1 Motivation

1.1 Heutige und geplante Kommunikationsnetze

Immer mehr benutzen wir öffentliche Kommunikationsnetze:

- Hörfunk, Fernsehen, Videotext z. B. sind Dienste, die (genauer: deren Informationen) heute vorwiegend über das Rundfunksendernetz der Deutschen Bundespost, mehr und mehr aber über das entstehende Breitbandkabelverteilnetz **verteilt** werden. Zweck des Breitbandkabelverteilnetzes ist die Verbesserung der Dienstqualität durch Erhöhung der verfügbaren Bandbreite: mehr empfangbare Fernsehprogramme heißt dann Kabelfernsehen, größeres Informationsangebot bei Videotext heißt dann Kabeltext.

- Fernsprechen, Bildschirmtext, elektronisches Postfach (TELEBOX) für elektronische Brief- und Sprachpost, Fernschreiben (TELEX, TELETEX), Fernkopieren (TELEFAX) und Fernwirken (TEMEX) sind Dienste, die heute über das Fernsprechnetz bzw. das digitale Text- und Datennetz der Deutschen Bundespost **vermittelt** werden. Das heute im Teilnehmeranschlußbereich noch überall analoge, im Fernbereich bereits weitgehend digitale Fernsprechnetz wird seit 1988 in einigen Ortsnetzen und ab etwa 1993 in der ganzen Bundesrepublik auch im Teilnehmeranschlußbereich nach und nach digitalisiert und danach jeweils alle diese Dienste mit höherer Qualität erbringen.
  Das heutige analoge Fernsprechnetz und das dieselben Teilnehmeranschlußleitungen benutzende digitale Fernsprechnetz sind **schmalbandige** Netze, d. h. sie sind im Gegensatz zu **breitbandigen** Netzen nicht in der Lage, Bewegtbilder (z. B. Fernsehen) zu übertragen.

Zur Zeit benutzen wir also zwei grundsätzlich verschiedene Typen von Netzen:

- **Verteilnetze**, in denen alle Teilnehmerstationen vom Netz dasselbe erhalten und jeder Teilnehmer lokal auswählt, ob und, wenn ja, was er tatsächlich empfangen will, und

- **Vermittlungsnetze**, in denen jede Teilnehmerstation vom Netz individuell nur das erhält, was der Teilnehmer angefordert oder ein anderer Teilnehmer an ihn gesendet hat.

In fast allen realisierten und geplanten öffentlichen Verteilnetzen findet Kommunikation nur in einer Richtung, vom Netz zum Teilnehmer, statt [Krat_84, DuD_86]; in Vermittlungsnetzen wird generell in beiden Richtungen kommuniziert.

Da langfristig ein Netz für alle Dienste, ein sogenanntes **diensteintegrierendes** Netz, zumindest im Teilnehmeranschlußbereich preiswerter als mehrere verschiedene Netze ist, und da alle verteilten Dienste auch vermittelt werden können, strebt die Deutsche Bundespost (DBP) an, beginnend ab 1992 alle Dienste in einem Netz zu vermitteln [Schö_84, ScSc_84, ScS1_84, Rose_85, Thom_87].

Damit auch breitbandige Dienste vermittelt zum Teilnehmer übertragen werden können, müssen neue Teilnehmeranschlußleitungen (**Glasfaser**) verlegt werden. Nach und nach wird dadurch ein Breitbandkabelverteilnetz überflüssig. Über ein breitbandiges diensteintegrierendes Vermittlungsnetz können nicht nur alle Dienste angeboten werden, die über ein Breitbandkabelverteilnetz und ein schmalbandiges Vermittlungsnetz zusammen angeboten werden können, sondern auch noch zusätzlich Dienste, die breitbandige Kommunikation zwischen Teilnehmern erfordern, z. B. Bildfernsprechen.

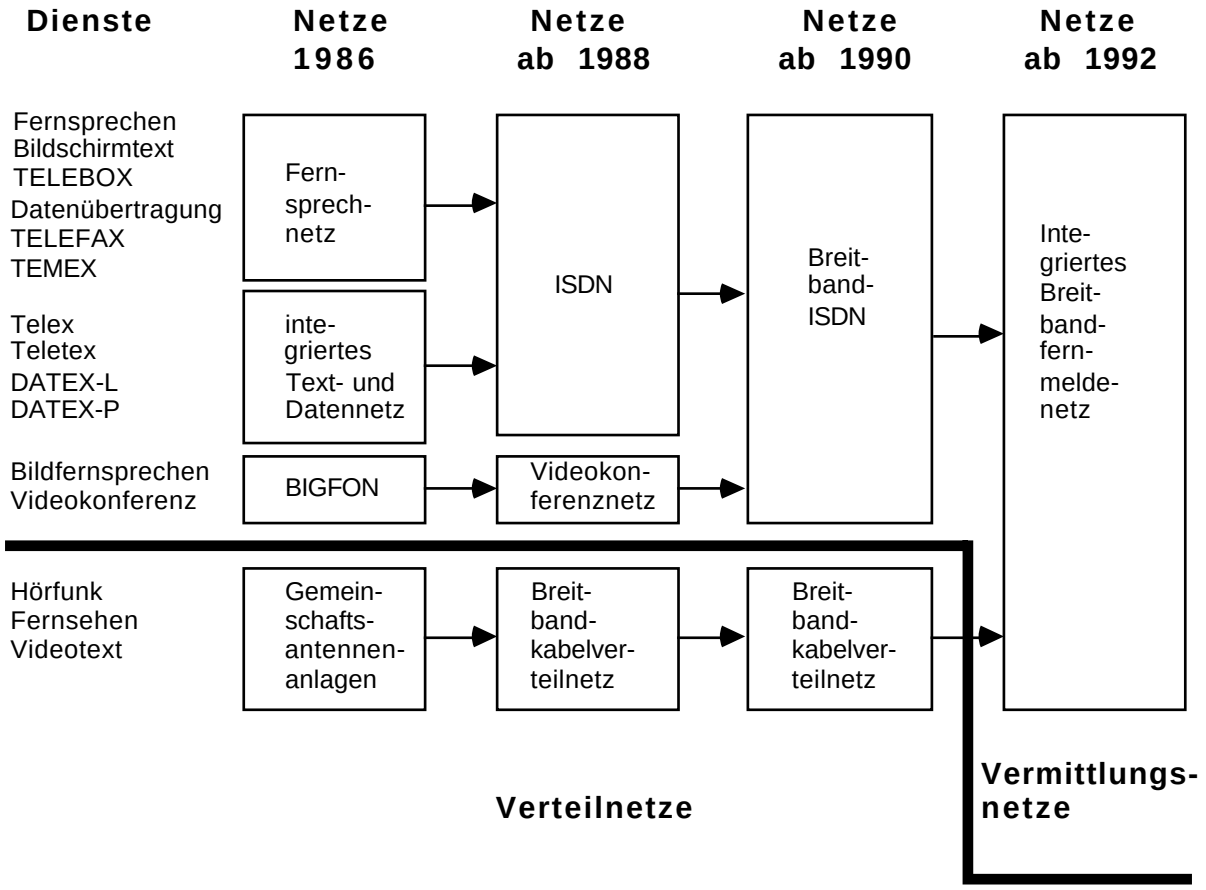


Bild 1: Geplante Entwicklung der Netze der Deutschen Bundespost

Da digitale Werte in modernen technischen Systemen nicht nur leichter übertragen, sondern auch leichter verarbeitet, insbesondere vermittelt, werden können, wird das diensteintegrierende Netz ein von Teilnehmerstation zu Teilnehmerstation **digitales** Netz sein. Solch ein Kommunikationsnetz nenne ich **„diensteintegrierendes Digitalnetz“**, während ich die Abkürzung

ISDN (Integrated Services Digital Network) nur für das von der DBP geplante, weiteren Einschränkungen genügende diensteintegrierende Digitalnetz verwende. Nach [Schö_86] erspart die Digitalisierung schon heute 40% der Kosten bei der Fernvermittlungstechnik und ist bei der Ortsvermittlungstechnik mit fallender Tendenz schon heute nicht teurer. Beide bringen je 65% Raumersparnis, was die Hochbaukosten enorm senkt, und der schnellere Verbindungsaufbau und die größere Netzflexibilität sparen 15 bis 20% der erforderlichen Übertragungskapazität.

Ersetzt das breitbandige ISDN (abgekürzt Breitband-ISDN oder B-ISDN [Steg_85]) das Breitbandkabelverteilnetz, wodurch einerseits der Unterhalt des Breitbandkabelverteilnetzes eingespart und andererseits hochauflösendes Fernsehen (High Definition TV, HDTV) in größerem Umfang angeboten werden kann, nennt es die DBP Integriertes Breitbandfernmeldenetz (IBFN) [ScS1_84, Thom_87]. Pilotversuche zur Erprobung des IBFN sind unter der Abkürzung BIGFON (Breitbandiges integriertes Glasfaser-Fernmeldeortsnetz) bekannt [Brau_82, Brau_83, Brau_84, Brau_87].

Die geschilderte Entwicklung ist in Bild 1 etwas detaillierter dargestellt. Nicht dargestellt und im folgenden auch nicht explizit behandelt sind Funknetze, die als Anschlußnetz für ortsbewegte Teilnehmer, als Ersatznetz in Katastrophenfällen und als Garanten grenzüberschreitender Informationsfreiheit bleibende Bedeutung haben.

1.2 Welche Beobachtungsmöglichkeiten bieten diese Netze?

Bei diesen, wie bei allen Kommunikationsnetzen, hat man sich zu fragen, wie gut ihre Teilnehmer vor Schaden geschützt sind. Die resultierenden Probleme lassen sich grob in zwei Kategorien einteilen.

Das **Sicherheitsproblem**: Ein Teilnehmer kann geschädigt werden, indem eine Dienstebringung für ihn verhindert, verzögert oder verändert wird, oder indem eine Kommunikationsbeziehung unter seinem Namen oder auf seine Kosten, jedoch ohne sein Wissen oder seine Billigung aufgebaut wird.

Das **Datenschutzproblem**: Ein Teilnehmer kann auch durch Beobachten seiner Kommunikation Schaden erleiden.

Die zur Lösung des Sicherheitsproblems notwendigen Maßnahmen sind ausführlich in [VoKe_83] beschrieben und von denjenigen zur Lösung des Datenschutzproblems größtenteils unabhängig. Daher sollen sie im folgenden nur am Rande betrachtet werden.

Somit bleibt zunächst die Frage zu beantworten, welche Auswirkung die oben geschilderte Entwicklung der Netze der DBP auf den Datenschutz haben wird.

In Bild 2 ist die Endsituation dieser Entwicklung dargestellt: Alle Dienste, z. B. Fernsehen, Radio, Telefon, Bildschirmtext, werden über eine Glasfaser von der Vermittlungszentrale der DBP zum Netzabschluß eines Teilnehmers vermittelt.

Die Glasfaser ist diesem Teilnehmer (bzw. seiner Familie, seiner Firma o. ä.) eindeutig zugeordnet, und über sie wird, da es sich um ein Vermittlungsnetz handelt, nur übertragen, was von ihm oder speziell für ihn bestimmt ist. Folglich stellt die physikalische Netzadresse eine Art Personenkennzeichen (Kennzeichen für natürliche oder juristische Personen) dar, unter dem Daten über diesen Teilnehmer gesammelt werden können.

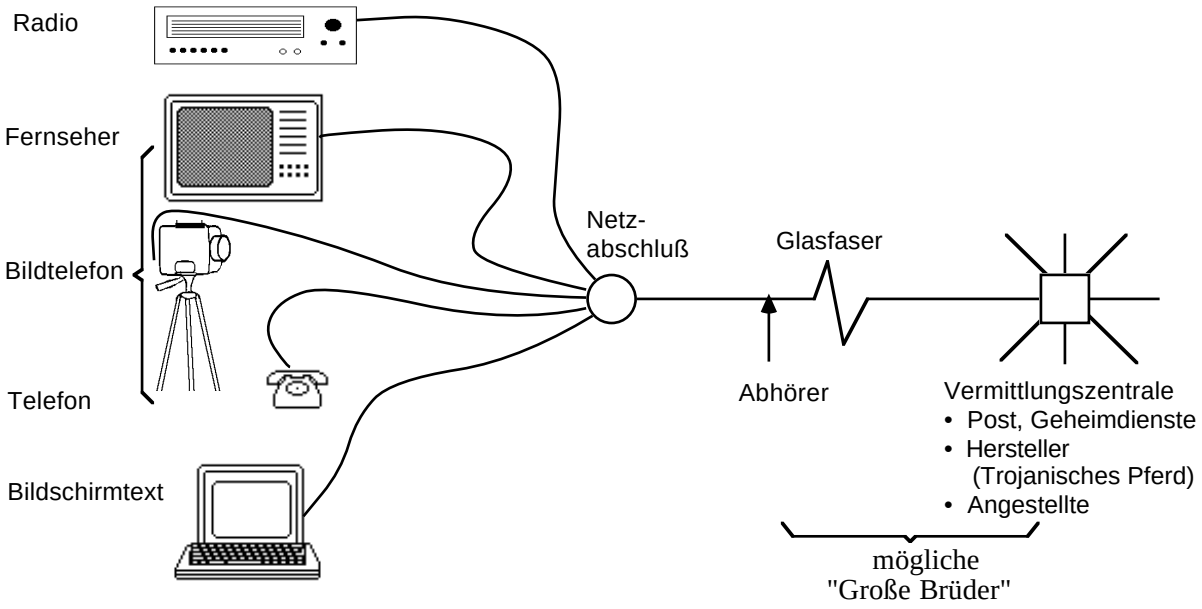


Bild 2: Beobachtbarkeit des Benutzers im sternförmigen, alle Dienste vermittelnden IBFN

Die dafür auf der Glasfaser und in den Vermittlungszentralen anfallenden Informationen bestehen *technisch* gesehen aus

- den transportierten **Nutzdaten** (Bild, Ton, Text) und
- den **Vermittlungsdaten** (Adressen und Absender der Kommunikationspartner, Datenumfang, Dienstart, Zeit).

Die daraus zu gewinnenden personenbezogenen Daten des Teilnehmers kann man *inhaltlich* gesehen einteilen in:

- **Inhaltsdaten**, d. h. Inhalte vertraulicher (persönlicher oder geschäftlicher) Nachrichten, z. B. von Telefongesprächen oder elektronischer Post.
- **Interessensdaten**, d. h. Informationen über das Interesse des Teilnehmers an Nachrichten, deren Inhalt nicht vertraulich ist. Hierzu zählen die Beobachtungen, welche Zeitungsartikel sich der Teilnehmer schicken läßt, welche Auskünfte er aus Datenbanken, z. B. Bildschirmtext, einholt und was genau er im Fernsehen sieht bzw. im Radio hört:

Will der Teilnehmer z. B. das Fernsehprogramm wechseln, teilt er dies über seinen Fernseher und seinen Netzabschluß der Vermittlungszentrale mit. Diese überträgt dann statt des bisher gesehenen das angeforderte Fernsehprogramm über die Glasfaser.

Interessensdaten charakterisieren nicht nur natürliche Personen, sondern auch Firmen:

Welche Fachaufsätze und Patente eine Firma anfordert, gibt z. B. ziemlich genaue Hinweise auf die von ihr gerade durchgeführten Forschungs- und Entwicklungsarbeiten.

Interessensdaten waren vor der Einrichtung öffentlich zugänglicher Datenbanken, insbesondere Bildschirmtext, in Kommunikationsnetzen überhaupt nicht zu gewinnen.

- **Verkehrsdaten**, also z. B. wann der Teilnehmer wie lange mit wem kommuniziert. Diese können aus den Vermittlungsdaten gewonnen werden.
  Bei natürlichen Personen ergeben die Verkehrsdaten bereits interessante Bausteine für ein Persönlichkeitsbild, z. B. Konsumgewohnheiten, Freundeskreis, Tagesablauf, Kontakte mit Polizei und Gesundheitsamt.
  Verkehrsdaten können aber nicht nur Persönlichkeitsrechte natürlicher Personen, sondern auch Geschäftsinteressen juristischer Personen verletzen. Steigt z. B. das Verkehrsaufkommen zwischen zwei konkurrierenden Firmen sprunghaft an, so legt dies Vermutungen über eine gemeinsame Produktentwicklung nahe.

Eine Möglichkeit, an diese Daten zu gelangen, ist das **Abhören** der den Teilnehmer mit seiner Ortsvermittlungsstelle verbindenden **Leitung** (bzw. bei mehreren dienstespezifischen Netzen: Leitungen) oder von Leitungen zwischen Vermittlungsstellen. Letzteres erlaubt zwar bei geeigneter Gestaltung des Protokolls zwischen Ortsvermittlungsstelle und „Lieferanten“ von Radio- und Fernsehprogrammen nicht die Überwachung der Massenkommunikation [Kais_82] (Radio- und Fernsehempfang) sowie keine Überwachung von Individualkommunikation zwischen an dieselbe Ortsvermittlungsstelle angeschlossenen Teilnehmern. Es erlaubt aber die gleichzeitige Überwachung vieler Teilnehmer bezüglich der über den Bereich ihrer Ortsvermittlungsstelle hinausgehenden Individualkommunikation.

Glasfasern, die für den Endausbau des Netzes vorgesehenen Leitungen, sind zwar etwas abhörsicherer als elektrische Leitungen, aber auch ihr Abhören stellt kein schwieriges technisches Problem dar und kann selbst im laufenden Übertragungsbetrieb begonnen werden [Horg_85 Seite 36, Hig1_87 Seite 108]: Ein Abhörer muß entweder passiv eine Stelle finden, an der Licht z. B. aufgrund von Verunreinigungen in genügender Stärke austritt, oder er muß aktiv (etwa durch Biegen der Glasfaser) Licht zum Austritt veranlassen, das dann in beiden Fällen wie üblich von einem Photodetektor ausgewertet werden kann. Gelingt es dem aktiven Abhörer, den Anteil des austretenden Lichtes bei Verwendung eines hinreichend empfindlichen Photodetektors klein genug zu halten, so stört dieser Lichtverlust den Empfänger nicht. Er arbeitet dann ohne nennenswerte Erhöhung der Übertragungsfehler weiter, so daß die von Herstellern und DBP für Verbindungen mit Zwischenregeneratoren vorgesehenen, lediglich die Bit-

fehlerrate kontrollierenden *digitalen* Überwachungssysteme [BrS1_86, BrS2_86] nicht ansprechen.

Entsprechende Überwachungssysteme sind zudem bei allen mir bekannten Beschreibungen der BIGFON-Pilotversuche nicht oder nur in einem Satz [Brau_82, Brau_83, Brau_84, Bra2_83, Bauc_83, BrMo_83, KlKl_83] erwähnt oder aber technisch nur sehr ungenau beschrieben [BaWe_83 Seite 144, 146, Scha_83 Seite 66 bis 68]. Deshalb ist zu befürchten, daß für die allermeisten, nämlich alle ohne Zwischenregenerator auskommenden, Teilnehmeranschlußleitungen keinerlei kontinuierlich arbeitendes Überwachungssystem vorgesehen ist. Gibt es in diesem Fall Zeiten, zu denen mit großer Wahrscheinlichkeit einige Sekunden nichts übertragen wird, so kann der Abhörer die Glasfaser durchtrennen und sie mit einer Abzweigung und ggf. einem Verstärker versehen, so daß danach kein Energieverlust des Signals mehr feststellbar ist. Entsprechendes läßt sich bei den in Deutschland [BrS1_86, BrS2_86] und England [CoBD_86 Seite 1401] (aus anderen Ländern liegen mir keine genügend genauen Systembeschreibungen vor) vorgesehenen Überwachungssystemen aktiv herbeiführen: In irgendeinem mittleren Übertragungsabschnitt, d. h. zwischen zwei Zwischenregeneratoren, werden die Glasfasern physisch unterbrochen, was der DBP sofort mit Angabe des Übertragungsabschnitts gemeldet und nach einiger Suchzeit in diesem Übertragungsabschnitt behoben wird. In der Zwischenzeit wird vom Abhörer auf beiden Seiten des unterbrochenen Übertragungsabschnitts jeweils die Glasfaser durchtrennt und mit einer Abzweigung und ggf. einem Verstärker versehen, die zum zuerst unterbrochenen Übertragungsabschnitt führen. Da Störungen nur in der Übertragungsrichtung der jeweiligen Glasfaser weitergemeldet und von den Zwischenregeneratoren nicht gespeichert, sondern dauernd vom aktuellen Status überschrieben werden, erfährt der Netzbetreiber diese zusätzlichen Unterbrechungen nicht und wird deshalb keinen Grund sehen, alle Übertragungsabschnitte physisch zu inspizieren, da dies sehr aufwendig ist. Bei Glasfasern mit Zwischenregeneratoren kann statt der abstrahlungsarmen optischen Glasfaser natürlich auch ein abstrahlungsreicher elektrischer Zwischenregenerator abgehört werden.

Natürlich könnte man in Glasfasernetzen auch *analoge* Überwachungssysteme vorsehen. Beispielsweise könnten die Empfänger die Lichtintensität zu Überwachungszwecken messen und Schwankungen oberhalb eines Schwellwertes melden. Diesen Schwellwert kann man aber aus Gründen der begrenzten Meßgenauigkeit sowie den bei den Sendern zu erwartenden Spannungsschwankungen sowie Alterungserscheinungen nicht beliebig niedrig ansetzen, ohne den Meldungen durch dauernde Fehlalarme jeden Sinn zu nehmen. Damit ist ein Wettlauf zwischen DBP und Abhörern vorauszusehen, bei dem die Abhörer den bedeutsamen Vorteil haben, daß sich ihre Technik laufend verbessert, während die Technik der Post durch die lange Nutzungsdauer der nachrichtentechnischen Infrastruktur viele Jahrzehnte unverändert bleibt. Außerdem hilft auch diese Technik natürlich nicht gegen passives Abhören der Streustrahlung der Glasfaser bzw. ggf. der abstrahlungsreicheren elektrischen Zwischenregeneratoren.

Betrachtet man nicht, wie in Bild 2 gezeigt, die von der DBP geplante Endsituation, sondern die des heutigen Fernsprechnetzes, des heutigen integrierten Text- und Datennetzes oder die des ab 1988 errichteten schmalbandigen ISDN, so fallen auf den Teilnehmeranschlußleitungen, da Radio, Fernsehen und Bildtelefon noch nicht vermittelt werden können, zwar weit weniger, aber nicht weit weniger sensitive Daten an. Sie werden über eine normale „Telefon“leitung übertragen, die bekanntermaßen mit einfachen technischen Mittel und so gut wie unentdeckbar abgehört werden kann [Horg_85].

Weit einfacher und zudem vollständig für viele Teilnehmer auf einmal erhalten diejenigen die Daten, die sie sich direkt **aus der Vermittlungszentrale beschaffen** können.

Zunächst einmal *kann* die DBP (und damit der Staat, genauer seine Geheimdienste) als **Betreiber** die Vermittlungsanlagen beliebig Daten speichern und auswerten lassen. Innerhalb weiter Grenzen *darf* sie dies auch, wie dem 4. Tätigkeitsbericht der Landesbeauftragten für den Datenschutz in Baden-Württemberg, Dr. Ruth Leuze, sowie der im Neunten Tätigkeitsbericht des Bundesbeauftragten für den Datenschutz wiedergegebenen „Entschließung der Konferenz der Datenschutzbeauftragten des Bundes und der Länder vom 18. April 1986 zum Entwurf einer Telekommunikationsordnung“ (TKO) zu entnehmen ist:

> *„... der Betreiber von Bildschirmtext darf über jeden Teilnehmer speichern, wann, wie lange und wie oft er auf welche Weise den Bildschirmtext in Anspruch nahm.“*
>
> [Leuz_83 Seite 114, vgl. auch Leuz_84 Seite 24, 25]

> *„Die Allgemeinen Vorschriften zum Datenschutz in Teil VII Abschnitt 2 TKO bedürfen der Ergänzung und Präzisierung:*
> - *Wann Bestandsdaten, Verbindungsdaten und Gebührendaten zu löschen sind, ist teils überhaupt nicht, teils nur unbestimmt geregelt. So dürfen Verbindungsdaten aus „betriebsbedingten Gründen“ auf unbestimmte Zeit gespeichert werden. Gleiches gilt, wenn der Teilnehmer „eine andere Art der Verarbeitung“ beantragt hat; die Voraussetzungen hierfür werden nämlich nicht festgelegt.*
> - *Auch die Bestimmungen über die Verarbeitung der Gebührendaten müssen so formuliert werden, daß keine unzulässigen Schlüsse auf ein Teilnehmerverhalten gezogen werden können.*
> - *Die Regelungen umfassen nicht alle bei der Post anfallenden Daten. Beispielsweise fehlen Regelungen zu den Inhalten der Informationen und den beim Betrieb der Dienste anfallenden Daten.*
> - *Die Vorschriften erlauben der Post, alle Daten zu beliebigen „Telekommunikationszwecken“ zu verwenden. Unerläßlich ist eine Nutzungsbeschränkung auf die Zwecke der jeweils in Anspruch genommenen Dienste.*
> - *Die Regelung der Befugnis, Daten weiterzugeben, ist zu umfassend und unklar. Insbesondere sollten Verbindungsdaten von jeder Übermittlung ausgeschlossen bleiben.*
> - *Trotz der Fülle und der besonderen Sensitivität der bei der Post vorhandenen personenbezogenen Daten fehlen spezielle Vorschriften über die Datensicherung.“*
>
> [BfD_87 Seite 92, 93]

Weiter können Personen (z. B. **Postbedienstete**, **Wartungstechniker**) oder Organisationen (z. B. **Hersteller**), die Zugang zur Vermittlungszentrale haben oder hatten, alle dort erfaßbaren Informationen erhalten: Vermittlungszentralen sind heute komplexe frei speicherprogrammierbare Rechensysteme mit vielfältigen Möglichkeiten zum Installieren **„Trojanischer Pferde“** [Home_??], d. h. von Systemteilen, die Information auf verborgenen Kanälen einem nicht empfangsberechtigten Empfänger zukommen lassen [Lamp_73, Lipn_75, PoKl_78, Denn_82 Seite 281, Loep_85]. Das Finden Trojanischer Pferde ist äußerst schwierig

[Thom_84] und, da eine diesbezügliche Systemüberprüfung auch nach jeder Wartungsmaßnahme nötig ist, sehr aufwendig. (Es sei angemerkt, daß bei Fernwartung der Betreiber der Anlage das Stattfinden einer Wartungsmaßnahme möglicherweise gar nicht erfährt. Diese erstmals in [PfPW_87 Seite 286] geäußerte Befürchtung wurde leider durch [Hig1_87 Seite 108] erhärtet.)

Werden die bereits in [Denn_82 Seite 317f] skizzierten und in [PoGr_86, PoGr_87, Hoff_87] ausführlicher beschriebenen Maßnahmen zur Verhinderung der Selbstausbreitung Trojanischer Pferde in andere Systemteile (Programme mit dieser Fähigkeit werden „Computer-Viren“ genannt [Cohe_84, Cohe_87]) nicht während des gesamten Entwurfs- und Produktionsprozesses von Hard- und Software der Vermittlungszentralen und auch bezüglich der dazu verwendeten Werkzeuge, der zur Herstellung der Werkzeuge benutzten Werkzeuge usw. angewandt, ist der Kreis derjenigen, die Trojanische Pferde in den Vermittlungszentralen plazieren können, noch erheblich größer. Alle, die direkt oder indirekt Einfluß auf die verwendeten Werkzeuge, die zu Herstellung der verwendeten Werkzeuge verwendeten Werkzeuge usw. hatten, können auch Trojanische Pferde plazieren.

Unterstellt, diese heute noch nicht angewandten Maßnahmen würden vollständig und permanent angewandt, so bliebe von der Selbstausbreitungs-Problematik Trojanischer Pferde „lediglich“ übrig, daß zumindest manche Werkzeuge beim Entwurfs- und Produktionsprozeß manche anderen Werkzeuge oder Systemteile generieren oder verändern *müssen*, beispielsweise Übersetzer das übersetzte Programm [Thom_84]. Ist die Entwurfs- oder Produktionskomplexität dieser Werkzeuge bzw. Systemteile groß genug, so können sich Trojanische Pferde durch bzw. in sie weitgehend unbemerkbar transitiv ausbreiten: Von der Bedrohung durch Computer-Viren bliebe also nur die durch **transitive Trojanische Pferde** übrig. Aber auch dies ist im Zuge immer größerer, insbesondere auch internationaler Verflechtungen und immer komplexerer Werkzeuge und Systeme eine völlig unakzeptable Situation, sofern die bisherigen Systemstrukturen einfach beibehalten werden.

Alle diese Bedrohungen werden noch dadurch verstärkt, daß die DBP anscheinend nicht einmal den Versuch unternehmen will, nach Trojanischen Pferden zu suchen, wie das folgende Beispiel der Bildschirmtextzentrale in Ulm zeigt:

Der Hersteller der Bildschirmtextzentrale muß den Systemaufbau der DBP nicht offenlegen, was das Aufspüren Trojanischer Pferde vollends unmöglich macht. Der Bundesbeauftragte für den Datenschutz, Dr. Reinhold Baumann, schreibt darüber [BfD_85 Seite 25]:

> *„Wer Daten verarbeitet, muß die Wirkung der dafür eingesetzten Programme genau kennen. Deshalb hat es überrascht, daß der Deutschen Bundespost als Betreiber des Bildschirmtext-Systems keine umfassende Dokumentation aller eingesetzten Programme vorliegt. Zur Begründung dafür hat sie auf ihre vertraglichen Regelungen mit der Lieferfirma IBM hingewiesen. Dadurch ist es der Deutschen Bundespost verwehrt, sich genaue Kenntnis der Programme in allen Details ohne Hilfe Dritter zu verschaffen. Vor diesem Hintergrund erscheint schwer vorstellbar, wie die Deutsche Bundespost ihrer Verantwortung für die ordnungsgemäße Anwendung der Datenverarbeitungsprogramme (Par. 15 Nr. 2 BDSG) gerecht werden kann.“*

Zwar schreibt Dr. Reinhold Baumann ein Jahr später in [BfD_86 Seite 20]:

> *„... die fehlende EDV-Programmdokumentation liegt inzwischen der DBP vor.“*

Diese Aussage bezieht sich aber nach Auskünften eines ehemaligen und eines derzeitigen Mitarbeiters des Bundesbeauftragten für den Datenschutz (wie die Datenschutzkontrolle im öffentlichen Bereich allgemein) nur auf die Anwendungsprogramme und nicht etwa, was ein mit gesundem Menschenverstand ausgerüsteter Informatiker darunter verstehen würde, auf alle Programme, also insbesondere auch Betriebssystem, Datenbanksystem, Übersetzer etc. Von der auch notwendigen Kontrolle der Firm- und Hardware redet sowieso niemand.

Da der Personenkreis, der an die im Netz, insbesondere in den Vermittlungszentralen, anfallenden Daten gelangen kann, so groß ist, wird es auch ausländischen Geheimdiensten möglich sein, die Daten zu erhalten.

Interessensdaten können außer durch Abhören von Leitungen oder über die Vermittlungszentralen auch noch von großen **Kommunikationspartnern**, etwa Datenbanken oder Zeitungsverlagen, gesammelt werden, sofern diese die Identitäten der Dienstnutzer erfahren.

1.3 Notwendigkeit vorbeugenden Datenschutzes als Gegenmaßnahme

Außer durch vorbeugende technische Datenschutzmaßnahmen kann man auf den geschilderten Sachverhalt auf folgende Weisen reagieren:

- Man verdrängt, bagatellisiert oder bestreitet ihn.

- Man verbietet per Gesetz das Erfassen dieser Daten oder das Erstellen von Persönlichkeitsprofilen aus ihnen (was für einen Teil der oben genannten Daten durch das Fernmeldegeheimnis bereits der Fall ist). Aber ein Verbot ist nur dann wirkungsvoll, wenn seine Einhaltung mit angemessenem Aufwand **überprüft** und durch Strafverfolgung gesichert und der ursprüngliche Zustand durch Schadensersatz **wiederhergestellt** werden kann. Beides ist in der geschilderten Situation leider nicht gegeben:

  „Datendiebstahl“ allgemein, speziell das direkte Abhören von Leitungen oder Kopieren von Daten aus Vermittlungsrechnern, ist kaum feststellbar, da sich an den Originaldaten nichts ändert. Ebenso ist, wie oben erwähnt, das Installieren Trojanischer Pferde kaum festzustellen und erst recht nicht die unerlaubte Weiterverarbeitung von Daten, die man legal (oder auch illegal) erhalten hat. Dies bedeutet, daß auch dann, wenn die DBP sich an der Durchsetzung des Gesetzes zu beteiligen versucht, nicht einmal entdeckt werden kann, wenn Mitarbeiter, Hersteller von Vermittlungszentralen oder Datenbanken, die die Identitäten der Dienstnutzer erfahren, es übertreten.

  Die Wiederherstellung des ursprünglichen Zustands müßte vor allem darin bestehen, alle entstandenen Daten zu löschen. Man ist aber nie sicher, ob nicht noch weitere Kopien existieren. Außerdem können sich Daten im Gedächtnis von Menschen festsetzen, wo das Löschen besser nicht angestrebt werden sollte.

- Man versucht, die Weiterentwicklung der Kommunikationsnetze zu verhindern, insbesondere die Errichtung diensteintegrierender Digitalnetze mit ihren gegen Angriffe anfälligen komplizierten Vermittlungszentralen [KuRo_86, Kub1_87]. Sofern aber an den durch solche Netze preiswert ermöglichten neuen Diensten und Qualitätsverbesserungen für schon existierende Dienste Interesse besteht, wird man ihre Einführung nicht verhindern können und wollen, zumal ein Beibehalten des heutigen im Teilnehmeranschlußbereich analogen Fernsprechnetzes bei der zu erwartenden Weiterentwicklung der Technik (automatische Sprecher- [DaPr_84 Seite 213, 214, Bake_85 Seite 210, 211] und später auch Spracherkennung [Wall_87]) innerhalb weniger Jahre auch zu unlösbaren Datenschutzproblemen führt [Pfit_86, PPfW_88].

So bleibt nur die Möglichkeit, zu untersuchen, ob durch **vorbeugende (größtenteils technische) Datenschutzmaßnahmen** in solchen Netzen das Erstellen von Persönlichkeitsprofilen verhindert werden kann. Dies bedeutet, daß man die Benutzung der Netze, die auf den Netzen angebotenen Dienste oder gar die Netze selbst soweit anders gestaltet, daß von vornherein im Netz keine Möglichkeit besteht, ohne explizite Einwilligung des Teilnehmers über ihn Daten zu erfassen [Pfit_83, Riha_85].

1.4 Diskussion möglicher Einwände

Ein oft gehörter Einwand gegen vorbeugende (und deshalb nicht unterlaufbare) technische Datenschutzmaßnahmen ist, daß sie gesellschaftlich nicht wünschenswert seien, da ein Interesse bestehe, bei begründetem Verdacht das Verhalten Einzelner beobachten zu können (G 10-Gesetz).

Hier kann man entgegnen, daß der technische Fortschritt auch auf anderen Gebieten als Kommunikationsnetzen eine Fülle neuer Überwachungstechniken hervorbringt:

- Immer kleinere und perfektere Abhörmikrofone [Horg_85 Seite 31] oder die Abtastung von Fensterscheiben mittels Laserstrahl durch mehrere hundert Meter entfernte Geräte [Hor2_85] erlauben die Wiedergabe aller Geräusche, z. B. aller Gespräche im Zimmer und am Telefon sowie des gewählten Fernseh- oder Radioprogramms.

- Alles auf Bildschirmen Angezeigte kann mit einem handelsüblichen Gerät im Wert von rund fünfzig US-Dollar über Entfernungen von 1000 m aufgefangen und auf einem Fernsehgerät dargestellt werden [Eck_85]. Ebenso ist es aus größerer Nähe möglich, die von Rechnern abgestrahlten elektromagnetischen Wellen zu empfangen, zu analysieren und auf die in ihnen verarbeiteten Programme und Daten zu schließen [Schl_87 Seite 9].

- Optische Überwachung ist durch immer bessere und billigere Kameras möglich (neben der klassischen Methode der persönlichen Verfolgung). In Zukunft erlauben vielleicht auch Aufklärungssatelliten [Adam_86, Adam_87] nicht nur die Beobachtung militärischer Operationen, sondern auch die Erstellung von Bewegungsprofilen privater Bürger.

Dadurch ist eine umfassendere Überwachung **weniger Einzelner** möglich als durch die bisherige Telefonüberwachung. Gegen manche dieser Techniken gibt es Gegenmaßnahmen, z. B. sehr schmutzige Fensterscheiben gegen die Laserüberwachung oder Abschirmung von Terminals und Rechnern (Verfahren und Normen für letzteres sind unter den Kürzeln COMSEC und TEMPEST bekannt [Horg_85]). Da aber auch bisher niemand gezwungen war, am Telefon Belastendes von sich zu geben, ergibt sich nur dasselbe Problem wie mit der bisherigen Telefonüberwachung: gerade diejenigen, die wirklich Gesetze übertreten, greifen vermutlich zu Schutzmaßnahmen (z. B. Verschlüsselung; Steganographie; Kuriere; spezielle, die Ortung des Senders weitgehend verhindernde Funktechniken [Harb_86]), während jemand, der sich keines Unrechts bewußt ist, beobachtbar ist.

Die geplanten Kommunikationsnetze hingegen würden nicht nur eine viel umfassendere Beobachtung Einzelner, sondern auch ohne großen Aufwand das Beobachten der **gesamten Bevölkerung** ermöglichen, und zwar nicht nur durch den eigenen Staat (Geheimdienst), sondern auch durch Fernmeldefirmen, Systemprogrammierer, fremde Staaten (und deren Geheimdienste) usw.

Dieser Aufwandsunterschied zwischen Beobachtung über diensteintegrierende Digitalnetze und anderen Überwachungstechniken beantwortet auch den umgekehrten Einwand, ob es sich überhaupt lohne, Daten in Netzen zu schützen, ohne gleichzeitig Gegenmaßnahmen gegen alle anderen Überwachungsmöglichkeiten anzugeben und zu ergreifen.

In entfernterer Zukunft könnten aber einige der anderen Überwachungsmöglichkeiten zu ebenso großen Datenschutzproblemen führen. Außerdem werden einige der Daten, die in Kommunikationsnetzen geschützt werden sollen, über Personalinformationssysteme, maschinenlesbare Personalausweise, Zahlungssysteme (vgl. Kapitel 8) u. ä. ebenfalls in Rechenanlagen gelangen. Hier ergibt sich (wie bei den Vermittlungszentralen) das Problem, daß die Einhaltung von Datenschutzgesetzen und -vereinbarungen nicht mit vernünftigem Aufwand überprüfbar ist.

Bisher habe ich zwar gefordert, daß Datenschutz überprüfbar sein soll, jedoch nicht spezifiziert, durch wen. Hierzu, sowie zu dem eben diskutierten Argument, nicht unterlaufbarer Datenschutz in Kommunikationsnetzen gefährde unsere Demokratie, ist der folgende Teil der Urteilsbegründung des Volkszählungsurteils des Bundesverfassungsgerichts vom Dezember 1983 [Bund_83 Seite 272] sehr aufschlußreich:

> *„Wer nicht mit hinreichender Sicherheit überschauen kann, welche ihn betreffende Informationen in bestimmten Bereichen seiner sozialen Umwelt bekannt sind, und wer das Wissen möglicher Kommunikationspartner nicht einigermaßen abzuschätzen vermag, kann in seiner Freiheit wesentlich gehemmt werden, aus eigener Selbstbestimmung zu planen oder zu entscheiden. Mit dem Recht auf informationelle Selbstbestimmung wären eine Gesellschaftsordnung und eine diese ermöglichende Rechtsordnung nicht vereinbar, in der Bürger nicht mehr wissen können, wer was wann und bei welcher Gelegenheit über sie weiß. Wer unsicher ist, ob abweichende Verhaltensweisen jederzeit notiert und als Information dauerhaft gespeichert, verwendet oder weitergegeben werden, wird versuchen, nicht durch solche Verhaltensweisen aufzufallen. Wer damit rechnet, daß etwa die*

Teilnahme an einer Versammlung oder einer Bürgerinitiative behördlich registriert wird und daß ihm dadurch Risiken entstehen können, wird möglicherweise auf eine Ausübung seiner entsprechenden Grundrechte (Art. 8, 9 GG) verzichten. Dies würde nicht nur die individuellen Entfaltungschancen des Einzelnen beeinträchtigen, sondern auch das Gemeinwohl, weil Selbstbestimmung eine elementare Funktionsbedingung eines auf Handlungs- und Mitwirkungsfähigkeit seiner Bürger begründeten freiheitlichen demokratischen Gemeinwesens ist.“

Nun sind Urteilsbegründungen, auch wenn von höchsten Gerichten abgegeben, keine Wahrheiten per se. Deshalb habe ich seit 1983 alle mir begegnenden demoskopischen Untersuchungen über Hoffnungen und Befürchtungen bezüglich des Rechner-Einsatzes gesammelt. Sie bestätigen alle den in der obigen Urteilsbegründung angeführten Sachverhalt. Die wichtigsten Ergebnisse aller sechs mir bekannten Untersuchungen sind im folgenden kurz wiedergegeben:

1. 81% der bundesrepublikanischen Bevölkerung über 14 Jahren halten im Mai 1983 den Satz *„Der Computer wird dazu beitragen, daß man uns besser überwachen kann“* für eher zutreffend, 8% für eher nicht zutreffend [Lang_84]. Damit ist Mangel an Datenschutz die ausgeprägteste Meinung über den Rechner-Einsatz in dieser Untersuchung.

2. Sowohl 87% der erwachsenen US-Amerikaner als auch 87% der Leser von IEEE The Institute beantworteten die Frage *„Do you think if someone wanted to put together a master file on you that it could be done fairly easily?“* mit ja [Perr_84]. 78% der erwachsenen US-Amerikaner und 76% der Leser von IEEE The Institute beantworteten die Frage *„If such a file were put together, would you feel that your privacy had been violated?“* mit ja [Perr_84].

3. 51% aller Amerikaner (und 40% aller amerikanischen Ingenieure) äußerten, daß Rechner eine tatsächliche Bedrohung der Vertraulichkeit darstellen und 77% (70%) äußerten, daß sie sehr oder etwas über Bedrohungen der Vertraulichkeit in den Vereinigten Staaten besorgt seien (*„51 (40) percent of the respondents indicated that computers are an actual threat to privacy and 77 (70) percent indicated that they are very or somewhat concerned about threats to privacy in the United States.“*) [Surv_84].

4. Nach einer vom „Daily Telegraph“ gedruckten Meinungsumfrage, in welchem Maße der Orwellsche Alptraum vom totalen Überwachungsstaat in den Augen der Bürger bereits Realität geworden ist, glauben 38% der Deutschen, 37% der Schweizer und sogar 72% der Engländer, daß *„es keine Privatsphäre gibt, weil die Regierung über einen alles erfahren kann, was sie wissen will“* [Dail_84].

5. Nach einer in [Gins_85] abgedruckten Umfrage des Atlantic Institute äußerten auf die Frage: *„Sagen Sie, ob Sie dieser Aussage eher zustimmen oder sie eher ablehnen: Die Wahrscheinlichkeit wird immer größer, daß Computer-Datenbanken dazu benutzt werden, persönliche Rechte zu verletzen.“* in der Bundesrepublik Deutschland 51/20/29%, in Frankreich 71/19/10%, in Großbritannien 75/13/12%, in den USA 68/28/4% und in Japan 50/18/32% Zustimmung/Ablehnung/Keine Meinung.

6. Nach einer von IBM beim Sample Institut, Mölln, in Auftrag gegebenen Repräsentativbefragung von 2000 Bundesbürgern, die von Dr. Susanne Schröder in [IBM_87] zusammengefaßt und kommentiert wird, wird zwar die generelle Einstellung zum Rechner immer positiver. Aber: *„Weitgehend unberührt von der Trendwende zum Positiven bleiben Ängste und Befürchtungen über negative Einflüsse von Computern in Staat und Gesellschaft auf hohem Niveau bestehen. So ist auch 1986 die übergroße Mehrheit von 70 Prozent aller Befragten der Meinung, daß Computer in den nächsten Jahren viele Arbeitsplätze ersetzen werden, und die Angst vor dem ‚gläsernen Menschen' bleibt ebenfalls unverändert stark: Der Aussage ‚Computer geben dem Staat zuviel Macht und zu viele Möglichkeiten, Kontrolle auszuüben', stimmt eine Mehrheit von 58 Prozent zu."*

Wie in einer vergleichenden Analyse verschiedener Befragungen [SymG_84 Seite 410] hervorgehoben ist, ist bei der Bewertung dieser Zahlen zu berücksichtigen, daß Personen, denen besonders viel an Datenschutz liegt bzw. deren Ängste besonders groß sind, sich nicht in Befragungen äußern. Dies verschiebt die Zahlen etwas in Richtung größerer Zufriedenheit mit dem bestehenden Zustand.

Informationssysteme im weitesten Sinne sind also soweit als möglich so zu gestalten, daß der Betroffene, in unserem Fall also der **Teilnehmer** eines Kommunikationsnetzes, den **Datenschutz selbst überprüfen kann**.

Im folgenden wird gezeigt, daß es effizient und zuverlässig realisierbare Kommunikationsnetze gibt, die dem Teilnehmer das „Recht auf informationelle Selbstbestimmung ... grundsätzlich selbst über die Preisgabe und Verwendung seiner persönlichen Daten zu bestimmen" gewähren. Diese Kommunikationsnetze erschweren die Lösung des Sicherheitsproblems nicht und ermöglichen auch die Kommunikation mit Partnern an anderen, etwa ausländischen, Kommunikationsnetzen.

2 Grundverfahren für teilnehmerüberprüfbaren Datenschutz

In diesem Kapitel wird zunächst aus den in der Motivation beschriebenen Anforderungen einer demokratischen Gesellschaft an Kommunikationsnetze eine informatische Problemstellung abgeleitet. An ihr werden zwei komplementäre Lösungsansätze gemessen.

Danach werden Hilfsmittel aus der Kryptographie eingeführt und ihr Einsatz bei der Verschlüsselung, aber auch deren Grenzen aufgezeigt.

Um diese Grenzen zu überwinden, werden Grundverfahren zum Schutz der Verkehrs- und Interessensdaten der Teilnehmer entwickelt. Diese Grundverfahren sind teilweise außerhalb des Kommunikationsnetzes angesiedelt, d. h. jeder Benutzer muß sie für sich selbst anwenden. Da diese Grundverfahren allein nicht ausreichen, müssen sie durch solche innerhalb des Kommunikationsnetzes ergänzt werden, was, wie wir sehen werden, erheblichen Einfluß auf die Gestaltung des Kommunikationsnetzes hat.

2.1 Informatische Problemstellung und Lösungsansätze

2.1.1 Informatische Problemstellung

Das **Ziel** der zur Lösung des Datenschutzproblems zu entwickelnden Verfahren ist es, einem *Angreifer* das Erfassen sensitiver Daten unmöglich zu machen: Die Kommunikation sollte gegenüber Unbeteiligten weitgehend *unbeobachtbar* und gegenüber Beteiligten (z. B. Kommunikationspartnern) üblicherweise *anonym* erfolgen. Damit sich nicht doch nach und nach unspezifiziertes Wissen über Personen ansammeln kann, sollten einzelne Verkehrsereignisse auch durch Beteiligte üblicherweise *unverkettbar* sein.

Sofern durch die Verfahren zur Lösung des Datenschutzproblems die Lösung des Sicherheitsproblems (vgl. Abschnitt 1.2) erschwert wird, ist zu zeigen, wie auch dies weiterhin gelöst werden kann.

Wegen ihrer großen Bedeutung für das Verständnis möchte ich die gerade nebenbei eingeführten Begriffe etwas genauer erläutern. Die dabei verwendeten, hier teilweise noch undefinierten Sachverhalte werden bei der Anwendung der Begriffe jeweils konkretisiert.

Ohne moralisch/rechtliche Bedeutungsassoziation wird **Angreifer** im folgenden als kurze Bezeichnung für jemanden verwendet, der versucht, sensitive Daten ohne Einwilligung der Betroffenen zu sammeln.

Da Schutz vor einem allmächtigen Angreifer, der alle Leitungen, alle Vermittlungszentralen, alle Teilnehmerstationen außer der des Angegriffenen und den Kommunikationspartner kontrolliert, durch Maßnahmen innerhalb des Kommunikationsnetzes nicht möglich ist, sind alle

folgenden Maßnahmen nur Annäherungen an den perfekten Schutz der Teilnehmer vor jedem möglichen Angreifer. Die Annäherung wird im allgemeinen durch Angabe der **Stärke eines Angreifers** in der Form eines **Angreifermodells** bestimmt: *wie weit verbreitet* ist der Angreifer (wie viele und welche Leitungen und/oder Stationen kann er maximal beobachten und/oder gar aktiv kontrollieren) und mit *wieviel Rechenkapazität ausgestattet* ist er (vgl. Abschnitt 2.2.2.2)?

(Um Mißverständnissen bei den folgenden Diskussionen der Stärke von Datenschutzmaßnahmen vorzubeugen sei hier ein für alle mal darauf hingewiesen, daß das In-Betracht-Ziehen einer Organisation oder Personengruppe als Angreifer in den Kapiteln 2 bis 10 ausschließlich der technisch-naturwissenschaftlichen Klärung der Stärke der Datenschutzmaßnahme dient und alle Organisationen lediglich als Rollenträger bezüglich des Kommunikationsnetzes betrachtet werden. Es wird im folgenden also lediglich diskutiert, ob eine Organisation oder Personengruppe mit Erfolg angreifen *könnte*, nicht ob sie dies tat, tut oder tun wird!)

Ein **Ereignis** *E* (z. B. Senden einer Nachricht, Abwickeln eines Geschäftes) heißt **unbeobachtbar** bezüglich eines Angreifers *A*, wenn die Wahrscheinlichkeit des Auftretens von *E* nach jeder für *A* möglichen Beobachtung *B* sowohl echt größer 0 als auch echt kleiner 1 ist. Dies kann natürlich nur für an *E unbeteiligte* Angreifer der Fall sein und lautet in der üblichen wahrscheinlichkeitstheoretischen Schreibweise: Für *A* gilt für alle *B*: $0 < P(E|B) < 1$.

Das Ereignis *E* heißt *perfekt* unbeobachtbar bezüglich eines Angreifers *A*, wenn die Wahrscheinlichkeit des Auftretens von *E* vor und nach jeder für *A* möglichen Beobachtung *B* gleich ist, d. h. für *A* gilt für alle *B*: $P(E) = P(E|B)$. Dies bedeutet nach [Sha1_49], daß die Beobachtung *A* keinerlei zusätzliche Information über *E* liefern kann.

Eine **Instanz** (z. B. Person) heißt in einer Rolle *R* **anonym** bezüglich eines Ereignisses *E* (z. B. als Sender einer Nachricht, Käufer in einem Geschäft) und eines Angreifers *A*, wenn für jede mit *A* nicht kooperierende Instanz die Wahrscheinlichkeit, daß sie bei *E* die Rolle *R* wahrnimmt, nach jeder für *A* möglichen Beobachtung sowohl echt größer 0 als auch echt kleiner 1 ist. Dies kann auch für an *E beteiligte* Angreifer (z. B. den Empfänger der Nachricht bzw. den Verkäufer) der Fall sein.

Eine Instanz heißt in einer Rolle *R* bezüglich eines Ereignisses *E* und eines Angreifers *A* *perfekt* anonym, wenn für jede mit *A* nicht kooperierende Instanz die Wahrscheinlichkeit, daß sie bei *E* die Rolle *R* wahrnimmt, vor und nach jeder für *A* möglichen Beobachtung gleich ist. Letzteres bedeutet nach [Sha1_49], daß die Beobachtung dem Angreifer keinerlei zusätzliche Information darüber liefert, wer bei *E* die Rolle *R* wahrnimmt.

Zwei **Ereignisse** *E* und *F* heißen bezüglich eines Merkmals *M* (z. B. zwei Nachrichten bezüglich ihres Senders oder bezüglich der Transaktion, zu der sie gehören) und bezüglich eines Angreifers *A* **unverkettbar**, wenn die Wahrscheinlichkeit, daß sie in *M* übereinstimmen, nach jeder für *A* möglichen Beobachtung sowohl echt größer 0 als auch echt kleiner 1 ist. Dies kann auch für an *E beteiligte* Angreifer der Fall sein.

Zwei Ereignisse *E* und *F* heißen bezüglich eines Merkmals *M* und bezüglich eines Angreifers *A* *perfekt* unverkettbar, wenn die Wahrscheinlichkeit, daß sie in *M* übereinstimmen, vor und nach jeder für *A* möglichen Beobachtung gleich ist. Letzteres bedeutet nach [Sha1_49], daß

die Beobachtung dem Angreifer keinerlei zusätzliche Information darüber liefert, ob diese Ereignisse in *M* übereinstimmen.

Die **Vertrauenswürdigkeit der Unverkettbarkeit, Unbeobachtbarkeit bzw. Anonymität** wird dadurch bestimmt, ein wie *starker* Angreifer den vorherigen Definitionen unterlegt wird.

Unbeobachtbarkeit und Anonymität kann man zusätzlich noch dadurch parametrisieren, daß man sie nur innerhalb gewisser Klassen von Ereignissen (z. B. Nachrichtentypen) bzw. Instanzen (z. B. denen eines bestimmten Teilnetzes eines größeren Kommunikationsnetzes, vgl. Abschnitt 4.3) verlangt. In den obigen Definitionen wurden keinerlei Klassen definiert, sie definieren daher den maximal erreichbaren **Grad an Unbeobachtbarkeit bzw. Anonymität**. Mit einer **Klasseneinteilung** parametrisierte Definitionen ergeben sich kanonischerweise. Als Beispiel wird die für unbeobachtbar angegeben:

Ein **Ereignis** *E* heißt **unbeobachtbar** (bzw. *perfekt* unbeobachtbar) bezüglich eines Angreifers *A* und einer gegebenen Klasseneinteilung von Ereignissen, wenn die bedingte Wahrscheinlichkeit des Auftretens von *E*, gegeben daß ein Ereignis seiner Klasse auftritt, nach jeder für *A* möglichen Beobachtung *B* sowohl echt größer 0 als auch echt kleiner 1 (bzw. vor und nach den Beobachtungen gleich) ist.

Beides kann natürlich nur für an *E unbeteiligte* Angreifer der Fall sein. Es ist sowohl im Falle der Unbeobachtbarkeit als auch im Falle der perfekten Unbeobachtbarkeit möglich, daß sich für den Angreifer durch die Beobachtung die Wahrscheinlichkeiten des Auftretens von Ereignisse aus bestimmten Klassen ändern. Der Angreifer gewinnt also möglicherweise Information über diese Klassen, im zweiten Fall aber nicht über einzelne Ereignisse innerhalb einzelner Klassen.

2.1.2 Diskussion von Lösungsansätzen

Bezüglich der Gestaltung von Kommunikationsnetzen gibt es zwei komplementäre Ansätze, deren Vor- und Nachteile im Lichte obiger informatischer Problemstellung diskutiert werden sollen.

Der von den Netzbetreibern verfolgte, **zentralistische Ansatz** sieht möglichst viele der benötigten Funktionen für die zentral betriebenen Netzkomponenten, insbesondere Vermittlungszentralen, vor. Aus Sicht des Netzbenutzers sollen **vertrauenswürdige fremde Instanzen** das Sicherheitsproblem lösen, indem alle Vorgänge im Netz von ihnen beobachtet und, um die Funktion eines Zeugen, nicht etwa die eines „Großen Bruders“ wahrzunehmen, protokolliert werden. Das Datenschutzproblem soll gelöst werden, indem rechtliche Regelungen (Fernmeldegeheimnis, TKO) bestehen und die Mitarbeiter des Netzbetreibers sowie die Lieferanten der Fernmeldeanlagen zu deren Einhaltung angehalten werden. Begründet wird dieses Vorgehen vor allem mit seiner ökonomischen Sparsamkeit.

In Kapitel 1 wurde jedoch klar, daß dieses Vorgehen verfassungsrechtlich zumindest bedenklich ist, da die Einhaltung der Datenschutz-Regelungen nicht überprüft werden kann.

Von Herbert Kubicek und Arno Rolf wurde zur Abschwächung des Datenschutzproblems vorgeschlagen, die Vermittlungszentralen nicht frei speicherprogrammierbar, sondern mittels ROMs fest speicherprogrammiert oder in der Form überhaupt nicht programmierbarer Spezialschaltungen auszulegen:

> *„Grundsätzlich spricht auch nichts gegen eine Digitalisierung der Vermittlungstechnik im Sinne des Ersatzes von Elektromechanik durch Mikroelektronik, wenn dadurch Kosten und Gebühren gesenkt werden können. Nur sollte die Digitalisierung unter Risikogesichtspunkten nicht in der vorgesehenen Form softwaregesteuerter speicherprogrammierter Systeme durchgeführt werden, weil ... Verdatungsrisiken erst mit der Speicherprogrammierung und der Protokollation von Verbindungsdaten entstehen. Dem Grundbedarf angemessener und unter Datenschutzgesichtspunkten vorsichtiger sind festgeschaltete Vermittlungssysteme."* [KuRo_86 Seite 325]
> *„... mit festgeschalteter Mikroelektronik anonym vermittelt ..."* [Kub2_86 Seite 807]

Dies erschwert zwar ein von den Bediensteten des Betreibers unentdecktes Verändern der Vermittlungszentralen, bietet aber keinerlei Schutz vor beim Entwurf der Programme oder der zahlreichen und komplexen Spezialschaltungen gleich mitgeschaffenen Trojanischen Pferden.

Leider gibt es selbst bei formeller Kooperation aller am Entwurfsprozeß Beteiligten keine allgemein anwendbaren Verfahren, das Schaffen Trojanischer Pferde zu verhindern oder sie, falls sie doch entstanden sind, zu finden oder zumindest den durch sie anrichtbaren Schaden beliebig klein zu halten [Denn_82 Seite 281]. Die bekannten Verfahren zur Erschwerung ihrer Erschaffung, nämlich weitgehend automatisch arbeitende Werkzeuge zur Gewinnung effizient ablauffähiger Implementierungen aus (möglichst kurzen, verständlichen) Problembeschreibungen wie Programmtransformatoren [Freu_87], Übersetzer [WaGo_84] und Chipentwurfshilfsmittel, sollten natürlich eingesetzt werden. Dabei dürfen diese Werkzeuge ihrerseits keine transitiven Trojanischen Pferde enthalten, da sie sie sonst weitgehend unentdeckbar zumindest in ihre genügend komplexen Produkte (transformiertes bzw. übersetztes Programm oder Chip) weitergeben können [Thom_84], was das Problem nur verschlimmern würde.

Außerdem sollte selbstverständlich bereits bezüglich des Entstehungsprozesses aller zum Entwurf und zur Produktion der Vermittlungszentralen verwendeten Werkzeuge verhindert sein, daß Programme mit Trojanischen Pferden in andere Programme, die sie eigentlich nicht zu modifizieren haben, Trojanische Pferde einpflanzen. Wie bereits in [Pfi1_87] kommentiert, ist die Verhinderung dieser Virus-Eigenschaft technisch vollständig möglich – bereits *vor* Beschreibung der sogenannten Computer-Viren in [Cohe_84] wurde in [Denn_82 Seite 317f] skizziert, wie ein Mechanismus zur vollständigen Verhinderung der Virus-Eigenschaft aussehen muß: Programme werden vom autorisierten Generierer durch nur ihm mögliche, aber von allen prüfbare Verschlüsselung (Signatursystem, vgl. Abschnitt 2.2) gegen unerkannte Modifikation geschützt und ihre Unverändertheit zur Laufzeit überprüft.

Da auch dieses Verfahren die Existenz normaler oder transitiver Trojanischer Pferde in keiner Weise ausschließt, sollten bei der Übersetzung und zur Laufzeit von Programmen die in [DeDe_77, Denn_82] beschriebenen Analyse- und Schutzmechanismen implementiert sein und damit die Trojanischen Pferden zur Kommunikation mit ihren Partnern zur Verfügung stehende Bandbreite verkleinert werden, indem legitime Kanäle bzw. Speicherkanäle überwacht bzw. ge-

schlossen werden (es sei dahingestellt, inwieweit dies vollständig und beweisbar geschehen kann, vgl. [HarM_85]). Nicht überwachbar und leider auch in ihrer Bandbreite nicht beliebig verkleinerbar sind die in Abschnitt 1.2 schon erwähnten verborgenen Kanäle (covert channels [Lamp_73, Denn_82 Seite 281]).

Das Problem, das Schaffen Trojanischer Pferde zu verhindern, sie zu finden oder zumindest auszuschalten, ist also mit den bekannten informatischen Methoden nicht vollständig lösbar. Es wiegt umso schwerer, je größer die Entwurfs- und Produktionskomplexität der betrachteten Systemteile ist. Dies legt nahe, nach Alternativen zum geschilderten zentralistischen Ansatz zu suchen. Gibt es einen Handel mit geheimem „Know-how“ (was heute bezüglich der Informationstechnik der Normalfall ist, vgl. die in Abschnitt 1.2 im Zitat aus [BfD_86] angesprochene Vereinbarung zwischen der DBP und IBM), ist folglich die formelle Kooperation aller am Entwurfsprozeß Beteiligten zwangsläufig nicht gegeben oder zumindest stark eingeschränkt, so sind Alternativen zum zentralistischen Ansatz unabdingbar.

Selbst wenn Trojanische Pferde in den Vermittlungszentralen vollständig vermieden werden konnten und diese nicht frei speicherprogrammierbar, sondern mittels ROMs fest speicherprogrammiert oder überhaupt nicht programmierbar ausgelegt wurden, so daß nicht einfach durch Neuladen der Betriebssoftware (z. B. mittels Fernwartung) die Funktionalität beliebig geändert werden kann, können Netzbetreiber oder Geheimdienste durch Installation von Zusatzgeräten, Austausch von ROMs oder einiger Spezialschaltungen innerhalb der Vermittlungszentralen das Kommunikationsnetz ohne Einwilligung der Netzbenutzer oder detaillierte Rechtsgrundlage unentdeckt in seiner Funktion erweitern. Letzteres kann dadurch vermieden werden, daß die Vermittlungszentralen in auf Manipulationsversuche reagierenden und unerwünschte elektromagnetische oder mechanische Abstrahlung unter die Sensitivität professioneller Empfangsgeräte dämpfenden Behältern (kurz: unmanipulierbaren Gehäusen, tamper responding container [Kent_80, Chau_84, DaPr_84 Seite 3, 80, Simm_85, HeFi_80, Wein_87]) untergebracht werden. Dies erschwert aber die notwendige Wartung der Vermittlungszentralen, außerdem sind solche Behälter nur unter Schwierigkeiten öffentlich zu validieren und verglichen mit Mikroelektronik sowohl sehr teuer als auch bei weitem nicht vergleichbar schnell im Preis fallend [Pfi1_85 Seite 7]. Datenschutz ist also in solch einem Kommunikationsnetz trotz teils aufwendiger Zusatzmaßnahmen nicht überprüfbar.

Dies ändert sich auch dann nicht, wenn, wie ebenfalls in [KuRo_86, Kub2_86] vorgeschlagen, Teilnehmer, die Dienste von etwa der Übertragungsleistung des geplanten ISDN in Anspruch nehmen wollen, über zwei Leitungen und zwei unterschiedliche Vermittlungszentralen angeschlossen werden, und damit zumindest im Teilnehmeranschlußbereich dienstespezifische Netze errichtet werden. Der Aufwand eines *Angreifers* würde dadurch allenfalls verdoppelt, was bei einer ebenfalls zu erwartenden näherungsweisen Verdopplung der Kosten zur Errichtung und zum Unterhalt der Netze alles andere als effizient ist. Es ist fast überflüssig zu erwähnen, daß eine Verallgemeinerung von zwei auf *n* Kommunikationsnetze den Effizienzmangel lediglich eskaliert. Wünschenswert ist also ein Kommunikationsnetz, das trotz Diensteintegration Sicherheit und Datenschutz dienstespezifisch in dem gewünschten Umfang bietet. Wir werden im folgenden sehen, daß dies gerade wegen der Diensteintegration effizient erreicht werden kann.

Der von mir verfolgte **dezentrale Ansatz** verlagert (verglichen mit dem zentralistischen Ansatz) möglichst viele Funktionen in (aus der Sicht des Netzbenutzers) **eigene vertrauenswürdige Geräte**.

Das Sicherheitsproblem kann – wie üblich und deshalb in dieser Arbeit nur gelegentlich angesprochen – durch Fehlertoleranz-Maßnahmen und sichere (digitale) Unterschriften gelöst werden. Bei letzteren werden Zeugen nicht mehr für einzelne Geschäftsvorfälle, sondern nur zur völlig dezentral durchführbaren Authentikation der Unterschrift benötigt. Dadurch wird den Zeugen erst gar keine Möglichkeit gegeben, zu „Großen Brüdern“ zu werden.

Das Datenschutzproblem wird gelöst, indem die Erfassungsmöglichkeit aller überflüssigen Daten verhindert wird.

Mit der hergeleiteten informatischen Problemstellung, den dabei erläuterten Begriffen, den zwei geschilderten komplementären Ansätzen zur Lösung des Sicherheits- und Datenschutzproblems sowie den diskutierten Schutz-Maßnahmen im Sinn können wir uns in den folgenden zwei Abschnitten zuerst den „klassischen“ Hilfsmitteln zur Lösung des Datenschutz- und Sicherheitsproblems zuwenden und danach ihren Einsatz, die durch sie wirklich gelösten Probleme und die verbleibenden noch zu lösenden Probleme analysieren.

2.2 Hilfsmittel aus der Kryptographie

Kommunikation wird technisch üblicherweise durch Verschlüsselung der Daten vor unbefugter Kenntnisnahme sowie vor unerkannter unbefugter Veränderung geschützt [VoKe_83].

Hierzu werden Kryptosysteme, das sind Familien von Chiffrierfunktionen zur Ver- und Entschlüsselung, vereinbart. Die zu verwendenden Chiffrierfunktionen des Kryptosystems werden durch sogenannte Schlüssel ausgewählt. Voraussetzung jeglichen Schutzes durch Verschlüsselung ist, daß die verwendeten Schlüssel geeignet generiert und verteilt werden und die verwendeten Kryptosysteme die Eigenschaft haben, daß Ver- und Entschlüsselung ohne Kenntnis der Schlüssel ebenso wie das Folgern der verwendeten Schlüssel aus passiver Beobachtung und aktiver Beeinflussung der Kommunikation praktisch unmöglich sind.

Zunächst werden deshalb zwei Klassen von Kryptosystemen sowie die jeweils zugehörige Schlüsselverteilung beschrieben. Danach werden die Eigenschaften von Kryptosystemen genauer untersucht.

2.2.1 Kryptosysteme und ihre Schlüsselverteilung

Verschlüsselung der Daten mittels eines **Kryptosystems** zum Zwecke der *Konzelation* (Geheimhaltung) [Riha_84 Seite 22] bzw. *Integrität* soll garantieren, daß der Inhalt einer Nachricht nur den Besitzern eines bestimmten Schlüssels zugänglich ist bzw. ohne Kenntnis des Schlüssels nicht unerkennbar verändert werden kann. Damit der Inhalt einer verschlüsselten Nachricht vom Empfänger wiedergewonnen werden kann, müssen die zur Verschlüsselung verwendeten Chiffrierfunktionen eines zum Zwecke der Konzelation verwendeten Kryptosystems invertierbar, d. h. injektiv sein.

Hinsichtlich der erlaubten und möglichen Verteilungen ihrer Schlüssel unterscheidet man zwischen **symmetrischen** und **asymmetrischen** Kryptosystemen. Erstere werden oftmals auch konventionelle Kryptosysteme, letztere Kryptosysteme mit öffentlichen Schlüsseln genannt.

Für manche Anwendungen ist die Forderung nach Integrität des Nachrichteninhalts nicht ausreichend. Bei ihnen darf eine gesendete Nachricht nur von dem (bzw. den) Besitzer(n) eines bestimmten Schlüssels stammen. Daß sie dies tut muß auch einer unbeteiligten Dritten Partei beweisbar sein. Diese *Authentifikation* der Nachricht und ihres Inhalts soll durch Verschlüsselung der Daten mittels eines **Signatursystems** (spezielles asymmetrisches Kryptosystem) garantiert werden.

Zur Beurteilung der Stärke von Kryptosystemen wird unterstellt, daß der Angreifer die Familien der verwendeten Chiffrierfunktionen sowie die Wahrscheinlichkeitsverteilung der Schlüssel und der zu verschlüsselnden Nachrichten kennt. Es werden folgende *Angriffstypen* betrachtet [Denn_82 Seite 2f, DaPr_84 Seite 36f, GoMT_82, GoMR_84, GoMR_88, Bras_88 Seite 8f, 27]:

Es gibt die weniger mächtigen *passiven* Angriffe, bei denen der Angreifer entweder nur verschlüsselte Daten besitzt und aus diesen allein die unverschlüsselten gewinnen will (Schlüsseltext-Angriff, ciphertext-only attack) oder aber zusätzlich auch viele Klartext-Schlüsseltext-Paare erfährt und mit deren Hilfe die unverschlüsselten gewinnen will (Klartext-Schlüsseltext-Angriff, known-plaintext attack).

Es gibt die mächtigeren *aktiven* Angriffe, bei denen der Angreifer zunächst zu von ihm im ersten Schritt gewählten Klartexten im zweiten Schritt die jeweils zugehörigen Schlüsseltexte (gewählter Klartext-Schlüsseltext-Angriff, chosen-plaintext attack) und/oder zu von ihm im ersten Schritt gewählten Schlüsseltexten im zweiten Schritt die jeweils zugehörigen Klartexte (gewählter Schlüsseltext-Klartext-Angriff, chosen-ciphertext attack) erhält und danach im dritten Schritt andere Schlüsseltexte zu entschlüsseln oder Klartexte zu verschlüsseln versucht. Die Gefährlichkeit dieser aktiven Angriffe wird erheblich gesteigert, wenn der Angreifer seine Klar- und/oder Schlüsseltexte *adaptiv* (adaptive) wählen darf: Zunächst wählt er, dann erhält er Antwort, dann darf er nochmals wählen, erhält abermals Antwort, usw. Eine zur bisherigen Einteilung aktiver Angriffe orthogonale ist die, ob der Angreifer schon vor seinem aktiven Angriff die von ihm schlußendlich zu entschlüsselnden Schlüsseltexte bzw. die zu verschlüsselnden Klartexte kennt (*nachrichtenbezogener* aktiver Angriff) oder erst danach erfährt (*schlüsselbezogener* aktiver Angriff). Im ersteren Fall hat der Angreifer es möglicherweise sehr viel leichter.

In den folgenden Abschnitten und Kapiteln wird sich herausstellen, daß es manche Situationen gibt, wo ein sich adaptiv verhaltender aktiver Angreifer von einem normalen Benutzer nicht zu unterscheiden und die ihn interessierende Nachricht ihm schon vorher bekannt ist. Zumindest dann müssen Kryptosysteme verwendet werden, die auch einem nachrichtenbezogenen adaptiven aktiven Angriff standhalten.

2.2.1.1 Symmetrische Kryptosysteme

In einem symmetrischen Kryptosystem (Bild 3), wozu alle klassischen Kryptosysteme gehören [Denn_82, DaPr_84, Hors_85, Bras_88], wird die Kommunikation zwischen zwei Partnern dadurch vor unbefugter Kenntnisnahme oder unerkennbarer Verfälschung durch Dritte gesichert, daß beide einen gemeinsamen geheimen Schlüssel kennen, der sowohl zur Ver- als auch zur Entschlüsselung dient.

Um Verfälschungen durch Dritte ohne Schlüsselkenntnis erkennen zu können, müssen Klartexte so redundant codiert sein oder werden, daß eine Änderung des Schlüsseltextes mit sehr hoher Wahrscheinlichkeit einen „Klartext“ mit unpassender Redundanz ergibt. Konzelation und Integrität können auf diese Weise ohne Schwierigkeit erreicht werden, nicht aber Authentifikation: Da der Empfänger einer Nachricht den zugehörigen Schlüsseltext genauso gut hätte selbst bilden können, da der ihm bekannte Schlüssel zum Verschlüsseln genauso befähigt wie zum Entschlüsseln, kann er einer unbeteiligten Dritten Partei nicht beweisen, daß er dies nicht getan hat, sondern die Nachricht vom Sender (oder einer vom Sender zur Verschlüsselung befähigten Instanz) gebildet wurde.

Die Anforderungen an ein symmetrisches Kryptosystem lauten präziser formuliert:

1. Für alle Klartexte x und alle Schlüssel k (key) gilt:
   Anwendung der durch Schlüssel k aus der Familie der Chiffrierfunktionen zur Verschlüsselung ausgewählten Funktion auf x ergibt den Schlüsseltext. Anwendung der durch Schlüssel k aus der Familie der Chiffrierfunktionen zur Entschlüsselung ausgewählten Funktion auf den Schlüsseltext ergibt wieder den Klartext. In funktionaler Schreibweise, bei der die Verwendung der Verschlüsselungsfunktion implizit angenommen und die Verwendung der Entschlüsselungsfunktion durch $^{-1}$ symbolisiert wird, lautet dies kürzer und übersichtlicher: $k^{-1}(k(x)) = x$

2. Kennt ein Angreifer einen Schlüssel k vor seinem passiven Angriff nicht, so ist es ihm trotz Kenntnis der Familie von Chiffrierfunktionen zur Ver- und Entschlüsselung auch nach Erhalt vieler Klartext-Schlüsseltext-Paare x_i, $k(x_i)$ praktisch unmöglich, irgendeinem anderen Klartext x den Schlüsseltext $k(x)$ oder irgendeinem anderen Schlüsseltext $k(x)$ den Klartext x mit größerer Wahrscheinlichkeit als durch pures Raten zuzuordnen. Dies impliziert, daß er auch den Schlüssel k nicht ermitteln kann.

3. Kennt ein Angreifer einen Schlüssel k vor seinem aktiven Angriff nicht, so ist es ihm trotz Kenntnis der Familie von Chiffrierfunktionen zur Ver- und Entschlüsselung auch nach (adaptiver) Wahl vieler Klartexte x_i und Erhalt der zugehörigen Schlüsseltexte $k(x_i)$ sowie nach (adaptiver) Wahl vieler Schlüsseltexte y_j und Erhalt der zugehörigen Klartexte $k^{-1}(y_j)$ praktisch unmöglich, irgendeinem anderen Klartext x den Schlüsseltext $k(x)$ oder irgendeinem anderen Schlüsseltext $k(x)$ den Klartext x mit größerer Wahrscheinlichkeit als durch pures Raten zuzuordnen. Dies impliziert, daß er auch den Schlüssel k nicht ermitteln kann.

Klarerweise impliziert eine Erfüllung der 3. die Erfüllung der 2. Anforderung.

Obwohl die Erfüllung der 1. und 2. Anforderung für manche Anwendungen auszureichen scheint, halte ich es nicht für sinnvoll, symmetrische Kryptosysteme zu verwenden, die (nach gründlicher öffentlicher Untersuchung) nicht auch die 3. Anforderung erfüllen.

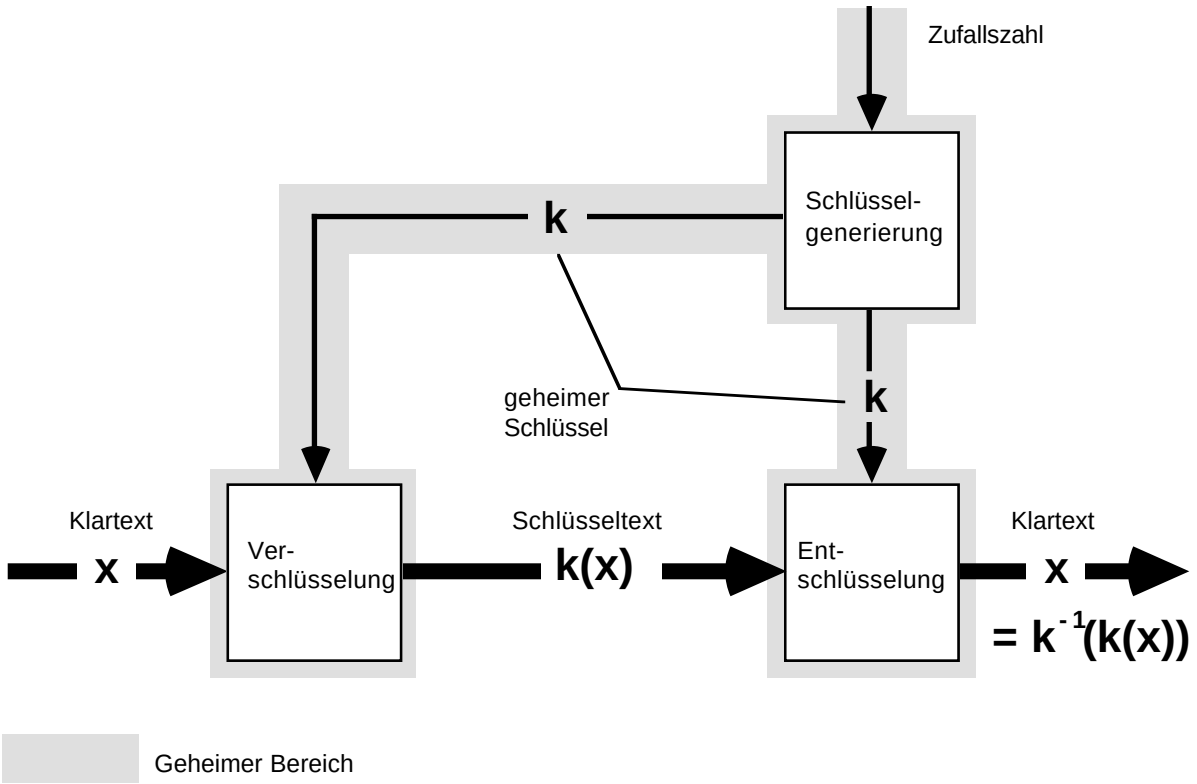


Bild 3: Symmetrisches Kryptosystem

Bei Verwendung eines symmetrischen Kryptosystems in einem offenen Kommunikationsnetz ist es völlig sinnlos, wenn ein Teilnehmer für alle seine Kommunikationsbeziehungen denselben Schlüssel benutzt. Das Bestmögliche und in offenen Kommunikationsnetzen Nötige ist, für jede Kommunikationsbeziehung einen anderen Schlüssel zu verwenden. Um eine Kommunikationsbeziehung gesichert beginnen zu können, müssen sich beide Partner zuvor auf einen gemeinsamen Schlüssel einigen. Bei offenen diensteintegrierenden Kommunikationsnetzen muß dies über das Netz selbst erfolgen, da man nicht davon ausgehen kann, daß die Partner vorher schon in direktem Kontakt miteinander standen. Für dieses Schlüsselverteilproblem existieren im wesentlichen zwei Lösungsansätze.

Der klassische sieht die Verwendung einer Schlüsselverteilzentrale vor, mit der jeder Teilnehmer vor Beginn seiner Teilnahme am Netzgeschehen außerhalb des Netzes einen Schlüssel vereinbart hat. Auf Anfrage generiert sie einen Schlüssel für eine Kommunikationsbeziehung und teilt ihn den künftigen Kommunikationspartnern unter Verwendung der mit den jeweiligen Teilnehmern vereinbarten Schlüssel in Konzelation und Integrität garantierender Weise mit [DaPr_84, NeSc_87, OtRe_87].

Da eine solche Zentrale durch Kenntnis aller verwendeten Schlüssel potentiell alle gesendeten Nachrichten entschlüsseln, verändern und verschlüsseln kann, ist aus Datenschutz- und Sicherheitsgründen diese einfache Lösung zumindest für diensteintegrierende offene Netze nicht akzeptabel.

Ein Ausweg ist die Verwendung vieler unabhängiger Schlüsselverteilzentralen, die je einen Schlüssel generieren und beiden Kommunikationspartnern mitteilen (Bild 4). Die Kommunikationspartner verwenden als Schlüssel die Summe aller mitgeteilten Schlüssel, so daß alle Schlüsselverteilzentralen zusammenarbeiten müßten, um die Summe zu errechnen und die Kommunikation zu überwachen [DiH1_76].

Der zweite Ansatz verwendet zur Schlüsselverteilung ein asymmetrisches Kryptosystem (genauer: ein asymmetrisches Konzelationssystem und Signatursystem), wie es in Abschnitt 2.2.1.2 beschrieben wird.

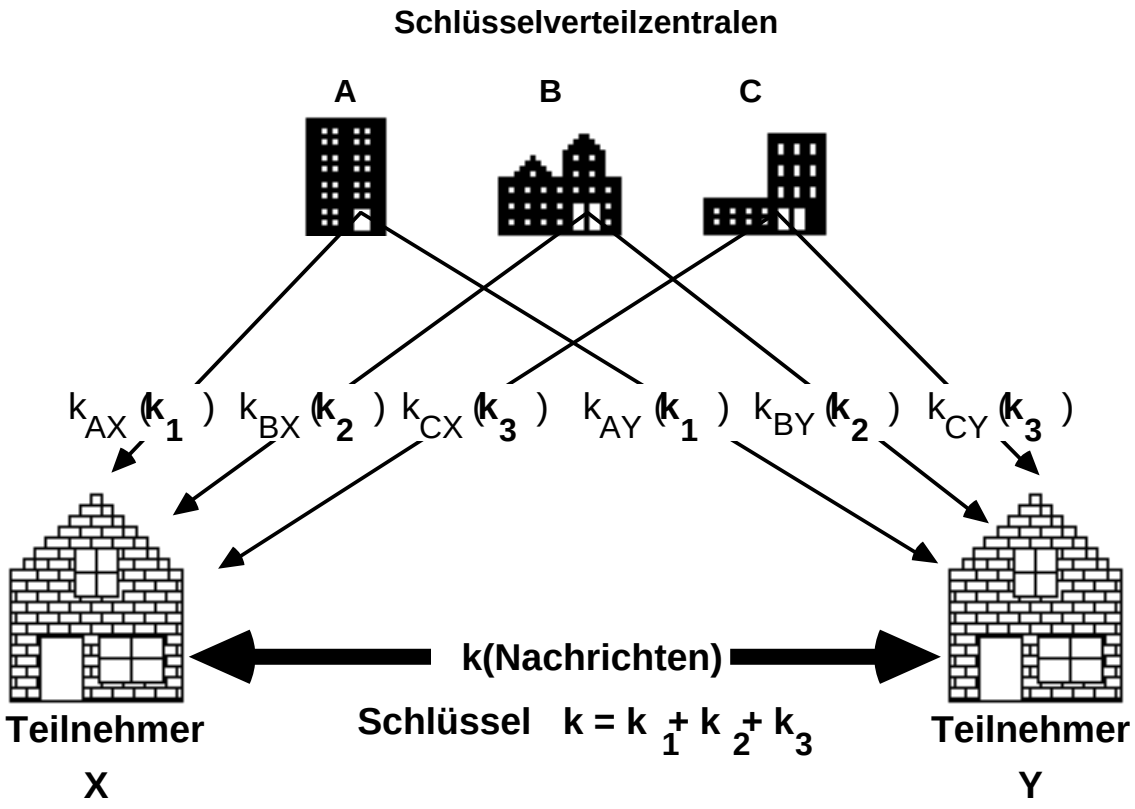


Bild 4: Schlüsselverteilung bei symmetrischem Kryptosystem

Das bekannteste moderne symmetrische Kryptosystem ist DES (Data Encryption Standard) [DES_77, DaPr_84, Hors_85], das von der amerikanischen Normungsbehörde für den öffentlichen Bereich (NBS = National Bureau of Standards) als Zwischenlösung (bis zur Normung eines besseren) definiert wurde. Seine Sicherheit ist zwar nicht bewiesen, es hat aber bis heute allen (bekanntgegebenen) Versuchen standgehalten, es auch mittels adaptiver aktiver Angriffe schneller als durch Durchprobieren seines Schlüsselraumes von 2^{56} Schlüsseln zu bre-

chen. Ob dieses Durchprobieren mittels Spezialrechnern schon 1980 preiswert möglich war oder erst 1995 sein wird, wird heftig diskutiert [DiH2_76, DiHe_77, DaPr_84 Seite 73ff].

2.2.1.2 Asymmetrische Kryptosysteme

Die Idee der asymmetrischen Kryptosysteme wurde erst 1976 veröffentlicht [DiHe_76] und löst das Schlüsselverteilproblem auf überraschend einfache Art: Statt je einen einzigen Schlüssel zum Ver- und Entschlüsseln zu verwenden, verteilt man diese Funktion auf je zwei zusammengehörige Schlüssel. Gelingt dies in einer Art und Weise, daß man zumindest bezüglich eines von beiden keine realistische Möglichkeit hat, aus ihm den anderen herzuleiten, so kann dieser eine veröffentlicht werden.

Während bei jedem symmetrischen Kryptosystem sowohl Konzelation als auch Integrität erreicht werden kann (und es im allgemeinen Fall sogar unmöglich ist, beides zu trennen), ist dies bei asymmetrischen Kryptosystemen nicht der Fall:

- Nur solche asymmetrischen Kryptosysteme ermöglichen *Konzelation*, bei denen der Schlüssel zur Entschlüsselung realistischerweise nicht aus dem zur Verschlüsselung hergeleitet werden kann und die – wie bereits in Abschnitt 2.2.1 erwähnt – invertierbare, d. h. injektive Chiffrierfunktionen zur Verschlüsselung verwenden. Im folgenden werden sie **asymmetrische Konzelationssysteme** genannt und in Abschnitt 2.2.1.2.1 ausführlicher behandelt.

- Nur solche asymmetrischen Kryptosysteme ermöglichen *Integrität*, bei denen der Schlüssel zur Verschlüsselung realistischerweise nicht aus dem zur Entschlüsselung hergeleitet werden kann. Im folgenden werden sie **Signatursysteme** genannt und in Abschnitt 2.2.1.2.2 ausführlicher behandelt. Diese asymmetrischen Kryptosysteme ermöglichen sogar mehr als Integrität, nämlich *Authentifikation*: Verschlüsselung der Daten soll garantieren, daß der Inhalt einer gesendeten Nachricht nur von dem (bzw. den) Besitzer(n) eines bestimmten Schlüssels stammen und dies auch einem unbeteiligten Dritten bewiesen werden kann.
  Bei Signatursystemen ist es nicht nötig, daß die verschlüsselte (= digital unterschriebene) Nachricht dazu verwendet werden kann, die nicht verschlüsselte Nachricht durch Entschlüsselung wiederzugewinnen, da die nicht verschlüsselte Nachricht mit der verschlüsselten übertragen werden kann. Konzelation muß ggf. durch eine zusätzliche Verschlüsselung mit einem anderen Kryptosystem erreicht werden. Im allgemeinen Fall erhält die „Ent“schlüsselungsfunktion eines Signatursystems neben dem Schlüssel zur „Ent“schlüsselung als Eingabe die Nachricht und die verschlüsselte (= digital unterschriebene) Nachricht und gibt TRUE aus, sofern die Unterschrift stimmt, und FALSE anderenfalls.

Da es bei asymmetrischen Kryptosystemen für Konzelation und Authentifikation zweier verschiedener Verschlüsselungen (mit ggf. unterschiedlichen Sorten von asymmetrischen Kryptosystemen) bedarf, können Konzelation und Integrität (als Abschwächung der Authentifikation) getrennt werden. Möchte man die Kommunikation sowohl gegen unbefugte Kenntnisnahme des

Kommunikationsinhalts als auch vor unerkennbarer Veränderung schützen, so wird man den Nachrichteninhalt günstigerweise zuerst zum Zwecke der Authentifikation und das Ergebnis dieser Verschlüsselung dann noch einmal zum Zwecke der Konzelation verschlüsseln.

Durch ein asymmetrisches Kryptosystem entsteht die Möglichkeit, einen Schlüssel, oder genauer ein Schlüsselpaar, statt einer Kommunikationsbeziehung einem einzelnen Teilnehmer zuzuordnen, der sich zudem diesen Schlüssel selbst generieren kann. Möchte jemand mit diesem Teilnehmer gesichert kommunizieren, so muß er sich lediglich dessen veröffentlichten Schlüssel (im Falle gewünschter Konzelation und Authentifikation sowie unterschiedlicher asymmetrischer Kryptosysteme zur Konzelation und Authentifikation: dessen zwei veröffentlichte Schlüssel) besorgen. Dies kann entweder durch eine offene Anfrage an den gewünschten Teilnehmer geschehen, wodurch allerdings Authentifikationsprobleme entstehen [Inge_84, RiSh_84], oder unter Verwendung zentraler, gegen Manipulation gesicherter Register, deren Verwalter jetzt durch die Kenntnis der veröffentlichten Schlüssel keine Möglichkeit zum Mithören oder unbemerkbaren Manipulation verschlüsselter Nachrichten mehr erhalten.

1978 wurde von Ronald L. Rivest, Adi Shamir, Leonard M. Adleman das erste und nach wie vor bekannteste asymmetrische Kryptosystem (Ver- und Entschlüsselungsfunktion sowie Schlüsselgenerierungsfunktion entsprechend obigen Anforderungen) veröffentlicht, das nach den Anfangsbuchstaben seiner Entdecker allgemein mit RSA bezeichnet wird [RSA_78, Denn_82, DaPr_84, Hors_85, Bras_88]. Es ist sowohl als asymmetrisches Konzelationssystem als auch als Signatursystem verwendbar. Seine Sicherheit beruht auf (teilweise unbewiesenen) Annahmen über den Lösungsaufwand zahlentheoretischer Probleme. Es wird allgemein vermutet, daß das *schlüsselbezogene* Brechen von RSA, d. h. das Ableiten des nicht veröffentlichten aus dem veröffentlichten Schlüssel, auch mittels adaptiver aktiver Angriffe so schwer ist wie die Gewinnung der Primfaktoren einer gegebenen Zahl. Eine Faktorisierung vorgebener Zahlen, die aus sehr großen Primfaktoren zusammengesetzt sind, ist bisher praktisch unmöglich. RSA ist auch mittels passiver Angriffe schlüsselbezogen gebrochen, sollte dies möglich werden. Je nach seiner Verwendung ist *nachrichtenbezogenes* Brechen von RSA, das heißt die Entschlüsselung einer Nachricht bei Verwendung als asymmetrisches Konzelationssystem bzw. die Erzeugung einer unterschriebenen Nachricht bei Verwendung als Signatursystem auch mittels nicht-adaptiver aktiver Angriffe ohne die Fähigkeit zur Faktorisierung möglich. Der für das folgende relevante Angriff bei Verwendung als Konzelationssystem wird in Abschnitt 2.2.1.2.1 beschrieben.

Es sei angemerkt, daß es weniger bekannte asymmetrische Kryptosysteme als RSA gibt, für die bewiesen wurde, daß ihr Brechen so schwer wie Faktorisierung ist: in [BlGo_85, Will_85] werden asymmetrische Konzelationsysteme beschrieben, die allerdings nur passiven Angriffen standhalten und bei adaptiven aktiven schlüsselbezogen gebrochen werden, in [GoMR_84, Gol1_87, GoMR_88] Signatursysteme, die auch adaptiven aktiven Angriffen standhalten.

Damit ist der Einsatz von RSA als Konzelationssystem nur dann angebracht, wenn mit unbemerkbaren adaptiven aktiven Angriffen gerechnet werden muß. Der Einsatz von RSA als Signatursystem ist nur noch da angebracht, wo, wie bei den in Abschnitt 8.1 und 8.3 beschriebenen Anwendungen, spezielle Eigenschaften von RSA benötigt werden.

2.2.1.2.1 Asymmetrische Konzelationssysteme

Statt einen einzigen Schlüssel zum Ver- und Entschlüsseln zu verwenden, verteilt man diese Funktion auf zwei zusammengehörige Schlüssel. Der eine soll nur zum Entschlüsseln dienen und muß natürlich geheimgehalten werden. Er wird **Dechiffrierschlüssel** (in manchen Texten auch privater Schlüssel oder geheimer Schlüssel) genannt und im folgenden mit *d* bezeichnet. Der andere hingegen soll nur das Ver-, nicht jedoch das Entschlüsseln ermöglichen, weshalb er veröffentlicht werden kann. Er wird als **Chiffrierschlüssel** (in manchen Texten auch öffentlicher Schlüssel oder allbekannter Schlüssel) genannt und im folgenden mit *c* bezeichnet (Bild 5). Insbesondere darf man also keine realistische Möglichkeit haben, ein unbekanntes *d* aus dem zugehörigen *c* herleiten zu können.

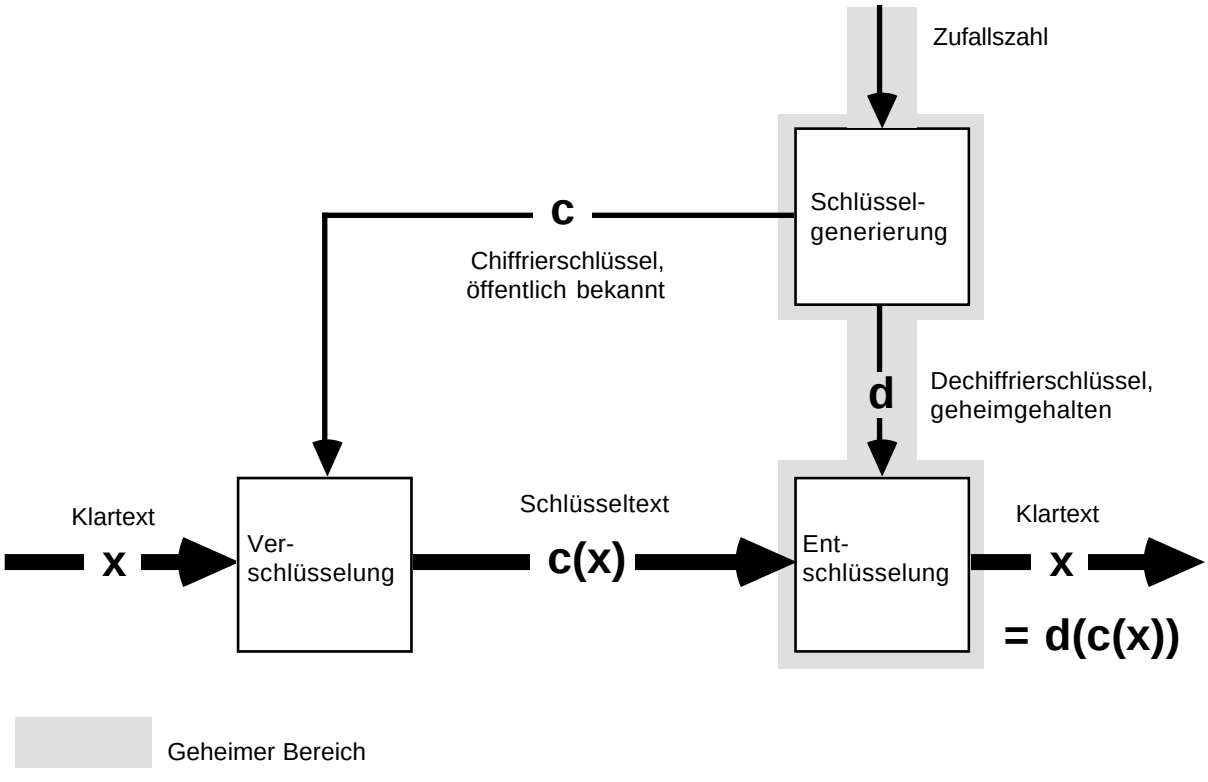


Bild 5: Asymmetrisches Konzelationssystem

Rein theoretisch könnte man *d* aus *c* natürlich immer durch Durchprobieren aller möglichen Chiffrierschlüssel bestimmen, denn nur das richtige *d* entschlüsselt alle mit *c* verschlüsselten Nachrichten richtig. Es muß daher so viele Möglichkeiten geben, daß dies praktisch keinerlei Erfolg verspricht. Außerdem müssen, damit niemand kurze Standardnachrichten erraten und mit dem öffentlich bekannten Chiffrierschlüssel testen kann, Nachrichten mehrere Verschlüsselungen besitzen. Dies kann man erreichen, indem man die Nachrichten vor der Verschlüsselung mit zufällig gewählten Zeichenketten verlängert oder indem man ein bereits *indeterministisch*

verschlüsselndes asymmetrisches Konzelationssystem (probabilistic encryption, [BlGo_85]) verwendet. In beiden Fällen hat jede Klartextnachricht auch bei festgehaltenem Chiffrierschlüssel sehr viele (genauer: in der Länge der zufälligen Verlängerung bzw. eines Sicherheitsparameters exponentiell viele) Schlüsseltexte.

Einfacher als das Herleiten von *d* aus *c* muß hingegen das Generieren eines beliebigen Paares von zueinander passenden *d* und *c* sein, da dies der Schlüsselbesitzer am Anfang selbst durchführen muß.

Nachdem nun eine Notation für Verschlüsselung mit einem asymmetrischen Konzelationssystem eingeführt ist, zeigt Bild 6 ein Vorgehen bei der Schlüsselverteilung, wenn die Kommunikation zwischen den Teilnehmern lediglich gegen unbefugte Kenntnisnahme des Nachrichteninhalts gesichert werden soll.

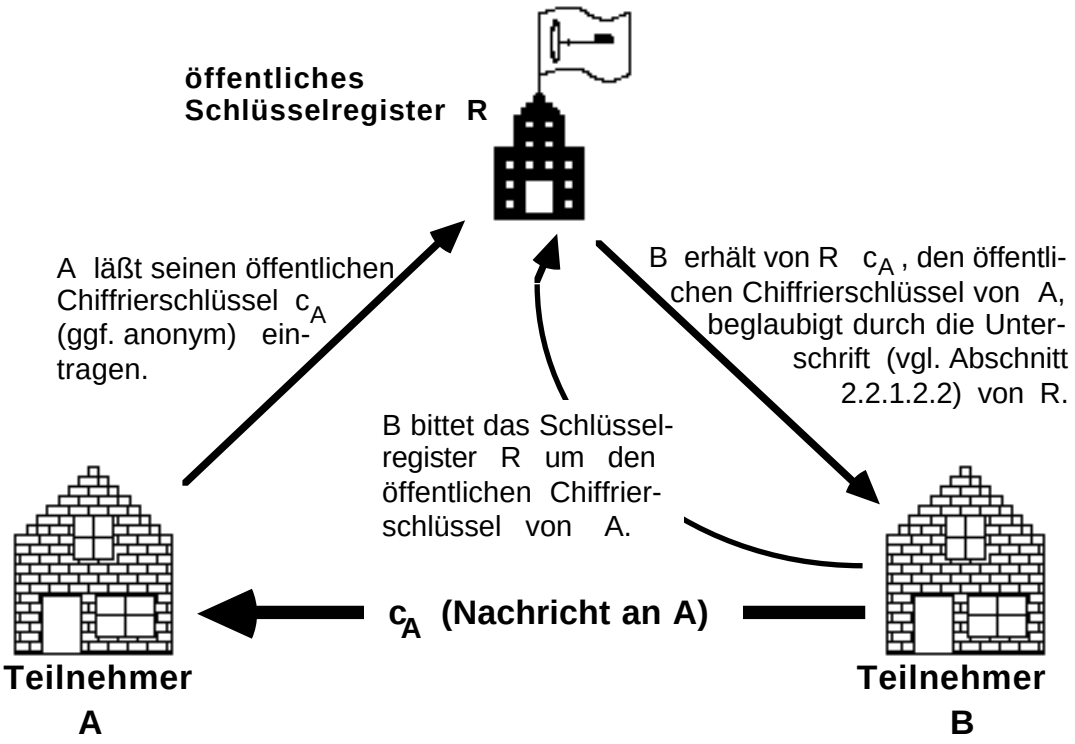


Bild 6: Schlüsselverteilung bei asymmetrischem Konzelationssystem

Am Beispiel von RSA wird gezeigt, wie ein erfolgreicher aktiver Angriff gegen ein asymmetrisches Konzelationssystem geführt werden kann. Im Falle RSA nutzt dieser Angriff von Davida [Merr_83, Denn_84, Bras_88] aus, daß bei RSA nicht nur für alle Klartexte *x* und jedes Schlüsselpaar *c*, *d* die Gleichung $d(c(x)) = x$ gilt, sondern es gilt auch für alle Klartexte *x*, *y* die zusätzliche Gleichung $c(x) \bullet c(y) = c(x \bullet y)$ (d. h. die Verschlüsselung mit RSA bildet einen Homomorphismus bezüglich der Multiplikation innerhalb des durch den öffentlich bekannten Chiffrierschlüssel *c* bestimmten Restklassenrings). Erhält ein aktiver Angreifer nun den für den Besitzer des Dechiffrierschlüssels *d* bestimmten Schlüsseltext $c(x)$, so ist er mit folgendem nicht-adaptiven gewähltem Schlüsseltext-Klartext-Angriff erfolgreich: er wählt sich einen belie-

bigen Klartext y, bildet $c(y)$ und sodann $c(x)•c(y)$. Dies läßt er sich vom Besitzer von d entschlüsseln und erhält $d(c(x)•c(y)) = d(c(x•y)) = x•y$. Da er y kennt, kann er $x•y$ durch y dividieren und so x erhalten. Damit ist RSA nachrichtenbezogen gebrochen.

Es ist beachtenswert, daß der Besitzer von d keinerlei Verdacht schöpfen kann, wenn er beliebige Schlüsseltexte entschlüsselt und ausgibt, da $c(x)$ und $c(x)•c(y)$ für ihn perfekt unverkettbar sind: Für alle Schlüsseltexte w gibt es genau einen Klartext y, so daß $w = c(x)•c(y)$. Denn es gilt $w = c(x)•c(y) \Leftrightarrow c(x)^{-1}•w = c(y) \Leftrightarrow d(c(x)^{-1}•w) = d(c(y)) \Leftrightarrow d(c(x)^{-1}•w) = y$.

Das nachrichtenbezogene Brechenvon RSA mittels dieses Angriffs kann vereitelt werden, indem Klartexte ein geeignetes Redundanzprädikat erfüllen müssen. Dann erfüllt das Produkt $x•y$ dies Redundanzprädikat mit an Sicherheit grenzender Wahrscheinlichkeit nicht, so daß es der Besitzer des Dechiffrierschlüssels nicht ausgibt.

2.2.1.2.2 Signatursysteme

Statt einen einzigen Schlüssel zum Ver- und Entschlüsseln zu verwenden, verteilt man diese Funktion auch beim Signatursystem auf zwei zusammengehörige Schlüssel. Der eine soll nur zum Verschlüsseln (in diesem Zusammenhang wird auch *signieren* und gleichbedeutend auch *unterschreiben* gesagt) dienen, weshalb er Schlüssel zum Signieren oder (kürzer:) **Signierschlüssel** (in manchen Texten auch Unterschriftenschlüssel, privater oder geheimer Schlüssel) genannt und im folgenden mit s bezeichnet wird. Er muß, wie manche alternativen Namen schon sagten, geheimgehalten werden. Der andere Schlüssel hingegen soll nur das „Ent“-, nicht jedoch das Verschlüsseln ermöglichen, weshalb er veröffentlicht werden kann und, da er zum Testen der Echtheit der Unterschrift verwendet wird, als Schlüssel zum Testen der Unterschrift oder (kürzer:) **Testschlüssel** t (in manchen Texten auch Prüfschlüssel, öffentlicher oder allbekannter Schlüssel) bezeichnet wird (Bild 7). Insbesondere darf man hier also keine realistische Möglichkeit haben, einen unbekannten Verschlüsselungsschlüssel s aus dem zugehörigen „Ent“schlüsselungsschlüssel t herleiten zu können.

Rein theoretisch könnte man s aus t natürlich immer durch Durchprobieren aller möglichen Schlüssel bestimmen, denn nur das richtige s verschlüsselt alle Texte so, daß sie den Test mit t bestehen. Es muß daher so viele Möglichkeiten geben, daß dies praktisch keinerlei Erfolg verspricht.

Einfacher als das Herleiten von s aus t muß hingegen das Generieren eines beliebigen Paares von zueinander passenden s und t sein, da dies der Schlüsselbesitzer am Anfang selbst durchführen muß.

Nachdem nun eine Notation für Signatursysteme eingeführt ist, zeigt Bild 8 ein Vorgehen bei der Schlüsselverteilung, wenn die Kommunikation zwischen den Teilnehmern lediglich gegen unbefugte Veränderung des Nachrichteninhalts gesichert werden soll.

In den Bildern 7 und 8 (sowie an allen entsprechenden Stellen in den folgenden Bildern und Texten) bedeutet ein *Komma* zwischen Nachrichtenteilen *Konkatenation von Zeichenketten*, wobei je nach Anwendung die Konkatenationsstelle auch für den Empfänger sichtbar oder auch unsichtbar sein kann.

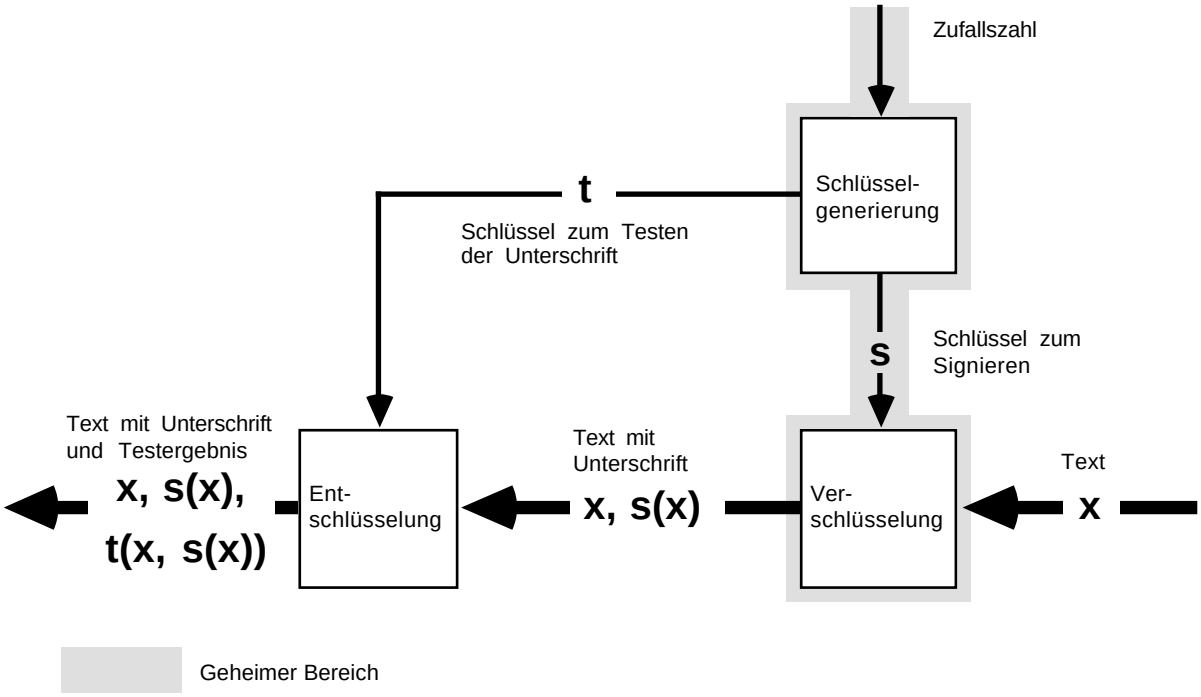


Bild 7: Signatursystem

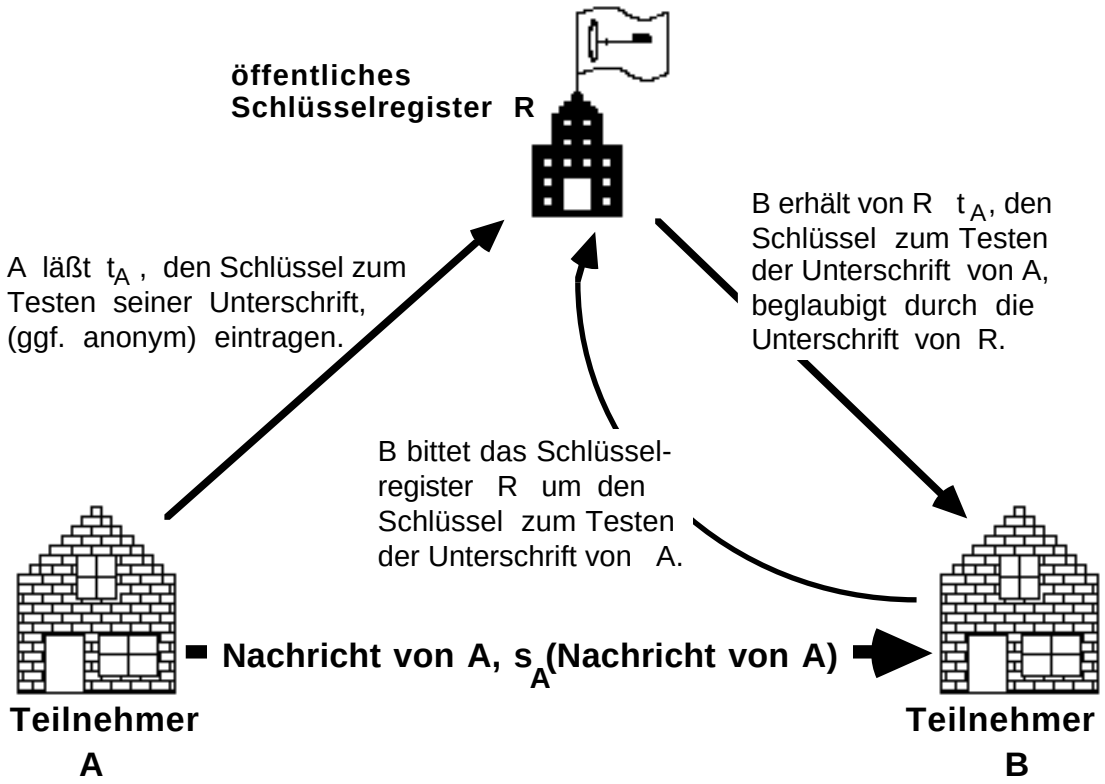


Bild 8: Schlüsselverteilung bei Signatursystem

2.2.2 Eigenschaften von Kryptosystemen

Im vorhergehenden Abschnitt 2.2.1 wurden die im folgenden benötigten Sorten von Kryptosystemen zunächst allein unter den Gesichtspunkten „Zweck der Verschlüsselung“ und „Möglichkeiten der Schlüsselverteilung“ eingeteilt und beschrieben, was in Bild 9 zusammengefaßt ist.

Kryptosysteme und ihre Ziele \ Kenntnis des Schlüssels	einer (Generierer)	zwei (beide Partner)	alle (öffentlich bekannt)
asymmetrisches Konzelationssystem: Konzelation	**d** Dechiffrierschlüssel		**c** Chiffrierschlüssel
symmetrisches Kryptosystem: Konzelation und Integrität		**k** geheimer Schlüssel	
Signatursystem: Authentikation (beinhaltet Integrität)	**s** Signierschlüssel		**t** Testschlüssel

Bild 9: Übersichtsmatrix der Verschlüsselungsziele und zugehörigen Schlüsselverteilungen von asymmetrischem Konzelationssystem, symmetrischem Kryptosystem und Signatursystem

In den folgenden Unterabschnitten dieses Abschnitts werden weitere allgemein relevante Eigenschaften von Kryptosystemen betrachtet: Betriebsarten (Block-, Stromchiffre; deterministisches, indeterministisches Kryptosystem), Sicherheit (informations-, komplexitätstheoretisch), Aufwand ihrer Realisierung bzw. Verschlüsselungsleistung und Standardisierung (geheime oder öffentlich validierte Kryptosysteme).

2.2.2.1 Betriebsarten: Blockchiffre, Stromchiffre

Bisher wurden Ver- und Entschlüsselung immer direkt auf mit einer umgangssprachlichen Semantik und Syntax versehene Einheiten angewendet: auf „Klartext“ und „Schlüsseltext“ in Bild 3 und Bild 5, auf „Text“ in Bild 7 sowie auf „Nachrichten“ in Bild 4, Bild 6 und Bild 8.

Für die Definition und Implementierung der Kryptosysteme, genauer der sie darstellenden Familien von Chiffrierfunktionen, ist es jedoch nötig, sie nicht auf semantisch oder syntaktisch mehr oder minder genau definierten Einheiten, sondern auf den Elementen eines zur Codierung dieser Einheiten geeigneten, präzis definierten *Klartext-* bzw. *Schlüsseltextraumes* operieren zu lassen. Sind deren Elemente jeweils *Blöcke*, das sind *Zeichenketten fester Länge über*

einem endlichen Alphabet, und wird eine Nachricht, die länger als ein Block ist, in eine Folge von Blöcken zerlegt und jeder Block unabhängig, d. h. insbesondere mit demselben Schlüssel, ver- und später entschlüsselt, so heißt das Kryptosystem **Blockchiffre** (im Deutschen auch Blocksystem [HeKW_85], im Englischen block cipher genannt). Falls bei der Zerlegung der Nachricht in Blöcke gleiche Blöcke entstehen, werden diese gleichen Klartextblöcken von einer *deterministisch* verschlüsselnden Blockchiffre auf gleiche Schlüsseltextblöcke abgebildet. Dies gilt natürlich nicht nur für Blöcke einer Nachricht, sondern, solange der Schlüssel nicht verändert wird, auch für gleiche Blöcke verschiedener Nachrichten. Da die heute üblichen Kryptosysteme deterministisch sind und eine Häufigkeitsanalyse einem Angreifer auch bei großer und ein vollständiges Brechen hoffentlich vereitelnder Blocklänge bereits wertvolle Information liefern kann, ist bei deterministischen Blockchiffren also Vorsicht angebracht. Ein ähnliches Problem wurde im Kontext der asymmetrischen Konzelationssysteme in Abschnitt 2.2.1.2.1 geschildert und gelöst. Seine Lösung ist auch auf das Problem der Blockchiffre anwendbar.

Jede deterministische Blockchiffre mit genügend großen Blöcken kann zu einer **indeterministischen Blockchiffre** modifiziert werden, die die oben erwähnte Häufigkeitsanalyse vereitelt: die Nachricht wird in wesentlich kleinere Blöcke zerlegt als die deterministische Blockchiffre verschlüsselt und diese werden zusammen mit einer jeweils eigens generierten *echten* (im Gegensatz zu den später noch zu besprechenden *pseudozufälligen*) Zufallszahl (von etwa hundert Bit Länge) verschlüsselt. Nach dem weiterhin *deterministischen* Entschlüsseln werden die Zufallszahlen wieder von den kleineren Nachrichtenblöcken getrennt (und die Nachricht wie üblich zusammengesetzt). Der große Nachteil dieses Verfahrens ist, daß man den Übertragungskanal und die Verschlüsselungsfähigkeit des deterministischen Kryptosystems schlecht ausnutzt:

- Ist die Blocklänge sehr groß, so entsteht, da jede Nachricht auf ein Vielfaches der sehr großen Blocklänge aufgefüllt werden muß, wegen der sehr großen Blocklänge durch jeden „letzten“ Block im Schnitt viel zusätzlicher Übertragungs- und Verschlüsselungsaufwand. Dieser Aufwand ist, da bei einer sehr großen Blocklänge Nachrichten im Mittel nur wenige Blöcke umfassen, nicht zu vernachlässigen.
- Ist die Blocklänge mittel, so entsteht durch den in jedem Block nicht vernachlässigbaren Anteil der Zufallszahl bei jedem Block nichtvernachlässigbar viel zusätzlicher Übertragungs- und Verschlüsselungsaufwand.

Die im folgenden zu definierende **Stromchiffre** operiert auf *Zeichenketten variabler Länge über einem endlichen Alphabet* und erreicht denselben Zweck ohne diese Nachteile.

Bei einer Stromchiffre (im Deutschen auch als kontinuierliche Chiffre [Hors_85] oder Stromsystem [HeKW_85], im Englischen als stream cipher bezeichnet) werden Nachrichten als eine Folge von *Zeichen* codiert, so daß einzelne Zeichen des Alphabets verschlüsselt werden. Diese Zeichen werden jedoch nicht unabhängig voneinander verschlüsselt, sondern ihre Verschlüsselung hängt entweder auch

1. von ihrer Position innerhalb der Nachrichten oder allgemeiner von allen vorhergehenden Klartext- und/oder Schlüsseltextzeichen ab oder
2. nur von einer beschränkten Anzahl direkt vorhergehender Schlüsseltextzeichen.

Im ersten Fall spricht man von *synchronen Stromchiffren* (synchronous stream ciphers), da Ver- und Entschlüsselung streng synchron erfolgen muß: bei Verlust oder Hinzufügen eines Schlüsseltextzeichens, d. h. bei Verlust der Synchronisation, kann nicht mehr ohne weiteres entschlüsselt werden, Ver- und Entschlüsseler müssen sich neu synchronisieren. Sofern Ver- und Entschlüsselung nicht nur von der Position innerhalb der Nachrichten abhängt, sondern auch von allen vorhergehenden Klartext und/oder Schlüsseltextzeichen, müssen Ver- und Entschlüsseler sich auch bei Verfälschung eines Schlüsseltextzeichens neu synchronisieren.

Im zweiten Fall spricht man von *selbstsynchronisierenden Stromchiffren* (self-synchronous stream ciphers), da sich bei ihnen der Entschlüsseler auch bei Verlust oder Hinzufügen beliebig vieler zusätzlicher Schlüsseltextzeichen spätestens nach Entschlüsselung der oben erwähnten beschränkten Anzahl Schlüsseltextzeichen wieder auf den Verschlüsseler synchronisiert hat.

Für jede symmetrische bzw. asymmetrische deterministische Stromchiffre und beliebige nichtleere Texte x_1, x_2 und Schlüsselpaare (c,d) bzw. Schlüssel k gilt demnach:

Werden zwei Texte separat, aber direkt hintereinander verschlüsselt, so ist das Gesamtergebnis das gleiche, wie wenn die zwei Texte erst konkateniert und dann verschlüsselt werden. In der eingeführten Notation und bei kollateraler Auswertung beider Seiten der Gleichungen jeweils von links lautet dies:

$k(x_1),k(x_2) = k(x_1,x_2)$ bzw. $c(x_1),c(x_2) = c(x_1,x_2)$.

Werden zwischen den separat verschlüsselten Texten noch weitere verschlüsselt oder wird nach den beiden separaten Verschlüsselungen die der konkatenierten Texte als dritte ausgeführt, so gelten die obigen Gleichungen mit sehr großer Wahrscheinlichkeit nicht.

Um Zusammenhänge zwischen Block- und Stromchiffren aufzuzeigen, werden im folgenden die in der Literatur beschriebenen und teilweise genormten wichtigen **Konstruktionen von Stromchiffren aus Blockchiffren** angegeben. Auch wenn heute allgemein davon ausgegangen wird, daß in allen Konstruktionen die Verwendung einer sicheren Blockchiffre üblicherweise den Erhalt einer sicheren Stromchiffre impliziert, erscheinen mir zwei einschränkende Bemerkungen notwendig:

1. Selbst unter einer sehr starken Definition, was eine sichere Blockchiffre ist, sind mir weder Beweise (im Sinne des folgenden Abschnitts 2.2.2.2) für die Sicherheit der erhaltenen Stromchiffren noch eine Quantifizierung des „üblicherweise“ bekannt.
2. Zumindest für manche dieser Konstruktionen konnten pathologische Gegenbeispiele konstruiert werden. Ich überlasse es dem Leser zu entscheiden, ob die konstruierte Blockchiffre von ihm als „sicher“ betrachtet wird.

Aus jeder symmetrischen oder asymmetrischen deterministischen Blockchiffre kann mittels der Konstruktionen

- *Blockchiffre mit Blockverkettung* (cipher block chaining, abgekürzt CBC [DaPr_84]) oder
- *Schlüsseltextrückführung* (cipher feedback, abgekürzt CFB [DaPr_84])

eine *selbstsynchronisierende* Stromchiffre gewonnen werden.

Blockchiffre mit Blockverkettung ist in Bild 10 gezeigt: Vor dem Verschlüsseln jedes (außer des ersten) Blockes wird zu seinem Klartext der Schlüsseltext des vorherigen modular

addiert und entsprechend nach dem Entschlüsseln jedes Blockes der Schlüsseltext des vorherigen von seinem „Klartext“ modular subtrahiert.

Diese Konstruktion hat folgende Vor- und Nachteile bzw. ambivalente Eigenschaften:

+ Die Verwendung einer indeterministischen Blockchiffre ist möglich.
+ Wird eine asymmetrische Blockchiffre verwendet, so ist die entstehende Stromchiffre ebenfalls asymmetrisch.
– Die Länge der verschlüsselbaren Einheiten ist durch die Blocklänge der verwendeten Blockchiffre bestimmt und kann deshalb nicht einfach auf die Einheiten des Übertragungs- oder Speichersystems abgestimmt werden. Deshalb müssen die Blockgrenzen für die Selbstsynchronisation ggf. gesondert kenntlich gemacht werden.
* Bei einer noch so kleinen Verfälschung einer einem Block entsprechenden Einheit des Schlüsseltextstromes sind alle Zeichen des Klartextes dieser Einheit mit der Wahrscheinlichkeit

  (Anzahl der Zeichen - 1) / Anzahl der Zeichen

  gestört. Zusätzlich ist die der Verfälschung entsprechende Stelle im nächsten Klartextblock gestört.

Alle Linien führen der Blocklänge entsprechend viele Alphabetzeichen
⊕ Addition bezüglich passend gewähltem Modulus
⊖ Subtraktion bezüglich passend gewähltem Modulus

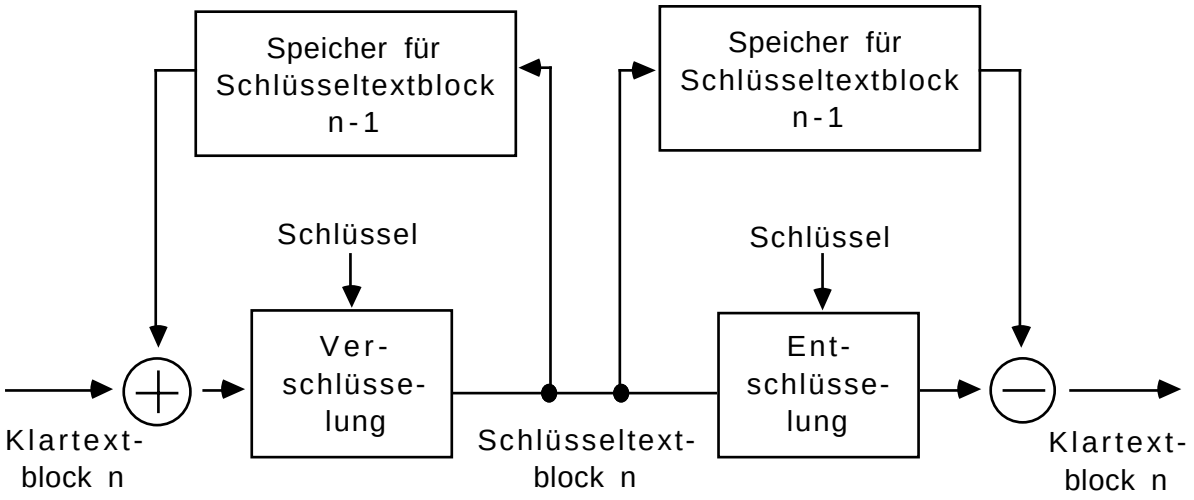


Bild 10: Konstruktion einer symmetrischen bzw. asymmetrischen selbstsynchronisierenden Stromchiffre aus einer symmetrischen bzw. asymmetrischen Blockchiffre: Blockchiffre mit Blockverkettung

Pathologisches Gegenbeispiel zu *Blockchiffre mit Blockverkettung*: Addition und Subtraktion erfolge modulo 2. Die Blockchiffre verschlüssele gerade Blöcke, d. h. letztes Bit 0, sicher auf ungerade Blöcke, d. h. letztes Bit 1, und ungerade Blöcke unsicher auf gerade Blöcke. Bei

Beschränkung des Klartextraumes auf gerade Blöcke, d. h. Betrachtung der Blockchiffre als um ein Bit expandierende Blockchiffre, handelt es sich also um eine sichere Blockchiffre.

Die resultierende Stromchiffre ist aber auch für den beschränkten Klartextraum unsicher: Da für das letzte Bit des Blockes gilt $0 \oplus 1 = 1$, Verschlüsselung ergibt 0, $0 \oplus 0 = 0$, Verschlüsselung ergibt 1, usw., ist jeder zweite Eingabeblock in die Blockchiffre ungerade. Der Angreifer kann diese Eingabeblöcke aus den geraden Schlüsseltextblöcken errechnen und durch Subtraktion der vorher beobachteten Schlüsseltextblöcke die zugehörigen Klartextblöcke errechnen. Die Stromchiffre ist also bezüglich jedes zweiten Klartextblockes und damit insgesamt unsicher.

Schlüsseltextrückführung ist in Bild 11 gezeigt: Es wird nicht der Klartext, sondern der Inhalt eines Schieberegisters mit der Blockchiffre verschlüsselt und ein Teil des Ergebnisses vom Verschlüsseler zum Klartext modular addiert (wodurch der Schlüsseltext entsteht) und vom Entschlüsseler vom Schlüsseltext modular subtrahiert wird (wodurch wiederum der Klartext entsteht). In die beim Ver- und Entschlüsseler jeweils dieselben Werte enthaltenden Schieberegister wird jeweils der Schlüsseltext geschoben, also rückgeführt, weshalb diese Konstruktion *Schlüsseltextrückführung* genannt wird.

In Bild 11 ist der allgemeine Fall gezeigt, daß der Schlüsseltext nicht direkt in das Schieberegister übernommen, sondern einer Auswahl unterworfen oder gar um feste Werte ergänzt wird. Letzteres ist zwar in der Norm [DINISO8372_87] vorgesehen, erscheint mir aber bezüglich der kryptographischen Sicherheit nicht sinnvoll, da dadurch die Zahl möglicher Werte im Schieberegister verkleinert wird. Da die Funktion „Wähle aus oder ergänze“ öffentlich festgelegt sein dürfte, kann ein Angreifer gleiche Werte in den Schieberegistern erkennen und durch Differenzbildung der Schlüsseltexte die Differenz zweier Klartexte erhalten.

Schlüsseltextrückführung hat gegenüber Blockchiffre mit Blockverkettung folgende Vor- und Nachteile bzw. ambivalente Eigenschaften:

+ Es können kleinere Einheiten als die durch die Blocklänge der verwendeten Blockchiffre bestimmten ver- und entschlüsselt werden. Wird die elementare Einheit des Übertragungs- oder Speichersystems als Verschlüsselungseinheit verwendet, so ist Selbstsynchronisation immer, insbesondere auch ohne Kenntlichmachen der „Blockgrenzen“, gegeben.
– Die verwendete Blockchiffre muß deterministisch sein.
– Unabhängig davon, ob eine symmetrische oder asymmetrische Blockchiffre verwendet wird, entsteht eine symmetrische Stromchiffre, da die bei einer asymmetrischen Blockchiffre von der Verschlüsselungsfunktion verschiedene Entschlüsselungsfunktion bei der Konstruktion überhaupt nicht verwendet wird.
* Bei einer noch so kleinen Verfälschung einer Einheit des Schlüsseltextstromes sind alle Zeichen des Klartextes dieser und der folgenden

  $$\lceil \text{Blocklänge / Länge der Rückkopplungseinheit} \rceil$$

  Rückkopplungseinheiten gestört ($\lceil x \rceil$ bezeichnet die kleinste ganze Zahl z mit $z \geq x$). Die erste Einheit ist genau an der Störungsstelle gestört, bei letzteren sind alle Zeichen mit der Wahrscheinlichkeit

  $$\text{(Anzahl der Zeichen - 1) / Anzahl der Zeichen}$$

  gestört.

b Blocklänge
a Länge der Ausgabeeinheit, a ≤ b
r Länge der Rückkopplungseinheit, r ≤ b
⊕ Addition bezüglich passend gewähltem Modulus
⊖ Subtraktion bezüglich passend gewähltem Modulus

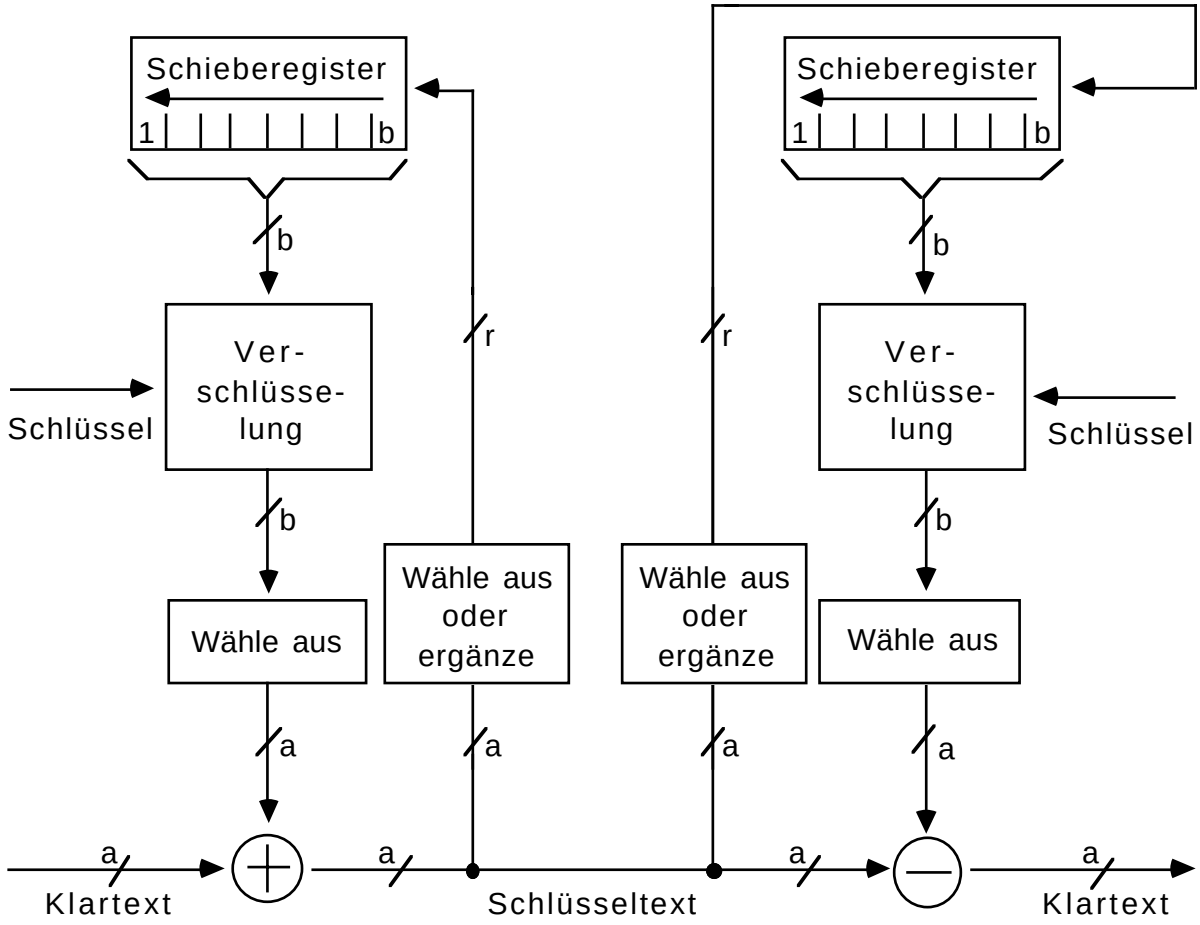


Bild 11: Konstruktion einer symmetrischen selbstsynchronisierenden Stromchiffre aus einer deterministischen Blockchiffre: Schlüsseltextrückführung

In den Bildern 10 und 11 sowie den folgenden Bildern 12, 13 und 14 sowie den zugehörigen Verfahrensbeschreibungen ist jeweils nur der übliche Fall dargestellt, daß die Blockchiffre Klartextblöcke in Schlüsseltextblöcke gleicher Länge abbildet. Ist dies nicht der Fall, d. h. sind die *Schlüsseltextblöcke länger*, so muß in Bild 10 jeweils – geschickterweise vor dem „Speicher für Schlüsseltextblock *n*-1“ – eine Auswahl getroffen werden. Entsprechendes gilt für Bild 13. In den Bildern 11, 12 und 14 ist jeweils nur die vorhandene Auswahl zu modifizieren.

Ebenfalls nicht eingegangen wird darauf, daß der „Speicher für Schlüsseltextblock *n*-1“ in den Bildern 10 bzw. 13 geeignet *initialisiert* werden sollte bzw. muß und daß das Schiebe-

register in den Bildern 11 bzw. 12 und 14 geeignet initialisiert werden sollte bzw. muß [MeMa_82, DaPr_84].

Aus jeder symmetrischen oder asymmetrischen deterministischen Blockchiffre kann mittels der Konstruktionen

- *Ergebnisrückführung* (output feedback, abgekürzt OFB [DaPr_84]) oder
- *Blockchiffre mit Blockverkettung über Schlüssel- und Klartext* (plaintext-ciphertext feedback, [EMMT_78 Seite 110, 111, MeMa_82 Seite 69])

eine *synchrone* Stromchiffre gewonnen werden.

Ergebnisrückführung ist in Bild 12 gezeigt: Im Gegensatz zur Schlüsseltextrückführung wird nicht der Schlüsseltext, sondern das Ergebnis (output) der Blockverschlüsselung in das Schieberegister rückgeführt. Entsprechend heißt diese Konstruktion *Ergebnisrückführung* (output feedback).

Wie in Bild 11 ist auch in Bild 12 der allgemeine Fall gezeigt, daß das Ergebnis der Blockverschlüsselung erst einer Auswahl unterworfen oder, kombiniert damit, um feste Werte ergänzt wird. Letzteres ist zwar möglich, erscheint aber bezüglich der kryptographischen Sicherheit nicht sinnvoll, da dadurch die Zahl möglicher Werte im Schieberegister und damit die Periode des Pseudozufallszahlengenerators verkleinert wird. Ist die Funktion „Wähle aus oder ergänze“ wie in [DaPa_83] empfohlen und in [DINISO8372_87] vorgesehen die Identität, so kann statt einem Schieberegister ein normaler Speicher verwendet werden.

Diese Konstruktion hat folgende Vor- und Nachteile bzw. ambivalente Eigenschaften:

+ Die Länge der ver- und entschlüsselbaren Einheiten ist nicht durch die Blocklänge der verwendeten Blockchiffre bestimmt und kann deshalb einfach auf die Einheiten des Übertragungs- oder Speichersystems abgestimmt werden.
– Die verwendete Blockchiffre muß deterministisch sein.
– Unabhängig davon, ob eine symmetrische oder asymmetrische Blockchiffre verwendet wird, entsteht eine symmetrische Stromchiffre, da die bei einer asymmetrischen Blockchiffre von der Verschlüsselungsfunktion verschiedene Entschlüsselungsfunktion bei der Konstruktion überhaupt nicht verwendet wird.
* Bei Verfälschung von Zeichen des Schlüsseltextstromes ist immer nur das entsprechende Zeichen des Klartextes gestört, es findet also keine *Fehlererweiterung* (error extension [DaPr_84]) statt. Je nach Anwendung kann dies günstig, z. B. bezüglich Konzelation, oder ungünstig, z. B. bezüglich Integrität, sein. Um einem falschen Eindruck vorzubeugen, sei an dieser Stelle noch an eine generelle Eigenschaft von synchronen Stromchiffren (und damit auch von Ergebnisrückführung) erinnert: Nur bei Verfälschung von Zeichen des Schlüsseltextstromes findet keine Fehlererweiterung statt. Bei verlorenen oder hinzugefügten Schlüsseltextstromzeichen sind bis zur Wiederherstellung der Synchronisation alle folgenden Klartextzeichen mit der Wahrscheinlichkeit

  (Anzahl der Zeichen - 1) / Anzahl der Zeichen

  gestört.

b Blocklänge
a Länge der Ausgabeeinheit, a ≤ b
r Länge der Rückkopplungseinheit, r ≤ b
⊕ Addition bezüglich passend gewähltem Modulus
⊖ Subtraktion bezüglich passend gewähltem Modulus

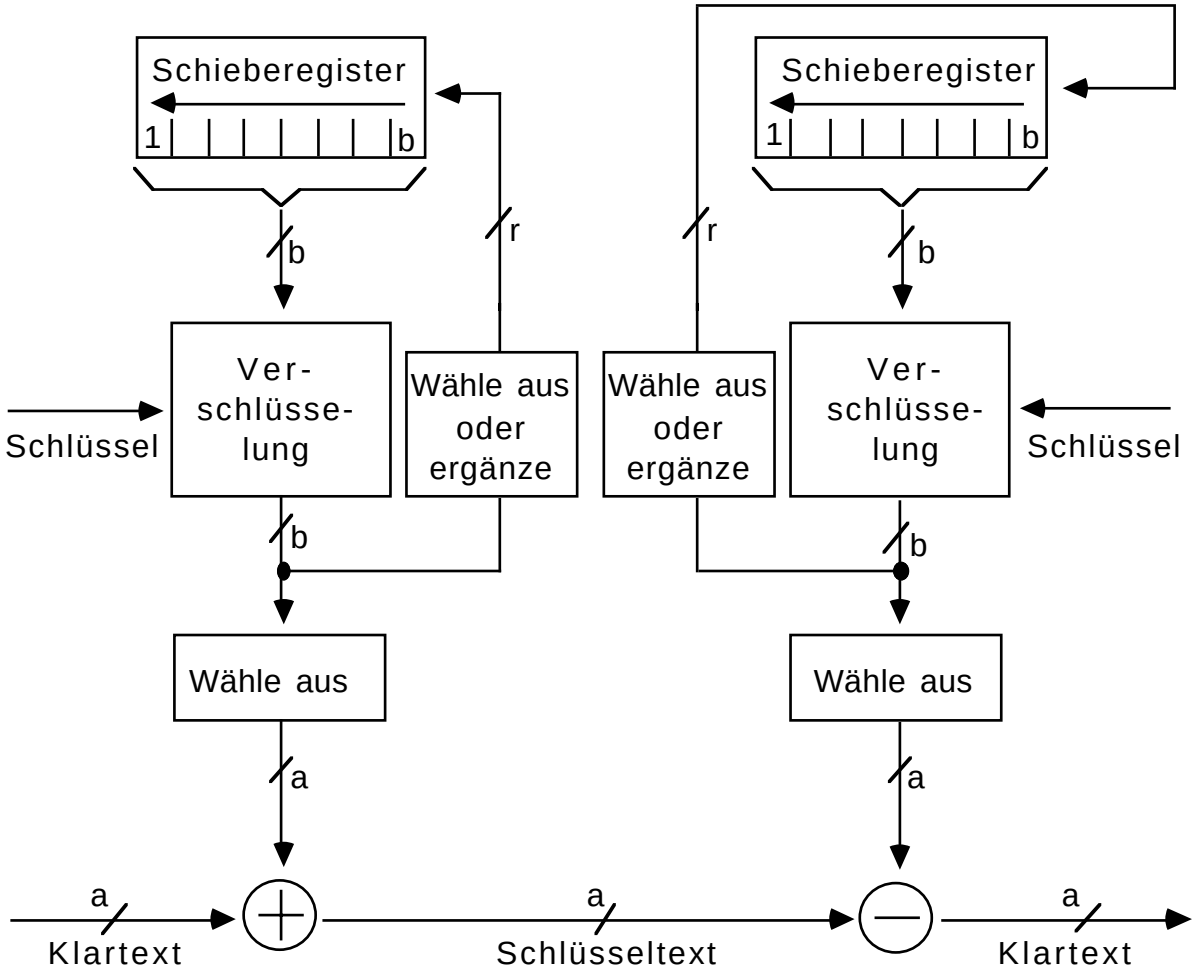


Bild 12: Konstruktion einer symmetrischen synchronen Stromchiffre aus einer deterministischen Blockchiffre: Ergebnisrückführung

Pathologisches Gegenbeispiel zu *Ergebnisrückführung*: Es finde keine Auswahl oder Ergänzung statt. Die Blockchiffre bilde Klartextblöcke in Schlüsseltextblöcke gleicher Länge ab und werde wie folgt definiert: Wähle einen Klartextblock *K*, dem noch kein Schlüsseltextblock zugeordnet ist, aus und ordne ihm als Verschlüsselung zufällig einen Schlüsseltextblock *S* zu, der bisher noch keinem Klartextblock zugeordnet wurde. Ordne *S* als Verschlüsselung *K* zu. Wiederhole beide Schritte, bis allen Klartextblöcken Schlüsseltextblöcke und allen Schlüsseltextblöcken Klartextblöcke zugeordnet sind. Diese Konstruktion erzeugt eine sichere Blockchiffre, da die Zuordnung Klartextblöcke – Schlüsseltextblöcke „zufällig“ ist.

Die resultierende Stromchiffre ist aber unsicher, da zum Klartext abwechselnd immer dasselbe addiert wird.

Erfordert eine Anwendung, daß anders als bei Ergebnisrückführung auch ab einem verfälschten Schlüsseltextstromzeichen alle folgenden Klartextzeichen mit der oben angegebenen Wahrscheinlichkeit gestört sind, so kann man bei der Blockchiffre mit Blockverkettung (Bild 10) zusätzlich zum Schlüsseltext des vorherigen Blockes auch über dessen Klartext verketten (Bild 13) und erhält so eine **Blockchiffre mit Blockverkettung über Schlüssel- und Klartext**. Es ist bemerkenswert, daß zur Entschlüsselung eine Invertierung der Funktion *h*, die den vorherigen Schlüssel- und Klartextblock verknüpft, nicht nötig ist. Damit diese Konstruktion aus einer sicheren symmetrischen oder asymmetrischen Blockchiffre eine sichere symmetrische oder asymmetrische *synchrone* Stromchiffre gewinnt, muß die Funktion *h* in Abhängigkeit vom für die Addition bzw. Subtraktion gewählten Modulus geeignet gewählt werden, vgl. [MeMa_82].

Diese Konstruktion hat folgende Vor- und Nachteile bzw. ambivalente Eigenschaften:

+ Die Verwendung einer indeterministischen Blockchiffre ist möglich.
+ Wird eine asymmetrische Blockchiffre verwendet, so ist die entstehende Stromchiffre ebenfalls asymmetrisch.
– Die Länge der verschlüsselbaren Einheiten ist durch die Blocklänge der verwendeten Blockchiffre bestimmt und kann deshalb nicht einfach auf die Einheiten des Übertragungs- oder Speichersystems abgestimmt werden.
* Bei einer noch so kleinen Verfälschung einer einem Block entsprechenden Einheit des Schlüsseltextstromes sind ab diesem Block einschließlich alle Zeichen mit der Wahrscheinlichkeit

  (Anzahl der Zeichen - 1) / Anzahl der Zeichen

  gestört.

Alle Linien führen der Blocklänge entsprechend viele Alphabetzeichen
⊕ Addition bezüglich passend gewähltem Modulus, z. B. 2
⊖ Subtraktion bezüglich passend gewähltem Modulus, z. B. 2
h beliebige Funktion, z. B. Addition mod $2^{\text{Blocklänge}}$

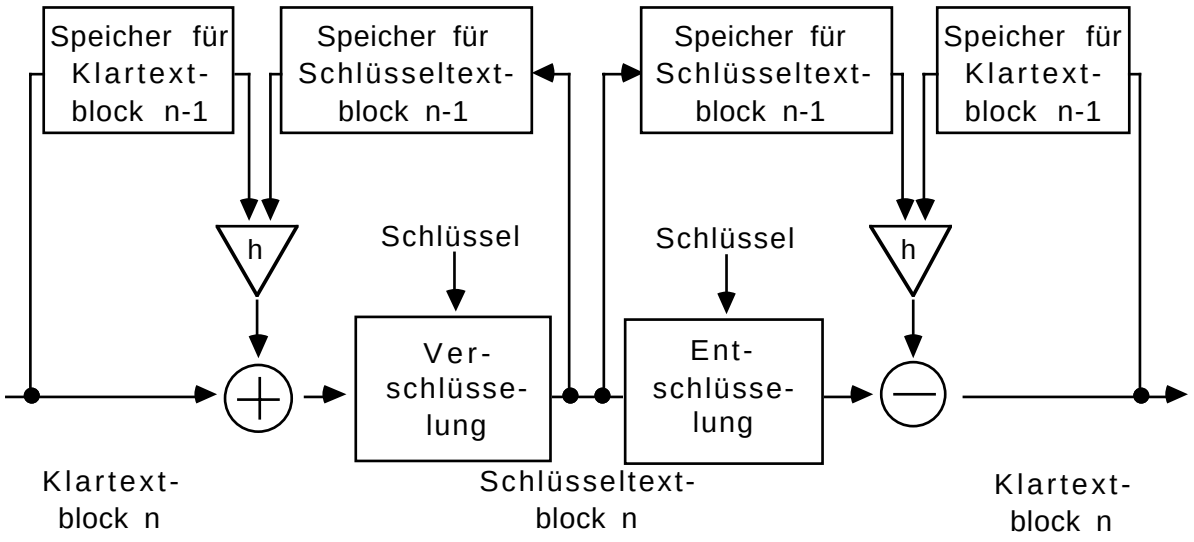


Bild 13: Konstruktion einer symmetrischen bzw. asymmetrischen synchronen Stromchiffre aus einer symmetrischen bzw. asymmetrischen Blockchiffre: Blockchiffre mit Blockverkettung über Schlüssel- und Klartext

Vergleicht man „Blockchiffre mit Blockverkettung“ mit „Blockchiffre mit Blockverkettung über Schlüssel- und Klartext“ so fällt zweierlei auf:

- Die Konstruktionen sind sehr ähnlich – beide verwenden Ver- *und* Entschlüsselung der Blockchiffre und operieren invertierbar auf dem „Klartext“. Genauer gesagt umfaßt die letztere Konstruktion die erstere (man wähle die Funktion *h* so, daß sie den eingegebenen Schlüsseltextblock ausgibt und den Klartextblock ignoriert).
- Aus dieser Ähnlichkeit resultieren gleiche Vor- und Nachteile. Die ambivalente Eigenschaft der Fehlererweiterung unterscheidet sich darin, ob sie begrenzt (selbstsynchronisierende Stromchiffre) oder potentiell unbegrenzt ist (synchrone Stromchiffre). Die Fehlererweiterung kann dadurch potentiell unbegrenzt sein, daß die Funktion *h* den vorherigen Klartextblock verwenden kann, der wiederum von allen vorherigen Schlüsseltextblöcken abhängen kann.

Entsprechendes gilt für „Schlüsseltextrückführung“ und „Ergebnisrückführung“:

- Die Konstruktionen sind sehr ähnlich – beide verwenden nur die Verschlüsselung der Blockchiffre zur Erzeugung eines pseudozufälligen Zeichenstromes mit dem durch modulare Addition verschlüsselt und durch modulare Subtraktion entschlüsselt wird. In Bild 14 ist eine allgemeine Konstruktion angegeben, die durch geeignete Wahl der Funktion *h* „Schlüsseltextrückführung“ und „Ergebnisrückführung“ umfaßt. Wie in Bild 13 ist für die Entschlüsselung eine Invertierung der Funktion *h* nicht nötig.

- Aus dieser Ähnlichkeit resultieren gleiche Vor- und Nachteile. Die ambivalente Eigenschaft der Fehlererweiterung unterscheidet sich darin, ob sie begrenzt (selbstsynchronisierende Stromchiffre) oder potentiell unbegrenzt ist (synchrone Stromchiffre). Die Fehlererweiterung kann dadurch potentiell unbegrenzt sein, daß die Funktion *h* das Ergebnis der Blockverschlüsselung verwenden kann, das wiederum vom gesamten vorherigen Schlüsseltextstrom abhängen kann.

b Blocklänge
a Länge der Ausgabeeinheit, a ≤ b
r Länge der Rückkopplungseinheit, r ≤ b
⊕ Addition bezüglich passend gewähltem Modulus
⊖ Subtraktion bezüglich passend gewähltem Modulus
h beliebige Funktion

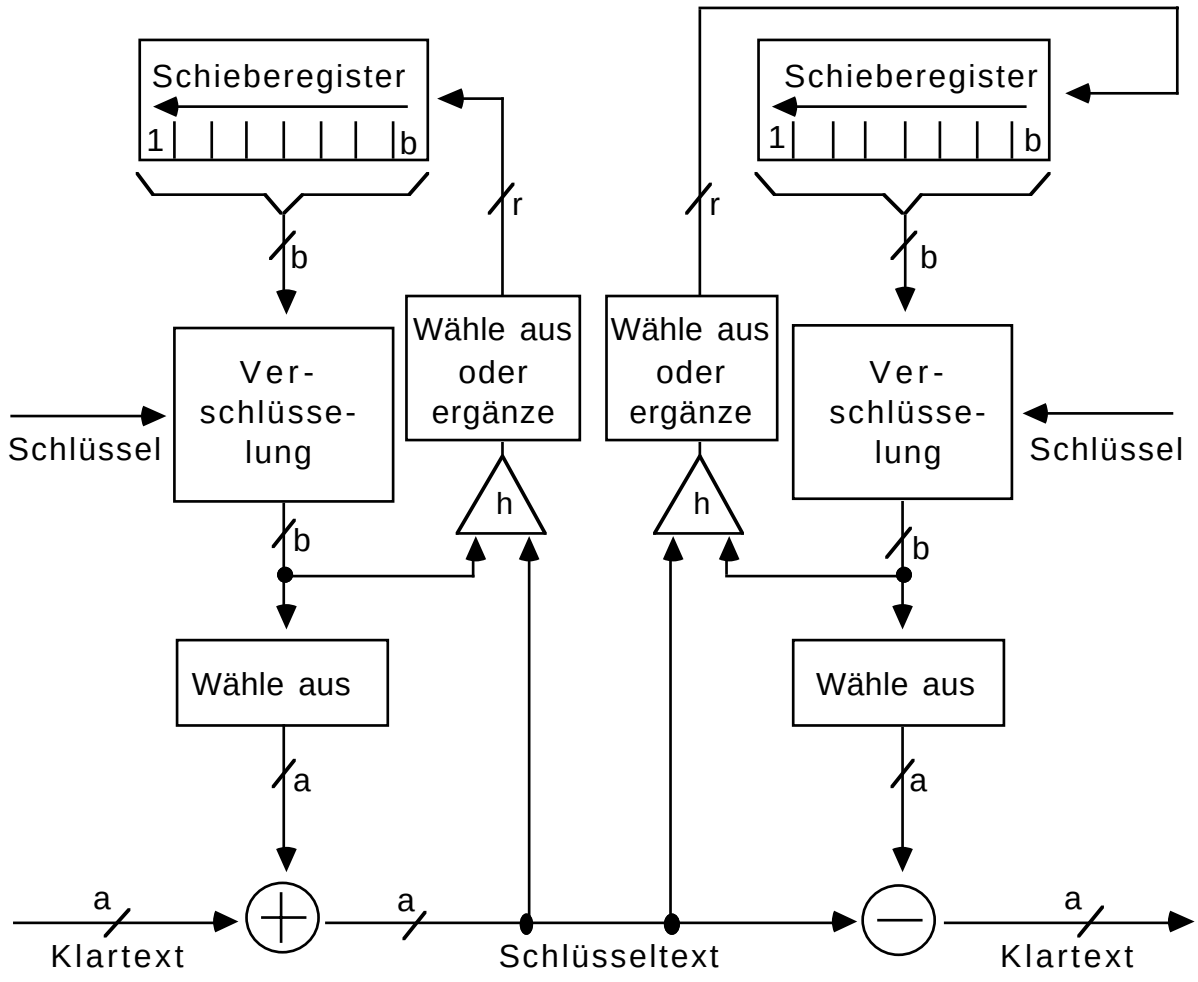


Bild 14: Konstruktion einer symmetrischen Stromchiffre aus einer deterministischen Blockchiffre: Schlüsseltext- und Ergebnisrückführung

2.2.2.2 Sicherheit: informationstheoretisch, komplexitätstheoretisch

Zu Beginn von Abschnitt 2.2.1 wurde eine sehr vorsichtige Formulierung gebraucht: „Verschlüsselung ... soll garantieren, daß der Inhalt einer gesendeten Nachricht nur den Besitzern eines bestimmten Schlüssels zugänglich ist bzw. ohne Kenntnis des Schlüssels nicht unerkennbar verändert werden kann.“ Hier wird der Frage nachgegangen, ob, wie und wieweit heute diese Eigenschaften *bewiesen*, *beweisbar* oder zumindest *validierbar* sind.

Die Sicherheit eines Kryptosystems kann in zwei Modellwelten untersucht und ggf. bewiesen werden.

Die eine, **informationstheoretische Modellwelt** wurde von Claude Shannon in seinen fundamentalen Arbeiten [Shan_48, Shan_49, Sha1_49] weitgehend vollständig geschaffen, Erweiterungen sind in [Hell_77, BeB2_88] zu finden. In ihr wird dem Angreifer unbegrenzte Rechenkapazität zugebilligt, so daß er alle durch seine Beobachtungen theoretisch gewonnene Information auch praktisch zur Verfügung hat. Durch Abstraktion von der praktischen Durchführbarkeit von Berechnungen ist die informationstheoretische Modellwelt vergleichsweise einfach, elegant und leicht anwendbar. Auch hat sie die Eigenschaft, daß in ihr gewonnene Aussagen immer auf der sicheren Seite liegen: ein Angreifer kann (bei genügend genauer Modellierung auch in der Realität) praktisch höchstens das wissen, was er in ihr wissen kann. Neben diesen Vorteilen hat die informationstheoretische Modellwelt aber auch einen für praktische Zwecke gravierenden Nachteil: In ihr ist bei weitem nicht alles möglich und deshalb erst recht nicht beweisbar, was von den Anwendungen her erforderlich, geschweige denn wünschenswert ist. Da jeder Angreifer alle möglichen Dechiffrier- bzw. Signierschlüssel in der informationstheoretischen Modellwelt durchprobieren und wegen der Kenntnis des Chiffrier- bzw. Testschlüssels einen geeigneten auswählen könnte, gibt es in ihr prinzipiell keine asymmetrischen Kryptosysteme und damit prinzipiell keine Möglichkeit, Authentifikation auf kryptographischem Wege zu erreichen. Es bliebe nur die Möglichkeit, gegen ihre Besitzer sichere Geräte, die Schlüssel eines symmetrischen Kryptosystems gemeinsam haben, so zu bauen, daß mit einem bestimmten Schlüssel jeweils nur eines verschlüsselt, d. h. „unterschreibt“, während alle anderen mit diesem Schlüssel nur entschlüsseln, d. h. die „Unterschrift“ prüfen. Die in Abschnitt 2.1.2 gemachten Bemerkungen über die Schwierigkeit einer Sicherheitsvalidierung solcher auf Manipulationsversuche reagierenden und unerwünschte elektromagnetische oder mechanische Abstrahlung unter die Sensitivität professioneller Empfangsgeräte dämpfenden „gegen ihre Besitzer sicheren Geräten“ sowie ihren Preis gelten auch hier.

Die andere, **komplexitätstheoretische Modellwelt** ist im Aufbau befindlich und es ist ungewiß, ob und ggf. wann ihre Entwicklung soweit abgeschlossen sein wird, daß (fast) alle praktisch relevanten Fragen in ihr ohne unbewiesene Annahmen behandelt werden können. In der komplexitätstheoretischen Modellwelt wird dem Angreifer nur begrenzte Rechenkapazität zugebilligt, wobei diese Begrenzung je nach mehr theoretischer oder mehr praktischer Ausprägung des Modells als nur polynomial viel in der Länge eines Sicherheitsparameters, z. B. der Schlüssellänge, [Yao1_82, GoMi_84] oder als sogar absolut beschränkt, z. B. durch die physischen Ressourcen des Universums oder der Erde [DiHe_79 Seite 399, DaPr_84 Seite 42], angenommen wird.

Innerhalb der informationstheoretischen Modellwelt wurde bewiesen, daß Verschlüsselung mit speziellen symmetrischen Kryptosystemen sowohl Konzelation [Sha1_49] als auch Integrität [Simm_85, Simm_86, Simm_88, Stin_88] garantiert.

Garantierte Konzelation bedeutet, daß ein Angreifer, der den Klartext zumindest nicht vollständig kennt, aber den zugehörigen Schlüsseltext vollständig erhält sowie den verwendeten geheimen Schlüssel nicht kennt, den zugehörigen Klartext zumindest nicht eindeutig bestimmen kann (*informationstheoretische Konzelation*, im Englischen als ideal secrecy [Sha1_49 Seite 699] bezeichnet) oder aber durch das Erfahren des Schlüsseltextes überhaupt keine Information über den Klartext erhält (*perfekte informationstheoretische Konzelation*, im Englischen als perfect secrecy [Sha1_49 Seite 679] bezeichnet). Ersteres bedeutet, daß auch nach der mit unbegrenzter Rechenkapazität durchgeführten Kryptoanalyse aus der Sicht des Angreifers immer mehrere mögliche Klartexte übrigbleiben. Letzteres bedeutet, daß aus der Sicht des Angreifers für alle Klartexte die a-priori-Wahrscheinlichkeiten, daß sie gesendet werden, gleich den a-posteriori-Wahrscheinlichkeiten, daß sie gesendet wurden, sind. Sofern der Angreifer den Klartext vor Erhalt des Schlüsseltextes nicht vollständig kennt, schließt letzteres ersteres natürlich ein.

Garantierte Integrität bedeutet, daß ein Angreifer, der den Klartext und Schlüsseltext vollständig aber den zur Sicherung der Integrität verwendeten geheimen Schlüssel nicht kennt, zumindest keine mit Sicherheit funktionierende Möglichkeit und im günstigsten Fall keine bessere Möglichkeit als zufälliges Raten zum Erzeugen eines anderen zulässigen Schlüsseltextes (zu dem dann wiederum ein anderer, dem Angreifer möglicherweise unbekannter Klartext gehört) hat. Ersteres sei mit *informationtheoretischer Integrität*, letzteres mit *perfekter informationstheoretischer Integrität* bezeichnet. Im Gegensatz zur Konzelation hat der Angreifer bezüglich Integrität also immer eine Erfolgschance. Sie ist allerdings bei perfekter informationstheoretischer Integrität in der Länge eines Sicherheitsparameters, z. B. der Schlüssellänge, exponentiell klein.

Das Erreichen von informationstheoretischer Konzelation bzw. perfekter informationstheoretischer Konzelation erfordert, daß der rein zufällige Informationsgehalt des Schlüssels (in der Fachsprache: seine Entropie) echt größer als die Redundanz des Klartextes bzw. mindestens so groß wie seine gesamte Länge ist. Insbesondere letzteres erfordert sehr lange Schlüssel, die rein zufällig generiert und in Konzelation und Integrität garantierender Weise verteilt werden müssen, was einen sehr hohen Aufwand bedingt. Obwohl die eigentliche Ver- bzw. Entschlüsselung dann denkbar einfach, nämlich durch zeichenweise Addition von Klartext und Schlüssel bzw. zeichenweise Subtraktion des Schlüssels vom Schlüsseltext (beides modulo Alphabetgröße) erfolgen kann (bei binären Zeichen in beiden Fällen sogar dieselbe Operation, nämlich XOR), ist der Einsatz dieser nach ihrem Erfinder Gilbert Vernam benannten **Vernam-Chiffre** (im Englischen auch one time pad [Denn_82 Seite 86] und auf Neudeutsch One-Time-Tapes [HeKW_85 Seite 15] genannt) nur in äußerst sensitiven Bereichen möglich. Nach der Klassifikation von Abschnitt 2.2.2.1 ist die Vernam-Chiffre eine synchrone Stromchiffre, was neben dem Aufwand zum Schlüsselgenerieren und -austauschen auch noch Aufwand zum Erhalt bzw. der Wiederherstellung der Synchronisation verursacht, vgl. die Bemerkungen zur die Vernam-Chiffre nachbildenden Konstruktion Ergebnisrückführung in Abschnitt 2.2.2.1.

Das Erreichen von informationstheoretischer Integrität bzw. perfekter informationstheoretischer Integrität erfordert, daß der rein zufällige Informationsgehalt des Schlüssels (in der

Fachsprache: seine Entropie) proportional zum Logarithmus des Kehrwertes der zulässigen Erfolgswahrscheinlichkeit des Angreifers sowie (was schlimmer ist) proportional zur Anzahl der Blöcke, deren Integrität gewährleistet werden soll, ist.

Innerhalb der komplexitätstheoretischen Modellwelt konnten bisher lediglich Beweise der folgenden Struktur erbracht werden: Unter der Annahme, daß ein bestimmtes wohluntersuchtes Problem schwierig ist, wird bewiesen, daß ein Kryptosystem genauso schwierig zu brechen ist. Hierbei werden vorzugsweise in der Mathematik seit langem gründlich untersuchte, aber bisher nicht effizient lösbare „Probleme“, z. B. Faktorisieren von Zahlen mit großen Primfaktoren oder Ziehen diskreter Logarithmen, als Beweisannahme verwendet. „Schwierig“ bedeutet dann, daß kein indeterministischer Lösungsalgorithmus mit genügend kurzer (meist: in der Länge des Sicherheitsparameters, z. B. der Schlüssellänge, polynomialer) Laufzeit bekannt ist. „Brechen“ bedeutet dann, daß der Angreifer mit einem indeterministischem Algorithmus mit kurzer (meist: mit in der Länge des Sicherheitsparameters polynomialer) Laufzeit irgendwelche Information über Klartext oder Schlüssel gewinnen bzw. (mit größerer als exponentiell kleiner Erfolgschance) Schlüssel- oder Klartext unerkannt verändern kann. Es werden damit Analoga zu perfekter informationstheoretischer Konzelation bzw. Integrität gebildet, die entsprechend *perfekte komplexitätstheoretische Konzelation* bzw. *perfekte komplexitätstheoretische Integrität* genannt werden. „Beweisen“ bedeutet dann, daß jedes Brechen des Kryptosystems (meist: mit einem indeterministischen Algorithmus mit in der Länge des Sicherheitsparameters polynomialer Laufzeit) zugleich einen effizienten (meist: polynomialen) indeterministischen Lösungsalgorithmus für das gemäß Annahme schwierige (meist: indeterministisch nicht polynomial lösbare) Problem liefert. Hier merkt man, daß die meistens verwendete Definition von „schwierig = indeterministisch nicht polynomial“ vor allem beweistechnische Ursachen hat: wird ein Polynom in ein anderes eingesetzt, so entsteht wieder ein Polynom. Für praktische Zwecke wäre es jedoch vollkommen ausreichend, wenn bei gegebenem Sicherheitsparameter s und zum Ver- und Entschlüsseln zu leistendem Aufwand $a(s)$ ein Angreifer mindestens den Aufwand $(a(s))^x$ zu treiben hätte und x hierbei beispielsweise den Wert 5 besäße.

Es sei angemerkt, daß mit Hilfe von *kryptographisch starken Pseudozufallsbitfolgengeneratoren* (cryptographically strong pseudorandom bit generators [BlMi_84]) aus jedem perfekte informationstheoretische Konzelation bzw. Integrität garantierendem symmetrischen Kryptosystem ein perfekte komplexitätstheoretische Konzelation bzw. Integrität garantierendes hergeleitet werden kann, das statt der langen (da proportional zur Nachrichtenlänge bzw. -anzahl wachsenden) echt zufällig erzeugten Schlüssel nur kurze (da nur proportional zur Länge des Sicherheitsparameters) echt zufällige Schlüssel benötigt. Aus diesen echt zufälligen kurzen Schlüsseln, Startwert (seed) genannt, erzeugt ein Pseudozufallsbitfolgengenerator eine (in der Länge des Startwertes überpolynomial) lange Pseudozufallsbitfolge. Er heißt kryptographisch stark, wenn ein Angreifer, der den Pseudozufallsbitfolgengenerator und ein beliebiges (in der Länge des Startwertes) polynomial langes Stück der Pseudozufallsbitfolge, nicht aber den Startwert kennt und (in der Länge des Startwertes) polynomial viel rechnen darf, durch all seine Kenntnisse und Rechnerei keinen signifikanten Vorteil gegenüber blindem Raten erhält. Genauer heißt dies, daß er das nächste Bit der Pseudozufallsbitfolge mit keiner Wahrscheinlichkeit signifikant größer als 0,5 vorhersagen kann. Ganz genau heißt dies, daß die Wahrscheinlichkeit einer richtigen Vorhersage des Angreifers für alle Polynome P ab genügender Länge des Startwertes kleiner

als $0{,}5 + 1 / P$(Länge des Startwertes) ist. [BlMi_84] enthält neben der originären Definition kryptographisch starker Pseudozufallsbitfolgengeneratoren einen, dessen kryptographische Stärke als äquivalent zum Ziehen diskreter Logarithmen bewiesen wird. Die Stärke eines effizienteren wird in [VaVa_85] als äquivalent zur Faktorisierung bewiesen.

Neben diesen im obigen Sinne bewiesenen perfekten komplexitätstheoretischen Simulationen perfekter informationstheoretischer (symmetrischer) Kryptosysteme gibt es auch entsprechend bewiesene, in der informationstheoretischen Modellwelt prinzipiell nicht existierende Blockchiffren und asymmetrische Kryptosysteme, wenn die gerade gegebene, sehr starke Definition von „brechen“ soweit abgeschwächt wird, wie dies zwangsläufig nötig ist.

Eine deterministische Blockchiffre hat schon per definitionem die in Abschnitt 2.2.2.1 erwähnte Eigenschaft, gleiche Klartextblöcke solange auf gleiche Schlüsseltextblöcke abzubilden, solange der Schlüssel beibehalten wird. Sie kann also nur durch geeignete Verwendung (siehe Abschnitt 2.2.2.1 und auch am Ende dieses Abschnitts) jede Information vor einem Angreifer verbergen, d. h. perfekte Konzelation gewährleisten. Tut sie dies, so heiße sie *perfekte komplexitätstheoretische deterministische Blockchiffre*.

Ähnlich, wie auch bei perfekter informationstheoretischer Integrität nicht (in Analogie zu perfekter informationstheoretischer Konzelation) gefordert werden kann, daß der Angreifer keine Erfolgschance hat, so kann dies auch bei asymmetrischen Kryptosystemen nicht gefordert werden: bei einem asymmetrischen Konzelationssystem kann der Angreifer immer eine Nachricht raten (und bei indeterministischen Kryptosystemen immer auch den bei der Verschlüsselung zusätzlich verwendeten zufälligen Parameter, vgl. Abschnitt 2.2.2.1) und die Richtigkeit mit dem öffentlich bekannten Chiffrierschlüssel überprüfen. Bei beiden möglichen Ergebnissen der Überprüfung erhält er Information, bei richtigem Raten (üblicherweise) viel, bei falschem (meist erheblich) weniger. Das Raten einer Nachricht mag aber bei weitem leichter (wahrscheinlicher) sein als das eines zufälligen Parameters eines indeterministischen Kryptosystems. Ein asymmetrisches Konzelationssystem, bei dem der Angreifer bei beliebiger Wahrscheinlichkeitsverteilung der Nachrichten nur eine in der Länge eines zufälligen, die Verschlüsselung beeinflussenden Parameters höchstens exponentiell kleine Chance hat, die Nachricht und den zufälligen Parameter richtig zu raten, er also durch sein Probieren fast keine (genauer: exponentiell wenig) Information erhält, heiße *perfektes komplexitätstheoretisches asymmetrisches Konzelationssystem*.

Entsprechend der Situation bei einem asymmetrischen Konzelationssystem kann ein Angreifer bei einem Signatursystem die Unterschrift raten und die Richtigkeit mit dem öffentlich bekannten Schlüssel zum Testen überprüfen. Das Signatursystem heiße *perfektes komplexitätstheoretisches Signatursystem*, wenn es für den Angreifer keine bessere Strategie als zufälliges Durchprobieren aller möglichen Unterschriften gibt, er also, solange er nur polynomial viel probieren kann, fast keine (genauer: nur eine exponentiell kleine) Erfolgschance hat.

In [GoGM_84, LuRa_86, LuR1_86] werden perfekte komplexitätstheoretische symmetrische deterministische Blockchiffren, in [Will_85, Wil1_85] (nur gegen passive Angriffe sichere) asymmetrische Kryptosysteme, in [BlGo_85] (nur gegen passive Angriffe sichere) perfekte komplexitätstheoretische asymmetrische Konzelationssysteme und in [GoMR_84, GoMR_88] ein (sogar gegen den stärksten bei sinnvoller Systemgestaltung möglichen, nämlich einen Angriff mit vom Angreifer adaptiv gewählten zu unterschreibenden Nachrichten sicheres) perfektes komplexitätstheoretisches Signatursystem beschrieben und jeweils die Äquivalenz

ihrer Sicherheit im oben definierten Sinne zur Faktorisierung bewiesen (bzw. in den Kurzfassungen nur behauptet).

In [BlFM_88] ist ein selbst gegen adaptive aktive Angriffe (adaptive chosen ciphertext attack) beweisbar sicheres asymmetrisches Konzelationssystem angekündigt. Es soll auf einer nicht-interaktiven Zero-Knowledge Beweistechnik basieren. Da ich trotz Anfrage bisher von den Autoren keine genauere Information erhalten habe, mir dafür aber von anderer Seite gesagt wurde, daß es mit dieser nicht-interaktiven Zero-Knowledge Beweistechnik ernsthafte Schwierigkeiten gäbe, bleibt zur Zeit nur der Rückgriff auf *mehrschrittige* Verfahren zur Realisierung eines gegen adaptive aktive Angriffe beweisbar sicheren asymmetrischen Konzelationssystems [GoMT_82, Bött_89, PfPf_89]. Bei diesen mehrschrittigen Verfahren kann nicht anhand eines öffentlich bekannten Schlüssels des Empfängers eine Nachricht für ihn in einem Schritt verschlüsselt und ihm zugeschickt werden. Es muß zuerst zwischen Sender und Empfänger ein Dialog geführt werden, in dem zwischen Sender und Empfänger ein geheimer Schlüssel ausgetauscht wird.

Aus dieser Begriffsbildung folgt direkt:

Alle innerhalb der informationstheoretischen Modellwelt (perfekte) informationstheoretische Konzelation bzw. Integrität bewiesenermaßen garantierenden Kryptosysteme garantieren selbstverständlich auch innerhalb jeder entsprechenden komplexitätstheoretischen Modellwelt (perfekte) komplexitätstheoretische Konzelation bzw. Integrität.

Aus jedem innerhalb irgendeiner Modellwelt sicheren asymmetrischen Konzelationsystem kann natürlich ein in derselben Modellwelt sicheres symmetrisches gewonnen werden, indem die das Schlüsselpaar generierende Partei der anderen nicht nur den Chiffrierschlüssel, sondern auch den Dechiffrierschlüssel oder allgemeiner die die Schlüsselgenerierung parametrisierende Zufallszahl mitteilt.

Es sei noch darauf hingewiesen, daß es Kryptosysteme gibt, die in keiner der beiden Modellwelten als sicher bewiesen sind, aber dennoch von praktisch allen Experten für sicher gehalten werden. Bekannte Beispiele sind

- die in Abschnitt 2.2.1.1 bereits kurz erwähnte symmetrische deterministische Blockchiffre Data Encryption Standard (DES), der von der amerikanischen Normungsbehörde für den öffentlichen Bereich (NBS = National Bureau of Standards) 1977 genormt wurde [DES_77]. DES bildet Blöcke von 64 Bit auf Blöcke von ebenfalls 64 Bit ab, d. h. er führt eine *Permutation* durch. Wegen seiner kurzen Schlüssellänge von 56 Bits und der Geheimhaltung der Entwurfskriterien seiner Permutations- und Substitutionsboxen ist er sehr umstritten, er hat aber allen bekanntgegebenen Versuchen, ihn wesentlich effizienter als durch vollständiges Durchprobieren aller Schlüssel zu brechen, widerstanden [KaRS_86, KaR1_86, KaRS_88, ChEv_86]. Die Sicherheit von DES beruht nicht auf der Schwierigkeit klassischer mathematischer Probleme, sondern auf dem von Claude Shannon empfohlenen organisierten Chaos der iterativen Anwendung von einzeln relativ leicht umkehrbaren schlüsselabhängigen Elementartransformationen des zu verschlüsselnden Blocks.
  Es sei angemerkt, daß sich mit demselben Prinzip wie DES viele weitere symmetrische Blockchiffren, die ebenfalls eine Permutation durchführen und deren Sicherheit

mindestens so groß wie die von DES ist, bilden lassen (z. B. LUCIFER-artige, vgl. [FeNS_75, HeKW_85]), indem die Blocklänge (vgl. [Goel_86 Seite 25]) und vor allem die Schlüssellänge erhöht und die Permutations- und Substitutionsboxen variabel gehalten (etwa als Teil des Schlüssels), durch ein öffentlich durchgeführtes Zufallsexperiment bestimmt (nur bei vorher spezifizierten statistischen Anomalien muß das Zufallsexperiment wiederholt werden) oder aber zumindest ihre Entwurfskriterien offengelegt werden. Es gibt zwei Gründe, warum die Erhöhung der Schlüssellänge weit wichtiger als eine Erhöhung der Blocklänge ist: es gibt $(2^{\text{Blocklänge}})!$ mögliche Permutationen, aber „nur“ $2^{\text{Schlüssellänge}}$ mögliche Schlüssel; mit den am Ende von Abschnitt 2.2.2.1 beschriebenen Techniken kann eine kleine reale Blocklänge effizient in eine große virtuelle transformiert werden, während das Transformieren einer großen realen Blocklänge in eine kleine virtuelle mit erheblichen Effizienzeinbußen verbunden ist [Hell_82].
Im Anhang werden einige sinnvolle Verallgemeinerungen von DES diskutiert.

- das in Abschnitt 2.2.1.2 bereits erwähnte erste veröffentlichte asymmetrische deterministische Kryptosystem RSA. Es wird zwar allgemein vermutet, daß schlüsselbezogenes Brechen von RSA so schwer wie Faktorisierung ist. Da andererseits RSA aber schlüsselbezogen gebrochen ist, wenn Faktorisierung nicht schwierig ist, und auch ein nachrichtenbezogenes (oder auch nur wenig Information über einzelne Klartexte ergebendes, sogenanntes partielles [GoMi_84, BlGo_85]) Brechen möglich ist, ist RSA bezüglich passiver Angriffe unsicherer als alle in der komplexitätstheoretischen Modellwelt unter der Annahme, daß Faktorisierung schwierig ist, bewiesenen Kryptosysteme (vgl. Abschnitt 2.2.1.2). Da diese inzwischen bezüglich des Ver- und Entschlüsselungsaufwands genauso effizient bzw. aufwendig wie RSA und bezüglich des Übertragungsaufwands nur wenig schlechter sind, gibt es eigentlich keinen Grund, RSA weiterhin als normales asymmetrisches Kryptosystem dort zu verwenden, wo nur passive Angriffe möglich sind (genauer: wo auf aktive Angriffe keine Reaktion erfolgt). Es sei daran erinnert, daß die bisher bewiesenen asymmetrischen Konzelationssysteme bei aktiven Angriffen schlüsselbezogen gebrochen werden können. Speziellere Eigenschaften von RSA, die ganz neue Anwendungen von Verschlüsselung ermöglichen und die mit anderen Kryptosystemen bisher nur teilweise oder unter erheblichem Mehraufwand nachgebildet werden können, werden in Kapitel 8 beschrieben.

Zu guter Letzt sei noch erwähnt, daß es selbstverständlich auch Kryptosysteme gibt, die nur von manchen Experten, nämlich ihren Erfindern und Vermarktern, für sicher gehalten werden, von großen Gruppen von Experten jedoch unter „zweifelhafte Sicherheit“ eingestuft werden, da nah verwandte Systeme in den letzten Jahren gebrochen wurden. Beispiele solcher Kryptosysteme sind alle Varianten von auf rückgekoppelten Schieberegistern basierenden (Strom-)Chiffren [Plum_82, Sieg_84, Sieg_85, Sieg_86, Sie1_86, Rue1_86, Rue2_86, MeSt_88] sowie alle auf dem Rucksackproblem (knapsack problem) basierenden asymmetrischen Kryptosysteme [Sham_84, Bric_85].

Leider gibt es zur Zeit also weder praktikable symmetrische noch irgendwelche asymmetrischen Kryptosysteme, deren Sicherheit ganz ohne unbewiesene Annahmen über den Lösungsaufwand von Problemen bewiesen werden kann.

2.2.2.3 Realisierungsaufwand bzw. Verschlüsselungsleistung

Während im vorherigen Abschnitt Kryptosysteme unter dem Aspekt Sicherheit behandelt und lediglich hin und wieder eine Bemerkung über den Aufwand ihrer Realisierung gemacht wurde, wird dieser Aspekt nun vertieft, indem für die erwähnten Klassen von Kryptosystemen in gleicher Reihenfolge der Aufwand ihrer Realisierung bzw. ihre Verschlüsselungsleistung diskutiert wird.

Bei informationstheoretisch sicheren (und damit zwangsläufig symmetrischen) Kryptosystemen ist die eigentliche Ver- und Entschlüsselung sehr einfach und schnell durchführbar, da z. B. für die perfekte informationstheoretische Konzelation garantierende Vernam-Chiffre – wie erwähnt – lediglich Klartext und Schlüssel addiert werden müssen, um den Schlüsseltext zu erhalten, und lediglich der Schlüssel vom Schlüsseltext subtrahiert werden muß, um wieder den Klartext zu erhalten. Werden Addition und Subtraktion modulo 2 durchgeführt, was für eine Implementierung günstig ist, so sind sie zudem gleich, nämlich die Operation XOR. Sie kann mit wenigen Gattern in Hardware oder mit wenigen Befehlen in Software implementiert werden, wobei im ersteren Fall zur Zeit Verschlüsselungsraten von etlichen Gbit/s, im zweiten Fall bei Implementierung auf handelsüblichen und durchschnittlich leistungsfähigen PCs von etlichen Mbit/s erreichbar wären, gäbe es nicht das Problem, auch den Schlüssel in gleicher Geschwindigkeit zuzuführen. Da er früher rein zufällig erzeugt und in Integrität und Konzelation garantierender Weise ausgetauscht worden sein muß, befindet er sich zum Zeitpunkt der Ver- und Entschlüsselung auf einem Speicher, dessen (Zugriffs-)Bandbreite und Kapazität zum begrenzenden Faktor wird. Zur Verdeutlichung sei angeführt, daß der im Frühjahr 1987 vorgestellte Apple Macintosh II, ein PC der gehobenen Leistungsklasse, eine Transferrate zwischen Rechner und eingebauter 40 Mbyte SCSI-Festplatte von etwas mehr als 1 Mbyte/s ermöglicht [IEEE_87].

Offen wäre natürlich noch, wie der Schlüssel in Integrität und Konzelation garantierender Weise auf die Festplatte kommt – über das Kommunikationsnetz, dessen Kommunikation durch ihn geschützt werden soll, jedenfalls nicht. So bietet es sich an, kurz die Speicherkapazitäten wechselbarer Massenspeicher zur Kenntnis zu nehmen, nämlich z. B. 800 kbyte für die weit verbreiteten und robusten doppelseitig beschreibbaren 3,5 Zoll Disketten [IEEE_87] und 600 Mbyte für die angekündigten 5,25 Zoll „Erasable-Laser-Optical-Disks“ der Firma 3M [DuD4_87, Free_88]. Übergibt man in Zukunft statt oder als Ergänzung zu einer Visitenkarte einen solchen wechselbaren Massenspeicher, so kann man mit dem Partner zwar etwa 100 bzw. 75000 „Elektronische Briefe“ geschützt austauschen, diese imposanten Zahlen schrumpfen aber gewaltig, will man miteinander telefonieren (oder z. B. den TELEFAX-Dienst in Anspruch nehmen): bei der vorgesehenen Übertragungsrate von 64000 bit/s ist dies 100 bzw. 75000 Sekunden geschützt möglich. Macht man sich klar, daß ein durchschnittliches dienstliches Telefongespräch etwa 180 Sekunden dauert, so reicht die 3,5 Zoll Diskette nicht einmal für ein Gespräch und die 5,25 Zoll „Erasable-Laser-Optical-Disk“ nur für 416 Telefongespräche. Die letztere Zahl mag zwar beruhigend klingen, man denke aber auch an Bildfernsprechen, das, sofern eine dem heutigen PAL Fernsehbild zumindest entsprechende Qualität gewünscht wird, statt 64000 bit/s mindestens 34000000 bit/s benötigt [Kais_82 Seite 102, BrMo_83 Seite 204]. Damit sind dann nur noch 141 Sekunden Bildtelefonzeit selbst bei der 5,25 Zoll „Erasable-

Laser-Optical-Disk“ möglich. Fazit: Perfekte informationstheoretische Konzelation ist beim heutigen Stand der Speichertechnik für Dienste mit geringem Übertragungsvolumen, nicht aber generell für alle Dienste eines diensteintegrierenden Kommunikationsnetzes einsetzbar. Aus organisatorischen Gründen ist sie nur zwischen Partnern möglich, die sich früher einmal getroffen haben, sich in gewisser Weise also kennen. Die denkbare Möglichkeit, wechselbare Massenspeicher per Post zu verschicken, wird hier nicht betrachtet, da das Risiko von Kenntnisnahme durch Angreifer für mein Dafürhalten bei weitem größer als deren Chance zum Brechen der bereits genannten und im folgenden bezüglich Aufwand und Verschlüsselungsleistung bewerteten Kryptosysteme ist.

Der gerade dargelegte hohe Bedarf an Schlüsselaustauschkapazität kann drastisch verringert werden, indem nur der etwa 1000 bit lange Startwert eines kryptographisch starken Pseudozufallsbitfolgengenerators [VaVa_85, BlMi_84] ausgetauscht wird. Leider sind selbst die effizientesten sehr aufwendig (d. h. nur etwas effizienter als das weiter unten diskutierte RSA) und ihre Implementierungen also sehr langsam: Softwareimplementierungen dürften einige zig bit/s, Hardwareimplementierungen einige hunderttausend bit/s generieren können. Diese Raten sind zwar für die meisten Anwendungen bei weitem zu klein, aber warum sollte nicht unser PC in Zukunft, statt, wann immer wir ihn gerade nicht auslasten, seine Warteschleife abzuarbeiten, für alle unsere Kommunikationsbeziehungen kryptographisch starke Pseudozufallsbitfolgen generieren und für den späteren Gebrauch abspeichern. Die oben diskutierte Vernam-Chiffre garantiert dann zwar nur noch perfekte komplexitätstheoretische Konzelation, ein umfangreicher und zyklisch notwendiger Schlüsselaustausch wird dadurch aber unnötig.

Ähnlichen Aufwand dürften die in Abschnitt 2.2.2.2 erwähnten, in der komplexitätstheoretischen Modellwelt unter der Annahme, daß Faktorisierung oder Ziehen diskreter Logarithmen schwierig ist, bewiesenen sym- oder asymmetrischen Blockchiffren sowie Signatursysteme verursachen. Im Gegensatz zu kryptographisch starken Pseudozufallsbitfolgengeneratoren kann bei ihnen leider nicht auf Vorrat gearbeitet werden, da die zu ver- oder entschlüsselnde Nachricht von Anfang an in die Berechnung einbezogen werden muß. Eine Hardwareimplementierung ist also für die meisten Anwendungen unumgänglich.

Leider sind mir keine Implementierungen der innerhalb der komplexitätstheoretischen Modellwelt als sicher bewiesenen Kryptosysteme und folglich auch keine genauen Leistungsdaten bekannt. Alle genannten Zahlenbereiche sind also nur den Veröffentlichungen der Erfinder entnommene sowie von mir durch Vergleich mit den Leistungsdaten der im folgenden beschriebenen RSA-Implementierungen konkretisierte Schätzungen. Es sei noch darauf hingewiesen, daß sie die Zeiten nicht enthalten, die zur *Schlüsselgenerierung* aufzuwenden sind, z. B. zur Generierung eines Schlüsselpaares – (Chiffrierschlüssel, Dechiffrierschlüssel) oder (Schlüssel zum Signieren, Schlüssel zum Testen) – eines asymmetrischen Kryptosystems. Diese Zeiten können auch erheblich sein, es kann jedoch – wie oben erwähnt – auf Vorrat gearbeitet werden, so daß (außer für die Erzeugung der die Schlüsselgenerierung parametrisierenden echten Zufallszahlen) für die Schlüsselgenerierung keine Hardwareimplementierung erforderlich ist. Als Beispiel sei die Erzeugung von RSA-Schlüsselpaaren genannt, die nach [Jung_87] bei einer Blocklänge von 512 bit auf einer SIEMENS 7.541 (0,8 MIPS mit einem der IBM-370 ähnlichen Befehlssatz) im Durchschnitt 40 Sekunden dauert.

Realisierungen von Kryptosystemen, die weder in der informationstheoretischen noch in der komplexitätstheoretischen Modellwelt als sicher bewiesen sind, sind aus der Fachliteratur in großer Zahl bekannt. Bekannteste Vertreter sind hierbei – wie bereits erwähnt – RSA [RSA_78] als asymmetrisches Konzelationssystem und Signatursystem sowie DES (Data Encryption Standard, [DES_77]) als symmetrisches Kryptosystem.

Beide Systeme lassen sich ohne große Schwierigkeiten auf jedem PC in Software implementieren, können dann aber nur für schmalbandige Kommunikation verwendet werden und verhindern während der Verschlüsselung natürlich die sonstige Benutzung des PCs, sofern dieser nur zu Einprogrammbetrieb fähig ist, bzw. vermindern dessen ansonsten verfügbare Nutzleistung ganz erheblich, sofern dieser zu Mehrprogrammbetrieb fähig ist.

In [MüSc_83] wird als Leistung einer auf einem 5 MHz 8086-Prozessor der Firma Intel durchgeführten RSA Implementierung 220 Sekunden pro Block bei einer Blocklänge von 288 bit angegeben, in [Bras_88 Seite 31] 9 Sekunden pro Block bei einer Blocklänge von 512 bit für den IBM Personal Computer (Intel 8088, 4,77 MHz). In [Jung_87] werden für RSA 1,5 Sekunden pro Block bei einer Blocklänge von 512 bit für die SIEMENS 7.541 genannt und 45 bzw. 1,5 Sekunden pro Block bei einer Blocklänge von 256 bit für den EPSON PX-8 (Z80 kompatibler Mikroprozessor, 2,45 MHz) bzw. SIEMENS PC-X (Intel 80186). Diese Leistung ist durch Verwendung von 32-Bit-Prozessoren mit schnellerer Taktrate sicherlich wesentlich steigerbar, jedoch ist bereits 288 bit Blocklänge deutlich unterhalb dessen, was bei RSA für die in Abschnitt 2.2.1.2 angesprochene Anwendung veröffentlichter Schlüssel als ausreichende Schlüssellänge betrachtet werden kann (selbst 512 bit können dafür nicht mehr als sehr reichlich dimensioniert gelten) [PoST_88]. Selbst wenn der Angreifer, etwa bedingt durch das Protokoll, in dem RSA verwendet wird, für seinen Angriff nur wenige Sekunden zur Verfügung hat, ist 288 bit an der unteren Grenze dessen, was als ausreichende Schlüssellänge betrachtet werden kann.

Softwareimplementierungen von DES erreichen eine weit höhere Verschlüsselungsleistung. In [Goel_86 Seite 26] wird als Wert einer guten Softwareimplementierung auf einem 4 MHz Z-80-Prozessor der Firma Zilog eine Verschlüsselungsleistung von 1 kbit/s angegeben, in [WiHi_80] 3,5 kbit/s für einen 2 MHz 8080-Prozessor der Firma Intel und in [KaRS_88 Seite 27, Bras_88 Seite 17] eine von 19 bis 20 kbit/s für den IBM Personal Computer. Eine von Ralf Aßmann durchgeführte Assembler-Implementierung für den MC680XY-Prozessor der Firma Motorola erreicht auf dem Apple Macintosh Plus (MC68000, 7,83 MHz) eine Verschlüsselungsleistung von 67 kbit/s, auf dem Apple Macintosh II (MC68020, 15,67 MHz) 322 kbit/s und auf dem Apple Macintosh IIx (MC68030, 15,67 MHz) 359 kbit/s [Aßma_88]. Für die Betriebsart Ergebnisrückführung (ohne Auswahl oder Ergänzung, vgl. Bild 12) wurde sogar jeweils eine um 19%, 28% bzw. 25% höhere Verschlüsselungsleistungen erzielt. In Ralf Aßmanns Programm scheinen höchstens noch Optimierungen möglich, die die Verschlüsselungsleistung unwesentlich steigern.

Die Leistung softwareimplementierter Verschlüsselung mag zwar den Anforderungen geschlossener und mit genügend vielen oder leistungsfähigen PCs ausgestatteter Benutzergruppen genügen; um effizient zu ver- und entschlüsseln oder die für breitbandige Dienste notwendigen Verschlüsselungsraten zu erreichen, müssen die Kryptosysteme aber in Hardware implementiert werden. Für beide Systeme stehen Hardwareimplementierungen zur Verfügung, mit denen gegenwärtig RSA 64 kbit/s (bei einer Blocklänge von 660 bit [SeGo_86]; in [Sedl_88] werden

200 kbit/s bei gleicher Blocklänge angekündigt) und DES 15 Mbit/s [AT&T_86, Abbr_84] (in [Bras_88 Seite 16] werden 20 Mbit/s genannt, in [VHVD_88] 32 Mbit/s angekündigt) zu verschlüsseln erlaubt.

Deshalb ist hinsichtlich der Leistung der Einsatz eines Systems wie RSA (oder besser: der eines innerhalb der komplexitätstheoretischen Modellwelt als sicher bewiesenen, eher etwas effizienteren asymmetrischen Konzelationssystems [BlGo_85]) zur Schlüsselverteilung sowie von DES (oder besser: einem auf demselben Prinzip beruhenden mit längerer Block- und vor allem Schlüssellänge, vgl. Abschnitt 2.2.2.2 sowie den Anhang) zur Verschlüsselung großer Datenströme möglich. Mit dem Masseneinsatz der heute auf einem, demnächst als Teil eines Chips hardwareimplementierten Kryptosysteme wäre zugleich auch deren Preisgünstigkeit garantiert. Bereits heute ist das oben erwähnte, noch nicht für den Masseneinsatz produzierte DES-Chip für etwa 45$ erhältlich [Sumn_87]. Die Entwürfe derartiger DES-Chips (und natürlich auch die oben erwähnten Software-Implementierungen von DES) können in einfachster Weise so modifiziert werden, daß aller bisher an der Sicherheit von DES geäußerten Kritik entsprochen ist. Für den mit DES vertrauten Leser sind diese Modifikationen als Anhang beschrieben.

Der Vollständigkeit halber sei noch erwähnt, daß Implementierungen (auf einem oder als Teil eines Chips) von auf Schieberegistern basierenden Stromchiffren ohne weiteres Verschlüsselungsraten nahe der Schaltrate eines einzelnen Gatters erreichen, heutzutage also etliche Gbit/s. Ebenso sind Implementierungen von asymmetrischen Kryptosystemen, die auf dem Rucksackproblem basieren, mit Verschlüsselungsraten von 10 Mbit/s denkbar [DeVG_84 Seite 192]. Jedoch sei nochmals an die fragwürdige Sicherheit dieser Kryptosysteme erinnert.

2.2.2.4 Registrierung geheimer oder Standardisierung und Normung öffentlicher Kryptosysteme?

Es sei angemerkt, daß die amerikanische Sicherheitsbehörde NSA (National Security Agency) 1985 mit einem recht eigentümlichen Vorschlag zur Realisierung von Verschlüsselung im nichtmilitärischen, nichtstaatlichen Bereich in der Nachfolge von DES hervorgetreten ist, nachdem ihre Zuständigkeit vom Schutz militärischer oder geheimdienstlicher Kommunikation auf den Schutz der Kommunikation der staatlichen Verwaltung und Industrie ausgedehnt wurde [Kola_85, Horg_86, Hor1_86, NePi_86, Rose_86, Atha_86, Ath1_86, Jurg_86, Hig1_86, High_87, NBS_87, DuDR_87]:

Das Kryptosystem soll von der NSA gewählt werden und (im Gegensatz zu DES und auch allen anderen, vor 1985 für einen breiten öffentlichen Einsatz vorgesehenen Systemen) geheim bleiben – angeblich, um ein Brechen des Systems zu erschweren und Gegnern keine guten Kryptosysteme zu verraten. Die Ver- und Entschlüsselungsalgorithmen würden dazu nur in *vor Ausforschung geschützten*, d. h. auf Manipulationsversuche reagierenden und unerwünschte elektromagnetische oder mechanische Abstrahlung unter die Sensitivität professioneller Empfangsgeräte dämpfenden, Chips bzw. Geräten ausgeliefert. Der Schlüsselgenerierungsalgorithmus soll sogar völlig bei der NSA verbleiben, die Benutzer des Systems müßten sich für jede Kommunikationsbeziehung von der NSA Schlüssel zuweisen lassen. Dies würde vermutlich

dadurch realisiert, daß die Geräte selbstgewählte Schlüssel mit großer Wahrscheinlichkeit ablehnen. Teilweise wird gesagt, daß Benutzer auch einen Algorithmus zur Generierung von Schlüsseln erhalten können, die damit generierten Schlüssel aber schlechter seien als die von der NSA gelieferten [Kola_85]. Letzteres könnte daran liegen, daß für den Schlüsselgenerierungsalgorithmus keine ausforschungssicheren Geräte vorgesehen sind und die NSA ihn nicht vollständig verraten möchte, es könnte einfach eine abschreckende Behauptung sein oder die ausforschungsgeschützten Chips könnten für diese nach dem nicht völlig geheimen Algorithmus erzeugten Schlüssel einen ganz anderen, leichter zu brechenden Ver- und Entschlüsselungsalgorithmus enthalten. Sicherheitsargumente dafür, den Schlüsselgenerierungsalgorithmus nicht auch in die ausforschungsgeschützten Chips aufzunehmen, gibt es nicht, Leistungs- oder Kostenüberlegungen könnten aber eine Rolle spielen.

Schon die Sicherheitsargumente für die Geheimhaltung der Algorithmen sind sehr zweifelhaft:

Zum einen ist die Wahrscheinlichkeit sehr groß, daß ein Angreifer mit ausreichenden finanziellen oder technischen Mitteln die Algorithmen doch erfährt, entweder durch Spionage oder durch Brechen des Ausforschungsschutzes (die Bewertung der Sicherheit von Geräten ist heutzutage noch viel problematischer als die von Kryptosystemen). Zum anderen verliert man durch die Geheimhaltung den Vorteil einer Sicherheitsvalidierung durch den heutzutage recht großen öffentlich arbeitenden Teil der kryptologischen Fachwelt, durch die ein Algorithmus mit Schwächen (und nur ein solcher bedürfte ja einer Erschwerung des Brechens durch Geheimhaltung) vermutlich ausgeschieden wird, bevor es überhaupt zu seiner Standardisierung kommt. Aus diesen Gründen wurde bei der Auswahl von DES die Öffentlichkeit ausdrücklich verlangt, und es wird auch befürwortet, daß die für die Sicherheit des Systems Verantwortlichen für das Mitteilen von Mängeln hohe Belohnungen aussetzen [Bara_64, DiHe_79 Seite 420 und 421, DaPr_84 Seite 51]. Das Argument, den Gegnern keine guten Algorithmen verraten zu wollen, ist zudem irrelevant, solange es eine genügende Auswahl an guten Algorithmen gibt, wie dies zur Zeit der Fall zu sein scheint (z. B. LUCIFER-artige [FeNS_75, HeKW_85] für symmetrische und [BlGo_85] für asymmetrische Kryptosysteme).

Es ist natürlich auch denkbar (wie dies auch schon bei DES diskutiert wurde), daß absichtlich ein Kryptosystem gewählt werden soll, das zumindest insoweit Schwächen hat, daß es der NSA auch in dem Fall, daß sich die Schlüsselzuteilung durch die NSA nicht durchsetzen sollte, noch gestattet, ausgewählte Nachrichten zu entziffern, und das Risiko, daß dies nach einer Weile auch Gegner können, bewußt eingegangen wird (für militärische Geheimnisse sollen andere Kryptosysteme verwendet werden), ein öffentliches Bekanntwerden dieses Punktes aber unerwünscht ist.

Wenn natürlich die Schlüssel tatsächlich von der NSA zugewiesen würden, wären solche absichtlichen Schwächen überflüssig: Dann wäre die NSA ohnehin in die einzigartige Lage versetzt, jede für sensitiv gehaltene Nachricht mühelos mitlesen zu können. Ein solches System wäre damit die offizielle Installation eines „Großen Bruders“ [Orwe_49].

Es ist zu befürchten, daß die entsprechenden bundesdeutschen Stellen, namentlich die Zentralstelle für das Chiffrierwesen (ZfCh) in Bonn, versuchen werden, es ihrem amerikanischen Bruder nachzumachen:

International und national (durch ISO und DIN) wurde versucht, asymmetrische und symmetrische Kryptosysteme, z. B. RSA und DES, zu normen. Diese, sich teilweise bereits in einem fortgeschrittenen Stadium der Normung befindenden Versuche sind teils durch offenes, teils durch verdecktes Betreiben von NSA, ZfCh (und vielleicht auch anderen) abgebrochen worden – nur an der Normung von Protokollen, die Kryptosysteme *verwenden*, soll noch gearbeitet werden [Folt_87, Pric_88]. Lediglich ein Register ist geplant, in das sowohl vollständig veröffentlichte Kryptosysteme als auch solche, deren genaue Beschreibung geheim bleiben soll, aufgenommen werden. Für letztere Klasse von Kryptosystemen enthielte es nur einen Namen und ggf. die für die Verwendung in Protokollen relevante äußere Spezifikation, z.B. „symmetrische Blockchiffre mit 64 Bit Block- und 56 Bit Schlüssellänge“. Während bisher mit der Normung auch eine gewisse Aussage über die Güte der Kryptosysteme gegeben war, soll die Aufnahme in das geplante Register hierüber keinerlei Aussage machen.

Wenn auch in der bisherigen Diskussion offene diensteintegrierende Netze nicht explizit vorkommen, so hat all dies auf den Datenschutz in ihnen besonders schwerwiegende, wenn auch nur implizite Auswirkungen (vgl. den aus meiner Sicht das Problem verharmlosenden Artikel [Hell_87]).

Mittlerweile ist in den USA die Verantwortung zur Normierung von Kryptosystemen für den nichtmilitärischen, nichtstaatlichen Bereich allerdings zurück an die amerikanische Normungsbehörde für den öffentlichen Bereich (NBS = National Bureau of Standards) übertragen worden [CACM6_87, Prei_88]. Bisher hat dies aber keinen erkennbaren Einfluß auf die US-amerikanische, internationale oder deutsche Normung.

Inwieweit die ZfCh ihrem amerikanischen Bruder nacheifern wird, ist offiziell nicht bekannt. Mit der Normung im Bereich Kryptographie Befaßte teilten mir jedoch im Dezember 1986 und Januar 1987 mündlich mit, die ZfCh habe erreicht, daß das DIN die Normung von DES ersatzlos einstellte und auch in der ISO für einen solchen, alle symmetrischen Kryptosysteme betreffenden Entschluß stimmte.

Anscheinend übersieht diese dem Bundeskanzleramt nachgeordete Bundesbehörde sowie eine Mehrheit des zuständigen DIN Ausschusses AA-20 den in Abschnitt 1.4 wiedergegebenen Teil der Urteilsbegründung des Volkszählungsurteils des Bundesverfassungsgerichts vom Dezember 1983. Da bei Verwirklichung des oben beschriebenen NSA-Vorschlags in der Bundesrepublik die Bürger sicherlich „nicht mit hinreichender Sicherheit überschauen ...“, und nach den im selben Abschnitt angeführten Argumenten keine triftigen Gründe zur Einschränkung des Rechtes auf informationelle Selbstbestimmung existieren, stellt sich auch hier die Frage, ob der Verzicht auf Normung von Kryptosystemen bei sinngemäßer Auslegung des Volkszählungsurteils nicht verfassungswidrig ist?

Erfreulicherweise teilen Vertreter der DBP diese Meinung, so daß man, wenn schon nicht auf die formelle Normung durch ISO oder DIN, so doch auf eine kurzfristige Standardisierung öffentlicher Kryptosysteme durch den Betreiber der Kommunikationsnetze und in Folge davon kompatible und effiziente Hardwareimplementierungen hoffen kann. Die am Ende des vorherigen Abschnitts 2.2.2.3 andiskutierten (und im Anhang etwas ausführlicher erläuterten und begründeten) abwärtskompatiblen Modifikationen von DES sind mein Vorschlag für eine kurzfristig verfügbare und effiziente Lösung für den Bereich symmetrischer Kryptosysteme.

Weitere Argumente zum Thema dieses Abschnitts sind in [Pfi2_87, Riha_87, Rih1_87, WaPP_87, PWP_87] zu finden.

2.3 Einsatz und Grenzen von Verschlüsselung in Kommunikationsnetzen

2.3.1 Einsatz von Verschlüsselung in Kommunikationsnetzen

Für den Einsatz eines Kryptosystems zum Schutz der Kommunikation (vor allem zum Zweck der Konzelation, aber auch zum Zweck der Integrität oder Authentifikation) hat man zwei Strategien zur Auswahl, die leider beide Nachteile haben: Verbindungs-Verschlüsselung und Ende-zu-Ende-Verschlüsselung.

2.3.1.1 Verbindungs-Verschlüsselung

Die erste Strategie besteht darin, *alle* Daten jeweils zwischen benachbarten Netzknoten, d. h. Teilnehmerstationen und Vermittlungszentralen, zu verschlüsseln (**Verbindungs-Verschlüsselung**, link-by-link encryption) [Bara_64, Denn_82, VoKe_83, DaPr_84].

Es sollte ein *gleichmäßiger Zeichenstrom* übertragen werden, damit ein Abhörer nicht beobachten kann, wann keine Nachrichten übertragen werden. Ein gleichmäßiger Zeichenstrom ist bei allen Übertragungsstrecken, die den verbundenen Netzknoten *statisch* zugeordnet sind, z. B. Punkt-zu-Punkt-Leitungen und Richtfunkstrecken, ohne Mehraufwand möglich.

Aus den in Abschnitt 2.2.2.1 dargelegten Gründen, sollte und kann eine sichere *Stromchiffre* verwendet werden: Würde eine Blockchiffre verwendet, so könnte, solange der Schlüssel nicht gewechselt würde, der Abhörer zumindest manchmal beobachten, daß sich gewisse Nachrichten(fragmente) wiederholen. Der Abhörer könnte also manche Nachrichten *verketten* (vgl. Abschnitt 2.1.1).

Wird ein gleichmäßiger, mit einer sicheren Stromchiffre verschlüsselter Zeichenstrom übertragen, erhält ein Angreifer durch Abhören der Übertragungsstrecken, d. h. von Leitungen (Glasfasern, Koaxialkabel, Kupferdoppeladern), Richtfunk- oder Satellitenstrecken keine Information mehr. Ob und ggf. welche Nachrichten übertragen werden, ist für ihn *perfekt unbeobachtbar* (vgl. Abschnitt 2.1.1).

Die Nachteile von Verbindungs-Verschlüsselung sind:

– In den Vermittlungszentralen liegen alle Daten unverschlüsselt vor, von allen in Abschnitt 1.2 genannten möglichen Angreifern werden also nur diejenigen ausgeschlossen, die Übertragungsstrecken abhören.

– In der in Bild 2 bzw. 15 gezeigten Endsituation der von der DBP geplanten Entwicklung der Kommunikationsnetze werden auf der den Netzabschluß des Teilnehmers mit der

Vermittlungszentrale verbindenden Leitung, einer Glasfaser, mindestens 560 Mbit/s übertragen. Dies ist eher oberhalb dessen, was heute mit Kryptogeräten, die auf einem für halbwegs sicher gehaltenen Kryptosystem beruhen und halbwegs preiswert sind, verschlüsselt werden kann, vgl. Abschnitt 2.2.2.3. Es wäre zumindest von Vorteil, wenn man Fernsehen, insbesondere hochauflösendes Fernsehen (High Definition TV, HDTV), als breitbandigen Dienst, bei dem es nicht um den Schutz von Inhalts-, sondern von Interessensdaten geht, nicht verschlüsseln müßte. Wieviel hierdurch eingespart werden kann, verdeutlichen die folgenden Zahlen: für Fernsehen heutiger Qualität (PAL) werden ohne bzw. mit Redundanzreduktion etwa 140 Mbit/s bzw. 34 Mbit/s benötigt, für hochauflösendes Fernsehen etwa viermal soviel.

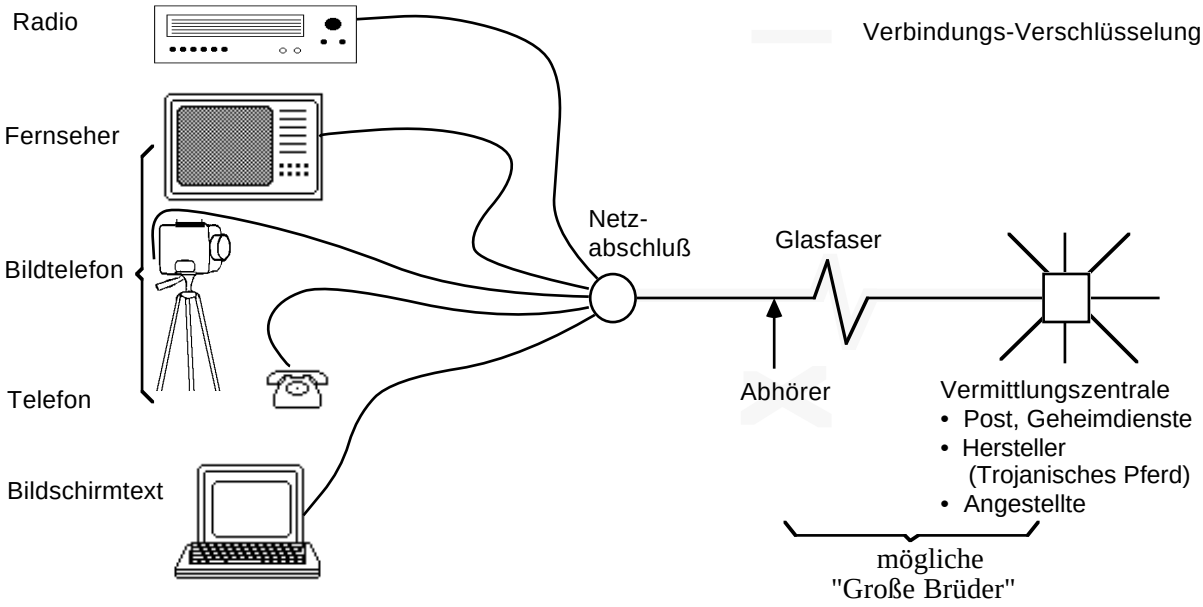


Bild 15: Verbindungs-Verschlüsselung zwischen Netzabschluß und Vermittlungszentrale

In jedem Fall benötigen benachbarte Netzknoten jeweils zueinander passende und leistungsfähige, also direkt in Hardware implementierte Kryptosysteme, zweckmäßigerweise jeweils selbstsynchronisierende Stromchiffren. Da die Nachbarschaft von Netzknoten vergleichsweise statisch ist, können Schlüssel einer Verbindung zugeordnet werden, so daß ein symmetrisches Kryptosystem den Anforderungen vollauf genügt.

2.3.1.2 Ende-zu-Ende-Verschlüsselung

Die zweite Strategie ist, die Daten zwischen Teilnehmerstationen verschlüsselt zu übertragen (**Ende-zu-Ende-Verschlüsselung**, end-to-end encryption) [Bara_64, Denn_82, VoKe_83, DaPr_84], damit sie in der (bzw. bei erweiterter Betrachtung: den) Vermittlungszentrale(n) nicht interpretiert werden können (Bild 16).

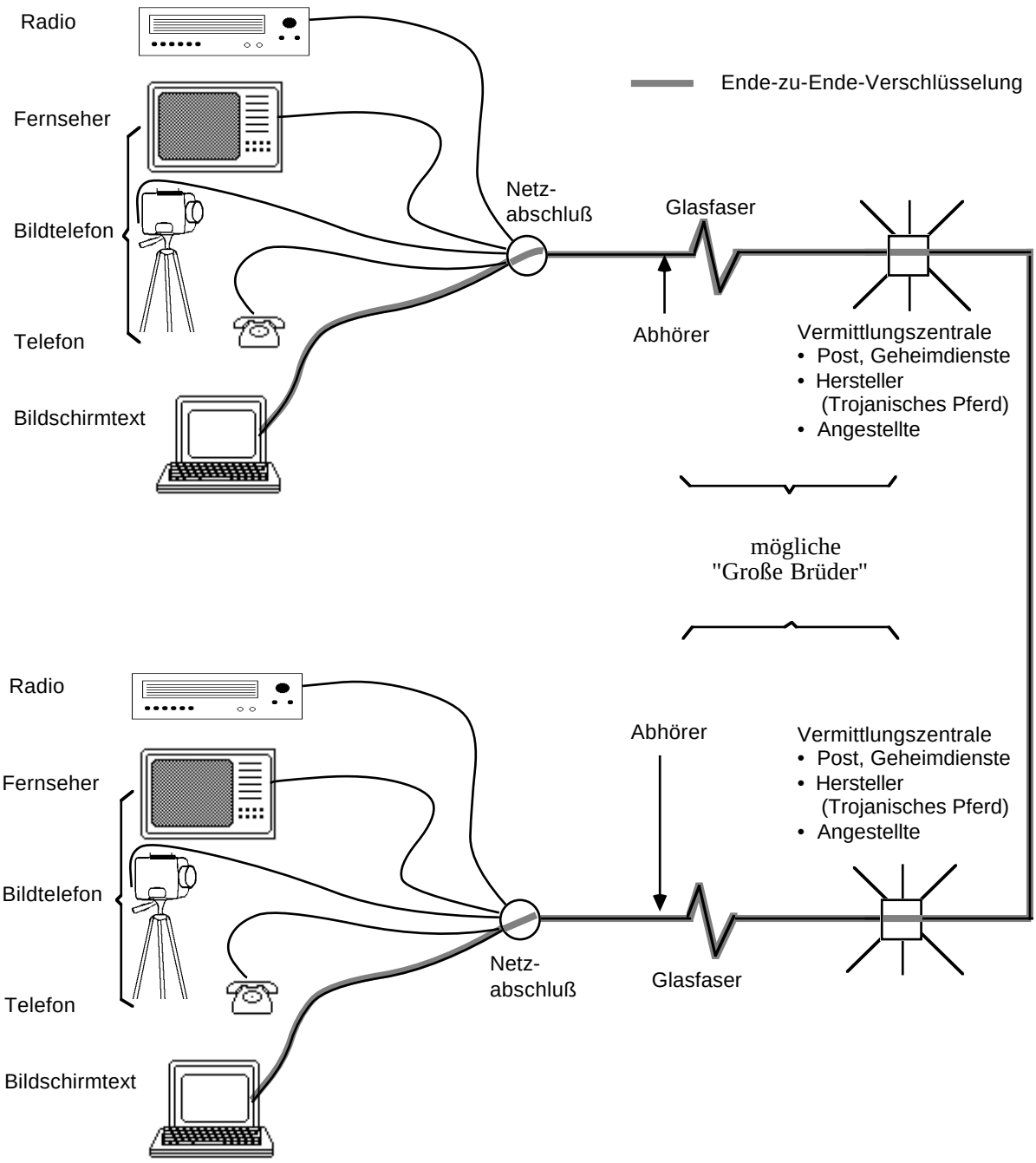


Bild 16: Ende-zu-Ende-Verschlüsselung zwischen Teilnehmerstationen

Aus den schon bei Verbindungs-Verschlüsselung dargelegten Gründen sollte auch für Ende-zu-Ende-Verschlüsselung möglichst ein gleichmäßiger Zeichenstrom (zumindest bei Kanalvermittlung für die Dauer des Kanals) und in jedem Fall eine Stromchiffre verwendet werden.

Die Nachteile von Ende-zu-Ende-Verschlüsselung sind:

– Durch Ende-zu-Ende-Verschlüsselung können nur die Nutzdaten, nicht die Vermittlungsdaten und damit auch nicht die *Verkehrsdaten* geschützt werden.

– Vor jemandem, der die Nutzdaten schon vorher kannte und nun die Vermittlungsdaten erhält, sind damit auch die *Interessensdaten* (vgl. Abschnitt 1.2) ungeschützt, was zu weiteren Verkettungen von Verkehrsdaten und dann wiederum zur Gewinnung von weiteren Interessensdaten usw. verwendet werden kann.

  Insbesondere kann dieses Verfahren also nicht als Schutz vor dem Netzbetreiber und anderen, die Zugriff auf die Rechner des Netzbetreibers haben, dienen, wenn diese gleichzeitig Kommunikationspartner sind oder wenn sich die Nutzdaten in Form einer Datenbank in einem Rechner des Netzbetreibers (z. B. Bildschirmtext-Zentrale in Ulm) befinden. Die Nutzdaten können zwar in verschlüsselter Form in der Datenbank abgespeichert werden. Dies nützt jedoch nur etwas, wenn der Angreifer (Netzbetreiber bzw. andere mit Zugriff auf die Rechner des Netzbetreibers) die zugehörigen Schlüssel nicht kennt und auch nicht in Erfahrung bringen kann. Letzteres erscheint außer bei kleinen geschlossenen Benutzergruppen unrealistisch, da der Angreifer in große geschlossene Benutzergruppen einen Strohmann einschleusen und bei offenen Benutzergruppen als normaler Nutzdateninteressent auftreten kann.
  Daneben ist auch eine Zusammenarbeit von jemandem, der an Daten in der Vermittlungszentrale gelangen kann, und dem Kommunikationspartner denkbar. Dies könnte z. B. von einem Geheimdienst ausgehen, der über den Netzbetreiber die Vermittlungsdaten erhält und Kommunikationspartner, z. B. Datenbanken, Zeitungsverlage, veranlaßt, ihm die Nutzdaten offenzulegen, aber auch vom Kommunikationspartner, der über Mitarbeiter o. ä. Zugang zu den Vermittlungsdaten erhält.

  Außerdem ist es trivialerweise sinnlos, sich mit Ende-zu-Ende-Verschlüsselung vor Kommunikationspartnern schützen zu wollen.

– Wie bei Verbindungs-Verschlüsselung ist es auch hier recht umständlich, auch Fernsehen, insbesondere hochauflösendes Fernsehen, verschlüsseln zu müssen, um die Interessensdaten vor den Angreifern in den Vermittlungszentralen zu schützen.

Soll Ende-zu-Ende-Verschlüsselung zwischen beliebigen Teilnehmerstationen möglich sein, benötigen sie jeweils paarweise zueinander passende Kryptosysteme. Soll Ende-zu-Ende-Verschlüsselung auch für breitbandigere Kommunikation möglich sein, muß zumindest ein Kryptosystem direkt in Hardware implementiert sein (vgl. Abschnitt 2.2.2.3). Wegen der dynamischen Natur der Kommunikationsbeziehungen zwischen Teilnehmern kann man diesen Beziehungen nicht statisch Schlüssel zuordnen, da es bei großen Teilnehmerzahlen zu viele potentielle Beziehungen, nämlich $n \cdot (n-1)/2$ bei n Teilnehmern, gibt. Aus all diesen Gründen ist eine Standardisierung, oder besser noch eine formelle Normung, eines asymmetrischen Kryptosystems zum Schlüsselaustausch und zur Authentifikation sowie eines schnellen symmetrischen Kryptosystems zur Ver- und Entschlüsselung der Nutzdaten dringend notwendig, soll das

Kommunikationsnetz bezüglich Datenschutz (oder auch Rechtssicherheit, vgl. [PWP_87, WaPP_87]) ein **offenes** System bilden, wie dies insbesondere für das ISDN geplant ist. Denn nur auf der Basis von Standards oder Normen können Implementierungen von Kryptosystemen so gestaltet werden, daß sie effizient sind und jeweils *beliebige* Paare zueinander passen.

Ohne Standardisierung und Normung von geeigneten Kryptosystemen ist Datenschutz (ebenso wie Datensicherheit) nur innerhalb **geschlossener** Benutzergruppen, die sich jeweils auf ein Kryptosystem geeinigt und eine genügend leistungsfähige Implementierung haben, erreichbar. An dieser Stelle sei noch einmal unterstrichen, welch verheerende Folgen die in Abschnitt 2.2.2.4 beschriebenen Pläne der NSA (bzw. ZfCh) zur Verwendung geheimer Kryptosysteme im öffentlichen Bereich haben können – und sei es „nur“ eine langjährige Verzögerung der Einführung standardisierter oder genormter, öffentlich validierter Kryptosysteme und der entsprechenden Implementierungen.

2.3.2 Grenzen von Verschlüsselung in Kommunikationsnetzen

Selbst wenn, wie in Bild 17 gezeigt, Verbindungs- und Ende-zu-Ende-Verschlüsselung eingesetzt werden [Bara_64, Denn_82, VoKe_83, DaPr_84], bleiben im wesentlichen alle in Abschnitt 2.3.1.2 für Ende-zu-Ende-Verschlüsselung aufgezählten Nachteile erhalten. Lediglich der erste Nachteil wird in der Form eingeschränkt, daß (wie bei Verbindungs-Verschlüsselung allein) Abhörer der Übertragungsstrecken bei Übertragung eines gleichmäßigen, mit einer sicheren Stromchiffre verschlüsselten Zeichenstroms bei der Verbindungs-Verschlüsselung keine Information mehr erhalten.

Auch die bekannten kryptographischen Techniken erlauben also auf der heute üblichen und auch für die Zukunft geplanten, für Kommunikation in beiden Richtungen vorgesehenen Netzstruktur, nämlich der reiner Vermittlungsnetze, keinen ausreichenden und mit vernünftigem Aufwand **überprüfbaren** Datenschutz für Verkehrs- und Interessensdaten vor vielen möglichen Angreifern in den Vermittlungszentralen und Kommunikationspartnern. Dies ist bei dem als reines Vermittlungsnetz geplanten, diensteintegrierenden Kommunikationsnetz eine besonders schwerwiegende und bei Realisierung der Planung langandauernde Beeinträchtigung des Rechtes auf informationelle Selbstbestimmung.

Deshalb wird im folgenden untersucht, wie der noch fehlende Schutz durch andere Maßnahmen erreicht werden kann. Dabei muß darauf geachtet werden, daß nicht jemand, der bisher als möglicher Angreifer gar nicht auftrat, z. B. Nachbarn, plötzlich Beobachtungsmöglichkeiten erhält.

Technisch gesehen ist dabei das Hauptziel, die Verkehrsdaten vor dem Betreiber der Vermittlungseinrichtungen zu schützen und sich auch gegenüber dem Kommunikationspartner nicht identifizieren zu müssen. Damit sind Verkehrs- und Interessensdaten nicht nur vor diesen geschützt, sondern (und das ohne zusätzliche Verschlüsselung von nicht vertraulichen Nutzdaten wie Fernsehen) erst recht vor anderen Angreifern in den Vermittlungseinrichtungen oder Abhörern, da alle diese höchstens genausoviel Information erhalten können wie der Betreiber selbst. Der Schutz der Inhaltsdaten wird dann zusätzlich durch Ende-zu-Ende-Verschlüsselung

sensitiver Nutzdaten erreicht. Obwohl also durch Verschlüsselung allein die Datenschutzprobleme in einem Kommunikationsnetz nicht gelöst werden können, so bildet sie doch die Basis vieler der im folgenden noch zu beschreibenden technischen Datenschutzmaßnahmen.

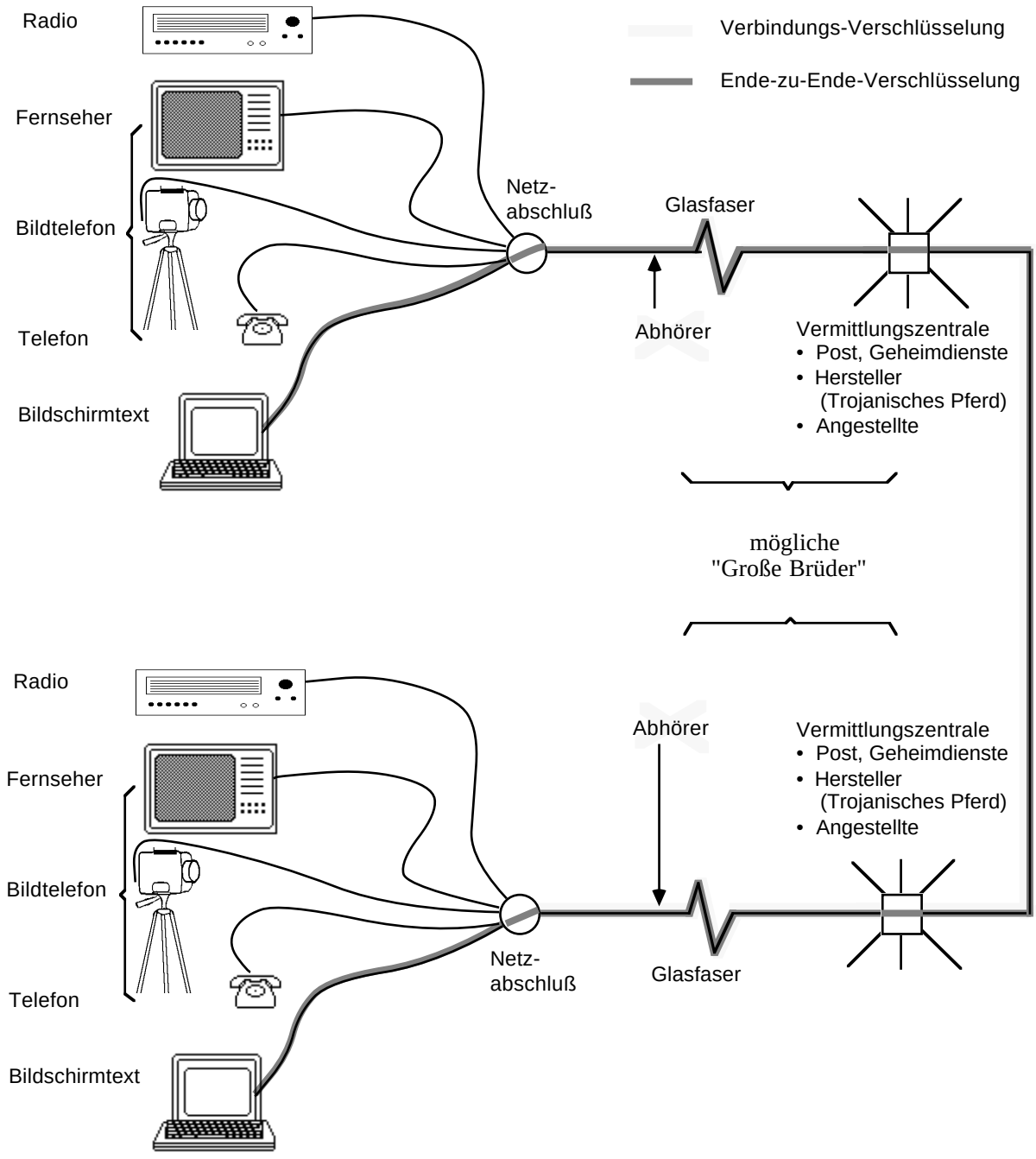


Bild 17: Ende-zu-Ende-Verschlüsselung zwischen Teilnehmerstationen und Verbindungs-Verschlüsselung zwischen Netzabschlüssen und Vermittlungszentralen sowie zwischen Vermittlungszentralen

In den folgenden zwei Abschnitten werden grundlegende Verfahren zum Schutz von Verkehrs- und Interessensdaten vor Angreifern in Vermittlungszentralen und Kommunikationspartnern beschrieben und wo angebracht, wird ihre Wirksamkeit bewiesen:

In Abschnitt 2.4 werden Schutzmaßnahmen beschrieben, die *außerhalb* des Kommunikationsnetzes angesiedelt sind, d. h. die jeder Benutzer für sich trifft. Diese werden sich als nicht ausreichend erweisen.

Danach werden in Abschnitt 2.5 solche Schutzmaßnahmen beschrieben, die *innerhalb* des Kommunikationsnetzes angesiedelt sind, d. h. die Benutzung des Netzes nicht verändern, jedoch den Transport innerhalb des Netzes.

Im darauf folgenden letzten Abschnitt dieses Kapitels werden dann alle bekannten Verfahren in ein *Schichtenmodell* eingeordnet, was zur Strukturierung des darauffolgenden Kapitels 3, in dem Überlegungen zu ihrer Implementierung dargestellt werden, dient.

2.4 Grundverfahren außerhalb des Kommunikationsnetzes zum Schutz der Verkehrs- und Interessensdaten

Bei Schutzmaßnahmen außerhalb des Kommunikationsnetzes sind Ziel- und Herkunftsadresse einer Nachricht weiterhin im Netz als Vermittlungsdaten sichtbar. Dazu sind natürlich die Zeit, zu der eine Nachricht im Netz ist, und zumindest für den Kommunikationspartner die Nutzdaten sichtbar. Man muß verhindern, daß daraus Schlüsse auf Verkehrs- und Interessensdaten gezogen werden können.

2.4.1 Öffentliche Anschlüsse

Gibt es für das Kommunikationsnetz öffentliche Anschlüsse, z. B. Telefonzellen für das Fernsprechnetz, oder ist das Kommunikationsnetz zumindest von einem anderen erreichbar, für das es öffentliche Anschlüsse gibt, so ist die folgende Datenschutzmaßnahme praktizierbar:

Die Herkunftsadresse und Zieladresse einer Nachricht werden weitgehend bedeutungslos, wenn man **verschiedene öffentliche Anschlüsse benutzt**. Dies gilt natürlich nur, wenn man sich nicht zu Zwecken der Zugangskontrolle oder der Zahlung von Gebühren identifizieren muß, d. h. anonym bleibt [Pfit_85].

Die DBP beabsichtigt, dies mittels und bei ihren Buchungskarten zu verhindern [BfD_88 Seite 38]:

> *„Der BMP hat mich jedoch über seine Absicht informiert, in Zukunft auf der Karte nicht nur den Standort des Kartentelefons zu registrieren, von dem aus ein Gespräch geführt wurde, sondern auch die Einzelgesprächsdaten, insbesondere die angewählte Rufnummer. Unter Datenschutzgesichtspunkten werfen solche Registrierungen der dem Fernmeldegeheimnis unterliegenden und daher sehr schutzbedürftigen Gesprächsdaten erhebliche Probleme auf. Sorgfältiger Untersuchung bedarf dabei die*

> *Frage, ob schutzwürdige Belange des* Angerufenen *dadurch beeinträchtigt werden können, daß ohne seine Einwilligung nicht nur registriert wird, daß, zu welchem Zeitpunkt und mit welcher Dauer sein Anschluß benutzt wurde, sondern – sofern es sich um ein ISDN-Anschluß handelt – auch, ob telefoniert oder welche anderen Kommunikationsform genutzt wurde."*

Die Anwendung und Wirkung der Datenschutzmaßnahme „Benutzung verschiedener öffentlicher Anschlüsse" ist stark eingeschränkt:

Wer etwa will für jedes einzelne Telefonat die Wohnung oder das Büro verlassen bzw., um auch eine zeitliche Verkettung zu erschweren (vgl. Abschnitt 2.4.2), den öffentlichen Anschluß wechseln? Wer will sich gar zu vorher vereinbarten Zeitpunkten in eine Telefonzelle begeben, um einen Rückruf unter nur erhoffter (denn es ist ja vorher klar, wo er sich physisch befinden wird: konventionelle Observierung!) Wahrung seiner Anonymität entgegennehmen zu können? Wer will in (Bild-)Telefonzellen noch so hochauflösendes Fernsehen genießen?

Selbst wenn alle diese Fragen positiv zu beantworten wären, bliebe die Wirkung dieser Maßnahme eingeschränkt: Selbst wenn jeder zwischen verschiedenen Telefonaten die Telefonzelle wechselt, bleibt (aus Gründen des Energie- und Zeitaufwandes) eine starke Lokalität erhalten. Selbst wenn der *einzelne* so unter einigen tausend Leuten anonym wäre, gäben seine *Kommunikationsbeziehungen* doch eine Möglichkeit, seine Sende- oder Empfangsereignisse zu verketten und ihn somit im Laufe der Zeit doch nach und nach zu identifizieren – und sei es nur mithilfe der Hypothese, daß sein Wohn- bzw. Arbeitsort ganz in der Nähe des Schwerpunktes des durch die von ihm benutzten Telefonzellen gebildeten, mit der Häufigkeit ihrer Benutzung in den entsprechenden Zeiträumen gewichteten Telefonzellenpolygons liegt.

2.4.2 Zeitlich entkoppelte Verarbeitung

Die **Zeit**, zu der sich eine Nachricht im Kommunikationsnetz befindet, wird weitgehend bedeutungslos, wenn man Teilnehmerstationen (z. B. Personal Computer) Informationen nicht erst dann **anfordern** läßt, wenn der Teilnehmer sie benötigt, sondern **zu einem beliebigen Zeitpunkt vorher** [Pfi1_83 Seite 67, 68, Pfit_84]:

> *Will man eine Zeitung lesen, so kann die Teilnehmerstation sie bereits zum Erscheinungszeitpunkt oder einem Zeitpunkt mit besonders geringen Übertragungskosten (z. B. Nachttarif) anfordern und zum späteren Lesen abspeichern.*

Diese Schutzmaßnahme verhindert, daß der Netzbetreiber allein aus dem Vermittlungsdatum „Zeit" und Kontextwissen schließen kann, wer mit wem kommuniziert:

> *Wenn tagsüber eine Zeitung angefordert wird und 10 Leute als Anforderer in Frage kommen, von denen bekanntlich 9 in Tag- und einer in Nachtschicht arbeiten, wäre ohne zeitliche Entkopplung sehr wahrscheinlich, daß sie der Nachtarbeiter liest.*

Entsprechend kann die Teilnehmerstation Information **zu einem beliebigen Zeitpunkt nach ihrer Entstehung senden**.

Kann der Netzbetreiber auf andere Art erkennen, wer mit wem kommuniziert, so erschwert diese Maßnahme doch zumindest das Erstellen von Persönlichkeitsbildern über den Tagesablauf.

Natürlich greift diese Maßnahme nicht bei Kommunikationsformen mit Realzeitanforderungen, z. B. Telefon. Bei allen anderen erschwert sie aber die Verkettung vom zeitlich entkoppelten Verkehrsereignis mit vom Teilnehmer später veranlaßten Folge-Verkehrsereignissen (vgl. Abschnitt 2.1.1).

2.4.3 Lokale Auswahl

Um Interessensdaten zu schützen, kann man **Information in großen Einheiten anfordern und lokal auswählen (lassen), was einen wirklich interessiert**:

> *Bestellt man sich statt eines bestimmten Zeitungsartikels mehrere Zeitungen verschiedener politischer Richtungen, so können aus der Bestellung keinerlei Rückschlüsse auf die politischen Interessen und Meinungen des Bestellers gezogen werden. Zusätzlich ließe sich durch „intelligente" Teilnehmerstationen die Dienstleistung sogenannter Ausschnittsbüros, die einem Kunden auf Wunsch zu einem bestimmten Thema Artikel aus mehreren Zeitungen zusammenstellen, jedem Teilnehmer bieten.*

Durch die lokale Auswahl verschleiert man selbst gegenüber dem Kommunikationspartner, was einen wirklich interessiert, wie man (unter anderem die genaue Bedienung der Teilnehmerstation) lernt und reagiert, sowie vieles mehr.

Das Anfordern von großen Informationseinheiten vom Kommunikationspartner ist folglich als Schutz bei solchen Informationsarten sinnvoll, die unverschlüsselt übertragen werden oder bei denen Netzbetreiber und Kommunikationspartner identisch sind oder als zusammenarbeitend angenommen werden können.

In zukünftigen Kommunikationsnetzen, in denen (zumindest für schmalbandiges Senden) reichlich Bandbreite zur Verfügung steht, verursacht diese Maßnahme bei Diensten, bei denen der Benutzer nur bereits bereitgestellte Information abruft, real beinahe keine Kosten. Es ist jedoch eine Frage des Abrechnungsmodus, ob dies auch dem Anwender dieser Maßnahme zugutekommt (vgl. Abschnitt 7.2). Selbst wenn kein akzeptabler Abrechnungsmodus gefunden werden kann, der es erlaubt, z. B. mehrere Zeitungen verschiedener politischer Richtungen und damit vermutlich auch aus verschiedenen Verlagen zum Preise einer zu beziehen, so sollten doch zumindest wie heutzutage z. B. ganze Zeitungen verkauft und übertragen werden. Eine auf kleine Bildschirmseiten orientierte Struktur wie die des Bildschirmtext-Systems, die damit begründet wurde, die übliche Teilnehmerstation müsse billig und deshalb *zwangsweise* ohne lokale Verarbeitungs- und Speicherfähigkeit realisiert werden, ist zumindest für die Zukunft technisch obsolet und könnte nur damit gerechtfertigt werden, weiterhin das zur Teilnehmerbeobachtung technisch optimal geeignete System behalten zu wollen.

Lokale Auswahl wurde (mindestens) dreimal unabhängig voneinander erfunden: zuerst wird sie bezüglich *eines* Informationsanbieters und ohne generelle Einordnung lediglich im Kontext

des Bildschirmtext-Systems in [BGJK_81 Seite 153, 159] vorgeschlagen, danach in [Pfit_83] und schließlich zusammen mit zeitlich entkoppelter Verarbeitung in [GiLB_85].

2.5 Grundverfahren innerhalb des Kommunikationsnetzes zum Schutz der Verkehrs- und Interessensdaten

Im Gegensatz zu den in Abschnitt 2.4 genannten Schutzmaßnahmen sollen in diesem Abschnitt Maßnahmen vorgestellt werden, die die im Kommunikationsnetz anfallenden Vermittlungsdaten (vgl. Abschnitt 1.2) selbst verringern.

Dadurch soll für den Teilnehmer einerseits ohne die Netzbenutzung für ihn zu ändern oder gar komplizierter zu machen und andererseits unter Beibehaltung der Möglichkeit, den erhaltenen Datenschutz selbst zu überprüfen, Senden und Empfangen, zumindest aber die Kommunikationsbeziehungen zwischen Teilnehmern vor möglichen Angreifern (bösen Nachbarn, dem Netzbetreiber, den Herstellern der verwendten Soft- und Hardware, großen Organisationen und Konzernen usw.) verborgen werden, so daß das Erfassen von Verkehrs- oder Interessensdaten unmöglich wird.

Da Schutz vor einem allmächtigen Angreifer, der alle Leitungen, alle Vermittlungszentralen und alle Teilnehmerstationen außer der eigenen kontrolliert, nicht möglich ist, sind alle folgenden Maßnahmen nur Annäherungen an den perfekten Schutz der Teilnehmer vor jedem möglichen Angreifer. Die Annäherung wird – wie schon in Abschnitt 2.1.1 erwähnt – im allgemeinen durch Angabe des unterstellten *Angreifermodelles* (Was kann er maximal beobachten oder gar aktiv kontrollieren; mit wieviel Rechenkapazität ist er ausgestattet?) beschrieben.

Verbirgt ein Kommunikationsnetz für alle Kommunikationsereignisse wer *sendet* oder *empfängt* oder zumindest die *Kommunikationsbeziehung* vor dem unterstellten Angreifer, so wird es *anonym* genannt. Es wird dann von **anonymer Kommunikation** gesprochen.

Die meisten der heute bekannten und in den folgenden Unterabschnitten beschriebenen Grundverfahren innerhalb des Kommunikationsnetzes zum Schutz der Verkehrs- und Interessensdaten schützen für alle realistischerweise in Betracht gezogenen Angreifer entweder die Kommunikationsbeziehung oder wer sendet oder wer empfängt. Gegen wesentlich schwächere Angreifer bieten diese Grundverfahren vollständigen, d. h. es ist Sender und Empfänger und damit auch die Kommunikationsbeziehung geschützt, und gegen wesentlich stärkere Angreifer keinen Schutz. Wegen dieser kanonischen, d. h. von den Details einer realistischen Angreiferdefinition unabhängigen Zuordnung der Grundverfahren zu Schutzzielen sind die Unterabschnitte entsprechend diesen drei Möglichkeiten gegliedert.

2.5.1 Schutz des Empfängers (Verteilung)

Indem das Kommunikationsnetz alle Informationen an alle Teilnehmer sendet (**Verteilung**, broadcast), kann man den Empfänger der Information vor dem Kommunikationsnetz und dem Kommunikationspartner schützen [FaLa_75]. Verteilung ermöglicht das Analogon zu perfekter informationstheoretischer Konzelation (vgl. Abschnitt 2.2.2.2), nämlich *perfekte informationstheoretische Anonymität* des Empfängers. Dies bedeutet nach der Definition von perfekter Anonymität in Abschnitt 2.1.1 und dem in Abschnitt 2.2.2.2 über die informationstheoretische Modellwelt Gesagtem, daß für einen alle Leitungen und ggf. einige Stationen kontrollierenden Angreifer für alle Nachrichten die Wahrscheinlichkeit, daß eine Station die Nachricht empfangen hat, für alle von ihm nicht kontrollierten Stationen vor und nach seiner Beobachtung gleich ist. Der Angreifer gewinnt durch seine Beobachtung also keine Information darüber, welche von ihm unkontrollierte Station welche Nachricht empfängt. Es ist für den Angreifer sogar nicht einmal beobachtbar, ob die Nachricht überhaupt empfangen wird. Verteilung garantiert also auch *perfekte informationstheoretische Unbeobachtbarkeit* des Empfangens. Ebenso verkettet Verteilung keine Verkehrsereignisse, so daß auch *perfekte informationstheoretische Unverkettbarkeit* gegeben ist.

Während die Realisierung von Verteilung keinerlei konzeptionelles Problem darstellt, solange nur *passive Angriffe* betrachtet werden, ist dies anderenfalls nicht der Fall.

1) Ein aktiver Angreifer kann die *Diensterbringung verhindern.*
   Hiergegen helfen die üblichen Fehlertoleranztechniken gegebenenfalls kombiniert mit einem Wechsel der Betreiberschaft, falls „zufällig“ häufig alle redundanten Subsysteme ausfallen.
2) Ein aktiver Angreifer kann *durch Störung der Konsistenz der Verteilung manche Empfänger zumindest teilweise deanonymisieren*.
   Ist der aktive Angreifer beispielsweise der Sender einer Nachricht *N*, auf die der Empfänger antworten wird, und möchte er den Empfänger deanonymisieren, so kann er dies bei unsicher implementierter Verteilung folgendermaßen tun: Er sendet *N* und verhindert die Verteilung von *N* an alle anderen Teilnehmerstationen außer einer. Dann wartet er, bis der Empfänger geantwortet haben müßte. Erhält er keine Antwort, so wiederholt der seinen Angriff, indem er eine andere Teilnehmerstation *N* empfangen läßt. Ist auf *N* nicht nur *eine* Antwort zu erwarten, sondern ist dieser einfachstmögliche Dialog in einen längeren eingebettet, so kann die Länge dieses Angriffs, bei dem durchschnittlich die Hälfte aller anderen Teilnehmerstationen durchprobiert werden muß, auf den Logarithmus der Anzahl der anderen Teilnehmerstationen gedrückt werden: die Verteilung wird nur für die Hälfte der anderen Teilnehmerstationen gestört. Erhält der Angreifer Antwort, halbiert er für den nächsten Schritt die Gruppe, die die Nachricht erhielt, anderenfalls halbiert er deren Komplement zum vorherigen Schritt [Waid_89, WaPf1_89].
   Diese Angriffsart kann entweder durch analoge oder digitale Maßnahmen vereitelt werden. Eine **analoge Maßnahme** könnte in der Verwendung eines Mediums bestehen, daß *Konsistenz der Verteilung garantiert* – viele schreiben *Funknetzen* diese Eigenschaft zu. Da solche Eigenschaften schwer nachgewiesen werden können (vgl. das in Abschnitt 2.1.2 über unmanipulierbare Gehäuse Gesagte) und Funknetze anfällig für Angriffe auf die Diensterbringung sowie in der verfügbaren Bandbreite beschränkt sind,

da das elektromagnetische Spektrum im wesentlichen nur einmal genutzt werden kann, ist es wichtig, digitale Maßnahmen zur Verfügung zu haben. **Digitale Maßnahmen** können *inkonsistente Verteilung natürlich nur erkennen* aber nicht verhindern, denn ein physisch genügend verbreiteter Angreifer kann die Signalausbreitung zu einzelnen Teilnehmerstationen immer durch Durchtrennen von Leitungen, Errichtung Faradayscher Käfige oder – für den Angreifer meist einfacher – Beschädigung der Funkantenne unterbinden. Die Erkennung inkonsistenter Verteilung kann dabei entweder *komplexitätstheoretisch* realisiert werden, indem Nachrichten mit Sequenznummern, Uhrzeit etc. versehen und von einer als bezüglich eines Angriffes auf die Verteilung als nichtkooperierend angenommenen Gruppe von Stationen unterschrieben werden [PfPW_89]. Oder sie kann sogar *informationstheoretisch* erreicht werden, indem das in Abschnitt 2.5.3.1 beschriebene überlagernde Senden durch geeignetes Einbeziehen der verteilten Nachrichten in die dortige Schlüsselgenerierung erweitert wird [Waid_89, WaPf1_89].

Will man Verteilung auch bei bisher vermittelten Diensten einsetzen, so muß jede Teilnehmerstation anhand eines **implizite Adresse** genannten Merkmals entscheiden können, welche Nachrichten wirklich für sie bestimmt sind. Im Gegensatz zu einer **expliziten Adresse** enthält eine implizite keinerlei Information, wo sich der Empfänger befindet und ggf. auf welchem Weg er zu erreichen ist.

Kann eine Adresse nur vom Empfänger ausgewertet werden, so spricht man von **verdeckter Adressierung**. Kann eine Adresse von jedem ausgewertet, d. h. auf Gleichheit mit anderen Adressen getestet werden, so nennt man dies **offene Adressierung** [Waid_85]. Bei offener Adressierung sind Nachrichten prinzipiell immer dann, wenn gleiche Adressen auftreten, über ihre Adresse, d. h. den Empfänger *verkettbar*.

Die übliche Implementierung von verdeckter impliziter Adressierung verwendet Redundanz innerhalb des Nachrichteninhalts und ein *asymmetrisches Konzelationssystem* (vgl. Abschnitt 2.2.1.2.1). Jede Nachricht wird ganz oder teilweise mit dem Chiffrierschlüssel des adressierten Teilnehmers verschlüsselt (wodurch bei der ersten Möglichkeit gleichzeitig Ende-zu-Ende-verschlüsselt wird). Nach der Entschlüsselung mit dem zugehörigen Dechiffrierschlüssel kann die Teilnehmerstation des adressierten Teilnehmers anhand der Redundanz feststellen, daß die Nachricht für sie bestimmt ist. Da die Implementierungen von asymmetrischen Konzelationssystemen aufwendig und damit bei begrenztem Implementierungsaufwand langsam sind (vgl. Abschnitt 2.2.2.3) und jede Teilnehmerstation alle Nachrichten entschlüsseln muß, ist dieses Verfahren im allgemeinen viel zu aufwendig [Ken1_81]. Es kann durch Austausch eines geheimen Schlüssels und Verwendung eines schnelleren *symmetrischen Kryptosystems* [Karg_77 Seite 111, 112] (statt eines asymmetrisches Konzelationssystems) nach Aufnahme der Kommunikationsbeziehung jedoch etwas effizienter gestaltet werden: Beginnt jede Nachricht mit einem Bit, das angibt, ob das asymmetrische Konzelations- oder symmetrische Kryptosystem zur Adressierung verwendet wurde, so kann durch Verwendung von Letzterem bei langandauernden Kommunikationsbeziehungen asymptotisch soviel Aufwand gespart werden, wie das symmetrische Kryptosystem effizienter als das asymmetrische Konzelationssystem implementierbar ist. Die Unverkettbarkeit von Nachrichten über die verdeckte implizite Adressierung ist natürlich nur so sicher, wie das unsicherste zur Adressierung verwendete Kryptosystem (bei

der gegeben Öffentlichkeit der Schlüssel, vgl. übernächster Unterabschnitt über öffentliche und private Adressen).

Daß die verdeckte Adressierung der ersten Nachricht zwischen zwei Parteien, die noch keinen geheimen Schlüssel in Konzelation und Integrität garantierender Weise ausgetauscht haben, nicht unverkettbarer und in Bezug auf die verwendete Adresse anonymer (beides also bestenfalls in einer komplexitätstheoretischen Modellwelt beweisbar, vgl. Abschnitt 2.2.2.2) als mit dem sichersten asymmetrischen Konzelationssystem möglich ist, kann man sich wie folgt überlegen: Mit jeder Möglichkeit zur verdeckten Adressierung kann man die Funktion eines asymmetrischen Konzelationssystems erbringen, indem die ein Geheimnis übermitteln wollende Partei der anderen die Nachricht *„Die geheime Nachricht ist in der Form codiert, daß jede an dich adressierte Nachricht eine 1, jede nicht an dich adressierte Nachricht eine 0 bedeutet. <adressierte Nachricht 1>, <adressierte Nachricht 2>, ... <adressierte Nachricht n>“* unverschlüsselt zusendet [Pfi1_85 Seite 9, PfWa_86]. Dieses auf Empfänger-Anonymität basierende Konzelations-Protokoll entspricht dem auf Sender-Anonymität basierenden Konzelations-Protokoll von Alpern und Schneider [AlSc_83].

Wird in diesem Beweis unterstellt, daß beide Parteien einen geheimen Schlüssel in Konzelation und Integrität garantierender Weise ausgetauscht haben, so folgt, daß mit jeder Möglichkeit zur verdeckten Adressierung die Funktion eines symmetrischen Kryptosystems erbracht werden kann. Beide Beweise zusammen und die oben beschriebenen Implementierungen verdeckter impliziter Adressierung mittels Konzelationssystemen ergeben, daß verdeckte Adressierung und Konzelation sowohl im asymmetrischen wie im symmetrischen Fall äquivalent sind.

Aus den Beweisen resultieren in natürlicher Weise Schranken für die mögliche Effizienz, d. h. Konzelationsysteme und verdeckte Adressierung können sowohl im asymmetrischen wie auch im symmetrischen Fall jeweils „ähnlich“ effizient sein.

Offene Adressierung läßt sich einfacher realisieren (als verdeckte), indem man z. B. Nachrichten mit einem Adreßfeld versieht und die Teilnehmer(stationen) beliebige Zahlen als Adressen erzeugen. Eine Teilnehmerstation muß dann nur bei allen erhaltenen Nachrichten dieses Adreßfeld mit ihren Adressen vergleichen. Dies kann sie z. B. mittels eines *Assoziativspeichers*, in den sie alle ihre Adressen schreibt und an den dann alle Adreßfelder angelegt werden, sehr schnell und effizient tun, selbst wenn sie (momentan) eine große Zahl von verschiedenen Adressen besitzt. Adreßerkennung ist bei offener Adressierung also wesentlich effizienter als bei verdeckter möglich.

Hinsichtlich der Adreßverwaltung kann man bei beiden Formen der impliziten Adressierung öffentliche und private Adressen unterscheiden: **Öffentliche Adressen** stehen in allgemein zugänglichen Adreßverzeichnissen (z. B. in einem „Telefonbuch“) und dienen meist einer ersten Kontaktaufnahme. **Private Adressen** werden an einzelne Kommunikationspartner gegeben. Dies kann entweder außerhalb des Kommunikationsnetzes oder innerhalb als Absenderangabe in Nachrichten geschehen. Es sei angemerkt, daß die Kategorien „öffentliche Adresse“ und „private Adresse“ noch nichts über den Personenbezug aussagen: Öffentliche Adressen können (zumindest vor ihrem Gebrauch) informationstheoretisch anonym vor allen Instanzen außer dem Adressaten sein. Private Adressen können für den Benutzer durchaus einer Person zuordenbar sein.

			Adreßverwaltung	
			öffentliche Adresse	private Adresse
Adressierungsart	implizite Adresse	verdeckt	sehr aufwendig, für Kontaktaufnahme nötig	aufwendig
		offen	abzuraten	nach Kontaktaufnahme ständig wechseln
	explizite Adresse		sehr abzuraten	sehr abzuraten

Bild 18: Bewertung der Kombinationen von Adressierungsart und Adreßverwaltung

Bei offener Adressierung, im Gegensatz zur verdeckten, kann das Netz – wie bereits oben erwähnt – prinzipiell Informationen über die Empfänger von Nachrichten gewinnen. Dies kann bei Verwendung privater Adressen geschehen, wenn man dieselbe Adresse mehrfach verwendet, weil dann selbst für ansonsten unverkettbare Nachrichten erkennbar wird, daß sie an denselben Empfänger gerichtet sind. Die mehrmalige Verwendung offener Adressen muß also durch fortlaufendes Generieren und Mitübertragen oder durch Vereinbarung eines Generieralgorithmus vermieden werden. Der Generieralgorithmus „Verwende die nächsten *n* Zeichen eines rein zufällig generierten, nur einmal benutzten und in perfekte informationstheoretische Konzelation und Integrität garantierender Weise vorher verteilten Schlüssels“ erlaubt *perfekte informationstheoretische Unverkettbarkeit und Anonymität* vor Unbeteiligten. Mit den bis hierher geschilderten Verfahren ist dann allerdings aus organisatorischen Gründen keine wirkliche Anonymität zwischen den Beteiligten möglich (vgl. das in Abschnitt 2.2.2.2 über perfekte informationstheoretische Konzelation Gesagte). In Abschnitt 2.5.3.1.3 wird dann mit dem paarweisen überlagernden Empfangen eine auf Senderanonymität basierende Methode zum Austausch eines Schlüssels zwischen anonymen Beteiligten beschrieben. Ist die Senderanonymität informationstheoretisch, so erfolgt der Schlüsselaustausch (ebenfalls in der informationstheoretischen Modellwelt) in Konzelation und Integrität garantierender Weise. Da informationstheoretische Senderanonymität möglich, aber mit sehr, sehr großem Aufwand verbunden ist, ist das angedeutete Schlüsselaustausch-Verfahren zwar möglich, aber praktisch kaum anwendbar. Erzeugt man hingegen den nur einmal benutzten Schlüssel mit einem kryptographisch starken Pseudozufallsbitfolgengenerator (vgl. Abschnitt 2.2.2.2), so kann nur *perfekte komplexitätstheoretische Unverkettbarkeit und Anonymität* gewahrt werden – dafür ist dann Anonymität zwischen den Beteiligten auch praktisch kein Problem mehr, da der Startwert des Pseudozufallsbitfolgengenerators mit einem perfekten komplexitätstheoretischen asymmetrischen Konzelationssystem ausgetauscht werden kann.

Bei Verwendung öffentlicher offener Adressen kann sogar festgestellt werden, unter welcher Bezeichnung der Empfänger im Adreßverzeichnis eingetragen ist. Die Verwendung solcher Adressen sollte also möglichst vermieden werden.

Die Datenschutz- und Aufwandseigenschaften der Kombinationen von Adressierungsart und Adreßverwaltung sind in Bild 18 zusammengefaßt.

Durch Verteilung und geeignete implizite Adressierung kann der Empfänger einer Nachricht also vollständig geschützt werden.

Leider sind nicht alle Typen von Kommunikationsnetzen für diese Maßnahme gleichermaßen geeignet. Dies wird in den Abschnitten 2.6 und 3.2.1 ausführlich diskutiert.

2.5.2 Schutz der Kommunikationsbeziehung (MIX-Netz)

Statt zusätzlich zum Empfänger auch den Sender verborgen zu halten, kann man zunächst versuchen, nur deren Verbindung geheimzuhalten und so die Anonymität der Kommunikation herzustellen.

Diese Idee wird durch das Verfahren der **umcodierenden MIXe** verwirklicht [Chau_81, Cha1_84].

Bei diesem von David Chaum 1981 für elektronische Post vorgeschlagenen Verfahren werden Nachrichten von ihren Sendern nicht notwendigerweise auf dem kürzesten Weg zu ihren Empfängern geschickt, sondern über mehrere möglichst bezüglich ihres Entwurfes, ihrer Herstellung [Pfit_86 Seite 356] und ihres Betreibers [Chau_81, Cha1_84 Seite 99] *unabhängige* sowie Nachrichten *gleicher Länge puffernde*, *umcodierende* und *umsortierende* Zwischenstationen, sogenannte MIXe, geleitet. Jeder MIX codiert Nachrichten gleicher Länge um, d. h. ent- oder verschlüsselt sie unter Verwendung eines Konzelation garantierenden Kryptosystems, so daß ihre Wege über ihre Länge und Codierung, zusammen ihr äußeres Erscheinungsbild, nicht verfolgt werden können (vgl. die 3. Anforderung an ein symmetrisches Kryptosystem in Abschnitt 2.2.1.1 und die Definition eines perfekten komplexitätstheoretischen asymmetrischen Konzelationssystems in Abschnitt 2.2.2.2).

Damit dieses Verfolgen des Weges auch über zeitliche oder räumliche Zusammenhänge nicht möglich ist, muß jeder MIX jeweils mehrere Nachrichten gleicher Länge abwarten (oder ggf. auch selbst erzeugen oder von Teilnehmerstationen erzeugen lassen – sofern nichts Nützliches zu senden ist, eben bedeutungslose Nachrichten, die von der Teilnehmerstation des Empfängers oder bei geeigneter Verschlüsselungsstruktur und Adressierung bereits von einem der folgenden MIXe ignoriert werden), sie dazu puffern und nach der Umcodierung umsortiert, d. h. in anderer zeitlicher Reihenfolge bzw. auf anderen Leitungen ausgeben. Eine geeignete Reihenfolge ist etwa die alphabetische Ordnung der umcodierten Nachrichten. (Solch eine von vornherein vorgegebene Reihenfolge ist besser als eine zufällige, da so der verborgene Kanal (covert -, hidden channel) der „zufälligen“ Reihenfolge für ein eventuell im jeweiligen MIX vorhandenes Trojanisches Pferd geschlossen wird. Da auch die Anzahl der jeweils gleichzeitig zu mixenden Nachrichten sowie die Zeitverhältnisse exakt vorgegeben werden können und dem MIX das Erzeugen von Nachrichten verboten werden kann, braucht damit keinerlei Möglichkeit zu bestehen, über die ein eventuell vorhandenes Trojanisches Pferd einem nicht empfangsberechtigten Empfänger Information zukommen lassen kann [PoKl_78, Denn_82 Seite 281].)

Die zum Zwecke des Verbergens zeitlicher oder räumlicher Zusammenhänge zusammen gemixten, d. h. zusammen gepufferten (oder erzeugten), umcodierten und umsortiert (beispiels-

weise in alphabetischer Reihenfolge) ausgegebenen Nachrichten gleicher Länge werden als ein *Schub* (batch [Chau_81 Seite 85]) bezeichnet.

Zusätzlich zum Puffern, Umcodieren und Umsortieren von Nachrichten gleicher Länge muß jeder MIX darauf achten, daß *jede Nachricht nur einmal gemixt wird* – oder anders herum gesagt: Nachrichtenwiederholungen ignoriert werden. Eine Eingabe-Nachricht stellt genau dann eine Nachrichtenwiederholung dar, wenn sie schon einmal bearbeitet wurde und die jeweils zugehörigen Ausgabe-Nachrichten von Unbeteiligten verkettbar (beispielsweise gleich) wären. (Wann dies genau der Fall ist, hängt vom verwendeten Umcodierungsschema ab und wird deshalb zusammen mit den Umcodierungsschemata weiter unten diskutiert.)

Wird eine Nachricht innerhalb *eines* Schubes mehrfach bearbeitet, so entstehen über die Häufigkeiten der Eingabe- und Ausgabe-Nachrichten dieses Schubes unerwünschte Entsprechungen: einer Eingabe-Nachricht, die *n*-mal auftritt, entspricht eine Ausgabe-Nachricht, die ebenfalls *n*-mal auftritt. Treten alle Eingabe-Nachrichten eines Schubes verschieden häufig auf, so schützt das Umcodieren dieses Schubes also überhaupt nicht.

Wird eine Nachricht in *mehreren* Schüben bearbeitet, so können zwischen diesen Schüben nichtleere Durchschnitte (und Differenzen) gebildet werden, indem jeweils der Durchschnitt (die Differenz) der Eingabe- und Ausgabe-Nachrichten gebildet wird, wobei beim Durchschnitt (bei der Differenz) von letzteren im allgemeinen Fall statt Gleichheit Verkettbarkeit geprüft wird. Natürlich können diese beiden Operationen wiederum auf beliebige Operationsergebnisse angewendet werden. Bei Nachrichten, die in einem Durchschnitt (einer Differenz) der Kardinalität 1 liegen, ist die Entsprechung zwischen Eingabe- und Ausgabe-Nachricht klar – bezüglich ihnen trägt dieser MIX zum Schutz der Kommunikationsbeziehung nichts bei.

2.5.2.1 Grundsätzliche Überlegungen über Möglichkeiten und Grenzen des Umcodierens

Das **Ziel** dieses Verfahrens der umcodierenden MIXe ist, daß alle anderen Sender und Empfänger der zusammen in den Schüben der MIXe gemixten Nachrichten [Pfi1_85 Seite 11] oder alle MIXe, die von einer Nachricht durchlaufen wurden, [Chau_81 Seite 85] zusammenarbeiten müssen, um die Kommunikationsbeziehung gegen den Willen von Sender oder Empfänger aufzudecken. Solange beides nicht gegeben ist, sollte aus der Sicht des Angreifers jede von einem Sender während eines bestimmten Zeitintervalls gesendete Nachricht zu jeder von einem Empfänger empfangenen Nachricht entsprechender Länge gehören können – außer der Angreifer ist Sender *und* Empfänger der Nachricht. Dabei wird die Länge der Zeitintervalle durch die maximal tolerierbaren Verzögerungszeiten der Dienste bestimmt: Jede von einem Sender gesendete Nachricht kann natürlich erst nach ihrem Senden, muß aber vor Ablauf der maximal tolerierbaren Verzögerungszeit des Dienstes beim Empfänger eintreffen.

Dieses Ziel, das in der Diktion von Abschnitt 2.1.1 die Nachrichten und die Teilnehmer bezüglich ihres Sendens und Empfangens lediglich nach Zeitintervallen und Nachrichtenlängen in Klassen einteilt, bezüglich dieser Klassen und des beschriebenen Angreifermodells aber *Unbeobachtbarkeit der Kommunikationsbeziehung* vor Unbeteiligten und – sofern gewünscht –

Anonymität der Kommunikationspartner voreinander als auch *Unverkettbarkeit der Kommunikationsbeziehungen* garantiert, ist das Maximum des Erreichbaren:

Arbeiten alle anderen Sender und Empfänger von Nachrichten, die von einem MIX zusammen gepuffert, umcodiert und umsortiert ausgegeben wurden (d. h. in einem Schub gemixt wurden), zusammen, so kann der MIX von ihnen prinzipiell bezüglich der von einem anderen gesendeten Nachrichten überbrückt werden. Praktisch ist dies besonders schlimm, wenn es einem Angreifer möglich ist, von n+1 Nachrichten, die von MIXen zusammen gepuffert, umcodiert und umsortiert ausgegeben wurden (d. h. jeweils in einem Schub gemixt wurden), selbst n zu liefern. Aus diesen für einen einzelnen MIX in [Pfi1_85 Seite 11] gemachten Aussagen folgt, daß, wenn alle anderen Sender und Empfänger von Nachrichten gleicher Länge eines bestimmten Zeitintervalls zusammenarbeiten, sie einen weiteren Sender und Empfänger trotz der Benutzung beliebig vieler, vom Angreifer nicht kontrollierten MIXe, vollständig beobachten können.

Daß das Verfahren der umcodierenden MIXe, sofern alle von einer Nachricht durchlaufenen MIXe zusammenarbeiten, für diese Nachricht nichts nützt, ist trivial.

Aus dem oben formulierten Ziel folgt, daß *das Umcodieren einer Nachricht (bzw., wie später erläutert wird, eines Nachrichtenteils) relevanten Inhalts nie aus Ver- gefolgt von Entschlüsseln besteht, sofern überflüssiges Umcodieren unterlassen wird und das verwendete Kryptosystem nur die in Abschnitt 2.2 genannten Eigenschaften besitzt*: Gibt es eine Ver- gefolgt von einer Entschlüsselung und besitzt das verwendete Kryptosystem nur die in Abschnitt 2.2 genannten Eigenschaften, muß dabei die Entschlüsselung mit dem zur Verschlüsselung passenden Schlüssel geschehen, da anderenfalls die Nachricht nie mehr verstanden werden kann. Dies bedeutet, daß sie in diesen beiden MIXen in gleicher Gestalt vorliegt, so daß diese beiden MIXe die Kommunikationsbeziehung dieser Nachricht genausogut ohne diese Ver- und Entschlüsselung wie mit ihr aufdecken können – dieses Ver- und Entschlüsseln ist also für den Schutz der Kommunikationsbeziehung im Sinne des Erreichens des gerade formulierten Zieles ohne Belang, zumal sich dadurch die Beteiligung von Sendern und Empfängern nicht ändert.

Aus dem oben formulierten Ziel folgt ebenfalls, daß *alle Nachrichten gleicher Länge des betrachteten Zeitintervalls die MIXe jeweils gleichzeitig (und deshalb auch in gleicher Reihenfolge) durchlaufen müssen*: Durchlaufen nicht alle Nachrichten gleicher Länge des betrachteten Zeitintervalls die MIXe jeweils gleichzeitig, so gibt es einen MIX *M*, den zwei Nachrichten *N1* und *N2* nicht gleichzeitig durchlaufen. Arbeiten nun alle anderen MIXe zusammen, so können sie dies beobachten und damit die zu *N1* bzw. *N2* gehörigen Sender und Empfänger jeweils unterscheiden.

Durchlaufen alle Nachrichten gleicher Länge des betrachteten Zeitintervalls die MIXe jeweils gleichzeitig und sendet und empfängt jede Teilnehmerstation in jedem Zeitintervall mindestens eine Nachricht jeder Länge, wird das *Ziel* des Verfahrens der umcodierenden MIXe sogar *ohne Klasseneinteilung der Teilnehmer* erreicht.

Sendet und empfängt überdies jede Teilnehmerstation in jedem Zeitintervall jeweils die gleiche Anzahl Nachrichten jeder Länge (Teilnehmerstationen dürfen durchaus eine andere Anzahl Nachrichten senden als empfangen; verschiedene Teilnehmerstation auch jeweils verschieden

viele), so wird *perfekte komplexitätstheoretische Unbeobachtbarkeit der Kommunikationsbeziehung* vor Unbeteiligten und – sofern gewünscht – *perfekte komplexitätstheoretische Anonymität der Kommunikationspartner voreinander* als auch *perfekte komplexitätstheoretische Unverkettbarkeit der Kommunikationsbeziehungen* erreicht (vgl. Abschnitt 2.1.1).

Das Umcodieren der MIXe muß natürlich so gestaltet werden, daß der Empfänger die Nachricht trotzdem verstehen kann: sofern die Nachricht bei ihm verschlüsselt ankommt, muß er alle zur Entschlüsselung notwendigen **Schlüssel** sowie die richtige Reihenfolge ihrer Anwendung bereits **kennen** oder während des Entschlüsselungsprozesses selbst kennenlernen. Außerdem muß natürlich jeder MIX in der Lage sein, die von ihm erwartete Umcodierung durchzuführen, er muß also den Schlüssel zur Ver- oder Entschlüsselung entweder bereits kennen oder während des Prozesses der Umcodierung selbst kennenlernen. Eine verschlüsselnde Umcodierung einer Nachricht muß also letztlich vom Empfänger dem durchführenden MIX spezifiziert werden, eine entschlüsselnde Umcodierung einer Nachricht von ihrem Sender.

Außerdem dürfen die vom MIX zu verwendenden Schlüssel diesem nichts Wesentliches über die Identität des Senders oder Empfängers der Nachricht verraten. Es ist wohl akzeptabel, daß der erste MIX einer Nachricht ihren Sender kennt und, sofern keine Verteilung zum Schutz des Empfängers verwendet wird, der letzte MIX den Empfänger. Einem „mittleren“ MIX sollte der zu verwendende Schlüssel nichts über den Sender oder Empfänger verraten, so daß in diesem Fall die Verwendung eines statisch fest ausgetauschten geheimen Schlüssels und damit informationstheoretische Konzelation (vgl. Abschnitt 2.2.2.2 und 2.2.2.3) nicht möglich ist. (Wie bereits in Abschnitt 2.5.1 angemerkt, wird in Abschnitt 2.5.3.1.3 mit dem paarweisen überlagernden Empfangen eine auf Senderanonymität basierende Methode zum Austausch eines Schlüssels beschrieben. Ist die Senderanonymität informationstheoretisch, so erfolgt der Schlüsselaustausch – ebenfalls in der informationstheoretischen Modellwelt – in Konzelation und Integrität garantierender sowie Anonymität erhaltender Weise. Da informationstheoretische Senderanonymität möglich, aber mit sehr, sehr großem Aufwand verbunden ist, ist das angedeutete Schlüsselaustausch-Verfahren zwar möglich, aber praktisch kaum anwendbar.)

Also können Nachrichten an und damit auch Nachrichten von „mittleren“ MIXen (praktisch) nur in perfekte komplexitätstheoretische Konzelation garantierender Weise verschlüsselt werden, die durch sie erreichbare Anonymität ist also nur *komplexitätstheoretischer* Natur.

Nach diesen grundsätzlichen Überlegungen über Möglichkeiten und Grenzen des Umcodierens können nun konkrete Umcodierungsschemata systematisch hergeleitet werden. Zunächst soll dies mit Verallgemeinerungen der drei in [Chau_81] einfach angegebenen und dort bezüglich ihrer Grenzen und Notwendigkeit nicht genau diskutierten Umcodierungsschemata geschehen.

2.5.2.2 Senderanonymität

Gesucht sei ein Umcodierungsschema, das es einem Sender erlaubt, einem Empfänger eine Nachricht zukommen zu lassen, dabei aber vor ihm und allen (außer dem ersten) verwendeten MIXen anonym zu bleiben, sowie die Kommunikationsbeziehung zu verbergen, sofern nicht alle MIXe, die von der Nachricht durchlaufen werden, oder alle anderen Sender und Empfänger

von Nachrichten im gleichen Zeitintervall zusammenarbeiten. Nach dem vorher Gesagten kann der Sender keinen geheimen Schlüssel mit dem Empfänger und den MIXen, vor denen er anonym bleiben will, verwenden. Also muß bezüglich ihnen ein asymmetrisches Konzelationssystem verwendet werden und der Sender dazu ihre Chiffrierschlüssel kennen. Mit den zur Verschlüsselung geeigneten Schlüsseln kann er dann entweder die Nachricht direkt (*direktes Umcodierungsschema für Senderanonymität*) oder einen eine weitere Nachrichtenumcodierung spezifierenden geheimen Schlüssel (*indirektes Umcodierungsschema für Senderanonymität*, wird weiter unten und dort gleich in der allgemeineren Form eines *indirekten Umcodierungsschemas* behandelt) verschlüsseln.

Direktes Umcodierungsschema für Senderanonymität: Der Absender einer Nachricht verschlüsselt sie so mit öffentlich bekannten Chiffrierschlüsseln eines asymmetrischen Konzelationssystems (bzw. für den ersten MIX der zu durchlaufenden MIX-Folge ggf. mit einem mit ihm vereinbarten geheimen Schlüssel eines symmetrischen Kryptosystems), daß sie nacheinander von der von ihm gewählten Folge von MIXen mit deren zugehörigen geheimgehaltenen Dechiffrierschlüsseln (bzw. ggf. mit dem mit dem ersten MIX vereinbarten geheimen Schlüssel eines symmetrischen Kryptosystems) entschlüsselt werden muß. Dadurch ändert sich das Erscheinungsbild der Nachricht auf jedem Stück ihres Weges, so daß ihr Weg (außer wenn alle MIXe, die sie durchläuft, zusammenarbeiten oder der Sender kooperiert oder alle anderen Sender und Empfänger von Nachrichten im gleichen Zeitintervall (genauer: Schub) zusammenarbeiten) nicht verfolgt werden kann.

Sei $A_1,...,A_n$ die Folge der Adressen und $c_1,...,c_n$ die Folge der öffentlich bekannten Chiffrierschlüssel der vom Sender gewählten MIX-Folge $MIX_1,...,MIX_n$, wobei c_1 auch ein geheimer Schlüssel eines symmetrischen Kryptosystems sein kann. Sei A_{n+1} die Adresse des Empfängers, der zur Vereinfachung der Notation MIX_{n+1} genannt wird, und c_{n+1} sein Chiffrierschlüssel. Sei $z_1,...,z_n$ eine Folge zufälliger Bitketten (bit strings; die Bildung des deutschen Begriffes erfolgte in Analogie zu „Zeichenkette“). Ist c_1 ein geheimer Schlüssel eines symmetrischen Kryptosystems, so kann z_1 die leere Bitkette sein. Ist c_i ein Chiffrierschlüssel eines bereits von sich aus indeterministisch verschlüsselnden asymmetrischen Konzelationssystems, so kann z_i ebenfalls die leere Bitkette sein. Der Sender bildet die Nachrichten N_i, die MIX_i erhalten wird, ausgehend von der Nachricht N, die der Empfänger (MIX_{n+1}) erhalten soll:

$$N_{n+1} = c_{n+1}(N)$$
$$N_i = c_i(z_i,A_{i+1},N_{i+1}) \qquad \text{für } i=n,...,1$$

Der Sender sendet N_1 an MIX_1. Jeder MIX erhält nach der Entschlüsselung die Adresse des nächsten und die für diesen bestimmte Nachricht. Die Mitverschlüsselung zufälliger Bitketten ist bei Verwendung eines deterministischen asymmetrischen Kryptosystems nötig, da ein Angreifer ansonsten nicht nur – wie in Abschnitt 2.2.1.2.1 erwähnt – kurze Standardnachrichten erraten und mit dem öffentlich bekannten Chiffrierschlüssel testen kann, sondern in diesem Fall sogar (ganz ohne Raten) die gesamte Ausgabe des MIXes testen könnte.

Bei diesem direkten Umcodierungsschema für Senderanonymität ist das Umcodieren einer Nachricht (zumindest bei allen außer dem ersten MIX) vollständig durch den jeweils öffentlich bekannten Chiffrierschlüssel bestimmt. Um nicht über Nachrichtenhäufigkeiten Entsprechungen zwischen Ein- und Ausgabe dieser MIXe entstehen zu lassen, *bearbeitet jeder MIX, solange*

er sein Schlüsselpaar beibehält, mit dem zugehörigen geheimgehaltenen Dechiffrierschlüssel nur unterschiedliche Eingabe-Nachrichten. Erhält ein MIX während dieser Zeitspanne eine Eingabe-Nachricht mehrmals mit dem Auftrag, sie mit seinem geheimgehaltenen Dechiffrierschlüssel zu entschlüsseln, ignoriert er ihr wiederholtes Eintreffen.

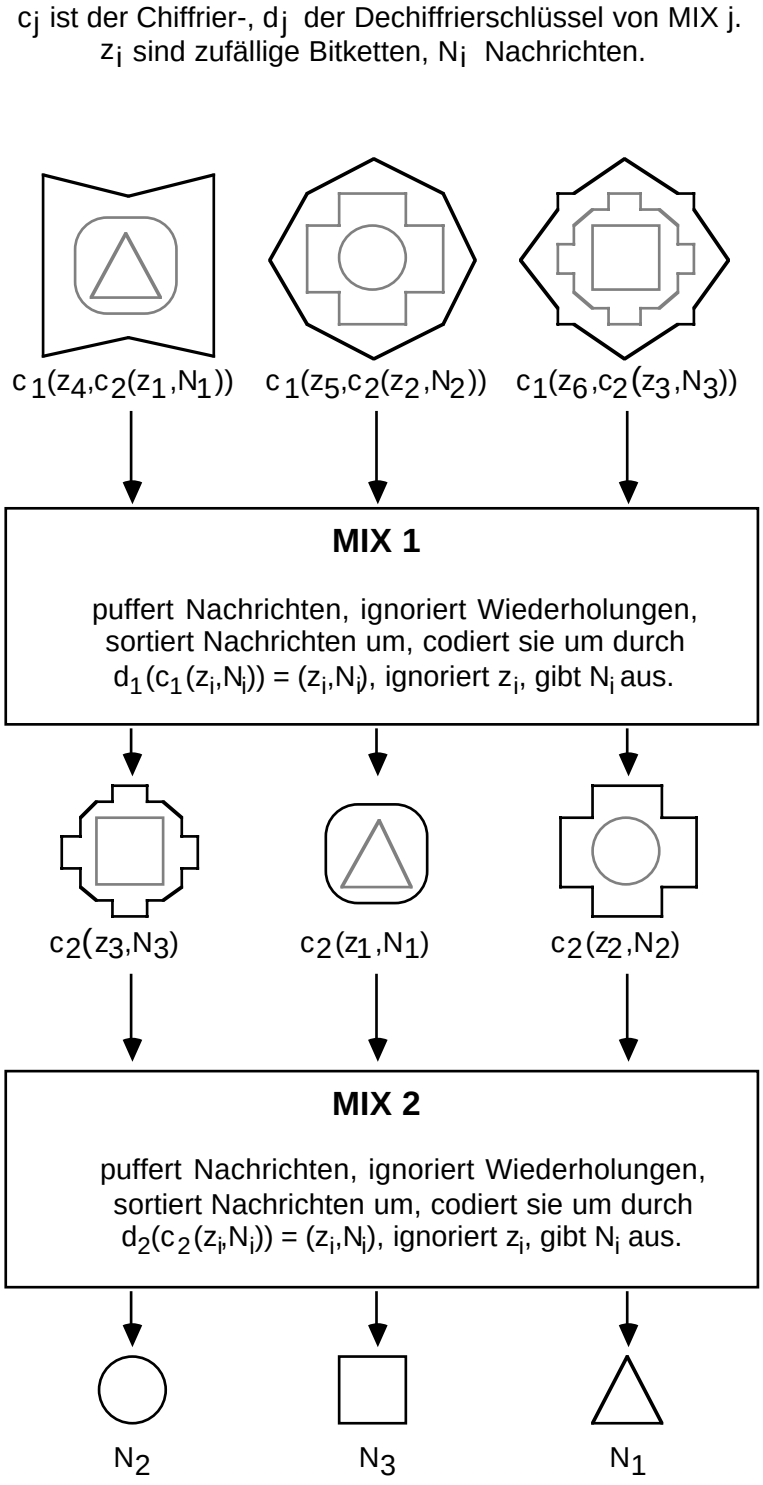


Bild 19: MIXe verbergen den Zusammenhang zwischen ein- und auslaufenden Nachrichten

Das bereits verbal beschriebene direkte Umcodierungsschema für Senderanonymität ist in Bild 19 anhand zweier MIXe noch einmal graphisch dargestellt. Allerdings wurden dort die Adressen weggelassen und die Indizes von Nachrichten und zufälligen Bitketten nur zu deren Unterscheidung benutzt, da bei Verwendung der Indizierungskonvention des rekursiven Bildungsschemas eine Doppelindizierung nötig geworden wäre.

In Bild 20 wird ein etwas größeres Beispiel (n=5) in einer graphischen Darstellung gezeigt, die betont, wer welche Schlüssel kennt und in welcher Reihenfolge Nachrichten verschlüsselt, transferiert und entschlüsselt.

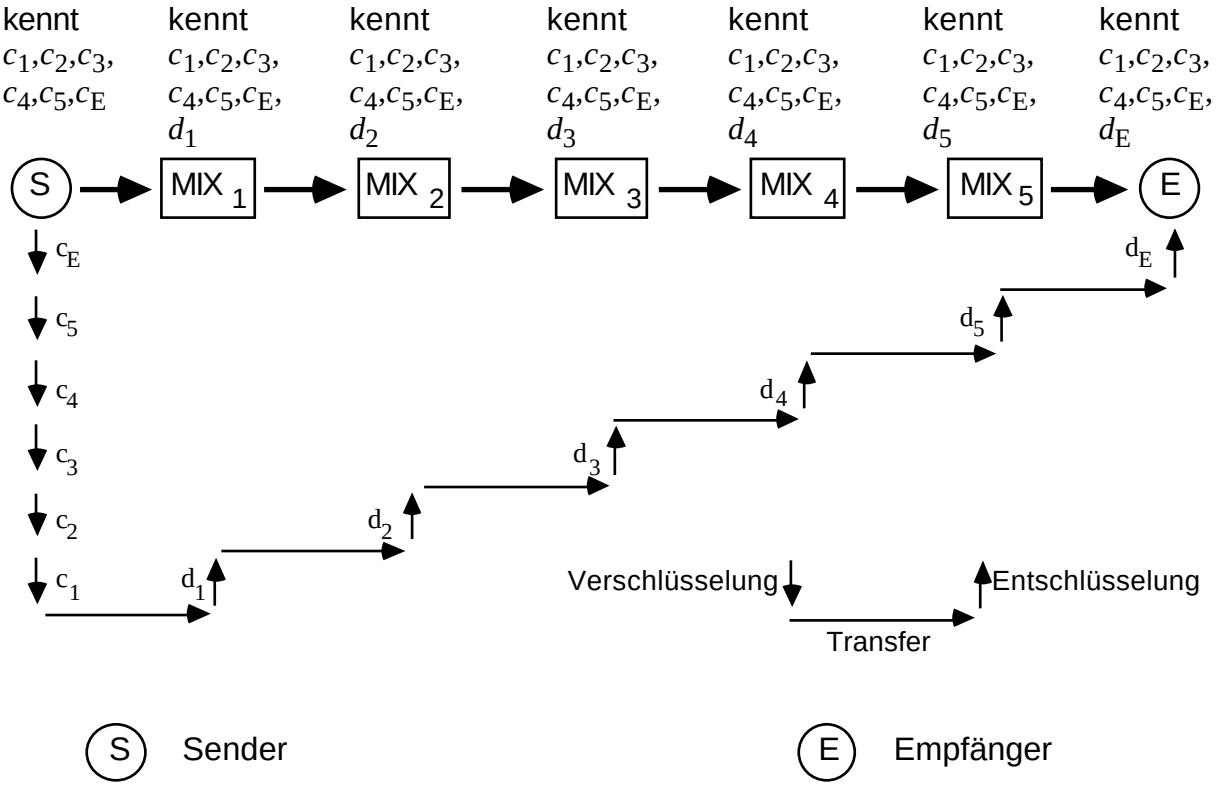


Bild 20: Transfer- und Verschlüsselungsstruktur der Nachrichten im MIX-Netz bei Verwendung eines direkten Umcodierungsschemas für Senderanonymität

2.5.2.3 Empfängeranonymität

Neben der Möglichkeit, eine Nachricht bezüglich einer Kommunikationsbeziehung anonym *senden* zu können, ist es zum Schutz der Kommunikationsbeziehung auch nötig, eine Nachricht bezüglich einer Kommunikationsbeziehung anonym *empfangen* zu können. Dies leistet das gerade beschriebene Umcodierungsschema lediglich in der Hinsicht nicht, daß der Sender die Nachricht N_{n+1} natürlich kennt und deshalb den Empfänger in einem normalen Vermittlungsnetz identifizieren kann, sofern

1. der Sender selbst oder jemand, der mit ihm zusammenarbeitet, die entsprechende *Leitung abhören* kann oder die Kooperation des Betreibers oder eines Herstellers des dem

MIX-Netz zugrundeliegenden Kommunikationsnetzes hat, oder in vielen Fällen wesentlich einfacher,

2. die Adresse A_{n+1} eine *öffentliche oder explizite Adresse* ist, die einen Personenbezug aufweist oder deren, meist an der Topologie des Kommunikationsnetzes orientiertes Bildungsschema entweder allgemein bekannt ist bzw. auch anderenfalls meist erschlossen werden kann oder dem Sender vom Betreiber oder einem Hersteller des Kommunikationsnetzes der zu dieser Adresse gehörige „Ort" mitgeteilt wird, oder
3. zusätzlich zu 1. oder 2. der Sender die *Kooperation von* MIX_n hat, was besonders wahrscheinlich ist, wenn er ihn auswählen kann.

Dies Problem kann nun natürlich bezüglich der 1. Bedrohung durch virtuelle Verbindungs-Verschlüsselung zwischen MIX_n und Empfänger [Pfi1_85 Seite 12, 14], bezüglich der 2. Bedrohung durch die Verwendung von impliziten privaten Adressen (beispielsweise könnte A_{n+1} eine zwischen MIX_n und dem Empfänger vereinbarte implizite private Adresse sein und von MIX_n durch eine explizite öffentliche des Kommunikationsnetzes ersetzt werden) und selbst bezüglich aller Bedrohungen mit dem in Abschnitt 2.5.1 beschriebenen Verfahren zum Schutz des Empfängers durch implizite Adressierung und Verteilung gelöst werden. Durch Letzteres wird sogar mehr als nur die Kommunikationsbeziehung geschützt. Je nach Struktur und Leistungsfähigkeit des zugrundeliegenden Kommunikationsnetzes kann die Verteilung „letzter" (d. h. direkt an den Empfänger gerichteter) Nachrichten die Nutzleistung des Kommunikationsnetzes jedoch in unakzeptabler Weise reduzieren, so daß eine bezüglich der Belastung des Kommunikationsnetzes schonendere Art des Schutzes des Empfängers bezüglich seiner Kommunikationsbeziehung vor dem Sender gesucht ist. Diese Art des Schutzes muß hinwiederum nicht unbedingt den Sender vor dem Empfänger schützen, denn sie könnte ja mit einem Umcodierungsschema für Senderanonymität kombiniert werden: der Sender sendet seine Nachricht unter Verwendung des Umcodierungsschemas für Senderanonymität an irgendeine Instanz *X*, beispielsweise einen MIX, mit der Bitte, sie auf die den Empfänger vor dem Sender schützende Art dem Empfänger zukommen zu lassen. Für den Schutz der Kommunikationsbeziehung zwischen Sender und Empfänger (genauer: ihre *gegenseitige Anonymität*) ist es nun völlig unkritisch, daß der Sender den Empfang seiner Nachricht durch *X* und der Empfänger das Senden seiner Nachricht durch *X* beobachten kann. Nachdem Instanz *X* ihren Zweck – eine einfachstmögliche Erklärung des Prinzips – erfüllt hat, wird sie nun wegrationalisiert: Die Konkatenation der Umcodierungsschemata kann leicht so modifiziert werden, daß der letzte MIX des Senderanonymitätsschemas die bisher von *X* gesendete Nachricht gleich an den ersten MIX des Empfängeranonymitätsschemas sendet.

Ignoriert man (in Analogie zum Umcodierungsschema für Senderanonymität) also den Schutz des Senders bezüglich einer Kommunikationsbeziehung vor dem Empfänger, so ist ein Umcodierungsschema gesucht, das es einem Empfänger erlaubt, von einem Sender einen Nachrichteninhalt zu erhalten, dabei aber vor ihm und allen (außer dem letzten) verwendeten MIXen anonym zu bleiben, sowie die Kommunikationsbeziehung zu verbergen, sofern nicht alle MIXe, die von der Nachricht durchlaufen werden, oder alle anderen Sender und Empfänger von Nachrichten im gleichen Zeitintervall zusammenarbeiten. Um vor dem Sender anonym zu bleiben, muß der Empfänger die Umcodierungen den durchlaufenen MIXen spezifizieren – nicht etwa wie im Umcodierungsschema für Senderanonymität der Sender. Bezüglich des vom Sender generierten Nachrichteninhalts muß es sich also um verschlüsselndes Umcodieren han-

deln, wozu jedem MIX der von ihm zu verwendende Schlüssel mitgeteilt werden muß, da nach dem vorher Gesagten der Empfänger keinen statisch ausgetauschten geheimen Schlüssel mit dem Sender und den MIXen, vor denen er anonym bleiben will, verwenden kann. Diese Schlüsselmitteilung geschieht am praktischsten dadurch, daß dem vom Sender generierten Nachrichteninhalt noch ein zweiter, *Rückadreßteil* genannter, Nachrichtenteil mitgegeben wird, der Teil einer vom Empfänger gebildeten sogenannten *anonymen Rückadresse* (untraceable return address, [Chau_81]) ist. Der Rückadreßteil muß von jedem MIX mit dem zu seinem öffentlich bekannten Chiffrierschlüssel gehörigen geheimgehaltenen Dechiffrierschlüssel entschlüsselt werden (ggf. kann bezüglich des letzten MIXes stattdessen auch ein zwischen ihm und dem Empfänger ausgetauschter geheimer Schlüssel eines symmetrischen Kryptosystems verwendet werden), um ihm als Inhalt nicht nur den Rückadreßteil für den nächsten MIX, sondern auch den von ihm für die Verschlüsselung der vom Sender generierten Nachricht zu verwendenden Schlüssel bekannt zu machen. Bei einem *Umcodierungsschema für Empfängeranonymität* muß es sich also um ein *indirektes* Umcodierungsschema handeln.

Indirektes Umcodierungsschema für Empfängeranonymität: Sei $A_1,...,A_m$ die Folge der Adressen und $c_1,...,c_m$ die Folge der öffentlich bekannten Chiffrierschlüssel der vom Empfänger gewählten MIX-Folge $MIX_1,...,MIX_m$, wobei c_m auch ein geheimer Schlüssel eines symmetrischen Kryptosystems sein kann. Die von der anonymen Rückadresse begleitete Nachricht wird diese MIXe in aufsteigender Reihenfolge ihrer Indizes durchlaufenen. Sei A_{m+1} die Adresse des Empfängers, der zur Vereinfachung der Notation MIX_{m+1} genannt wird. Ebenfalls zur Vereinfachung werde der Sender MIX_0 genannt. Der Empfänger einer Nachricht bildet eine **anonyme Rückadresse** (k_0,A_1,R_1), wobei k_0 ein für diesen Zweck generierter Schlüssel eines symmetrischen Kryptosystems (ein asymmetrisches Konzelationssystem wäre natürlich auch möglich, aber aufwendiger) ist, mit dem MIX_0 den Nachrichteninhalt verschlüsseln soll, damit MIX_1 ihn nicht lesen kann, und R_1 der Teil der anonymen Rückadresse ist, der von MIX_0 dem von ihm generierten und mit k_0 verschlüsselten Nachrichteninhalt mitgegeben wird. R_1 wird vom Empfänger ausgehend von einem zufällig gewählten eindeutigen Namen e der Rückadresse nach dem folgenden rekursiven Schema gebildet, bei dem R_j jeweils den Rückadreßteil bezeichnet, den MIX_j erhalten wird, und k_j jeweils den Schlüssel eines symmetrischen Kryptosystems (ein asymmetrisches Konzelationssystem wäre auch möglich, aber aufwendiger), mit dem MIX_j den den Nachrichteninhalt enthaltenden Nachrichtenteil verschlüsseln soll:

$$\begin{aligned} R_{m+1} &= e \\ R_j &= c_j(k_j,A_{j+1},R_{j+1}) \qquad \text{für } j=m,...,1 \end{aligned}$$

Diese Rückadreßteile R_j und der ggf. bereits mehrmals verschlüsselte, vom Sender generierte Nachrichteninhalt *I*, *Nachrichteninhaltsteil* I_j genannt, bilden zusammen die Nachrichten N_j, die jeweils von MIX_{j-1} gebildet und an MIX_j gesendet werden – nach dem folgenden rekursiven Schema also zuerst vom Sender MIX_0 und dann der Reihe nach von den durchlaufenen MIXen $MIX_1,...,MIX_m$:

$$\begin{aligned} N_1 &= R_1,I_1; \quad I_1 = k_0(I) \\ N_j &= R_j,I_j; \quad I_j = k_{j-1}(I_{j-1}) \qquad \text{für } j=2,...,m+1 \end{aligned}$$

Der Empfänger MIX_{m+1} erhält also $e,N_{m+1} = e,k_m(...k_1(k_0(I))...)$ und kann, da er dem eindeutigen Namen *e* der Rückadresse alle geheimen Schlüssel k_j (im Falle der Verwendung eines asymmetrischen Konzelationssystems: alle Dechiffrierschlüssel) in richtiger Reihenfolge zuordnen kann, ohne Probleme entschlüsseln und so den Nachrichteninhalt *I* gewinnen.

Die Mitverschlüsselung zufälliger Bitketten ist diesmal auch bei Verwendung von deterministischen Kryptosystemen nicht nötig, da die mit öffentlich bekannten Chiffrierschlüsseln der MIXe verschlüsselten Nachrichtenteile R_j mit dem nicht ausgegebenen k_j genügend dem Angreifer nicht bekannte Information enthalten und bezüglich des Umcodierens mit dem Angreifer unbekannten Schlüsseln diesem sowieso kein Testen möglich ist.

Bei diesem indirekten Umcodierungsschema für Empfängeranonymität ist nur das Umcodieren des Rückadreßteils (zumindest bei allen außer dem letzten MIX) vollständig durch den jeweils öffentlich bekannten Chiffrierschlüssel bestimmt. Um nicht über Nachrichtenhäufigkeiten Entsprechungen zwischen Ein- und Ausgabe dieser MIXe entstehen zu lassen, *bearbeitet jeder MIX (solange er sein Schlüsselpaar beibehält) nur unterschiedliche Rückadreßteile*. Eine Wiederholung des Nachrichteninhaltsteils I_j ist unkritisch, sofern zum Umcodieren jeweils ein unterschiedlicher Schlüssel verwendet wird. Dies ist, da jede anonyme Rückadresse nur einmal verwendet werden kann, immer dann der Fall, wenn beim Bilden anonymer Rückadressen jeweils neue Schlüssel k_j verwendet werden. Beachtet ein Empfänger beim Bilden von anonymen Rückadressen diese Regel nicht, schadet er sich nur selbst. Verwendet ein MIX_j einen alten Schlüssel k_j zur Bildung einer Rückadresse, kann er dadurch dem Generierer von k_j auch nicht mehr schaden als durch Verraten des Nachrichtenzusammenhangs, der durch das ursprüngliche Umcodieren mit k_j verborgen werden sollte.

2.5.2.4 Gegenseitige Anonymität

Wird dieses Umcodierungsschema für Empfängeranonymität allein verwendet, so verbirgt es (wie das Umcodierungsschema für Senderanonymität auch) zwar die Kommunikationsbeziehung vor Unbeteiligten vollständig, da hier der Empfänger jedoch den Rückadreßteil R_1 kennt, kann er den Sender beobachten, sofern

1. der Empfänger selbst oder jemand, der mit ihm zusammenarbeitet, die entsprechende *Leitung abhören* kann oder die Kooperation des Betreibers oder eines Herstellers des dem MIX-Netz zugrundeliegenden Kommunikationsnetzes hat, oder in vielen Fällen wesentlich einfacher,
2. der Empfänger die *Kooperation von* MIX_1 hat, was besonders wahrscheinlich ist, wenn er ihn auswählen kann, und das dem MIX-Netz zugrundeliegende Kommunikationsnetz gesendeten Nachrichten *öffentliche oder explizite Absenderadressen* zuordnet (und den Netzbenutzern mitteilt – wie im geplanten ISDN vorgesehen), die einen Personenbezug aufweisen oder deren, meist an der Topologie des Kommunikationsnetzes orientiertes Bildungsschema entweder allgemein bekannt ist bzw. auch anderenfalls meist erschlossen werden kann oder dem Sender vom Betreiber oder einem Hersteller des Kommunikationsnetzes der zu dieser Adresse gehörige „Ort“ mitgeteilt wird, oder
3. zusätzlich zu 1. der Empfänger die Kooperation von MIX_1 hat, was – wie schon erwähnt – besonders wahrscheinlich ist, wenn er ihn auswählen kann.

Dies Problem kann nun natürlich auch bezüglich der 1. Bedrohung durch virtuelle Verbindungs-Verschlüsselung [Pfi1_85 Seite 12, 14], bezüglich der 2. Bedrohung durch geeignete Protokollgestaltung im dem MIX-Netz zugrundeliegende Kommunikationsnetz und bezüglich aller Bedrohungen mit den im folgenden Abschnitt 2.5.3 beschriebenen Verfahren zum Schutz des Senders gelöst werden. Durch Letzteres wird sogar mehr als nur die Kommunikationsbeziehung geschützt. Je nach Struktur und Leistungsfähigkeit des zugrundeliegenden Kommunikationsnetzes können diese Verfahren die Nutzleistung des Kommunikationsnetzes in unakzeptabler Weise reduzieren, so daß dann – sofern die Anwendung gegenseitige Anonymität fordert – nur die bereits beschriebene Kombination von einem Umcodierungsschema für Sender- und einem für Empfängeranonymität möglich ist [Pfi1_85 Seite 14f]. (Die Idee ist vermutlich auch in [Chau_81 Seite 85] enthalten, die dortige Formulierung in ihrer Allgemeinheit jedoch falsch.) Diese gegenseitige Anonymität muß natürlich auch bei der ersten Nachricht vorhanden sein, was durch den schon früher beschriebenen Indirektionsschritt über eine (hier tatsächlich nötige) zusätzliche Instanz *X* (bzw. zumindest Funktionalität, die natürlich auch von einer vorhandenen Instanz erbracht werden kann), hier ein **Adreßverzeichnis mit anonymen Rückadressen**, erreicht wird. Da jede anonyme Rückadresse nur einmal verwendet werden kann, ist die Verwendung gedruckter Adreßverzeichnisse sehr umständlich, die Verwendung elektronischer, die jede Adresse nur einmal ausgeben, jedoch kein Problem.

2.5.2.5 Längentreue Umcodierung

Nachdem nun die einfachstmöglichen Umcodierungsschemata für Sender- oder Empfängeranonymität hergeleitet und ihre Einsatzmöglichkeiten beschrieben wurden, soll noch auf eine weitere gemeinsame Eigenschaft hingewiesen werden: in beiden Umcodierungsschemata ändert sich beim Umcodieren die Länge von Nachrichten – sie werden kürzer. Dies ist, wenn alle Nachrichten die gleichen MIXe in gleicher Reihenfolge durchlaufen, was – wie zu Beginn dieses Abschnittes gezeigt – zum Erreichen des maximal möglichen Anonymitätszieles nötig ist, kein Problem. Nun können – wie in Abschnitt 3.2.2.4 gezeigt wird – vor allem Leistungs-, aber auch Kostengesichtspunkte verhindern, daß alle Nachrichten in allen MIXen umcodiert werden. In solchen MIX-Netzen können dann die abnehmenden Nachrichtenlängen einem Angreifer durchaus wertvolle Information liefern. Deshalb ist es wünschenswert, ein Umcodierungsschema zu haben, daß die Länge einer Nachricht beim Umcodieren gleich läßt, so daß ihr (außer ihren ursprünglichen Generierern) niemand ansehen kann, wie oft so bereits bzw. noch umcodiert wurde bzw. werden muß. Außerdem sollten – wie zu Beginn dieses Abschnittes gezeigt – alle Nachrichten gleiche Länge haben und auch durch das verwendete Umcodierungsschema nicht unterschieden werden. Gesucht wird also ein universelles *längentreues Umcodierungsschema*, das nach dem bei der Herleitung eines Umcodierungsschemas für Empfängeranonymität Gesagten zwangsläufig ein *indirektes* Umcodierungsschema sein muß.

Indirektes längentreues Umcodierungsschema: Zusätzlich zu den beim indirekten Umcodierungsschema für Empfängeranonymität bereits eingeführten Bezeichnungen bedeuten eckige Klammern Blockgrenzen. Ausgehend vom Sender (auch MIX_0 genannt) sollen der Reihe nach die MIXe MIX_1,...,MIX_m durchlaufen und schließlich der Empfänger (auch MIX_{m+1}

genannt) erreicht werden. Der Empfänger braucht bei Erhalt der Nachricht noch nicht zu wissen, daß er ihr Empfänger ist. Er bearbeitet sie zunächst wie alle vorherigen MIXe.

Jede Nachricht N_j besteht aus b Blöcken gleicher, auf das von den MIXen verwendete asymmetrische Konzelationssystem abgestimmter Länge und wird von MIX_{j-1} gebildet. Die ersten m+2-j Blöcke von N_j enthalten den (Rück-)Adreßteil R_j, die letzten b-m-1 Blöcke den Nachrichteninhalt. Jeder MIX_j entschlüsselt den ersten Block der Nachricht N_j mit seinem geheimgehaltenen Dechiffrierschlüssel d_j und findet als Ergebnis dieser Entschlüsselung einen Schlüssel k_j eines auf die Blocklänge abgestimmten symmetrischen Kryptosystems (ein asymmetrisches Konzelationssystem, praktischerweise dasselbe, das von den MIXen verwendet wird, ist auch möglich, aber aufwendiger) zum Umcodieren der übrigen Nachricht und die Adresse A_{j+1} des nächsten MIXes (oder Empfängers). Der Empfänger findet als Adresse des nächsten MIXes (oder Empfängers) seine eigene und erkennt daran, daß er der Empfänger dieser Nachricht ist. Dieser erste Block von N_j wird von den folgenden MIXen (bzw. dem Empfänger) nicht benötigt und deshalb von MIX_j weggeschmissen. Mit k_j codiert MIX_j die restlichen b-1 Blöcke um und hängt vor den ersten Block mit Nachrichteninhalt, um die Länge der Nachricht nicht zu ändern, einen Block zufälligen Inhalts Z_j (in [Chau_81 Seite 87] wird auch dieser Block mit k_j umcodiert, was, da er zufälligen Inhalt hat, überflüssig ist).

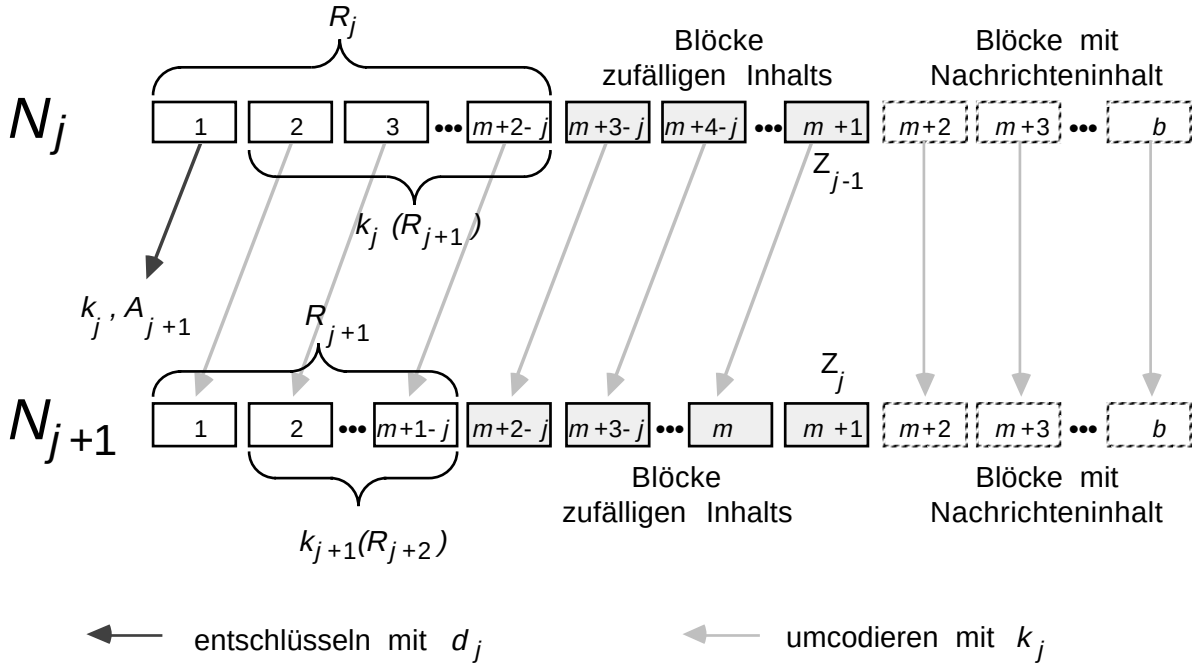


Bild 21: Indirektes längentreues Umcodierungsschema

Damit dies klappt, wird der vom Sender verwendete (Rück-)Adreßteil R_1 (ausgehend von einem zufällig gewählten eindeutigen Namen e) nach dem folgenden rekursiven Schema gebildet:

$$R_{m+1} = [e]$$
$$R_j = [c_j(k_j,A_{j+1})],k_j(R_{j+1}) \qquad \text{für } j=m,\ldots,1$$

Das Bildungsschema der Nachrichten N_j und insbesondere die Längentreue dieses rekursiven Umcodierungsschemas ist in Bild 21 veranschaulicht. Da MIX_j seinen Index j nicht zu kennen braucht, braucht er zwischen Blöcken des (Rück-)Adreßteils und Blöcken zufälligen Inhalts nicht unterscheiden zu können. Bezogen auf Bild 21 bedeutet dies, daß MIX_j lediglich von Block m+1 der Nachricht N_{j+1} weiß, daß er ein Block zufälligen Inhalts (im Bild: grau hinterlegt) ist. Da jedoch alle MIXe m erfahren müssen, um zu wissen, wo sie ihren Block zufälligen Inhalts einfügen sollen, kennen sie damit sowohl m als obere Schranke für die Anzahl der zu durchlaufenden MIXe als auch b-m-1 als obere Schranke für die Länge (in Blöcken) des Nachrichteninhalts.

Dem Bildungsschema des (Rück-)Adreßteils ist zu entnehmen, daß alle Blöcke des (Rück-)Adreßteils außer dem ersten von MIX_j mit k_j zu *entschlüsseln* sind. Ob die letzten, den Nachrichteninhalt enthaltenden b-m-1 Blöcke mit k_j zu *ent-* oder zu *verschlüsseln* sind, hängt davon ab, ob die Umcodierung durch MIX_j der Sender- oder Empfängeranonymität dient. Soll die Umcodierung mit k_j der Senderanonymität dienen, muß MIX_j entschlüsseln, da der Empfänger ansonsten den Schlüssel k_j auch erfahren müßte – dann wäre die Umcodierung als Schutz vor ihm aber nutzlos – oder den Nachrichteninhalt nicht verstehen könnte. Entsprechend muß verschlüsselt werden, soll die Umcodierung der Empfängeranonymität dienen. Statt in obiges rekursives Bildungsschema für die (Rück-)Adreßteile noch zusätzliche Bits aufzunehmen, die den MIXen anzeigen, ob sie die Blöcke mit Nachrichteninhalt jeweils ver- oder entschlüsseln sollen, wird im folgenden davon ausgegangen, daß diese Information Teil der Schlüssel k_j ist.

Damit MIX_j seinen Index j nicht zu kennen braucht, sollten die von ihm erhaltenen Blöcke zufälligen Inhalts genauso behandelt werden wie alle Blöcke (außer dem ersten) des (Rück-)Adreßteils R_j. Sie sollten also entschlüsselt werden. Dies wird formal dadurch ausgedrückt, daß die (Rück-)Adresse R_j und die Blöcke zufälligen Inhalts zum Nachrichtenkopf H_j (header) zusammengefaßt werden. Im folgenden bedeute $[H_j]_{x,y}$ die Blöcke x bis y (jeweils einschließlich) von H_j. Ist $x>y$, so bedeutet $[H_j]_{x,y}$ keinen Block.

$$\begin{aligned} H_1 &= R_1 \\ &= [c_1(k_1,A_2)],k_1(R_2) \end{aligned}$$

$$\begin{aligned} H_j &= k_{j-1}^{-1}([H_{j-1}]_{2,m+1}),[Z_j] \\ &= [c_j(k_j,A_{j+1})],k_j(R_{j+1}),k_{j-1}^{-1}([H_{j-1}]_{m+3-j,m+1}),[Z_j] \qquad \text{für } j=m,\ldots,2 \end{aligned}$$

Soll das indirekte längentreue Umcodierungsschema für **Senderanonymität** eingesetzt werden, so muß der Sender die Schlüssel k_1, k_2, ..., k_m generieren und den Chiffrierschlüssel c_{m+1} und die Adresse A_{m+1} des Empfängers kennen. Damit der Nachrichteninhalt vom Empfänger verstanden werden kann, muß ihn der Sender vor dem Absenden der Nachricht mit den von ihm generierten Schlüsseln verschlüsseln. Also wird die Nachricht N_1 vom Sender nach dem folgenden Schema ausgehend von dem in i=b-m-1 Blöcke zerlegten Nachrichteninhalt I =[I1],...,[Ii] (ist der Nachrichteninhalt zu kurz, muß er auf die passende Länge aufgefüllt werden) und dem Nachrichtenkopf H_1 gebildet:

$$N_1 = H_1,I_1; \qquad \begin{aligned} I_1 &= k_1(k_2(\ldots k_m(c_{m+1}(I_1))\ldots)) \\ &= k_1(k_2(\ldots k_m(c_{m+1}([I1]))\ldots)),\ldots,k_1(k_2(\ldots k_m(c_{m+1}([Ii]))\ldots)) \end{aligned}$$

$N_j = H_j,I_j;$ $\quad I_j = k_{j-1}^{-1}(I_{j-1});$ für j=2,...,m+1

Der aus jeweils m+2-j Blöcken bestehende (Rück-)Adreßteil R_j, j-1 Blöcke zufälligen, ggf. bereits mehrmals „entschlüsselten“ Inhalts sowie der ggf. bereits mehrmals entschlüsselte, vom Sender generierte und von ihm mit den Schlüsseln c_{m+1}, k_m, ... k_1 verschlüsselte Nachrichteninhaltsteil I_j bilden zusammen die Nachricht N_j. Aus ihr bildet MIX_j nach obigem rekursiven Schema die Nachricht N_{j+1}.

Bild 22 veranschaulicht in der von Bild 20 bekannten graphischen Darstellung den Einsatz des längentreuen Umcodierungsschemas für Senderanonymität am Beispiel m=5.

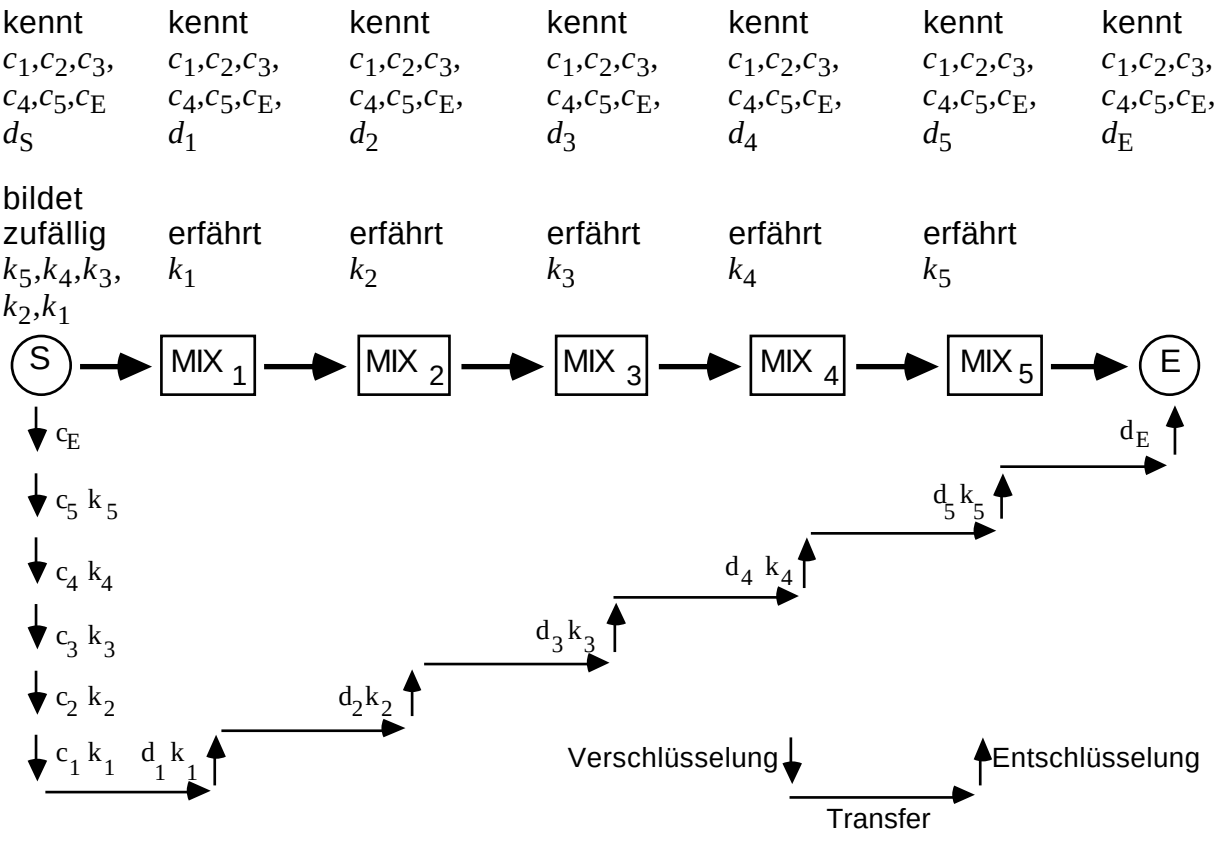


Bild 22: Indirektes Umcodierungsschema für Senderanonymität

Soll das indirekte längentreue Umcodierungsschema für **Empfängeranonymität** eingesetzt werden, so generiert der Empfänger die Schlüssel k_0, k_1, ..., k_m und die anonyme Rückadresse (k_0,A_1,R_1), die er dem Sender mitteilt. Der Sender bildet unter Verwendung von (k_0,A_1,R_1) die Nachricht N_1 sowie jeder MIX_j unter Verwendung von N_j die Nachricht N_{j+1} nach dem folgenden rekursiven Schema:

$N_1 = H_1,I_1;$ $\quad I_1 = k_0(I)$
$\quad = k_0([I_1]),...,k_0([I_i])$

$N_j = H_j,I_j;$ $\quad I_j = k_{j-1}(I_{j-1})$ $\quad$ für j=2,...,m+1

Nach Erhalt von N_{m+1} erkennt der Empfänger an dem eindeutigen Namen *e* die Rückadresse und kann mit den zugehörigen, ihm bekannten Schlüsseln k_m, k_{m-1}, ..., k_1, k_0 entschlüsseln und so den Klartext erhalten.

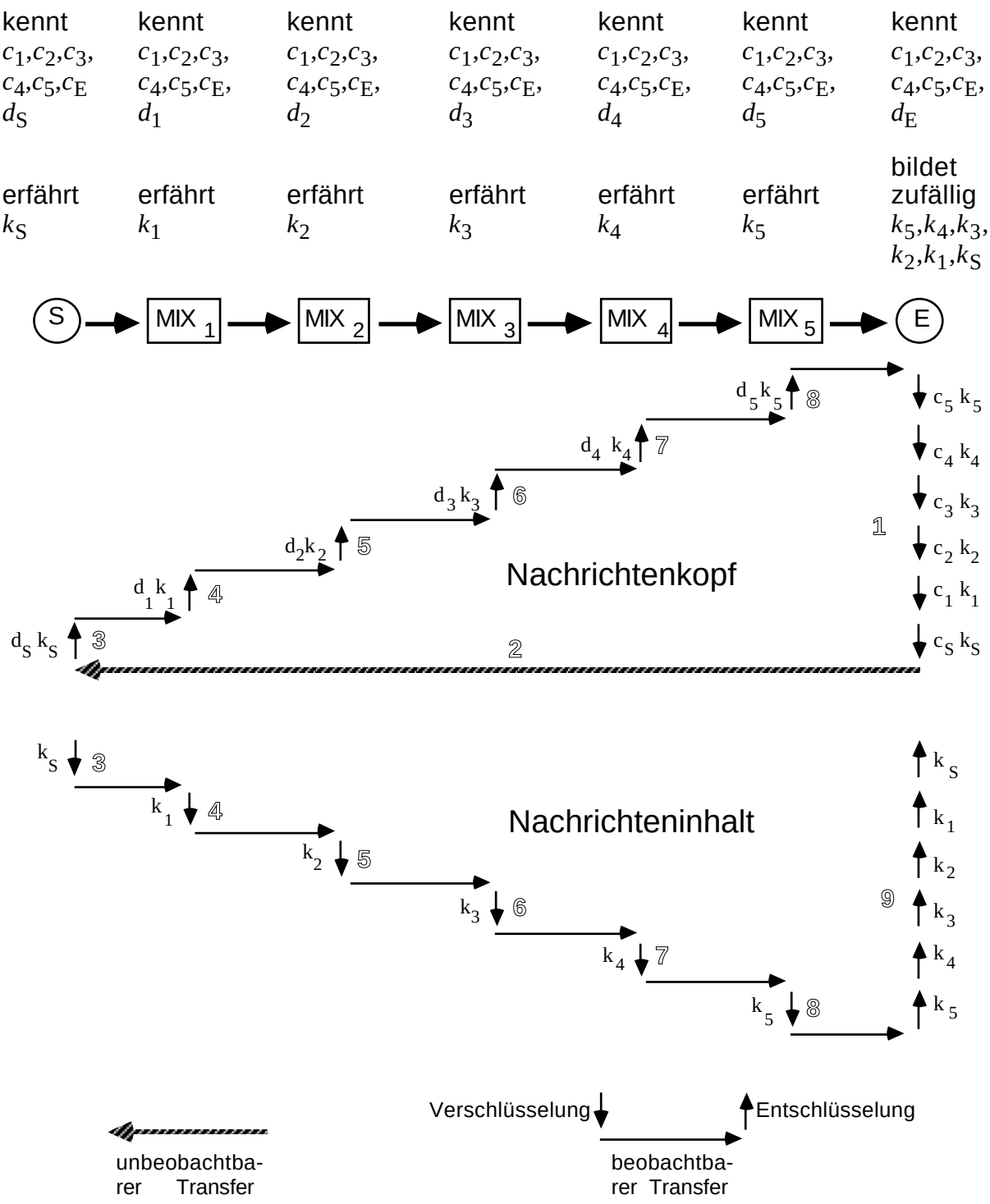


Bild 23: Indirektes Umcodierungsschema für Empfängeranonymität

Dies ist in Bild 23 für das Beispiel m=5 in der graphischen Darstellung von Bild 20 gezeigt. Nicht dargestellt ist, wie genau *E* dem Sender *S* die anonyme Rückadresse zukommen läßt. Die Zahlen in Konturschrift geben die Reihenfolge der Schritte an.

Sollen mit diesem indirekten längentreuen Umcodierungsschema **Sender und Empfänger** voreinander in überprüfbarer Weise **anonym** bleiben, müssen die Blöcke der (Rück-)Adresse von m bis zu einem gewissen Index g ($1<g\leq m$) vom Sender (sei es als (Rück-)Adresse in einem Adreßverzeichnis oder als Absender in der vorherigen Nachricht) und ab g-1 vom Empfänger gebildet werden. Der Sender erhält also R_g und bildet aus diesen m+2-g Blöcken die aus m+1 Blöcken bestehende (Rück-)Adresse R_1. Er verschlüsselt den Nachrichteninhalt mit dem ihm als Teil der Rückadresse mitgeteilten Schlüssel k_S und den g-1 von ihm generierten Schlüsseln. Der Empfänger muß nach Erhalt des mehrfach verschlüsselten Nachrichteninhaltes mit den Schlüsseln k_g, ..., k_m, k_S entschlüsseln.

Bild 24 verdeutlicht dies am Beispiel m=5 und g=4 in der graphischen Notation von Bild 20. Wie in Bild 23 geben die Zahlen in Konturschrift die Reihenfolge der Schritte an.

Sollen Nachrichten zwischen gegenseitig anonymen Partnern mit Absendern versehen sein, so fällt an dieser Stelle auf, daß von den i Nachrichteninhaltsblöcken mitnichten alle für den eigentlichen Nachrichteninhalt zur Verfügung stehen: Es werden m+2-g für den Absender benötigt – und dies in jeder Nachricht (wenn auch g für jede Nachricht unterschiedlich gewählt werden kann), denn genau wie beim indirekten Umcodierungsschema für Empfängeranonymität *darf jeder MIX (solange er sein Schlüsselpaar beibehält) auch bei dem indirekten längentreuen Umcodierungsschema nur unterschiedliche (Rück-)Adreßteile bearbeiten.*

Entsprechendes gilt für die umzucodierenden Blöcke, die zur *selben* Nachricht gehören und folglich mit demselben Schlüssel umcodiert werden: ist das verwendete Kryptosystem eine Blockchiffre, so dürfen keine zwei umzucodierenden Blöcke gleich sein. Entsprechendes gilt für eine selbstsynchronisierende Stromchiffre, vgl. Abschnitt 2.2.2.1. Lediglich bei Verwendung einer synchronen Stromchiffre brauchen MIXe keine diesbezüglichen Maßnahmen zu ergreifen. Das Gesagte gilt sinngemäß bei jedem indirekten Umcodierungsschema, ist hier aber, da explizit von „Blöcken“ geredet wird, besonders hervorzuheben. Dieser **aktive Angriff durch Wiederholung indirekt umzucodierender Nachrichtenteile** wurde in [Chau_81, Pfi1_85 Seite 14, 27] übersehen.

Außerdem sei noch einmal hervorgehoben, daß die durch die Schlüssel c_2 bis c_{m-1} spezifizierten Umcodierungen des (Rück-)Adreßteils natürlich nur komplexitätstheoretisch sicher sein können und die von den MIXen MIX_2 bis MIX_{m-1} erreichbare Anonymität der Kommunikationsbeziehung also auch nur komplexitätstheoretischer Natur sein kann. Die durch c_1 bzw. c_m spezifizierten Umcodierungen können jedoch durchaus mit informationstheoretischer Sicherheit erfolgen, indem Sender bzw. Empfänger mit MIX_1 bzw. MIX_m einen geheimen Schlüssel ausgetauscht haben. Sofern Sender oder Empfänger die Reihenfolge der MIXe frei wählen können, sollten sie diesen geheimen Schlüssel jeweils mit dem MIX austauschen, dem sie am meisten vertrauen. Wie bereits begründet schließt eine solche freie Wahl der MIX-Reihenfolge das Erreichen des maximalen Zieles des Verfahrens der umcodierenden MIXe jedoch aus.

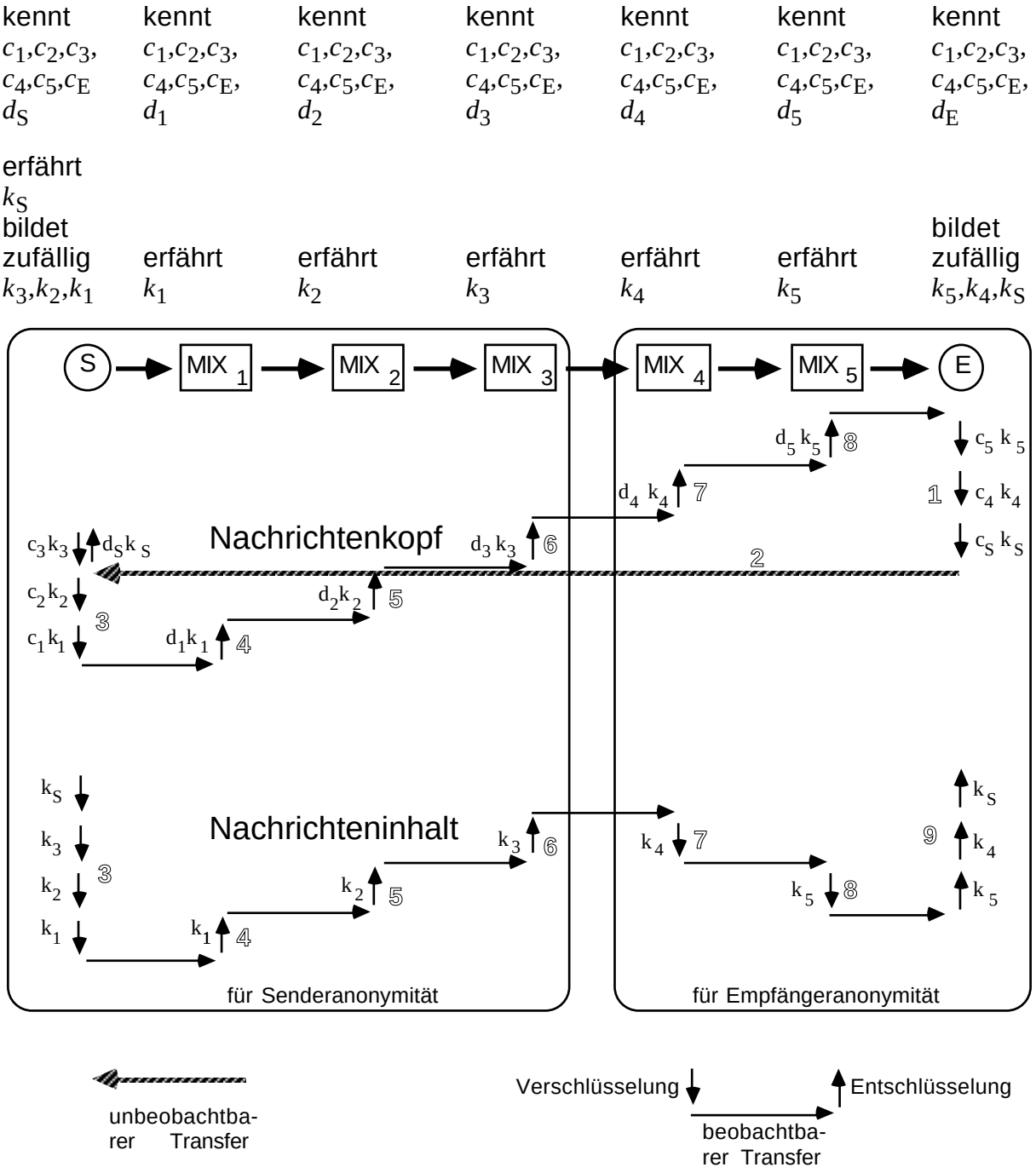


Bild 24: Indirektes Umcodierungsschema für Sender- und Empfängeranonymität

Hat das symmetrische Kryptosystem zusätzlich zu der in Abschnitt 2.2.1.1 für alle Schlüssel *k* und Klartexte *x* geforderten Eigenschaft $k^{-1}(k(x)) = x$ **auch die Eigenschaft $k(k^{-1}(x)) = x$**, d. h. sind nicht nur Ver- und Entschlüsselung, sondern auch Ent- und Verschlüsselung zueinander invers, so braucht beim obigen indirekten längentreuen Umcodierungsschema nicht zwischen dem Gebrauch für Sender- oder Empfängeranonymität unterschieden zu werden, wie im folgenden erläutert wird. Zusätzlich kann damit dem einzelnen MIX auch die Kenntnis von

m als obere Schranke für die Anzahl der zu durchlaufenden MIXe als auch *b*-*m*-1 als obere Schranke für die Länge (in Blöcken) des Nachrichteninhalts vorenthalten werden: Statt als *m*+1-ten Block fügt jeder MIX seinen Block zufälligen Inhalts als letzten Block an. Der Empfänger erhält den Nachrichteninhalt dann in den Blöcken ab Block zwei statt in den letzten Blöcken, vgl. Bild 25.

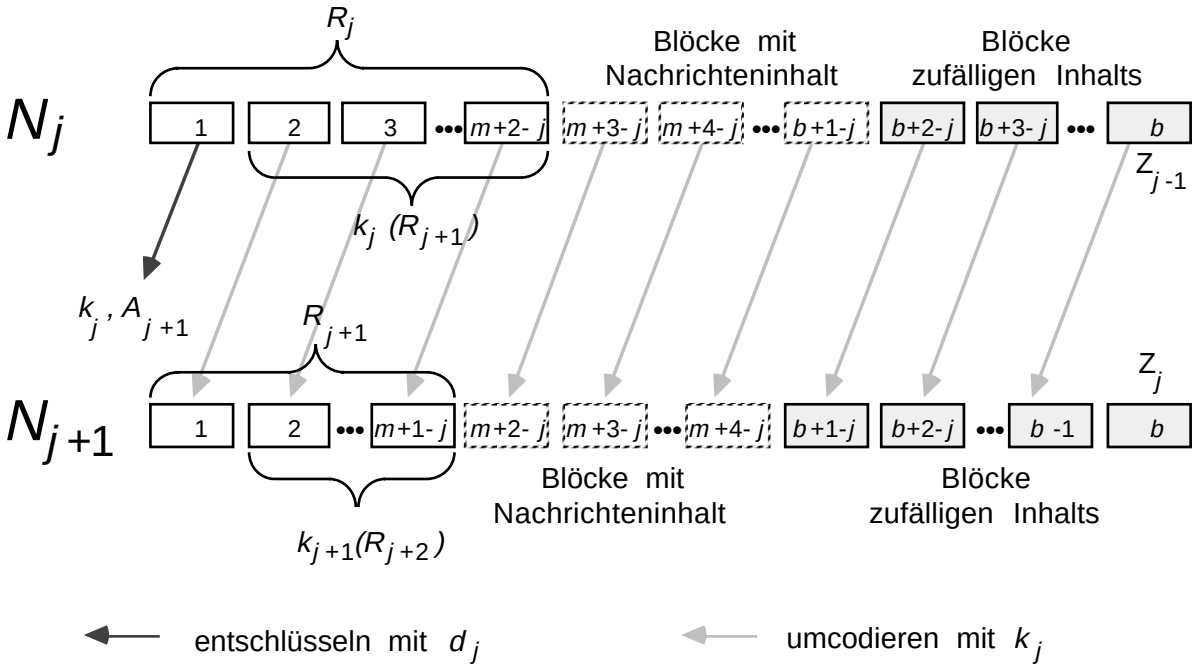


Bild 25: Indirektes längentreues Umcodierungsschema für spezielle symmetrische Kryptosysteme

Wird willkürlich festgelegt, daß jeder MIX alle Blöcke der Nachricht verschlüsselt, so erhält man das einzige längentreue Umcodierungsschema, das in [Chau_81] angegeben ist. Wie bereits erwähnt, wird eine überflüssige Verschlüsselung des angefügten Blockes zufälligen Inhalts vermieden:

$$R_{m+1} = [e]$$
$$R_j = [c_j(k_j,A_{j+1})],k_j^{-1}(R_{j+1}) \quad \text{für } j=m,\ldots,1$$

$$N_1 = R_1,I_1; \quad I_1 = k_0(I) = k_0([I_1]),\ldots,k_0([I_i])$$
$$N_j = R_j,I_j; \quad I_j = k_{j-1}(I_{j-1}),[Z_{j-1}] \quad \text{für } j=2,\ldots,m+1$$

Handelt es sich beim Nachrichteninhalt *I* um Klartext, kann der Empfänger ihn nur verstehen, wenn er die Schlüssel k_0, k_1, ..., k_{m+1} kennt, es sich also mit anderen Worten bei (k_0,A_1,R_1) um eine anonyme Rückadresse handelt und damit das Umcodierungsschema für *Empfängeranonymität* eingesetzt wird.

Soll das Umcodierungsschema für *Senderanonymität* eingesetzt werden, so generiert der Sender die Schlüssel k_1, k_2, ..., k_m und muß den Chiffrierschlüssel c_{m+1} und die Adresse A_{m+1} des Empfängers kennen. Damit der Nachrichteninhalt vom Empfänger verstanden werden

kann, muß ihn der Sender vor dem Absenden der Nachricht mit den von ihm generierten Schlüsseln entschlüsseln. Von den obigen Formeln ist lediglich die für I_1 zu modifizieren:

$$\begin{aligned} I_1 &= k_1^{-1}(k_2^{-1}(...k_m^{-1}(c_{m+1}(I_1))...)) \\ &= k_1^{-1}(k_2^{-1}(...k_m^{-1}(c_{m+1}([I1]))...)),...,k_1^{-1}(k_2^{-1}(...k_m^{-1}(c_{m+1}([Ii]))...)) \end{aligned}$$

2.5.2.6 Effizientes Vermeiden wiederholten Umcodierens

Wie nach den bisherigen drei Beispielen zu vermuten, müssen MIXe bezüglich aller Umcodierungsschemata darauf achten, daß **keine spezielle Umcodierung** (d. h. gleicher umzucodierender Nachrichtenteil, gleiches Kryptosystem und gleicher, die Umcodierung dann genau spezifizierender Schlüssel) **mehrmals** ausgeführt wird. Anderenfalls könnte ein Angreifer ihnen mehrmals unterschiedliche Nachrichten mit demselben Nachrichtenteil schicken und beobachten, welcher Ausgabe-Nachrichtenteil sich in den entsprechenden MIX-Ausgaben in der entsprechenden Häufigkeit wiederholt und damit diesen MIX auch bezüglich der den Nachrichtenteil ursprünglich enthaltenden Nachricht überbrücken.

Eine, leider falsche, Idee, die man bezüglich der Lösung dieses ggf. in jedem MIX viel Speicher- und auch Nachrichtenvergleichsaufwand verursachenden Problems haben kann, besteht darin, dem Angreifer den Zugriff auf Nachrichten (und damit auch auf Nachrichtenteile) zwischen MIXen bzw. zwischen Sender und erstem sowie letztem MIX und Empfänger jeweils durch virtuelle Verbindungs-Verschlüsselung zu verwehren. Der Fehler dieser Idee besteht darin, daß man so natürlich nur den am Umcodieren dieser Nachricht Unbeteiligten den beschriebenen Angriff unmöglich macht, nicht jedoch den Beteiligten, insbesondere den beteiligten MIXen. Der erste und letzte MIX zusammen könnten dann nämlich immer noch die Kommunikationsbeziehung aufdecken, indem sie einen Nachrichtenteil der vom ersten MIX auszusendenden Nachricht mehrmals auf die Reise durch alle „mittleren“ (und in diesem Fall völlig unnützen) MIXe schicken.

Somit bleiben nur *drei Möglichkeiten, den Speicher- und Nachrichtenvergleichsaufwand zu verkleinern*: Wie in [Chau_81 Seite 85] beschrieben, können

1. die *öffentlich bekannten Dechiffrierschlüssel der MIXe öfter gegen neue ausgetauscht werden* oder
2. die Teile, deren Umcodierung durch den öffentlichen Dechiffrierschlüssel des betreffenden MIXes spezifiziert ist, einen „*Zeitstempel*“ enthalten, der den eindeutigen Zeitpunkt ihrer einzigzulässigen Umcodierung bestimmt.

Letzteres senkt den Aufwand fast auf Null, jedoch ist beides für die Anwendung bezüglich bzw. in anonymen Rückadressen denkbar schlecht geeignet, soll der Sender frei wählen können, wann er antwortet.

Deshalb ist die in [Pfi1_85 Seite 74f] genannte

3. Möglichkeit wichtig, statt der mit den geheimgehaltenen Dechiffrierschlüsseln der MIXe umzucodierenden Nachrichtenteile, die nach der Definition eines asymmetrischen Konzelationssystems mindestens hundert, aus Gründen der Sicherheit der heute bekannten asymmetrischen Konzelationssysteme viele hundert Bit lang sein müssen, das Bild

dieser Nachrichtenteile unter einer ihre variable Länge auf eine konstante Länge, z. B. 50 Bit, verkürzende *Hash-Funktion* zu speichern.

Diese Hash-Funktion muß lediglich ihren Urbildraum in etwa gleichmäßig auf ihren Bildraum abbilden – es darf durchaus mit vertretbarem Aufwand möglich sein, zwei Urbilder zu finden, die auf dasselbe Bild abgebildet werden. Bevor ein MIX seinen geheimgehaltenen Dechiffrierschlüssel auf einen Nachrichtenteil anwendet, bildet er sein Bild unter der Hash-Funktion und prüft, ob es schon in seiner Bereits-gemixt-Liste verzeichnet ist. Wenn ja, ignoriert er die Nachricht, wenn nein, wird das Bild in seine Bereits-gemixt-Liste eingetragen und die Entschlüsselung durchgeführt. Der Nachteil dieses Verfahrens ist, daß mit sehr kleiner Wahrscheinlichkeit die Situation auftritt, daß zufällig zwei verschiedene Nachrichtenteile auf dasselbe Bild abgebildet werden und deshalb der später ankommende fälschlicherweise nicht bearbeitet wird. In diesem Fall muß der Sender es mit einer anderen Nachricht, etwa indem er Schlüssel oder zufällige Bitketten ändert, noch mal probieren. Daß er dies kann, erfordert Fehlertoleranz-Maßnahmen, die aber aus anderen Gründen viel dringender und häufiger (kurz: sowieso) gebraucht und in Abschnitt 5.3 ausführlich behandelt werden.

Mittels des im folgenden beschriebenen Verfahrens des **anonymen Abrufs** kann die 1. und 2. Möglichkeit zur Verkleinerung des Speicher- und Nachrichtenvergleichsaufwands der MIXe auch bei Schutz des Empfängers angewandt werden: Statt einer anonymen Rückadresse (mit langer Gültigkeit) wird einem Partner nur eine *Kennzahl* mitgeteilt. Der Partner sendet seine mit der Kennzahl adressierte und Ende-zu-Ende-verschlüsselte Antwort unter Verwendung eines Senderanonymitätsschemas an die in diesem Abschnitt schon mehrmals erwähnte, beim anonymen Abruf eine Zwischenspeicherung durchführende, nichtanonyme Instanz *X*. Hin und wieder sendet die Teilnehmerstation des Empfängers unter Verwendung eines Senderanonymitätsschemas eine anonyme Rückadresse an *X*, mit der sie in einer festgelegten Schubfolge die Antwort erhält, sofern diese schon eingetroffen ist. Obwohl beim anonymen Abruf offensichtlich mehr Nachrichten zu senden sind, dürfte der Vorteil, daß jeweils vorher genau bekannt ist, wer welche Umcodierung in welchem Schub durchführen wird, und deshalb sowohl die 1., als auch die 2. Möglichkeit anwendbar sind, den Speicher- und Nachrichtenvergleichsaufwand der MIXe so sehr verkleinern, daß trotzdem insgesamt der Aufwand wesentlich gesenkt wird.

Das Verfahren des anonymen Abrufs ist insbesondere dort relevant, wo wegen schmalbandiger Teilnehmeranschlußleitungen Verteilung zum Schutz des Empfängers nicht anwendbar ist, vgl. Abschnitt 6.2.

2.5.2.7 Kurze Vorausschau

Wird all das bisher Gesagte beachtet, so ist die schlimmste Folge eines **aktiven Angriffs** (oder Fehlers) durch einen MIX der Verlust einer Nachricht, nicht jedoch der Verlust der Anonymität der Kommunikationsbeziehung. Durch öffentlichen Zugriff auf die von MIXen gesendeten Nachrichten oder – auch im Falle von virtueller Verbindungs-Verschlüsselung wirksam – ihr Unterschreiben mit einem MIX-spezifischen Signierschlüssel ist ein Verlust jedoch feststellbar und beweisbar, vgl. Abschnitt 5.8. Durch geeignete Fehlertoleranz-Maßnahmen ist zudem die Wahrscheinlichkeit des Verlustes von Nachrichten durch Ausfall eines

MIXes (ein ausgefallener MIX innerhalb der MIX-Folge einer Nachricht reicht, um sie zu verlieren!) zu vermindern, wie in dem bereits angekündigten Abschnitt 5.3 gezeigt wird.

Wegen der Zeitverzögerung durch Warten auf Nachrichten, durch Prüfen auf Wiederholungen und Umsortieren der Nachrichten sowie wegen des Zeitaufwandes für die Entschlüsselungen in einem asymmetrischen Konzelationssystem ist das soweit beschriebene Verfahren der umcodierenden MIXe nicht für Anwendungen geeignet, die kurze Übertragungszeiten fordern.

Wandelt man dieses Verfahren aber wie erstmals in [Pfi1_85 Seite 25 bis 29] beschrieben ab, so kann man **anonyme Kanäle** schalten, die z. B. den Realzeitanforderungen des Telefonverkehrs genügen und in Abschnitt 3.2.2.1 genauer erklärt werden. Hierzu wird zum Kanalaufbau eine spezielle Nachricht nach dem oben beschriebenen Verfahren übertragen, die jedem gewählten MIX einen Schlüssel eines schnelleren symmetrischen Kryptosystems übergibt, den dieser MIX von da an für die Entschlüsselung des Verkehrs auf diesem Kanal verwendet. Das Umcodieren bei Kanälen nützt natürlich nur dann etwas, wenn mindestens zwei Kanäle von gleicher Kapazität durch denselben MIX gleichzeitig auf- und abgebaut werden.

Will man durch umcodierende MIXe nicht nur Kommunikationsbeziehungen, sondern auch das **Senden** und **Empfangen** von Teilnehmerstationen schützen, muß jede Teilnehmerstation ein MIX sein oder sehr viele bedeutungslose Nachrichten senden, wie dies im folgenden Abschnitt 2.5.3 sowie für die Randbedingung schmalbandiger Teilnehmeranschlußleitungen in Abschnitt 6.2 beschrieben wird. Ersteres erfordert, wie in Abschnitt 3.2.2.4 gezeigt wird, bei normaler Verkehrsdichte große Wartezeiten auf mehrere Nachrichten bzw. auf Auf- oder Abbau von mehreren Kanälen. Bei hoher Verkehrsdichte erfordert es Teilnehmerstationen, die sehr viele Nachrichten und breitbandige Kanäle umcodieren und vermitteln können. Es sei hier schon angedeutet, daß derartige Teilnehmerstationen sehr aufwendig und in der überschaubaren Zukunft zu teuer sind.

2.5.2.8 Notwendige Eigenschaften des asymmetrischen Konzelationssystems und Brechen der direkten RSA-Implementierung

Als Schluß dieses Abschnitts sei ausdrücklich darauf hingewiesen, daß bei (mittleren) MIXen das verwendete asymmetrische Konzelationssystem nicht nur – wie jedes asymmetrische Konzelationssystem – einem adaptiven aktiven Angriff mit gewähltem Klartext, sondern im wesentlichen (d. h. bis auf die Erschwernis, daß der MIX die zufälligen Bitketten nicht ausgibt) auch einem **aktiven Angriff mit gewähltem Schlüsseltext** (chosen-ciphertext attack) widerstehen muß. Zumindest wenn anonyme Rückadressen verwendet werden, können die Schlüsselpaare der MIXe nicht nach jedem Schub ausgetauscht werden, so daß das verwendete asymmetrische Konzelationssystem dann sogar einem **adaptiven aktiven Angriff mit gewähltem Schlüsseltext** (adaptive chosen ciphertext attack) widerstehen muß.

Es sei an die Abschnitt 2.2.1.2 und 2.2.2.2 erinnert, wo bereits gesagt wurde, daß es bisher kein *einschrittiges* asymmetrisches Konzelationssystem gibt, das solch einem Angriff in im Sinne von Abschnitt 2.2.2.2 beweisbarer Form standhält und damit ein kanonischer Kandidat

zur Implementierung (mittlerer) MIXe wäre. Da es *mehrschrittige* asymmetrische Konzelationssysteme gibt, die dies in beweisbarer Form tun, kann man aus ihnen beweisbar sichere MIX-Implementierungen gewinnen. Sie sind wegen der Mehrschrittigkeit zwischen Sender (bzw. dem die Umcodierung Spezifizierenden) und *jedem* durchlaufenen MIX allerdings sehr aufwendig: Sollen die (mittleren) MIXe MIX_1 bis MIX_n verwendet werden, so muß der Schlüsselaustauschdialog zwischen Sender und MIX_i, $i>1$, jeweils durch alle MIX_j mit $1 \leq j < i$ geschehen [Bött_89 Abschnitt 3.3.1]. Dies macht den Hauptvorteil von MIXen gegenüber den anderen Verfahren zum Schutz der Verkehrs- und Interessensdaten zunichte, nämlich die vergleichsweise geringe benötigte Bandbreite zwischen Teilnehmerstation und Kommunikationsnetz. Im Rest dieser Arbeit wird deshalb jeweils nur der Fall explizit behandelt, daß zur Implementierung von (mittleren) MIXen ein einschrittiges asymmetrische Konzelationssystem verwendet wird.

In Abschnitt 2.2.1.2.1 wurde gezeigt, daß RSA ohne geeignetes, vom Entschlüßler geprüftes Redundanzprädikat bereits einem nicht-adaptiven aktiven Angriff mit gewähltem Klartext nicht widerstehen kann. Im folgenden wird gezeigt, daß die Verwendung von **RSA ohne ein zusätzliches Redundanzprädikat und mit zusammenhängenden zufälligen Bitketten**, d. h. so wie von David Chaum in [Chau_81 Seite 84] beschrieben, **vollkommen unsicher ist**. Dies wird, um die Notation einfach zu halten, anhand der in Bild 19 verwendeten gezeigt.

Der Angreifer beobachte eine Nachricht $c(z,N)$ am Eingang des MIXes. Der Angreifer wählt sich einen geeigneten Faktor f und reicht dem MIX die Nachricht $c(z,N) \bullet c(f)$ zum Mixen im selben bzw. einem späteren Schub ein. Nachdem in Abschnitt 2.2.1.2.1 die perfekte Unverkettbarkeit von $c(z,N)$ und $c(z,N) \bullet c(f)$ bei Wahl beliebiger Faktoren f gezeigt wurde, gilt „praktisch“ perfekte Unverkettbarkeit, wenn nur „viele“ Faktoren möglich sind, d. h., der MIX hat dann keine Chance, einen aktiven Angriff zu erkennen. Es bleibt also nur zu zeigen, daß der Angreifer durch seinen Angriff etwas erhält, das es ihm erlaubt, N aus den Ausgabe-Nachrichten des entsprechenden Schubes herauszufinden. Intuitiv und näherungsweise richtig kann dies in der bisherigen allgemeinen Notation folgendermaßen dargestellt werden: der MIX bildet intern $d(c(z,N) \bullet c(f)) = (z,N) \bullet f$. Kann der Angreifer f so wählen, daß an der Stelle der auszugebenden Nachricht $N \bullet f$ entsteht, was in der bisherigen Notation $(z,N) \bullet f = (?, N \bullet f)$ lautet, ist sein Angriff gelungen: er muß nur noch prüfen, welche beiden Nachrichten des bzw. der beiden Schübe die Gleichung $X = Y \bullet f$ erfüllen. Was an der Stelle der zufälligen Bitkette entsteht ist irrelevant, da dies vom MIX weder geprüft noch ausgegeben wird.

Warum dieser Angriff gelingt, ist ohne die Verwendung weiterer Details von RSA nicht zu erklären. Die für das Verständnis notwendigen lauten, daß bei RSA Ver- und Entschlüsselung durch Exponentiation innerhalb eines Restklassenrings erfolgen. Der den Restklassenring charakterisierende Modulus $n = p \bullet q$, wobei p und q zufällig gewählte Primzahlen von – aus Sicherheitsgründen – je mindestens 250 Bit Länge sind, bildet mit dem zur Verschlüsselung verwendeten Exponenten den öffentlich bekannten Chiffrierschlüssel. In jedem Block dieser Blockchiffre können $\mathbb{b}$ Bit lange zufällige Bitketten und $\mathbb{B}$ Bit lange Nachrichten untergebracht werden, sofern nur $2^{\mathbb{b}} \bullet 2^{\mathbb{B}} < n$ ist. Bei der Implementierungsbeschreibung von David Chaum, wie bei jeder effizienten Implementierung, ist $\mathbb{b}$ sehr viel kleiner als $\mathbb{B}$. Werden, wie bei David

Chaum und in der obigen Notation angedeutet die zufälligen Bitketten an den Bitstellen höherer Wertigkeit untergebracht, so gilt $(z,N) = z \bullet 2^{\mathbb{B}} + N$ und $(z,N) \bullet f \equiv (z \bullet 2^{\mathbb{B}} + N) \bullet f \equiv z \bullet 2^{\mathbb{B}} \bullet f + N \bullet f$.

Aus der die Bezeichner z' und N' definierenden Kongruenz $(z,N) \bullet f \equiv z' \bullet 2^{\mathbb{B}} + N'$ folgt $z \bullet 2^{\mathbb{B}} \bullet f + N \bullet f \equiv z' \bullet 2^{\mathbb{B}} + N'$, daraus folgt $2^{\mathbb{B}} \bullet (z \bullet f - z') \equiv N' - N \bullet f$ und daraus folgt $z \bullet f - z' \equiv (N' - N \bullet f) \bullet (2^{\mathbb{B}})^{-1}$. Wählt der Angreifer $f \leq 2^{\mathbb{b}}$, so gilt $-2^{\mathbb{b}} < z \bullet f - z' < 2^{2\mathbb{b}}$. Der Angreifer setzt nun in die Formel $z \bullet f - z' \equiv (N' - N \bullet f) \bullet (2^{\mathbb{B}})^{-1}$ für N und N' jeweils alle Ausgabe-Nachrichten des bzw. der entsprechenden Schübe ein und prüft, ob $z \bullet f - z'$ im entsprechenden Intervall liegt. Dies ist, da $\mathbb{b}$ sehr viel kleiner als $\mathbb{B}$ ist, mit sehr hoher Wahrscheinlichkeit nur für genau ein Paar der Ausgabe-Nachrichten der Fall. Die erste Nachricht N dieses Paares ist die Nachricht, bezüglich derer der MIX mittels des aktiven Angriffs überbrückt werden soll. Die zweite Nachricht N' ist die der vom Angreifer gebildeten Eingabe-Nachricht entsprechende Ausgabe-Nachricht.

Erfüllen zwei oder mehr Paare diese Bedingung, so wiederholt der Angreifer seinen Angriff mit einem anderen Faktor und führt bei seinem zweiten sowie ggf. allen weiteren Versuchen das paarweise Einsetzen nur noch mit den durch die bisherigen Tests nicht ausgeschlossenen Nachrichten des zuerst erwähnten Schubes durch. Dieser Angriff auf die Kommunikationsbeziehung einer Nachricht N ist mit in der bei N verwendeten Schubgröße quadratischem Aufwand durchzuführen: der Angreifer muß bei jedem Versuch nur einmal mit RSA verschlüsseln (sowie das Ergebnis mit N multiplizieren und modulo n reduzieren) sowie für jede Nachricht des Schubes eine Addition, zwei Multiplikationen, zwei Reduktionen modulo n und zwei Vergleiche durchführen.

Die bisher unterstellte Annahme, daß $\mathbb{b}$ sehr viel kleiner als $\mathbb{B}$ ist, ist nicht nötig. Ein entsprechender Angriff ist bereits immer dann möglich, wenn $\mathbb{B}$ so groß ist, daß eine Wahl von $f \leq 2^{\mathbb{B}-1}$ für den zu überbrückenden MIX „praktische“ Unverkettbarkeit von (z,N) und $(z,N) \bullet f$ ergibt. Die resultierende Ungleichung $-2^{\mathbb{b}} < z \bullet f - z' < 2^{\mathbb{b}+\mathbb{B}-1}$ genügt bereits, damit der Angreifer in jeder Iteration seines Angriffs einen von der Iterationszahl unabhängigen Anteil der noch möglichen Nachrichten des ursprünglichen Schubes ausscheiden kann. Der Aufwand des Angriffs steigt damit lediglich um einen logarithmischen Faktor.

Ist eine adaptive Wiederholung des Angriffs nicht möglich, etwa weil der MIX nach jedem Schub sein Schlüsselpaar wechselt, kann der Angreifer dieselbe Information wie durch seinen iterativen Angriff dadurch erhalten, daß er gleich mehrere Faktoren wählt, entsprechend mehrere Nachrichten bildet und alle diese zusammen mit der Nachricht, bezüglich der er den MIX überbrücken will, zum Mixen im selben Schub einreicht.

Es wurde bisher nicht geprüft, ob mit den in jüngerer Vergangenheit veröffentlichten Angriffen auf RSA eine Senkung des Angriffs-Aufwands möglich ist.

Alles bisher am Beispiel eines direkten Umcodierungsschemas für Senderanonymität Gezeigte gilt natürlich auch für jedes indirekte Umcodierungsschema, indem für den Angriff statt ganzer Nachrichten nur (Rück-)Adreßteile verwendet werden. Dies ist bei allen in [Chau_81] angegebenen Umcodierungsschemata möglich [PfPf_89 Abschnitt 3.2]. Dies kann jedoch durch Verwendung geeigneter Umcodierungsschemata, d. h. solcher, die keine direkt sondern nur indirekt umcodierte Teile in beobachtbarer Weise verwenden, vermieden werden. Hierbei sollte das zur indirekten Umcodierung verwendete (symmetrische) Konzelationssystem völlig anders als RSA arbeiten.

Werden die zufälligen Bitketten an den Bitstellen niedrigerer Wertigkeit untergebracht, so existiert ein analoger und ebenfalls erfolgreicher Angriff [PfPf_89 Abschnitt 3.2].

Ein analoger Angriff versagt, wenn statt einer zusammenhängenden zufälligen Bitkette in nicht zu großen Abständen (möglichst also äquidistantem Abstand) genügend viele einzelne zufällige Bits in die Nachricht „eingestreut“ und vom MIX entsprechend „herausgesiebt“ werden. Allerdings sind schwächere Angriffe möglich, wenn entweder die zufälligen Bitketten kürzer als die mitverschlüsselten Nachrichten sind oder auch nur ein größerer Block von Nachrichtenbits nicht von zufälligen Bits unterbrochen ist [PfPf_89 Abschnitt 3.2]. Hierbei können 10 Bits bereits als größerer Block gelten.

Allerdings kann selbst der schwächere Angriff leicht dadurch verhindert werden, daß die eingestreute zufällige Bitkette und die mitverschlüsselte Nachricht durch **Verschlüsselung in einem völlig anderen (symmetrischen) Konzelationssystem** (z. B. DES mit einem öffentlich bekannten Schlüssel) vollständig „gemischt“ werden, bevor mit RSA verschlüsselt wird.

Ein alternativer Ansatz ist, wie in Abschnitt 2.2.1.2.1 bereits erwähnt, daß die Klartexte ein geeignetes **Redundanzprädikat** erfüllen müssen. Erfüllt der vom MIX entschlüsselte Klartext z,N das Redundanzprädikat nicht, gibt der MIX keinen Teil dieser Nachricht aus. Um ausgiebiges Probieren zu verhindern, falls man sich der kryptographischen Stärke des Redundanzprädikates nicht sicher ist, können die Sender falsch gebildeter Nachrichten durch Zusammenarbeit aller MIXe identifiziert werden: Jeder MIX_i gibt zu der rückzuverfolgenden Nachricht die im Falle der Verwendung eines deterministischen asymmetrischen Konzelationssystems explizit mitverschlüsselte zufällige Bitkette z_i an (bzw. bei Verwendung eines indeterministisch verschlüsselnden asymmetrischen Konzelationssystems die hierbei verwendete. Kann MIX_i dies nicht, ist das indeterministisch verschlüsselnde asymmetrische Konzelationssystem für diesen Zweck ungeeignet). Die Sender können zur Rechenschaft gezogen werden, sofern alle Nachrichtenübertragungen öffentlich sind oder alle Nachrichten vom jeweiligen Sender (Teilnehmerstation oder MIX) authentifiziert werden. Bei der Rückverfolgung von Nachrichten ist zu beachten, daß bei Verwendung von anonymen Rückadressen nicht der Verwender, sondern nur der Generierer zur Rechenschaft gezogen wird. Der Verwender muß dabei anonym bleiben können, da ansonsten Kommunikationspartner einander durch Zusenden falsch gebildeter Rückadressen identifizieren könnten.

2.5.3 Schutz des Senders

Der Sender einer Nachricht kann sich (bezüglich des Sendens und bei expliziter Adressierung auch den Empfänger bezüglich seines sonstigen Empfangens) schützen, indem er **bedeutungslose Nachrichten** (dummy traffic) sendet. Werden diese im Falle expliziter Adressierung in ihrem verschlüsselten Nachrichteninhaltsteil mit einem speziellen Kennzeichen versehen bzw. im Falle von Verteilung und impliziter Adressierung mit nicht existenten impliziten Adressen versehen, können sie von der explizit adressierten bzw. allen Teilnehmerstationen weggeworfen werden. Werden auch bedeutungslose Nachrichten gesendet, kann das Netz nicht mehr entscheiden, wann genau und wieviele bedeutungsvolle Nachrichten ein Teilnehmer sendet

[Chau_81]. (Symmetrisch dazu kann das Netz selbst im Falle expliziter Adressierung nicht mehr entscheiden, wann genau und wieviele bedeutungsvolle Nachrichten ein Teilnehmer empfängt. Da das Wegwerfen von an andere Stationen adressierten Nachrichten jedoch nicht aufwendiger ist als das Wegwerfen von an die eigene Station adressierten, und ersteres Empfängeranonymität auch vor dem Sender der Nachricht und erhebliche Aufwandsersparnis in geeignet entworfenen Kommunikationsnetzen ermöglicht, wurde das Verfahren der bedeutungslosen Nachrichten in Abschnitt 2.5.1 nicht erwähnt. Mag oder kann man sich vollständige Verteilung (broadcast) nicht leisten, wird man aus der Sicht des Empfängers statt bedeutungsloser Nachrichten immer partielle Verteilung (multicast) vorziehen.)

Das Verfahren der bedeutungslosen Nachrichten verursacht stets einen sehr hohen Aufwand und erfüllt zugleich nicht die Forderung nach Anonymität des Senders auch bei Angriffen, an denen sich der Empfänger bedeutungsvoller Nachrichten beteiligt. In der Diktion von Abschnitt 2.1.1 wird durch dieses Verfahren also lediglich das *Senden* (bzw. symmetrisch dazu: das *Empfangen*) bedeutungsvoller Nachrichten *für Unbeteiligte unbeobachtbar und unverkettbar*. Anonymität des Senders vor Beteiligten ist mit diesem Verfahren allein jedoch bezüglich realistischer Angreifermodelle nicht zu erreichen. Trotz dieser offensichtlichen Schwächen des Verfahrens der bedeutungslosen Nachrichten ist es das einzige zum Schutz der Verkehrs- und Interessensdaten, das auch in der neueren, nicht von David Chaum oder Mitgliedern unserer Karlsruher Datenschutz-Arbeitsgruppe geschriebenen, weit verbreiteten und zitierten Literatur zum Thema „Datenschutz und Sicherheit in Kommunikationsnetzen“, insbesondere auch im Entwurf einer Ergänzung des ISO Modells für „Offene Kommunikation“ in bezug auf Datenschutz und Sicherheit steht, vgl. [VoKe_83 Seite 157f, VoKe_85 Seite 18, ISO7498SA_86 Seite 28f, 32, 34, 46, 54, 57, ISO7498-2_87 Seite 17, 24-29, 31-33, 36, 39f, 53, 56, 60f, Rul1_87, AbJe_87 Seite 28, Folt_87 Seite 45] (natürlich gibt es auch neuere weit verbreitete Literatur, die überhaupt keine Lösungsmöglichkeit nennt, vgl. [Brau_87]). Kennen diese weltberühmten Kapazitäten ihr eigenes Gebiet nicht – oder wollen oder dürfen sie es nicht kennen, vgl. Abschnitt 2.2.2.4 ?

Wird das Senden (und Empfangen) bedeutungsloser Nachrichten mit einem Verfahren zum Schutz der Kommunikationsbeziehung geeignet kombiniert, so hält es auch Angriffen stand, an denen sich der Netzbetreiber und der Empfänger bedeutungsvoller Nachrichten beteiligen. Soll durch die Kombination der zwei Verfahren der Sender (oder Empfänger) geschützt werden, so muß er mit einer von seinen Sende- und Empfangswünschen unabhängigen Rate Nachrichten senden und empfangen. Dies bedeutet, daß die Bandbreite des Kommunikationsnetzes im wesentlichen *statisch* auf die Teilnehmer aufgeteilt werden muß, was für viele Anwendungen ungeschickt ist. Außerdem kann auf diese Weise höchstens *komplexitätstheoretischer* Schutz des Senders (und Empfängers) erreicht werden. Trotz dieser Einschränkungen kann diese Kombination bei bestimmten äußeren Randbedingungen sinnvoll sein, siehe Abschnitt 6.2.

Die in den beiden folgenden Unterabschnitten beschriebenen Verfahrensklassen zum Schutz des Senders haben die gerade genannten Nachteile nicht. Sie sind in gewisser Weise effizient, indem sie eine *dynamische* Aufteilung der Bandbreite erlauben sowie bedeutungslose Nachrichten und Umcodieren vermeiden, und es ist in der *informationstheoretischen* Modellwelt

bewiesen, daß der Sender selbst bei Identität oder Zusammenarbeit von Empfänger und Netzbetreiber anonym ist.

Zunächst wird mit dem überlagernden Senden ein sehr mächtiges, aber auch entsprechend aufwendiges Verfahren zum Realisieren von Senderanonymität auf einem beliebigen Kommunikationsnetz beschrieben.

Danach wird erklärt, wie ein Kommunikationsnetz insbesondere durch Wahl einer geeigneten Leitungstopologie (Ring oder Baum) und digitale Signalregenerierung gleich so – und damit preiswerter als durch „Aufsetzen“ eines zusätzlichen Verfahrens – realisiert werden kann, daß realistische Angreifer nur unter hohem Aufwand das Senden von einzelnen Stationen beobachten können. Ist hierbei ihr Aufwand genauso groß wie der anderer Formen der Individualüberwachung, so ist dem in einer Demokratie nötigen Datenschutz Genüge getan, vgl. Abschnitt 1.4.

Die beiden Verfahrensklassen können sinnvoll kombiniert werden, was bei der Beschreibung der zweiten jeweils am Schluß der Unterabschnitte erklärt wird.

2.5.3.1 Überlagerndes Senden (DC-Netz)

Nach einem ersten Überblick über das verallgemeinerte überlagernde Senden in Abschnitt 2.5.3.1.1 wird in Abschnitt 2.5.3.1.2 die Senderanonymität definiert und bewiesen.

In Abschnitt 2.5.3.1.3 wird mit dem überlagernden Empfangen ein Verfahren eingeführt, das eine wesentlich effizientere Nutzung des überlagernden Sendens erlaubt.

Schließlich werden in Abschnitt 2.5.3.1.4 einige Überlegungen zur Optimalität des überlagernden Sendens bezüglich Senderanonymität, sowie zu seinem Aufwand und zu möglichen Implementierungen angestellt.

2.5.3.1.1 Ein erster Überblick

In [Cha3_85, Cha8_85, Chau_88] gibt David Chaum eine von ihm DC-Netz genannte Möglichkeit zum anonymen Senden an (DC-network als Abkürzung für Dining Cryptographers network; der Name ist passend zu seinem Einführungsbeispiel gewählt; als zusätzliche Merkhilfe sei erwähnt, daß er David Chaums Initialen darstellt). Diese Möglichkeit zum anonymen Senden (und unbeobachtbaren Empfangen gemäß Abschnitt 2.5.1) wurde in [Pfi1_85 Seite 40, 41] zur im folgenden beschriebenen Möglichkeit verallgemeinert.

Bekanntlich bildet jedes endliche Alphabet bezüglich einer ab 0 beginnenden Numerierung seiner Zeichen und der Addition (modulo der Zeichenanzahl) bezüglich dieser Numerierung eine abelsche Gruppe. Wie üblich werde unter der Subtraktion (eines Zeichens) die Addition seines inversen Gruppenelementes verstanden. Das **verallgemeinerte überlagernde Senden** geschieht dann folgendermaßen:

Teilnehmerstationen erzeugen für jedes zu sendende Nutzzeichen ein oder mehrere Schlüsselzeichen zufällig und gemäß einer Gleichverteilung. Jedes dieser Schlüsselzeichen teilen sie genau einer anderen Teilnehmerstation mittels eines (noch zu diskutierenden) Konzelation garantierenden Kanals mit. (Wer mit wem solch einen Konzelation garantierenden Kanal unter-

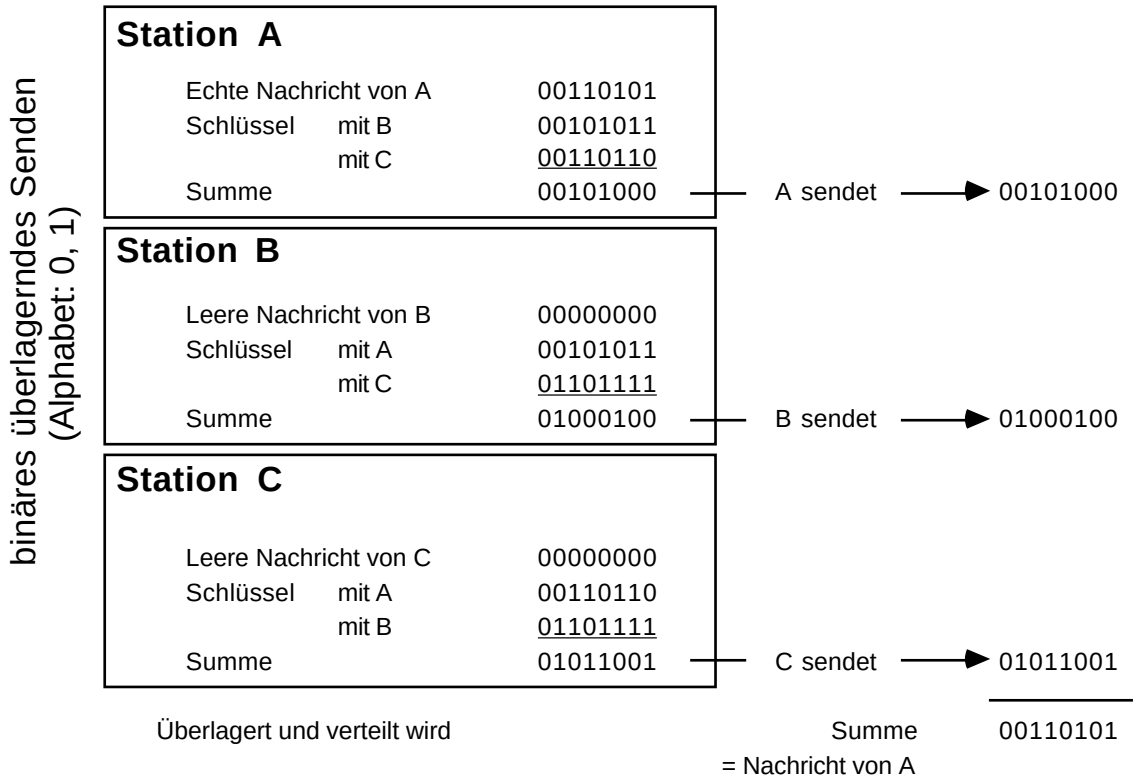
hält, muß nicht geheimgehalten werden, sondern sollte sogar, um Angriffen gegen die Diensterbringung leichter begegnen zu können, öffentlich bekannt sein, vgl. Abschnitt 5.8.) Jede Teilnehmerstation addiert (modulo Zeichenanzahl) lokal alle von ihr erzeugten Schlüsselzeichen, subtrahiert (modulo Zeichenanzahl) davon lokal alle ihr mitgeteilten Schlüsselzeichen und addiert (modulo Zeichenanzahl), sofern sie ein Nutzzeichen senden will, lokal ihr Nutzzeichen. Dieses Addieren (bzw. Subtrahieren) modulo der Zeichenanzahl des verwendeten Alphabets wird *Überlagern* genannt. Jede Teilnehmerstation sendet das Ergebnis ihrer lokalen Überlagerung (daher der den Mechanismus betonende Name *überlagerndes Senden*). Alle gesendeten Zeichen werden global überlagert (modulo Zeichenanzahl addiert) und das entstehende Summenzeichen an alle Teilnehmerstationen verteilt.

Da jedes Schlüsselzeichen genau einmal addiert und subtrahiert wurde, sich nach der globalen Überlagerung also alle Schlüsselzeichen gegenseitig wegheben, ist das Summenzeichen die Summe (modulo Zeichenanzahl) aller gesendeten Nutzzeichen. Wollte keine Teilnehmerstation senden, ist das Summenzeichen das 0 entsprechende Zeichen, wollte genau eine Teilnehmerstation senden, ist das Summenzeichen gleich dem gesendeten Nutzzeichen.

Wählt man als Alphabet die Binärzeichen 0 und 1, so erhält man den für praktische Zwecke wichtigen, von David Chaum angegebenen Spezialfall des **binären überlagernden Sendens**, bei dem zwischen Addition und Subtraktion von Zeichen nicht zu unterschieden werden braucht (Bild 26).

Natürlich können (digitale) *Überlagerungs-Kollisionen* auftreten, falls mehrere Teilnehmerstationen gleichzeitig senden wollen: alle Stationen erhalten die (wohldefinierte) Summe des gleichzeitig Gesendeten. Kollisionen sind ein übliches Problem bei Verteil-Kanälen mit Mehrfachzugriff, zu dessen Lösung es eine große Zahl von Zugriffsverfahren gibt. Alle publizierten Zugriffsverfahren sind auf (analoge) *Übertragungs-Kollisionen*, z. B. in Bussystemen (Bsp. Ethernet), Funk- und Satellitennetzen ausgelegt, bei denen es kein wohldefiniertes „Kollisionsergebnis“ gibt. Die „Arbeitsbedingungen“ der Zugriffsverfahren sind beim überlagernden Senden in dieser Hinsicht also wesentlich besser als bei üblichen Verteil-Kanälen. Allerdings darf man natürlich nur solche Zugriffsverfahren verwenden, die die Anonymität des Senders und – wenn möglich – auch die Unverkettbarkeit von Sendeereignissen erhalten. Daneben sollten sie bei zu erwartender Verkehrsverteilung den zur Verfügung stehenden Kanal günstig nutzen, was in Abschnitt 3.1.2 noch ausführlich behandelt werden wird. Beispiele anonymer und nichtverkettender Zugriffsverfahren sind das einfache, aber nicht sehr effiziente Verfahren slotted ALOHA und eine für Kanäle mit großer Verzögerungszeit entworfene, effiziente Reservierungstechnik [Tane_81 Seite 272, Cha3_85].

Vereinbart man, daß ein Teilnehmer einen Übertragungsrahmen, in dem er ohne Kollision gesendet hat, weiterbenutzen darf und andere Teilnehmer in diesem Rahmen erst wieder senden dürfen, wenn er einmal nicht benutzt wurde, so kann man durch die Verwendung mehrerer Übertragungsrahmen (slots) sehr effizient Kanäle schalten [Höck_85, HöPf_85], was ebenfalls in Abschnitt 3.1.2 ausführlich behandelt werden wird. Natürlich ist alles, was über solch einen Kanal gesendet wird, verkettbar – andererseits werden Kanäle typischerweise für solche Dienste verwendet, in deren Natur dies liegt. Dies wird ausführlich in Abschnitt 2.6 diskutiert.



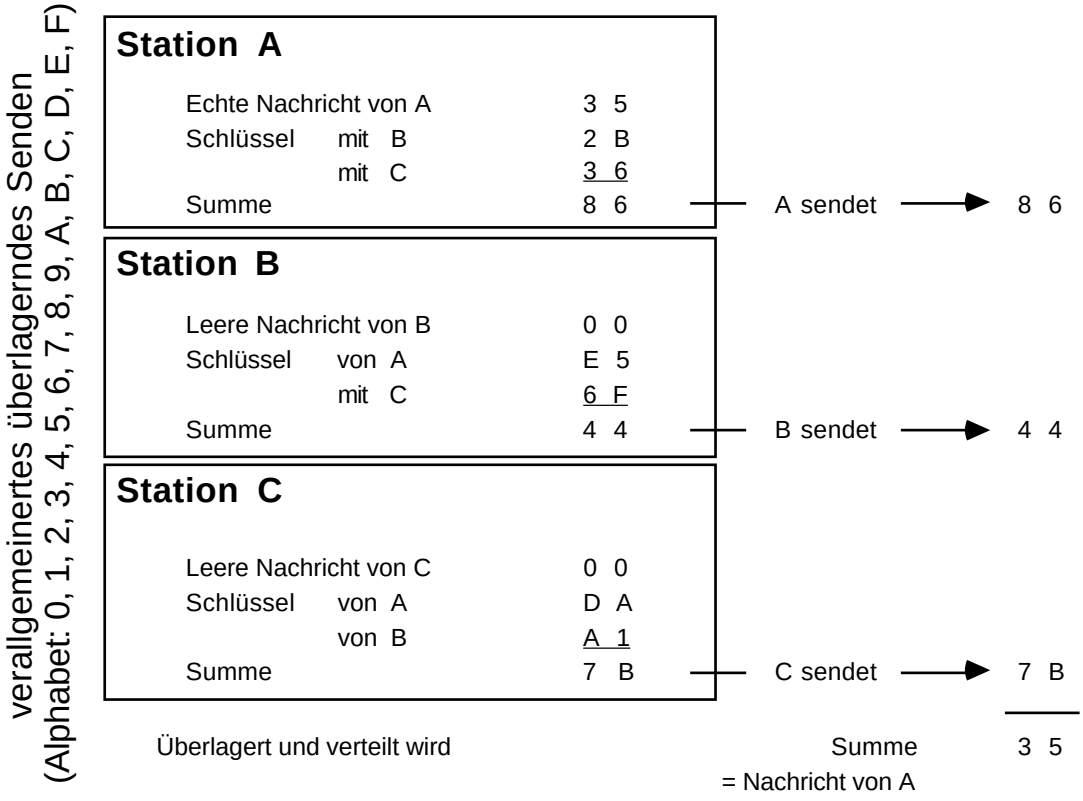


Bild 26: Überlagerndes Senden

2.5.3.1.2 Definition und Beweis der Senderanonymität

Nach dieser dem Überblick dienenden Kurzbeschreibung des überlagernden Sendens wird zunächst die *Senderanonymität* genau definiert und für den allgemeinst-möglichen Fall der Überlagerung (beliebige abelsche Gruppe) bewiesen.

Danach wird gezeigt, wie mittels *überlagerndem Empfangens* ausgenutzt werden kann, daß Überlagerungs-Kollisionen einen wohldefinierten und für alle am überlagernden Senden beteiligten Stationen wahrnehmbaren Wert haben. Beides im Gegensatz zu Übertragungs-Kollisionen in Bussystemen (Bsp. Ethernet) und Funknetzen und nur ersteres auch im Gegensatz zu Satellitennetzen.

Einige Bemerkungen zur Realisierung des Schlüsselaustausches und zur Implementierung des überlagernden Sendens schließen diesen Abschnitt ab.

David Chaum beweist in [Cha3_85] für binäres überlagerndes Senden (genauer: unter Ausnutzung der Körpereigenschaften von GF(2)), daß, solange eine Gruppe von Teilnehmerstationen Schlüssel nur in der beschriebenen Form weitergibt und solange diese Gruppe bezüglich der ausgetauschten Schlüssel zusammenhängend ist, andere an diesem Verfahren Beteiligte auch durch Zusammentragen all ihrer Information keine Information darüber erhalten, wer innerhalb der Gruppe sendet. „Bezüglich der ausgetauschten Schlüssel zusammenhängend“ bedeutet, daß für je zwei beliebige Teilnehmerstationen T_0, T_1 der Gruppe es eine möglicherweise leere Folge von Teilnehmerstationen T_2 bis T_n der Gruppe gibt, so daß für $1 \le i \le n$ gilt: T_i hat mit $T_{(i+1) \bmod (n+1)}$ einen Schlüssel ausgetauscht.

Im folgenden wird ein vereinfachter und auf verallgemeinertes überlagerndes Senden erweiterter Beweis dieses Sachverhaltes gegeben.

Der Beweis verwendet nur, daß die Zeichen und die auf ihnen definierte Addition eine *abelsche Gruppe* bilden, nicht aber, daß die Addition nach der Numerierung der Zeichen modulo der Zeichenanzahl durchgeführt wird (mit anderen Worten: daß es sich um eine *zyklische Gruppe* handelt). Die dadurch beim verallgemeinerten überlagernden Senden implizit gegebene *Ringstruktur* wird für den Beweis nicht verwendet, so daß er so allgemein wie irgend wünschenswert ist:

1. Damit eine Station kein Zeichen senden kann, muß es ein neutrales Element geben.
2. Damit sich die Schlüssel nach der Überlagerung paarweise wegheben, muß es zu jedem Zeichen ein inverses geben.
3. Damit man Freiheiten bei der Organisation der lokalen und insbesondere globalen Überlagerung hat, sollte die Addition kommutativ sein.
   Damit ein Schlüsselaustausch zwischen beliebigen Paaren von Teilnehmerstationen möglich ist, muß die Addition kommutativ sein. Denn anderenfalls gäbe es eine durch die globale Überlagerungsreihenfolge gegebene lineare Ordnung der von den Teilnehmerstationen gesendeten Nutzzeichen und damit auch der Teilnehmerstationen. Da Schlüsselzeichen nicht mit Nutzzeichen kommutierten, könnte jede Teilnehmerstation nur mit ein bzw. zwei bezüglich der linearen Ordnung benachbarten Teilnehmerstationen Schlüssel austauschen.

Der Beweis ist also etwas allgemeiner als die Definition des verallgemeinerten überlagernden Sendens – allerdings gibt es für diese größere Allgemeinheit keine Nutzanwendung. Denn der

Hauptsatz über endliche abelsche Gruppen (vgl. [Waer_67 Seite 9]) besagt, daß jede endliche abelsche Gruppe als direktes Produkt zyklischer Gruppen geschrieben werden kann. Das direkte Produkt, d. h. die komponentenweise Ausführung der Addition in der jeweiligen zyklischen Gruppe, entspricht aber genau dem kollateralen, d. h. seriellen oder parallelen Betrieb mehrerer verallgemeinerter überlagernder Sendeverfahren.

Beh. Überlagern bezüglich der konzeliert ausgetauschten Schlüssel zusammenhängende Teilnehmerstationen jeweils ihre Nachrichten und die ihnen bekannten Schlüssel und geben nur das Ergebnis aus, so *erfährt* ein alle Ergebnisse beobachtender Angreifer *nur* die Summe aller gesendeten Nachrichten. Letzteres bedeutet gemäß den Abschnitten 2.1.1 und 2.2.2.2, daß für den Angreifer für alle Zufallsvariablen Z gilt: P(Z|alle Ergebnisse) = P(Z|Summe aller gesendeten Nachrichten).

Anm. Je nach Vorwissen des Angreifers kennt er damit möglicherweise auch die die Summe bildenden einzelnen Nachrichten sowie, sofern einzelne Nachrichten mit dem Vorwissen des Angreifers Rückschlüsse auf ihren Sender zulassen, den Sender dieser Nachrichten. Der Angreifer *erhält* aber durch Kenntnis aller Ergebnisse, d. h. der Ausgaben der einzelnen Teilnehmerstationen, keine über die Kenntnis ihrer Summe hinausgehende *zusätzliche* Information. Er kann sich also das Beobachten der Ausgaben der einzelnen Teilnehmerstationen sparen, da ihm das keine zusätzliche Information darüber liefert, welche Teilnehmerstation welche Nachricht gesendet hat. Gemäß der Begriffsbildung in Abschnitt 2.1.1 bleiben die Sender von Nachrichten also anonym bezüglich des Angreifers und des Ereignisses des Sendens.

Bew. O.B.d.A. wird der Beweis im folgenden auf solche Situationen beschränkt, in denen die m Teilnehmerstationen bezüglich der ausgetauschten Schlüssel *einfach* zusammenhängen. Selbiges heißt, daß es für je zwei beliebige Teilnehmerstationen T_0, T_1 *genau eine* möglicherweise leere Folge von Teilnehmerstationen T_2 bis T_n gibt, so daß für $1 \leq i \leq n$ gilt: T_i hat mit $T_{(i+1)\bmod(n+1)}$ einen Schlüssel ausgetauscht. Gibt es mehrere Folgen, so werden dem Angreifer weitere Schlüssel mitgeteilt, was ihn nur stärker macht. Er kann dann diese Schlüssel von den Ausgaben der betroffenen Teilnehmerstationen subtrahieren, wodurch er jeweils genau die „Ausgabe“ erhält, die im folgenden durch vollständiges Weglassen dieser Schlüssel entsteht.

Der Beweis selbst wird mit **vollständiger Induktion** über die Anzahl m der Teilnehmerstationen geführt.
Unter einer *Nachrichtenkombination* wird die Zuordnung von je einer Nachricht zu jeder Teilnehmerstation, unter einer *Schlüsselkombination* die Zuordnung von Werten zu allen zwischen Teilnehmerstationen konzeliert ausgetauschten und dem Angreifer unbekannten Schlüsseln verstanden. Für jedes m wird bewiesen, daß es zu jeder die Summe aller gesendeten Nachrichten ergebenden Nachrichtenkombination genau eine Schlüsselkombination gibt, so daß Nachrichtenkombination und Schlüsselkombination mit den Ausgaben aller Teilnehmerstationen verträglich sind. Da zusätzlich die Schlüssel und damit auch die Schlüsselkombinationen zufällig und gemäß einer Gleichverteilung generiert werden, liefert die Beobachtung der Ausgaben der Teilnehmerstationen dem

Angreifer keine über die Summe aller gesendeten Nachrichten hinausgehende Information gemäß den Shannon'schen Definitionen [Shan_48, Sha1_49].

Ein von T_i generierter und T_j konzeliert mitgeteilter Schlüssel werde mit $S_{i\to j}$ bezeichnet, die von T_i gesendete Nachricht mit N_i und ihre Ausgabe mit A_i, wobei nach der Beschreibung des verallgemeinerten überlagernden Sendens gilt:

$$A_i \;=\; N_i \;+\; \sum_j S_{i\to j} \;-\; \sum_j S_{j\to i}$$

Induktionsanfang: $m = 1$
Die Behauptung gilt trivialerweise, da der Angreifer genau die Ausgabe der einzigen Teilnehmerstation erfährt.

Induktionsschritt von $m - 1$ auf m:
Da die Teilnehmerstationen bezüglich der ausgetauschten Schlüssel *einfach* zusammenhängen, gibt es stets eine Teilnehmerstation, die nur mit einer anderen Teilnehmerstation durch einen ausgetauschten Schlüssel verbunden ist. Erstere werde o.B.d.A mit T_m, letztere mit T_i mit $1 \le i \le m-1$ und der Schlüssel o.B.d.A. mit $S_{i\to m}$ bezeichnet.
T_i bildet N_i und $A_i = N_i+..+S_{i\to m}$, T_m bildet N_m und $A_m = N_m-S_{i\to m}$.
Der Angreifer beobachtet das Senden von $A_1, A_2,.., A_m$.
Sei $N' = (N'_1, N'_2,.., N'_m)$ eine beliebige Nachrichtenkombination, deren Summe gleich der Summe aller gesendeten Nachrichten ist, d. h. $N'_1+N'_2+..+N'_m = N_1+N_2+..+N_m$ $(= A_1+A_2+..+A_m)$.
Es muß nun gezeigt werden, daß es zur Nachrichtenkombination N' genau eine passende Schlüsselkombination S' gibt. Der zwischen T_i und T_m ausgetauschte passende Schlüssel $S'_{i\to m}$ hat wegen $A_m = N_m-S_{i\to m}$ den Wert $S'_{i\to m} = N'_m-A_m$.
Der Rest der passenden Schlüsselkombination S' wird wie in der Induktionsvoraussetzung bestimmt, wobei hierfür als Ausgabe von T_i der Wert $A_i - S'_{i\to m}$ verwendet wird. ♦

Hiermit ist bewiesen, daß das verallgemeinerte (und damit natürlich auch das binäre) überlagernde Senden *perfekte informationstheoretische Anonymität* des Senders innerhalb der Gruppe schafft, die über rein zufällig gewählte und in perfekter informationstheoretischer Konzelation garantierender Weise ausgetauschte Schlüssel zusammenhängt. Ebenso sind Nachrichten über das Senden in keiner Weise verkettbar, überlagerndes Senden ermöglicht also auch *perfekte informationstheoretische Unverkettbarkeit* von Sendeereignissen. Erleidet eine Nachricht eine Überlagerungs-Kollision, so muß sie allerdings neu und damit anders (Ende-zu-Ende-)verschlüsselt werden – anderenfalls könnte ein Angreifer bei genügend langen Nachrichten schließen, daß Nachrichten, deren Summe vorher schon einmal gesendet wurde, wohl nicht von der gleichen Teilnehmerstation gesendet wurden, da diese anderenfalls Übertragungsbandbreite des DC-Netzes verschwenden würde.

David Chaums Beweis für das binäre und der hier gegebene für das verallgemeinerte überlagernde Senden gelten nicht nur bei *passiven*, sondern auch bei koordinierten *aktiven* Angriffen – im hier gegeben Beweis z. B. kommt das Verhalten der Angreiferstationen gar nicht vor, ihr

Verhalten ist also beliebig. (Nicht protokollgemäßes Verhalten der Angreiferstationen stellt „nur“ einen Angriff auf die Diensterbringung dar, was in Abschnitt 5.8 behandelt wird.)

Jedoch betrachten beide Beweise jeweils nur die Anonymität des *Senders*. Ist das Senden einer Nachricht *N* abhängig vom Empfangen vorhergehender Nachrichten N_i, so kann ein aktiver Angriff auf die Anonymität des *Empfängers* einer der N_i (vgl. Abschnitt 2.5.1) trotz noch so perfekten Schutzes des Sendens den Sender von *N* identifizieren. Wie bereits in Abschnitt 2.5.1 erwähnt kann dieses Problem durch geeignetes Einbeziehen der verteilten Nachrichten in die Schlüsselgenerierung selbst innerhalb der informationstheoretischen Modellwelt perfekt gelöst werden [Waid_89, WaPf1_89].

2.5.3.1.3 Überlagerndes Empfangen

Die Bandbreite des DC-Netzes kann erheblich besser genutzt werden, wenn die Teilnehmerstationen nutzen, daß wer immer das globale Überlagerungsergebnis *ü* von *n* gleichzeitig überlagernd gesendeten Nachrichten sowie *n*-1 davon einzeln kennt, die ihm noch fehlende Nachricht durch Überlagerung von *ü* mit den *n*-1 ihm einzeln bekannten Nachrichten erhalten kann. Dieses **überlagernde Empfangen** kann **global**, d. h. alle Teilnehmerstationen überlagern gleich und empfangen so dasselbe, oder **paarweise**, d. h. genau zwei Teilnehmerstationen können überlagern, geschehen:

Beim *globalen überlagernden Empfangen* speichern alle Teilnehmerstationen nach einer Überlagerungs-Kollision diese (zunächst) nicht verwertbare Nachricht ab, so daß bei *n* kollidierten Nachrichten nur *n*-1 noch einmal gesendet werden müssen: die letzte Nachricht ergibt sich durch Subtraktion (modulo Zeichenanzahl) der *n*-1 Nachrichten von der (zunächst) nicht verwertbaren Nachricht. Empfangen die Teilnehmerstationen global überlagernd, so verschwendet eine Teilnehmerstation, die hin und wieder mehrere Nachrichten gleichzeitig überlagert und alle bis auf eine noch einmal einzeln sendet, keine Bandbreite des DC-Netzes, so daß dies ohne weiteres zulässig und allgemein bekannt sein kann. Dadurch wird der obige Schluß, daß kollidierte Nachrichten nicht von derselben Teilnehmerstation stammen, falsch, so daß das überlagernde Empfangen trotz der Effizienzsteigerung Sendeereignisse bei geeignet gewählten Wahrscheinlichkeiten für das gleichzeitige Überlagern von mehreren Nachrichten informationstheoretisch unverkettbar läßt. Zusätzlich zur besseren Nutzung der Bandbreite des DC-Netzes spart das globale überlagernde Empfangen auch noch das neu und damit anders (Ende-zu-Ende-)Verschlüsseln von kollidierten Nachrichten.

Paarweises überlagerndes Empfangen ist dann möglich, wenn sich zwei möglicherweise vollständig anonyme Teilnehmerstationen (etwa mittels eines anonymen Reservierungsschemas, vgl. Abschnitt 3.1.2.3.5) mit allen anderen am überlagernden Senden Beteiligten darauf geeinigt haben, daß und wann beide exklusives Senderecht haben: Obwohl *beide* gleichzeitig senden, können beide, indem sie je das von ihnen Gesendete mit dem globalen Überlagerungsergebnis überlagern, empfangen, was der andere gesendet hat (Bild 27). Dies kann für zwei Zwecke genutzt werden: Senden beide Teilnehmerstationen je eine exklusiv für die andere Teilnehmerstation bestimmte Nachricht, so wird die Bandbreite des DC-Netzes doppelt so gut wie mit den in [Cha3_85, Chau_88, Pfi1_85] beschriebenen Zugriffsverfahren genutzt, beispielsweise kann in der Bandbreite eines Verteil-Simplex-Kanals ein Punkt-zu-Punkt-Duplex-Kanal

untergebracht werden. Sendet eine der beiden Teilnehmerstationen einen rein zufällig generierten Schlüssel, so kann sie von der anderen eine Nachricht in perfekter informationstheoretischer Konzelation erhalten, obwohl beide Teilnehmerstationen voreinander vollständig anonym sein können, wenn das zugrundeliegende DC-Netz gegenüber dem Angreifer perfekte informationstheoretische Anonymität innerhalb irgendeiner Gruppe bietet, die beide Teilnehmerstationen enthält. Letzteres ist eine besonders effiziente Implementierung des auf Sender-Anonymität basierenden Konzelations-Protokolls von Alpern und Schneider [AlSc_83], das schon vor der Erfindung von Kommunikationsnetzen mit Sender-Anonymität veröffentlicht wurde. Mit dem von Axel Burandt entworfenen Code [Bura_88] kann erreicht werden, daß beide Partner mit informationstheoretischer Sicherheit überprüfen können, ob Dritte den Nachrichtentransfer störten, so daß *perfekte informationstheoretische Integrität* erreichbar ist.

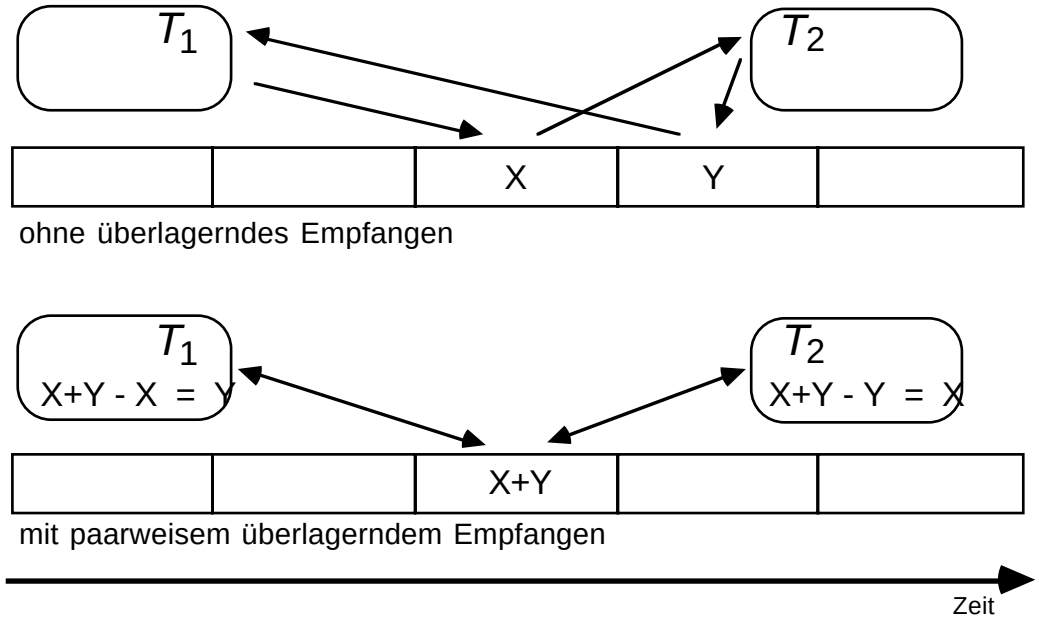


Bild 27: Paarweises überlagerndes Empfangen der Teilnehmerstationen T_1 und T_2

Das genaue Zusammenspiel von für das DC-Netz geeigneten Zugriffsverfahren und beiden Varianten des überlagernden Empfangens wird in Abschnitt 3.1.2 ausführlich behandelt.

2.5.3.1.4 Optimalität, Aufwand und Implementierungen

Da aus Gründen der Empfängeranonymität potentiell alle Teilnehmerstationen die (Ende-zu-Ende-verschlüsselten) Nachrichten erfahren sollen (und dies damit – außer beim paarweisen überlagernden Empfangen – auch jeder realistische Angreifer kann), ist das überlagernde Senden mit Austausch eines Schlüssels zwischen jedem Teilnehmerstationenpaar und neuer Verschlüsselung kollidierter Nachrichten bzw. globalem überlagerndem Empfangen mit geeignet häufigem gleichzeitigem Überlagern von mehreren Nachrichten das bezüglich Senderanonymität *optimale* Verfahren.

Der Einsatz des Verfahrens des überlagernden Sendens ist jedoch sehr, sehr aufwendig, weil große Mengen an Schlüsseln in Konzelation garantierender Weise ausgetauscht werden müssen: Jedes Paar von Teilnehmerstationen, das Schlüssel miteinander austauscht, benötigt dazu einen Konzelation garantierenden Kanal mit derselben Bandbreite, wie sie das Kommunikationsnetz allen Benutzern zusammen zum Austausch ihrer Nachrichten bereitstellt.

Diesen *Aufwand beim Schlüsselaustausch* kann man reduzieren, indem Pseudozufallszahlen-, bzw. im Falle des binären überlagernden Sendens Pseudozufallsbitfolgengeneratoren verwendet werden. Dann müssen nur relativ kurze Schlüssel geheim ausgetauscht werden, aus denen dann sehr lange Schlüssel, die äußeren Betrachtern zufällig erscheinen, erzeugt werden können.

Das Verfahren ist dann nicht mehr informationstheoretisch sicher, sondern nur noch komplexitätstheoretisch mehr oder weniger sicher; bei Verwendung kryptographisch starker Pseudozufallszahlen-, bzw. -bitfolgengeneratoren schafft das Verfahren des überlagernden Sendens *perfekte komplexitätstheoretische Anonymität* des Senders und *perfekte komplexitätstheoretische Unverkettbarkeit* von Sendeereignissen (vgl. Abschnitt 2.2.2.2).

Leider sind die bekannten schnellen Pseudozufallszahlen-, bzw. -bitfolgengeneratoren alle leicht brechbar oder zumindest von zweifelhafter Sicherheit (z. B. rückgekoppelte Schieberegister), während bisher die als kryptographisch stark bewiesenen Pseudozufallszahlen-, bzw. -bitfolgengeneratoren [VaVa_85, BlMi_84] sehr aufwendig und sehr langsam sind (vgl. Abschnitte 2.2.2.2 und 2.2.2.3).

David Chaum schlägt in [Cha3_85] vor, das überlagernde Senden *auf einem Ring zu implementieren*. Dadurch wird einem Angreifer das Brechen von schnellen (und vielleicht leicht brechbaren) Pseudozufallszahlen-, bzw. -bitfolgengeneratoren erschwert, weil er die Ausgaben von aufeinanderfolgenden Teilnehmerstationen im Ring nur durch sehr aufwendige physikalische Maßnahmen, also realistischerweise nicht erfährt (vgl. Abschnitt 2.5.3.2.1).

Bei dieser Implementierung kreist jedes Bit einmal zum Zwecke des Sendens durch sukzessives Überlagern und einmal zum Zwecke des Empfangens um den Ring.

Da diese Implementierung im Mittel nur den vierfachen Übertragungsaufwand verursacht wie ein übliches Sendeverfahren auf einem Ring, während eine Implementierung auf einem sternförmigen Netz bei n Teilnehmerstationen den n-fachen Übertragungsaufwand gegenüber einem gewöhnlichen Sendeverfahren auf einem Stern verursacht, wirkt sie recht effizient.

Da aber die Übertragungsmenge auf jeder einzelnen Leitung, also die geforderte Bandbreite, bei allen Implementierungen gleich ist, kann die Implementierung auf einem *Stern* (oder allgemeiner: *Baum*) trotzdem effizienter sein. Die Implementierung eines Kanals ist um so besser, je kürzer die Verzögerungszeit ist, also die Zeit, die zwischen Sendeversuch und der Rückmeldung, ob eine Überlagerungs-Kollision auftrat, verstreicht. Für dieses Verfahren können die Knoten von Stern- und Baumnetzen wesentlich einfacher als übliche Vermittlungszentralen sein und so entworfen werden, daß die Summe der Schaltzeiten nur logarithmisch mit der Teilnehmeranzahl wächst. Die reine Laufzeit wächst ungefähr mit der Wurzel der Teilnehmeranzahl, während beide bei Ringnetzen stets proportional zur Teilnehmeranzahl wachsen, was in Abschnitt 3.3.3 noch genauer behandelt werden wird.

2.5.3.2 Unbeobachtbarkeit angrenzender Leitungen und Stationen sowie digitale Signalregenerierung

Das aufwendige Generieren, ggf. Verteilen und Überlagern von Schlüsseln und Nutzdaten im vorherigen Abschnitt 2.5.3.1 ist nötig, da davon ausgegangen wird, daß ein Angreifer die Ein- und Ausgänge aller Teilnehmerstationen abhört bzw. die jeweils angrenzenden Stationen kontrolliert, so daß Verbindungs-Verschlüsselung gegen ihn nicht hilft.

Die Idee für weit weniger aufwendige Verfahren besteht darin, das Kommunikationsnetz bereits physikalisch (bzw. – wesentlich aufwendiger – mittels Verbindungs-Verschlüsselung) so zu gestalten, daß es für einen Angreifer nicht einfacher ist, alle Ein- und Ausgänge einer Teilnehmerstation abzuhören bzw. die jeweils angrenzenden Stationen zu kontrollieren, als den Teilnehmer bzw. die Teilnehmerstation direkt zu beobachten (vgl. Abschnitt 1.4). Wo immer der Angreifer mittels Abhören von Leitungen oder Kontrolle von Stationen Nachrichten beobachten kann, sollten diese von vielen (am besten von allen von ihm nicht kontrollierten Teilnehmerstationen) stammen und für viele (am besten für alle von ihm nicht kontrollierten Teilnehmerstationen) bestimmt sein können.

Eine wesentliche Voraussetzung hierzu ist *digitale Signalregenerierung*: Jede Station regeneriert von ihr empfangene und weitergeleitete Signale (im Falle zweiwertiger Signale: Bits) so, daß sie von entsprechenden (d. h. in der digitalen Welt gleichen) von ihr generierten Signalen (auch anhand analoger Charakteristika) nicht unterschieden werden können. Dies impliziert, daß auch einander entsprechende Signale, die eine Station von unterschiedlichen Stationen empfangen hat, nach ihrer Regenerierung nicht unterschieden werden können.

Nicht gefordert wird hingegen, daß entsprechende Signale, die von verschiedenen Stationen gesendet werden, nicht unterschieden werden können. Während letzteres wegen der analogen Charakteristika des Senders und ihrer bei jedem Produktionsprozeß unvermeidbaren Streuung eine praktisch und wegen der Änderung des Signals bei seiner Ausbreitung, z. B. breiten sich verschiedene Frequenzkomponenten verschieden schnell aus (Dispersion), auch theoretisch nicht erfüllbare Forderung wäre [Pfi1_83 Seite 17], ist das Geforderte vergleichsweise einfach zu realisieren. Dies wird an den passenden Stellen in Abschnitt 3.2 ausführlich erläutert.

In den folgenden zwei Unterabschnitten werden zwei Beispiele zu der eben geschilderten und begründeten Idee behandelt.

Das **RING-Netz** ist das Paradebeispiel, da es bei bezüglich aller Verteilnetze (für Empfängeranonymität) minimalem Aufwand die obige maximale Forderung bezüglich eines Angreifers, der eine beliebige Station kontrolliert, erfüllt. Der Aufwand des RING-Netzes ist in dem Sinne minimal, daß es aus Gründen der Empfängeranonymität nötig ist, daß jede Station jede Nachricht mindestens einmal empfängt (diese untere Grenze wird mit den in Abschnitt 3.1.4 beschriebenen effizienten anonymen Zugriffsverfahren erreicht) und daß es aus Gründen der Senderanonymität nötig ist, daß jede Station mindestens mit der Summenrate aller eigentlichen Senderaten sendet (wobei bei Punkt-zu-Punkt-Duplex-Kanälen beides ohne wesentliche Anonymitätseinbuße halbiert werden kann, wie mit dem paarweisen überlagernden Empfangen auf dem DC-Netz schon gezeigt wurde und mit einem speziellen Verfahren zum Schalten von Punkt-zu-Punkt-Duplex-Kanälen auf dem RING-Netz in Abschnitt 3.1.4 gezeigt wird). Auch diese untere Grenze wird durch die erwähnten effizienten anonymen Zugriffsverfahren bis auf

einen beliebig kleinen Verwaltungsaufwand (overhead) erreicht. Zusätzlich ist günstig, daß es im lokalen Bereich viel Erfahrung mit seiner physischen Struktur gibt und zumindest in den USA Pläne zum Einsatz seiner Struktur auch als Großstadtnetz (Metropolitan Area Network = MAN, [Sze_85, Roch_87]) bestehen.

Das **BAUM-Netz** ist ein aus pragmatischen Gründen wichtiges Beispiel: Seine physische Struktur ist die der heute üblichen Breitbandkabelverteilnetze, die nach leichten Erweiterungen (und damit in kurzer Zeit) für die anonyme Kommunikation benutzt werden können. Das BAUM-Netz bietet zwar nur bezüglich eines schwächeren Angreifers als das RING-Netz Anonymität, es kann aber wie jenes für *überlagerndes Senden*, was bezüglich jedes Angreifers stärkstmögliche Anonymität bietet, erweitert werden. Wie in Abschnitt 3.3.3 gezeigt wird, ist die Struktur des BAUM-Netzes hierfür sogar noch besser geeignet als die des RING-Netzes.

2.5.3.2.1 Ringförmige Verkabelung (RING-Netz)

Die effizienteste Möglichkeit, ein Kommunikationsnetz bereits physikalisch so zu gestalten, daß es für einen Angreifer nicht einfacher ist, alle Ein- und Ausgänge einer Teilnehmerstation abzuhören bzw. die jeweils angrenzenden Stationen zu kontrollieren, als den Teilnehmer bzw. die Teilnehmerstation direkt zu beobachten, ist, die Teilnehmerstationen ringförmig anzuordnen, wie dies im Bereich lokaler Netze seit langem praktiziert wird. Da dank digitaler Signalregenerierung in jeder Teilnehmerstation aus den Signalformen keine Information über den ursprünglichen Sender gewonnen werden kann, müssen, um bei ringförmiger Anordnung der Teilnehmerstationen das Senden einer Station zu überwachen, entweder ihre beiden Nachbarn zusammenarbeiten oder ihre beiden Leitungen abgehört werden. Durch bauliche Maßnahmen, z. B. direkte Verkabelung von Wohnungen in Mehrfamilienhäusern sowie direkte Verkabelung aneinandergrenzender privater Grundstücke [Pfi1_83 Seite 18], ergänzt, keineswegs aber allein durch Verwendung eines möglichst schwierig abhörbaren Übertragungsmediums (Glasfaser, vgl. Abschnitt 1.2), kann man erreichen, daß letzteres ebenso aufwendige physikalische Maßnahmen erfordert wie das direkte Abhören des Teilnehmers oder der Teilnehmerstation innerhalb der Wohnung. Damit verspricht es keinen zusätzlichen Vorteil. Versuchen die zwei Ringnachbarn eines Teilnehmers, diesen ohne Auftrag Dritter (d. h. hier ohne Zusammenarbeit mit großen Kommunikationspartnern) zu beobachten, so erfahren sie beinahe nichts, da alle ausgehenden Nachrichten verschlüsselt und die Adressen bei geeigneter Verwendung impliziter Adressierung für sie nicht interpretierbar sind.

Zu erwarten sind also im wesentlichen Angreifer, die nur eine Gruppe von vielen zusammenhängenden Stationen eingekreist haben. Sie können durch Vergleich der ein- und auslaufenden Signalmuster (für preiswerte Ringe: Bitmuster) nur feststellen, welche davon von den Mitgliedern dieser Gruppe gesendet bzw. vom Ring entfernt wurden, nicht aber von welchem Mitglied genau. Es gilt sogar die stärkere Aussage, daß diese Angreifer kein Mitglied der Gruppe als Sender oder Entferner der Signalmuster ausschließen können. (Diese Behauptungen gelten per Definition der digitalen Signalregenerierung in Verbindung mit der Ring-Topologie trivialerweise. Wer trotzdem einen „Beweis“ sehen möchte, sei auf Abschnitt 3.1.4 vertröstet, wo er bei Modellierung und Beweis eines komplizierteren Sachverhaltes mit abfällt.)

Senden Stationen auf dem gemeinsamen Verteil-Kanal „Ring“ unkoordiniert, so kann es, wie in Abschnitt 2.5.3.1 für das DC-Netz bereits erwähnt, Kollisionen von Nachrichten geben. Im Gegensatz zum DC-Netz und dem im folgenden Abschnitt 2.5.3.2.2 beschriebenen BAUM-Netz ist die Sichtweise von Empfänger und Sender einer Nachricht bezüglich einer eventuellen Kollision im RING-Netz nicht naturgegebenermaßen konsistent – die Kollision kann „nach“ dem Empfänger auftreten, so daß nur der Sender sie bemerkt. (Fehler, die auch beim DC- und BAUM-Netz eine inkonsistente Sicht von Empfänger und Sender bezüglich einer eventuellen Kollision verursachen können, seien hierbei zunächst einmal außer Betracht gelassen – sie treten hoffentlich um Größenordnungen seltener als Kollisionen auf und werden deshalb separat in Kapitel 5 behandelt.)

Unter der (realistischen) Annahme, daß nie zwei verschiedene Nachrichten dieselbe Codierung haben, kann der Sender aber feststellen, daß der Empfänger die Nachricht empfangen konnte. Dies ist immer dann der Fall, wenn der Sender seine Nachricht nach einem Ringumlauf unverändert zurückerhält (und er sie mit einer anderen Nachricht oder 0 überschreiben kann). Durch die globale Verabredung, daß Empfänger Nachrichten, die sie schon erhalten haben, für alle Zukunft ignorieren (durchaus nichts Ungewöhnliches in Rechnernetzen, wenn Nachrichten einen Zeitstempel enthalten), kann sich der Sender durch ggf. wiederholtes Senden Gewißheit über den Empfang seiner Nachricht verschaffen. Auf diese Art kann für Sender und Empfänger eine konsistente Sicht über die Tatsache des Empfangs hergestellt werden, nicht aber über den genauen Zeitpunkt. Sollte dieser wichtig sein oder aber eine stochastische Angriffsmöglichkeit zur Ermittlung des Senders einer Nachricht über ihre Senderate vermieden werden, so sind entweder geschicktere Zugriffsprotokolle oder eine andere Gestaltung des RING-Netzes nötig.

Einerseits bietet ein Ring im Gegensatz zum DC-Netz durch seine Struktur eine effiziente Möglichkeit, durch spezielle Verfahren zum Mehrfachzugriff, nämlich verteiltes und (wie in Abschnitt 3.1.4 gezeigt wird) anonymes Abfragen (distributed and unobservable polling [Pfi1_85 Seite 19]), Kollisionen vollständig zu vermeiden, was den gerade beschriebenen Nachteil einer möglicherweise inkonsistenten Sicht bezüglich Kollisionen und damit auch bezüglich des Zeitpunktes des Nachrichtenempfangs mehr als kompensiert. Sowohl das Senden von Nachrichten als auch das Schalten von Kanälen ist mit Hilfe dieser Zugriffsverfahren möglich. Bei Punkt-zu-Punkt-Duplex-Kanälen kann man sogar ohne große Anonymitätseinbußen darauf verzichten, daß alle Information einmal ganz um den Ring läuft. Statt dessen nimmt der Empfänger die Information vom Ring und ersetzt sie sofort durch Information in Gegenrichtung. Dadurch ist die Kapazität des Rings doppelt so gut nutzbar. All dies wird in Abschnitt 3.1.4 ausführlicher erläutert.

Andererseits bietet ein Ring auch bei unkoordiniertem (und damit – wie in Abschnitt 3.1.4 erklärt wird – noch etwas anonymerem) Senden zwei Möglichkeiten, eine (abgesehen von Fehlern) konsistente Sicht von Empfänger und Sender bezüglich Empfang und Zeitpunkt des Empfangs einer Nachricht herstellen. Hierzu läuft die Nachricht zweimal um den Ring. Während des ersten Umlaufs können Kollisionen auftreten, wohingegen der zweite Umlauf zur Verteilung des Ergebnisses des ersten ohne Kollisionen abgewickelt werden muß und kann. Die beiden Möglichkeiten unterscheiden sich darin, ob die Umläufe bei einer festen und damit statisch ausgezeichneten Station oder jeweils beim Sender und damit bei einer nur dynamisch ausgezeichneten Station beginnen. Die statisch ausgezeichnete Station darf global bekannt sein, während dynamisch ausgezeichnete Stationen dies aus Gründen der Senderanonymität tunlichst nicht sein

sollten. Während die erste Möglichkeit eher eine Modifikation des RING-Netzes darstellt und deshalb als Schluß dieses Abschnitts beschrieben wird, stellt die zweite lediglich ein spezielles Verfahren zum Mehrfachzugriff dar und wird deshalb erst in Abschnitt 3.1.4.4 genauer beschrieben.

RING-2-f-Netz (RING-Netz mit 2 Umläufen, die bei einer festen Station beginnen): Da der erste Umlauf bei einer festen Teilnehmerstation beginnt, gibt es eine erste Teilnehmerstation T_1, die senden kann. Da dies statisch festgelegt ist, muß davon ausgegangen werden, daß es auch einem Angreifer bekannt ist. Damit nicht die auf T_1 folgende Teilnehmerstation T_1 bezüglich ihres Sendens beobachten kann, sendet T_1 einen zufällig und gemäß einer Gleichverteilung generierten Schlüssel S als Ausgabe A_1. Die anderen Teilnehmerstationen T_2 bis T_m können eine Nachricht N_i ($2 \le i \le m$) senden, indem sie sie überlagern (vgl. Abschnitt 2.5.3.1.1), d.h. $A_i = A_{i-1}+N_i$ ausgeben. Um für ihre Nachrichten keine unnötig langen Verzögerungszeiten zu erhalten, überlagert T_1 erst nach Eintreffen von A_m als letzter ihre Nachricht N_1 und subtrahiert danach den (im Gegensatz zum DC-Netz einzigen) Schlüssel S. T_1 sendet das Ergebnis (die globale Ausgabe A = Summe aller Nachrichten) an T_2 bis T_m, was wie beim überlagernden Senden auf beliebige Art und Weise geschehen kann, etwa indem die Kapazität des Ringes in zwei gleich große Teile geteilt wird und T_1 die Nachricht vom Sendeteil in den Empfangsteil umsetzt (vgl. Bild 28).

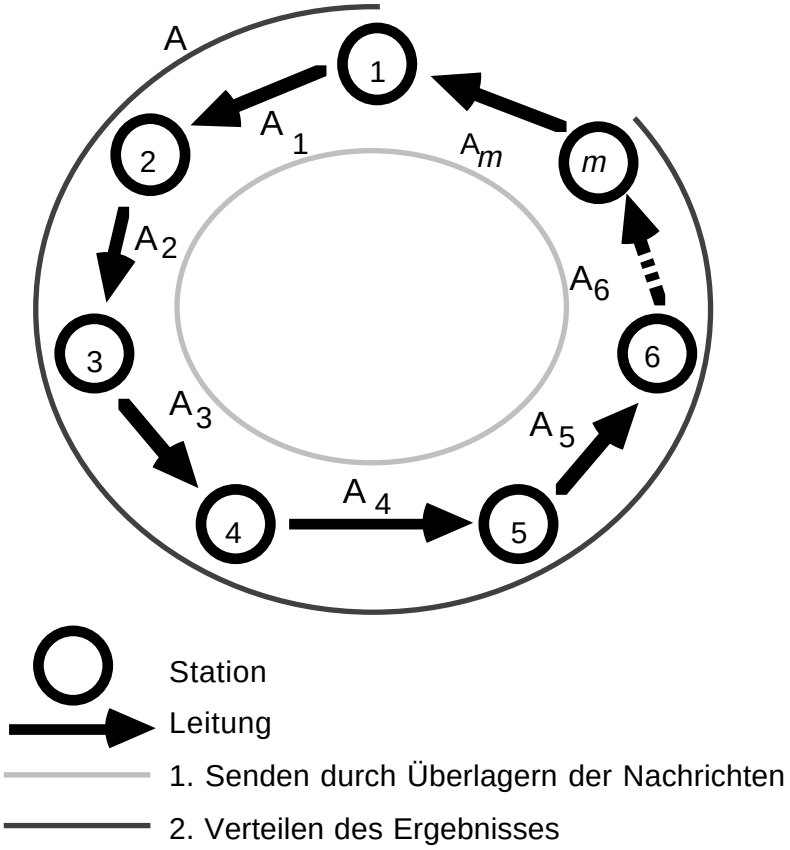


Bild 28: RING-2-f-Netz mit Verteilung des Ergebnisses durch zweiten Ringumlauf

Es ist bemerkenswert, daß der Schlüssel *S* bei diesem Verfahren nicht ausgetauscht werden muß, so daß auch für breitbandige Kommunikationsnetze kein Rückgriff auf Pseudozufallszahlen nötig ist! Es genügt völlig, daß T_1 *S* nach seiner physischen Erzeugung und erstmaligen Verwendung z. B. in ein Schieberegister schreibt, dessen Verzögerung der des Ringes entspricht, und jeweils den am Ausgang des Schieberegisters anliegenden Wert von A_m+N_1 subtrahiert.

Wie der gleich folgende Beweis zeigt, ist es für die Senderanonymität wichtig, daß T_1 ihre Entscheidung, ob sie eine Nachricht sendet, nicht davon abhängig macht, ob eine andere Teilnehmerstation vor ihr gesendet hat. Dies könnte sie leicht, indem sie von A_m erst *S* subtrahiert und nur dann eine Nachricht sendet, wenn das Ergebnis Null ist. Macht T_1 ihr Senden nicht vom Ergebnis der Subtraktion von *S* abhängig, kann sie natürlich auch zuerst *S* subtrahieren und danach ihre Nachricht überlagern. Dies verkürzt die Verzögerungszeiten ihrer Nachrichten abermals, diesmal aber nur sehr geringfügig.

Beh. Beobachtet ein Angreifer eine bezüglich der Ringtopologie zusammenhängende Gruppe von Teilnehmerstationen T_i bis T_j $(1 \le i \le j \le m)$, indem er alle Ausgaben mit echt kleinerem Index als *i*, d. h. A_1 bis A_{i-1}, und mit Index größer gleich *j*, d. h. A_j bis A_m und *A* beobachtet, so erfährt er über das Senden dieser Teilnehmerstationen *nur* die Summe ihrer gesendeten Nachrichten, also $N_i + ... + N_j$.

Anm. Der Angreifer erfährt über das Senden der Teilnehmerstationen T_i bis T_j also genau das, was er erführe, wenn diese miteinander rein zufällig gewählte Schlüssel in perfekter informationstheoretischer Konzelation garantierenden Weise ausgetauscht hätten und mit ihnen überlagerndes Senden praktizieren würden. Jede vom Angreifer nicht abgehörte Leitung des RING-2-f-Netzes entspricht also einem ausgetauschten Schlüssel bezüglich des überlagernden Sendens. Anders herum ausgedrückt: der perfekte informationstheoretische Konzelation garantierende Kanal „Leitung“ würde es den bezüglich der Ringtopologie aneinandergrenzenden Teilnehmerstationen erlauben, einen Schlüssel auszutauschen, den sie dann zum überlagernden Senden auf einem beliebigen anderen Kommunikationsnetz verwenden könnten.

Da, wie in Abschnitt 2.5.3.1.4 begründet, das DC-Netz bezüglich Senderanonymität das optimale Netz ist, ist das RING-2-f-Netz bezüglich der Senderanonymität innerhalb der betrachteten Gruppe ebenfalls optimal. Es ist im Gegensatz zum DC-Netz natürlich nicht optimal schlechthin, da bei diesem Schlüsselaustauschgraphen beliebigen Zusammenhangs möglich sind, während der „Schlüsselaustauschgraph“ des RING-2-f-Netzes nur den Zusammenhang 2 hat, also bezüglich eines Außenstehenden nur die Kompromittierung eines „Schlüssels“, sprich das Abhören einer Leitung bzw. bezüglich eines Teilnehmers die Kontrolle einer Teilnehmerstation ohne Anonymitätseinbuße verkraftet.

Bew. Der Beweis erfolgt, indem jeweils alle durch das RING-2-f-Netz gegebenen Gleichungen, in denen von der Behauptung betroffene Nachrichten enthalten sind, aufgestellt werden und durch deren Äquivalenzumformung gezeigt wird, daß die Behauptung gilt. Da T_1 im RING-2-f-Netz eine Sonderrolle spielt, ist eine Fallunterscheidung danach nötig, ob die von T_1 gesendete Nachricht von der Behauptung betroffen ist.

Für $1 < i \leq m$ gilt nach dem RING-2-f-Netz:

$$A_j = A_{i-1} + N_i + N_{i+1} + ... + N_j$$

Bringt man die Werte, die der Angreifer gemäß Angreifermodell erfährt, per Äquivalenzumformung auf die linke Seite, erhält man die Behauptung:

$$A_j - A_{i-1} = N_i + N_{i+1} + ... + N_j$$

Für $1 = i$ gilt nach dem RING-2-f-Netz:

$$A_j = S + N_2 + N_3 + ... + N_j$$

$$A = A_m + N_1 - S$$

Per Äquivalenzumformung (untere Gleichung nach S auflösen und in obere einsetzen) erhält man

$$A_j = A_m - A + N_1 + N_2 + N_3 + ... + N_j$$

Bringt man die Werte, die der Angreifer gemäß Angreifermodell kennt, auf die linke Seite, erhält man die Behauptung:

$$A_j + A - A_m = N_1 + N_2 + N_3 + ... + N_j \qquad \blacklozenge$$

Hiermit ist bewiesen, daß das RING-2-f-Netz für das beschriebene Angreifermodell *perfekte informationstheoretische Anonymität* des Senders innerhalb der Gruppe schafft, die über nicht abgehörte Leitungen zusammenhängt. Ebenso sind Nachrichten über das Senden in keiner Weise verkettbar, das RING-2-f-Netz ermöglicht also auch *perfekte informationstheoretische Unverkettbarkeit* von Sendeereignissen. Bezüglich einer neuen und damit anderen (Ende-zu-Ende-)Verschlüsselung kollidierter Nachrichten bzw. überlagerndem Empfangen gilt das in Abschnitt 2.5.3.1.2 Gesagte.

Wie die Verfahrensbeschreibung und der Beweis zeigen, können Stationen im ersten Umlauf beliebig *paarweise Schlüssel überlagern*, ohne daß dadurch die Arbeitsmöglichkeiten eines Zugriffsverfahrens oder die Empfangsmöglichkeiten eingeschränkt und damit der erzielbare Durchsatz und die erzielbare Verzögerungszeit verschlechtert werden. Genauso kann die Überlagerung von paarweise ausgetauschten Schlüsseln die Anonymität des Senders natürlich nicht vermindern, so daß beim RING-2-f-Netz auch der Einsatz von schnellen (und vielleicht leicht brechbaren) Pseudozufallszahlen-, bzw. -bitgeneratoren erwogen werden sollte. Erfolgt die Überlagerung von paarweise ausgetauschten Schlüsseln, dann unterscheidet das RING-2-f-Netz nur noch die Verwendung des (echt) zufällig und gemäß einer Gleichverteilung generierten Schlüssels S, der von derselben Teilnehmerstation zweimal überlagert wird und deshalb nicht ausgetauscht zu werden braucht, vom im Abschnitt 2.5.3.1.4 erwähnten und in Abschnitt 3.3.3 noch genauer diskutierten Implementierungsvorschlag für das DC-Netz von David Chaum. Wie der Beweis zeigt, ist dieser Schlüssel für die Senderanonymität essentiell, wenn keine oder nur dem Angreifer bekannte Schlüssel (weil er etwa die schnellen Pseudozufallszahlen-, bzw. -bitgeneratoren gebrochen hat) paarweise überlagert werden.

Während für „normale“ Ringe, d. h. für Ringe mit „Senden durch Ersetzen“, geeignete Zugriffsprotokolle – wie schon erwähnt – in Abschnitt 3.1.4 diskutiert werden, müssen für das RING-2-f-Netz, d. h. einen Ring mit „Senden durch Überlagern“, die in Abschnitt 3.1.2 beschriebenen Zugriffsprotokolle des DC-Netzes verwendet werden. Sie haben zwar die Möglichkeit zum in Abschnitt 2.5.3.1.3 beschriebenen überlagernden Empfangen, nicht aber die zum

anonymen Abfragen. Da beim RING-2-f-Netz im Mittel ein Bit erst nach einem ganzen Umlauf empfangen werden kann, während dies beim RING-Netz im Mittel schon nach einem halben Umlauf der Fall ist, ist die Verzögerungzeit beim RING-2-f-Netz bei jedem Zugriffsprotokoll im Mittel um mindestens diese Zeit größer. Ist die Alphabetgröße der Überlagerung beim RING-2-f-Netz groß genug, so kann mit dem besten bekannten Zugriffsprotokoll für überlagerndes Senden bei geringer Last (abgesehen von der unvermeidbaren Verzögerungszeit) eine gleich gute Verzögerungzeit und bei hoher Last ein gleich guter Durchsatz (mit allerdings durch das überlagernde Empfangen bedingten etwas höherer mittlerer Verzögerungszeit) erzielt werden. Die Zugriffsprotokolle von Abschnitt 3.1.2 erhalten dafür die Anonymität des Senders vollständig und nicht, wie das anonyme Abfragen, nur fast vollständig. Außerdem sorgen sie in stärkerem Maße für eine faire Aufteilung der Bandbreite bei Überlast.

2.5.3.2.2 Kollisionen verhinderndes Baumnetz (BAUM-Netz)

Fast alle im Teilnehmeranschlußbereich bereits vorhandenen Breitbandkabel sind baumförmig verlegt. Ein aus pragmatischen Gründen, nämlich ihrer Benutzung, wichtiges Beispiel für die Idee „Unbeobachtbarkeit angrenzender Leitungen und Stationen sowie digitale Signalregenerierung“ ist deshalb ein Kollisionen verhinderndes Baumnetz (BAUM-Netz) [Pfit_86 Seite 357].

Die inneren Knoten des Baumnetzes werden mit **Kollisionen verhindernden Schaltern** (collision-avoidance circuits in [Alba_83], collision-avoidance switches in [SuSY_84]) ausgerüstet. Ein Kollisionen verhindernder Schalter schaltet einen aus Richtung der Blätter kommenden Informationsstrom nur dann in Richtung der Wurzel durch, wenn der Übertragungskanal in Richtung Wurzel frei ist. Bewerben sich mehrere Informationsströme aus Richtung der Blätter gleichzeitig um den freien Übertragungskanal in Richtung Wurzel, wählt der Kollisionen verhindernde Schalter genau einen zufällig aus und ignoriert die anderen. Sind Teilnehmerstationen auch direkt an Kollisionen verhindernde Schalter angeschlossenen, so daß es auch innere Teilnehmerstationen des BAUM-Netzes gibt, so werden von diesen kommende Informationsströme genauso wie aus Richtung der Blätter kommende behandelt.

Der von der Wurzel des Baumes durchgeschaltete Informationsstrom wird an alle Stationen verteilt, d. h. die Wurzel des Baumes und die Kollisionen verhindernden Schalter senden ihn auf allen Leitungen in Richtung der Blätter.

Um das Senden einer Teilnehmerstation an den Blättern des BAUM-Netzes beobachten zu können, muß eine spezielle Leitung beobachtet werden (beim RING-Netz müssen für alle Stationen je zwei spezielle Leitungen beobachtet werden); um das Senden einer inneren Teilnehmerstation des BAUM-Netzes beobachten zu können, müssen alle ihre Eingänge und ihr Ausgang beobachtet werden. Da üblicherweise fast alle Teilnehmerstationen Blätter des BAUM-Netzes sind und ein Angreifer durch Abhören einer Leitung oder durch Kontrolle eines Kollisionen verhindernden Schalters in jedem Fall das BAUM-Netz bezüglich der Senderanonymität partitioniert, ist die durch das BAUM-Netz realisierte Senderanonymität geringer als die des RING-Netzes. Wie bei diesem gilt auch hier, daß, wer ohne Auftrag Dritter (d. h. hier ohne Zusammenarbeit mit großen Kommunikationspartnern) versucht, seine Nachbarn zu beobachten, so

gut wie nichts erfährt, da alle Nachrichten verschlüsselt und die Adressen bei geeigneter Verwendung impliziter Adressierung für ihn nicht interpretierbar sind.

Leistungsbewertungen [Alba_83, SuSY_84] attestieren mit Kollisionen verhindernden Schaltern ausgerüsteten Baumnetzen ein hervorragendes Leistungsverhalten auch bei völlig unkoordiniertem Zugriff.

Soll das BAUM-Netz, wie in Abschnitt 2.5.3.2 schon erwähnt, durch *überlagerndes Senden* erweitert werden, so müssen die Kollisionen verhindernden Schalter durch modulo-Addierer ersetzt werden. Wie in Abschnitt 2.5.3.2.1 in der Anmerkung des Beweises für das RING-2-f-Netz ausführlich erklärt wurde, entspricht jeder vom Angreifer nicht beobachteten Leitung ein ausgetauschter Schlüssel bezüglich des überlagernden Sendens, was auch beim BAUM-Netz – wie bei jedem Netz mit digitaler Signalregenerierung und auf die Übertragungstopologie abgestimmter Überlagerungstopologie – gilt. Wie beim RING-2-f-Netz müssen auch beim BAUM-Netz mit überlagerndem Senden die in Abschnitt 3.1.2 beschriebenen Zugriffsprotokolle des DC-Netzes verwendet werden. Da die Kollisionen verhindernden Schalter bezüglich der Anonymität des Senders bezogen auf einen statisch definierten, d. h. zeitinvarianten Angreifer ein optimales „Zugriffsverfahren“ darstellen, sind die in Abschnitt 3.1.2 beschriebenen Zugriffsprotokolle in ihrer Anonymitäts*erhaltung* nicht besser – allerdings ist die Senderanonymität in einem Netz mit überlagerndem Senden weitaus höher.

Bei Verwendung von modulo-Addierern statt Kollisionen verhindernden Schaltern kann mittels überlagerndem Empfangen nicht nur bei allen Verkehrsarten ein gleich guter Durchsatz, sondern bei manchen Verkehrsarten sogar mittels paarweisem überlagerndem Empfangen der doppelte Durchsatz wie mit Kollisionen verhindernden Schaltern erzielt werden, vgl. Abschnitte 2.5.3.1.3 und 3.1.2.5. Dies spricht dafür, Kollisionen verhindernde Schalter als durch das überlagernde Empfangen „überholt“ zu betrachten.

2.6 Einordnung in ein Schichtenmodell

Um den Entwurf, das Verständnis, die Implementierung und die Verbindung von Kommunikationsnetzen zu erleichern, werden sie als geschichtete Systeme entworfen. Jede Schicht benutzt die Dienste der nächst tieferen Schicht sowie durch ihr sogenanntes Protokoll reglementierte Kommunikation zwischen ihren Instanzen, um der nächst höheren Schicht einen komfortableren Dienst anzubieten. Die Internationale Normungsbehörde (International Standards Organization, abgekürzt ISO) normte ein sieben Schichten Modell, das „Grundlegende Referenzmodell für die Verbindung offener Systeme“ (Basic reference model for Open Systems Interconnection, abgekürzt OSI), dessen Schichten – wenn nötig – verfeinert werden, etwa um es Lokalen Netzen anzupassen. Da alle internationale Normungsarbeit sich auf dieses ISO OSI Referenzmodell bezieht, geben die folgenden beiden Bilder die passenden Schichten für Ende-zu-Ende- und Verbindungs-Verschlüsselung bzw. die Grundverfahren zum Schutz der Verkehrs- und Interessensdaten an.

Um beispielsweise Vermittlungszentralen keine unnötige Protokollinformation zugänglich zu machen, muß Ende-zu-Ende-Verschlüsselung in der tiefsten Schicht erfolgen, deren Proto-

koll Ende-zu-Ende, d. h. direkt zwischen den Teilnehmerstationen der Kommunikationsnetzbenutzer arbeitet. Deshalb ist Schicht 4 (Transportschicht, transport layer) die für Ende-zu-Ende-Verschlüsselung geeignete Schicht.

Da Angreifer, die Leitungen (oder allgemeiner: Kommunikationskanäle) abhören, auch keine unnötige Protokollinformation erhalten sollten, muß Verbindungs-Verschlüsselung in der tiefsten Schicht erfolgen, die mit digitalen Daten (im Gegensatz zu analogen Signalen) arbeitet. Deshalb ist Schicht 1 (Bitübertragungsschicht, physical layer) die für Verbindungs-Verschlüsselung geeignete Schicht. Auf Kommunikationskanälen, die jeweils zwei Instanzen der Schicht 1 exklusiv zugeordnet sind, kann – wie in den Abschnitten 2.2.2.1 und 2.2.2.2 beschrieben – durch Einsatz einer Stromchiffre vor dem Angreifer sogar ohne zusätzliche Kosten verborgen werden, ob und wieviel Information auf dem Kommunikationskanal fließt.

Diese Schichten (aber leider noch viele andere) werden in [ISO7498SA_86, ISO7498-2_87] als mögliche Implementierungsorte für Ende-zu-Ende- und Verbindungs-Verschlüsselung genannt. Ich hoffe, daß die obigen Argumente dazu führen, daß sie (und nicht nur jeweils höher liegende oder gar keine) als Implementierungsort gewählt werden, auch wenn sich manche Implementierer davon eine leichtere oder schnellere Implementierung und alle Geheimdienste (vgl. Abschnitt 2.2.2.4) umfassendere Arbeitsmöglichkeiten versprechen mögen.

Mißtraut man der Implementierung einzelner (Teil)Schichten bezüglich Datenschutz oder Sicherheit (vgl. Abschnitt 1.2), sollten die Nutzdaten dieser Schicht von der nächst höheren zum Zwecke der Konzelation oder Integrität (vgl. Abschnitt 2.2.1) verschlüsselt werden. Hierdurch können zusätzliche Ende-zu-Ende- oder Verbindungs-Verschlüsselungen entstehen. Letzteres erscheint weniger notwendig, da die Entwurfskomplexität tieferer (Teil)Schichten geringer als die höherer (Teil)Schichten und unter anderem deshalb Fernwartung (vgl. Abschnitt 1.2) überflüssig und nicht vorgesehen ist.

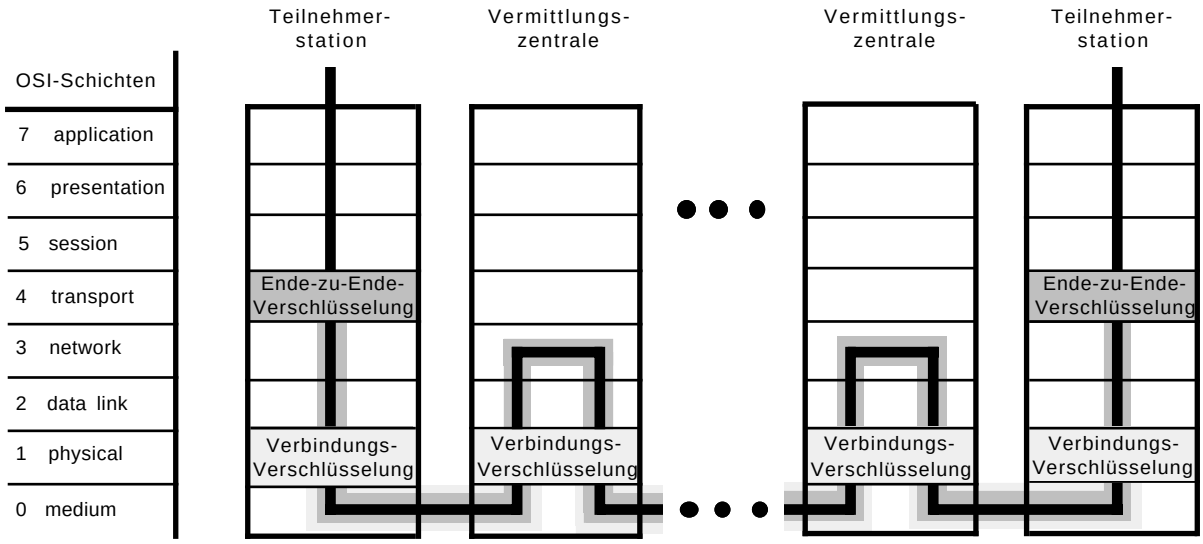


Bild 29: Einordnung der Ende-zu-Ende- und Verbindungs-Verschlüsselung in das ISO OSI Referenzmodell

Das Grundverfahren zum Schutz des Empfängers verwendet Verteilung (broadcast).

Verteilung kann in einem beliebigen Kommunikationsnetz durch geeignete Wegeermittlung (routing) in Schicht 3 (Vermittlungsschicht, network layer) realisiert werden, so daß Schicht 4 nur noch implizite Adressen generieren und auswerten muß. Eine mögliche Realisierung von Verteilung ist Überflutung (flooding [Tane_81]), die mit einer sehr einfachen Wegeermittlungs-Strategie erreicht werden kann: jede Station überträgt jede Nachricht an alle Nachbarstationen, von denen sie diese Nachricht (noch) nicht empfangen hat.

In speziell für Verteilung entworfenen Kommunikationsnetzen wird Verteilung effizienter durch Schicht 1 realisiert. Zumindest für manche Dienste kann Schicht 1 auch gleich die Information auswählen, die für die Teilnehmerstation bestimmt ist. Dies wird in Bild 30 Kanalselektion genannt und in den Abschnitten 3.1.1 und 3.2.1 ausführlich behandelt.

MIX- und DC-Netz verwenden spezielle kryptographische Mechanismen, um innerhalb des Kommunikationsnetzes Anonymität und Unverkettbarkeit zu schaffen (bei einer globalen Sicht wäre der Terminus „Anonymität und Unverkettbarkeit schaffen“ ein Widerspruch in sich, da einem Angreifer natürlich durch die Gestaltung des Kommunikationsnetzes keine Information weggenommen werden kann, so daß bei globaler Sicht alle Maßnahmen nur Anonymität und Unverkettbarkeit erhalten können, vgl. Abschnitt 2.1.1). Diese Mechanismen ordne ich in die höchsten Teilschichten (sublayers) von Schicht 3 bzw. 1 ein, da

- das MIX-Netz die Wegeermittlung (routing) von (den tieferen Teilschichten von) Schicht 3 nutzt, aber keine Ende-zu-Ende-Maßnahme ist, die in die Schicht 4 einzuordnen wäre;
- das DC-Netz die Übertragung von Zeichen (heutzutage vor allem: Bits) von (den tieferen Teilschichten der) Schicht 1 nutzt, aber nicht mit der Entdeckung von Übertragungsfehlern, Mehrfachzugriff oder anderen Diensten der Schicht 2 (Sicherungsschicht, data link layer) befaßt ist.

RING- und BAUM-Netz – wie alle nach der Idee „Unbeobachtbarkeit angrenzender Leitungen und Stationen sowie digitale Signalregenerierung“ funktionierenden Kommunikationsnetze – schaffen innerhalb des Kommunikationsnetzes Anonymität, indem sie ein ring- bzw. baumförmiges – allgemeiner: passend förmiges – Medium in Schicht 0 (medium) und digitale Signalregenerierung in Schicht 1 verwenden.

Beim RING-2-f-Netz gehört die Addition bzw. Subtraktion des Schlüssels (wie beim DC-Netz) in die höchste Teilschicht von Schicht 1.

Bild 30: Einordnung der Grundverfahren zum Schutz der Verkehrs- und Interessensdaten in das ISO OSI Referenzmodell

Hieraus folgt, daß Verteilung als Grundverfahren zum Schutz des Empfängers beliebig implementierte Schichten 0, 1, und 2 ohne Effizienzeinbuße nutzen kann, sofern eine Implementierung in einem nicht speziell für Verteilung entworfenen Kommunikationsnetz erfolgt, oder nur eine beliebig implementierte Schicht 0, sofern eine Implementierung in einem speziell für Verteilung entworfenen Kommunikationsnetz erfolgt.

Das MIX-Netz kann beliebig implementierte Schichten 0, 1, 2, und tiefere Teilschichten von Schicht 3 ohne Effizienzeinbuße nutzen, das DC-Netz eine beliebig implementierte Schicht 0 und tiefere Teilschichten von Schicht 1.

Alle diese (Teil)Schichten können ohne Einschränkungen durch Anonymitätsanforderungen implementiert werden, so daß in ihnen alle Verfahren zur Steigerung von Leistung und Zuverlässigkeit eingesetzt werden können.

Alle Schichten oberhalb der (innerhalb des Kommunikationsnetzes) Anonymität und Unverkettbarkeit schaffenden (Teil)Schichten müssen die **Anonymität** und ggf. auch die **Unverkettbarkeit erhalten**. Dies bedeutet, daß keine (oder zumindest nicht allzu viele) der auf der Anonymität und Unverkettbarkeit schaffenden Schicht möglichen und vom Angreifer ununterscheidbaren Alternativen von den höheren Schichten ausgeschlossen (oder verkettet) werden. Anderenfalls könnte ein Angreifer das allgemein verfügbare Wissen über die höheren Schichten verwenden, um Sender oder Empfänger zu identifizieren oder sie (oder Sendeereignisse) miteinander in Beziehung zu setzen. Beispielsweise müssen die Protokolle zum anonymen Mehrfachzugriff nicht nur die Bandbreite des gemeinsamen Kanals des DC-, RING-, oder BAUM-Netzes effizient nutzen, sondern auch die Anonymität der Sender (sowie möglichst die Unverkettbarkeit von Sendeereignissen) erhalten.

Für die Normung der Protokolle dieser Schichten hat dies Konsequenzen, die von den zuständigen Normungsgremien entweder noch nicht als solche erkannt oder aber ignoriert werden. Wenn eine Norm nicht eine Teilmenge mit hinreichender Funktionalität enthält, die die Anonymität (und ggf. auch die Unverkettbarkeit) erhält, ist sie für Kommunikationsnetze mit überprüfbarem Datenschutz nutzlos. Folglich sind solche internationalen Normen in Ländern, deren

Verfassung oder Gesetze Datenschutz vorschreiben, nicht anwendbar, und nationale Normen möglicherweise illegal. Insbesondere die erste Möglichkeit und die Tatsache, daß das Ändern von Normen sehr lange dauert, kann in der Zukunft internationale Kommunikation wesentlich behindern. Außerdem ist es eine offene Frage, ob Menschen in anderen Ländern gewillt sind, ein Kommunikationsnetz zu benutzen, das sie leicht beobachten kann.

Während es aus den gerade geschilderten Gründen nötig ist, daß die Protokolle der Schichten oberhalb der (innerhalb des Kommunikationsnetzes) Anonymität und Unverkettbarkeit schaffenden (Teil)Schichten die Anonymität der Beteiligten und möglichst auch die Unverkettbarkeit der Sendeereignisse erhalten, so sind insbesondere für letzteres bezüglich der Beteiligten sinnvolle Grenzen durch die Natur der zu erbringenden Dienste vorgegeben. Beispielsweise kann vor den an einem Telefongespräch Beteiligten nicht verborgen werden, daß sich Sätze und damit auch die sie transportierenden Informationseinheiten, seien es nun durch Kanalvermittlung geschaltete kontinuierliche Bitströme oder nur Sprachpakete, wenn tatsächlich gerade etwas zu hören ist (TASI = time assignment speech interpolation [Amst_83, LiFl_83, Rei1_86]), in beiden Kommunikationsrichtungen aufeinander beziehen. Allgemeiner ausgedrückt: Es ist nicht möglich, vor Beteiligten mehr Anonymität oder Unverkettbarkeit von Kommunikationsereignissen zu schaffen als der gerade abgewickelte Dienst zuläßt. Die Forderung, daß einem Ereignis der höheren Schicht möglichst alle Ereignisse der tieferen Schicht entsprechen und daß Ereignisse möglichst nicht verkettet werden können, bezieht sich also ausschließlich auf die Unbeobachtbarkeit durch Unbeteiligte. Wenn die zu berücksichtigenden Angreifer entweder selbst Beteiligte sind oder die Kooperation von Beteiligten haben, so kann ggf. erheblicher Aufwand gespart werden, wenn gar nicht erst versucht wird, einem Angreifer die Verkettung von über den Dienst bereits verketteten Verkehrsereignissen unmöglich zu machen.

An dieser Stelle soll noch ein universeller **aktiver Verkettungsangriff über Betriebsmittelknappheit** beschrieben werden: Ist die

- lokale *Rechenleistung* von Stationen begrenzt oder die
- anonym und unverkettbar durch *Senden* oder *Empfangen* nutzbare Bandbreite des Kommunikationsnetzes pro Station auf einen echten Bruchteil der Gesamtbandbreite beschränkt,

so kann eine Station Dienstanforderungen, die diese Grenzen überschreiten, natürlich nicht in Realzeit erfüllen oder gar gar nicht gleichzeitig empfangen. Hieraus resultieren drei Angriffsarten. Sie haben gemeinsam, daß von zwei unterschiedlichen Pseudonymen (z. B. impliziten Adressen) jeweils Dienste angefordert werden, die nur erbracht werden können, wenn die beiden Pseudonyme unterschiedlichen Stationen gehören. Hierzu muß der Angreifer entweder

- die Rechenleistung der angegriffenen Station schätzen können, oder
- ihre maximale Sendebandbreite kennen, was leicht ist, wenn sie durch das Kommunikationsnetz, etwa beim MIX-Netz, fest vorgegeben ist. Anderenfalls muß er auch dies schätzen. Um seine Schätzung zu erleichtern, kann er selbst natürlich auch gegen ein faires Zugriffsprotokoll verstoßen (vgl. z. B. Abschnitt 3.1.2.3.2 Kollisionsauflösungsalgorithmus mit Mittelwertbildung und überlagerndem Empfangen) und dadurch der bzw. den angegriffenen Stationen die maximale Sendebandbreite verknappen.
- Kennt der Angreifer die maximale Empfangsbandbreite, was leicht ist, wenn sie durch das Kommunikationsnetz, etwa beim MIX-Netz ohne Verteilung „letzter“ Nachrichten,

fest vorgegeben ist, oder kann er sie schätzen, so müssen sich also soviele Angreiferstationen zusammentun, daß sie überschritten wird. Ggf. können natürlich auch wenige Angreiferstationen gegen ein faires Zugriffsprotokoll verstoßen.

Wird der Dienst rechtzeitig erbracht, so ist das Ergebnis des Angriffs, daß (ggf. unter der Annahme richtiger Schätzung) zwei Pseudonyme nicht zur selben Station gehören können. Wird der Dienst nicht rechtzeitig erbracht, so kann dies natürlich auch an sonstigen Dienstanforderungen an eine der beiden Stationen liegen. Kennt der Angreifer von jeder Station ein Pseudonym, z. B. eine ihr öffentlich zugeordnete Adresse, und erbringen Stationen unter diesen Adressen auch Dienste, so wählt der Angreifer als eines der beiden Pseudonyme jeweils ein nichtanonymes. Damit kann er jedes Pseudonym mit in der Stationenanzahl linearem Aufwand einer Station zuordnen.

Dieser *aktive* Verkettungsangriff über Betriebsmittelknappheit kann einerseits von der angegriffenen Station erkannt werden: Solange eine Station nicht aufgrund externer Dienstanforderungen an ihre Kapazitätsgrenzen stößt, wurde sie nicht angegriffen. Dies ist eine für die Überprüfbarkeit von Datenschutz weit bessere Situation als die der in Kapitel 1 und Abschnitt 2.1 erwähnten *passiven* Angriffe.

Andererseits kann der aktive Angriff leicht erschwert werden:

- Jeder Teilnehmer ist sowohl frei, wieviel Rechenleistung er „besitzt“ als auch, wieviel davon er für externe Dienstanforderungen bereitstellt. Letzteres kann er frei wählen und dies sogar von seiner Station (pseudo)zufällig über die Zeit verteilt tun lassen.
- Beim DC-, RING- und BAUM-Netz werden alle Stationen so ausgelegt, daß sie die gesamte Bandbreite zum Senden nutzen können. Dies verhindert allerdings einige in den folgenden Kapiteln beschriebene Optimierungen zur Einsparung von Betriebsmitteln.
- Bei Verteilung werden alle Stationen so ausgelegt, daß sie die gesamte Bandbreite zum Empfangen nutzen können. Dies verhindert allerdings einige in den folgenden Kapiteln beschriebene Optimierungen zur Einsparung von Betriebsmitteln.

Mit großem Aufwand und einschneidenden Einschränkungen der nutzbaren Leistung kann der aktive Angriff auch durch *statische Aufteilung der Betriebsmittel* vollständig verhindert werden: Vor der eigentlichen Dienstanforderung werden alle Pseudonyme anonym und unverkettbar bekanntgegeben, unter denen in der anstehenden Dienstphase ein Dienst angefordert werden wird. Sei n die Anzahl dieser Pseudonyme. Jede Station stellt dann jeder Dienstanforderung genau den n-ten Teil der minimalen im Kommunikationsnetz verfügbaren Rechenleistung sowie der minimalen im Kommunikationsnetz verfügbaren Sende- und Empfangsbandbreite zur Verfügung.

Für den praktischen Einsatz ist ein geeigneter Kompromiß zwischen der statischen und dynamischen Betriebsmittelaufteilung zu finden.

(Es sei angemerkt, daß Angriffe über Betriebsmittelknappheit und ihre Verhinderung durch statische Aufteilung der Betriebsmittel in einem anderen Kontext altbekannt sind: Benutzen zwei Prozesse gemeinsame Betriebsmittel, z. B. zwei Rechenprozesse denselben Prozessor, so kann bei fairer dynamischer Betriebsmittelaufteilung jeder der beiden Prozesse den anderen dadurch verzögern, daß er Betriebsmittel länger benutzt. Hat einer Zugriff auf die reale Zeit, so kann er diese Verzögerung feststellen. Diese modulierbare Verzögerung stellt einen verborgenen Kanal vom anderen Prozeß zu ihm dar, vgl. Abschnitte 1.2 und 2.1.2.)

Es sollte erwähnt werden, daß die speziellen Verfahren zum Schutz des Empfängers, des MIX-, DC-, RING- und BAUM-Netzes natürlich auch in *höheren* (Teil)Schichten implementiert werden können, beispielsweise auf einer beliebig gestalteten Schicht 3 (Vermittlungsschicht). Verteilung und MIXe würden dann in der Schicht 4 (Transportschicht) implementiert, die dann einige Funktionen der Schicht 3, z. B. Wegeermittlung, nochmals enthalten müßte. Wenn überlagerndes Senden in der Schicht 4 (Transportschicht) implementiert würde, müßte sogar anonymer Mehrfachzugriff nochmals in den höheren (Teil)Schichten implementiert werden. Dasselbe gilt für das RING- und BAUM-Netz, bei denen zusätzlich noch Verschlüsselung zwischen den Einheiten der Schicht 4 (virtuelle Verbindungs-Verschlüsselung) nötig wäre, um die physische Unbeobachtbarkeit der Leitungen zu simulieren.

Verteilung, MIX-, DC- und sogar RING- und BAUM-Netz können daher als *virtuelle*, d. h. in fast beliebigen Schichten implementierbare Konzepte betrachtet werden. Aber eine effiziente Implementierung muß unnötig komplexe Schichtung und nochmalige Implementierung von Funktionen vermeiden. Deshalb gibt Bild 30 den kanonischen und vernünftigen Entwurf an.

Aus Bild 30 ist ebenfalls zu ersehen, wie die im Abschnitt 2.5.3.2 und seinen Unterabschnitten beschriebene Erweiterung des Konzeptes der „Unbeobachtbarkeit angrenzender Leitungen und Stationen sowie digitale Signalregenerierung“ um „überlagerndes Senden“ kanonischer- und vernünftigerweise implementiert wird: Auf einem Medium passender Topologie (Schicht 0) und einer (Teil)Schicht 1 mit digitaler Signalregenerierung setzt eine weitere, höhere (Teil)Schicht 1 mit Überlagerung von Schlüsseln und Nachrichten auf. Darüber arbeitet dann das anonyme Mehrfachzugriffsverfahren.

Wie in [DiHe_79 Seite 420] erklärt, darf die Implementierung der (Teil)Schichten, die (innerhalb des Kommunikationsnetzes) Anonymität und Unverkettbarkeit schaffen oder erhalten müssen, keine Eigenschaften einführen, die ein Angreifer verwenden kann, Alternativen zu unterscheiden, die im abstrakten Entwurf für ihn ununterscheidbar sind. Beispiele solch unbrauchbarer Implementierungen wären

- Modulation der Ausgabe eines MIXes gemäß den in den Nachrichten enthaltenen zufälligen Bitketten,
- direkte Ausgabe des Ergebnisses eines modulo-2-Addierers, für dessen Ausgabe $1+1 \neq 0+0$ und $1+0 \neq 0+1$ gilt, wenn die analogen Signalcharakteristika betrachtet werden, oder
- zeichenfolgensensitive Taktverschiebungen der digitalen Signalregenerierung (pattern sensitive timing jitter) beim RING-Netz.

Diese und andere Fehler, vor allem aber geeignete Implementierungen werden in den folgenden Abschnitten 3.1 und 3.2 noch in größerer Tiefe betrachtet.