ecash: Das Geld auf der Festplatte

Oliver Vornberger
Fachbereich Mathematik/Informatik
Universität Osnabrück
49069 Osnabrück
oliver@uos.de

`http://www-lehre.inf.uos.de/vsin/vsin04`

Zahlreiche Verfahren und Protokolle für den elektronischen Zahlungsverkehr sind in den vergangenen Jahren vorgeschlagen worden. Das System $ecash^{TM}$ der Firma *eCash Technologies, Inc.* zeichnet sich durch Eleganz und Einfachheit aus. Dieser Artikel stellt die theoretischen Grundlagen und die Anwendung von $ecash^{TM}$ vor.

1 Einleitung

Mit der Verbreitung des Internets wächst auch seine Eignung, als Medium zwischen Kaufmann und Kunde zu wirken. Zum einen möchte der Kaufmann die Waren in seinem Online-Shop optimal präsentieren, was mit den heutigen multimedialen Techniken bereits sehr gut funktioniert. Zum anderen möchte er aber auch für seine Dienstleistung entlohnt werden, wobei sich neben dem konventionellen Zahlungsverkehr über Kreditkarte oder Banküberweisung wiederum das Internet anbietet.

Grundlage eines jeden elektronischen Zahlungssystems muß die Fähigkeit sein, Nachrichten in einer Weise zu übermitteln, daß diese von unberechtigten Dritten nicht gelesen oder gar manipuliert werden können. Das heißt, der Kunde möchte sein Konto nur in der von ihm autorisierten Weise belastet sehen. Neben dieser Forderung nach Korrektheit einer Zahlung gesellt sich nun die Forderung nach Anonymität einer Zahlung. Damit ist gemeint, den Zahlungsverkehr zwischen Kunde, Kaufmann und Bank, bei der beide ein Konto unterhalten, so zu gestalten, daß bei der Belastung des Kontos des Auftraggebers zugunsten des Zahlungsempfängers die Bank über die Geschäftsbeziehung dieser beiden nichts erfahren kann. Damit kann der Kunde im Internet Käufe tätigen und abrechnen, ohne daß seine Bank Einblicke in sein Kaufverhalten erhält.

Was sich hier wie ein Ding der Unmöglichkeit anhört, läßt sich in der Tat mit wenigen pfiffig aufeinander abgestimmten Verfahren realisieren. Es handelt sich um das System *ecash* der Firma

eCash Technologies, Inc., welches zur Zeit in mehreren Ländern für einige Banken lizensiert und im Großversuch getestet wird. Grundlage für *ecash* bilden spezielle kryptographische Verfahren, die sogenannten *Public Key Systems*. Eine kurze Einführung in diese Thematik liefern Kapitel 2 und 3. Kapitel 4 und 5 schildern, wie darauf aufbauend ein Protokoll entstehen kann, welches die Anonymität von Zahlungen eines Kunden gegenüber seiner Bank wahrt. Kapitel 6 erläutert den praktischen Einsatz der von *eCash Technologies* lizensierten Software. Kapitel 8 faßt die Vorteile dieses Ansatzes zusammen. Weitere Einzelheiten und begleitende Literatur finden sich unter der Web-Adresse `http://www.ecashtechnologies.com`.

2 Kryptographie

Der Wunsch nach verschlüsselter Übertragung von Nachrichten ist so alt wie die Schrift. Der grundsätzliche Ablauf ist in Abbildung 1 skizziert: Der Klartext *x* dient als Eingabe für ein Verschlüsselungsverfahren *encode*, welches über einen Schlüssel *e* parametrisiert ist. Das heißt, der grundsätzliche Ablauf der Verschlüsselung ist allen Beteiligten bekannt, mit Hilfe des Schlüssels *e* kann der Vorgang jeweils individuell beeinflußt werden. Auf der Gegenseite wird mit dem Verfahren *decode* und seinem Schlüssel *d* der Vorgang umgekehrt und somit der Klartext rekonstruiert. Zum Beispiel könnte das generelle Verfahren zum Verschlüsseln eines Wortes *x* darin bestehen, jeden seiner Buchstaben um eine gewisse Zahl von Einheiten im Alphabet weiterzuschieben. Die Schlüssel *e* und *d* geben dann die konkrete Zahl von Einheiten an. So wird für *e* = 3 aus dem Wort BACH das Wort EDFK und auf der Gegenseite durch *d* = -3 daraus wieder BACH.

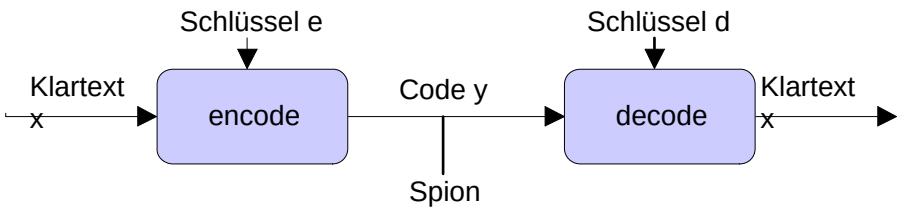

Abbildung 1: Ablauf einer Verschlüsselung

Diese Technik funktioniert so lange gut, wie es gelingt, die zum Bearbeiten einer Nachricht verwendeten Schlüssel *e* und *d* auf einem sicheren Kanal zu übertragen, z.B. durch einen Kurier. Ein Spion, der ohne Kenntnis der Schlüssel die Leitung anzapft, ist dann nicht in der Lage, den beobachteten Code zu entschlüsseln (immer vorausgesetzt, der Raum der möglichen Schlüssel wurde zur Abwehr eines vollständigen Durchsuchens groß genug gewählt). Im Zeitalter der globalen Vernetzung besteht natürlich der Wunsch, auch die beiden Schlüsselpaare *e* und *d* per Leitung auszutauschen. Nun aber laufen wir Gefahr, daß der Spion von ihnen Kenntnis erhält und damit den Code knackt.

3 Public Key Systems

Als Antwort auf dieses Dilemma präsentierten drei Wissenschaftler vom MIT in Boston im Jahre 1978 eine verblüffende Lösung [1]. Das nach den Autoren Rivest, Shamir und Adleman benannte RSA-Verfahren verwendet das Konzept des öffentlichen Schlüssels und zählt daher zu den *Public Key Systems*. Im folgenden sei der Einfachheit halber angenommen, der zu übertragende Klartext bestehe aus einer einzigen ganzen Zahl *x* (bei realen Anwendungen muß der Klartext zuvor als eine Folge von Zahlen dargestellt werden).

Abbildung 2 zeigt, wie die Kommunikationspartner *Alice*, welche die zu übertragende Nachricht *x* verfaßt hat und ihr Kommunikationspartner *Bob*, für den die Nachricht bestimmt ist, dabei vorgehen. *Bob* konstruiert auf Grundlage eines mathematischen Verfahrens drei Zahlen *e*, *d* und *n* mit der Eigenschaft $(x^e \bmod n)^d \bmod n = x$. Auf deutsch: Wenn man *x* *e*-mal mit sich selbst malnimmt und dann das Ergebnis *d*-mal mit sich selbst malnimmt, kommt wieder *x* heraus. Alle Rechnungen erfolgen mit den Resten, die bei Division durch *n* entstehen, genannt modulo. Nun können die Zahlen *n* und *e* von *Bob* veröffentlicht werden für jeden potentiellen Sender der an ihn eine Nachricht schicken möchte. Verschlüsselt wird dann mit der für den Adressaten *Bob* konstruierten öffentlichen Funktion $y := enc_B(x) = x^e \bmod n$, entschlüsselt wird vom Adressaten mit der nur ihm bekannten, aber sonst geheimen Funktion $x := dec_B(y) = y^d \bmod n$.

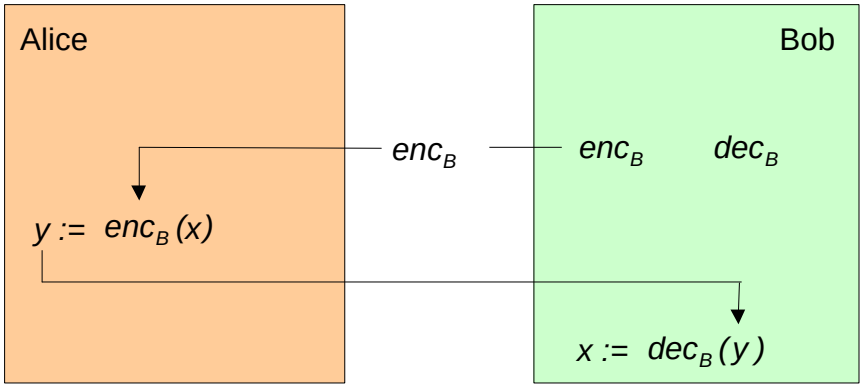

Abbildung 2: Nachricht verschlüsseln

Als Beispiel seien $n = 143, e = 47$ und $d = 23$ genannt. Dann wird x verschlüsselt durch Bilden von $y := x^{47} \bmod 143$, auf der Gegenseite wird y entschlüsselt durch Bilden von $x := y^{23} \bmod 143$. In der Praxis werden die Zahlen n, e und d unter Verwendung 300-stelliger Primzahlen konstruiert und sind so gewählt, daß aus der Kenntnis von n, e und der verschlüsselten Nachricht y die Rekonstruktion von d und damit die Rekonstruktion von x in akzeptabler Rechenzeit nicht möglich ist. Ein abhörender Spion könnte daher die beobachtete Nachricht y erst in Jahrzehnten entschlüsseln.

4 Signieren mit Public Key Systems

Wie in Kapitel 3 erläutert, veröffentlicht der Empfänger B an alle potentiellen Sender sein öffentliches Verschlüsselungsverfahren enc_B , welches vom Sender A benutzt wird, um $y := \text{enc}_B(x)$ an den Empfänger B zu schicken. Der Inhalt der Nachricht bleibt dabei vor etwaigen Spionen geschützt und kann nur durch den Empfänger B durch Anwendung von $x := \text{dec}_B(y)$ wieder lesbar gemacht werden.

Eine Umkehrung der Reihenfolge der beiden Schlüssel führt nun zu einem interessanten Effekt , visualisiert in Abbildung 3 . Wendet der Sender B zunächst seine eigene, geheime Entschlüsselung dec_B an, so kann dieser Schritt durch den Empfänger unter Anwendung von enc_B wieder rückgängig gemacht werden. Übertragen wird also vom Sender B der Code $y := \text{dec}_B(x)$. Entschlüsselt wird von Empfänger A durch Anwendung des öffentlichen Schlüssels von B. Das heißt, Empfänger A berechnet $x := \text{enc}_B(y)$. Sollte bei einem solchen Schritt aus dem erhaltenen Code y eine sinnvolle Nachricht x entstehen, so kann Empfänger A sicher sein, daß diese Nachricht vom Sender B stammt, denn nur Sender B ist in der Lage, den zu enc_B inversen Schritt dec_B durchzuführen. Somit wirkt sich diese Kombination von öffentlichen und privaten Schlüsseln wie eine digitale Signatur aus, mit der ein Empfänger von Nachricht y vor Gericht nachweisen kann, daß als Urheber dieser Nachricht nur Sender B in Frage kommt. (Geheim zu haltende signierte Nachrichten können natürlich zusätzlich mit dem öffentlichen Schlüssel von Empfänger A verschlüsselt werden).

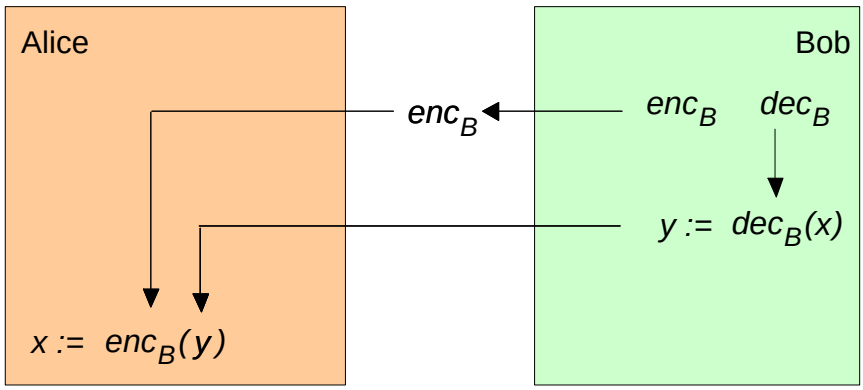

Abbildung 3: Nachricht signieren

5 Blinde Signaturen

Wie können nun *Public Key Systems* im elektronischen Zahlungsverkehr verwendet werden ? Eine Möglichkeit wäre es, wenn die Bank elektronische Geldscheine in Form von Nachrichten herausgibt, welche von ihr mit der digitalen Signatur der Bank unterschrieben sind und daher als

echt erkannt werden können von jedermann, der über den öffentlichen Schlüssel enc_{Bank} der Bank verfügt. Beim Einreichen eines solchen digitalen Geldscheins würde aber die Bank erfahren, wofür ihr Kunde sein Geld ausgegeben hat, denn schließlich muß sie sein Konto belasten. Um die Anonymität des Bezahlenden zu sichern, schlug David Chaum daher folgendes Prinzip der blinden Signatur vor [2], visualisiert in Abbildung 4 .

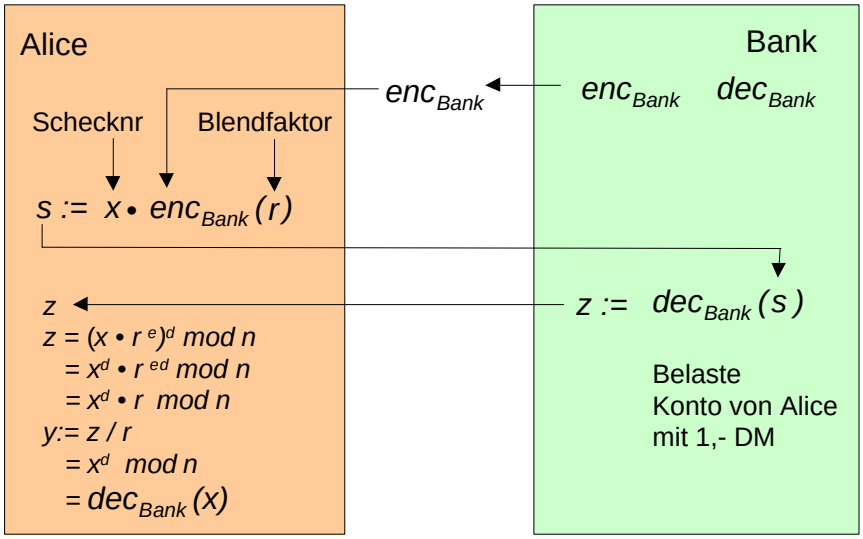

Abbildung 4: Münzen erzeugen mit blinder Signatur

Zunächst erzeugt der Kunde A eine zufällige Schecknummer x, die so groß ist, daß die Wahrscheinlichkeit vernachlässigbar ist, daß dieselbe Schecknummer auch von einem anderen Kunden generiert werden könnte. Als nächstes würfelt der Kunde einen individuellen Ausblendfaktor r, verschlüsselt ihn mit dem öffentlichen Schlüssel der Bank und schickt das Produkt $s := x \cdot \text{enc}_{\text{Bank}}(r)$ an die Bank. Die Bank belastet nun das Konto des Kunden A mit einem vorher vereinbarten Betrag, z.B: 1,- DM. Dann signiert sie die Nachricht y mit Ihrer digitalen Unterschrift $z = \text{dec}_{\text{Bank}}(s)$ und schickt z an den Kunden A zurück. Aufgrund der Konstruktion der Nachricht s ist die Bank nicht in der Lage, die in ihr versteckte Schecknummer x zu sehen, während der Kunde A unter Verwendung seines Ausblendfaktors r aus der von der Bank zurückerhaltenen Nachricht z die unterschriebene Schecknummer $y := \text{dec}_{\text{Bank}}(x)$ extrahieren kann. Bildlich gesprochen hat Kunde A seine von ihm ausgedachte Schecknummer s in einem verschlossenen Umschlag mit Durchschlagpapier an die Bank geschickt und die Bank hat durch den Umschlag hindurch, ohne die Schecknummer zu sehen, den Scheck unterschrieben.

Nun verfügt Kunde A über einen von der Bank signierten Geldschein s im Werte von 1,- DM und kann ihn im Internet zum Bezahlen verwenden. Abbildung 5 zeigt den Ablauf. Jeder Kaufmann, bei dem der Kunde Ware bestellen möchte und der diesen Geldschein y präsentiert bekommt, kann durch Anwendung des öffentlichen Schlüssels der Bank überprüfen, ob $x = \text{enc}_{\text{Bank}}(y)$ eine gültige Banknote darstellt. Falls dies der Fall ist, kann der Kaufmann die Banknote s bei der Bank einreichen. Dort wird sie erneut von der Bank verifiziert. Danach schreibt die Bank

dem Konto des Kaufmanns den Betrag von 1,- DM gut und vermerkt die Schecknummer x als bereits ausgegeben. Das Konto von Kunde A war bereits bei Ausstellung der signierten Banknote belastet und kann von der Bank nicht in Verbindung zur eingereichten Schecknummer gebracht werden, da diese zum Zeitpunkt der Unterschrift durch den Ausblendfaktor r unzugänglich war.

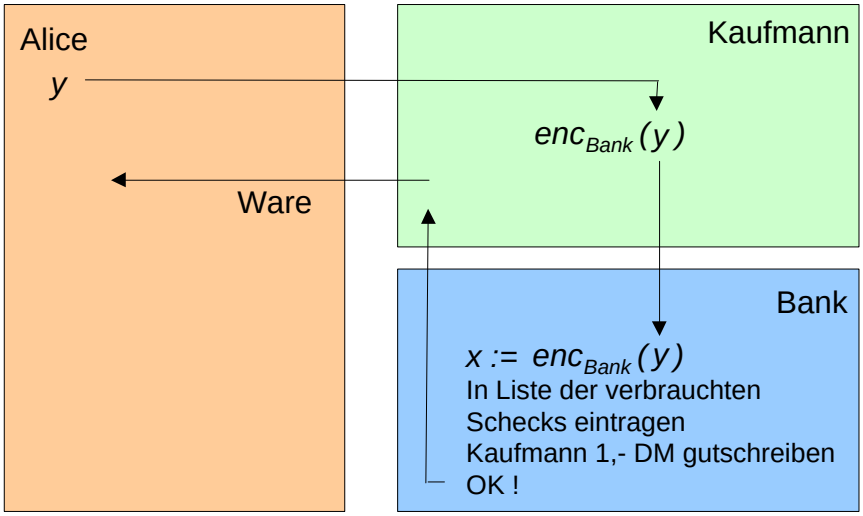

Abbildung 5: Münzen ausgeben

6 ecash

Die im Kapitel 5 vorgestellte Idee wurde 1994 von David Chaum mit seiner Firma *DigiCash* in die Tat umgesetzt und inzwischen von *eCash Technologies* weitergeführt. Bei mehreren europäischen und australischen Banken läuft inzwischen ein Feldversuch, um die Praktikabilität des Verfahrens im Alltag zu testen. Teilnehmer in Europa sind z.B. die *Deutsche Bank* (Deutschland), die *Bank Austria* (Österreich), die *Credit Suisse* (Schweiz) und die *Norske Bank* (Norwegen).

Um in Deutschland an dem Pilotprojekt teilnehmen zu können, muß der Kunde ein sogenanntes *ecash*-Konto bei der Deutschen Bank eröffnen und mit seinem Girokonto bei seiner Bank assoziieren. Per Banküberweisung kann dann das *ecash*-Konto mit maximal 400,- DM gefüllt werden. Gleichzeitig erhält der Kunde von der Deutschen Bank die sogenannte *ecash*-Geldbörse, eine Software, die nach Installation auf dem heimischen PC den Kontakt zum assoziierten *ecash*-Konto aufnimmt (Abbildung 6). Nach dem Betanken der *ecash*-Börse vom ecash-Konto aus liegt nun auf der Festplatte des Kunden sein Guthaben in Form von mehreren digital signierten Münzen. Die Münzen haben Denominationen in 2er-Potenzen, also 1 Pfennig, 2 Pfennig, 4 Pfennig, 8 Pfennig, etc., damit sich jeder gewünschte Betrag damit zusammenstellen läßt. Sie sind zudem mit einem Verfallsdatum versehen, um die Buchhaltung für bereits ausgegebene Münzen zu vereinfachen (Abbildung 7).

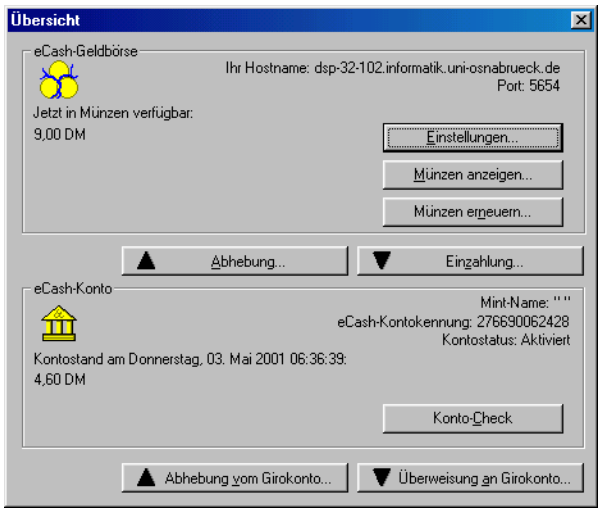

Abbildung 6: Oberfläche der ecash-Wallet

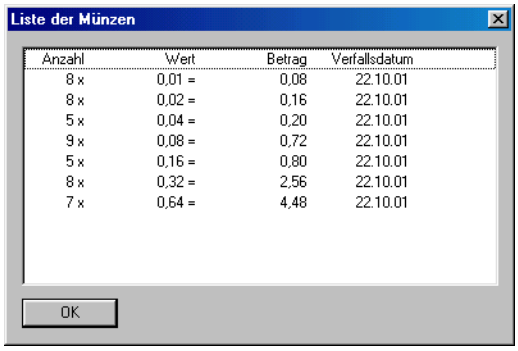

Abbildung 7: Bestand an Münzen in der Wallet

Besucht nun der Kunde mit seinem Web-Browser im Internet eine WWW-Seite eines Kaufmanns mit *ecash*-Zahlungsmöglichkeit, so wählt er zunächst wie gewohnt das zu kaufende Produkt aus. Danach bekommt er ein Popup-Fenster präsentiert (Abbildung 8), welches die relevanten Daten der bevorstehenden Transaktion noch einmal zusammenfaßt, nämlich Beschreibung und Preis der Ware. Stimmt der Kunde jetzt zu, so werden von der *ecash*-Geldbörse digital signierte Münzen in der angekündigten Gesamtsumme zum Server des Kaufmanns transferiert. Dieser Vorgang läuft ohne weiteres Zutun des Kunden ab, insbesondere sind keine Angaben zu Kreditkarten oder Kontoverbindungen mehr einzugeben. Nachdem der Kaufmann sich auf dem *Double-Spending-Server* der Bank vergewissert hat, daß diese Münzen nicht schon einmal ausgegeben worden sind, kann der Kunde mit der Ware beliefert werden. Dies kann entweder unmittelbar online geschehen, wenn es sich z.B. um Software, digitale Bilder, Audio- oder Videoclips handelt, oder per Post, wenn es sich um klassische Waren eines Versandhauses handelt, wie z.B. Bücher oder CDs.

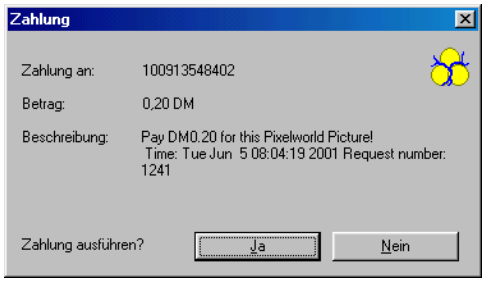

Abbildung 8: Bestätigung der Zahlung

Das System *ecash* ist auch in der Lage, Überweisungen von Geldbeträgen zwischen Privatpersonen durchzuführen, die über *ecash*-Konten verfügen. Weitere Leistungsmerkmale umfassen das Führen einer Protokolldatei, sowie Stornierung von Buchungen und Recovery nach einem Plattencrash. Eine umfangreiche Präsentation der Funktionalität findet sich unter der Web-Adresse `http://www.ecashtechnolgies.com`

7 Zusammenfassung

Das von David Chaum entworfene *ecash*-System besticht durch Einfachheit, Eleganz und Sicherheit. Seine Installation erfordert bei Kunde und Kaufmann nur geringen Aufwand. Innerhalb der Sparte bargeldloser Zahlungsverkehr läßt es sich deutlich einfacher benutzen als Verfahren zum verschlüsselten Austausch von Kreditkartendaten und Überweisungsinformationen. Aufgrund seines Designs ist *ecash* nicht nur im Internet einsetzbar, sondern auch in Verbindung mit Telefon-, Radio- oder Satellitenverbindungen. Ob es sich durchsetzen wird, hängt im wesentlichen davon ab, ob *eCash Technologies* genügend Kaufleute dafür gewinnen kann, *ecash* als Zahlungsweise für ihren Online-Shop anzubieten. Denn das schönste Geld auf der Festplatte des Clienten ist wertlos, wenn sich kein Server findet, der es akzeptiert.

8 Literatur

[1] R. Rivest, A. Shamir, L. Adleman: *A method for obtaining digital signatures and public key cryptosystems*, Communications of the ACM, Vol. 21, N. 2, 1978, S. 120-126.

[2] David Chaum: *Security without Identification: Transaction Systems to make Big Brother Obsolete*, Communications of the ACM, Vol. 28, N. 10, 1985, pp. 1030-1044; Revised version (in German), Informatik-Spektrum, vol. 10, 1987, pp. 262-277 1987.