Rechnergestützte Steganographie: Wie sie funktioniert und warum folglich jede Reglementierung von Verschlüsselung unsinnig ist

*von Steffen Möller, Andreas Pfitzmann, Ingo Stierand**

Zusammenfassung: *Zukünftig werden Nachrichten, z.B. Sprache, Text, Bilder, zunehmend digitalisiert übertragen, da dies billiger, perfekter und flexibler ist. In solchen digitalisierten Nachrichten können, für Außenstehende nahezu unerkennbar, weitere – allerdings notwendigerweise erheblich kürzere – Nachrichten versteckt werden. Wie diese rechnergestützte Steganographie funktioniert und welche Ergebnisse wir mit einem hierfür geschriebenen Programm, nach einer Märchengestalt DigiStilz genannt, erzielt haben, wird ausführlich dargestellt.*

Seit der technische Fortschritt den Einsatz von Verschlüsselung potentiell billig und einfach macht, wird diskutiert und in einigen Ländern bereits versucht, Verschlüsselung gesetzlich zu reglementieren. Andernfalls sei die Bekämpfung der internationalen Schwerkriminalität erheblich behindert. Selbst wenn wir unterstellen, daß Schwerkriminelle sich ausgerechnet diesem Verbot zu unterwerfen bereit seien: Durch die leicht einsetzbare rechnergestützte Steganographie ist für Ermittler nicht einmal das Vorliegen einer geheimen Nachricht, geschweige denn ihr Inhalt, feststellbar. Da rechnergestützte Steganographie nicht verhindert werden kann, solange der Besitz von programmierbaren Rechnern verbreitet ist, und sie für die Planung und Koordinierung von Straftaten ausreichende „versteckte" Bandbreite liefert, ist jeder Versuch einer Reglementierung von Verschlüsselung für die Strafverfolgung wirkungslos. Da für die Bürger vertrauenswürdige Verschlüsselung für die Wahrnehmung ihres Rechtes auf informationelle Selbstbestimmung unverzichtbar ist und Vertrauen in eine Schutzmaßnahme freie Kenntnis, Gestaltung und Benutzung voraussetzt, raten wir dringend von Reglementierungsversuchen ab.

1 Einleitung

„Das geht den überhaupt nichts an!" haben Sie bestimmt bereits schon mal jemanden sagen hören. Tatsächlich ist es ja so, daß Menschen schon immer Dinge vor anderen verbergen wollten. Sei das im Privatleben oder aus geschäftlichem Interesse. Wahrscheinlich ist es auch eine der ersten Erfindungen des Menschen gewesen, etwas dagegen zu tun, daß Fremde Geheimes erfahren.

Es gibt zwei Grundgedanken, wie das zu erreichen ist. Zum einen kann man das Objekt an einem geheimen Ort verstecken und hoffen, daß kein anderer, außer Eingeweihten es findet. Die zweite Möglichkeit ist, etwas so zu hinterlegen, daß kein anderer außer einem bestimmten Personenkreis etwas mit dem Objekt anfangen kann. Als Beispiel sei hier ein Safe angegeben. Zugriff auf ein Objekt darin haben nur diejenigen, die die richtige Kombination zum Öffnen des Safes haben. Dabei ist egal, wieviele Leute wissen, daß es sich im Safe befindet.

Aber nicht nur irgendwelche Gegenstände möchte man vor fremdem Zugriff schützen. Oft existiert auch der Wunsch, Nachrichten so zu übermitteln, daß nur der vorgesehene Empfänger damit etwas anfangen kann. Die Idee, wie man so etwas machen kann, ist schon in der Antike benutzt worden: man verschlüsselt die Nachricht. Nun kann den Text nur noch jemand lesen, der weiß, wie man ihn entschlüsselt.

Während jedoch früher ein Bote oft tagelang unterwegs war, um eine Nachricht zu übermitteln, hat sich die Situation heute grundlegend geändert. Immer mehr, immer öfter werden zur Kommunikation Rechner eingesetzt und mit ihnen oft weltweit gespannte digitale Nachrichtennetze. Mit dem Einsatz von Rechnern zur Datenkommunikation hat sich auch die Verschlüsselung von Daten mit ihnen bewährt. So kann ein Rechner Algorithmen verwenden, für die ein Mensch Jahre benötigen würde, um auch nur eine Seite zu verschlüsseln. So ist also den Menschen mit diesen Kryptographie-Systemen möglich geworden, ihre Daten bequem zu verschlüsseln und dann über das Netz zu schicken.

1.1 Verschlüsselungs-Kontrolle

Aber auch Menschen mit weniger ehrenhaften Absichten verwenden solche Systeme, um ihre dunklen Geschäfte abzuwickeln. Denn gerade sie sind sicherlich bestrebt, ihre Unternehmungen im Geheimen zu führen. Aus diesem Grund wird in einigen Regierungskreisen darüber nachgedacht, die Kryptographie zu verstaatlichen. Eine Möglichkeit, dies zu tun, so stellt man sich vor, ist eine Zulassungspflicht für Kryptographie-Systeme, ob Hardwaremaschinen oder Softwareprodukte.

Doch das ist nicht alles. Manche möchten auch, daß jeder, der seine Daten verschlüsseln will, sich zunächst sein „Codewort", sprich Schlüssel, von einer staatlich kontrollierten Vergabestelle holen muß.

Motivation dazu seien also die Mühen, die der Staat mit dem Feststellen der legalen Nutzung von Kryptographie und dem Entschlüsseln abgefangener geheimer Botschaften habe.

Solch ein Vorschlag mag zunächst ungewollte Heiterkeit auslösen, doch gleich darauf setzt das Erwachen ein und die Hoffnung: „Das versuchen die doch wohl nicht wirklich". Denn eine derartige Kontrolle von Kryptographie würde bedeuten, einen der wichtigsten und effizientesten Schutzmechanismen, die in der EDV zur Verfügung stehen, erheblich zu entwerten.

* Institut für Informatik, Universität Hildesheim, Postfach 101 363, D-31113 Hildesheim

Die Argumentation ist sicherlich ok, wenn jemand sagt: „Naja, zwar würde ich gerne ein Recht haben, meine Daten gegen fremden Zugriff zu sichern, doch geht das Interesse der Ermittlungsbehörden, Bösewichte zu erwischen, sicherlich vor." Es ist aber vermutlich nicht jedem klar, daß eine Verstaatlichung von Kryptographie genauso wenig gegen ein Übertragen von geheimen Daten über Drogentransporte schützt, wie ein Vermummungsverbot gegen Banküberfälle. Daß dem jedoch so ist, wollen wir in dieser Arbeit zeigen.

1.2 Verstecken der Nachrichten zum Umgehen des Verbotes: Steganographie

1.2.1 Allgemeines

Wenn das Verschlüsseln von Daten tatsächlich derart überwacht wird, so kann man einen Brief oder ein Gemälde schicken, das einem Betrachter absolut unauffällig erscheinen würde. Beispielsweise schickt man seinem Bruder einen Brief mit einem von seiner dreijährigen Tochter erstellten Gemälde – einem gemalten Apfelbaum. Ein normaler Betrachter (sprich Polizist) würde dort einen etwas unexakt gemalten Apfelbaum erkennen und der Begleitbrief sagt ihm: „Ah, da hat ein stolzer Vater ein Bild seiner Tochter verschickt". Der Bruder würde die Anzahl der Äpfel auf diesem Bild zählen und wissen: „Ah, unser subversives Treffen beginnt um 15 Uhr". Schickt man also einen Brief, den anscheinend jeder lesen darf, so wird keiner auf den Gedanken kommen, dort irgendeine geheime Information zu suchen – selbst wenn man so etwa vermutet, weil ersterer gar keine Tochter hat, können sich beide Parteien wohl noch immer rausreden. Wie könnte man definitiv nachweisen, daß die beiden irgendeine geheime Information übertragen haben?

1.2.2 Für unseren Artikel verwenden wir digitalisierte Sprache

In diesem Artikel geht es nicht um Äpfel. Unser „Bild" ist ein ganz normales Telefongespräch. Es kann ganz kurz sein wie zwischen zwei völlig fremden Leuten: „... Tschuldigung, ich habe mich wohl verwählt ... macht nichts ...", oder aber, wie oben das Erzählen über die Entwicklung seines Kindes, Planung eines gemeinsamen Urlaubs, Frage nach neuesten Umsatzdaten, oder ähnlich unauffälligen Inhalt haben. Wir wollen in diesem Artikel klären, in welchem Umfang es möglich ist, zusätzlich zur Sprachinformation beim Telefonieren verdeckte Nachrichten zu übermitteln – ohne daß dies in irgendeiner Weise zu bemerken wäre.

Daß wir gerade das Telefon untersuchen, hat zwei gute Gründe: Erstens ist das Telefon unter allen Telekommunikationsmitteln am weitesten verbreitet (z.B. besitzt nahezu jeder Haushalt in Deutschland eines) und zweitens bietet uns das neue ISDN (Integrated Services Digital Network) der Deutschen Bundespost Telekom die Grundvoraussetzung für unsere Manipulationen: die Telefongespräche werden **digital** übertragen.

Wir hoffen, mit dieser Arbeit einen Teil dazu beizutragen, daß ein Gesetz zur Reglementierung von Verschlüsselung auf breitere Ablehnung stößt. Schließlich ist solch ein Gesetz, sollte es irgendwann verabschiedet werden, eine große Einschränkung der persönlichen Rechte des Einzelnen, und innerhalb weniger Jahre werden die Ermittlungsbehörden erkennen müssen, daß sie sich ins eigene Fleisch geschnitten haben, da alle Bösewichte plötzlich anfangen, ihre Nachrichtenübermittlungen zu verstecken. Als Ergebnis würde somit eine Verfolgung ihrer Aktivitäten schwieriger – statt leichter.

Im folgenden wollen wir das ISDN ein wenig näher betrachten. Wie wir in ein digital übertragenes Telefongespräch Daten, und damit verdeckte Nachrichten, einflechten, werden wir in den darauffolgenden Abschnitten erläutern.

2 Was ist ISDN?

ISDN ist die Abkürzung für Integrated Services Digital Network, und bedeutet diensteintegrierendes digitales Netz.

2.1 ISDN allgemein

Bereits 1980 begann die Deutsche Bundespost Telekom (damals noch Deutsche Bundespost) mit der Planung eines leistungsfähigen digitalen Netzes, das die bis dahin existierenden Netze, das ebenfalls digitale DATEX- und das Telefon-Netz in sich vereinigen sollte. Doch damit nicht genug. Die Post stellte sich ein Netz vor, das als Basis für die verschiedensten Telekommunikationsdienste dienen soll. Man kann auf Grundlage von ISDN Einrichtungen wie Telefax, Telex, Teletex, DATEX-P und eben auch den Telefondienst, sowie in einer weiteren Ausbaustufe (dem sog. Breitband-ISDN) sogar Fernsehen und Radio nutzen.

Die Post begann dann Mitte der 80er Jahre mit dem Aufbau, und bereits 1989 konnten die ersten ISDN-Telefone in Betrieb genommen werden. Den Grundaufbau des Netzes bilden die bereits im Telefonnetz verwendeten Zwillingsdrähte, auf denen zwei 64 kbit-Daten-Kanäle je Richtung, sowie ein Steuerkanal von 16 kbit je Richtung betrieben werden. Es ist auch möglich, bis zu 30 Daten-Kanäle auf diesem Netz zu betreiben, dazu werden jedoch 4 Drähte benötigt. Damit die verschiedenen Geräte, z.B. das Telefon, an das Netz angeschlossen werden können, benötigt man eine spezielle Anschluß-Einrichtung, die sogenannte NT-Dose.

Diese Dose ist nicht nur eine einfache Steckdose, wie dies bei den heutigen Telefon-Dosen der Fall ist, sondern sie ist „intelligent". Sie verwirklicht auf der Teilnehmerseite eine einheitliche und genau definierte Schnittstelle, die sogenannte S_0-Schnittstelle und sorgt so dafür, daß die Teilnehmer allerorts eine einheitliche Netzumgebung vorfinden.

2.2 Für uns wichtig: Digitalisierung von Sprache

Schallwellen, wie Sprache oder Musik, können mit einem Mikrofon in analoge Spannungen umgesetzt werden. Mit einem Oszilloskop dargestellt erscheinen diese Spannungen als übereinandergelagerte Sinuskurven, obwohl es bei dem Gewirr, das man auf dem Schirm sieht, nicht gerade offensichtlich ist, daß es sich dabei wirklich „nur" um Sinuskurven handelt.

Diese Spannungen lassen sich nun mittels eines sogenannten Analog/Digital-Wandlers digitalisieren.

Der A/D-Wandler mißt dabei in kurzen Zeitabständen die momentane Amplitude (also die Größe der Spannung) und wandelt sie in eine Binärzahl (Bild 1). Da dieses „Abtasten" viele Male in der Sekunde geschieht, läßt sich das ursprüngliche Signal anhand der ermittelten Daten wieder rekonstruieren (Bild 2).

Umsetzung einer Dualzahl

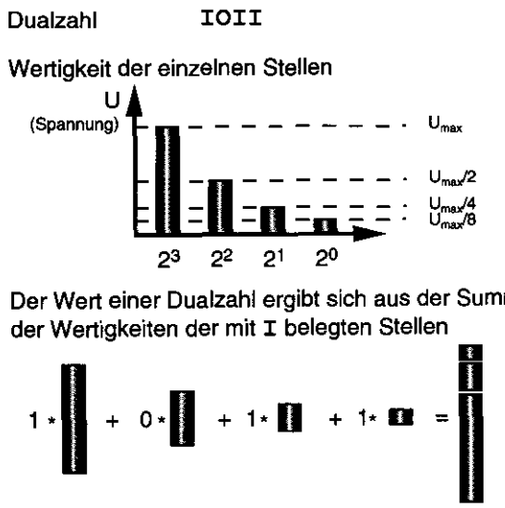

Bild 1 Wandlung von Dualzahlen in analoge Größen und umgekehrt

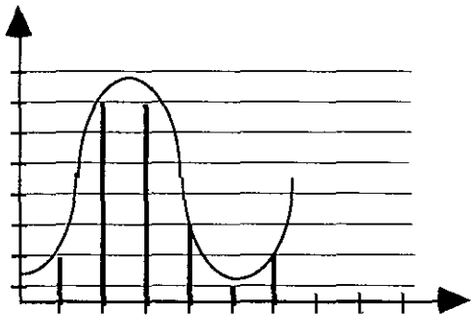

Bild 2 Codierung von Geräuschen als Folge von Binärzahlen

3 Verstecken von Nachrichten in digitalisierter Sprache

Unser Ziel ist es, während eines Telefonats neben der reinen Sprache gleichzeitig und unbeobachtbar beliebige andere Daten zu übertragen. Wir wollen also in der Sprache Daten „verstecken". Rein intuitiv kommt einem dieses ziemlich absurd vor. Wie können während eines Gespräches Daten übertragen werden, ohne daß man dieses bemerken, hören würde? Im Prinzip ist diese Vorstellung auch richtig. Daß es aber doch möglich ist, beruht auf etwas, das man am liebsten gar nicht hätte: Rauschen. Dem Rauschen verdanken wir es, daß wir der Sprache unsere eigenen Daten hinzufügen können, die sich dann, gleich einem Schwarzfahrer unbemerkt und im allgemeinen Getümmel von Geräuschen untergehend, über die Leitung tragen lassen.

Wenn wir also von Verstecken sprechen, so meinen wir nicht das Einfügen von Daten in die Sprache, sondern die Ersetzung von Geräuschen. Wir ersetzen das Rauschen „in der Leitung" durch unser eigenes.

3.1 Unser Begriff des Rauschens

Wir wollen uns in diesem Artikel von einem formalen Begriff des Rauschens lösen und werfen ihn mit Hintergrundgerä(a)uschen zusammen.

In diesem Sinne muß Rauschen nicht immer unbedingt eine „schlechte Leitung" bedeuten (obwohl bei der Verarbeitung von analogen Signalen eigentlich immer welches entsteht). Bei jedem Pusten in den Telefonhörer entsteht es, daher auch durch das normale Ausatmen beim Sprechen. In einem Autotelefon hört man das Rauschen des Fahrwindes und die Motorgeräusche. Eine Telefonzelle am Waldrand überträgt auch Blätterrauschen. Es gibt hierfür Beispiele wie Sand am rauschenden Meer.

Der Gesprächsteilnehmer muß zum Dekodieren der Nachricht dieses Rauschen sozusagen zerpflücken. Das kann er jedoch nur, wenn es genauso ankommt wie es gesendet wurde. Da in analogen Leitungen immer ein Rauschen durch Verstärken des Signals entsteht, ist leider nicht gewährleistet, daß das klappt. Unsere geheimen Daten würden verfälscht. Daher nutzen wir den Telefondienst des ISDN. Dort werden alle Daten digital und *ohne jede ungewollte Modifikation* übertragen. Somit ist es dann tatsächlich möglich, auch unser Rauschen sicher ans Ziel zu bringen.

Eine weitere und sehr wichtige Rauschquelle sind zudem die sogenannten Quantisierungsgeräusche. Jede Digitalisierung ist nämlich grundsätzlich mit einem Fehler behaftet, der mit der Auflösung der Wandlung zusammenhängt. Diese Quantisierungsfehler spielen auch eine Rolle in Hinblick auf die Wahrscheinlichkeit, mit der unsere Manipulationen entdeckt werden können. Doch dazu später noch mehr.

3.2 Grundlegende Ideen zum Verstecken von Nachrichten in digitalisierter Sprache

Es ist äußerst wichtig zu bedenken, wie empfindlich Ohren und Meßgeräte sein können. Wenn wir wirklich unentdeckt bleiben wollen, müssen wir uns daher einige grundlegende Gedanken darüber machen, was wir bei unseren „Manipulationen" zu beachten haben.

Wir möchten die gesprochene Sprache so unauffällig wie möglich ändern. Dazu gibt es sicherlich die verschiedensten Ansätze. Hier stellen wir einen ersten einfachen Ansatz dar. Wenn wir als Verstecker bestimmte Abtastwerte auf einen geraden oder ungeraden Wert setzen, erfährt der Empfänger mit jedem dieser Abtastwerte Information von einem Bit. Das allerdings nur, wenn er weiß, welche Werte geändert wurden. Die geänderten Stellen zeichnen sich für einen Beobachter in keiner Weise gegenüber den Nachbarstellen aus, da das oben erklärte Rauschen sich über die eigentliche Sprache legt.

Es ist sinnvoll, die Stellen, in den Bits eingefügt werden sollen, zufällig zu wählen und den zu übertragenden Bitstrom zufällig aussehen zu lassen. Letzteres würde automatisch erledigt, wenn man nur komprimierte oder verschlüsselte Daten übermittelt. All dies dient vor allem einer erschwerten Nachweisbarkeit einer verstecken Übertragung.

Außerdem ist klar: Je weniger Bits wir verändern, desto weniger Veränderungen können evtl. bemerkt werden.

Man kann sich zusätzlich beliebig viele verschiedene Bedingungen überlegen, unter denen eine Änderung eines Amplitudenwertes erfolgen oder nicht erfolgen darf. Mit der amplitudenabhängigen Modifikation stellen wir ein solches Bedingungspaar vor. Sollten Gutachten veröffentlicht werden (z.B. während eines Gerichtsverfahrens) zum Nachweis

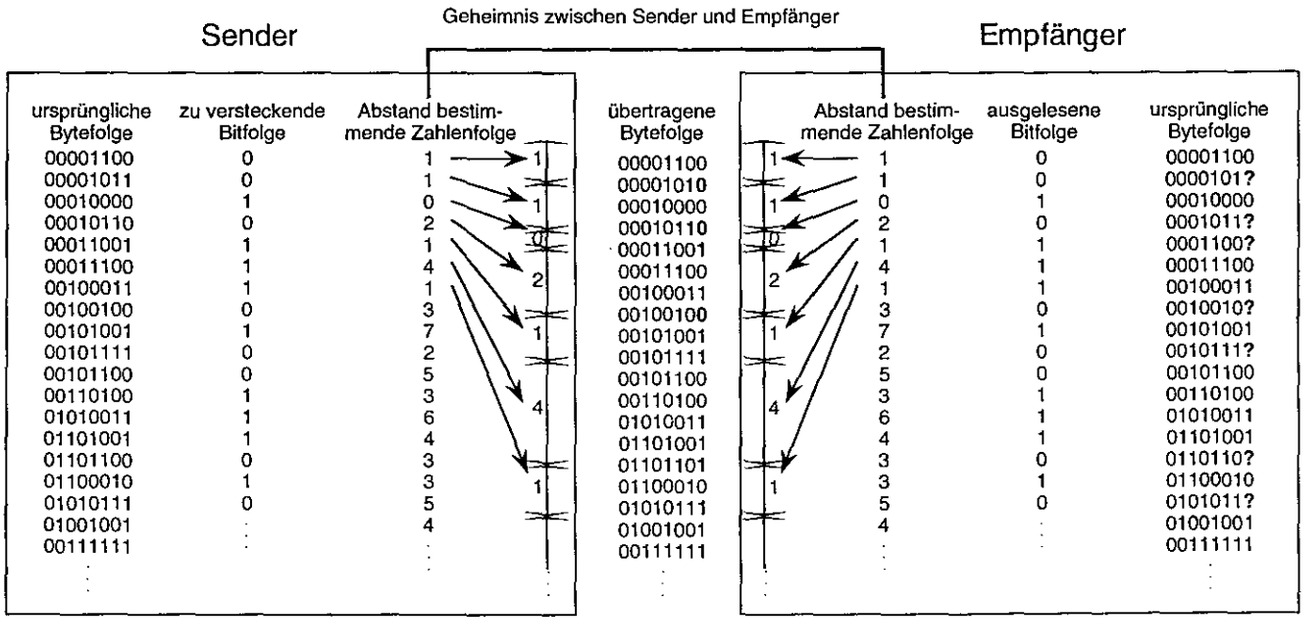

Bild 3 Sender und Empfänger bekannte Zufallswerte steuern den Abstand zwischen den geänderten Bits

einer versteckten Übertragung, so können die dort verwendeten Algorithmen, die zur Entdeckung geführt haben, implementiert und somit als a-priori-Prüfverfahren verwendet werden, zur Ermittlung ob eine Veränderung eines Amplitudenwertes ‚erlaubt' ist oder nicht. Steganographie ist also, falls überhaupt, keinesfalls auf Dauer vor Gericht beweisbar.

3.3 Das Verfahren

Das Prinzip unserer Simulation besteht darin, aus einer Datei mit Sprachinformationen an beliebigen Stellen das niederwertigste Bit auf einen Wert eines Bitstromes einer zweiten Datei setzen zu können. Diese zweite Datei stellt die unbemerkt zu übertragenden Informationen dar. **Der Abstand zwischen zwei geänderten Bits wird durch einen Zufallswert festgelegt.** Dessen Maximalwert ist ein Parameter für unser Verfahren. Ermittelbar ist er entweder aus einer Datei mit Zufallszahlen, die dann auch der Empfänger haben muß, oder beide nutzen einen Pseudo-Zufalls-Generator als die elegantere Möglichkeit, diese Zufallszahlen zu erzeugen. Letzteres hätte den Vorteil, daß nur noch ein ‚Startwert' zwischen den Gesprächspartnern übermittelt werden müßte. Ein solcher Startwert kann verstanden werden wie ein Paßwort, das man eingeben muß, um an die eigentlichen Informationen gelangen zu können.

Will man sich auf ein vermeintliches Geräusch konzentrieren, so wird man automatisch seine Stimme senken, um sich besser darauf konzentrieren zu können. Womöglich bittet man sogar Umstehende, für einen Moment still zu sein. Wir haben uns überlegt, daß es Sinn macht, **unsere Daten nur dann zu übertragen, wenn gerade gesprochen wird.** Dazu implementieren wir ein Verfahren, bei dem ein Byte nur dann modifiziert wird, wenn die Momentanlautstärke einen Schwellwert unter- bzw. überschreitet. Dieses Verfahren schließt keines der anderen aus. Sie können parallel angewandt werden.

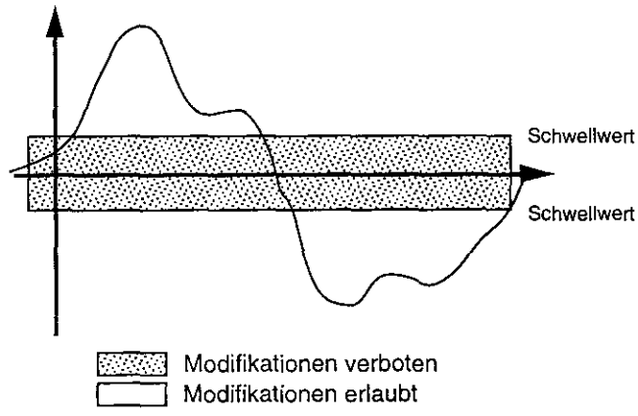

Bild 4 Nur große Amplitudenwerte werden ein bißchen geändert

4 Bestimmung der Qualität

Wir haben ein kleines Programm – genannt DigiStilz – zur Simulation einer solchen versteckten Übertragung auf einem Apple Macintosh implementiert. Mit DigiStilz konnten erste Erfahrungswerte für die Hörbarkeit einer solchen Übertragung ermittelt werden.

4.1 Grundlegende Fragen zur Messung der Qualität

Um die Qualität einer Maschine zu bestimmen, muß man erstens wissen, welchen Zweck sie haben soll, und zweitens, wie man die Erfüllung dieses Zwecks messen kann. Mit dem Zweck ist es einfach: wir haben eine Maschine, die während eines Telefongesprächs unbemerkt zusätzliche Daten übertragen soll. Mit der Messung wird es schon schwieriger. Wie kann man messen, ob übertragene Sprache bemerkbar verfälscht wurde?

4.2 Tests und Messungen

4.2.1 Hörtests

Als objektiv kann man das Gehör wohl nicht bezeichnen, aber das einfachste und sicher nicht das schlechteste Hilfsmittel zum Überprüfen unserer Veränderungen waren unsere Ohren. Durch Hörproben haben wir zunächst festgestellt, ob unsere Dateien ‚hörbar' waren, was darauf schließen lassen konnte, daß unsere gewählte Art des Versteckens nicht ausreichend war. Ein wichtiges Kriterium für die Güte des Versteckens war der „Leer-Test". Für ihn nahmen wir eine Sounddatei, die durch die Aufnahme „absoluter Ruhe" entstand. Wenn man also etwas hören konnte, dann durch unsere Modifikation. Auch haben wir in verschiedenen natürlichen Umgebungen Gespräche aufgenommen. Wir wollten testen, welche Einflüsse Hintergrundgeräusche auf die Empfindlichkeit für unsere Störungen haben.

Man muß bei der Beurteilung der modifizierten Aufnahmen immer daran denken, daß bei der Beurteilung des Verstekkens immer mit dem Original verglichen wurde. Das kann ein Mithörer jedoch nicht, so daß wir mit unseren Beurteilungen auf der vorsichtigen Seite liegen.

4.2.2 Sonogramm

Mit dem Programm „SoundEdit" haben wir die Frequenzanteile in unseren modifizierten Sounddateien analysiert. Das Sonogramm filtert die einzelnen Frequenzanteile aus dem Sprachspektrum heraus und trägt die Intensitäten, mit denen die Anteile vertreten sind, mit der Zeit ab. Mißt man mit einem Sonogramm eine einzelne Frequenz, z.B. 1000 Hz, so wird man bei der 1000 Hz-Marke eine klare Linie erkennen. Mit diesem Programm konnten wir also feststellen, ob und wie stark unsere Manipulationen den Charakter unserer Geräusche veränderten, und hatten somit ein gutes Meßwerkzeug an der Hand.

So wurde z.B. bei der Übertragung von ASCII-Text mit konstantem Abstand 1 ein (übrigens auch hörbares) Datenrauschen als konstante Linie im Spektrogramm deutlich. Interessant ist, daß bereits mit variablem Abstand von maximal zwei diese nicht mehr erkennbar war. Das Spektrogramm liefert auch tatsächlich immer minimalste Unterschiede, nicht jedoch irgendwelche erkennbaren Charakteristika, die typisch wären für versteckte Sounds. In der Auswertung wird auf das Sonogramm nicht weiter eingegangen.

4.2.3 Verteilungen

Messen kann man natürlich auch nur das, was vorhanden ist. Wir wollten also auch die Anzahl der Bits, die im Mittel bei den jeweiligen Verfahren verändert werden, als Maßstab für die Qualität verwenden. Dazu machen wir zunächst die vereinfachende Annahme, daß wir auch tatsächlich ein Bit ändern, wenn wir ein Bit ersetzen. Wenn wir das Verfahren mit den zufälligen Abständen verwenden und die Bits der die Einfügeabstände steuernden Datei gleichverteilt und zufällig sind, können wir davon ausgehen, daß es die Abstände ebenfalls sind. Bei einem maximalen Abstand von d_{max} erhalten wir somit im Mittel einen Abstand von $(d_{max} + 1)/2$.

Nehmen wir z.B. an, daß wir einen Maximalabstand von 4 Bits gewählt haben. Im Mittel verändern wir dann jedes 2,5 te Bit, bei einem Maximalabstand von 32 Bit jedes 16,5 te Bit, usw. Wie man sieht, können wir durch die Vergrößerung des Abstandes sehr kleine Änderungswahrscheinlichkeiten erreichen.

Hieraus absolute Aussagen über die Entdeckbarkeit unserer Änderungen abzuleiten, ist jedoch nur sehr bedingt möglich. Zwar ist es so, daß Texte, deren Bits *nicht* gleichverteilt und zufällig sind, eher auffallen, als solche, die es sind, doch hieraus Rückschlüsse auf Entdeckbarkeit zu ziehen, würde bedeuten, über die Eigenschaften der Originaldaten hinwegzuschauen. Konkrete Aussagen, die auf Verteilungsannahmen beruhen, können wir nur dann machen, wenn wir auch in der Lage sind, gesicherte Behauptungen über die Verteilungen der digitalisierten Geräusche aufzustellen. Wir sprachen davon, daß Rauschen meist gleichverteilt ist. An dieser Stelle sind wir gezwungen, einen Schritt rückwärts zu tun und zuzulassen, daß Korrelationen zwischen den Bits der Originaldaten existieren können. Wir wollen uns davon jedoch nicht entmutigen lassen und behaupten, daß wir trotzdem in der Lage sind, unsere Daten unbemerkt einzufügen. Warum wir dies tun können, soll in dem nächsten Kapitel erläutert werden.

Für unseren einfachen Versteckalgorithmus macht eine solche Überlegung Sinn. Man kann sich auch noch andere geschicktere Verfahren ausdenken, mit denen Daten einzuflechten wären. Beispielsweise kann man ein Byte um mehrere Werte verändern und, damit das nicht so auffällt, die Umgebung dieser Änderung anpassen. Es werden also besonders viele Bits geändert, die Auffälligkeit muß dadurch aber nicht größer werden.

4.3 Resultate bei den Verfahren

Es liegt an der Thematik, daß es schwer ist, absolute Aussagen über Qualitäten zu machen. Wir hoffen, dies deutlich gemacht zu haben. Insbesondere ist es kaum möglich, genau abzuschätzen, welche Meßmethoden noch angewendet, oder gar noch erdacht werden könnten, um unsere Änderungen zu entdecken. Wir wollen uns davon aber nicht abschrecken lassen und ein möglichst objektives Bild davon vermitteln, wie gut wir unsere Daten verstecken können.

4.3.1 Sonogramm

Durch direkten Vergleich konnten wir feststellen, welche Frequenzen ihre Intensität durch unsere Manipulationen geändert hatten und wie stark. Absolute Aussagen zu machen, ist hier allerdings schwer. Die Antwort auf die Frage, ab wann diese Änderungen hörbar sind, ist zunächst stark vom jeweiligen Zuhörer abhängig.

4.3.2 Statistischer Ansatz

Um einen vernünftigen statistischen Ansatz zu finden, d.h., einen Ansatz, der es uns erlaubt, konkrete Aussagen über die Güte unserer Änderungen von Geräusche zu machen, muß man sich zunächst die Frage stellen, wie die Originaldaten ‚verteilt' sind.

Existieren vielleicht zeitliche Korrelationen zwischen den Daten, oder sogar solche zwischen den einzelnen Bits einer einzigen Abtastung?

Diese Fragen zu beantworten, würde bedeuten, alle nur erdenklichen Einflußgrößen zu spezifizieren und zu untersuchen. Man muß sich dazu vor Augen halten, daß wir es bei Geräuschen nicht mit einfachen Modellen zu tun haben, sondern daß hier die Möglichkeiten für Einflüsse schier endlos sind.

Eines können wir jedoch mit Sicherheit annehmen: Die Folge von Zahlen, die bei der Digitalisierung entstehen, ist nicht deterministisch. Es *muß* also ‚Zufälligkeiten' in der

Verteilung der Bits geben. Dies aber wiederum bedeutet, daß es möglich sein *muß*, Bits ‚zufällig' zu ändern. Wir können also davon ausgehen, daß wir mit unseren Änderungen unentdeckt bleiben, solange wir diese nur selten genug machen und bestehende Korrelationen nicht ‚zerstören'.

Ein einfaches Beispiel für den bestehenden Nichtdeterminismus sind Quantisierungsfehler. Um dies zu verdeutlichen, wollen wir ein Beispiel geben:

Das zu digitalisierende Signal beträgt zu einem bestimmten Zeitpunkt 4,9 mV. Der verwendete lineare 8 Bit A/D-Wandler kann Spannungen in 2 mV-Schritten digitalisieren (dies entspricht einer maximalen zu digitalisierenden Spannung von etwa 0,5 V). Der gewandelte Wert beträgt damit 2, oder 10 in Binärdarstellung.

Ändern wir diesen Wert zu binär 11, oder 3 (dezimal), dann wird zwar eine Spannung von 6 mV, anstatt 4 mV bei der Rückumwandlung erzeugt, aber der Fehler, der dabei gemacht wird, ist nur um 0,2 mV größer, als bei dem ursprünglich ermittelten Wert.

Was aber passiert, wenn neue statistische Verfahren entwickelt werden, die gut genug sind, um unsere Veränderungen aufzuspüren? Auch hier können wir mit dem Nichtdeterminismus argumentieren. Jedes solche Verfahren muß mit diesem Nichtdeterminismus arbeiten, was bedeutet, daß es ‚Lücken' aufweisen muß, d.h., daß es Bit-Konstellationen akzeptieren muß, die nicht eindeutig als verändert erkannt werden können. Diese ‚Lücken' könnten dann, unter der Voraussetzung, daß dieses Verfahren bekannt wird, in unsren Algorithmen verwendet werden, um unsere Datenbits doch noch ändern zu können.

Letzte Sicherheit, daß wir nicht entdeckt werden, können wir also vermutlich nicht erreichen. Indem wir aber die sicher existierende Unsicherheit trotz (eventuell) bestehender Korrelationen nutzen, um unsere Daten einzufügen, dürfen wir annehmen, daß wir letztendlich auch sehr großem Aufwand widerstehen können.

4.3.3 Zu den Versuchsreihen

Für die Hörexperimente, die wir machten, haben wir zunächst eine Anzahl von ‚Sounds' mit unserem Programm aufgenommen. Dazu verwendeten wir ein PowerBook, um Gespräche in verschiedenen Umgebungen aufnehmen zu können. Insgesamt nahmen wir auf diese Weise 12 ‚Sounds' von jeweils 10–30 Sekunden Länge auf: in großen und in kleinen Räumen, mit viel und wenig Hintergrundgeräuschen, in einer Telefonzelle, auf der Straße und in der Natur. Wir wollten auf diese Weise verschiedene Standardsituationen simulieren, die während eines Telefonats auftreten. Alle Experimente, bis auf das letzte, führten wir mit ein und demselben Text durch, so daß wir von hier aus keine Ergebnisverfälschungen zu erwarten hatten.

Wir starteten mehrere Versuchsreihen:

1. Zunächst beschränkten wir uns auf das Verfahren mit konstanten Abständen. Wir vergrößerten bei jedem ‚Sound' die Abstände so lange, bis wir keine Unterschiede mehr zwischen dem Original und demjenigen mit den eingefügten Daten hören konnten. Hierdurch verschafften wir uns einen Überblick über den Einfluß der Umgebung, und damit der Geräusche selbst auf die Qualität der Verfahren. Diese Phase bildete außerdem die Grundlage für die nächsten Versuchsreihen.

2. Hier wendeten wir das Verfahren mit den zufälligen Abständen an. Dazu begannen wir mit dem Abstand, bei dem mit konstanten Abständen gerade keine Unterschiede mehr zu hören waren, und verkleinerten diese dann, bis die ‚Kopie' zu rauschen begann. In dieser Versuchsreihe ging es uns um die Frage, um wieviel besser das Verfahren mit den variablen Abständen gegenüber dem mit den konstanten ist.

3. In der dritten Versuchsreihe nahmen wir einen Sound und fügten mit konstantem Abstand 1 den Text mit verschiedenen Schwellwerten ein, vgl. § 4.3.3.3.

4. In der vorletzten Versuchsreihe suchten wir (zugegebenermaßen auf sehr heuristische Weise) ein ‚optimales' Verfahren. Dazu drehten wir an den verschiedenen Verfahren und Parametern, bis wir der Meinung waren, daß hier besonders gute Ergebnisse erzielt wurden. Gut heißt, daß die Änderungen nicht zu hören waren, aber dennoch möglichst viele Daten versteckt übertragen werden konnten.

5. Hier untersuchten wir Einflüsse der Verteilung der Bits in den zu übertragenden Daten auf die Qualität unserer Änderungen. Wie in der ersten Versuchsreihe erhöhten wir die konstanten Abstände, bis keine Änderungen mehr zu hören waren. Allerdings beschränkten wir uns auf einen ‚Sound'.

Natürlich ist dies nur eine kleine Auswahl von Versuchsreihen, die möglich bzw. sinnvoll sind. Wir mußten uns jedoch für einige Dinge entscheiden, da mit den Möglichkeiten auch die Anzahl der Versuchsreihen steigt. Alles in allem haben wir über 200 Hörexperimente gemacht.

In den nächsten Abschnitten wollen wir die Ergebnisse erläutern und Schlußfolgerungen ziehen. Die Tabelle zeigt die Ergebnisse im Überblick.

4.3.3.1 Versuchsreihe 1

Dieser Test sollte uns Aufschlüsse über die Einflüsse der Umgebungsgeräusche geben. Wir waren, obwohl uns klar war, daß die Umgebung sicher wichtig ist, etwas erstaunt, wie einflußreich dieser Faktor ist. So konnten wir in dem Sound, den wir direkt an der Straße aufgenommen haben, 2750 Bits pro Sekunde einfügen (jedes 8 Bit), während in einer sehr ruhigen Umgebung gerade noch ca. 170 Bits (jedes 128. Bit) vertretbar waren. Fazit: hier lohnen sich ganz sicher weitere Betrachtungen in der Zukunft.

4.3.3.2 Versuchsreihe 2

Wir hatten hier absichtlich eine Gegenüberstellung der beiden Verfahren im Auge. Es liegt ganz klar auf der Hand: Das Verfahren mit den zufälligen Abständen ist *wesentlich* besser als das Verfahren mit konstanten Abständen. In den meisten Sounds konnten wir mit zufälligen Abständen etwa doppelt so viele Daten verstecken.

4.3.3.3 Versuchsreihe 3

Diese Versuchsreihe bestätigte unsere Vermutung, daß es sinnvoll ist, nur dann Daten versteckt zu übertragen, wenn das Grundgeräusch laut ist. Durch Angabe eines unteren und eines oberen Schwellwertes, die beide zwischen 0 und 128 einstellbar sind, konnten wir darauf Einfluß nehmen (Bild 5). Eine Modifikation wurde nur dann vorgenommen, wenn die Amplitude zwischen diesen Schwellwerten lag.

Bei einem oberen Schwellwert von 5 (wobei der untere auf 0 gesetzt wurde) sind die relativen Fehler der Änderungen groß – und dies war tatsächlich zu hören. Hingegen konnte man einen unteren Schwellwert von 80 benutzen (und einen oberen von 128), ohne daß Änderungen hörbar waren.

4.3.3.4 Versuchsreihe 4

Als Referenz-Sound wählten wir denjenigen mit den ‚schlechtesten' Eigenschaften, also den, bei dem man die größten

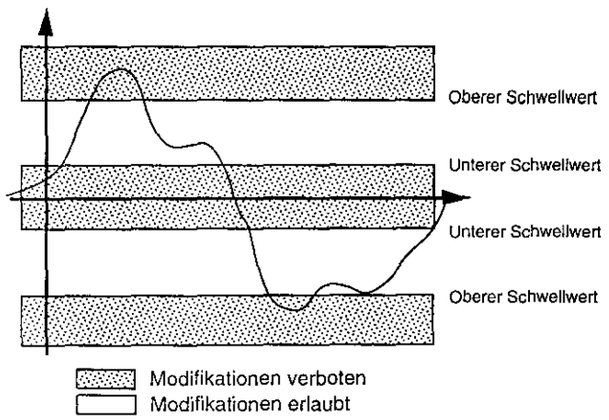

Bild 5 Verbot von Modifikationen abhängig von Schwellwerten

Abstände wählen mußte. Wir begannen mit dem variablen Abstand, der die Änderungen gerade hörbar werden ließ. Dann setzten wir den Schwellwert solange herauf, bis wir unsere Änderungen nicht mehr hören konnten. Dies wiederholten wir mit verschiedenen kleineren Abständen. Zum Vergleich ermittelten wir für die verschiedenen Abstände die Menge an Daten, die wir in dem Sound unterbringen konnten. Mit variablen Abständen von maximal 2 und einem unteren Schwellwert von 60 konnten wir im Schnitt mehr Daten übertragen, als mit allen anderen Verfahren, bei denen keine Geräusche durch die Änderungen auftraten. Dieses Ergebnis war insofern voraus zu sehen, als die Abstände im wesentlichen die Menge an Daten bestimmen, die übertragen werden können. Da wir dieses Verfahren an dem am schlechtesten konditionierten Sound entwickelten, konnte es auf alle anderen Sounds (erwartungsgemäß) angewendet werden.

4.3.3.5 Versuchsreihe 5

In allen vorherigen Versuchsreihen arbeiteten wir mit einem Text, der aus einer zufällig gleichverteilten Folge von Buchstaben, Ziffern und einigen Standard-Sonderzeichen bestand. In einem letzten Experiment fügten wir einen Text mit nur Nullen, einen ASCII-Text, einen Text mit abwechselnden Nullen und Einsen und einen Text, der aus einem zufällig gleichverteilten Strom von Einsen und Nullen bestand in die Sounds ein. Erstaunlicherweise bekamen wir in den Ergebnissen praktisch keine Unterschiede. Die Verteilung der Texte scheint also nicht sehr signifikant zu sein. Vermutlich liegt dies daran, daß bei größeren Abständen die Werte der Bits keine allzu große Rollen spielen.

4.3.3.6 Anmerkungen

Unsere Versuchsergebnisse haben sich nicht immer mit unseren Erwartungen gedeckt. Einiges war doch sehr überraschend; so entdeckten wir, daß die Geräusche, die bei zu niedrigen bzw. zu hohen Schwellwerten entstanden, eher Verzerrungen glichen als Rauschen. Wir haben auch nicht damit gerechnet, daß die Umgebungsgeräusche einen so starken Einfluß auf die Hörbarkeit unserer Veränderungen haben.

Eine Bemerkung noch zu dem Vorgehen bei den Versuchen: wenn man sich einige Stunden Sounds anhört, dann kann es schon mal vorkommen, daß man einige Dinge nicht mehr so gut wahrnimmt. Trotzdem haben wir uns große Mühe gegeben, so objektiv wie nur möglich zu sein. So haben wir weitgehend ‚Blindtests' gemacht. Dabei mußte einer raten, welcher von beiden das Original und welches die ‚Fälschung' war. Außerdem haben wir mit Kopfhörer gearbeitet, um eine möglichst gute Qualität bei der Wiedergabe zu haben. Viele Tests haben wir am nächsten Tag noch einmal wiederholt, um Fehler zu verringern.

Abstand	Sound 1		Sound 2		Sound 3		Sound 4	
	konstant	zufällig	konstant	zufällig	konstant	zufällig	konstant	zufällig
1	++++	--	++++	--	++++	--	--	--
2	++++	--	++++	--	++++	--	++++	--
4	++++	++++	+++	--	+++	+++	++++	--
8	+++	+++	+	+++	++	++	++++	--
16	+++	+++	0	++	+	+	+	--
32	+++	++	--	0	0	0	+	+
64	+	0	--	--	--	--	0	0
128	0	--	--	--	--	--	--	--

Abstand	Sound 5		Sound 6		Sound 7		Sound 8	
	konstant	zufällig	konstant	zufällig	konstant	zufällig	konstant	zufällig
1	++++	--	++++	--	+++	--	++++	--
2	++++	--	++++	--	++	+++	+++	--
4	+++	--	+++	--	+	++	++	--
8	++	--	+++	--	0	+	+	++
16	+	+	++	--	--	0	0	0
32	+	+	+	+	--	--	--	--
64	0	0	0	0	--	--	--	--
128	--	--	--	--	--	--	--	--

++++ = gut hörbar ++ = kaum hörbar 0 = nicht hörbar
+++ = hörbar + = gerade noch hörbar -- = nicht getestet

Tabelle für 3. Versuchsreihe

Sound (1), konstanter Abstand 1, variable Schwellwerte (Einstellbar zwischen 0 und 128)

Schwellwerte unten – oben	Ergebnis
0 – 1	hörbar
0 – 5	hörbar
0 – 10	hörbar
120 – 128	nicht hörbar
100 – 128	nicht hörbar
80 – 128	nicht hörbar
70 – 128	praktisch nicht
60 – 128	kaum
50 – 128	erste Verzerrungen
40 – 128	hörbare Verzerrungen

Tabelle für 4. Versuchsreihe

Sound (1), Abstände zufällig, variable Schwellwerte

Abstand (z)uf/(k)onst	max	Schwellwerte unten – oben	Ergebnis
z	16	30–128	hörbar
z	16	40–128	nicht hörbar
z	8	40–128	praktisch nicht
z	4	40–128	hörbar
z	4	50–128	nicht hörbar
z	2	50–128	praktisch nicht
z	2	60–128	nicht hörbar
k	1	80–128	nicht hörbar

Tabelle für 5. Versuchsreihe

Sound (5), konstanter Abstand, variable, verschiedene Texte

Abstand	Ergebnis			
	Null-Text	Null-Eins-Text	ASCII-Text	gleichverteilter Text
4	hörbar	–	–	–
8	hörbar	–	–	–
16	kaum	kaum	kaum	kaum
32	praktisch nicht	praktisch nicht	praktisch nicht	praktisch nicht
64	nicht hörbar	geraten/eingebildet	nicht hörbar	nicht hörbar

–: nicht betrachtet

4.3.3.7 Ergebnisse

Soundbeschreibung

(1) großer Raum mit relativ wenigen Nebengeräuschen
(2) ‚Natur' mit starken Hintergrundgeräuschen
(3) ‚Natur' mit wenigen Hintergrundgeräuschen
(4) Telefonzelle in ruhiger Lage
(5) Telefonzelle sehr ruhig
(6) Telefonzelle an befahrener Straße
(7) stark befahrene Straße (direkt am Bürgersteig)
(8) kleinerer ruhiger Raum

Gegenüberstellung von der 1. und 2. Versuchsreihe

1. Versuchsreihe: **konstante** Abstände, keine Schwellwerte
2. Versuchsreihe: **zufällige** Abstände (angegeben ist der maximale Abstand, so daß im Mittel fast doppelt so viele Bits wie bei gleichem konstanten Abstand eingefügt werden), keine Schwellwerte

5 Aussichten

5.1 Ist das Verfahren in der Realität einsetzbar?

Unser System wäre sicher kein relevanter Beitrag zur bestehenden Diskussion, wäre es nicht auch zur **Realzeit-Anwendung** geeignet. Wir sind also der Frage nachgegangen, welche Möglichkeiten uns zur Verfügung stehen, dieses System in das ISDN zu implementieren.

Die Macintosh-Toolbox unterstützt das Verarbeiten von Bitfolgen ausgezeichnet. Mit deren Hilfe war es uns möglich, einen Datendurchsatz weit über dem notwendigen Maß von 8000 Bytes pro Sekunde zu erhalten. Weniger hörbar bedeutet weniger zu verändern. Das Programm wird daher bei erhöhten Anforderungen schneller statt langsamer, wenn es auch paradox klingen mag.

5.2 Der Schluß

Steganographie ist die Wissenschaft vom Verstecken von Daten. Wir haben hier ein anschauliches Beispiel gegeben, wie man im ISDN Daten in Telefongesprächen verstecken kann. Wenn wir es auch nicht als Echtzeit-Lösung vorgestellt haben, so ist hoffentlich deutlich geworden, daß dies kein prinzipielles Problem darstellt. Wir sind der Meinung, daß ein Softwarehaus wenige Monate bräuchte, um solch eine Anwendung unter Verwendung geeigneter Hardware zu realisieren. Mit Hilfe eines solchen Apparates ist eine Verfolgung der Aktionen auf diesem Wege nicht mehr möglich.
Es gibt jedoch durchaus noch weitere Möglichkeiten für das Verstecken von Daten. Stego für den Macintosh und eine durch die Independent JPEG Group zur Verfügung gestellte Software zum Verstecken von Daten in Bildern. Von Peter Wayner wurde eine Methode vorgestellt, um aus Daten einen Text zu konstruieren. Der Text ist zwar ohne jeden Sinn, man merkt es aber erst beim näheren Hinsehen, und doch enthält er alle Informationen des Urtextes.
Zusätzlich zum Verstecken der Daten mittels Steganographie können die geheimzuhaltenden Nachrichten vorher mittels **Kryptographie** verschlüsselt werden. Dann sind die versteckten Daten, selbst wenn sie entdeckt werden sollten,

für Fremde nicht interpretierbar. Damit ist die Existenz einer geheimgehaltenen versteckten Nachricht noch schwerer zu beweisen – ein Verbot von Steganographie in einem Rechtsstaat also überhaupt nicht durchsetzbar.

Ein Gesetz zur staatlichen Kontrolle von Kryptographie zwingt Verbrecher geradezu zu solchen Schritten, ist daher gefährlich und darf keinesfalls in Kraft treten.

Literatur

Peter Bocker: ISDN – das diensteintegrierende digitale Nachrichtennetz; Konzept, Verfahren, Systeme; Springer-Verlag, 1987

Joachim Claus (Hrsg.): CCITT-Empfehlungen-ISDN; R. v. Decker's Verlag, G. Schenck, 1992

U. Tietze, Ch. Schenk: Halbleiter-Schaltungstechnik; Springer-Verlag, 1989

Fellbaum: Elektronische Sprachverarbeitung, Verfahren, Anwendungen, Wirtschaftlichkeit; Franzis-Verlag, München, 1991

Independent JPEG Group: JPEG Announcement; auf ftp-Servern des InterNet wie ghost.dsi.unimi.it in/pub/security

Peter Wayner: Mimic Functions; Cryptologia XVI/3 1993, Seiten 193–213

Romana Machado: Stego; Paradigm Shift Research, Nov 93; auf ftp-Servern des InterNet wie ghost.dsi.unimi.it in/pub/security

Stichwörter: Codierung, Digitalisierung, Geräusche, Hintergrund-Geräusch, Hörtests, Kryptographie, ISDN, Rauschen, Schwellwertverfahren, Sonogramm, Steganographie, Verschlüsselung, versteckte Übertragung