Proc. Verläßliche IT-Systeme (VIS'97), DuD-Fachbeiträge, Vieweg, S.172-191

Systematisierung und Modellierung von Mixen[1]

Elke Franz, Anja Jerichow, Andreas Pfitzmann

Technische Universität Dresden, D-01062 Dresden, {jerichow, ef1, pfitza}@inf.tu-dresden.de

Zusammenfassung: Ausgehend von der ursprünglichen Definition der Mixe von D. Chaum und fortführenden Arbeiten von A. Pfitzmann et al. werden Erweiterungen dieses Konzepts betrachtet. Daraus wird ein mögliches allgemeines Modell abgeleitet, welches verschiedene Anwendungen des Mix-Konzepts berücksichtigt.

In einem ablauforientierten Modell werden die möglichen Konfigurationen eines Mixes vorgestellt. Anhand der allgemeinen Anforderungen sowie der durch die Betrachtung von Angriffen hinzukommenden Forderungen wird dieses Modell modifiziert. Das abgeleitete Modell zeigt die unter den getroffenen Annahmen sicheren Konfigurationen.

1 Schutz der Kommunikationsbeziehung durch Mixe

1.1 Beschreibung des Mix-Konzepts

Das Konzept der Mixe wurde erstmals in [Chau_81] beschrieben. Mixe dienen dem Schutz der Kommunikationsbeziehung. Die Verkettbarkeit zwischen Sendern und Empfängern von Nachrichten wird verhindert (Abbildung 1-1).

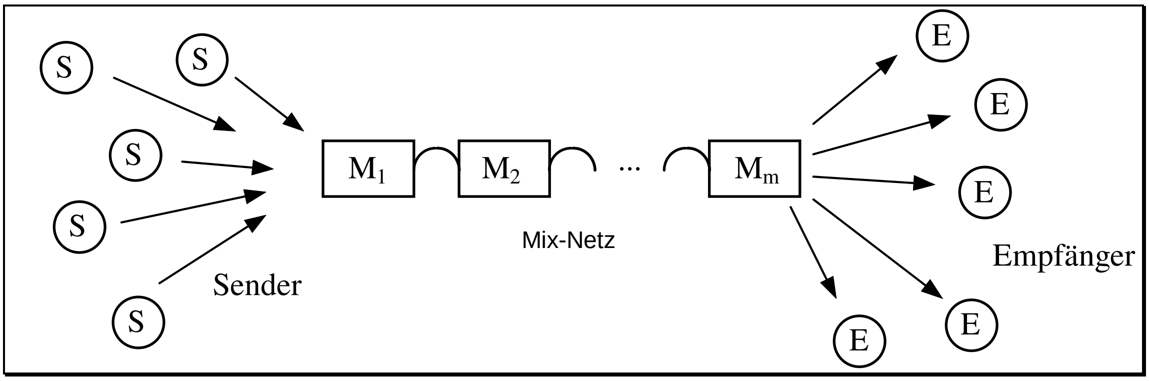


Abbildung 1-1: Schutz der Kommunikationsbeziehung durch Mixe

Ein Mix M_j sammelt Nachrichten, kodiert sie um, verändert ihre Reihenfolge durch Umsortieren (z.B. durch eine Sortierung nach alphabetischer Ordnung) und gibt die Nachrichten im Schub aus. Der nächste Mix, der diese Nachrichten erhält, verfährt genauso. Auf diese Weise kann der Weg der Nachrichten nicht durch das Mix-Netz verfolgt werden.

Durch Umkodierung wird erreicht, daß kein Außenstehender ein- und ausgehende Nachrichten anhand ihres äußeren Erscheinungsbildes einander zuordnen kann. Die Umkodierung innerhalb des Mixes erfolgt i.allg. mittels asymmetrischer Kryptographie. Das heißt: Ein Mix M_j hat einen öffentlich bekannten Schlüssel c_j und einen dazugehörigen privaten Schlüssel d_j. Die mit dem öffentlichen Schlüssel kodierten Nachrichten $c_j(N_i)$ gleicher Länge werden von M_j empfangen. Mittels seines privaten Schlüssels kann M_j dekodieren: $N_i := d_j(c_j(N_i))$. Die im Mix vorliegende Nachricht N_i wird zusammen mit den anderen in dieser Zeit angekommenen und bearbeiteten Nachrichten ausgegeben.

Bei deterministischer Umkodierung ist folgender Angriff möglich: Ein Mix kann überbrückt werden, indem ein Angreifer eine vom Mix ausgegebene Nachricht erneut mit dem öffentlichen Schlüssel des Mixes verschlüsselt. Das Ergebnis ist die zugehörige Eingangsnachricht. Durch Vergleich mit den zum Ausgabeschub gehörenden Ein-

[1] Teile der Arbeit wurden gefördert von der Deutschen Forschungsgemeinschaft (DFG).

gangsnachrichten erfolgt die Zuordnung der entsprechenden Ausgangsnachricht. Dieser Angriff kann durch **indeterministische Umkodierung** verhindert werden, indem der eigentlichen Nachricht ein zufälliger Teil z beigefügt wird: $c_j(N_i,z)$. Dieser kann vom Mix nach der Umkodierung einfach entfernt werden.

Trotz dieser Erweiterung erzeugt die Umkodierung aus gleichen Eingabenachrichten immer die gleichen Ausgabenachrichten. Um replay-Angriffe zu verhindern, darf jede Nachricht nur genau einmal vom Mix bearbeitet werden. Innerhalb eines Schubes wären ansonsten Verkettungen anhand von Nachrichtenhäufigkeiten möglich. Werden Nachrichten verteilt über mehrere Schübe wiederholt, so kann ein Mix durch das Bilden von Durchschnittsmengen bzw. Differenzen zwischen den jeweiligen Ein- und Ausgabenachrichten der entsprechenden Schübe überbrückt werden. **Wiederholungen von Nachrichten** sowohl innerhalb eines Schubes als auch verteilt über mehrere Schübe **sind zu ignorieren**.

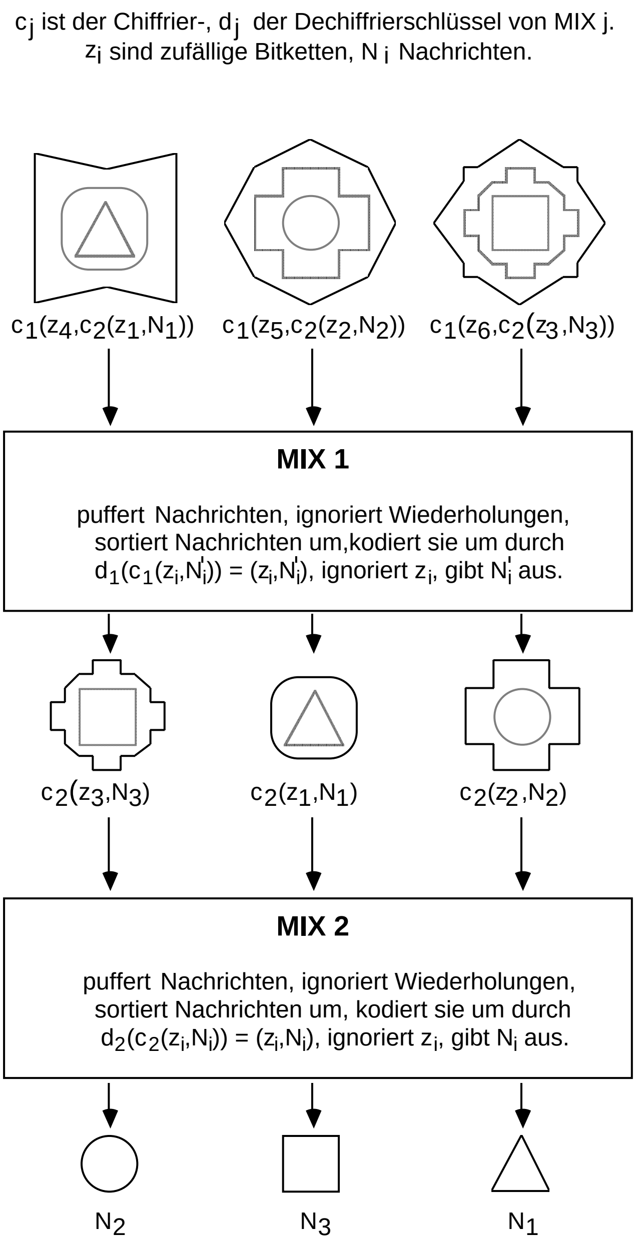


Abbildung 1-2: Mix-Prinzip für Senderanonymität nach [PfPW_88] (bei Kaskade entfallen Adressen)

Ein einzelner Mix weiß, von wem er Nachrichten erhält und an wen er sie weitergibt. Es sind also mindestens zwei Mixe nötig, damit einem einzelnen davon nicht Sender und Empfänger zugleich bekannt sind. Um Unverkettbarkeit zu erreichen, muß **mindestens einer** der verwendeten Mixe **vertrauenswürdig** sein, d.h. er darf keine Informationen über die Zuordnung der ein- und ausgehenden Nachrichten bekanntgeben.

Mixe sollten möglichst **unabhängig entworfen und hergestellt** werden sowie unabhängige Betreiber haben. Somit haben weder einzelne Personen noch Organisationen die Möglichkeit, den durch die Mixe erreichten Schutz der Kommunikationsbeziehung allein aufzuheben.

Diese Anforderungen wurden bereits in [Chau_81] gestellt. Mit weiteren Aspekten beschäftigen sich [Pfit_89] und [PfPW1_89].

Für die Wahl der Reihenfolge der durchlaufenen Mixe gibt es prinzipiell zwei Varianten: Entweder es wird von allen Nachrichten eine feste Reihenfolge durchlaufen, die Kaskade, oder die Reihenfolge der Mixe ist beliebig zu wählen.

In den Mixen kann durch verschiedene Umkodierungsschemata Sender- bzw. Empfängeranonymität sowie gegenseitige Anonymität erreicht werden (siehe 1.2 und 1.3). Bei diesen einfachen Schemata ändert sich jedoch die Länge der Nachrichten. Wird keine Kaskade verwendet, lassen sich daraus für einen Angreifer Informationen über den Weg der Nachricht gewinnen. Hier ist die Nutzung längentreuer Umkodierungsschemata nötig (siehe 1.5).

1.2 Senderanonymität

Der Sender verschlüsselt seine Nachricht N sowie Informationen über die Adresse A_{i+1} des jeweils nächsten Mixes M_{i+1} nacheinander mit den öffentlich bekannten Chiffrierschlüsseln c_j der Mixe, die auf dem Weg zum Empfänger (M_{m+1}) durchlaufen werden (Abbildung 1-2):

$$N_{m+1} = c_{m+1}(N)$$

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

Jeder Mix M_j wendet seinen geheimen Dechiffrierschlüssel d_j auf die Nachricht an und erhält so die nächste Darstellung der Nachricht und ggf. die Adresse des nächsten Mixes.

Hat der Sender mit dem ersten Mix einen geheimen Schlüssel ausgetauscht, so kann die entsprechende Verschlüsselung auch symmetrisch erfolgen [Pfit_89].

Der Empfänger kann bei diesem *direkten* Umkodierungsschema nicht nachvollziehen, wer ihm die Nachricht zugeschickt hat. Der Sender jedoch kennt das Aussehen der Nachricht in allen Punkten des Netzes einschließlich auf dem Weg vom letzten Mix zum Empfänger. Er kann infolgedessen den Empfänger beobachten.

1.3 Empfängeranonymität

Will der Empfänger B einer Nachricht anonym bleiben, so müssen zwei Verfahren kombiniert werden (Abbildung 1-3). Erstens muß B eine Adresse generieren, unter der er erreichbar ist, die sog. *anonyme Rückadresse* (RA).

$$R_{m+1} = e$$

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

Dabei ist e ein Kennzeichen für B. Die k_j sind symmetrische Schlüssel, die bei Anwendung der RA(:=R_1) von den Mixen für die Verschlüsselung der Nachricht genutzt werden. Der Index $m+1$ bezieht sich dabei auf den Empfänger B. A_j steht für die

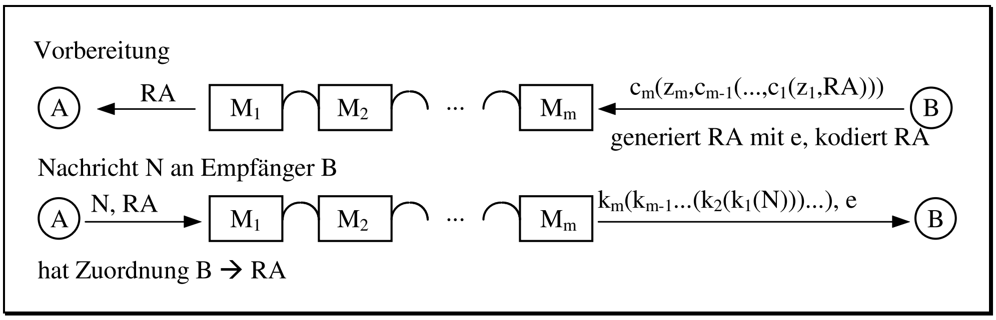


Abbildung 1-3: Empfängeranonymitätsschema

Adresse des Mixes j, c_j für seinen öffentlichen Schlüssel.

Zweitens sendet der Empfänger diese Rückadresse unter Wahrung seiner Anonymität an seinen Kommunikationspartner A gemäß dem Senderanonymitätsschema für Mixe. Die bei A hinterlegte RA wird beim Senden einer Nachricht an B an den ersten Mix geschickt, welcher die Nachricht verarbeitet und weiterleitet.

Der Empfänger kann mit diesem *indirekten* (oder *hybriden*) Umkodierungsschema seine Anonymität wahren, doch die Anonymität des Senders der Rückantwort wird nicht gewährt.

1.4 Gegenseitige Anonymität

Zur Wahrung gegenseitiger Anonymität können Sender- und Empfängeranonymitätsschema kombiniert werden. Auf dem Weg zwischen den Kommunikationspartnern gibt es einen Mix, welcher als Umlenkpunkt fungiert: Der Empfänger bildet eine anonyme Rückadresse, welche bei diesem Mix hinterlegt wird. Der Sender schickt seine Nachricht unter Benutzung des Senderanonymitätsschemas an diesen Mix.

Der Empfänger kennt den Nachrichtenweg vor dem Umlenk-Mix nicht, und der Sender verliert die Kontrolle über die Nachricht ab diesem Mix.

1.5 Längentreue Kodierung

Da jeder Mix aus der weiterzuleitenden Nachricht den Teil entfernt, in dem seine Adresse enthalten ist, zumindest aber die Zufallszahl, die er nicht ausgeben darf, erfolgt eine Verkürzung der Nachricht auf ihrem Weg durch die Mixe. Sollen keine Rückschlüsse aufgrund der Verkürzung der Nachrichten erfolgen, muß die Umkodierung *längentreu* erfolgen. Hierzu werden von jedem Mix zufällige Bitketten in die Nachricht eingefügt, so daß die Länge erhalten bleibt. Ein längentreues Umkodierungsschema ist immer indirekt.

Jede Nachricht hat die feste Länge von b Blöcken, in welchen eine anonyme Rückadresse, die zufälligen Bitfolgen und die eigentliche Nachricht enthalten sind. Die anonyme Rückadresse wird für dieses Verfahren folgendermaßen gebildet (die eckigen Klammern bedeuten Blockgrenzen):

$$R_{m+1} = [e]$$

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

Daraus ergibt sich, daß im ersten Block der Nachricht der vom empfangenden Mix M_j zu verwendende Schlüssel k_j sowie die Adresse A_{j+1} des nachfolgenden Mixes M_{j+1} enthalten ist. Dieser erste Block wird vom Mix M_j entfernt und der erhaltene Schlüssel k_j auf die restliche Nachricht angewandt. Vor den Blöcken, welche die eigentliche Nachricht enthalten, wird vom Mix M_j ein Block zufälligen Inhalts eingefügt und dadurch die Länge der Nachricht erhalten. Das Ergebnis dieser Operationen wird an den nachfolgenden Mix M_{j+1} geschickt, welcher wieder genauso verfährt.

Die Gesamtanzahl zu durchlaufender Mixe und die Nachrichtenlänge ist allen Mixen bekannt. Daraus können sie die entsprechende Stelle zum Einfügen ermitteln.

Der Teil der Nachricht, welcher die Rückadresse und die zufälligen Blöcke enthält, wird als Nachrichtenkopf (header) bezeichnet.

Ob die Anwendung des Schlüssels auf die Blöcke mit Nachrichteninhalt Ver- oder Entschlüsselung bedeutet, hängt davon ab, ob Sender- oder Empfängeranonymität erreicht werden soll.

Senderanonymität mit längentreuer Umkodierung:

Der Empfänger kennt die vom Sender an die Mixe übermittelten Schlüssel k_j nicht, folglich muß der Sender seine Nachricht mit allen k_j verschlüsseln. Die Anwendung des jeweiligen Schlüssels durch die einzelnen Mixe entspricht einer Entschlüsselung. Für die erste vom Sender anzuwendende Verschlüsselung muß dieser also den öffentlichen Schlüssel des Empfängers kennen, damit dieser durch Entschlüsselung mit seinem geheimen Schlüssel die Nachricht lesen kann.

Die Nachricht, die vom Sender gebildet werden muß, sieht folgendermaßen aus:

$$N_1 = H_1\, I \quad \text{mit} \quad H_1 = R_1$$

$$I_1 = k_1(k_2(...k_m(c_{m+1}(I))))$$

Dabei steht I für den Nachrichteninhalt. Statt der asymmetrischen Verschlüsselung im letzten Schritt könnte auch symmetrisch verschlüsselt werden, falls zwischen Sender und Empfänger Schlüssel ausgetauscht wurden. Das Kennzeichen *e* kann dann zur Zuordnung des richtigen Schlüssels verwendet werden.

Empfängeranonymität mit längentreuer Umkodierung:

Hierbei kennt der Sender die an die Mixe verteilten Schlüssel nicht. In seinem Rückadreßteil R ist der von ihm anzuwendende Schlüssel k_0 enthalten, mit welchem er seine Nachricht verschlüsselt und an den ersten Mix sendet.

Der Sender bildet also aus (k_0,A_1,R_1) die Nachricht

$$N_1 = H_1\, I_1 \qquad \textit{mit} \quad H_1 = R_1$$
$$I_1 = k_0(I)$$

Die Anwendung der Schlüssel durch die Mixe entspricht hierbei also einer Verschlüsselung. Der Empfänger kann durch das eindeutige Kennzeichen *e* die entsprechenden Schlüssel zuordnen und die Nachricht entschlüsseln.

2 Funktionen eines einzelnen Mixes

2.1 Sicherheit und Effizienz: Ableitung notwendiger Teilschritte

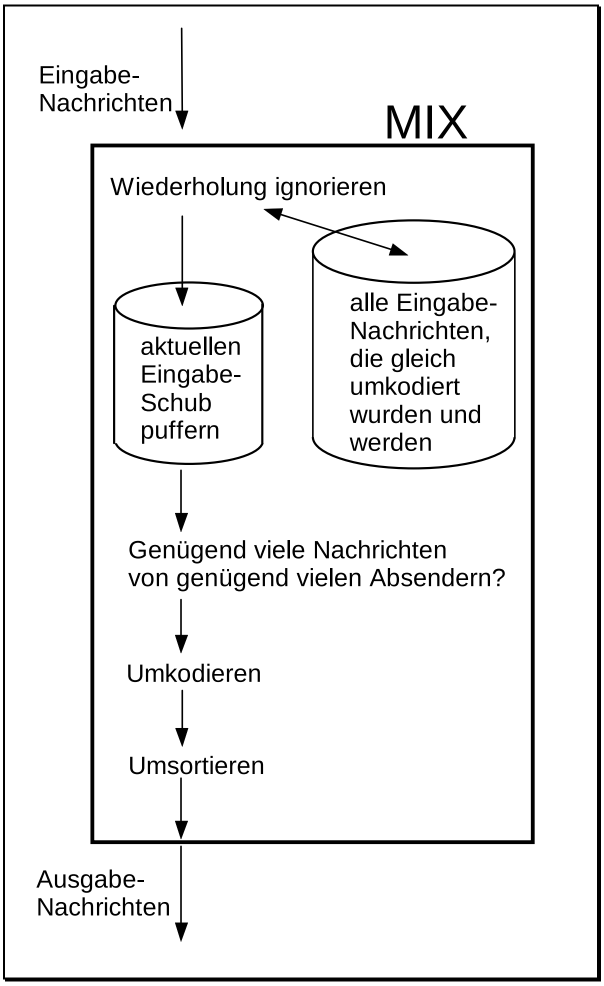


Abbildung 2-1: Grundfunktionen eines Mixes aus [Pfit_93]

Nach der Definition des Mixes in Kapitel 1.1 ergeben sich zunächst die folgenden Schritte:

- Sammeln von genügend vielen Nachrichten,
- Ignorieren von Wiederholungen,
- Umkodieren der Nachrichten,
- Umsortieren und Ausgeben der Nachrichten im Schub.

Abbildung 2-1 zeigt das Modell eines Mixes, welcher weitere notwendige Funktionen enthält. Die folgenden Funktionen könnte ein Mix zusätzlich zu den oben genannten Schritten noch ausführen:

- Erzeugen von dummies,
- Test auf genügend viele verschiedene Absender,
- Behandlung von Latenzzeiten.

Der Mix muß erst eine bestimmte Anzahl Nachrichten sammeln, ehe er sie weiterverarbeiten kann. Bei geringem Nachrichtenaufkommen können dadurch für die Nutzer große Wartezeiten entstehen. Die Festlegung von maximalen Wartezeiten erscheint daher sinnvoll. Ist die geforderte Anzahl von Nachrichten innerhalb der festgelegten Zeit noch nicht eingetroffen (Nachrichten könnten ja auch auf dem Weg durch das Kommunikationsnetz verloren gehen), so füllt der Mix seinen Nachrichtenpuffer mit bedeutungslosen Nachrichten, den **dummies**, auf. Die dummies sollen von den nachfolgenden Mixen ebenso wie alle anderen Nachrichten behandelt werden, sie sollen sich nicht von ihnen unterscheiden. Das Aussortieren der dummies könnte der letzte Mix vor dem Empfänger einer Nachricht übernehmen, wenn für ihn die dummies entsprechend gekennzeichnet sind.

Die Notwendigkeit einer weiteren Funktion ergibt sich aus dem **(*n*-1)-Angriff** auf einen Mix. Dabei kennt der Angreifer *n*-1 von *n* Nachrichten und kann somit die *n*-te, welche ihm unbekannt ist, leicht verfolgen. An die Nachrichten, die pro Schub umkodiert werden, wird darum die Forderung erhoben, daß sie von **genügend vielen verschiedenen Absendern** stammen müssen. Um die Forderung überprüfen zu können, müssen die **Absender verifizierbar** sein, was durch die Verwendung digitaler Signaturen erreichbar ist. Um digitale Signaturen leisten zu können, welche die Zuordnung zu einer Identität gewährleisten, sind ein asymmetrisches Kryptosystem und eine Zertifizierungsinstanz notwendig. Mit dem geheimen Schlüssel werden die Nachrichten verschlüsselt. Das Ergebnis dieser Umkodierung ist die Signatur, welche der Nachricht mitgegeben wird. Mit Hilfe des öffentlich bekannten Schlüssels kann jeder testen, ob die Nachricht wirklich von dem angegebenen Sender stammt. Um die Richtigkeit der Zuordnung von Schlüssel zu Schlüsselinhaber zu gewährleisten, ist die Zertifizierungsinstanz notwendig.

Die Überprüfung der Absender erfolgt vor dem ersten Mix (die anderen Mixe können an dieser Stelle mitprüfen). Ohne digitale Signaturen können die Absender jedoch nicht mit Sicherheit identifiziert werden - das Fälschen von e-mail-Absendern stellt kein ernsthaftes Problem dar. Nutzer können sich außerdem verschiedene Nutzernamen geben lassen.

Selbst wenn abgeprüft werden kann, ob genügend viele Nachrichten von genügend vielen verschiedenen Absendern gesammelt wurden, kann damit nicht ausgeschlossen werden, daß Angreifer zusammenarbeiten und damit trotz verschiedener Absender *n*-1 von *n* Nachrichten bekannt sind. Während der zuerst beschriebene Einsatz von dummies hauptsächlich der Garantie der Dienstqualität dient, kann hier ihr Einsatz aus Sicherheitsgründen erfolgen. Das Hinzufügen von dummies erhöht die Anzahl unbekannter Nachrichten pro Schub automatisch, so daß es den Sendern nicht mehr möglich ist, eine einzige unbekannte Nachricht zu verfolgen.

Dummies bieten jedoch keinen Schutz, falls die Empfänger die Angreifer sind. Im Extremfall könnte nur eine richtige Nachricht gesendet und der Rest vom Mix mit dummies aufgefüllt werden. Ist der Angreifer der Empfänger dieser einen Nachricht, so kann er den Sender bestimmen, da das Senden beobachtbar ist. Entsprechendes gilt für das Enthaltensein mehrerer richtiger Nachrichten: Schließen sich alle Empfänger bedeutungsvoller Nachrichten zusammen, können sie bestimmen, an welche Gruppe von Empfängern die in Frage kommenden Sender Nachrichten geschickt haben. Dummies bieten also vor allem Schutz gegen Angreifer, welche Sender von Nachrichten sind.

Um die Unsicherheit über den Weg einer Nachricht zu erhöhen, ist noch eine weitere Variante vorstellbar: von den Nutzern gewählte minimale (und/oder maximale) Wartezeiten der Nachrichten im Mix. Diese Verzögerungszeiten (**Latenzzeiten**) können von den Nutzern für jeden Mix, welchen die Nachricht passiert, gewählt werden, unabhängig von der Betriebsart des Puffers (Batch, Pool oder Mischform, siehe unten).

2.2 Beschreibung der Teilschritte in einem Mix

In Abbildung 2-2 wird ein allgemeines Modell für einen Mix vorgestellt, welches die soeben diskutierten Funktionen enthält.

Wie bereits erwähnt, kann die Erzeugung von dummies aus Sicherheits- bzw. aus Effizienzgründen notwendig sein. Da das Umkodieren bei Verwendung asymmetrischer Kryptographie eine aufwendige Funktion ist, wird die dummy-Erzeugung erst im letzten Schritt, dem Umsortieren, durchgeführt.

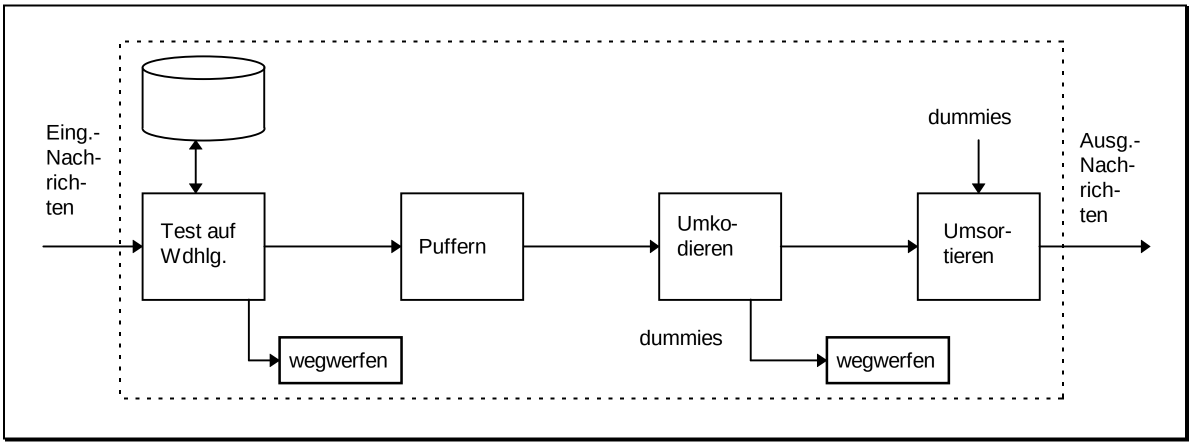


Abbildung 2-2: Allgemeines Modell eines Mixes

Nicht alle im Modell dargestellten Funktionen müssen in einem Mix auf einmal enthalten sein. Es gibt verschiedene Arten, sie zu kombinieren, so daß Unverkettbarkeit gewährleistet wird. Im folgenden sind die Aufgaben der Teilschritte zusammengefaßt.

1) Wiederholungen ignorieren

- um replay-Angriffe zu verhindern; mögliche Varianten:
  - Speicherung bereits gesendeter Nachrichten in einer Datenbank;
  - Vergabe von Zeitstempeln für die Nachrichten
  - Verhinderung von Wiederholungsangriffen durch Verwendung einer synchronen Stromchiffre (hybride Verschlüsselung)
- Wiederholungen werden vom Mix weggeworfen

2) Puffern

- Sammeln der Nachrichten, bevor diese weiterverarbeitet werden; zwei Varianten der Organisation des Puffers:
  - *Batchbetrieb*: m Speicherplätze für die Nachrichten vorgesehen; wenn diese Anzahl erreicht ist, wird der Puffer geleert,
  - *Poolbetrieb*: m Speicherplätze vorgesehen, trifft $(m+1)$-te Nachricht ein, wird aus Pool zufällig eine Nachricht ausgewählt und weiterverarbeitet
  - Mischformen: Pool von m Nachrichten, aus denen jedoch bei Eintreffen der $(m+1)$-ten Nachricht nicht nur eine, sondern eine Anzahl n ($n<m$) Nachrichten zufällig ausgewählt und weitergeschickt wird
- Prüfen der Absender (Forderung: es müssen genügend viele Nachrichten von genügend vielen verschiedenen Absendern vorhanden sein, ehe weitergearbeitet wird; zur Verhinderung von $(n$-$1)$-Angriffen *weniger* Angreifer, Absender müssen verifizierbar sein)
- Behandlung von Latenzzeiten: Möglichkeit der Vergabe von Zeitschranken für die Weiterverarbeitung bzw. Verzögerung der Nachrichten:
  - Vergabe der Latenzzeiten durch die Sender der Nachrichten
  - Verzögerung um zufällige Dauer durch Mix ebenfalls vorstellbar

3) Umkodieren

- Speicherung von Zustandsinformationen (z.B. Datensätze der Teilnehmer bei verbindungsorientierter Übertragung, zur Verwaltung von Paßwörtern/Accounts (Remailer), Informationen zur Schlüsselerzeugung...)
- Verändern des Aussehens der Nachricht, außerdem Erhalten der Folgeadresse, an die gesendet wird (falls Mix-Netz, bei fester Kaskade keine Adresse nötig)
- je nach Verfahren und Position des Mixes symmetrische (synchrone Stromchiffre) oder asymmetrische Verschlüsselung

- der verwendete Schlüssel kann auch im Mix selbst erzeugt werden, z.B. mit Hilfe eines Pseudozufallszahlengenerators aus einem vorher hinterlegten Startwert
- Auffüllen der Nachrichten mit Zufallszahlen (für längentreue Schemata (siehe 1.5), bei NDM (siehe 2.3.5))
- vom Mix erkannte dummies werden weggeworfen

4) Umsortieren

- falls ein Schub von Nachrichten ausgegeben wird: Umsortieren der Nachrichten, damit keine Zuordnung über die Reihenfolge möglich ist
- Umsortieren sollte keine Möglichkeiten für Angriffe bieten, darum beispielsweise Sortierung nach alphabetischer Ordnung der Ausgabenachrichten
- Erzeugung von dummies
- enthält auch die Ausgabe der Nachrichten

Die Zuordnung von Aufgaben zu den einzelnen Teilschritten könnte an einigen Stellen auch anders erfolgen. So könnte die Speicherung von Zustandsinformationen auch Teil des Pufferns sein, da dort ohnehin Speicherung vorgesehen ist. Die Zuordnung zur Umkodierung betont, daß oftmals die Informationen entweder erst nach der Umkodierung vom Mix gelesen und gespeichert werden können oder daß gespeicherte Informationen zur Umkodierung benötigt werden, so z.B. zur Schlüsselgewinnung.

Das Senden bleibt beobachtbar. Daher kann von den Teilnehmern gefordert werden, daß sie zu jeder Zeit (immer zu festgelegten Takten) Nachrichten senden müssen, egal, ob es sich dabei um richtige oder um bedeutungslose Nachrichten handelt. Die zu einem Zeitpunkt sendenden Teilnehmer bilden jeweils eine Anonymitätsgruppe. Senden die Teilnehmer bedeutungslose Nachrichten, so kann die Erzeugung von dummies im Mix evtl. weggelassen werden.

Aus diesem allgemeinen Modell lassen sich viele Möglichkeiten konstruieren. Im folgenden werden zunächst existierende Varianten der Mixe vorgestellt. Anschließend werden die möglichen Kombinationen, welche man aus dem allgemeinen Modell ableiten kann, im Hinblick auf Sicherheitsanforderungen betrachtet.

2.3 Anwendungen des Mix-Konzepts - Beispiele für andere Modelle

2.3.1 Telefon-Mixe (PfPW1_89)

Diese Variante der Mixe gewährleistet den Schutz von der Kommunikationsbeziehung auf der Basis von ISDN. Die asymmetrische Verschlüsselung innerhalb der Mixe ist zeitaufwendig. DieVerwendungvon Mix-Kanälen verringert die Verzögerung durch die Mixe. Dabei wird die hybride Verschlüsselung zur Erhöhung der Geschwindigkeit der Übertragung genutzt, insbesondere der Verringerung der Verzögerungszeit.

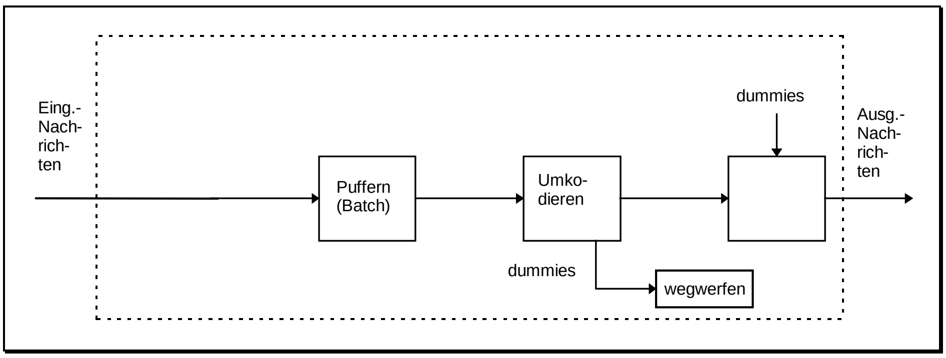


Abbildung 2-3: Funktionalität eines Telefon-Mixes (Senden der Nutzdaten)

Die Übertragung von Nachrichten wird aufgeteilt in die beiden Schritte:

1. Übertragung einer Kanalaufbaunachricht (asymmetrische Verschlüsselung) und
2. Übertragung des eigentlichen Nachrichteninhalts (symmetrische Verschlüsselung).

Bei der Übertragung des ersten Teils merkt sich jeder Mix die Zuordnung vom Eingabe- zum Ausgabeschub. Damit wird die Umsortierung der später gesendeten Nachrichten festgelegt. Die für die symmetrische Verschlüsselung der Nachrichten benötigten Schlüssel werden beim Kanalaufbau in den Mixen hinterlegt.

Baut sich der Sender einen Kanal auf, den Sendekanal, so kann er gegenüber dem Empfänger seine Anonymität wahren. Entsprechendes gilt für den Empfänger, welcher sich einen Empfangskanal aufbauen kann. Um gegenseitige Anonymität zu wahren, werden beide Kanäle nicht über die Teilnehmeridentitäten, sondern nur über ein sogenanntes Kanalkennzeichen verknüpft. Dieses müssen die Kommunikationspartner vor dem eigentlichen Kanalaufbau ausmachen. Dazu wird nach dem üblichen Schema eine Kanalwunschnachricht gesendet.

In Abbildung 2-3 ist das Modell eines solchen Mixes während der Übertragung der eigentlichen Nachrichten dargestellt. Diese werden mit einer synchronen Stromchiffre verschlüsselt. Durch die fortlaufende Schlüsselung sind Wiederholungsangriffe ausgeschlossen und der Test auf Wiederholungen entfällt.

Da Nachrichten eines Schubes gleich lang sein müssen, müßten gleichzeitig aufgebaute Kanäle auch gleichzeitig wieder abgebaut werden. Um zu verhindern, daß alle Kanäle auf das Ende des längsten gleichzeitig aufgebauten Kanals warten müssen, werden Zeitscheiben eingeführt: Der Auf- und Abbau von Zeitscheibenkanälen wird fortlaufend zu festen Zeittakten durchgeführt. Somit dürfen die Kanäle (die durch eine Abfolge von Zeitscheibenkanälen gebildet werden) beliebig lang sein, und alle zu den Zeittakten auf- bzw. abbauenden Teilnehmer bilden jeweils eine Anonymitätsgruppe.

2.3.2 Verbindungsorientierte Übertragung

[FFJM_97] beschreibt ein Verfahren zum Schutz der Aufenthaltsinformation in Mobilkommunikationsnetzen. Mixe werden hier mit erweiterter Funktionalität genutzt.

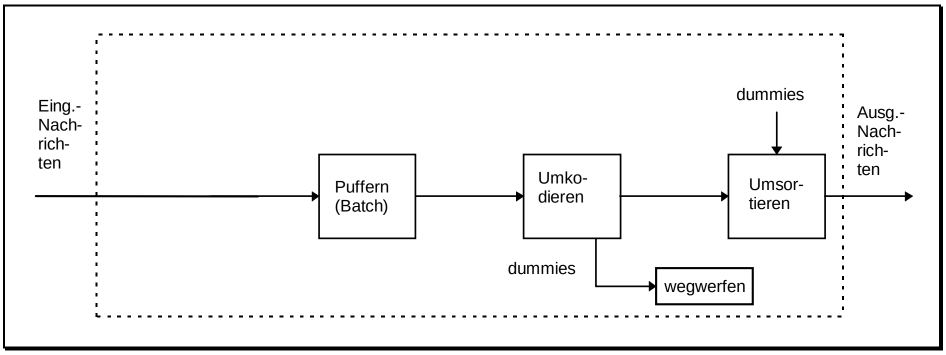


Abbildung 2-4: Mix für verbindungsorientierte Übertragung: Datensätze der Teilnehmer werden gespeichert

Um die Teilnehmer lokalisieren zu können, werden in existierenden Netzen Aufenthaltsinformationen in Registern gespeichert. Die Speicherung erfolgt hierarchisch aus Effizienzgründen. Ankommende Verbindungswünsche werden durch die Hierarchieebenen zu den Teilnehmern weitergeleitet. Da jedoch alle Hierarchieebenen einem Betreiber gehören, hat dieser die globale Sicht auf die Teilnehmerdaten.

Damit trotz der Speicherung der Aufenthaltsort der Teilnehmer geschützt bleibt, werden zum Weiterleiten der Informationen Mixe vorgeschlagen, welche die Speicherung der Aufenthaltsinformationen in den einzelnen Hierarchieebenen übernehmen (Abbildung 2-4). Von den Mixen wird gefordert, daß sie nicht heimlich miteinander kooperieren. Da in zukünftigen Systemen die Netze verschiedener Betreiber integriert werden sollen, ist die Realisierung dieser Forderung vorstellbar. Damit Verbindungswünsche anonym weitergeleitet werden können, erfolgt die Speicherung der Teilnehmerdaten pseudonym.

Der Test auf Wiederholungen kann durch die Verwendung einer synchronen Stromchiffre entfallen.

2.3.3 Remailer

Eine Übersicht über bestehende Remailer- und Mixmaster-Implementierungen ist in [Scha_95] enthalten. Besitzt ein Remailer bzw. Mixmaster auch nicht die Funktionalität eines Mixes, so ist doch auch er zum Schutz der Kommunikationsbeziehung vorgesehen. Dabei existieren mehrere Varianten, welche verschiedene Sicherheit bieten.

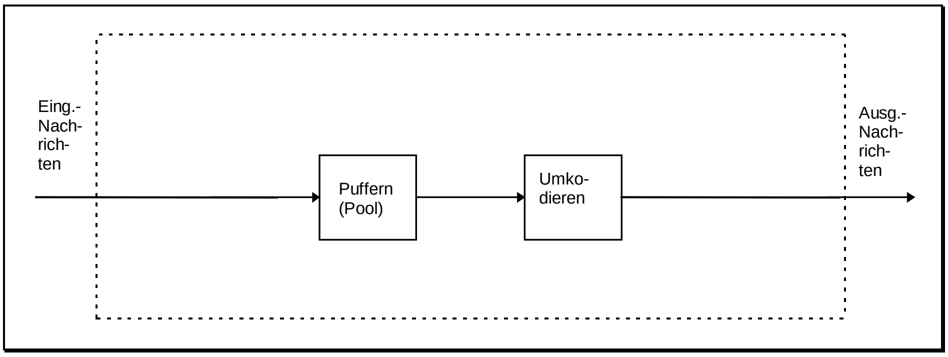


Abbildung 2-5: Remailer mit Behandlung von Latenzzeiten sowie Mindestpool

Eine einfache Variante wurde in Finnland geschaffen (heute nicht mehr aktiv). Dieser Remailer führt einfach eine Ersetzung des Absenders mit seinem Absender durch. Der Empfänger ist dabei aus der e-mail des Senders erkennbar. Vor dem Remailer sind also beide Kommunikationspartner in der Nachricht erkennbar. Der Remailer kann leicht überbrückt werden, wenn das Netz auf Sender- und auf Empfängerseite überprüft werden kann, da keine Umkodierung stattfindet. Dieses System hält also nur sehr eingeschränkten Angriffen stand.

Um Rückantworten zu ermöglichen, wird jedem Nutzer ein Account erteilt. Mit Hilfe eines Paßwortes kann verhindert werden, daß Absender gefälscht werden. Allerdings wird das Paßwort im Klartext an den Remailer übertragen.

Das Modell dieses Remailers würde nur aus der Funktion „Puffern" bestehen, welche hier nur die Speicherung von Zustandsinformationen auszuführen hat.

Cypherpunk-Remailer bieten schon mehr Funktionalität. Auch die unverschlüsselte Weiterleitung vom e-mails ist möglich, doch es gibt auch die Möglichkeit der Umkodierung. Um die zeitliche Zuordnung von e-mails zu verhindern, können die Nutzer Latenzzeiten angeben. Treffen während der Wartezeit der Nachricht im Remailer weitere Nachrichten ein, so ist die Zuordnung der ein- und ausgehenden Nachrichten erschwert. Trifft jedoch keine weitere Nachricht ein, so ist die Wahl der Latenzzeit nutzlos, da die Nachricht einfach verzögert weitergegeben wird.

In weiteren Varianten wird darum ein Mindestpool an Nachrichten geführt (Abbildung 2-5). Nach wie vor besteht jedoch die Möglichkeit, replay-Angriffe durchzuführen oder die Nachrichten anhand ihrer Länge zuzuordnen.

2.3.4 Mixmaster

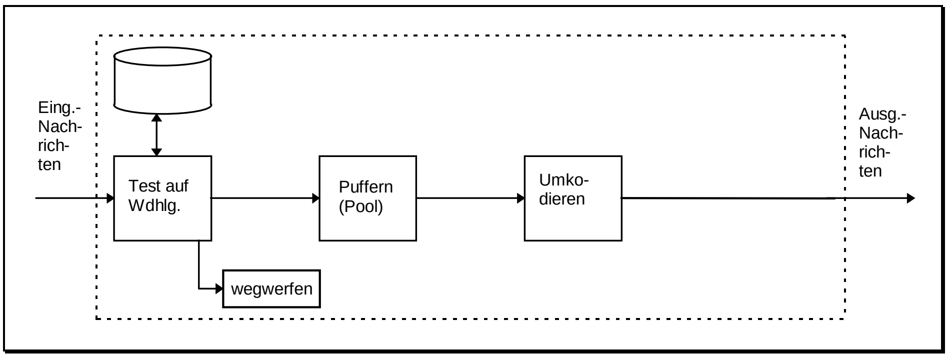


Abbildung 2-6: Funktionalität eines Mixmasters (Einzelbetrieb)

Die Mixmaster orientieren sich eng an der in [Chau_81] vorgestellten Variante. Die Nachrichten müssen ein festes Format haben, die Mixmaster führen Umkodierung durch und testen Wiederholungen anhand der Paket-ID. Arbeitet der Mixmaster im Einzelbetrieb, werden die Nachrichten in einem Pool gesammelt (Abbildung 2-6). Beim Stapelbetrieb werden sie zu festen Zeittakten ausgelesen, weiterverarbeitet und ggf. mit dummies aufgefüllt.

Ohne zusätzliche Sicherheitsmaßnahmen (wie Absenderprüfung) bietet der Pool mehr Sicherheit als ein Batch gleicher Größe.

Aus der Arbeitsweise des Pools ergibt sich, daß die Verweilzeit einer Nachricht im Mix ungewiß ist. Der Angreifer kann also im Gegensatz zum Batch nicht wissen, wann eine bestimmte Nachricht den Pool wieder verläßt. Um nun diese eine Nachricht dadurch zu verfolgen, daß sie die einzige unbekannte ist, müßte der Angreifer über längere Zeit den Pool nur mit seinen Nachrichten füllen und darüber hinaus in der Lage sein, nach dem Senden der interessierenden Nachricht jedes weitere Senden anderer Absender zu unterbinden, bis die Nachricht den Mix verläßt.

Für einen Angreifer, der dem Mix legal viele Nachrichten zuschicken, aber nicht unerlaubterweise die Kommunikation der anderen Teilnehmer stören kann, ist ein solcher Angriff sicher schwer durchzuführen. Kann der Angreifer jedoch das Senden anderer Teilnehmer verhindern, erscheint die gebotene Sicherheit nicht mehr ausreichend.

Ein anderes Konzept für Remailer wird in [GüTs_96] diskutiert. Die Nachrichten werden zufällig im Mix verzögert. Die Ausgabe erfolgt zu festen Zeittakten. Um einen (n-1)-Angriff durch Verzögerung von Nachrichten zu verhindern, werden hier Umwege zwischen den Mixen gewählt.

2.3.5 Non-Disclosure Method (NDM)

Bei der in [FaKK_96] vorgestellten Methode existieren eine Anzahl von unabhängigen SAs (security agents, Abbildung 2-7). Diese übernehmen die Rolle der Mixe. Die SAs benutzen für die Umkodierung ein asymmetrisches Kryptosystem (RSA). Aus den SAs wählt der Nutzer sich einen beliebigen aus. Ein SA sammelt keine Nachrichten im Batch oder Pool. Da auch Wiederholungen nicht abgewiesen werden, sind replay-Angriffe möglich.

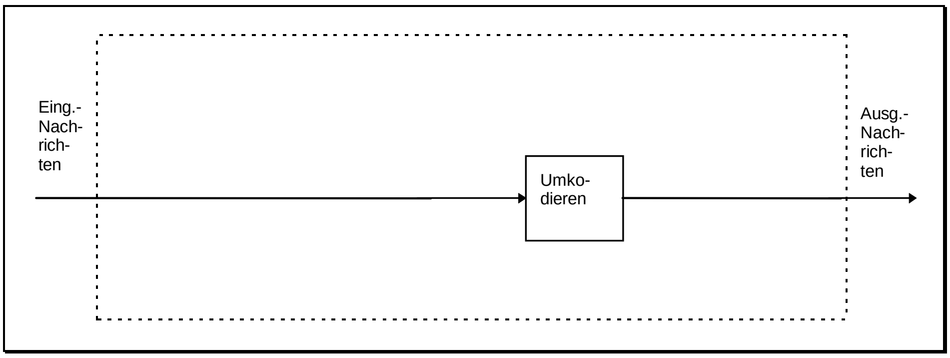


Abbildung 2-7: Security agent der NDM

Das Angreifermodell, welches ansonsten für Mixe angenommen wird, wurde für die Methode modifiziert. Ein Angreifer, welcher alle Kommunikation beobachten kann, wäre in der Lage, den Weg der Nachrichten zu verfolgen. Es wird jedoch davon ausgegangen, daß es schwierig ist, die Kommunikation zu beobachten. Dem Angreifer soll nicht bekannt sein, welche SAs gewählt werden. Er wird also nur mit einer gewissen Wahrscheinlichkeit die richtigen Stellen abhören, falls er nicht die gesamte Kommunikation im Netz beobachten kann. Die Möglichkeit, sämtliche Kommunikation zu beobachten, wird aber als zu aufwendig eingestuft. Gelingt es ihm jedoch, eine Nachricht an nur zwei Stellen abzuhören, können dazwischenliegende SAs durch replay-Angriffe überbrückt werden.

Um replay-Angriffe zu verhindern, werden noch zufällige Bitfolgen eingefügt. Der Sender fügt nur die erste Folge ein, die der erste SA entfernt. Nach dem Umkodieren erzeugt dieser eine neue Bitfolge und verschlüsselt sie zusammen mit der Nachricht mit dem öffentlich bekannten Schlüssel seines Nachfolgers. Dieser verfährt genauso.

Mit dieser Erweiterung wird eine Verbindungsverschlüsselung zwischen den einzelnen SAs erreicht. Eine Verbindungsverschlüsselung liefert jedoch immer nur Schutz gegen Angreifer auf der Verbindungsstrecke.

3 Kombinationen der Mix-Funktionen

3.1 Allgemeines ablauforientiertes Modell

Wie die Anwendungsbeispiele gezeigt haben, sind verschiedene Arbeitsweisen der einzelnen Teilschritte innerhalb eines Mixes möglich. Die Teilschritte selbst können auf verschiedene Arten miteinander kombiniert werden. Es ergibt sich theoretisch eine Vielzahl von Möglichkeiten. Um diese zu ordnen, soll ein Modell angegeben werden, welches die möglichen Kombinationen darstellt. Dazu werden zunächst die Varianten der Teilschritte in Tabelle 3-1 zusammengefaßt.

Teilschritt	**Varianten**	**Abk.**
Test auf Wiederholungen	Speicherung (Nachr., Hashwerte,...) in Datenbank Verwendung von Zeitstempeln kein Test erforderlich, falls synchrone Stromchiffre	DB Zst sSc
Puffern	Prüfung: genügend viele verschiedene Absender	Abs
	Behandlung von Latenzzeiten	Lz
	Batch Pool Mischformen	B P M
Umkodieren	Speicherung von Zustandsinformationen	Sp
	asymmetrisch	Ua
	symmetrisch (synchrone Stromchiffre)	Us
Umsortieren	Umsortieren	S
	Erzeugung von dummies	d

Tabelle 3-1: Übersicht über verschiedene Varianten zu den einzelnen Funktionen

In einem allgemeinen Modell, welches auch die vorgestellten Anwendungsbeispiele abdeckt, sind die einzelnen Teilschritte optional. Abbildung 3-1 zeigt eine solche *ablauforientierte* Darstellung der Funktionen eines Mixes.

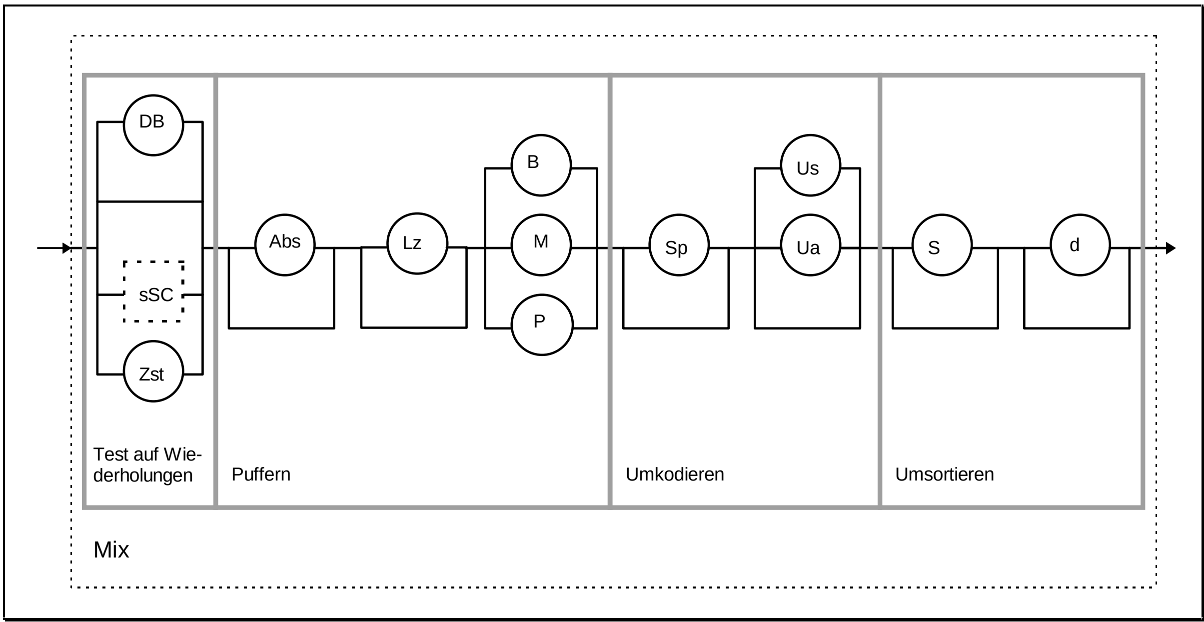


Abbildung 3-1: Ablauforientierte Darstellung der Teilschritte innerhalb eines Mixes (Abk. gemäß Tab. 3.1)

Nicht alle Systemparameter sind in einem solchen Modell darstellbar. So werden die Eingabenachrichten nicht näher charakterisiert. Die Teilnehmer können entweder nur bei Bedarf oder immer zu festen Zeittakten senden (dann mit dummy traffic).

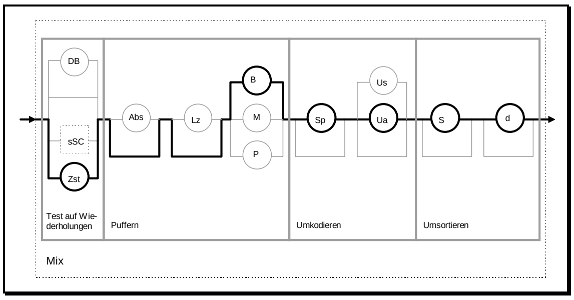
Die Batch- bzw. Poolgröße könnte fest oder variabel sein. I.allg. wird von einer festen Größe ausgegangen.

Weiterhin kann der Mix entweder ereignis- oder zeitorientiert betrieben werden. Im ersten Fall wird der Puffer nur geleert, wenn genügend Nachrichten gesammelt wurden. Im zweiten Fall erfolgt das Auslesen zu festen Zeittakten. Die Verwendung von dummies ist dann unbedingt erforderlich. Doch auch beim ersten Fall sollte durch die Möglichkeit, dummies zu erzeugen, die Dienstqualität des Mixes gesichert werden.

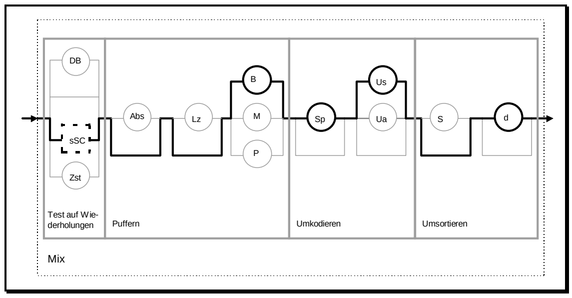
Auch über die Ausgabe der Nachrichten werden keine Aussagen getroffen. Die einzelnen Nachrichten können entweder direkt zugestellt oder innerhalb einer Gruppe verteilt werden.

3.2 Anwendungsbeispiele unter dem vorgestellten allgemeinen Modell

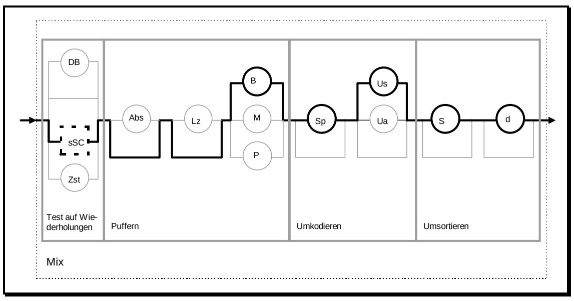
Bei der Nachrichtenübermittlung durch die Telefon-Mixe (3-2 b)) werden die Nachrichten auch umsortiert. Die Umsortierung ist jedoch fest (entsprechend der Zuordnung von Eingabe- zu Ausgabekanälen bei der Signalisierung). Da die Umsortierung bei der Nachrichtenübermittlung keinen Aufwand mehr erfordert, wird sie in der Abbildung nicht als Funktion angegeben.



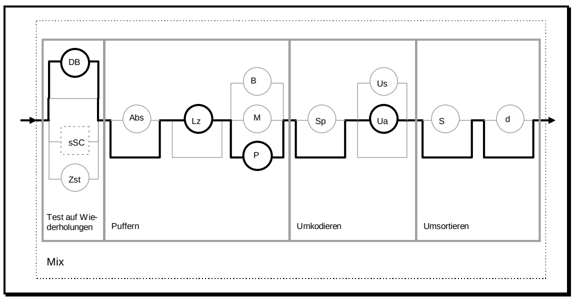
a) Telefon-Mix (Signalisierung)



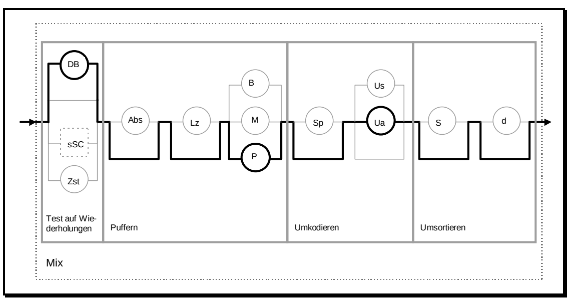
b) Telefon-Mix (Nachrichtenübermittlung)



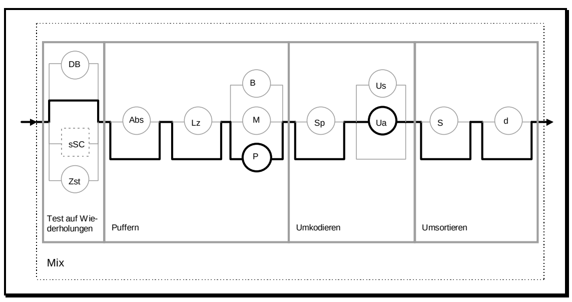
c) Verbindungsorientierte Übertragung (nach Aufbau)



d) Cypherpunk-Remailer



e) Mixmaster



f) NDM-security agent

Abbildung 3-2: Kombinationsmöglichkeiten der Mixfunktionen (Darstellung der Anwendungsbeispiele von 2.3)

Beim NDM-security agent (3-2 f)) werden die Nachrichten nicht gesammelt. Die Zuordnung zum Pool erfolgte willkürlich, ebenso hätte die Betriebsart Batch gewählt werden können. Bei beiden Betriebsarten ist die Größe des Puffers gleich Eins.

In allen Darstellungen wurde davon ausgegangen, daß die Absender nicht verifizierbar sind. Darum wurde keine Absenderprüfung einbezogen. Nur bei den Telefon-Mixen wird die Forderung nach genügend vielen verschiedenen Absendern erfüllt, da jeder Teilnehmer ständig einen Sende- und einen Empfangskanal unterhält.

Das vorgestellte Modell enthält durch die Betrachtung sicherheitseingeschränkter Verfahren auch Wege, die eine unsichere Betriebsweise des Mixes zulassen. Im folgenden werden notwendige Einschränkungen sowie mögliche Kombinationen der Funktionen des Puffers diskutiert. Ziel ist ein modifiziertes Modell, welches nur Wege enthält, die unter bestimmten Voraussetzungen Unverkettbarkeit gewährleisten.

3.3 Notwendige Modifikationen und Angreifermodelle

3.3.1 Einschränkungen der Wege aufgrund allgemeiner Anforderungen

Aus den notwendigen Teilschritten eines Mixes ergeben sich bereits einige Einschränkungen des allgemeinen Modells. Daraus ergeben sich die im folgenden erläuterten Umsortierungen des ablauforientierten Modells. Sich beeinflussende Funktionen werden darin zusammengefaßt.

Test auf Wiederholungen, Umkodieren: Eine Möglichkeit der Verhinderung von Nachrichtenwiederholungen besteht im Speichern der vom Mix bearbeiteten Nachrichten (bzw. Hashwerte dieser Nachrichten, Nummern, ...) in einer Datenbank. Nach einer gewissen Zeit sind die Datenbestände zu löschen. Dazu wird z.B. das Schlüsselpaar des Mixes gewechselt.

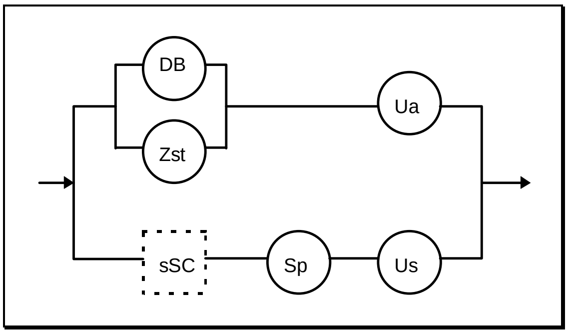


Abbildung 3-3: Test auf Wiederholungen

Auch das Überprüfen von Zeitstempeln verhindert Wiederholungen. Beide Varianten setzen keine symmetrische Umkodierung voraus, da für den Test nicht erforderlich (obwohl theoretisch möglich).

Anders sieht es mit dem unteren Zweig aus. Unter der Voraussetzung, daß mit einer synchronen Stromchiffre verschlüsselt wird, kann der Test auf Wiederholungen entfallen, da Wiederholungen nicht „möglich“ sind. Die Umkodierung erfolgt also symmetrisch. Dafür ist die Speicherung von Informationen über den zu verwendenden Schlüssel unerläßlich.

Organisation des Puffers, Umsortieren:

Arbeitet der Puffer im Pool-Betrieb, so entfällt das Umsortieren der Nachrichten. Wird dagegen ein Schub von Nachrichten weitergeschickt, so ist die Reihenfolge zu verändern.

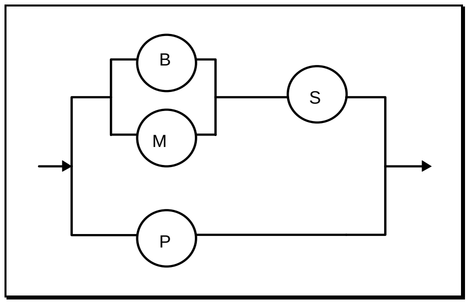


Abbildung 3-4: Organisation des Puffers

3.3.2 Systematisierung von Angriffen

Für Mixe ist es aufgrund der Komplexität des Systems generell schwierig, Sicherheitsbedingungen zu definieren. Andere Verfahren mit nur wenigen Parametern lassen klare Definitionen der Sicherheit zu. So kann beispielsweise die Vernam-Chiffre als informationstheoretisch sicher eindeutig beschrieben werden.

In den zugrunde gelegten Mix-Konzepten [Chau_81, Pfit_89] wird davon ausgegangen, daß Angreifer sämtliche Kommunikation im Netz abhören können. Es wird gefordert, daß mindestens ein Mix vertrauenswürdig ist.

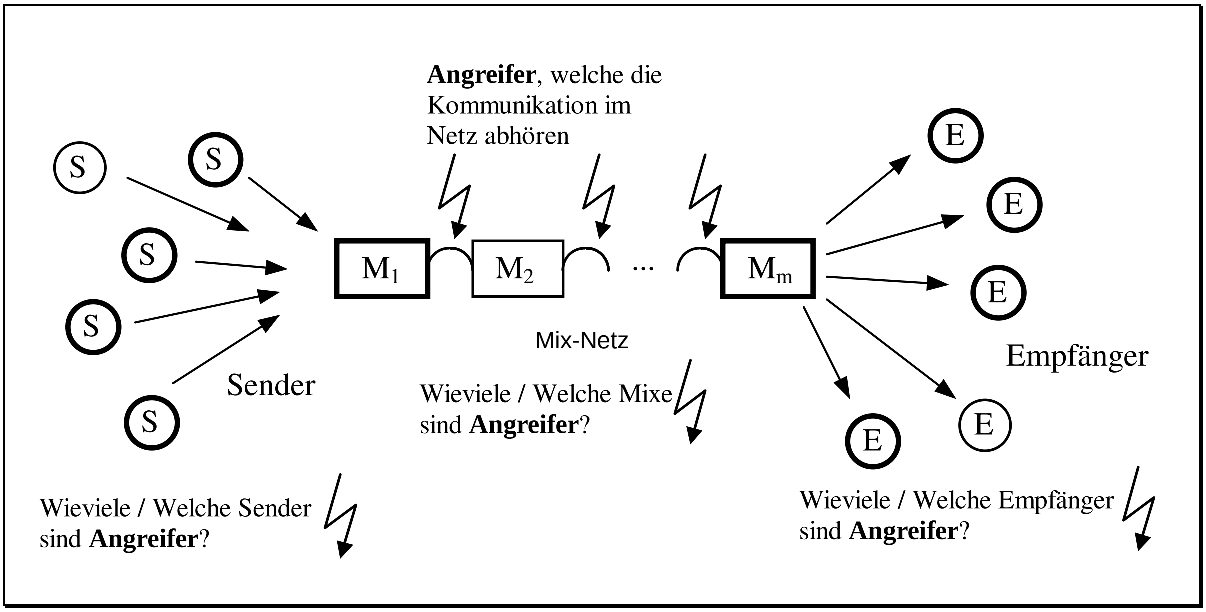


Abbildung 3-5: Mögliche Angreifer beim Mixkonzept

Bei der Betrachtung der Sicherheit ist eine Vielzahl von **Einflußgrößen** zu beachten:

- Es ist zu unterscheiden, **wer** angreift. Das könnten sowohl Outsider (Angreifer auf dem Netz) als auch Insider (Mixe oder Teilnehmer, also Sender oder Empfänger von Nachrichten) sein (Abbildung 3-5).
- Die **Stärke** der Angreifer muß beschrieben werden.
- Entscheidend sind auch **Systemparameter**, wie z.B.
  - die Organisation des Puffers (Batch, Pool oder Mischformen),
  - die Puffergröße oder
  - Aussagen über die Charakteristik des Ein- und Ausgabestroms von Nachrichten oder über den gewählten Weg durch das Mixnetz (Benutzung einer Kaskade oder Benutzung von Mixen in freier Reihenfolge).

In einem allgemeinen Angreifermodell werden Annahmen über die einzelnen Parameter getroffen. Die Verfahren fordern bestimmte Systemparameter als Voraussetzung. Anhand dieser Voraussetzungen kann die erreichte Sicherheit des Verfahrens unter einem eingegrenzten Angreifermodell betrachtet werden. Die ersten beiden Punkte - mögliche Angreifer und deren Stärke - sind unabhängig von einem speziellen Verfahren.

Die Stärke von Angreifern kann nach [Pfit_95] folgendermaßen charakterisiert werden: **Beobachtende** Angreifer führen im System nur erlaubte Handlungen und auch alle vorgeschriebenen aus, während **verändernde** Angreifer auch unerlaubte Handlungen im System ausführen oder vorgeschriebene unterlassen. Darüber hinaus kann noch unterschieden werden zwischen **passiven** und **aktiven** Angreifern. Passive Angreifer greifen nicht in den normalen Betrieb des Systems ein, sie beobachten nur die Aktionen des Angegriffenen. Aktive Angreifer dagegen führen schon vor dem eigentlichen Angriff gezielte Aktionen aus, mit denen sie den Angegriffenen zum vom Angreifer beeinflußten Einsatz seines kryptographischen Systems bringen.

In Tabelle 3-2 werden die bisher beschriebenen Angriffe aufgrund dieser Definitionen eingeteilt. Weitere aktive Angriffe werden in [PfPW1_89] beschrieben.

beobachtend	**passiv**	Abhören der Kommunikation auf dem Netz
		(n-1)-Angriff eines einzelnen Angreifers (wartet auf günstige Zusammensetzung des Puffers)
		(n-1)-Angriff mehrerer Angreifer (Zusammenschluß)
	aktiv	(n-1)-Angriff eines einzelnen Angreifers durch gezieltes Senden von Nachrichten (Fluten)
		(n-1)-Angriff mehrerer Angreifer durch gezieltes Senden von Nachrichten (Fluten, Zusammenschluß)
		replay-Angriffe
verändernd	**aktiv**	(n-1)-Angriffe durch Stören der Kommunikation anderer Teilnehmer (Komm. verhindern; Nachr. verzögern)

Tabelle 3-2: Einteilung der bisher beschriebenen Angriffe

Wie aus der Tabelle ersichtlich, sind mögliche Strategien für einen (n-1)-Angriff der Zusammenschluß von Angreifern sowie das Fluten und Verzögern von Nachrichten.

Im folgenden soll dieser Angriff näher betrachtet werden. Die Teilnehmer können immer nur innerhalb einer Gruppe von nichtangreifenden Teilnehmern anonym sein. Absolute Aussagen darüber, wieviele und welche der Teilnehmer Angreifer sind, können nicht getroffen werden. Es kann also nur mit einer gewissen Wahrscheinlichkeit Unverkettbarkeit erreicht werden. Die Funktionen des Mixes sollen so kombiniert werden, daß unter bestimmten Voraussetzungen dieser Angriff - ob nun durch einen einzelnen oder durch mehrere Angreifer - verhindert wird.

3.3.3 Modifikation des Modells zur Verhinderung von (*n*-1)-Angriffen

Übergeordnetes Kriterium der folgenden Angriffe ist die *Charakteristik des Eingabestroms* (Annahme A, B). Unter entsprechenden Annahmen werden Angreifer verschiedener *Stärke* betrachtet. Der Einfluß der *Puffergröße* stellt ein weiteres Unterscheidungsmerkmal dar. Die Verifizierbarkeit von Absendern - eine zur Verhinderung des Zusammenschlusses von Angreifern wichtige Funktion - wird in den folgenden Modellen mit „vAbs“ gekennzeichnet.

Annahme A: Die Teilnehmer senden immer zu festen Takten, ggf. dummies.
Das bedeutet, daß sich zu allen Takten genügend viele Nachrichten von allen Teilnehmer im Puffer befinden. Entsprechend dem Angreifermodell lassen sich verschiedene Modelle konstruieren.

A.1) beobachtende Angreifer

A.1.a) passive Angreifer

Im Puffer befinden sich genügend viele Nachrichten von genügend vielen verschiedenen Absendern. Die Angreifer könnten versuchen, sich zusammenzuschließen, um gemeinsam einen (n-1)-Angriff durchzuführen. Zwei Fälle sind zu unterscheiden:

- *Der Puffer ist groß genug, um den Zusammenschluß der Angreifer zu verhindern.*
  In diesem Fall sind keine Maßnahmen erforderlich.
- *Die Puffergröße ist nicht ausreichend groß.*
  Ein (n-1)-Angriff eines einzelnen Angreifers ist nach wie vor nicht möglich. Der Erfolg eines Angriffs durch Zusammenschließen von Angreifern muß jedoch durch weitere Maßnahmen (Erzeugung von dummies, Latenzzeiten) erschwert werden (Abbildung 3-6).

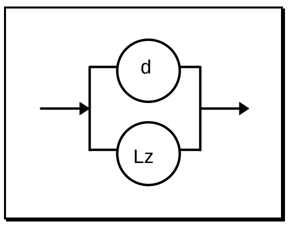


Abbildung 3-6: Maßnahmen bei beobachtenden, passiven Angreifern bei kleiner Puffergröße

A.1.b) aktive Angreifer (senden Mix viele Nachrichten zu)

Es befinden sich immer noch genug Nachrichten im Puffer, aber es ist nicht gewährleistet, daß sie auch von genügend vielen verschiedenen Absendern stammen. Wieder sind zwei Fälle zu unterscheiden:

- *Der Puffer ist groß genug.*
  Da alle Teilnehmer ständig senden, wird es dem Angreifer nicht gelingen, in einen großem Puffer für *n* Nachrichten *n*-1 eigene zu plazieren. Besondere Schutzmaßnahmen sind also nicht erforderlich.
- *Der Puffer ist nicht groß genug.*
  In diesem Fall sind die Absender zu prüfen, falls diese verifizierbar sind. Da der Puffer nicht ausreichend groß ist, könnten sich auch Angreifer zusammenschließen. Weitere Maßnahmen sind also unabhängig von der Verifizierbarkeit der Absender erforderlich (Abbildung 3-7).

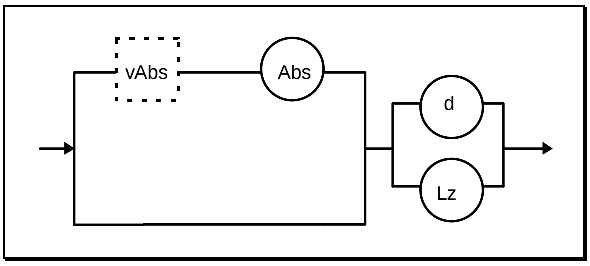


Abbildung 3-7: Maßnahmen gegen beobachtende, aktive Angreifer bei kleiner Puffergröße

A.2) verändernde Angreifer (die hier aufgeführten Angriffe sind aktiv)

Verändernde Angreifer können die Kommunikation der anderen Teilnehmer mit dem Mix stören. Die Wahrscheinlichkeit, daß sich im Mix dann genügend Nachrichten eines Angreifers für einen (*n*-1)-Angriff befinden, erhöht sich damit. Ebenso wird der Zusammenschluß von Angreifern erleichtert. Die Angreifer können außerdem Nachrichten mit gefälschten Absendern an den Mix schicken, falls Absender nicht verifizierbar sind. Auch hier sind die beiden Fälle zu unterscheiden:

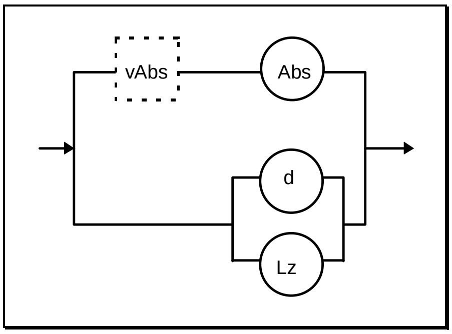


Abbildung 3-8: Maßnahmen gegen verändernde, aktive Angriffe bei großem Puffer

- *Der Puffer ist groß genug.*
  Im Gegensatz zu bisherigen Fällen sind auch in diesem Fall Schutzmaßnahmen erforderlich. Falls die Absender verifizierbar sind, ist eine Absenderprüfung vorzunehmen. Als Alternative dazu können dummies erzeugt bzw. Latenzzeiten verwendet werden (Abbildung 3-8).
- *Der Puffer ist nicht groß genug.*
  Die Absender sind ggf. zu prüfen. Darüber hinaus sind aber auf alle Fälle weitere Maßnahmen zu ergreifen (wie Abbildung 3-7).

Annahme B: Die Teilnehmer senden nur bei Bedarf.

Bei dieser Annahme kann nicht davon ausgegangen werden, daß sich normalerweise genügend viele Nachrichten von genügend vielen verschiedenen Absendern im Puffer befinden. Die Prüfung der Absender ist von noch größerer Bedeutung.

B.1) beobachtende Angreifer

B.1.a) passive Angreifer

Ist der Puffer ausreichend groß, sind keine Maßnahmen erforderlich. Ist er jedoch klein genug, daß (*n*-1) bekannte Nachrichten zusammenkommen könnten, so sind ggf. die Absender zu prüfen oder alternativ dummies zu erzeugen bzw. Latenzzeiten zu behandeln. Daraus ergibt sich ein Modell wie in Abbildung 3-8.

B.1.b) aktive Angreifer

Senden die Angreifer dem Mix viele Nachrichten zu, ohne daß die anderen Teilnehmer auch ständig senden, so sind unabhängig von der Puffergröße Maßnahmen erforderlich. Die Absender sind, falls möglich, zu prüfen. Außerdem ist der Erfolg eines Angriffs durch Zusammenschluß von Angreifern zu verhindern. Das Modell entspricht dem in Abbildung 3-7 gezeigten.

B.2) verändernde Angreifer (wiederum aktiv)
Der Angriff wird durch die Tatsache, daß die Teilnehmer nur bei Bedarf senden, erleichtert. Die Behinderung der Kommunikation ist nun nur für die Teilnehmer durchzuführen, welche wirklich senden wollen. Die Absender müssen überprüft und außerdem entweder dummies erzeugt oder Latenzzeiten behandelt werden. Auch hier ergibt sich das Modell aus Abbildung 3-7.

Betrachtet man den (n-1)-Angriff, so zeigt sich, daß das Vergrößern der Ungewißheit eines Angreifers über die Zusammensetzung des Puffers die wesentliche Aufgabe darstellt. Da eine sinnvolle Absenderprüfung nur bei verifizierbaren Absendern möglich ist, hat die Erzeugung von dummies bzw. die Verwendung von Latenzzeiten große Bedeutung. Daraus ergeben sich jedoch neue Probleme (siehe 2.1 sowie 3.3.4)

3.3.4 Diskussion weiterer Angriffsmöglichkeiten

Aus den unter 3.3.2 genannten Einflußgrößen für die Betrachtung von Sicherheit wurde hier nur ein Teil herausgegriffen. Es wurde nur ein Angriff näher untersucht, in welchem die Teilnehmer als Angreifer auftreten.

Obwohl m-1 der verwendeten m Mixe laut Angreifermodell korrupt sein dürfen, wurden die Mixe hier nicht explizit als Angreifer behandelt. Wollen sich die Teilnehmer gegen Angriffe der Mixe absichern, müssen sie die Möglichkeit haben, die Mixe zu kontrollieren. Kontrollierbar sind nur Operationen, welche deterministisch ablaufen. Haben die Mixe Gelegenheit, indeterministisch zu arbeiten, also etwa durch die Erzeugung von dummies, ergeben sich Möglichkeiten für nicht oder schwer entdeckbare Angriffe der Mixe. Die zum Schutz vor (n-1)-Angriffen vorgeschlagenen Maßnahmen sind demzufolge nicht ohne weiteres einzusetzen.

Daraus lassen sich weitere Ansatzpunkte für die Betrachtung von Angriffen ableiten. Auch für andere Funktionen ist zu untersuchen, ob sich für einen Mix Gelegenheiten zu unkontrollierbaren Manipulationen ergeben und ob ein vertrauenswürdiger Mix trotzdem noch Unverkettbarkeit garantiert.

So ist auch der Poolbetrieb für die Teilnehmer nicht überprüfbar. Verzögert der Mix einzelne Nachrichten, kann ihm nicht nachgewiesen werden, daß er damit einen Angriff versucht hat.

Es ist für getroffene Maßnahmen immer zu betrachten, gegen welche Angreifer Sicherheit erreicht werden kann und wessen Kontrolle erschwert wird. Es ist anzunehmen, daß es nicht möglich ist, „allgemeine Sicherheit“ zu erreichen.

Die Angriffe auf einen einzelnen Mix sind auf Angriffe auf die gesamte Mixfolge auszuweiten. Die beiden Varianten Kaskade und freie Folge sind zu betrachten. Die Kaskade bietet gute Möglichkeiten der Kontrolle für die Teilnehmer. In einer freien Folge dagegen ist die Kontrolle sehr schwer - jeder Mix kann sowohl Nachrichten von Teilnehmern als auch weitergeleitete Nachrichten anderer Mixe erhalten. Für einen Angreifer erhöht sich in einem Netz die Komplexität seiner Angriffsversuche, aber die Angriffe sind schwerer zu entdecken. Die verschiedenen Optionen zur Gestaltung eines Mixes müssen auch für die Folge von Mixen betrachtet werden. Besonders bei freien Folgen kann sich von Mix zu Mix die Konfiguration ändern.

Die Möglichkeit der Kontrolle bietet den Teilnehmern den besten Schutz gegen Angriffe der Mixe. Es ist jedoch auch zu betrachten, ob die für Sicherheit notwendigen Kontrollmaßnahme überhaupt effizient genug durchführbar sind.

3.4 Modifiziertes Modell

Nach den Diskussionen in 3.1 und 3.2 wird ein modifiziertes Modell vorgestellt (Abbildung 3-9). *Ein* allgemeines „sicheres“ Modell wird nicht angegeben. Durch die Komplexität der Einflußgrößen gilt die Sicherheit immer nur unter bestimmten Annahmen.

Das Modell teilt die Funktionen in drei Abschnitte auf:
1) realisiert den Test auf Wiederholungen sowie Umkodierung,
2) stellt die notwendigen Kombinationen von Organisationsformen des Puffers und der Funktion Umsortieren dar und in
3) werden weitere Funktionen kombiniert, um (*n*-1)-Angriffe abzuwehren.

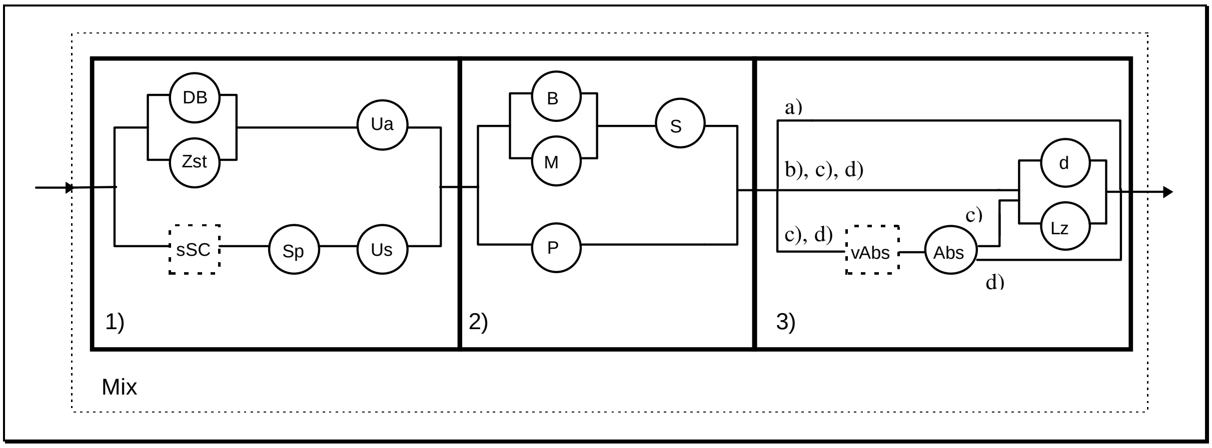


Abbildung 3-9: Modifiziertes Modell eines Mixes

Über die Sicherheit des Modells können folgende Aussagen getroffen werden: Durch die ersten beiden Teile wird hinreichende Sicherheit gegenüber beobachtenden, passiven Angreifern garantiert. Die Kombinationen in 1) gewähren außerdem hinreichende Sicherheit gegen replay-Angriffe beobachtender, aktiver Angreifer.

Der Einsatz von dummies und Latenzzeiten erhöht zwar die Unsicherheit angreifender Teilnehmer über den Inhalt des Puffers, stellt jedoch indeterministische Operationen dar. Über die erreichte Sicherheit im dritten Teil können darum keine allgemeinen Aussagen getroffen werden (siehe auch 2.1, 3.3.4). Unter der Annahme, daß nur die Teilnehmer angreifen, stellen die Wege a) bis d) notwendige Kombinationen dieser Funktionen dar, welche Maßnahmen gegen folgende Angreifer bieten:

Falls alle Teilnehmer ständig senden:
- beobachtende, passive / aktive Angreifer (ausreichend großer Puffer) (a),
- beobachtende, passive Angreifer (kein ausreichend großer Puffer) (b),
- beobachtende / verändernde, aktive Angreifer (kein ausr. großer Puffer) (c),
- verändernde, aktive Angreifer (ausreichend großer Puffer) (d).

Falls die Teilnehmer nur bei Bedarf senden:
- beobachtende, passive Angreifer (ausreichend großer Puffer) (a),
- beobachtende / verändernde, aktive Angreifer (kein ausr. großer Puffer) (c),
- beobachtende, passive Angreifer (kein ausreichend großer Puffer) (d).

4 Zusammenfassung und Ausblick

Ausgehend von Definitionen des Mixes in [Chau_81, Pfit_89, PfPW1_89] wurden Erweiterungen dieses Konzepts aus Sicherheits- und Effizienzgründen betrachtet. Daraus wurde ein allgemeines Modell abgeleitet, welches auf bekannte, teilweise Sicherheit bietende Anwendungsbeispiele angewendet wurde.

Aus der Systematisierung der verschiedenen aufgezeigten Varianten der einzelnen Funktionen eines Mixes wurde ein mögliches ablauforientiertes Modell entworfen.

Dieses Modell deckt die betrachteten Anwendungsbeispiele ab, enthält damit jedoch auch „unsichere“ Konfigurationen.

Aus den Forderungen an einen Mix ergeben sich einige notwendige Einschränkungen der Wege durch dieses Modell. Die Betrachtung von Angriffen führt zu weiteren Kombinationsmöglichkeiten.

Um Aussagen über Angreifermodelle für Mixe zu machen, wurden mögliche Einflußgrößen auf die erreichte Sicherheit zusammengetragen. Am Beispiel des (n-1)-Angriffs wurden die Funktionen eines Mixes unter verschiedenen Annahmen über solche Einflußgrößen, speziell Stärke der Angreifer und Systemparameter, kombiniert.

Ein „einfaches“ allgemeines Modell für einen „sicheren“ Mix läßt sich aufgrund der Komplexität der Annahmen nicht angeben. Die vorgestellten Modelle sind mögliche Sichten auf den Mix unter den oben beschriebenen Annahmen. Weitere Bedingungen erfordern die Betrachtung weiterer Modelle. Aufbauend auf vorliegenden Modellierungen, wird die Bewertung der verschiedenen Konfigurationen Ziel zukünftiger Arbeiten sein. Schwerpunkt der Bewertung wird wiederum die Betrachtung möglicher Angriffe und in Zusammenhang damit die Betrachtung der erreichten Anonymität sein. Die skizzierten Angriffe sind weiter zu untersuchen.

Ein Dankeschön für Kritik- und Diskussionsbereitschaft an Hannes Federrath, Andreas Graubner und Jan Müller.

5 Literatur

Chau_81 D. Chaum: Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms; Communications of the ACM 24/2 (1981) 84-88.

FaKK_96 Andreas Fasbender, Dogan Kesdogan, Olaf Kubitz: Analysis of Security and Privacy in Mobile IP. 4th International Conference on Telecommunication Systems, Modelling and Analysis, Nashville, March 21-24, 1996.

FFJM_97 Hannes Federrath, Elke Franz, Anja Jerichow, Jan Müller, Andreas Pfitzmann: Ein Vertraulichkeit gewährendes Erreichbarkeitsverfahren (Schutz des Aufenthaltsortes in künftigen Mobilkommunikationssystemen), GI Fachtagung Kommunikation in Verteilten Systemen (KiVS) 97, 17.-22.2.97 in Braunschweig.

GüTs_96 C. Gülcü, G. Tsudik: Mixing Email with BABEL; Proc. Symposium on Networking and Distributed System Security, San Diego, IEEE Comput. Soc. Press, 1996, pp 2-16.

Pfit_89 A. Pfitzmann: Diensteintegrierende Kommunikationsnetze mit teilnehmerüberprüfbarem Datenschutz; Universität Karlsruhe, Fakultät für Informatik, Dissertation, Feb. 1989, IFB 234, Springer-Verlag, Berlin 1990.

Pfit_93 A. Pfitzmann: Technischer Datenschutz in öffentlichen Funknetzen; Datenschutz und Datensicherung, DuD 17/8 (1993), 451-463.

Pfit_95 A. Pfitzmann: Datensicherheit und Kryptographie; Vorlesungsskript TU Dresden, WS 1995/96.

PfPW1_89 Andreas Pfitzmann, Birgit Pfitzmann, Michael Waidner: Telefon-MIXe: Schutz der Vermittlungsdaten für zwei 64-kbit/s-Duplexkanäle über den (2•64 + 16)-kbit/s-Teilnehmeranschluß; Datenschutz und Datensicherung DuD /12 (1989) 605-622.

PfPW_88 Andreas Pfitzmann, Birgit Pfitzmann, Michael Waidner: Datenschutz garantierende offene Kommunikationsnetze; Informatik-Spektrum 11/3 (1988) 118-142.

Scha_95 Steffen Schaarschmidt: Entwurf und Implementierung eines anonymen Remailers mit MIX-Funktionalität; Großer Beleg, TU Dresden, Institut für Theoretische Informatik, September 1995.