihm gespeicherten Transaktionsdaten an die Bank weitergeben kann, ist uns nicht bekannt.

Auch Mondex verspricht das Geld zu erstatten, das sich auf einer beschädigten Chipkarte befindet. Was aber geschieht, wenn die Karte so beschädigt ist, daß der auf ihr gespeicherte Wert nicht mehr ausgelesen werden kann, ist fraglich. Schließlich erlaubt Mondex den direkten Transfer digitalen Geldes zwischen den Chipkarten zweier Privatpersonen. Diese Transfers können von Mondex nicht nachvollzogen werden.

Bei CAFE wählte man eine klassische Lösung, um trotz Anonymität die Erstattung zu erlauben. Hier kann der Kunde ein verschlüsseltes Backup auf eine spezielle Chipkarte durchführen. Die darauf gespeicherten Informationen erlauben es der Bank, die Münzen zu rekonstruieren, die sich zum Zeitpunkt des Backups auf der Karte befanden. Diese werden mit den bereits eingelösten verglichen und die Differenz dem Kunden erstattet.

Dieses Verfahren dürfte bei *regelmäßiger* Erstellung von Backups relativ sicher sein. Unglücklicherweise ist dies auch sein Schwachpunkt. Durchschnittliche Anwender von Computersystemen (die Autoren dieses Textes eingeschlossen) sind in dieser Hinsicht eher nachlässig, wie eigene Erfahrungen aus der Virenberatung belegen.

In unseren Unterlagen ist kein Hinweis darauf zu finden, daß NetCash eine Beschädigung des Kundenrechners in Betracht zieht. Während der Händler nichts zu befürchten braucht, da er die Münzen direkt bei Erhalt einlöst, könnte dies für den Kunden bedeuten, daß er die auf seinem Rechner lagernden Münzen verliert.

Bei ecash können die letzten 16 Abhebungen rekonstruiert werden. Die dabei rekonstruierten Münzen werden eingelöst, wobei festgestellt wird, ob diese vorher schon benutzt wurden. Die unbenutzten Münzen werden dem eigenen Konto gutgeschrieben.

8.4.4.3 Zusammenfassung

Bei NetCash und ecash können bei einer Netzwerkstörung inkonsistente Zustände entstehen und digitale Münzen verloren gehen. Bei den POS - Systemen fehlt hierzu eine ausreichende Dokumentation. Dort ist das Problem aber auch nicht so gravierend wie im Internet.

Bei einer unverschuldeteten Beschädigung der Geldkarte wird der ihr zugeordnete Betrag ersetzt. Bei einer Beschädigung der Mondex-Karte wird dasselbe versprochen, es ist aber nicht sicher, ob dies immer eingehalten werden kann. Während CAFE mit einer Backup - Lösung arbeitet, braucht sich der Kunde bei ecash nur einmal bei der Installation seiner Software einen Wert zu merken, um in der Lage zu sein, sein Geld wiederherzustellen. Einzig bei NetCash scheint dieser Punkt nicht Gegenstand der Überlegungen gewesen zu sein.

Hier die obligatorische Rangliste:

1. Geldkarte
2. Mondex, ecash
3. CAFE
4. NetCash

8.4.5 Zurechenbarkeit

8.4.5.1 Entlarvung von Fälschern

Die Geldkarte ist per se nicht anonym gegenüber der Bank. Sollte es daher z.B. einem Kunden gelingen, seine Geldkarte so zu manipulieren, daß diese sich gewissermaßen nach jeder Transaktion wieder aufläd (u.U. durch Verwendung einer aufgezeichneten Transaktion), so kann dies in den Evidenzstellen festgestellt werden.

Wird bei Mondex eine derartig präparierte Karte gegenüber einer normalen Kundenkarte benutzt, so entstehen keine Daten, die die Bank benutzen könnte. Macht der Fälscher allerdings den Fehler, damit an einem Terminal zu bezahlen, das mit der Bank verbunden ist, so bestünde die Gefahr, daß er Spuren hinterläßt, die zu seiner Identifizierung führen könnten. In [Jones 96] wird darauf hingewiesen, daß jede Transaktion eine eindeutige Nummer hat. Sollte diese also mehrfach auftauchen, so handelt es sich um eine Fälschung. Die Beteiligten an der Transaktion können einfach ausfindig gemacht werden, da Mondex „privat" aber nicht anonym ist.

Bei CAFE ist die Identität des Kunden in die Münze in einer Weise eingearbeitet, die es erlaubt, diese widerherzustellen, sollte es dem Kunden gelingen, den Observer auszuschalten und dieselbe Münze mehrfach auszugeben. Sollte dagegen der Händler dieselbe Münze mehrfach einreichen, so ist er an Hand der eingebauten Zeitstempel überführt.

NetCash erlaubt es nicht, einen Fälscher dingfest zu machen, da die Währungsserver über die Zuordnung elektronischer Münzen zu seinen Klienten nicht Buch führt. Allerdings ist das Erzeugen von Falschgeld nicht möglich. Einzig der Diebstahl von digitalem Geld während der Übertragung ist denkbar.

Ähnlich sieht es bei ecash aus. Dort ist der Kunde völlig anonym, so daß er auch bei Fehlverhalten nicht zur Rechenschaft gezogen werden kann.

8.4.5.2 Beweis einer erfolgten Zahlung

Bei der Geldkarte ist es prinzipiell möglich, eine Transaktion zu reidentifizieren. Da es sich aber um ein POS - System handelt, wird der Kunde vom Händler eine physische Quittung erhalten, so daß dies in der Regel wohl unnötig sein wird.

Bei Mondex werden die letzten 10 Transaktionen auf der Karte gespeichert. Aber selbst wenn diese in einem Disput als Beweis annerkannt würden, besteht die Gefahr, daß der Kunde schon in 10 weiteren Geschäften eingekauft hat, bevor er das Problem bemerkt. Da Mondex auch mit der Zahlung über das Internet Werbung macht, scheint uns dieser Punkt nicht ausreichend berücksichtigt worden zu sein.

Macht ein Kunde des CAFE-Systems regelmäßig Backups, so könnte er mit Hilfe der Bank durchaus feststellen, wem er wieviel bezahlt hat. Das Verfahren ist dasselbe, das auch bei einer Beschädigung der Wallet benutzt wird, nur daß es hierbei nicht nur darauf ankommt, ob eine Münze ausgegeben wurde, sondern vor allem an wen.

Um bei NetCash einer Transaktion Kunden und Händler zuzuordnen, müßten derjenige Währungsserver, der dem Kunden Geld gegen NetCash-Münzen und derjenige, der es zurückwechselte, zusammenarbeiten. Da aber ein Währungsserver NetCash-Münzen prinzipiell in NetCash-Münzen statt in Geld umwandeln kann, ist es prinzipiell denkbar, daß Münzen über mehr als zwei Zwischenstationen laufen. In diesem Fall sind nur die erste und die letzte Station überhaupt reidentifizierbar, was einer Anonymität schon recht nahe kommt. Hinzu kommt, daß die Währungsserver idealerweise nicht darüber Buch führen, wer physisches Geld in digitales umwandelt. Ein Beweis einer erfolgten Zahlung ist damit nicht möglich.

Bei der Implementation der NetBank kommt noch hinzu, daß die Umwandlung von physischem Geld in NetCash und das Wechseln von Münzen nur den Besitz einer Email-Adresse voraussetzt. Da diese recht einfach zu erhalten sind, ohne daß dabei eine sichere Zuordnung zu einer physischen Person gemacht wird (AOL/Compuserve - CD's, Mailboxen, anonyme Remailer, etc.), ist es möglich, digitales Geld zu erzeugen, mit diesem einzukaufen, und sich mit ihm Bezahlen zu lassen, ohne daß dies hinterher nachvollzogen werden kann.

Bei ecash kann schließlich die Anonymität einer Zahlung nachträglich aufgehoben werden, womit der Beweis bei Kooperation der Bank leicht erbracht werden kann.

8.4.5.3 Authorisierung von Zahlungen

Bei ecash, NetCash und Cafe werden Zahlungen vom Kunden initiiert, der sich über die Höhe des Betrages im Klaren ist.

Mondex und die Geldkarte erfordern, daß der Kunde dem Händler die Karte aushändigt. dies mag man zwar als eine Authorisierung des Zahlungsvorganges sehen, damit wird aber nicht die Höhe der Zahlung authorisiert. Prinzipiell sind dafür, den Kunden über die Höhe des Betrages zu täuschen, nur handwerkliche Fähigkeiten nötig (Bau eines falschen Terminals mit einem Display, das zu niedrige Beträge anzeigt).

8.4.5.4 Gegenseitige Authentisierung

Bei ecash und Cafe findet keine gegenseitige Authentisierung statt, da der Kunde anonym ist. Auch der Händler kann den Kunden prinzipiell über seine Identität täuschen.

Bei NetCash findet keine Authentisierung statt. Einzig wenn die verwendeten öffentlichen Schlüssel zertifiziert wären und eine Angabe der Identität ihres Besitzers enthielten, wäre dies möglich. Dies ist aber weder in der Implementation der NetBank gegeben, noch wird in [MeNe 93a] die Verwendung von Zertifikaten auch nur erwähnt.

Bei Mondex und der Geldkarte authentisieren sich die Karten gegenseitig. Wer sie in der Hand hält, wird dabei nicht festgestellt.

8.4.5.5 Mißbrauch durch das organisierte Verbrechen

Ecash ist für das Tätigen dunkler Geschäfte denkbar ungeeignet. Der Händler ist nicht anonym und müßte immer damit rechnen, daß einer seiner Kunden verhaftet wird und als Kronzeuge gegen ihn auftritt. Nach Bekanntgabe der Blinding-Faktoren hätte die Polizei stichfeste Beweise gegen ihn.

Auch Geldwäscher riskieren, bei der Einzahlung ihres „schmutzigen Geldes“ von der Bank an ihren ungewöhnlichen Verhaltensmustern erkannt zu werden.

Solche Monitoring-Systeme sind bei der Geldkarte und Mondex ebenfalls Teil des Gesamtkonzeptes. Auch eine reale Implementation von Cafe wird sicherlich so einen Mechanismus beinhalten. Bei der Geldkarte kommt noch hinzu, daß auch der Kunde in einer Transaktion der Bank bekannt ist. Dies macht das System für dunkle Mächte mit Sicherheit uninteressant.

NetCash scheint als einziges System eine eingebaute Eignung für Geldwäsche und einen gewissen Schutz vor dem Nachvollziehen von Transaktionen durch Strafverfolgungsbehörden zu haben. Könnte man dies im Grundkonzept vielleicht noch durch die Verwendung von Zertifikaten und Monitoring der Währungsserver begrenzen, so ist die Implementation durch die NetBank dagegen gefeit. Mit ein bischen Aufwand seitens des Kunden ist man hier so anonym, als würde man Bargeld benutzen. Einzig die Begrenzung der Transaktionshöhe auf 100 $ begrenzt die Nützlichkeit des Verfahrens in dieser Hinsicht.

8.4.5.6 Zusammenfassung

Die Entlarvung von Fälschern ist sowohl bei Cafe als auch bei der Geldkarte anhand der der Bank vorliegenden Unterlagen prinzipiell möglich. Bei NetCash ist dies unnötig, da Fälschungen im strengen Sinne nicht möglich sind.

Bei Mondex entstehen keine Unterlagen, solange nicht an einem Terminal bezahlt wird, das einen Abgleich mit der Bank durchführt. Die Möglichkeiten sind damit eingeschränkt.

Ecash schließlich bietet dem Kunden uneingeschränkte Anonymität. Eine Entlarvung ist damit ausgeschlossen.

Der Beweis einer erfolgten Zahlung ist bei Digicash und der Geldkarte prinzipiell immer führbar, während er bei Cafe davon abhängt, ob der Kunde regelmäßig Backups gemacht hat.

Bei Mondex ist so ein Beweis eine eher theoretische Angelegenheit und nicht wirklich vorgesehen. Immerhin wurde Mondex genauso wie Cafe und die Geldkarte als POS-System entwickelt. In diesem Bereich kann man auch mit Recht eine Quittung auf Papier erwarten, die vor Gericht eine höhere Aussagekraft hat, als Bits und Bytes auf einem Datenträger.

NetCash bildet in dieser Hinsicht das Schlußlicht. Hier gibt es zwar eine optionale Möglichkeit, Quittungen zu garantieren, aber ohne das Wissen um die Identität des Ausstellers haben sie keinen praktischen Wert. Folgerichtig fehlt diese Möglichkeit in der konkreten Implementation.

Sowohl bei Digicash, NetCash als auch Cafe ist sichergestellt, daß der Kunde einer Zahlung so wie deren Höhe zustimmt, bevor diese stattfinden kann. Bei Mondex und der Geldkarte ist es dem Händler dagegen möglich, den Kunden über die tatsächliche Höhe einer Zahlung zu täuschen.

Eine gegenseitige Authentisierung findet nur bei Mondex und der Geldkarte statt. Auch dort erfolgt nur die Authentisierung der Karten untereinander. Es findet keine Authentisierung der Personen die sie benutzen statt. Auch wird dem Kunden nicht mitgeteilt, auf wen die Händlerkarte registriert ist.

Gegen einen Mißbrauch durch das organisierte Verbrechen ist die Geldkarte wegen ihrer fehlenden Anonymität sicherlich gut gesichert. Mondex, die Geldkarte und Cafe bieten wegen der Möglichkeit der Bank, Händler und Geldausgabeautomaten zu überwachen, einen ähnlich guten Ansatzpunkt, um das Problem in den Griff zu bekommen. Einzig bei NetCash (insbesondere in der betrachteten Implementation) erschiene ein sehr hohes Risiko von Geldwäsche und dunklen Geschäften gegeben, wäre da nicht das niedrige Limit von 100$.

Wägt man die obigen Überlegungen geneinander ab, so kommt man zu folgender Rangliste:

1. Geldkarte
2. ecash, Cafe, Mondex
3. NetCash

8.4.6 Funktionalität

8.4.6.1 Transferierbarkeit

Das Mondex - System ist sowohl zum Tätigen von Geschäften mit einem Händler als auch für Zahlungen zwischen Privatpersonen ausgelegt, wobei letztere auch mehrfach stattfinden können, ohne daß die Bank involviert ist.

Ecash und NetCash erlauben es Geschäfte sowohl zwischen Privatpersonen als auch zwischen einer Privatperson und einem Händler zu tätigen. Dabei ist allerdings in jedem Schritt die Bank online involviert.

Die Geldkarte und Cafe sind ausschließlich zum Bezahlen bei einem Händler konzipiert.

8.4.6.2 Skalierbarkeit

Bei Mondex, der Geldkarte und Cafe sind die Verwendung diverser Banken und Währungen Teil des Konzeptes. Da die Zahlungen zudem offline erfolgen, sind Engpässe im Zugriff auf den Bankenserver auch bei hohem Aufkommen nicht zu erwarten.

Netcash unterstützt ebenfalls mehrere Bankenserver, die untereinander einen Abgleich ausführen. Prinzipiell erfolgen Zahlungen aber online, was bei hohem Verkehrsaufkommen zu Verfügbarkeitsproblemen führen könnte.

Ecash schließlich ist das Schlußlicht dieser Betrachtung. Es gibt zwar verschiedene Währungen, die aber nicht untereinander konvertiert werden können. Bei jeder Zahlung ist ein zentraler Server involviert, was zu Verfügbarkeitsproblemen führen könnte. Bevor diese Probleme nicht gelöst sind, muß bezweifelt werden, ob das Verfahren einer weiten Verbreitung gewachsen wäre.

8.4.6.3 Zusammenfassung

Da sich das oben gesagte nicht kürzer zusammenfassen läßt, hier nur die Rangliste:

1. Mondex
2. NetCash, Cafe, Geldkarte
3. Ecash

8.4.7 Organisatorisches und Rechtliches

8.4.7.1 Haftung beim Auftauchen von Falschgeld

Bei ecash, NetCash und Cafe haftet der Kunde, falls Falschgeld auftaucht.

Dagegen werden bei der Geldkarte die Umsätze der Händler durch die Banken garantiert.

Bei Mondex ist die Lage unklar. Insbesondere ist nicht sicher, ob die Bank in der Lage ist das Auftauchen von Falschgeld überhaupt zu bemerken. Was aber geschieht, falls die Bank in den Transaktionsdaten eines Händlers einen Hinweis auf Falschgeld findet ist nicht dokumentiert.

8.4.7.2 Rücktauschbarkeit

Dieses Kriterium ist nur bei elektronischem Kleingeld von Belang. Bei allen hier betrachteten Systemen ist die Rücktauschbarkeit gegeben.

8.4.7.3 Erstattung verlorenen Geldes

Wie dies bei Netcash geregelt wird, ist nicht dokumentiert, es ist aber davon auszugehen, daß keine Erstattung stattfinden wird. Wegen der oben angeführten Schwächen des Systems wäre die Gefahr eines Mißbrauches wohl zu groß.

Ecash und Cafe bieten technische Lösungen, die eine Rückerstattung erlauben, falls der Kunde Vorsichtsmaßnahmen trifft, während Mondex und die Geldkarte eine Rückerstattung bei Beschädigung der Karte zusagen, bei Verlust oder Diebstahl aber jegliche Haftung ausschließen.

8.4.7.4 Zusammenfassung

Das obige läßt sich nicht weiter zusammenfassen. Auch das Erstellen einer Rangliste ist im Prinzip unmöglich, da für eine Bewertung nicht genug Informationen vorliegen.

Positiv ist natürlich im ersten Kriterium anzumerken, daß bei der Geldkarte prinzipiell eine Haftung der Bank vorzuliegen scheint.

Negativ bei NetCash ist dagegen, daß dort eine Erstattung verlorenen Geldes praktisch unmöglich ist. Hier könnte man zwar wie bei Cafe mittels Backup Abhilfe schaffen, es ist aber kein Teil des Konzepts. In den vorliegenden Informationen der NetBank fehlt jeglicher Hinweis, daß z.B. für einen Plattencrash Vorsorge durch den Kunden zu treffen ist.

Im ganzen gesehen werden wir aber Abstand davon nehmen, daraus eine Rangliste konstruieren zu wollen.

Kapitel 9

Resümee

In dieser Arbeit haben die Autoren diverse Internet- und Chipkartenzahlungssysteme beschrieben, in die Gruppen „Verfahren für kleine Beträge“, „digitale Kreditkarten“, „digitale Schecks“, „digitales Bargeld“ eingeteilt und nach Kriterien der Vertraulichkeit, Integrität, Verläßlichkeit, Zurechenbarkeit, Transferierbarkeit, Skalierbarkeit, sowie unter organisatorischen und rechtlichen Gesichtspunkten bewertet.

Bei der Bewertung der Verfahren für kleine Beträge wurde deutlich, daß in Zukunft sehr wohl ein Bedarf für Verfahren besteht, die es erlauben, Waren von geringem Wert zu bezahlen, da Informationen im World Wide Web aus technischen Gründen in viele Seiten aufgeteilt werden, die einzeln vom Benutzer angefordert werden. Eine einzelne Seite kann daher im Regelfall kaum zu einem Preis verkauft werden, der den Einsatz eines der in den anderen Gruppen beschriebenen Verfahren rechtfertigen würde. Berechnet man zum Beispiel einmal den Seitenpreis einer Illustrierten wie dem auch im Internet vertretenen Stern, der bei einem Umfang von 134 Seiten 4,00 DM kostet, so kommt man auf nur 3 Pfennig.

Millicent als einziges Verfahren, daß tatsächlich für so kleine Beträge, ja sogar für „Bruchteile eines Cent“ ausgelegt ist, fiel aber in der Bewertung durch. CyberCoin als überzeugenster Kandidat in der Bewertung ist nach Angaben des Herstellers für Beträge zwischen 0.25 \$ und 10 \$ gedacht. Inwieweit MPTP als Zweitplazierter schließlich dazu geeignet ist, den Bereich unter 25 Cents abzudecken, kann noch nicht abgesehen werden, da den Autoren noch keine Implementation vorliegt. Da es aber keinerlei Schutz der Kundenidentität und des Transaktionsinhaltes vorsieht, bleiben Zweifel, ob dies wünschenswert wäre.

Es bleibt festzuhalten, daß CyberCoin eine mit Einschränkungen (Pseudonyme statt Anonymität) akzeptable Lösung für kleine Beträge über 25 Cents darstellt. Für die Bezahlung noch kleinerer Beträge ist allerdings noch keine Lösung in Sicht.

Die Verfahren, die unter digitale Kreditkarten zusammengefaßt wurden, können durch die Bank nur als Notlösungen gesehen werden, die mit der Einführung der digitalen Scheckverfahren und des digitalen Bargeldes jede Existenzberechtigung verloren haben. Welchen Sinn hat schließlich ein Verfahren, das Angriffe Außenstehender einigermaßen abwehren kann, wenn es schon Verfahren gibt, die auch Betrug seitens des Händlers ausschließen können?

Die digitalen Schecks verdanken ihre guten Zukunftsaussichten nicht zuletzt der Tatsachen, daß SET als das vielversprechenste Verfahren von Visa und Mastercard entwickelt wird. Aber auch vom Standpunkt des Kunden gibt es Gründe, die für die Verwendung eines Scheckverfahrens auf Kreditkartenbasis sprechen. Zumindest im oberen Preisegment kann die zusätzliche Sicherheit, die sich aus der mangelnden Anonymität der Verfahren ergibt, von Vorteil sein. Anhand der Kreditkartenabrechnung kann gerichtsfest bewiesen werden, daß eine Zahlung stattgefunden hat. Außerdem bedeutet eine Abrechnung über Kreditkarte, daß der Kunde nicht sofort bezahlen muß, was auch in der physischen Welt zur Verbreitung der Kreditkarten beigetragen hat.

Ob diese Vorteile auch bei NetCheque gegeben sein werden, müßte geprüft werden, wenn eine Implementation tatsächlich auf breiter Basis verfügbar sein wird. Das Fehlen echter Signaturen und der mangelnde Datenschutz lassen NetCheque allerdings nicht als ernsthafte Konkurenz für SET erscheinen.

Leider sind die aufgeführten Vorteile auch von Nachteilen begleitet. Es ist nicht praktikabel, ein Kreditverfahren mit wirklicher Kundenanonymität zu implementieren. Da die Banken dem Käufer Geld leihen, haben sie ein legitimes Interesse, seine Identität zu kennen. Dies bedeutet nicht, daß diese Information auch dem Händler zur Verfügung stehen müssen. Ihm genügt es zu wissen, daß er sein Geld erhalten wird. Leider wurde diese Forderung nur bei CyberCash berücksicht. Dieser Vorteil wird aber bei der Umstellung auf SET verloren gehen.

Während bei hohen Beträgen die Vorteile der Scheckverfahren die Nachteile überwiegen, sieht die Situation im mittleren Preissegment anders aus. In diesem Bereich spielt Anonymität eine deutlich wichtigere Rolle, da die Anzahl der Transaktionen deutlich höher, die Gefahr der Profilbildung deutlich größer ist. Hinzu kommt, daß auch das Sicherheitsbedürfnis zwar wichtig, aber nicht so bedeutend wie bei großen Beträgen ist. Wo die Grenze liegt, wird ein Kunde zwar selbst entscheiden müssen, eine Faustregel wäre aber vielleicht, daß im dreistelligen Bereich der Verlust eines übermittelten Betrages so schmerzhaft wäre, daß alles getan werden muß, um ihn zu verhindern.

Anonymität kann aber nur durch Debitverfahren gewährleistet werden. Hier kommen die Bargeldverfahren ins Spiel. Sie könnten all das bieten, was wir auch vom physischen Bargeld kennen:

- Anonymität
- die Möglichkeit, Geld an Privatpersonen weiterzugeben
- die Möglichkeit, erhaltenes Geld sofort wieder auszugeben

Leider bestehen auch die selben Risiken wie bei Bargeld und müssen durch entsprechende technische Maßnahmen begrenzt werden:

- Verlust
- Diebstahl
- Fälschung
- Mißbrauch der Anonymität durch Geldwäscher und Schwarzarbeiter

Bei der Abwägung von Sicherheit und Datenschutz setzen die bewerteten Verfahren unterschiedliche Akzente. Während CAFE und ecash echte Anonymität bieten, macht es das Fehlen von Anonymität NetCash es möglich, das Erhöhen der Geldmenge im System zu verhindern. Auch bei der Geldkarte wird Anonymität zu Gunsten einer besseren Überwachung durch die Banken geopfert, die es erlauben soll, Mißbrauch festzustellen und die Schuldigen zur Rechenschaft zu ziehen. Dem Händler gegenüber soll die Geldkarte allerdings anonym sein.

Mondex liegt in dieser Hinsicht im Mittelfeld. Gegenüber dem Händler ist der Kunde pseudonym, die Bank kann eine Zuordnung treffen, erfährt aber nur von einem Teil der getätigten Transaktionen. Leider werden diese Vorteile dadurch mehr als ausgeglichen, daß es einem Händler sogar möglich ist, Transaktionen auszulesen, die in anderen Geschäften getätigt wurden.

Auch die Weitergabe von Geld an Privatpersonen ist nicht bei allen Verfahren möglich. CAFE und die Geldkarte sind als POS - Systeme nicht dazu konstruiert worden, die selbe Transferierbarkeit wie physisches Bargeld zu bieten. Mit ihnen kann nur bei authorisierten Händlern bezahlt werden, die daraufhin das erhaltene digitale Geld auf der Bank einlösen müssen.

Während alle Verfahren die Möglichkeiten, Geld zu fälschen, auf verschiedene Weisen zu verhindern suchen, kann der Gefahr, daß Geld verloren geht oder gestohlen wird, nur bedingt begegnet werden. Prinzipiell ist davon auszugehen, daß der Diebstahl elektronischen Geldes möglich ist[1] und daß die dabei entstehenden Verluste vom Kunden getragen werden müssen.

Geht dagegen durch höhere Gewalt Geld verloren (Plattencrash, Beschädigung einer Chipkarte), so werden zumindest bei einigen Verfahren Maßnahmen zur Schadensbegrenzung angeboten. Bei ecash reicht es, bei der Installation eine Zahl zu notieren, während CAFE ein regelmäßiges Backup benötigt. Mondex und die Geldkarte lehnen Erstattung bei Verlust der Chipkarte ab, sagen aber eine Erstattung bei Beschädigung der selben zu. NetCash schließlich bietet keinen Schutz.

Es bleibt festzuhalten, daß zum Bezahlen im Internet mit ecash ein Verfahren existiert, das es erlaubt anonym zu bezahlen und bei dem die Risken gering sind. Im POS - Bereich wäre ein Verfahren ideal, das die Transferierbarkeit von Mondex und die Anonymität von CAFE vereint. Dazu wäre es sicherlich ein guter Anfang, den Zugang zum Transaktionslog mit einer eigenen PIN zu schützen. Auch die fehlende Möglichkeit, mit einem eigenen Gerät den abzubuchenden Betrag zu authorisieren, sollte schleunigst nachgebessert werden. Bevor dieses Verfahren auch im Internet als sicher betrachtet werden kann, wäre es auch noch nötig, dem Kunden vor Beginn des Geldtransfers verläßlich mitzuteilen, wer das Geld erhalten wird. Auf diese Weise wäre es möglich zu verhindern, daß der Kunde über die Identität des Händlers getäuscht und so um sein Geld erleichtert wird, ohne seine Ware zu erhalten. Diese Verbesserungen ließen sich vermutlich ohne allzugroße Eingriffe in das System realisieren. Anonym wird es dadurch jedoch noch nicht. Eine Verbesserung in dieser Hinsicht ist nicht zu erwarten.

Zusammenfassend kann gesagt werden, daß sowohl die Verfahren für kleine Beträge, digitale Schecks und digitales Bargeld ihre Berechtigung haben.

In Tabelle 9.1 haben die Autoren noch einmal diejenigen Verfahren zusammengestellt, die am ehesten geeignet erscheinen, die aufgestellten Anforderungen zu erfüllen. Die Tatsache, daß Mondex aufgeführt ist, die Geldkarte aber nicht, liegt darin begründet, daß Mondex eine höhere Funtionalität bei

[1] CAFE, Mondex und die Geldkarte verwenden Chipkarten, die physisch gestohlen werden können. NetCash und ecash verwenden Nummern, die ausgespäht oder erraten werden können.

Preis	Anwendungsgebiet	Verfahren
0 - 30 Pfennig	Web - Seiten,	–
30 Pf. - 15 DM	Multimedia	CyberCoin
15 - 100 DM	Online Dokumente, physische Waren	ecash CAFE (POS) Mondex (nach Modifikation)
über 100 DM	physische Waren	SET

Die Grenze von 100 DM ist willkürlich gewählt und hängt von der Einschätzung des Kunden ab.

Tabelle 9.1: Übersicht der geeigneten Zahlungsverfahren und ihrer Einsatzgebiete

geringerer Überwachung durch die Banken bietet. Ohne die angemahnten Modifikationen kann es allerdings nicht guten Gewissens empfohlen werden.

Für die Zukunft wäre es allerdings wünschenswert, daß stärker auf die Anonymität des Kunden Wert gelegt wird. Es ist die Auffassung der Autoren, daß die Gefahr einer aussagekräftigen Profilbildung um so größer ist, je geringer der Betrag der Einzeltransaktion ist.

Daraus ergibt sich, daß im Bereich der kleinen und kleinsten Beträge Verfahren nötig wären, die absolut anonym sind. CyberCoin kann dies nicht leisten, hat aber den Vorteil daß ein einwandfreier Erhalt der Ware gesichert wird. Immerhin werden gegenüber dem Händler Pseudonyme verwendet.

Auch im Bereich der mittleren Beträge existieren mit ecash und CAFE sowohl ein Internet - als auch ein POS - System, das Anonymität gewährleistet. Es muß aber auch gesagt werden, daß beide Mondex in Teilaspekten unterlegen sind. CAFE bietet nicht die gleiche Transferierbarkeit, während bei ecash die Skalierbarkeit bei breitem Einsatz des Systems zum Problem werden könnte, da es von der Verfügbarkeit eines zentralen Servers zum Zeitpunkt der Transaktion abhängt. Ein System, daß alle diese Aspekte vereint wäre ideal.

SET schließlich scheint auf den ersten Blick durchaus durchdacht und dem Zweck angemessen. Es ist aber völlig unnötigt, daß der Händler die Identität des Kunden erfährt. Diese Information sollte ebenfalls verschlüsselt werden und nur dem Payment Gateway bekannt sein. Zum gegenwärtigen Zeitpunkt steht einer Profilbildung durch den Händler nichts im Weg.

Anhang A

Grundlagen der Kryptographie

Im folgenden werden wir versuchen, einige der kryptographischen Grundlagen zu erklären, die wir in der eigentlichen Arbeit vorausgesetzt haben. Dabei können wir natürlich nicht allzusehr in die Tiefe gehen. Dem interessierten Leser seien [Schneier 96], [Bauer 95] und [Kahn 67] als weitergehende Lektüre empfohlen.

A.1 Begriffe

A.1.1 Codes und Chiffren

In der Welt der Kryptologie wird heutzutage selten der Begriff Code verwendet. Klassischerweise versteht man darunter ein Verfahren, durch das Wörter und Phrasen mittels eines Codebuches in kurze Zeichengruppen umgewandelt werden. Ein Codebuch ist dabei wie ein Lexikon, in dem jeweils für ein Wort oder einen Satz eine Zeichenkette steht.

Solche Codebücher waren sehr aufwendig zu erstellen und konnten nur schwer gewechselt werden. Daher „überschlüsselte“ man das Ergebnis oft mit einem Verfahren, das einzelne Buchstaben oder kurze Zeichenketten fester Länge[1] in Abhängigkeit von einem Schlüssel in andere Buchstaben oder Zeichenketten umsetzt. So ein Verfahren nennt man Chiffre.

[1]D.h. die Zeichenketten, die man in einem Schritt verschlüsselt haben eine feste Länge. Natürlich will man ganze Texte verschlüsseln, deren Längen nicht von Anfang an feststehen.

Die Chiffren der damaligen Zeit waren nicht besonders gut, und bis nach dem ersten Weltkrieg konnte ein erfahrener Kryptograph so ziemlich jede Chiffre brechen, ohne dazu mehr als etwas chiffrierten Text, ein paar Bleistifte und viel Papier zu brauchen.

Dies änderte sich mit den maschinellen Chiffren des zweiten Weltkriegs. Mit ihnen wurde der Aufwand deutlich größer, und der Traum von der unknackbaren Verschlüsselung begann langsam Gestalt anzunehmen. Als Folge davon sind Codebücher inzwischen aus der Mode gekommen, und die Chiffren – insbesondere computergestützte – haben ihren Siegeszug angetreten.

So wäre es nun eigentlich egal ob man eine Verschlüsselung einen Code oder eine Chiffre nennt. Da aber die Kryptographen der alten Schule auf dieser Unterscheidung bestehen, haben wir versucht, die Begriffe im Text sauber zu trennen.

Daher werden wir den Begriff „Codieren" nur im Sinne der Nachrichtentechnik verwenden, d.h. als Ausdruck einer Idee durch ein Zeichen oder eine Zahl. So wird z.B. die Idee „Leerzeichen" im ASCII - Code als 32 codiert, ohne daß damit gemeint wäre, daß niemand wissen dürfe, daß eine 32 für „Leerzeichen" steht.

A.1.2 Hashverfahren

Hashverfahren werden in der Informatik schon eine ganze Weile benutzt. Es handelt sich dabei um Funktionen, die eine Eingabe beliebiger Größe auf Zahlen abbilden, deren Größe begrenzt ist.

Sie werden dazu verwendet, Zeichenketten unbegrenzter Länge in Tabellen fester Länge einzuordnen. Dabei ist es wichtig, daß ein errechneter Tabellenindex nicht außerhalb der Tabelle liegt, und daß möglichst selten zwei Eingaben auf den selben Tabellenplatz zu liegen kommen, was eine aufwendige Sonderbearbeitung erfordert.

In der Kryptographie werden derartige Funktionen oft in einer ähnlichen Art und Weise benutzt wie Fehlererkennungscodes in der Nachrichtentechnik. Dazu müssen sie aber neben den oben genannten Eigenschaften auch sicherstellen,

1. daß eine kleine Änderung der Eingabe eine deutliche Änderung des Ausgabewertes nach sich zieht, so daß schon die Änderung eines Bit erkannt werden kann.

2. daß es nicht praktikabel (d.h. nicht mit realistischem Aufwand berechenbar) ist, eine zweite Eingabe zu erzeugen, die sich von der ersten

unterscheidet, aber den selben Hashwert ergibt[2].

Prominente Vertreter kryptographischer Hashverfahren sind der **MD5** (Message Digest 5), der im April 1992 als RFC1321 veröffentlicht wurde und der relativ neue **SHA**, der am 31. Mai 1994 als Draft vom amerikanischen National Institute of Standards and Technology (NIST) veröffentlicht wurde. Während der MD5 eine 16 Byte Ausgabe erzeugt, liefert der SHA sogar 20 Byte.

A.1.3 Symmetrische und asymetrische Verschlüsselungen

Unter symmetrischen Verfahren versteht man jene, die ein und denselben Schlüssel sowohl zum Chiffrieren als auch zum Dechiffrieren benutzen. Die meisten der klassischen Verfahren beruhen auf diesem Prinzip.

Dabei besteht allerdings das Problem, daß dieser Schlüssel geheim gehalten werden muß. Das bedeutet, daß ein sicherer Kanal für seine Übertragung existieren muß, und daß möglichst wenig Personen den Schlüssel kennen.

Im Idealfall gibt es für je zwei Kommunikationspartner einen Schlüssel. Damit wären für n Personen $n!$ Schlüssel nötig. 10 Personen brauchen damit 3.628.800 Schlüssel, was die Generierung und Verteilung zu einem Albtraum macht.

Bei einem asymetrischen Verfahren – auch Public Key Verfahren genannt – gibt es dagegen zwei Schlüssel, einen zum Verschlüsseln und einen zum Entschlüsseln. Kennt ein Außenstehender einen der beiden Schlüssel, so ist es ihm (technisch)[3] nicht möglich daraus den anderen zu generieren.

Diese Eigenschaft ermöglicht es, den Schlüssel zum Chiffrieren in einer Art Telefonbuch öffentlich bekannt zu machen. Ist dies geschehen, so kann jeder, der dem Besitzer der Schlüssel eine Nachricht zu senden wünscht, diese mit dem öffentlichen Schlüssel chiffrieren. Die so verschlüsselten Nachrichten können dann nur noch vom Besitzer der Schlüssel mit dem geheimen Schlüssel dechiffriert werden.

Damit entfällt das Problem der geheimen Übertragung. Es muß nur noch sichergestellt werden, daß das Telefonbuch nicht von Angreifern verändert

[2]Diese Eigenschaft spielt bei den Fehlererkennungscodes der Nachrichtentechnik normalerweise keine Rolle, da es dort nur um zufällige Veränderungen von Nachrichten durch Leitungsstörungen, nicht aber um willkürliche Veränderungen durch Dritte geht.

[3]In der Kryptographie muß man oft damit leben, daß Sicherheit nicht bedeutet, eine Chiffre sei unknackbar, sondern sie ist nicht „mit vertretbarem Aufwand“ knackbar. Leider ist nur ein theoretisches Verfahren bekannt, das beweisbar unknackbar ist. Dieses ist aber für die Praxis in der Regel nicht zu gebrauchen. Außerdem handelt es sich um ein symmetrisches Verfahren mit allen aufgeführten Nachteilen.

werden kann. Auch die Anzahl der benötigten Schlüssel sinkt dramatisch. Pro Person sind nur noch zwei Schlüssel nötig. Unsere 10 Kommunikationsteilnehmer aus dem vorigen Beispiel kämen damit mit 20 Schlüsseln aus.

A.1.4 Digitale Signaturen

Hatten wir uns bisher mit dem Problem beschäftigt, wie eine Nachricht gegen Einsichtnahme durch Dritte geschützt werden kann, so kommen wir nun zu der Frage, wie eine Nachricht gegen Veränderungen durch potentielle Angreifer geschützt werden kann. Außerdem soll sichergestellt werden, daß die Nachricht auch tatsächlich von dem Absender geschickt wurde, von dem sie zu sein behauptet.

Mit einem symmetrischen Verfahren bleibt uns nur übrig, mit dem Absender einen geheimen Schlüssel zu vereinbaren und beim Empfang zu prüfen, ob die Nachricht wirklich mit diesem Schlüssel chiffriert wurde. (Andernfalls erhalten wir nur unverständliche Zeichen).

Dieses Verfahren hat zwei gravierende Nachteile:

1. Es tut mehr als es soll. Der Text wird chiffriert und kann damit von Dritten nicht mehr gelesen werden. Dies könnte bewirken, daß das Verfahren unter Exportbeschränkungen fällt, wie sie in den USA und Frankreich gelten.

2. Es ist nicht möglich einem Dritten (z.B. einem Richter) zu beweisen, wer die Nachricht gesendet hat. Wenn der vermeintliche Sender bestreitet, die Nachricht geschickt zu haben, bleibt die Möglichkeit offen, daß der vorgebliche Empfänger die Nachricht selber erzeugt hat, um den Anderen in Mißkredit zu bringen.

Während man das erste Problem dadurch lösen kann, daß man nur einen Hashwert der Nachricht kodiert, ist das zweite Problem nicht mit symmetrischen Verfahren zu lösen. Eine echte digitale Signatur, die in der digitalen Welt das erreicht, was in der physikalischen Welt die eigenhändige Unterschrift gewährleistet, ist damit nicht zu realisieren.

Anders sieht es aus, wenn man zu asymetrischen Verfahren übergeht. Wir veröffentlichen hierbei nicht den Schlüssel zum Chiffrieren, sondern den zum Dechiffrieren. Wenn wir nun wie gehabt einen Hashwert der Nachricht verschlüsseln, so kann ihn jeder, der den öffentlichen Schlüssel besitzt, entschlüsseln und mit einem Hashwert, den er selber über die Nachricht gebildet hat, vergleichen. Stimmen beide überein, so kann er sicher sein, daß die Nachricht von uns kommt und nicht unterwegs verändert wurde.

A.2 DES

A.2.1 Geschichte

Der DES (Data Encryption Standard) oder DEA (Data Encryption Algorithm) wurde 1977 vom National Bureau of Standards (NBS)[4] in den USA zur Verwendung für „unclassified computer data“ genormt. 1981 wurde er auch als ANSI[5] Standard unter der Nummer X3.92 als DEA eingetragen. 1986 entschied dann die ISO[6] den DES nicht zum internationalen Standard zu erklären. Diese Entscheidung war deshalb so ungewöhnlich weil sie zu einem derart späten Stadium des Standardisierungsprozesses getroffen wurde wie in keinem anderen Fall in der Geschichte der ISO.

1988 wollte die NSA[7] die Zulassung des DES rückgängig machen, da man dort der Meinung war, nach 11 Jahren entspreche er nicht mehr den aktuellen Sicherheitsanforderungen. Allgemeine Kritik, insbesondere von Seiten der Banken, die den DES als Grundlage für ihre Sicherheitskonzepte benutzen, brachte die NSA dann aber doch dazu, ihre Entscheidung zu revidieren. Daher wird der DES auch heute noch dazu benutzt, Eurocheque PIN's zu generieren oder UNIX - Passworte zu überprüfen.

A.2.2 Eigenschaften

Der DES ist ein symmetrisches Verfahren, welches aus einer Eingabe von 8 Byte und einem Schlüssel von 56 Bit (8 Byte, denen das Paritätsbit entfernt wurde) eine Ausgabe von 8 Byte erzeugt. Die Schlüssellänge wurde u.a. im Januar 1996 in einem Papier[8] einer Gruppe von führenden Kryptographen unserer Zeit als zu kurz bezeichnet.

Man kam zu der Überzeugung der DES sei

> *„unzulänglich gegen einen Angreifer aus der Wirtschaft oder einer Regierungsbehörde, der [diesem Zweck] ernstzunehmende Resourcen widmet. [...] Eine ernstzunehmende Anstrengung – in*

[4] legt die Standards für Regierungsorganisationen fest.

[5] entspricht der deutschen DIN

[6] internationales Standardisierungsgremium

[7] eigentlich ein Geheimdienst, aber auch dafür zuständig, die Sicherheit von Verschlüsselungsalgorithmen zu bewerten

[8] Matt Blaze, Whitfield Diffie, Ronald L. Rivest, Bruce Schneier, Tsutomu Shimomura, Eric Thomson, Michael Wiener, „Minimal Key Lengths for Symmetric Ciphers to provide Adequate Commercial Security“, January 1996

> *der Größenordnung von 300.000$ – durch eine legitime oder illegitime Firma könnte einen DES Schlüssel im Durchschnitt in 19 Tagen finden, wenn sie handelsübliche Technologie verwendet und bloß in 3 Stunden, wenn sie einen speziell gefertigten Chip verwendet. In letzterem Fall würde es sie 38$ kosten, einen Schlüssel zu finden (wiederum eine dreijährige Lebensspanne des Chips und ständige Verwendung angenommen). “*

Angesichts dieser Bewertung und der Komplexität des Verfahrens erscheint es uns nicht nötig die Funktionsweise des DES hier näher zu erläutern. Für unsere Zwecke reicht es aus, den DES als eine „black box“ anzusehen, in die Klartext und Schlüssel hineingehen und aus der der Schlüsseltext herauskommt. Dabei soll noch angemerkt werden, daß das Verfahren sicherstellt, das jedes Bit des Schlüsseltextes von jedem Bit des Schlüssels und jedem Bit des Klartextes abhängt. Minimale Änderungen der Eingaben bewirken eine maximale Änderung der Ausgabe („Lawineneffekt“).

A.2.3 Betriebsmodi

Das National Bureau of Standards normte seinerzeit 4 Betriebsmodi für den DES:

ECB Electronic Code Book Im ECB mode wird der Klartext in 8 Byte Blöcke aufgeteilt, die dann jeweils mit dem selben Schlüssel chiffriert werden. Dieser Modus wird im allgemeinen nicht empfohlen.

CBC Cipher Block Chaining In diesem Modus wird der Klartext mit einem 8 Byte Block xor'ed, bevor er mit dem DES chiffriert wird. Hierzu beim ersten Block der sogenannte Initialisierungsvektor zum Einsatz, der oft nur aus Nullbytes besteht. Alle weiteren Blöcke werden jeweils mit dem Chiffrat des Vorgängers xor'ed. Dieses Verfahren stellt sicher, daß das Chiffrat eines jeden Blocks von allen vorherigen Blocks abhängt.

OFB Output FeedBack, CFB Cipher FeedBack Im Gegensatz zu den vorherigen Modi sind die nun folgenden nicht Block- sondern Zeichenorientiert. Üblicherweise besteht ein Zeichen aus 1 bzw. 8 Bit. Dabei wird in jedem Schritt ein Klartextzeichen mit einem Schlüssel xor'ed. Der benötigte Schlüsselstrom wird mit Hilfe des DES erzeugt. Dieser verschlüsselt den Inhalt eines 64 Bit Schieberegisters. Von der Ausgabe werden die linken (obersten) Bits als Schlüssel für das Xor genommen. Vor der nächsten Runde wird ein Zeichen von rechts das Schieberegister eingelesen und so ein neuer Schlüssel erzeugt. Beim OFB ist dieser Wert das Ergebnis des DES, beim CFB das Ergebnis des Xor.

Von diesen Modi ist allerdings der CBC der wichtigste. OFB und CFB werden in der Regel nicht eingesetzt. Der eigentliche Schlüssel für den DES bleibt übrigens in allen 4 Modi für die gesammte Sitzung derselbe. Eben daher sind die Modi nötig, die für eine Variation der Verschlüsselung über einen längeren Text sorgen. Den Text einfach nur in 8 Byte Blöcke aufzuteilen und dann jeden Block mit dem selben Schlüssel zu chiffrieren, wie dies im ECB Modus geschiet, bietet einem potentiellen Angreifer zuviel Daten um halbwegs sicher zu sein.

A.2.4 MAC

Die MAC (Message Authentication Code) ist ein Versuch etwas ähnliches wie die digitalen Signaturen mit dem DES zu erreichen. Das generelle Vorgehen wurde schon in A.1.4 beschrieben, allerdings wird hier der DES nicht nur zur Verschlüsselung sondern auch gleichzeitig als Hashfunktion eingesetzt.

Dazu wird der Text im CBC Modus verschlüsselt, wobei alle Ausgabeblöcke außer dem letzten verworfen werden. Wie schon festgestellt, hängt dieser Block von allen vorhergehenden in einer Weise ab, daß eine minimale Änderung des Textes eine völlig andere MAC erzeugen würde.

Natürlich gilt auch hier, daß dies keine vollwertige digitale Signatur ist, da zur Generierung und zur Überprüfung der MAC der selbe Schlüssel verwendet wird. Zum Schutz einer Nachricht gegen Veränderung auf einem unsicheren Kanal ist sie aber ein brauchbarer Mechanismus.

A.2.4.1 bekanntgewordene Angriffe

Die Firma RSA Data Security Inc. bot am 28. Januar 1997 jedem 10.000$, der einen Satz entschlüsseln konnte, der mit DES verschlüsselt war.

Am 17. Juni 1997 wurde der Schlüssel gefunden. Eine Gruppe namens DESCHALL unter der Leitung von Rocke Verser aus Loveland, Colorado, hatte seit 13. März ca. 70.000 Rechner im Internet benutzt, um 24% der 72 Billiarden möglichen Schlüssel zu durchsuchen, als um 22.39 Uhr PST ein Angestellter der iNetZ Corporation in Salt Lake City namens Michael Sanders auf einem 90 Mhz Pentium mit 16Mb RAM und FreeBSD 2.2.1 den

Schlüssel fand. Der Satz lautete „Strong cryptography makes the world a safer place.“

Details können auf den Seiten der RSA Inc. nachgelesen werden: http://www.rsa.com/des

A.3 RSA

A.3.1 Eigenschaften

Das RSA Verfahren ist nach seinen Entwicklern Rivest, Shamir and Adleman benannt, die es 1978 der Weltöffentlichkeit vorstellten. Es wurde am 20. September 1983 in den USA als Patent unter der Nummer 4 405 829 registriert.

Bei diesem Verfahren handelt es sich um ein Public Key Verfahren, d.h. es gibt einen öffentlichen Schlüssel und einen geheimen Schlüssel. Nachrichten, die mit dem öffentlichen Schlüssel chiffriert wurden, können nur mit Kenntnis des geheimen Schlüssels wieder dechiffriert werden.

Außerdem hat RSA den großen Vorteil, daß es egal ist, in welcher Reihenfolge die Schlüssel auf eine Nachricht angewendet werden. Daher ist es möglich, dasselbe Schlüsselpaar sowohl zum Signieren von Nachrichten als auch zum Chiffrieren zu benutzen.

Das Verfahren gilt – eine ausreichende Schlüssellänge[9] vorausgesetzt – als sicher, allerdings sind die nötigen Berechnungen zeitaufwendig. Daher verwendet man RSA im Moment nur, um einen Schlüssel für ein symmetrisches Verfahren zu chiffrieren.

A.3.2 Die Mathematik

A.3.2.1 Herleitung des Verfahrens

Das ganze Verfahren beruht auf einer Folgerung aus Euler's Theorem:

> Gegeben zwei Primzahlen p, q und zwei positive ganze Zahlen m, k mit $k, m < pq$ dann gilt :
>
> $$m^{k(p-1)(q-1)+1} \equiv m \pmod{pq}, \tag{A.1}$$

[9] Als brauchbar gelten im Moment 1024 - 2048 Bit

Dieser Satz[10] gibt uns eine komplexe mathematische Operation an die Hand, bei der am Ende wieder die Eingabe m herauskommt. Könnten wir sie in zwei Schritte aufteilen, so hätten wir eine Operation, die m in etwas völlig anderes überführte und eine, die daraus wieder m berechnete. Sind die beiden Schritte so unterschiedlich, daß es nicht möglich ist, aus dem einen den anderen abzuleiten, dann hätten wir die Basis für ein Public Key Verfahren.

Die Antwort ist verblüffend einfach, wir berechnen zwei Zahlen e, d, so daß

$$m^{ed} \equiv m^{k(p-1)(q-1)+1} \equiv m \pmod{pq} \tag{A.2}$$

Nun erklären wir $n = pq$ und e zum öffentlichen Schlüssel und teilen ihn der Welt mit, während wir d für uns behalten. Auch p und q halten wir geheim. Wir geben nur ihr Produkt bekannt.

Will uns nun jemand eine Nachricht m schicken, so berechnet er

$$m_c = m^e \pmod{n} \tag{A.3}$$

Wollen wir die Nachricht entschlüsseln, so berechnen wir

$$(m_c)^d \equiv (m^e)^d \equiv m^{ed} \equiv m \pmod{n} \tag{A.4}$$

D.h. wir potenzieren modulo n mit d, worauf wir die Nachricht m erhalten.

A.3.2.2 Generierung der Schlüssel

Wir wissen nach (A.2), daß

$$m^{ed} \equiv m^{k(p-1)(q-1)+1} \pmod{pq}$$

Dies ist äquivalent zu

$$\begin{aligned} ed &\equiv k(p-1)(q-1)+1 \pmod{pq} \\ \Leftrightarrow ed &\equiv 1 \pmod{(p-1)(q-1)} \\ \Leftrightarrow d &\equiv \tfrac{1}{e} \pmod{(p-1)(q-1)} \end{aligned}$$

$$\Leftrightarrow d \equiv e^{-1} \pmod{(p-1)(q-1)} \tag{A.5}$$

[10] Für einen Beweis lesen Sie bitte die gängigen Bücher der Zahlentheorie und fragen Sie Ihren theoretischen Informatiker oder Mathematiker.

Das bedeutet, wir müssen ein e wählen, für das ein e^{-1} existiert. Da wir mit modulo rechnen, existieren zwar e für die das nicht gilt, aber es sind recht wenige. Es muß gelten

$$ggt(e, (p-1)(q-1)) = 1$$

Da nur wenige e diese Gleichung nicht erfüllen, ist es ausreichend auf das geradewohl ein e zu wählen und mit dem Euklidischen Algorithmus den größten gemeinsamen Teiler zu berechnen. Ist dieser 1, so berechnet man d, was z.B. mit dem erweiterten euklidischen Algorithmus in einem Schritt möglich ist Eine häufige Wahl für e ist 3.

Da sich d aus e einfach berechnen läßt, wenn p und q bekannt sind, erklärt sich, warum nur ihr Produkt veröffentlicht wird. In der Literatur[11] finden sich viele Diskussionen dazu, wie p und q beschaffen sein müssen, damit die Faktorisierung von n erschwert wird, aber wir beschränken uns hier auf die Anmerkung, daß diese Zahlen ausreichend groß sein[12] und sich als Dezimalzahlen in ihrer Länge um einige Stellen unterscheiden sollten.

A.3.2.3 Die euklidische Algorithmen

Das folgende wurde von uns der Vollständigkeit halber eingefügt. Es ist aber nicht spezifisch für RSA, sondern zeigt nur, wie der größte gemeinsame Teiler zweier Zahlen, bzw. das Inverse bezüglich der Multiplikation im Moduloraum berechnet werden kann.

Wen nur die Funktionsweise von RSA interessiert, mag diesen Abschnitt getrost überspringen.

Der euklidische Algorithmus

Der euklidische Algorithmus berechnet den größten gemeinsamen Teiler zweier natürlicher Zahlen u, v mit $v \leq u$. Er beruht auf dem von Euklid aufgestellten Satz, daß für besagte Zahlen und für den Rest einer Teilung

[11] [Bauer 95, Kap 11.3][Knuth81, Kap 4.5.5]

[12] Was „ausreichend" ist, hängt vom befragten Experten und dem Stand der Technik ab und ändert sich monatlich.

mit Rest[13] r gilt:

$$ggt(u, v) = ggt(v, r) \tag{A.6}$$

Daraus läßt sich nun einfach ein Algorithmus konstruieren, indem man rekusiv immer wieder v für u und r für v einsetzt, bis eine Teilung den Rest $r = 0$ ergibt. (Wenn $x = qy$, dann gilt $ggt(x, y) = y$.)[14]

```
r ← u mod v
WHILE r ≠ 0
    u ← v
    v ← r
    r ← u mod v
END

RETURN v
```

Der erweiterte Euklidische Algorithmus

Man kann den euklidischen Algorithmus nach [Knuth81, Kap. 4.5.2 Algorithmus X][15] nun so erweitern, daß auch u' und v' berechnet werden mit

$$uu' + vv' = ggt(u, v) \tag{A.7}$$

Dazu beötigen wir drei Vektoren $\vec{u} = (u_1, u_2, u_3)$, $\vec{v} = (v_1, v_2, v_3)$ und $\vec{t} = (t_1, t_2, t_3)$. Diese werden im Folgenden immer die Bedingungen

$$uu_1 + vu_2 = u_3 \tag{A.8}$$

$$uv_1 + vv_2 = v_3 \tag{A.9}$$

$$ut_1 + vt_2 = t_3 \tag{A.10}$$

[13] Die Teilung mit Rest ist das, was man in der Grundschule lernt. Für $u = 10$ und $v = 8$ wäre der Quotient $q = 1$ und der Rest $r = 2$. Mathematisch:

$$u = qv + r, \quad 0 \leq r \leq v - 1$$

[14] Der Euklidische Algorithmus findet sich u.a. in [Knuth 73, Kap. 1.1 Algorithmus E].

[15] Interressant ist vielleicht auch, daß sich dort auch Varianten befinden, die für das Rechnen mit Binärzahlen und sehr großen Zahlen optimiert wurden. In den folgenden Kapiteln werden Algorithmen zum Faktorisieren großer Zahlen, Primzahlentests und RSA behandelt.

Ein ausführlicher Beweis des erweiterten euklidischen Algorithmus befindet sich in [Knuth 73, Kap. 1.2.1 Algorithmus E] erfüllen. Der eigentliche Algorithmus sieht nun folgendermaßen aus:

$(u_1, u_2, u_3) \leftarrow (1, 0, u)$
$(v_1, v_2, v_3) \leftarrow (0, 1, v)$
WHILE $v_3 \neq 0$
  $q \leftarrow \lfloor u_3/v_3 \rfloor$
  $(t_1, t_2, t_3) \leftarrow (u_1, u_2, u_3) - q(v_1, v_2, v_3)$
  $(u_1, u_2, u_3) \leftarrow (v_1, v_2, v_3)$
  $(v_1, v_2, v_3) \leftarrow (t_1, t_2, t_3)$
END
RETURN u_1, u_2, u_3

Um nun einzusehen, daß die postulierten Bedingungen immer eingehalten werden, braucht man sich nur vergegenwärtigen, daß

- die Initialisierungswerte für $\vec{u}$ bzw. $\vec{v}$ in die Bedingungen (A.8) bzw. (A.9) eingesetzt

  $u = u$ bzw. $v = v$

  ergeben, was zweifelos wahr ist.

- die Zuweisung an $\vec{t}$ die Bedingung (A.10) erfüllt, falls (A.8) und (A.9) zutreffen.

- die Zuweisung an $\vec{u}$ die Bedingung (A.8) erfüllt, falls die Bedingung (A.9) erfüllt ist.

- die Zuweisung an $\vec{v}$ die Bedingung (A.9) erfüllt, falls die Bedingung (A.10) erfüllt ist.

Wir wissen also, es gilt:

$$uu_1 + vu_2 = u_3$$

Dies sieht nun fast nach (A.7) aus, wir müssen nur noch zeigen, daß

$$u_3 = ggt(u, v)$$

Dies gilt aber, da die Operationen auf t_3 u_3, v_3 denen des euklidischen Algorithmus entsprechen:

$$\begin{array}{rll} & t_3 & \leftarrow u_3 - qv_3 \\ \Leftrightarrow & t_3 & \leftarrow u_3 - \lfloor u_3/v_3 \rfloor v_3 \\ \Leftrightarrow & t_3 & \leftarrow u_3 - (u_3 - (u_3 \bmod v_3)) \\ \Leftrightarrow & t_3 & \leftarrow u_3 \bmod v_3 \\ \\ & u_3 & \leftarrow v_3 \\ \\ & v_3 & \leftarrow t_3 \end{array}$$

t_1, u_1, v_1 sind eigentlich überflüssig, da sich u_1 leicht aus

$$u_1u + u_2v = u_3 = ggt(u, v)$$

berechnen läßt. Diese Variablen wurden hier auch nur aufgeführt, um den oben skizzierten Beweis zu erleichtern, und weil der Algorithmus in der Literatur stets so vorgestellt wird.

u_2 ist dagegen sehr interessant. Setzen wir nun einmal voraus, daß

$$ggt(u, v) = 1$$

wie dies z.B. bei RSA gilt, und rechnen wir im Folgenden modulo u. Dann folgt

$$\begin{array}{rrl} & u_1u + u_2v & \equiv 1 \\ \Leftrightarrow & 0 + u_2v & \equiv 1 \\ \Leftrightarrow & u_2v & \equiv 1 \end{array}$$

Das bedeutet aber, daß u_2 das Inverse der Multiplikation von v ist. Womit wir einen Algorithmus haben, der es erlaubt, in einem Schritt festzustellen, ob ein Wert teilerfremd zu einem Modul ist und seinen Kehrwert bezüglich eben dieses Moduls auszurechnen. Damit ist es möglich, RSA auch in der Praxis einzusetzen.

Literaturverzeichnis

[AnKu] Ross Anderson, Markus Kuhn, „Tamper Resistance – a Cautionary Note“, *Proceedings of the 2nd Workshopon Electronic Commerce*, Oakland, Kalifornien, 18. - 20. November 1996
`http : //www.cl.cam.ac.uk/ftp/user/rja14/tamper.ps.gz`

[Back 95] Adam Back, „Another SSL breakage“,
`http : //www.dcs.ex.ac.uk/~aba/ssl/`

[Bauer 95] Friedrich L. Bauer, „Entzifferte Geheimnisse, Codes und Chiffren und wie sie gebrochen wurden“, Springer-Verlag Berlin Heidelberg 1995

[BankNet 96] Produktinformationen des BankNet Electronic Banking Service,
`http : //193.118.187.113/bank@$JPY`

[Brands 93] Stefan Brands, „Untraceable Off-line Cash in Wallets with Observers“, *Advances in Cryptology – Crypto '93*, S.302ff, Springer Verlag 1994

[CAFE 96] WWW-Seiten der C.A.F.E, Conditional Acces For Europe,
`http : //www.cwi.nl/cwi/projects/cafe.html`

[CaSiTy] L. Jean Camp, Marvin Sirbu, J.D. Tygar, „Token and Notational Money in Electronic Commerce“,
`http : //www.cs.cmu.edu/afs/cs/user/jeanc/www/usenix.html`

[Chaum 85] David Chaum, „Security without Identification: Trasaction Systems to Make Big Brother Obsolete“, *Communications of the ACM*, Vol 28, Nr.10, S.65-75, 1985

[Chaum 87] David Chaum, „Security without Identification: Card Computers to make Big Brother Obsolete“,
`http : //www.digicash.nl/publish/bigbro.html`

[Chaum 92] David Chaum, „Archieving Electronic Privacy“, *Scientific American*, August 1992, S.96-101,
http://www.digicash.nl/publish/siam.html

[Chaum 93] David Chaum, „Online Cash Checks“,
http://www.eff.org/pub/Privacy/Digital_money/online_cash_chaum.paper
oder direkt über Digicash,
http://www.digicash.nl/publish.

[Chaum 94] David Chaum, „Prepaid Smart Card Techniques: A Brief Introduction and Comparison“,
http://www.eff.org/pub/Privacy/Digital_money/smartcard_chaum.article
oder direkt über Digicash,
http://www.digicash.nl/publish.

[Checkfree 96] Produktinformationen der Firma Checkfree,
http://www.checkfree.com

[Cybercash 95] Stephen Crocker, Brian Boesch, Alden Hart James Lum, „CyberCash: Payment Systems for the Internet“, Vortrag auf der INET '95,
http://www.isoc.org/HMP/PAPER/181/abstract.html

[Cybercash 97] Produktinformationen der Firma Cybercash
http://www.cybercash.com

[Div 94] Diverse, „The ESPRIT Project CAFE – High Security Digital Payment Systems –“, ESORICS 94 (Third European Symposium on Research in Computer Security), LNCS 875, Springer-Verlag, Berlin 1994, 217-230
http://www.informatik.uni-hildesheim.de/FB4/sirene/projects/cafe/index.html

[Digicash 96] Produktinformationen von Digicash,
http://www.digicash.nl

[Doligez 95] Damien Doligez, (Ohne Titel), eine Übersicht über die Anstrengungen der Cypherpunks SSL zu knacken
http://pauillac.inria.fr/~doligez/ssl/

[FeBaDeWa 97] Edward W. Felten, Dirk Balfanz, Drew Dean, Dan S. Wallac, „Web Spoofing : An Internet Con Game“, Technical Report 540-96

(revised Feb. 1997) Department of Computer Science, Princeton University,
`http://www.cs.princeton.edu/sip/pub/spoofing.html`

[Finney 93a] Hal Finney, „Detecting Double Spending", Zu finden im Archiv der Electronic Frontier Foundation unter
`http://www.eff.org/pub/Privacy/Digital_money/`
Es handelt sich um eine Zusammenfassung von
Chaum, Fiat, Noar, „Untraceable Electronic Cash", *Advances in Cryptology – CRYPTO '88 Proceedings*, S.319ff, Springer Verlag 1990

[Finney 93b] Hal Finney, „Down with Observers!", 22.August 1993
`http://www.rain.org/~hal/anti_observers.html`

[FV 96a] Produktinformationen der Firma First Virtual,
`http://www.fv.com`

[FV 96b] „Vulnerability of Sotware-Based Credit Card Encryption", First Virtual 1996
`http://www.fv.com/ccdanger/techreview.html`

[Heirig ??] Werner Heirig, „Geldkarte - Einführung der elektronischen Geldbrse", Banken aktuell, Arbeitsmaterialien fr den Unterricht
`http://www.stam.de/geldkart.htm`

[HmbDSB 95] „Kartengestützte Zahlungsverfahren", 14. Tätigkeitsbericht des Hamburgischen Datenschutzbeauftragten,
`http://www.hamburg.de/Behoerden/HmbDSB/TB14/tb_gld00.htm\`

[ITSEC] „Kriterien für die Bewertung der Sicherheit von Systemen der Informationstechnik (ITSEC)", Version 1.2, Luxemburg: Amt für amtliche Veröfflichungen der Europäischen Gemeinschaften, 1991, ISBN 92-826-3003-X

[Jones 96] Tim Jones„Mondex on 'The Future of Money'", *submisssion to the US house of represetatives*, 11. Juni 1996,
`http://www.mondex.com/hsrep.htm`

[Kahn 67] D. Kahn, „The Codebreakers", Macmillan, New York 1967

[Knuth 73] Donald E. Knuth, „The Art of Computer Programming, Volume 1 / Fundamental Algorithms", 2.Auflage, Addison-Wesley 1973

[Knuth81] Donald E. Knuth, „The Art of Computer Programming, Volume 2 / Seminumerical Algorithms“, 2.Auflage, Addison-Wesley 1981

[Kristula 97] Dave Kristula, „The History of the Internet“, März 1997

[Kuhn 96] Markus Kuhn, „Sicherheitsanalyse eines Mikroprozessors mit Busverschlüsselung“, Diplomarbeit im Fach Informatik, Institut für Mathematische Maschinen und Datenverarbeitung (III), Friedrich-Alexander-Universität Erlangen-Nrnberg

[Lamport 81] Leslie Lamport, „Password Authentication with Insecure Communication“, *Communications of the ACM 24.11*, November 1981, S. 770 - 772

[Lexikon] „Großes Universallexikon“, Honos Verlags AG, Zug 1988

[Lockstone 96] Paul Lockstone, „NatWest introduces new University Smart Card“, Pressemitteilung 28.3.1996,
`http://www.ex.ac.uk/public_html/MONDEX/news1.html`

[MarkTwain 96] Produktinformationen der Mark Twain Bank,
`http://www.marktwain.com`

[Martin 97] Andreas Martin, „Praxiseinsatz der Chipkarte im Geldkarte-System der deutschen Kreditwirtschaft“, in: Matthias Fluhr (Hrsg.), „Die Chipkarte: Eine Welt der Möglichkeiten“, Konferenzdokumentation OMNICARD 1997, Internationale Konferenz in Berlin, 15. - 17. Januar 1997, inTIME Berlin

[Matthes 96] Nadja Matthes, „Bargeld bevorzugt, Die EC-Karte wird zum Chip-Portemonnaie aufgerüstet – doch der Handel zieht nicht mit“, Focus 52/1996

[MeNe 93a] Gennady Medvinsky, B. Clifford Neuman, „NetCash: A design for practical electronic currency on the Internet“, Artikel für *Proceedings of the First ACM Conference on Computer and Communications Security*, November 1993
zu finden unter [Netcash 96]

[MeNe 93b] Gennady Medvinsky, B. Clifford Neuman, „Electronic Currency for the Internet“,*EM - Electronic Markets*, No. 9-10, Oktober 93, S.23
zu finden unter [Netcash 96]

[MeNe 95] B. Clifford Neuman, Gennady Medvinsky, „Requirements for Network Payment: The Netcheque^{TM} Perspektive", *Proceedings of the IEEE Compcon'95*, San Francisco, März 1995,
zu finden unter [NetCheque 96]

[Miller 96] Jim Miller, „E-money mini-FAQ (release 2.0)",
`http://www.ex.ac.uk/~RDavies/arian/emoneyfaq.html`

[Millicent 96] Informationen und Protokollbeschreibungen von Millicent
`http://www.research.digital.com/SRC/millicent/`

[MjMi 97] Stig F. Mjølsnes, Rolf Michelsen, „Open Transnational System for Digital Currency Payments", *Proceedings of Hawaii International Conference on System Sciences HICSS-30*, Vol V: Advanced Technology Track, 1997
`http://www.delab.sintef.no/~cafe/HICSS-30.ps`

[Mondex 96a] Produktinformationen von Mondex,
`http://www.mondex.com/mondex/`

[Mondex 96b] „Mondex Magazine December 1996", TPD Publishing Ltd., Long Island House, 1-4 Warple Way, London W3 0RG on behalf of Mondex International Limited, 47-53 Cannon Street, London EC4M 5SQ.

[MPTP 95] „Micro Payment Transfer Protocol (MPTP), Version 0.1", W3C Working Draft 22.November 1995
Diese Version:
`http://www.w3.org/pub/WWW/TR/WD-mptp-951122`
Aktuelle Version:
`http://www.w3.org/pub/WWW/TR/WD-mptp`

[NetBank 97] Produktinformationen der NetBank,
`http://www.netbank.com/netcash`

[NetBill 96] Produktinformationen des Netbill Projektes,
`http://www.ini.cmu.edu/netbill/`

[Netcash 96] Produktinformationen des NetCash Projektes
`http://gost.isi.edu/info/netcash`

[NetCheque 96] Produktinformationen des NetCheque Projektes,
http://gost.isi.edu/info/netcheque

[Netscape 96a] „Netscape Data Security, An Overview of Implementations and Plans from Netscape Communications“,
http://www.netscape.com/newsref/ref/netscape-security.html

[Netscape 96b] „Key Challenge“,
http://www.netscape.com/newsref/std/key_challenge.html

[Neuman 93] B. Clifford Neuman, „Proxy-Based Authorization and Accounting for Distributed Systems“, *Proceedings of the 13th International Conference on Distributed Computing Systems*, Pittsburgh, Mai 1993, zu finden unter [NetCheque 96]

[PI 95] Privacy International, „Mondex and Anonymity“, 3. September 1996,
http://www.privacy.org/pi/activities/mondex

[Reif 96] Holger Reif, „Cyber - Dollars“, *c't* 5/96, S.144

[RFC 1760] N. Haller, „RFC 1760 - The S/KEY One-Time Password System“, Februar 1995

[RFC 1898] D. Eastlake 3rd, B. Boesch, S. Crocker, M. Yesil, „Cybercash Credit Card Protocol Version 0.8“,
ftp://ds.internic.net/rfc/rfc1898.txt

[Richards 96] Scott Richards, “Electronic Money / Internet Payment Systems“,
http://www2.cob.ohio-state.edu/~richards/bankpay.htm

[RiSha 96] Ronald L.Rivest, Adi Shamir, „PayWord and MicroMint: Two simple micropayment schemes“, 7. Mai 1996,
http://theory.lcs.mit.ed/~rivest/RivestShamir-mpay.ps

[Schneier 96] Bruce Schneier, „Angewandte Kryptographie“, Addison-Wesley (Deutschland) 1996, ISBN 0-471-11709-9

[Schröder 95] Holger Schröder, „Datenhandel, Der große Bruder schaut in jedes Portemonaie“, *Stern* 30/95 S.122

[SchuWe] Mathias Schunter, Arndt Weber, „CAFE - Ein sicheres datenschutzorientiertes Zahlungssystem“
`http://www.informatik.uni-hildesheim.de/FB4/sirene/projects/cafe/index.html`

[SET 96] „SET specification“, Visa und Mastercard 1996,
`http://www.visa.com/cgi-bin/vee/sf/set/intro.html`

[SFNB 96] Produktinformationen der Security First Network Bank,
`http://www.sfnb.com`

[SHTTP 94] E.Rescorla, A.Schiffman „INTERNET-DRAFT: The Secure Hypertext Transfer Protocol“, Dezember 1994,
`http://www.commerce.net/information/standarts/drafts/shttp.txt`
(über einen Link von
`http://www.hpl.hp.co.uk/projects/vishnu/main.html`)

[SoNa 92] Sebastiaan von Solms, David Naccache, „On Blind Sigmatures and Perfect Crimes“, Computers and Security, 11 (1992) S.581-583, Elsevier Publishers Ltd.

[Spiegel 96] „Einbruch ohne Spuren“, *Der Spiegel* 47/96, S.216/7

[SSL 96] Alan O. Freier, Philip Karlton, Paul C. Kocher, „INTERNET DRAFT: The SSL Protocol Version 3.0“,
`ftp://ds.internic.net/internet-drafts/draft-freier-ssl-version3-01.txt`

[Waidner 94] Michael Waidner, „Das ESPRIT-Projekt “Conditional Access for Europe”“, Konferenzdokumentation der Multicard '94, Berlin 23. - 25. Februar 1994, S. 236 - 249, inTIME Berlin, Seesener Str. 53, 10711 Berlin

[WSJ 95] „Visa to use Carnegie Mellons Netbill“, Wall Street Journal 2/15/95 B6,
`http://ganges.cs.tcd.ie/mepeirce/Press/netvisa.html`

[Zakon 94] Robert H'obbes' Zakon, „Hobbes' Internet Timeline v1.3a“, 20. Juni 1994

[Zimmerman 94] Philip Zimmerman, „PGP(tm) User's Guide, Volume I: Essential Topics“, überarbeitete Version vom 31.August 1994, erhältlich zusammen mit seinem Verschlüsselungsprogramm PGP von vielen ftp - Servern