Universität Karlsruhe (TH)
Institut für Telematik

TeleMatics Technical Reports

Datenschutz in Kommunikationsnetzen

Seminar SS06

Thomas Gamer, Christoph Sorge, Dr. Oliver Raabe
Prof. Dr. Martina Zitterbart
{gamer,sorge,zit}@tm.uka.de, raabe@ira.uka.de

September, 1st 2006

TM-2006-3

ISSN 1613-849X

http://doc.tm.uka.de/tr/

Institute of Telematics, University of Karlsruhe
Zirkel 2, D-76128 Karlsruhe, Germany

Vorwort

Das Seminar “Datenschutz in Kommunikationsnetzen” wurde im Sommersemester 2006 in Form eines Blockseminars am 26. Juli 2006 am Institut für Telematik abgehalten. In diesem Seminarband sind die Ausarbeitungen der Studenten in Form eines internen Berichts zusammengefasst.

Inhaltsverzeichnis

Techniken zur Sicherung von Anonymität im Internet

Marco Konigorski

Kurzfassung

Anonymität im Internet ist ohne zusätzliche Maßnahmen nicht gegeben. Um diese Sicherheit der eigenen Persönlichkeit und der eigenen personenbezogenen Daten dennoch gewährleisten zu können, wurden einige Verfahren entwickelt, die mehr oder weniger den Schutz der eigenen Identität sichern sollen. Beim Vergleich dieser Verfahren ist nicht nur das Ausmaß der Sicherheit, sondern auch die Nutzerfreundlichkeit und die Anwendbarkeit zu betrachten. Da sicher nicht jedes der verfügbaren Verfahren für jeden Anwender bzw. jede Kommunikationsform geeignet ist, ist ein Vergleich der Methoden in den verschiedenen Aspekten (wie zum Beispiel Performance) unvermeidbar.

1 Einleitung

Anonymität und Vertraulichkeit ist ein wichtiger Aspekt bei vielen Aktivitäten im täglichen Leben. Im Internet ist Anonymität von gleicher Bedeutung, aufgrund der offenen Struktur jedoch nicht gegeben. Für einen Angreifer, der eine Nachricht abfangen kann, ist es ohne Probleme möglich, die IP-Adresse des Absenders aus den Adressdaten des Nachrichten-Headers auszulesen. Mit diesen Daten kann er den Sender noch nicht direkt identifizieren, aber aufgrund der Zuordnung zu einem Provider gegebenenfalls schon regional eingrenzen.

Für viele Aktivitäten im täglichen Leben ist das Internet so gut wie unverzichtbar geworden, da es vieles erheblich erleichtert und vereinfacht. Bestellungen können bequem von zu hause aus getätigt werden, Geld wird mit wenigen Tastendrücken von einem Konto auf ein anderes transferiert. Die Frage nach Anonymität kommt dann ins Spiel, wenn Aktivitäten nicht vor anderen offengelegt werden sollen.

Das Recht auf informationelle Selbstbestimmung, das als Grundrecht anerkannt aber nicht im Grundgesetz verankert ist, ist ein Datenschutz-Grundrecht, dass es jedem Einzelnen ermöglicht, frei über die Verwendung und Offenlegung seiner personenbezogenen Daten zu verfügen. Der Wunsch, dieses Recht für sich selbst in Anspruch zu nehmen, muss nicht mit illegalen Handlungen verbunden sein. Oftmals kann es vorkommen, dass die Aktivitäten dem persönlichen Ruf schaden könnten. So wäre sicher niemand daran interessiert, wenn jeder ohne großes Zutun aus einer abgefangenen Bestellung bei einer Online-Apotheke auf den persönlichen Gesundheitszustand schließen könnte.

1.1 Gliederung

In dieser Einleitung soll, neben der hier angeführten Gliederung, ein Einblick in die bekannten Angriffsarten auf Anonymisierungsdienste gegeben werden. Diese sind teils theoretischer, teils praktischer Natur, werden aber dennoch zur Prüfung der Sicherheit eines Verfahrens herangezogen.

Abschnitt 2 befasst sich mit der Möglichkeit zur Bewertung der Sicherheit bezüglich der Anonymität. Hierbei wird als Grundlage auf zwei Abhandlungen verwiesen, in denen die Berechnungen vorgestellt wurden.

Im folgenden Abschnitt 3 wird auf gängige und bekannte Anonymisierungsdienste eingegangen, die ohne großen Aufwand für alle nutzbar sind. Im Anschluss daran werden in den Abschnitt 4 - 6 die Verfahren „Mixes“, „Onion Routing“ und „Crowds“ vorgestellt und näher untersucht.

Im abschließenden Abschnitt 7 wird eine Zusammenfassung der vorgestellten Verfahren und eine Bewertung sowie eine Gegenüberstellung derselbigen erstellt.

1.2 Angriffsszenarios

Ein System zur Wahrung der Anonymität muss bestimmten Angriffsarten standhalten, die im Folgenden erklärt werden sollen. Bei allen Verfahren wird davon ausgegangen, dass ein Angreifer immer das gesamte Netzwerk beobachten kann.

- **Message coding attack** (Analyse der Nachrichtendarstellung):
  Eine Nachricht, deren Darstellung sich auf dem gesamten Weg vom Sender zum Empfänger nicht ändert, kann von einem Beobachter verfolgt und somit Ziel und Sender dieser Nachricht ermittelt werden.

- **Timing attack** (Analyse der Sendezeiten auf Zwischensystemen):
  Nachrichten, die ein Zwischensystem passieren, auf dem sie sofort verarbeitet und weitergeleitet werden, können anhand ihres zeitlichen Aufeinanderfolgens unterschieden und damit der Weg vom Sender zum Empfänger ermittelt werden.

- **Message volume attack** (Analyse der Nachrichtengröße):
  Nachrichten, die in ihrer Größe gleich bleiben, können von anderen Nachrichten unterschieden werden, wenn sie ein System verlassen. Dadurch kann eine bestimmte Nachricht über den gesamten Weg verfolgt und eine Beziehung zwischen Sender und Empfänger ermittelt werden.

- **Flooding attack** (Überlasten von Zwischensystemen zur Separierung einzelner Nachrichten):
  Wenn ein Angreifer ein Zwischensystem mit eigenen, böswilligen Nachrichten so weit überlasten läßt, dass außer den eigenen Angriffsnachrichten nur noch eine legitime Nachricht verarbeitet werden kann, die beobachtet werden soll, kann diese Nachricht verfolgt werden. Wiederholt man diese Technik bei allen Zwischensystemen, kann der gesamte Weg vom Sender zum Empfänger korrumpiert und damit die Beziehung zwischen Sender und Empfänger ermittelt werden.

- **Intersection attack** (Analyse typischer Userverhalten):
  Die Kommunikation der meisten Internetnutzer beschränkt sich auf wenige Zielsysteme (Email, Websites, Newsgroups, etc.). Das Online-Verhalten ist ein weiterer Punkt, der beobachtet werden kann und anhand dessen sich verschiedene Nutzerprofile unterscheiden lassen.Ein Angreifer kann sich diese Tatsache zu Nutze machen, indem er über längere Zeit das Verhalten der Nutzer beobachtet und protokolliert. Auf diese Art kann zwar keine direkte Erkennung des Nutzers erreicht werden, jedoch kann eine Nachricht mittels Schnittmengenbildung mit den aufgezeichneten Verhaltensweisen auf einen kleineren Nutzerkreis eingeschränkt und damit die Wahrscheinlichkeit einer Zuordnung erhöht werden.

- **Collusion attack** (Zusammenarbeit zwischen Betreibern von Zwischensystemen): Wenn in einem dezentralen Kommunikationssystem mehrere Zwischensysteme korrumpiert werden, bzw. deren Betreiber zusammenarbeiten, kann die Anonymität der Benutzer insofern gefährdet werden, als dass Nachrichten von Sender zu Empfänger über die betroffenen Systeme geleitet werden. Nach außen erscheint der Ablauf normal, jedoch kann somit von den Beobachtern gemeinschaftlich eine Beziehung zwischen Quelle und Ziel ermittelt werden.

2 Bewertung der Anonymität

2002 wurde auf der Privacy Enhancing Technology (PET) Conference zwei Abhandlungen vorgestellt (siehe [DSCP02] und [SeDa02]), die sich mit der Bewertung von Anonymität und anonymitätsfördernden Verfahren auseinandersetzen. Beide erstellen eine Bewertung des Grades der Anonymität anhand der Entropie des Netzes. Da sich beide in der Notation etwas unterscheiden, wird hier die Notation von [DSCP02] verwendet.

Die Abhandlung wurde auf der Grundlage von Mixe-Systemen erstellt, jedoch wird auch gezeigt, dass es sich ohne Probleme auf andere Verfahren anwenden läßt. In Abschnitt 4.3 soll das Beispiel aus [DSCP02] aufgegriffen und erläutert werden.

Die Entropie des Netzes $H(X)$ berechnet sich aus der Gesamtzahl der im Netz befindlichen Knoten N und der Wahrscheinlichkeiten $p_i$für jeden einzelnen Knoten i, der Sender zu sein, zu

$$H(X) := -\sum_{i=0}^{N-1}[p_i \cdot log_2(\frac{1}{p_i})]$$

Die maximale Entropie wird erreicht, wenn die Wahrscheinlichkeit für alle Knoten gleich groß ist ($(p_i = \frac{1}{N})$ für alle i). Sie berechnet sich zu

$$H_M := log_2(\frac{1}{N})$$

An dieser Stelle unterscheiden die beiden Abhandlungen in eine „bounded degree“(gebundene Bewertung - die maximale Entropie fließt in die Berechnung der Bewertung mit ein) und eine „unbounded degree“(ungebundene Bewertung - die Bewertung wird nur aufgrund der Gesamtentropie ohne Berücksichtigung der Maximalentropie vorgenommen). Wieder Bezug nehmend auf wird hier nur die erste Variante unter Zuhilfenahme der Maximalentropie betrachtet. Diese definiert einen Bewertungsgrad d für das Maß der Anonymität, welcher sich zu

$$d := 1 - \frac{H_M - H(X)}{H_M} = \frac{H(X)}{H_M}$$

berechnet. Anhand dieses Bewertungsgrades können Anonymität und Verfahren zu Sicherung derselben miteinander verglichen werden.

3 Anonymizer

Obwohl generell jedes System, das der Wahrung der Anonymität im Internet dient, als Anonymizer bzw. Anonymisierer bezeichnet werden kann, wird in diesem Abschnitt auf Möglichkeiten eingegangen, die schnell und ohne zusätzliche Software eingesetzt werden können.

3.1 Web-Anonymizer

Bei einem Web-Anonymizer handelt es sich um einen frei verfügbaren Dienst, der als Zwischenschicht für den Zugriff auf Webseiten fungiert. Hierbei wird der Anonymizer selber als Website betrieben, auf welcher man mittels dort angebotener Eingabezeile die gewünschte Ziel-Website anfordern kann. Der Dienst gibt sich in diesem Fall als Anfragender aus und verschleiert somit die Idendität des eigentlichen Nutzers.

Verschieden Web-Anonymizer lassen sich kaskadieren (hintereinanderschalten), indem man mehrere von ihnen hintereinander aufruft, bevor man die eigentliche Seite anfordert. Diese Vorgehensweise ist jedoch sehr zeitaufwändig, da zuerst alle Anonymizer von Hand aufgerufen und geladen werden müssen.

Bei diesem Verfahren ist Anonymität in erster Linie gegenüber dem Endsystem und den nach dem Anonymizer angesiedelten Zwischensystemen gegeben. Damit ein Angreifer das eigentliche Ziel nicht aus dem Mithören der Kommunikation zwischen Nutzer und Anonymizer ermitteln kann, muss zwischen diesen beiden Systemen eine Verschlüsselung stattfinden. Hierzu setzen manche Dienste auf das SSL-Protokoll, jedoch übertragen die meisten Anonymizer die Daten unverschlüsselt, so dass in dem Fall kein ausreichender Schutz der Daten gegeben ist.

Da es sich bei den Web-Anonymisierern prinzipiell um einen zentralen Dienst handelt, muss der Anbieter vertrauenswürdig sein. Um auch nach erfolgreicher Kommunikation noch anonym zu bleiben, dürfen daher keine Aufzeichnungen über IP-Adresse des Nutzers und die besuchte Webseiten erstellt werden.

3.2 Anonyme Proxyserver

Die Funktionsweise von anonymen Proxyservern ähnelt der von Web-Anonymizern. Proxyserver können jedoch für mehr Dienste als ausschließlich WWW genutzt werden, sofern der Anbieter sie dafür freigibt. Im jeweiligen Programm (Browser, Email-Client, Filesharing), welches das Internet nutzt, wird der Server als Proxy eingetragen und leitet die Anfragen, die vom Nutzer kommen, an die jeweilige Adresse weiter. Hierdurch wird wieder die Anwesenheit des eigentlichen Nutzers verschleiert, da der Proxyserver sich als der Anfragende ausgibt.

Je nach Proxyserver kann die Kommunikation verschlüsselt sein, damit auf dem Weg zum Proxy ein Angreifer keinen Einblick in die Daten und das eigentliche Ziel nehmen kann. Ebenso wie beim Anonymisierer muss auch hier wieder dem Anbieter vertraut werden, da er als zentrale Stelle eine Zuordnung von Nutzer und Inhalt ermitteln kann.

Abbildung 1 [1] soll die Verwendung von Proxys veranschaulichen.

Abbildung 1: Anonyme Proxies

3.3 Schutz vor Angriffen

Beide Ansätze bauen auf dem gleichen Prinzip (Stellvertreterprinzip) auf. Einen Schutz gegen Angriffe bieten beide nur in begrenztem Maße, jedoch selbst dann nur gegen message coding

[1] Quelle: http://wwwbs.informatik.htw-dresden.de/svortrag/i00/Reiche/anonym.html)

attacks und auch nur, wenn die Kommunikation vom Nutzer zum Anonymisierungs-Dienst verschlüsselt übertragen wird.

Gegen Verkehrsanalyse durch timing, message volume und intersection attacks besteht bei den beiden Verfahren kein Schutz, da die Nachrichten nur ein System durchlaufen und dort auch sofort weiterverarbeitet werden. Ebenfalls besteht keine Abwehr gegen flooding attacks, da der Dienst von jedem frei genutzt werden kann. Als letztes entfällt auch der Schutz vor collusion attacks, da es sich bei beiden Verfahren um zentrale Dienste handelt und ein Schutz nur durch einen vertrauenswürdigen Anbieter gegeben ist.

4 Mixes

1981 veröffentlichte der Mathematiker David L. Chaum als erster ein technisches Verfahren zur Wahrung der Anonymität über ein unsicheres Medium wie das Internet [Chau81]. Seine Idee baute auf der Verfügbarkeit eines speziell dafür vorgesehenen Zwischensystems (eines sogenannten „Mix“) auf, das jedoch nicht alleine sondern in Kombination mit anderen Mixes (einer sogenannten „Mixe-Kaskade“) für Anonymität des Nutzers sorgen soll.

Das ursprüngliche System von Chaum war zur Anonymisierung von Emails gedacht, lässt sich jedoch auch (bedingt aufgrund der geringeren Performance durch hohe Verzögerungen) für andere Kommunikationsformen im Internet verwenden. Die Daten werden beim System der Mixes nicht nur verschlüsselt übertragen, sondern auf den Systemen selber auch in ihrer Reihenfolge verwürfelt, so dass ein Angreifer aus der Reihenfolge der ein- und ausgehenden Nachrichten keinen Bezug zwischen Sender und Empfänger erstellen kann.

Die Sicherheit des gesamten Mixe-Systems ist dann gewährleistet, wenn mindestens ein Mix nicht korrumpiert ist. Ein Nutzer sollte hierzu immer mehr als einen Mix zur Versendung seiner Daten nutzen und möglichst sicherstellen, dass die Betreiber des Mix keine gemeinsamen Interessen verfolgen. Die Auswahl eines Betreibers hängt jedoch einzig und allein vom Vertrauen des jeweiligen Nutzers ab, wenngleich in die einzelnen Betreiber weniger Vertrauen als in die Betreiber eines anonymen Proxys gesetzt werden muss. Geeignete Mixe-Betreiber könnten Organisationen sein, die generell auf Diskretion der Daten ihrer Kunden Wert legen, wie z.B. Banken oder Verbraucherschutzorganisationen.

Die allgemeine Funktionsweise von Mixes soll Abbildung 2 verdeutlichen.[2]

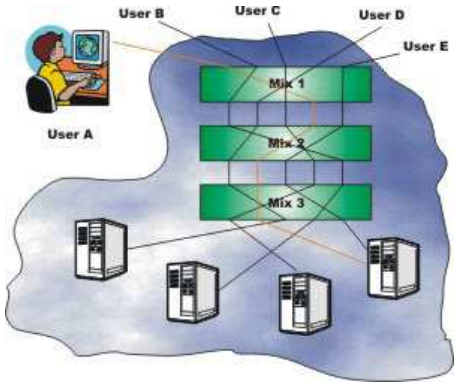


Abbildung 2: Kommunikation über Mixe-Kasskaden

[2]Quelle: http://www.ercim.org/publication/Ercim_News/enw49/federrath.html

4.1 Funktionsweise

Ein Mix bildet ein Zwischensystem innerhalb des Netzwerks, das zur Weiterleitung von Daten genutzt werden kann. Um die Idee von Chaum umsetzen zu können, müssen die Mixes über die Möglichkeit zur Public-Key-Kryptographie und dementsprechend über ein eigenes Schlüsselpaar verfügen.

Zunächst erstellt der Nutzer eine Mixe-Kaskade, eine Kette aus verschiedenen Mixe-Rechnern, an deren Ende das eigentliche Zielsystem steht. Im Anschluss daran werden die Daten vor dem Versenden vorbereitet.

Als ersten Schritt verschlüsselt der Sender die Daten mit dem öffentlichen Schlüssel des Endsystems. Damit dieses neue Paket das Endsystem auch erreichen kann, muss der Sender die Adressdaten des Endsystems an die neuen Daten anhängen. Diese neuen Gesamtdaten verschlüsselt er nun mit dem öffentlichen Schlüssel des Mixe-Rechners, der als letzter in der Kaskade und somit vor dem Endsystem steht. Dieses Vorgehen aus Verschlüsseln und Anhängen der betreffenden Adressinformationen wiederholt der Sender solange, bis er die gesamte Kette aufgebaut und das Datenpaket letztendlich mit dem Schlüssel des ersten Mixe-Rechners in der Kaskade verschlüsselt hat.

Das so vorbereitete Paket versendet er an den ersten Rechner innerhalb der Kaskade, welcher eine Entschlüsselung der Daten vornimmt und das darin enthaltene Paket an den nächsten Rechner der Kette weiterleitet, dessen Adresse er den entschlüsselten Daten entnehmen kann. Hierzu soll Abbildung 3 den Ablauf verdeutlichen.

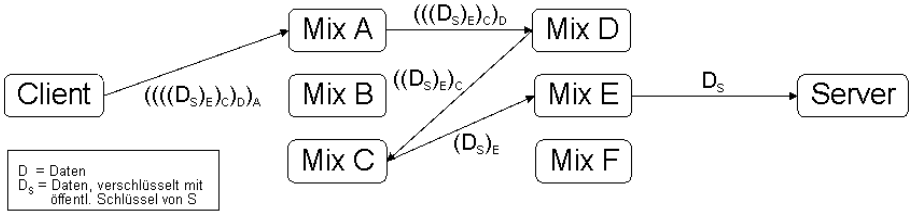


Abbildung 3: Ablauf einer Kommunikation über Mixe

Da dieses Vorgehen alleine noch keinen ausreichenden Schutz der Anonymität gewährleisten würde, wurde im Mixe-Konzept eingeführt, dass die verarbeitete Nachricht nicht direkt weitergeleitet wird, sondern zuerst mehrere Nachrichten gesammelt und diese dann in verwürfelter Reihenfolge weitergeleitet werden. Hier unterscheiden sich die verschiedenen Mixe-Systeme in zwei unterschiedlichen Vorgehensweisen:

- **Pool-Mixe**
  Pool-Mixe orientieren sich an der ursprünglichen Idee von Chaum, bei der ein Mixe-Rechner wartet, bis mehrere Nachrichten eingegangen sind und diese dann in anderer Reihenfolge weiterleitet. Jedoch erweiterten Pool-Mixe diese Idee um einen Nachrichtenpool, der auch einige Nachrichten aus vorangegangenen Runden aufbewahrt. Diese Arten von Mixe arbeiten rundenbasiert, wobei nach jeder Runde Nachrichten weitergeleitet werden, sofern sie nicht in die nächste Runde miteinbezogen werden sollen.

  Das Ende eine Runde kann, je nach System, auf zwei Arten gekennzeichnet sein: nach Erhalt der n-ten Nachricht (threshold mixes) oder nach Ablauf eines Timers (timed mixes). Die Mischung der Nachrichten über Runden hinweg erhöht die Anonymität der einzelnen Nachrichten, zeitgleich aber auch deren Verzögerung.

- **Continous Mixes** (Kontinuierliche Mixe)
  Kontinuierliche Mixe verwürfeln die Nachrichten ebenso wie beim normalen Mixe-Konzept, überlassen jedoch dem Nutzer die Entscheidung über die Verzögerung. Dieser

erstellt für jede Mixe-Station auf dem Weg eine Verzögerungszeit, die das betreffende System aus der Nachricht auslesen und dementsprechend die Nachricht zurückhalten kann, bevor sie sie weiterleitet.

Ein derartiges System erscheint auf den ersten Blick benutzerfreundlicher, kann jedoch die Sicherheit der Anonymität gefährden oder ganz aufheben, wenn die Parameter so gesetzt werden, dass es einem Angreifer wieder ermöglicht wird, aus den Nachrichtenaus- und -eingängen einer Mixe-Station den Weg einer Nachricht zu rekonstruieren und somit das Zielsystem einem anfragenden Nutzer zuzuordnen.

Abbildung 4 stellt die Verwürfelung der eingehenden Nachrichten auf einem Mix dar.

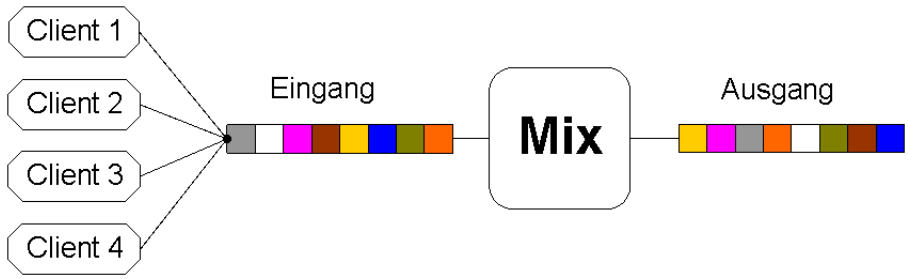


Abbildung 4: Verwürfelung auf einem Mix-Rechner

4.2 Schutz vor Angriffen

Die Verschlüsselung alleine bietet einen Schutz der übertragenen Daten und verhindert message coding attacks, da sich die Darstellung der Nachricht selber bei jedem Entschlüsselungsvorgang ändert. Sie bietet hingegen keine Sicherung des Kommunikationsweges. Ein Angreifer, der das Netz und damit die Mixes beobachtet, könnte aus der Reihenfolge, in der die Pakete dort ein- und ausgehen immer noch ermitteln, welches Paket woher kam und wohin es weitergeleitet wird (timing attack). Um dies zu verhindern, werden auf den einzelnen Mixes die eingehenden Nachrichten gesammelt und mit Verzögerung und in anderer Reihenfolge weitergeleitet. Für den Fall, dass auf einem Mix nur wenige bis gar keine Nachichten eingehen, muss „dummy traffic“ (leere Nachrichten ohne eigentlichen Sinn) versendet werden, da ansonsten ein Angreifer die Kommunikationsbeziehung aufdecken kann, da er bei der Beobachtung des Netzes feststellen kann, wann ein System keine Daten mehr sendet und welches weitere System keine Daten mehr empfängt. Aus diesen Informationen kann er auf die Beziehung der beiden Systeme innerhalb einer Kommunikation schließen.

Um gegen message volume attacks geschützt zu sein, ist es wichtig, dass alle versendeten Nachrichten dieselbe festgelegte Größe besitzen. Hierzu werden lange Nachrichten auf mehrere kleine aufgeteilt und zu kurze Nachrichten mit zufälligen Werten aufgefüllt (padding).

Collusion attacks können in dem dezentral angelegten Mixe-Netz nur dann ausgeführt werden, wenn sich die Betreiber der Mixe-Rechner zusammenschließen. Das System an sich bietet gegen diese Art des Angriffs keinen Schutz, dies muss der Nutzer durch die Auswahl möglichst vieler voneinander unabhängiger Mixes erreichen.

Das Mixe-System, wie es von Chaum erdacht wurde, bietet keinen Schutz gegen flooding attacks, ebensowenig wie gegen inclusion attacks. Um letztere zu vermeiden sollten die Nutzer, die daran interessiert sind, ihre Anonymität zu wahren, generell ihr Online-Verhalten so anpassen, dass es nicht allzu leicht wiedererkannt werden kann. Dazu gehören nicht nur die Zeiten, zu denen der Nutzer online geht und die Online-Dauer, sondern auch beispielsweise die Startseite, die jedesmal beim Öffnen des Browsers aufgerufen werden kann.

4.3 Bewertung der Anonymität

Wie in Abschnitt 2 erwähnt, soll hier auf das Beispiel aus [DSCP02] eingegangen werden. Zur Vereinfachung wird von einem Szenario mit zwei möglichen Sendern und deren Wahrscheinlichkeiten $p_1 = p$ und $p_2 = 1 - p$ ausgegangen. Die maximale Entropie ergibt sich dadurch zu $H_M = log_2(2) = 1$.

Der Bewertungsgrad d erreicht seinen maximalen Wert 1, wenn die Wahrscheinlichkeiten für beide Sender gleich groß sind ($p_1 = p_2$). Für andere Wahrscheinlichkeiten sinkt der Bewertungsgrad, was man sich unabhängig von mathematischen Formeln, so vorstellen kann, dass für jede Nachricht ein User mit größerer Wahrscheinlichkeit der Sender ist und somit eher als Absender identifiziert werden kann.

Um das Beispiel anzushließen, soll noch der Fall $p_1 = 0,75$ und $p_2 = 1 - p_1 = 0,25$ betrachtet werden. In diesem Fall berechnet sich $H(X)$ zu

$$H(X) := -\sum_{i=0}^{N-1}[p_i \cdot log_2(\tfrac{1}{p_i})] \approx 0,8$$

$$d := 1 - \tfrac{H_M - H(X)}{H_M} = \tfrac{H(X)}{H_M} \approx 0,8$$

4.4 Implementierungen

Zum Abschluss des Kapitels sollen hier noch kurz einige Implementierungen des Mixe-Systems vorgestellt werden.

- **Freedom**
  Freedom [t02c] ist eine kanadische Implementierung, die auf dem Prinzip basiert, nur die Funktionalitäten des Mixe-Systems zu realisieren, die zwar die Sicherheit bezüglich Anonymität erhöhen, aber gleichzeitig die Performance nicht stark beeinträchtigen.

- **AN.ON**
  Das AN.ON-System (Anonymität Online) [t02a] wurde von der TU Dresden entwickelt und besteht aus drei Komponenten: einem Java-Client (JAP), verschiedenen Mixe-Stationen und dem Info-Service. Der Client fungiert als Proxy, so dass die Konfiguration aller Software, die das Internet nutzt, sich als relativ einfach gestaltet. Über den Info-Service lassen sich festgelegte Mixe-Kaskaden und deren aktuelle Auslastung abrufen, da die Software dem Nutzer keine freie Mixe-Wahl ermöglicht. Der Client kann, in seiner Funktion als Proxy, auch von mehr als einem Benutzer gleichzeitig benutzt werden und somit beispielsweise ein ganzes Firmen- oder Heimnetzwerk anonymisieren.

- **Mixmaster**
  Mixmaster [t02d] bezeichnet nicht nur eine Client-Software sondern ein eigenes Remailer-Protokoll, das die ursprüngliche Mixe-Idee von Chaum umsetzt. Diverse Clients für dieses Protokoll sind für Windows und Unix-Derivate erhältlich, von denen einige auch das ältere Cypherpunk-Protokoll akzeptieren.

5 Onion Routing

Da Onion Routing das Thema einer anderen Ausarbeitung dieses Seminars ist, soll an dieser Stelle nicht weiter darauf eingegangen werden, sondern nur kurz die Technik als ein Verfahren zur Wahrung der Anonymität erwähnt und grob skizziert werden.

Onion Routing basiert auf dem oben vorgestellten Mixe-Prinzip, jedoch dient hier die Public-Key-Kryptographie nur zur Erstellung eines Kommunikationskanals. Über die Mixe-Pakete wird nicht nur der Aufbau der Kommunikation mit dem Ziel-System übertragen, sondern auch ein Geheimnis (symmetrischer Schlüssel), mit dessen Hilfe die späteren Daten verschlüsselt werden. Hierzu wird zu Beginn eine sogenannte „Onion“ an die beteiligten Router geschickt, die alle benötigten Informationen sowie eine Zeitangabe über die Gültigkeit der Onion enthält. Jeder Router auf dem Pfad bekommt vom Sender in der Initialisierungsphase einen eigenen Schlüssel zugewiesen, so dass die Router selber die Daten nicht einsehen können. Die Router speichern, neben dem Schlüssel und der Gültigkeitszeit, für jede aufgebaute Kommunikation ihren Vorgänger und Nachfolger, sowie eine eindeutige Pfad-ID.

Erhält ein Router Daten, die über einen etablierten Kommunikationspfad geschickt werden sollen, so verschlüsselt er diese Daten mit dem aus der Onion entnommenen Schlüssels und leitet sie an den ihm bekannten Router weiter. Dieser Schritt wird von jedem Router auf dem Weg durchgeführt, so dass am Ende beim Empfänger der Daten eine mehrfach verschlüsselte Nachricht vorliegt. Diese entschlüsselt er seinerseite mit den ihm bekannten Schlüsseln der Router, jedoch in umgekehrter Reihenfolge, um so wieder die ursprünglichen Daten zu erhalten.

Gegenüber dem normalen Mixe-Prinzip hat Onion Routing den Vorteil, dass symmetrische Kryptographie im Vergleich zu Public-Key-Kryptographie wesentlich schneller ausgeführt werden kann und weniger komplex in der Umsetzung ist.

5.1 Implementierungen

Die bekannteste Implementierung von Onion Routing ist Tor [t02e]. Hierbei dient eine Clientsoftware dem Nutzer als Proxyserver, so dass über Tor nicht nur ein einzelner Rechner, sondern auch ein ganzes Netzwert anonymisiert werden kann. Eine Liste aktueller und aktiver Tor-Router wird dabei von einem Verzeichnisserver geladen.

Der Client baut den Pfad schrittweise auf, indem er zuerst die Verbindung zu einem Router etabliert. Sobald diese verschlüsselte Verbindung besteht, wird der Pfad um einen weiteren Router erweitert. Dieses Verfahren wird fortgesetzt, bis der gewünschte Pfad aufgebaut ist. Um einen Schutz vor Reply-Angriffen zu gewährleisten, wird die Verbindung in regelmäßigen Abständen wiederholt.

Tor bietet noch weitere Funktionen, wie z.B. versteckte Dienste. Hierbei ist sichergestellt, dass beide Seiten einer Kommunikation anonym bleiben, so dass auch ein Dienst anonym angeboten werden kann.

Bisher nehmen mehrere Tausend Nutzer am Tor-Netzwerk teil, obwohl die Entwickler eingestehen, dass noch keine starke Anonymität gewährleistet werden kann. Die verfügbaren Server werden von Privatleuten betrieben, und jeder Interessierte kann seinerseits einen Tor-Server eröffnen.

Tor ist unter anderem das Thema einer anderen Arbeit in diesem Seminar, so dass hier, bis auf den vorangegangenen Überblick, nicht näher darauf eingegangen werden soll.

6 Crowds

Crowds bauen auf dem Grundsatz auf, dass ein User sich und seine Kommunikation in der Menge aller Nutzer des Systems verbergen kann, indem eine Nachricht zuerste über verschiedene Nutzerrechner geleitet wird, ehe sie beim eigentlichen Ziel angelangt. Hierbei wird nicht

auf spezielle Router gesetzt, deren Betreibern man vertrauen muss, dass sie keine Aufzeichnungen über Verbindungs- oder Inhaltsdaten führen, da die am System beteiligten User die Aufgabe der Router selber übernehmen. Ein bösartiger Nutzer kann dabei trotz allem nicht rekonstruieren, welche Nachricht, die über seinen Rechner geleitet wird, von welchem Sender stammt.

6.1 Funktionsweise

Zum Crowds-System gehört ein Client („jondo“ (sprich: „John Doe“), der, ebenso wie zum Onion Routing, als Proxy für den Netzverkehr fungiert und somit wieder ein komplettes Intranet anonymisieren kann. Dieser Client verbindet sich beim Starten mit einem „blender“ genannten Server, der ihm die Mitgliedschaft in der Gruppe ermöglicht.

Ein dezentral angelegtes Verfahren verzichtet auf den Server und überläßt den bisherigen Gruppenmitgliedern die Entscheidung, ob ein neuer Nutzer der Gruppe beitreten darf. Dieser Ansatz fand allerdings aufgrund der niedrigen Performance keinen Einzug in die Beispielimplementierung [t02b], die in [ReRu98] erwähnt wird.

Jeder Teilnehmer hält eine Liste von Teilnehmern, von denen er annimmt, dass sie aktiv sind und zu denen er selber gehört. Ein in der Gruppe befindlicher Nutzer stellt eine Anfrage an einen Dienst im Internet, leitet diese jedoch nur mit einer gewissen Wahrscheinlichkeit ($> 50\%$) direkt an den betreffenden Server weiter. Ein Zufallsgenerator entscheidet, ob die Anfrage beim angefragten Server oder bei einem Gruppenmitglied landet, welches in diesem Fall seinerseits den Zufall entscheiden läßt, wohin es die Nachricht weiterleitet. Jeder Knoten merkt sich dabei, woher eine Nachricht kam und wohin er sie weitergeleitet hat. Auf diese Weise entsteht ein Pfad von Mitgliedern, über den die Nachricht ihren Weg zum Server nimmt, und über den die Antwort in umgekehrter Richtung an den Initiator der Anfrage zurückgeschickt wird.

Alle von demselben User initiierten Verbindungen folgen demselben Pfad bis zum letzten jondo in der Reihe. Dies geschieht, damit böswillige Teilnehmer nicht an mehreren unterschiedlichen Pfaden desselben Nutzers zu gleichen oder verschiedenen Zielen beteiligt sein können, aus denen sie durch Analyse der Inhaltsdaten auf die Zugehörigkeit zum selbern Nutzer schließen könnten. Durch die Verwendung mehrerer Pfade würde sich die Wahrscheinlichkeit, dass sich auf einem von ihnen ein böswilliger Teilnehmer befindet, steigen.

Ein Teilnehmer, der seine eigene oder eine empfangene Nachricht weiterleitet, versieht diese mit einer Pfad-ID, die für ihn eindeutig ist. Ein Teilnehmer, der eine Nachricht von einem anderen Teilnehmer erhält, ermittelt mit Hilfe einer gespeicherten Übersetzungstabelle seine eigene Pfad-ID für diese Verbindung. Mittels dieser ID kann er aus seiner Nachfolgertabelle den zugehörigen nächsten Teilnehmer ermitteln, an den die Nachricht weitergeleitet werden muss. Der Sinn hinter den unterschiedlichen IDs wird deutlich, wenn man in Betracht zieht, dass ein Teilnehmer mehrfach auf ein und demselben Pfad vorkommen kann, und daher die Verwendung von nur einer Pfad-ID zu einer Endlosschleife führen könnte.

Um einen Schutz vor außenstehenden Beobachtern zu gewährleisten, wird beim Aufbau eines Pfades vom Initiator ein Schlüssel für den jeweiligen Pfad erstellt. Dieser wird an die Nachricht zur Initialisierung des Pfades angehängt und dadurch an alle Stationen auf dem Pfad weitergeleitet. Bei Anfrage und Antwort verschlüsselt jeweils der erste Crowd-Teilnehmer des Pfades die Daten mit dem zugehörigen Schlüssel. Die restlichen Teilnehmer leiten die Nachricht unverändert weiter bis zur letzten jondo-Instanz auf dem Übertragungsweg, welche die Nachricht wieder entschlüsselt und (abhängig von der Richtung) an den angefragten Server oder an den anfragenden User übergibt.

Crowds bieten gegenüber dem Zielsystem einen perfekten Schutz der Anonymität, da jedes Mitglied der Gruppe mit der gleichen Wahrscheinlichkeit der Absender einer Nachricht sein könnte.

In Abbildung 5 [3] ist der Ablauf der Kommunikation mittels Crowds noch einmal dargestellt.

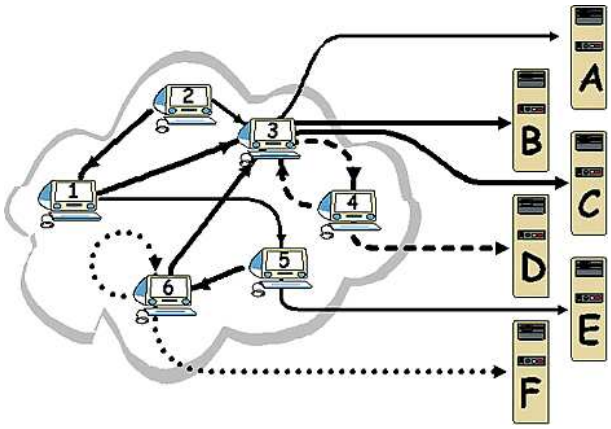


Abbildung 5: Crowds-Netzwerke in der Anwendung

6.2 Schutz vor Angriffen

Crowds sind von der Sicherheit und der Anfälligkeit gegen Angriffe mit Mixes vergleichbar. Im Gegensatz zu Mixes sind hier jedoch die Stationen, über die eine Nachricht weitergeleitet wird, selbst User, so dass kein Vertrauen in einen externen Anbieter gesetzt werden muss. Die Mitglieder der Gruppe müssen sich gegenseitig vertrauen, was aber in erster Linie den Inhalt der Daten betrifft, da auch die Nutzer nicht sagen können, von welcher Station eine Nachricht initiiert wurde.

Da kein dummy traffic verschickt wird, hängt die Sicherheit gegen Verkehrsanalyse stark von der Nutzerzahl und deren Aktivität ab. Nur so funktioniert der Grundgedanke von Crowds, die Aktivitäten eines Einzelnen in den Aktivitäten vieler Nutzer zu verbergen. Gegen timing attacks ist das System jedoch nicht gefeit, da es keine Verzögerungen und Verwürfelung der Weiterleitung von Daten auf den einzelnen Stationen gibt. Dies wurde der Performance zugunsten nicht realisiert.

Ebenso kann eine Nachricht aufgrund ihrer Größe und insbesondere ihrer Codierung auf dem Weg vom Sender zum Empfänger und umgekehrt verfolgt werden, wenn ein Angreifer in der Lage ist, sämtliche Nachrichten im Netz zu beobachten. Ein lokaler Beobachter (wie zum Beispiel ein Trojaner) kann nur die Nachrichten einsehen, die auf dem jeweiligen Rechner ein- und ausgehen.

Das System selbst bietet keinen Schutz gegen Flooding einzelner Rechner. Dies müßte in einzelnen Implementierungen bzw. in der Konfiguration des jeweiligen Rechners vorgenommen werden, so dass beispielsweise nur eine begrenzte Anzahl gleichzeitiger Verbindungen von ein und demselben Pfad akzeptiert werden.

Eine Anfälligkeit besitzt das System gegenüber der Kooperation von böswilligen Nutzern. Diese können in gegenseitiger Absprache den Initiator eines Pfades mit größerer Wahrscheinlichkeit eingrenzen. Diese Wahrscheinlichkeit hängt stark von der Anzahl der Nutzer im System und der Anzahl der zusammenarbeitenden Angreifer ab.

[3] Quelle: http://www.bsi.de/literat/anonym/anwmix.htm

6.3 Implementierungen

Konkrete Implementierungen existieren, mit Ausnahme der Beispielimplementierung [t02b], nicht.

7 Zusammenfassung und Fazit

Absolute Sicherheit kann keines der vorgestellten Verfahren bisher garantieren. Sicherheit wird bei allen mit geringerer Performance bezahlt, da zur Vermeidung der Verfolgung von Nachrichten diese schon allein durch das Weiterleiten über viele zusätzliche Zwischenstationen verzögert wird. Hinzu kommt bei Mixe-basierten Verfahren eine separate Verzögerung, die es einem Beobachter erschweren soll, die Nachricht anhand ihres Zeitverhaltens identifizieren zu können.

Die Frage, welches Verfahren für einen Benutzer geeignet ist, hängt stark von den persönlichen Bedürfnissen ab. Web-Anonymizer und anonymisierende Proxys bieten von den vorgestellten Verfahren den geringsten Schutz, aber zugleich die größte Performance. Besteht das Anliegen eines Nutzers darin, sich gegenüber dem Zielsystem seiner Anfrage unerkannt zu geben, sind die Anforderungen an die Sicherheit mit diesen Diensten erfüllt. Einen Schutz gegen Beobachter von außerhalb können sie nicht zusichern.

Mixes, und in leicht abgeschwächter Form Onion Routing, bieten einen recht hohen Schutz gegenüber den bekannten Angriffsarten, wenngleich auch hier absolute Anonymität nicht garantiert werden kann. Einzelne Implementierungen erweitern die ursprünglichen Konzepte mit eigenen Verfahren, so dass eventuelle Schwachstellen größtenteils ausgelöscht werden können (z.B. ein Ticket-System in JAP zur Vermeidung von flooding attacks). Aufgrund der hohen Verzögerung sind diese Verfahren jedoch nur bedingt für die meisten Aktivitäten geeignet, da mit immer günstiger werdenden Flatrate-Tarifen für DSL die Benutzer die bezahlte Geschwindigkeiten auch nutzen wollen. Mixe-basierte Verfahren sind daher am ehesten für den von Chaum erdachten Verwendungszweck (dem anonymisieren vo Emails) einzusetzen.

Crowds bieten ein Zwischenmaß zwischen der Sicherheit und der Performance von Mixes und Anonymizern. Durch die Verwendung der User-Rechner als Router entfällt die Notwendigkeit des Vertrauens in externe Anbieter. Die Kommunikationsbeziehung zwischen Sender und Empfänger einer Nachricht kann von einem Nutzer, über dessen Rechner die Nachricht geleitet wird, nicht erstellt werden und selbst böswillige, zusammenarbeitende Nutzer können diese nur mit einer gewissen Wahrscheinlichkeit erraten. Dennoch ist die Sicherheit im Vergleich zu Mixe-Verfahren niedriger, vor allem, da jeder Nutzer die Daten einsehen und darauß eventuell Informationen ziehen könnte.

Tabelle 1 wurde [BeFK] entnommen und stellt die Verfahren im Hinblick auf Schutz gegen die in der Einleitung vorgestellten Angriffsarten gegenüber.

	Message coding attack	Timing attack	Message volume attack	Inter-section attack	Floo-ding attack	Collusion attack
Anonymizer und anonyme Proxies	Geringer Schutz durch Verschlüsselung, kein Schutz ohne Verschlüsselung	Kein Schutz	Kein Schutz	Kein Schutz	Kein Schutz	Kein Schutz, Vertrauen in den Anbieter ist nötig
Crowds	Kompletter Schutz vor außenstehenden Beobachtern, hoher Schutz vor Teilnehmern der Gruppe	Kein Schutz	Kein Schutz	Kein Schutz	Kein Schutz	Hoher Schutz, aber nicht komplett zusicherbar
Mixes	Schutz durch assymetrische Kryptographie	Schutz durch Verwürfelung, schwächt die Performance	Schutz durch feste Nachrichtenlängen	Kein Schutz	Kein Schutz	Anonymität gesichert, solange mindestens ein Knoten vertrauenswürdig ist
Onion Routing	Schutz durch symmetrische und asymmetrische Kryptographie	Schutz nur zwischen den Onion Routern selbst	Schutz nur zwischen den Onion Routern selbst	Kein Schutz	Kein Schutz	Anonymität gesichert, solange mindestens ein Knoten vertrauenswürdig ist

Tabelle 1: Vergleich der Verfahren im Schutz gegen Angriffe

Als Fazit bleibt zu sagen, dass mit heutigen Techniken absolute Anonymität nicht zu realisieren ist. Die Nutzung der angeführten Verfahren erhöht die Sicherheit zur Wahrung der persönlichen Daten, kann sie aber niemals komplett gewährleisten. Die Vorsicht eines jeden Nutzers im Umgang mit seinen persönlichen Daten sollte mit oder ohne Verwendung eines Anonymisierungsdienstes immer an erster Stelle stehen, da aus diesen Informationen der meiste Nutzen für einen Beobachter gewonnen werden kann.

Literatur

[BeFK] Oliver Berthold, Hannes Federrath und Marit Köhntopp. “Project Anonymity and Unobservability in the Internet“.

[Chau81] David L. Chaum. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. *Communictaions of the ACM*, Februar 1981.

[DSCP02] Claudia Díaz, Stefaan Seys, Joris Claessens und Bart Preneel. Towards Measuring Anonymity. *Proceedings of Privacy Enhancing Technologies Workshop (PET 2002)*, April 2002.

[ReRu98] Michael K. Reiter und Aviel D. Rubin. Crowds: Anonymity for Web transactions. *ACM Transactions on Information and System Security* 1(1), 1998, S. 66–92.

[SeDa02] Andrei Serjantov und George Danezis. Towards an Information Theoretic Metric for Anonymity. *Proceedings of Privacy Enhancing Technologies Workshop (PET 2002)*, April 2002.

[t02a] AN.ON.

[t02b] Beispielimplementierung Crowds.

[t02c] Freedom.

[t02d] Mixmaster.

[t02e] Tor.

Abbildungsverzeichnis

Tabellenverzeichnis

Onion Routing

Konrad Miller

Kurzfassung

Dieser Seminarbeitrag stellt vor, wie mit Onion Routing auf Anwendungsschicht in TCP/IP Netzwerken - insbesondere dem Internet - ein Anonymisierungsdienst realisiert werden kann. Dabei sollen vor allem die Abläufe und Techniken beleuchtet werden, mithilfe derer anonyme Verbindungen hergestellt werden können. Protokollspezifika und Details werden bewusst ausgelassen und können in den angegebenen Quellen nachgeschlagen werden. Auf die Beschreibung der Funktionsweise folgt eine Bedrohungsanalyse und Bewertung von Onion Routing und die Vorstellung ähnlicher Systeme.

1 Einführung

In der Bundesrepublik Deutschland hat jeder Bürger das Recht auf Informationelle Selbstbestimmung. Diese wurde 1983 vom Bundesverfassungsgericht (Volkszählungsurteil) als Ausprägung des allgemeinen Persönlichkeitsrechts zu Artikel 1 (Menschenwürde) und Artikel 2 (Allgemeine Handlungsfreiheit; Freiheit der Person) des Grundgesetzes gehörig anerkannt [dBun99][dBun83]. Die Informationelle Selbstbestimmung umfasst das Recht, selbst darüber zu bestimmen, ob man personenbezogene Daten preisgeben möchte und wofür diese verwendet werden sollen. Ein Weg, um die Informationelle Selbstbestimmung, aber auch z.B. Presserecht und Meinungsfreiheit durchzusetzen, ist die Anonymisierung von Kommunikationswegen. So kann jeder selbst entscheiden, welche Informationen er über sich preisgibt.

Bei der Nutzung der Dienste des Internets mag schnell ein Gefühl der Anonymität aufkommen. Dieses Gefühl täuscht, denn falls keine besonderen Vorkehrungen getroffen werden, kann ein Nutzer mindestens anhand seiner IP-Adresse verfolgt werden. Hiermit kann nicht nur fest gestellt werden, wer mit wem kommuniziert, sondern es können, völlig ohne Wissen des Nutzers, Persönlichkeitsprofile erstellt werden. Weiterhin wird die Anonymität im Internet dadurch einschränkt, dass Kommunikation im Allgemeinen unverschlüsselt stattfindet, sie also leicht mitgehört werden kann (Sniffing/*Eavesdropping*). Im Datenstrom befinden sich oft Daten, welche die Kommunikationspartner identifizieren, wie zum Beispiel die Email-Adresse in einer Email.

Ein Anonymisierungsdienst sollte also Folgendes leisten:

- Anonymisierung (Trennung von Identifikation und Routing)
- Schutz der Privatsphäre (Verschlüsselung der Kommunikation)

Um gute Nutzbarkeit zu ermöglichen und weite Verbreitung zu erlangen, sollte der Dienst außerdem:

- Ohne Veränderung von bestehender Software funktionieren
- Resourceneffizient arbeiten (Netzwerkverkehr, CPU, RAM)

- Wenig Latenz verursachen
- Skalierbar sein
- Robust und ausfallsicher funktionieren
- Geringe Kosten für Inbetriebnahme und Betrieb verursachen

Auch ohne Veränderung der technischen Seite können einige dieser Punkte erfüllt werden. So könnte der Nutzer zum Beispiel öffentliche Netzzugänge benutzen. Außerdem könnte ein Nutzer mehr tun als er eigentlich möchte. Er könnte, anstatt eines einzigen Artikels, viele bestellen und nur den einen lesen. Diese Möglichkeiten sind meist entweder zeitaufwändig, unkomfortabel, teuer, oder eine beliebige Kombination davon. Es erscheint also wünschenswert, die Anonymisierung innerhalb des Netzwerks zu implementieren.

1.1 Gliederung

Es gibt verschiedene Systeme, die versuchen die oben genannten Anforderungen zu erfüllen. Hierzu gehören zum Beispiel Anonymizer, Crowds und Onion Routing. In diesem Beitrag soll hauptsächlich auf Onion Routing eingegangen werden. Nachdem in Abschnitt 2 die grundlegende Funktionsweise und Topologie von Onion Routing vorgestellt wurde, werden in Abschnitt 3 mögliche Schwächen dieses Systems aufgezeigt und bewertet. Darauf folgend werden weitere Herangehensweisen und daraus resultierende Techniken vorgestellt (Abschnitt 4), die ähnliche Ziele verfolgen. Abgeschlossen wird der Seminarbeitrag durch eine Zusammenfassung und Bewertung des Systems Onion Routing.

2 Onion Routing - Der Ansatz mit der Zwiebel

In diesem Abschnitt wird zuerst das Problem dargestellt, welches Onion Routing zu lösen versucht. Nachdem des Weiteren begründet wurde, warum Onion Routing auf Anwendungsschicht ansetzt, wird die zur weiteren Diskussion nötige Nomenklatur vorgestellt und eine grobe Übersicht über den Dienst gegeben. Im Folgenden werden die einzelnen Teile des Systems genauer betrachtet und anhand einer HTTP-Anfrage wird abschließend eine Verbindung durch das Netzwerk beschrieben.

2.1 Wo Onion Routing ansetzt/Das Problem, das zu lösen ist

Seine Herkunft in einem paketvermittelten Netz wie dem Internet zu verschleiern, ist nicht einfach. Es wird nämlich keine feste, bidirektionale Kommunikationsleitung aufgebaut, wie es bei leitungsvermittelten Netzen zum Beispiel dem Telefonnetz der Fall ist. Eine Verbindung zwischen zwei Geräten über das Internet ist eigentlich etwas, was nur für die Kommunikationsendpunke existiert (Schicht 4, z.B. TCP-Session). Der Einfachheit halber wird von nun an das die Verbindung initiierende Endgerät mit Alice und das Endgerät zu dem die Verbindung aufgebaut wird mit Bob bezeichnet. Die Paketvermittlung dieser Verbindung (Routing auf Schicht 3) ist unabhängig von dieser Verbindung — hier werden IP-Pakete einzeln geroutet. Die Schwierigkeit, die sich daraus ergibt, ist, dass Bob nicht einfach wie bei Telekommunikationsnetzen die von Alice aufgebaute Verbindung nutzen kann um zu antworten, sondern die Absenderadresse von Alice wissen muss (die IP-Adresse) um mit ihr bidirektional kommunizieren zu können. Auch wenn die IP-Adresse auf irgendeine Art und Weise verschleiert würde, könnte ein Beobachter immer noch das Paket anhand seines Inhalts verfolgen. Dieser bleibt

ja auf dem gesamten Weg gleich — und zwar unabhängig davon, ob das Paket verschlüsselt ist, oder nicht. Zudem könnte Bob auf eine solche Nachricht, ohne Zusatzinformationen, nicht antworten.

Die Eigenschaft, ein paketvermitteltes Netz zu sein und die damit verbundene Ende-zu-Ende Bedingung [Clar88] ist unerlässlich für die Skalierbarkeit des Internets. Bei Eingriff in die unteren Schichten des ISO/OSI Referenzmodells müssten die vorhandenen Netzwerkstrukturen angepasst werden. Für einen Anonymisierungsdienst ist es vorteilhaft auf Anwendungsschicht anzusetzen, weil dadurch:

- Anonymisierung optional — als Dienst — genutzt werden kann
- Neue Technologien schnell um-/eingesetzt werden können
- Die Komplexität des Internets in den unteren Schichten bleibt
- Bestehende Netzwerkstrukturen unverändert weiter genutzt werden können

2.2 Nomenklatur und allgemeiner Ablauf

Wird die Kommunikation mithilfe von Onion Routing anonymisiert, baut Alice nicht direkt eine Verbindung zu Bob auf, um mit ihm zu kommunizieren. Statt dessen baut sie die Verbindung über einen *Onion-Proxy* auf, der die Verbindung verschlüsselt über einen *Entry-Funnel* und mehrere *Core Onion-Router* (COR) zum *Exit-Funnel* leitet. Dieser Exit-Funnel baut dann, stellvertretend für Alice, eine unverschlüsselte Socket-Verbindung zu Bob auf (Abbildung 1). Es wird also eine *virtuelle Verbindung* durch das Netz zwischen Onion-Proxy und Exit Funnel aufgebaut, die bidirektional benutzt werden kann, aber deren Verlauf, wie wir im Folgenden sehen werden, nicht verfolgbar ist.

Der Onion-Proxy wählt die Route durch das *Onion-Routing-Netzwerk* und verteilt an jeden COR auf dem Weg, in einer sogenannten Zwiebel (*Onion*), einen symmetrischen Schlüssel. Mit diesen Schlüsseln wird der eigentliche Datenverkehr später verschlüsselt. Außerdem hat der Proxy die Aufgabe, die applikationsspezifischen Nachrichtenformate in ein generisches Format zu überführen, welches vom Entry-Funnel verstanden wird. Ein generisches Format ist deshalb notwendig, weil die Pakete der verschiedenen Datenströme — die durch einen COR verlaufen — mit verschiedenen symmetrischen Schlüsseln verschlüsselt werden, die einzelnen Pakete also den jeweiligen Datenströmen zugeordnet werden müssen. Das generische Format speichert dazu, pro COR, ein Paar: Label, Nachricht, wobei das Label die Zugehörigkeit zu einen Datenstrom kennzeichnet. Entry- und Exit-Funnel sind in der Regel CORs und sind topologisch der Ein- bzw. Ausgangspunkt des Onion-Routing-Netzwerks. Der Exit-Funnel dient als Proxy für die Rückrichtung, also für die Kommunikation von Bob zu Alice. Er hat in diese Richtung aber nur die Aufgabe, Bobs Nachrichten in das generische Format zu konvertieren. Auf die einzelnen Bestandteile des Systems und einen detaillierteren Ablauf wird im Folgenden kurz eingegangen, bevor in Abschnitt 2.3 mögliche Konfigurationen/Topologien behandelt werden.

2.2.1 Core Onion Router

Das Onion-Routing-Netzwerk ist ein Overlay-Netz, das auf einen unterliegenden zuverlässigen Dienst angewiesen ist. Im Falle des Internets stellt TCP [oSou81] diesen Dienst zur Verfügung. Die Core Onion Router sind durch langlebige TCP-Verbindungen (*Longstanding-Connections*, *Thick-Pipes*, *Links*) miteinander verbunden. Diese Verbindungen induzieren die

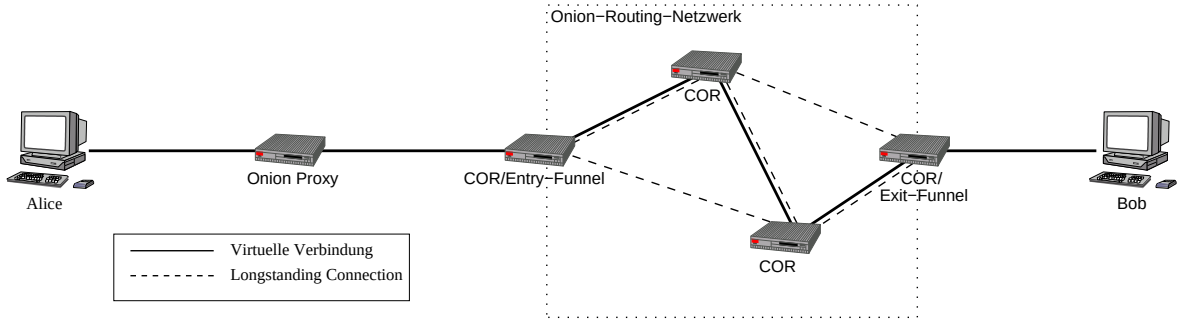


Abbildung 1: Grundlegender Aufbau von Onion Routing.

Topologie des Netzwerks und sind zusätzlich zur asymmetrischen Verschlüsselung der Onions und symmetrischen Verschlüsselung der Daten durch einen Stromchiffre verschlüsselt. Anonyme Verbindungen werden über die Longstanding Connections gemultiplext. Das heißt: baut ein Onion-Proxy eine Verbindung durch das Onion-Routing-Netzwerk auf, werden keine Verbindungen zwischen den CORs initiiert, sondern die schon bestehenden benutzt. Alle anonymen Verbindungen teilen sich also die schon bestehenden Thick-Pipes. Jeder COR besitzt ein Public-/Private-Key Paar. Der öffentliche Schlüssel ist allen anderen CORs — insbesondere den Entry-Funnels — bekannt.

Ein COR funktioniert ähnlich wie die 1981 von Chaum vorgestellten Mixe [Chau81]: Nach dem Annehmen verändert ein COR das Paket mit einer kryptographischen Funktion. Im Falle von Onion Routing ist dies die Verschlüsselung mit dem symmetrischen Schlüssel. Danach mixt er es mit anderen Paketen, verändert also die Reihenfolge der Pakete, die sich zur Zeit im Puffer befinden und leitet sie dann weiter. Der Hauptunterschied zwischen Mixen und CORs besteht darin, dass Mixe ein Paket unter Umständen sehr lange verzögern bis sie es weiterleiten. Sie warten, bis genügend Pakete zum Mixen zur Verfügung stehen. CORs leiten die Pakete möglichst schnell weiter und erzeugen gegebenenfalls künstlichen Netzwerkverkehr (*Dummy-Traffic*) zum Mixen. Wird eine Longstanding-Connection wenig genutzt, sollte ebenfalls Dummy-Traffic erzeugt werden (siehe Abschnitt 3).

Einzelne CORs können bei Bedarf auch redundant, als Router-Twins, ausgelegt werden. Router-Twins sind zwei CORs, die identisches Schlüsselmaterial verwenden, so dass ein beliebiger der beiden genutzt werden kann. Dies verbessert sowohl die Skalierbarkeit als auch die Ausfallsicherheit des Onion-Routing-Netzwerks.

2.2.2 Onion Proxy

Der Proxy kann in drei logische Ebenen unterteilt werden, die nacheinander durchlaufen werden (Abbildung 2). Zuerst kann der Datenstrom von einem anwendungsabhängigen (application-specific) *Privacy-Filter* daraufhin untersucht werden, ob er identifizierende Daten enthält. Diese werden dann gegebenenfalls gelöscht/verändert. Die Information über das Ziel der Nachricht bleibt in der, die Verbindung initialisierenden, Onion enthalten. Der Strom muss also keinerlei Informationen über den Adressaten enthalten. Da der Datenstrom bei Benutzung eines Anonymisierungsdienstes die einzige Möglichkeit für Bob darstellt, die Identität von Alice zu erkennen, ist es aber nicht immer wünschenswert den Strom von identifizierenden Informationen zu säubern. Das zu Beginn motivierte Ziel ist es schließlich, selber entscheiden zu können, wem man Informationen über sich preisgibt. Das Ziel ist nicht gegenüber jedem, jederzeit anonym zu sein. Oft ist es im Sinne von Alice, sich Bob gegenüber zu identifizieren und diese Identität sogar zu authentisieren. Der Datenstrom wird nur zwischen Alice und

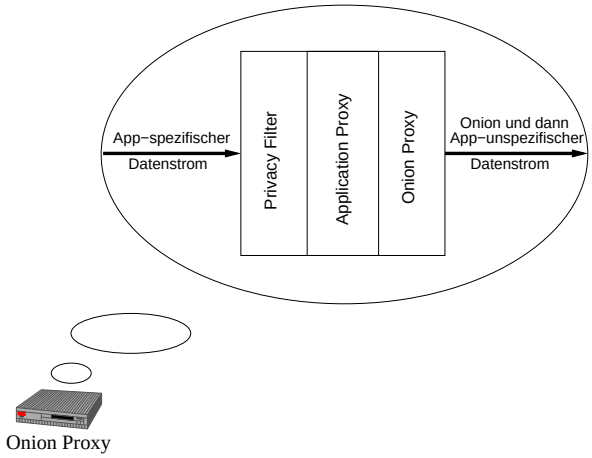


Abbildung 2: Die drei Ebenen des Onion-Proxy.

ihrem Proxy, sowie zwischen Exit-Funnel und Bob unverschlüsselt übertragen, daher wird bei einem nicht gesäubertem Strom (not sanatized) die Identität von Alice nur Bob offengelegt.

Hat der Datenstrom den Privacy-Filter durchlaufen, passiert er einen weiteren anwendungsabhängigen Teil des Proxies, den *Application-Proxy*. Dieser hat die Aufgabe den anwendungsabhängigen Datenstrom in eine anwendungsunabhängige Form zu bringen, die von den weiteren Knoten generisch, also unabhängig davon welche Anwendung anonymisiert wird, behandelt werden kann. Außerdem wird durch die Pakete im generischen Format eine Zuordnung zwischen Paket und Datenstrom ermöglicht. Dies ist für die Zuordnung zu dem symmetrischen Schlüssel und zum nächsten COR notwendig. Da alle Pakete im Onion-Routing-Netzwerk dieselbe Größe haben sollen ist es unter Umständen nötig, ein anwendungsabhängiges Paket auf mehrere generische Pakete zu verteilen, es also zu fragmentieren.

Die letzte Ebene wird *Onion-Proxy* genannt und ist das eigentliche Kernstück des Proxies. Hier wird die statische Route durch das Onion-Router-Netzwerk bestimmt und der Verbindungsaufbau über die CORs mithilfe der Onion initiiert. Dafür benötigt der Onion-Proxy Informationen über Topologie, Verbindungsstatus (Link-State), Zertifikate und öffentliche Schlüssel der CORs im Netzwerk. Diese Informationen werden automatisch beim Ein- und Austritt von CORs und Onion-Proxies in das beziehungsweise aus dem Onion-Routing-Netzwerk verteilt. Die sichere Verteilung dieser Informationen wird im Folgenden als gegeben angenommen. Aufgrund der Aufteilung in Application-Proxy, Onion-Proxy und Core-Onion Router ergibt sich:

- Alle Anwendungen die „normale“ Web-Proxy-Unterstützung bieten, können sofort und ohne Veränderung anonymisiert werden, denn für all diese Anwendungen muss nur ein einziger Application-Proxy geschrieben werden — einer der von dem Web-Proxy Format in das Onion-Router Format überführt

- Für Anwendungen, die keine native Web-Proxy-Unterstützung bieten, muss zuerst ein Application-Proxy geschrieben werden

- Onion Routing funktioniert sowohl für verbindungsorientierte (z.B. TCP), als auch für verbindungslose (z.B. UDP) Dienste. Verbindungslose Pakete werden allerdings genau wie verbindungsorientierte Pakete auch über die Longstanding Connections, also in TCP-Pakete eingepackt, verschickt

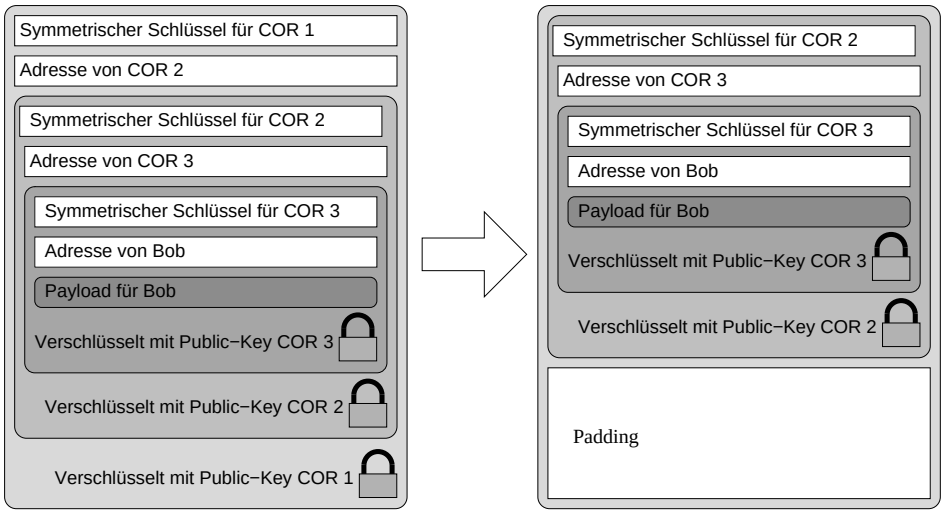


Abbildung 3: Das Abschälen einer Zwiebelschale

2.2.3 Onion

Die Onion ist eine verschachtelte Datenstruktur, die man sich, wie der Name vermuten lässt, wie eine Zwiebel vorstellen kann. Eine Schale repräsentiert die Verschlüsselung mit einem asymmetrischen Schlüssel. Schält man eine Schale ab (wird die Datenstruktur entschlüsselt), kommt darin die Adresse des nachfolgenden CORs, ein symmetrischer Schlüssel und eine kleinere Onion (ein weiteres Chiffrat) zum Vorschein, die wiederum mit dem öffentlichen Schlüsseln vom nachfolgenden COR verschlüsselt ist.

Auf dem Weg der Onion durch das Netz löst also jeder COR „seine“ Schale ab und speichert den darin gefundenen symmetrischen Schüssel zusammen mit dem, auch darin enthaltenen, nachfolgenden COR. Den Rest der Onion füllt er mit Füllbits auf (*Padding*), so dass sie eine feste Größe behält und schickt sie an den nachfolgenden COR weiter.

In Abbildung 3, soll eine Route über COR1, COR2 und COR3 zu Bob aufgebaut werden, über die dann, symmetrisch verschlüsselt, Daten fließen können.

Außer dem symmetrischen Schlüssel und der Adresse des folgenden CORs sind im Allgemeinen noch weitere Informationen in der Onion enthalten. Um Wiedereinspielungsangriffe zu verhindern, hat jede Route einen Zeitpunkt (*Expiration Time*), bis zu dem diese gültig ist. Bis zum Ende der Gültigkeit wird die Onion gespeichert und mit den später empfangenen Onions verglichen. Identische Onions werden in diesem Zeitraum verworfen und sind nach dessen Ablauf nicht mehr gültig. Außerdem muss der Datenstrom markiert werden, damit er seinem entsprechenden symmetrischen Schlüssel zugeordnet werden kann. Dies kann durch sogenannte Labels erfolgen. Die Labels sollten bei jedem COR unterschiedlich sein, um eine Verfolgung der Pakete anhand dieser, zum Beispiel durch kooperierende, bösartige CORs, zu verhindern. Es ist außerdem ratsam, statt nur einem symmetrischen Schlüssel, ein Tupel Verschlüsselungsalgorithmus/Schlüssel anzugeben. So ist der Algorithmus austauschbar und es können sogar für die verschiedenen Longstanding-Connections verschiedene Algorithmen gewählt werden.

Die Länge der Route ist auf den ersten Blick durch die Größe der Onion bestimmt, denn für jeden COR auf der Route ist eine Schale der Onion bestimmt und alle Datenpakete im Netzwerk — also auch die Onions — haben dieselbe Größe. Allerdings kann man Onions durch bestehende, anonyme Verbindungen tunneln, so dass die Routenlänge theoretisch unendlich lang werden kann. In der Onion sind nur diejenigen CORs angegebenen, die sich wie Mixe

verhalten. Es können noch andere CORs auf dem tatsächlichen Weg der Onion durch das Onion-Routing-Netzwerk liegen. Auf diese Weise können auch CORs erreicht werden, wenn dem Onion-Proxy die genaue Route zu ihnen nicht bekannt ist, oder der nächste COR auf der Route nicht direkt erreichbar ist. Diese Eigenschaft wird als *Loose-Source-Routing* bezeichnet. Damit die Anzahl der angesteuerten CORs nicht tatsächlich unendlich lang wird, sollte die maximale Anzahl Freier Hops (*Max-Loosecount*) zwischen je zwei eingetragenen CORs in der Onion mitgeführt werden. Entscheidet ein (in der Onion angegebener) COR auf dem Weg, die Onion über nicht angegebene CORs zu leiten, erstellt er eine neue Onion, mit maximal Max-Loosecount Schalen und der ursprünglichen Onion als Payload. Damit die neue Onion dieselbe Größe hat, wie alle anderen Pakete im Overlay, muss die innere Onion unter Umständen fragmentiert werden.

Zusammengefasst ergeben sich also die folgenden Eigenschaften:

- Jeder COR kennt jeweils nur seinen direkten Vorgänger und Nachfolger
- Jede Onion, im gesamten Overlay-Netz, hat dieselbe Größe
- Eine Onion sieht bei jedem COR anders aus — sowohl für die CORs als auch für Außenstehende, denn sie wird bei jedem COR kryptographisch mit je einem anderen asymmetrischen Schlüssel verändert
- Es wird für jede Verbindung, in jedem COR auf der Route, ein Zustand gehalten, der zumindest das Label, den End-Zeitpunkt der Verbindung, den symmetrischen Schlüssel, den nächsten und den vorherigen COR umfasst
- Alle Pakete haben dieselbe Größe — zu kleine Pakete werden aufgefüllt, zu große fragmentiert

2.2.4 Verbindungsaufbau, Datenaustausch, Verbindungstermination

Die Nutzung des Onion Routing läuft in drei Phasen ab: dem Verbindungsaufbau (Circuit Setup), Datenaustausch (Data Movement) und der Verbindungstermination (Circuit Tear-Down).

Angenommen Alice surfe im World Wide Web und Bob sei HTTP-Server. Alice klickt auf einen Hyperlink, woraufhin der Web-Browser ein HTTP-GET-Request an den Onion-Proxy, dessen Adresse im Web-Browser als HTTP-Proxy eingetragen ist, schickt. Hier wird von dem Privacy-Filter der Request von „GET www.bob.de/ HTTP/1.0“ auf „GET / HTTP/1.0“ umgesetzt. Die Information über das Ziel (www.bob.de) geht dadurch nicht verloren, denn sie ist weiterhin im IP-Header enthalten. Der veränderte Request wird daraufhin vom Application-Proxy in ein generisches Format gebracht. Hier beginnt der *Verbindungsaufbau*: Der Onion-Proxy wählt nun eine statische Route durch das Onion-Router-Netzwerk und erstellt eine Onion, mithilfe derer die Route und die zugehörigen symmetrischen Schlüssel den CORs bekannt gegeben werden. In der Onion ist das Ziel — www.bob.de — in der innersten Schicht, enthalten. Der Exit-Funnel kennt also, schon bevor die ersten Nutzdaten verschickt wurden, das Ziel des Nutzdatenstroms. Die Onion selbst wird bei jedem COR asymmetrisch entschlüsselt während der später stattfindende Datenaustausch zwischen Alice und Bob mit den, durch die Onion verteilten, symmetrischen Schlüsseln ver- und entschlüsselt wird. Das Circuit Setup ist an dieser Stelle abgeschlossen und der *Datenaustausch* kann beginnen. Der Onion-Proxy verschlüsselt den Payload, also den HTTP-Request in generischer Form, mit dem in der Onion verteilten symmetrischen Schlüssel auf die Art, in der auch schon die Onion verschlüsselt wurde (nur diesmal nicht mit dem asymmetrischen Schlüssel). Jeder COR auf der Route wird also dieses mal eine symmetrische Schale abpellen, das Datenpaket wieder

auf eine feste Größe auffüllen und an den nächsten COR (der durch die Onion ausgehandelt wurde) schicken. Entschlüsselt der Exit-Funnel sein Paket, erhält er den HTTP-Request im Klartext. Er übersetzt ihn wieder in das HTTP-spezifische Format und baut eine direkte Verbindung mit Bob auf, an den er den HTTP-Request schickt. Antwortet Bob auf die Anfrage so bringt der Exit-Funnel die Antwort in ein generisches Format. Dann durchläuft seine Antwort die Route in umgekehrter Reihenfolge. Diesmal wird dem Paket allerdings bei jedem COR eine Verschlüsselungsschicht hinzugefügt, nicht abgeschält. Der Onion-Proxy muss das erhaltene Paket also mit allen symmetrischen Schlüsseln der Route entschlüsseln, bevor er die Antwort an Alice übermitteln kann. Wie beim Verbindungsaufbau durch die Onion haben auch alle Datenpakete in der Datenaustausch-Phase identische Größe. Die dritte Phase, die *Verbindungstermination*, kann zu jeder Zeit und von beliebigen CORs eingeleitet werden. Insbesondere kann dies auch „aus der Mitte“ heraus geschehen. Dazu wird ein entsprechendes Steuerpaket in beide Richtungen versandt, mit der Anweisung die Verbindung zu terminieren. Diese muss von den CORs bis zu den Enden weitergeleitet werden. Ein Schutz vor einem Denial-of-Service Angriff, bei dem ein Angreifer einen COR betreibt und wilkürlich Verbindungen terminiert, findet nicht statt. Dieser Angriff stellt allerdings auch keine Gefahr für die Anonymität dar.

Bei einigen Netzwerkprotokollen ist es notwendig, dass Bob auch nach Verbindungstermination — also zum Beispiel am nächsten Tag — wieder eine Verbindung zu Alice herstellen kann. Bei asynchoner Kommunikation, wie Email, ist dies unerlässlich, aber nicht ohne weiteres möglich, weil ja Alice eine anonyme Verbindung zu Bob aufgebaut hat und Bob somit Alices Adresse unter Umständen nicht kennt. Hierfür wurde eine Erweiterung der Onions vorgeschlagen — die sogennannte *Reply-Onion*. Soll Bob nach Verbindungstermintation in der Lage sein zu antworten, so kann Alice beim Exit-Funnel eine Reply-Onion hinterlegen. In dieser speziellen Onion sind, wie in einer „normalen“ Onion, asymetrisch verschlüsselte Routinginformationen enthalten. Die CORs auf der Route werden allerdings in umgekehrter Reihenfolge gespeichert, so dass das Ziel, in der innersten Schale der Reply-Onion, Alice ist. CORs können eine Reply-Onion nicht von einer Onion unterscheiden. Genau wie bei „normalen“ Onions kann eine Reply-Onion nur ein einziges mal benutzt werden (um Replay-Angriffe zu verhindern). Soll Bob mehr als einmal erneut einen virtuelle Verbindung aufbauen können, müssen mehrere Reply-Onions hinterlegt werden.

2.3 Konfigurationen/Topologie

In [M. R00] werden verschiedene Topologie-Konfigurationen aufgezählt. Diese sollen hier kurz vorgestellt und bewertet werden. Es stellt sich hauptsächlich die Frage, welche Aufgaben von dem Benutzer/der Organisation, der/die Onion Routing nutzen möchte, übernommen werden und welche Aufgaben an fremde, potentiell böswillige administrative Domänen übergeben werden. Unabhängig von der Konfiguration sollte ein Onion-Proxy seine Route immer möglichst so konstruieren, dass sich die darin befindlichen CORs in verschiedenen administrativen Domänen befinden. So wird es einem Angreifer erschwert alle CORs auf einer Route zur Netzwerkverkehrsanalyse benutzen zu können. Onion Routing verschleiert die Verbindung (*Source-Destination-Linking*) zwischen Alice und Bob schon bei einem einzigen aufrichtigen Core Onion Router, ist also so stark wie das stärkste Glied der COR-Kette [Syve03].

2.3.1 Remote Proxy

Diese Konfiguration sieht vor, dass der Onion Proxy nicht lokal, also weder auf demselben Rechner, noch im selben lokalen Netzwerk, betrieben wird. Diese Situation ist denkbar schlecht, falls man dem Administrator, der den Onion Proxy betreibt nicht uneingeschränkt

vertraut. Ein bösartiger Administrator könnte die Daten unverschlüsselt weiterleiten, sie im schlimmsten Fall sogar selbst analysieren oder verändern (man in the middle). Ist die Verbindung zum Proxy nicht verschlüsselt, so ist diese bis zum Erreichen des Onion Proxies problemlos angreifbar. Eine Verschlüsselung der Nutzdaten hilft an dieser Stelle nicht, da an den Paketköpfen analysiert werden kann, mit wem kommuniziert wird. Ist allerdings der Administrator vertrauenswürdig, die Verbindung zwischen Alice und Remote Proxy getunnelt, der Tunnel verschlüsselt und kann der Proxy bei Verbindung seine Identität beispielsweise durch ein Zertifikat beweisen — wie es zum Beispiel bei einem SSH-Tunnel der Fall ist, so kann diese Konfiguration durchaus sinnvoll sein.

Es ergeben sich zwei große Vorteile dieser Konfiguration: zum einen kann Onion Routing genutzt werden, ohne Software bei Alice zu installieren, zum anderen gibt es keinen zusätzlichen Rechenaufwand bei Alice - alle durch Onion Routing entstehende Aufgaben werden vom Remote Proxy übernommen. Eine Verbindung, die den oben genannten Anforderungen genügt, kann zum Beispiel durch einen SSH-Tunnel zwischen Alice und dem Onion-Proxy realisiert werden. So aufgesetzt ist die Verbindung sowohl sicher gegen Abhören als auch gegen Verkehrsanalyse, allerdings nur bis zum Exit-Funnel. Die Verbindung von dort zu Bob ist in der Regel unverschlüsselt, falls er nicht selbst einen Exit-Funnel betreibt (siehe 2.3.3, 2.3.4, 2.3.5), dessen Verbindung zu Bob durch ein vertrauenswürdiges Netzwerk verläuft. Alice muss also, um anonym zu bleiben, entweder zusätzlich die Nutzdaten verschlüsseln (z.B. durch eine Ende-zu-Ende Verschlüsselung) oder den Datenstrom von sie identifizierenden Informationen säubern. Da dem Remote-Proxy in jedem Fall vertraut werden muss, kann diese Aufgabe auch, wie oben beschrieben, der Proxy übernehmen.

2.3.2 Customer-ISP

In dieser Konfiguration betreibt Alices Internet Service Provider (ISP) einen Funnel, während der Onion-Proxy auf Alices Rechner läuft. Alice muss ihrem ISP also nicht vertrauen, dass er die Routen korrekt konstruiert und somit ihre Kommunikation anonymisiert, sondern sie selbst hat die volle Kontrolle über diesen Teil des Systems. Der ISP auf der anderen Seite muss nur einen Funnel aufsetzen und nicht zwangsläufig selbst einen COR betreiben. Es wird in diesem Fall eine Longstanding-Connection zu einem entfernten COR aufgebaut, an den die Pakete übergeben werden. Betreibt er einen COR und leitet somit auch „fremden“ Netzwerkverkehr durch seinen COR, ist es unmöglich für einen außenstehenden Beobachter herauszubekommen, welcher Teil des Verkehrs im Netz des ISPs terminiert. Die Nachteile dieser Lösung sind das erhöhte Verkehrsaufkommen, sowie auch höhere Hardwareanforderungen an den Funnel/COR, die jeweils höhere Kosten verursachen. Betreibt der ISP keinen COR, liefert eine Verkehrsanalyse den Netzwerkverkehr, der im ISP-Netzwerk terminiert. Es ist aber nicht ersichtlich, an welche einzelnen Dienstnehmer er gerichtet ist. Folglich ist Alice nicht darauf angewiesen, dass der COR des ISP korrekt funktioniert, sollte aber darauf achten, die virtuelle Verbindung über mehrere administrative Domänen hinweg aufzubauen. Die oben genannte Schwachstelle der Verbindung zwischen Exit-Funnel und Bob existiert auch hier.

2.3.3 Island-Onto-Yourself

Hier betreibt Alice sowohl den Proxy, als auch einen Core Onion Router. Diese Konfiguration liefert den größtmöglichen Schutz unter allen besprochenen Konfigurationen, den Alice allerdings durch eine hohe Netzwerksverkehrs- und hohe Hardwarelast bezahlt. Außerdem muss Alice in der Lage sein beides zu installieren und sicher zu konfigurieren, was unter Umständen nicht einfach ist. Der Hauptvorteil dieser Lösung ist, dass eine Analyse nicht einmal ergibt, ob Alice überhaupt mit irgendjemandem kommuniziert, oder nicht. Auch der ISP von Alice kann

keine Aussage darüber treffen. Allerdings ist auch hier die Verbindung zwischen Exit-Funnel und Bob, wie oben beschrieben, angreifbar.

2.3.4 Proxy and Onion-Router at Firewall

Ein weiteres Beispiel, Onion Routing zu konfigurieren ist, sowohl Proxy als auch Onion-Router auf dem Rechner zu betreiben, der auch als Firewall dient. Es könnte der gesamte „direkte“ Netzwerkverkehr blockiert werden, so dass jeglicher Verkehr durch den Onion-Proxy verlaufen muss. Der Onion-Proxy dient also als eine Art Interface zwischen dem internen Netzwerk und der Außenwelt. Um zu verhindern, dass im internen Netzwerk terminierender Netzwerkverkehr identifiziert werden kann, sollte der COR bei der Firewall auch Verkehr anderer CORs routen. Damit dieser Ansatz funktioniert, muss das interne Netzwerk vertrauenswürdig sein. Wird diese Lösung auf beiden Seiten eingesetzt, zum Beispiel bei zwei Firmen, die miteinander kommunizieren, oder zwischen verschiedenen Zweigstellen einer Firma, kann dieser Ansatz sogar eine Ende-zu-Ende Verschlüsselung ersetzen. Zwischen den beiden Firewalls wird niemals Netzwerkverkehr unverschlüsselt übertragen. Wird sie allerdings nur einseitig eingesetzt besteht die unter Abschnitt 2.3.1 beschriebene Schwachstelle zwischen Exit-Funnel und Bob.

Gerade für Firmen ist dieser Ansatz interessant, zum einen weil der Verkehr — von außen beobachtet — anonym und verschlüsselt ist, zum anderen weil intern trotzdem Firmenpolicies durchgesetzt werden können (z.B. Blockieren von ebay.de während der Arbeitszeit, kein Filesharing) und der interne Netzwerkverkehr beobachtet (monitoring) werden kann. Ein weiterer Vorteil ist, dass auf den Clients keine zusätzliche Software eingerichtet werden muss, es muss lediglich der Proxy korrekt angegeben werden.

2.3.5 Local Proxy with Onion-Router at Firewall

Dieser Ansatz sieht auf den ersten Blick ähnlich aus wie der, bei dem Proxy und Onion-Router bei der Firewall laufen. Seine Eigenschaften unterscheiden sich jedoch in vielen Punkten. So läuft hier der Onion-Proxy auf den Clients im internen Netz. Daraus folgt nicht nur, dass die Clients einzeln konfiguriert und die Mitarbeiter unter Umständen geschult werden müssen. Da die Routenwahl nun dem Client obliegt, wird auch dem internen Netzwerk verborgen, mit wem einzelne Clients kommunizieren und welche Protokolle sie nutzen. Firmenpolicies können also praktisch nicht kontrolliert und durchgesetzt werden. Leitet der COR bei der Firewall auch externen Traffic ist diese Lösung allerdings genauso sicher wie die vorige. Auch hier wirkt die Firewall/der Onion Proxy als eine Art Interface, außerhalb dessen (mit Ausnahme der Verbindung zwischen Exit-Funnel und Bob, siehe Abschnitt 2.3.1) Netzwerkverkehr ausschließlich verschlüsselt und anonym übertragen wird.

Bei zwei gleichberechtigten Partnern (z.B. zwei Firmen) besteht bei dieser und bei der letzten Konfiguration ein politisches Problem: Eine Firma muss der anderen vertrauen, die Route korrekt aufzubauen. In [M. R96] wird eine Möglichkeit beschrieben, dieses Problem zu umgehen: beide Seiten können eine Verbindung zu einem designierten Onion-Proxy aufbauen, welcher die zwei Verbindungen miteinander „paart“. Man kann sich diese Paarung wie die Kommunikation über einen IRC Server vorstellen. Die verschiedenen Teilnehmer verbinden sich (anonym) mit dem Server, über den sie dann kommunizieren können.

3 Bedrohungsanalyse

Jedes sicherheitsrelevante Protokoll macht Annahmen über potentielle Angreifer und über den Zustand und Aufbau des Systems zum Zeitpunkt eines Angriffs. Anhand dieses Angrei-

fermodells (Adversary Model) kann dann abgewägt werden, welche Angriffe auf das System/Protokoll möglich sind und wie wahrscheinlich/schwierig diese Angriffe sind. An dieser Stelle wird zuerst ein Angreifermodell vorgestellt, anhand dessen dann die Durchführbarkeit üblicher Angriffe auf Anonymisierungsdienste bewertet wird.

3.1 Angreifermodell

Folgende Annahmen werden gemacht [M. R01]:

- Das Onion-Routing-Netzwerk ist vollvermascht
- Alle Longstanding Connections werden auf eine konstante Datenrate aufgefüllt (Padding)
- Jeder COR ist ein Exit-Funnel
- Die Route durch das Onion-Routing-Netzwerk ist zufällig gewählt
- Die Anzahl der CORs auf der Route ist zufällig zwischen $2 \leq n \leq \infty$ gewählt
- Die Island-Onto-Yourself-Konfiguration oder eine mit lokalem Proxy und entferntem COR wurde gewählt, d.h. Alice ist selbst Onion-Proxy

Die nachfolgenden Angriffspunkte sind in einem (Onion-Routing-) Netzwerk denkbar:

- Belauschen einer Verbindung
- Verzögern oder Stoppen des Datenstroms
- Aufbauen und Zerstören von Verbindungen sowie Erzeugen von Datenströmen (insbesondere auch DoS)
- Manipulation des Datenstroms

Alle genannten Punkte können von kompromittierten CORs aus durchgeführt werden. Ein von einem Angreifer kontrollierter COR kann beispielsweise den Datenstrom manipulieren. Daher müssen nur Angriffe betrachtet werden, die aus dem Kompromittieren und Kontrollieren von CORs folgen.

Es können allerdings auch mehrere CORs kompromittiert werden, die daraufhin kooperieren. Folgende Möglichkeiten werden in [M. R01] unterschieden:

- Single Adversary: Ein einzelner COR wird kontrolliert
- Multiple Adversary: Eine feste, zufällig verteilte Untermenge von CORs wird vom Angreifer kontrolliert
- Roving Adversary: Eine feste Anzahl von CORs wird jederzeit vom Angreifer kontrolliert, die kontrollierten CORs wechseln
- Global Adversary: Alle CORs werden vom Angreifer kontrolliert

Onion Routing bietet vor dem Global Adversary keinen Schutz. Sind alle CORs auf einer Route kompromittiert, ist diese nicht mehr anonym. Außerdem ist die Kommunikation offengelegt, wenn der Onion-Proxy, also mit oben genannten Annahmen Alice selbst, kompromittiert wurde. Wenn die Kommunikation zwischen Alice und Bob nicht zusätzlich Ende-zu-Ende verschlüsselt wurde, ist auch der Datenstrom aufgedeckt. Wird der Onion-Proxy nicht von einem Angreifer kontrolliert und ist mindestens ein COR ehrlich, reicht dies im Allgemeinen, um die Netzwerkverkehrsanalyse deutlich zu erschweren. Auf den Global Adversary soll deswegen nicht weiter eingegangen werden. Die beiden Klassen Single Adversary und Multiple Adversary sind eine Untermenge des Roving Adversary, deswegen wird nur der Roving Adversary betrachtet.

3.2 Angriffe auf Anonymisierungsdienste/Analysetechniken

Im Laufe der letzten Dekade wurden verschiedene Angriffe auf Anonymisierungsdienste vorgeschlagen. Das Review [R. S02] gibt hierzu einen guten Überblick. Nicht alle Angriffe funktionieren bei den Anonymisierungsdiensten gleichermaßen gut oder schlecht. Hier soll ein kurzer Überblick über die Angriffe und ihre Relevanz im Zusammenhang mit Onion Routing gegeben werden.

Die sogenannte *Nachrichten-Kodierungs-Attacke* ist die einfachste Möglichkeit, Netzwerkströme zu verfolgen. Nachrichten-Kodierung bezieht sich auf Informationen, die zum Beispiel in Netzwerk-Protokollköpfen (z.B. IP-Header) oder auch im Datagramm selbst enthalten sind. Es wird versucht, die durch Abhören ausgelesene Nachrichten-Kodierung auszuwerten und auf diese Weise Netzwerkströme mit ihren Benutzern zu korrelieren. Weiterhin können Nachrichten über verschiedene Hops am Inhalt verfolgt werden, um Versandtort und Ziel zu identifizieren. Dies funktioniert selbst bei verschlüsselten Datenpaketen, solange sich der Inhalt über die Zeit nicht ändert. Onion Routing schützt die Kommunikation sehr gut vor dieser Art der Netzwerksverkehrsanalyse. Es sehen nicht nur alle Pakete bei jedem COR unterschiedlich aus, sondern die Verbindungen zwischen den CORs sind außerdem durch einen Stromchiffre verschlüsselt, der die Pakete auf der Leitung wiederum anders aussehen lässt, als bei den CORs. Ein perfekter COR reicht in unserem Angreifermodell aus, um diesen Angriff so stark zu erschweren, dass er unmöglich scheint.

Der Angreifer kann außerdem versuchen, die Quantität des Netzwerkverkehrs, also die Anzahl der Pakete und deren Größe zu beobachten, um eventuelle Kommunikationsmuster ausfindig zu machen (*Packet Volume and Counting Attack/Communication Pattern Attack*). Ein Beispiel für ein solches Muster ist die Tatsache, dass in vielen Protokollen zu einer Zeit nur einer der beiden Kommunikationspartner Daten sendet. Die Annahme, dass alle Leitungen auf eine feste Datenrate aufgefüllt werden und alle Pakete dieselbe Größe haben, macht diesen Angriff unmöglich. Da ein Padding auf eine feste Datenrate nur theoretisch einfach zu realisieren ist, wären allerdings in der Realität Angriffe dieser Art möglich. Bursts in wenig benutzen Netzwerkabschnitten sind beispielsweise schwer auszuglätten.

Eine weitere Möglichkeit der Netzwerkverkehrsanalyse ist der *Timing Angriff*. Mehrere kooperierende Angreifer, deren Uhren sehr genau synchronisiert sein müssen, achten auf zeitliches Zusammentreffen von bestimmten Umständen. Dies können zum Beispiel das nahezu gleichzeitige Öffnen und Schließen von Sockets, beziehungsweise virtuellen Verbindungen, sein. Einem Angriff dieser Art kann nur schwer entgegengewirkt werden. Onion Routing verlässt sich darauf, dass es bei hinreichend starker Nutzung des Dienstes (bzw. starkem Padding) sehr schwierig ist zeitliche Umstände dieser Art zu korrellieren. In einer aktiven Variante dieses Angriffs, der *Message Delaying Attack*, wird versucht, das Finden dieser zeitlichen Aufälligkeiten durch Verzögern von Nachrichten zu erleichtern. Es ist beim Onion Routing kein Schutz

gegen einen Angriff dieser Art vorgesehen, weil eine starke Nutzung des Systems einen solchen Angriff hinreichend schwierig auszuführen macht.

Einige weitere aktive Angriffe wurden in der Vergangenheit vorgeschlagen. Beim *Message Tagging* versucht der Angreifer bestimmte Nachrichten an einem Punkt zu markieren, so dass ein kooperierender Angreifer an einem anderen Punkt diese Markierung erkennt. Auf diese Weise sollen Streckenverläufe erkannt werden und es könnte zum Beispiel eine Auswahl von (mit anderen Methoden gefundenen) potentiellen Pfaden auf ihre Richtigkeit überprüft werden. Meiner Meinung nach gibt es beim Onion Routing keine Möglichkeit diesen Angriff durchzuführen. Versucht ein Angreifer zwischen zwei CORs den Datenstrom zu verändern, so wird dieser aufgrund der verlorenen Stromchiffre-Synchronisation unwiderruflich zerstört. Am COR innerhalb der Datenpakete hinzugefügte Informationen werden von nachfolgenden (ehrlichen) CORs verschlüsselt und sind daher von einem kooperierenden Angreifer nicht mehr zu erkennen.

Reply Angriffe beim Verbindungsaufbau werden, wie bereits erwähnt, verhindert indem eine eingehende Onion mit alten, zeitlich noch gültigen, Onions verglichen wird. Doppelte Onions werden an dieser Stelle verworfen. „Natürliche“ Störungen dieser Art, also zum Beispiel Duplikate oder reihenfolgevertauschte Pakete, können nicht auftreten, da die gesamte Kommunikation (wie oben beschrieben) auf TCP-Verbindungen aufsetzt und TCP einen zuverlässigen Dienst bietet.

Schlecht synchronisierte Uhren sehen im Onion-Routing-Netz wie ein *Denial of Service* (*DoS*) Angriff aus. Bei einer vorgehenden Uhr in einem COR lässt dieser keine Verbindungen zu, beziehungsweise schließt bestehende Verbindungen vorzeitig. Bei Nachgehen der Uhr eines COR werden Onions bei diesem COR unnötig lang gespeichert, was zu unnötig belegten Kapazitäten führen kann.

Die oben ausgeführten Angriffe können natürlich auch kombiniert werden. Insbesondere kann ein gezielter DoS-Angriff, wie zum Beispiel das Schließen von virtuellen Verbindungen durch „normale“ Verbindungs-Abbau-Nachrichten, den Timing Angriff unterstützen. Ein gezieltes Ausschalten von einzelnen CORs wäre eine Möglichkeit, die Annahmen des Angreifermodells zu umgehen und normalerweise unmögliche Angriffe zu ermöglichen. Natürlich können geschickt plazierte, kompromittierte CORs einen großen Teil des Onion-Routing-Netzes dadurch lahmlegen, dass sie gerade initialisierte virtuelle Verbindungen sofort wieder zerstören. Hierdurch ist jedoch die Anonymität nicht gefährdet.

4 Related work

Es gibt derzeit drei populäre Ansätze zur Anonymisierung von Netzwerkverkehr: Proxies, Mix-basierte Systeme und Peer-to-Peer-Netze. Im Folgenden werden deren Funktionsweise, Annahmen und Ziele kurz vorgestellt. Außerdem werden jeweils einige Vertreter genannt.

4.1 Proxies

Proxies wurden ursprünglich dazu entwickelt, um Netzwerkverkehr zu cachen und ihn somit für den Endnutzer mit höherer Datenrate verfügbar zu machen, weil der primäre Server entlastet wird [I. C01]. Sie können aber zum Beispiel auch als Privacy-Filter benutzt werden. Baut Alice eine Verbindung zu einem Webserver Bob nicht direkt auf, sondern über einen Proxy, so kann Bob Alice nicht anhand ihrer IP-Adresse identifizieren. Er „sieht“ anstattdessen die IP-Adresse des Proxies. Filtert jetzt der Proxy zusätzlich alle identifizierenden Daten aus dem Netzwerkstrom, bleibt Alice Bob gegenüber anonym. Die Nutzung eines solchen

Proxies schützt allerdings in keinster Weise vor Netzwerkverkehrsanalyse, denn selbst wenn der Datenstrom verschlüsselt sein sollte, kann ein Eavesdropper an den entsprechenden IP-Kopf-Feldern erkennen mit wem Alice kommuniziert. Sie muss dazu nur Alices Datenstrom abhören bevor er den Proxy passiert. Dem Proxy selbst stehen alle Verbindungsinformationen zur Verfügung, so dass sein Administrator vertrauenswürdig sein muss. Ein Beispiel für einen weit verbreiteten, anonymisierenden Proxy ist *Anonymizer*. Anonymizer bietet sowohl ein Webinterface zum anonym Surfen an, als auch einen Client zum Download.

4.2 Mix-basiert

Das oben beschriebene Onion Routing ist ein Beispiel für einen Mix-basierenden Anonymisierungsdienst. Viele weitere, ähnlich funktionierende Prototypen wurden entwickelt. In *Mix-basierende Remailer* werden Emails als innerste Schicht in Onion-ähnlichen Datenstrukturen verschickt. Geantwortet wird mit einem den Reply-Onions ähnlichem Konzept. Auch *Pipe-Net* funktioniert ähnlich wie Onion Routing, wobei Pipe-Nets Angreifermodell noch misstrauischer ist, als das des Onion Routing. Pipe-Net geht davon aus, dass alle Mixe kompromittiert sein könnten - es also einen globalen Beobachter geben kann. Verbindungen sind hier permanent und übertragen jederzeit eine konstante Menge Netzwerkverkehr. Dieser extreme Ansatz hat sich als nicht praktikabel für größere Netzwerke herausgestellt. *Penet* ist ein Email Anonymisierungsdienst und verzichtet komplett auf Verschlüsselung. Jeder Knoten im Penet löscht den Email-Kopf, in dem identifizierende Daten stehen, und ersetzt ihn durch einen eigenen, bevor er die Email weiterleitet. Penet unterstützt Pseudonyme, die es ermöglichen auf solche Emails zu antworten. Wird ein Penet-Knoten kompromittiert bleibt der Absender einer Email, die den Pseudonym-Dienst nutzt, allerdings nicht anonym — hier ist das Mapping zwischen Pseudonym und Email-Adresse gespeichert.

4.3 Peer-2-Peer-Netz

Reiter und Rubin stellten 1999 sogenannte *Crowds* vor um Webbrowsing zu anonymisieren [M. R99]. Die Idee ist hierbei, in der Masse unterzutauchen („blending into a crowd“). Alle Crowd-Nutzer sind Mitglied in einem großen Peer-to-Peer-Netz. Möchte ein Nutzer eine Website aufrufen, so wird die Verbindung zuerst über eine zufällige Teilmenge anderer Crowd-Mitglieder geleitet, bis eines der Mitglieder die direkte Verbindung zum Webserver herstellt. Die Kommunikation zwischen Crowd-Mitgliedern findet dabei verschlüsselt statt. Auf Kryptographie wie sie bei Onion Routing Anwendung findet (gelayert), wird verzichtet. Crowds sind zum Beispiel über die Länge der versandten Nachrichten angreifbar. Einen ähnlichen Ansatz wie Crowds verfolgen *Tarzan* und *GNUnet*. Teilnehmer unterhalten verschlüsselte Verbindungen zu anderen Teilehmern des P2P-Netzes und über diese Verbindungen werden die Daten auf zufälligen Routen gesandt. Die Sicherheitsannahme besteht darin, dass jeder Teilnehmer abstreiten kann der Urheber einer Nachricht zu sein.

5 Zusammenfassung und Bewertung

Es gibt verschiedene Anonymisierungsdienste, deren Ziel es ist, Netzwerkverkehrsanalyse zu erschweren und somit eine anonyme Kommunikation durch feindliche Netzwerke zu ermöglichen. Eine Hauptaufgabe eines solchen Dienstes ist die Trennung von Identifikation und Routing. Onion Routing ist ein möglicher Ansatz dies zu realisieren. Wird Onion Routing zur Anonymisierung genutzt, baut der Dienstnehmer keine direkte Socket-Verbindung zu seinem Kommunikationspartner auf. Stattdessen kommuniziert der Dienstnehmer indirekt — über

verschiedene Core-Onion-Router mit dem Kommunikationspartner. Beim Verbindungsaufbau wird mithilfe einer verschachtelt asymmetrisch verschlüsselten Onion der Kontext einer anonymen Verbindung aufgebaut. In der Datenaustauschphase wird der eigentliche Inhalt der Kommunikation symmetrisch verschlüsselt übertragen. Es wird beim Onion Routing lediglich eine anonyme Verbindung aufgebaut — es findet nicht zwingendermaßen eine anonyme Kommunikation statt. Dazu müsste der Datenstrom zusätzlich verschlüsselt werden. Es sind verschiedene Angriffe auf Onion Routing denkbar. Um das System einfacher analysieren zu können, wurde ein Angreifermodell — also bestimmte Bedingungen an System und Angreifer — eingeführt. Werden alle Anforderungen an das System berücksichtigt, so ist der Erfolg der betrachteten Angriffe sehr unwahrscheinlich — ihre Durchführung wird also sehr stark erschwert. Die Voraussetzungen der Bedrohungsanalyse werden allerdings von realen Implementierungen, zumindest teilweise, nicht erfüllt. So werden die Longstanding-Connections in der Regel nicht zur maximalen Datenrate aufgefüllt. Trotzdem ist Onion Routing sehr resistent gegen Verkehrsanalyse. In der Praxis zeigt sich, dass ein Onion Routing-basiertes System zwar einfach aufzusetzen und zu benutzen ist, allerdings für die tägliche Benutzung unbrauchbar ist. Für Netzwerkprogramme, die viele Verbindungen aufbauen und wenige Daten übertragen ist Onion Routing ungeeignet, da ein Großteil des Overheads durch die asymmetrische Kryptographie im Verbindungsaufbau erzeugt wird. Aber auch für Programme mit wenigen, langlebigen Verbindungen ist Onion Routing nicht massentauglich — es skaliert durch die benötigte Zustandhaltung in den CORs nicht besonders gut, so dass hier die erreichten Datenraten sehr niedrig sind. Onion Routing ist zum jetztigen Zeitpunkt nur brauchbar, wenn der Nutzer zwingend auf Anonymität angewiesen ist.

Literatur

[Chau81] D. Chaum. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. *Communications of the ACM* 24(2), Februar 1981, S. 84–88.

[Clar88] D. Clark. The Design Philosophy of the DARPA Internet Protocols. *Proc SIGCOMM*, September 1988.

[dBun83] Erster Senat des Bundesverfassungsgerichts. Urteil zum Volkszählungsgesetz, BGBl I S. 369, Dezember 1983.

[dBun99] Der Parlamentarische Rat der Bundesrepublik Deutschland. *Grundgesetz für die Bundesrepublik Deutschland*. Bundeszentrale für politische Bildung. 1999.

[I. C01] G. Tomlinson I. Cooper, I. Melve. Internet Web Replication and Caching Taxonomy. RFC 3040 (Informational), Januar 2001.

[M. R96] D. Goldschlag M. Reed, P. Syverson. Proxies For Anonymous Routing. *Proceedings of the 12th ACSAC*, 1996, S. 95ff.

[M. R99] A. Rubin M. Reiter. Anonymous Web Transactions with Crowds. *Communications of the ACM* 42(2), 1999, S. 32–48.

[M. R00] D. Goldschlag M. Reed, P. Syverson. Onion Routing Access Configurations. *DARPA Information Survivability Conference and Exposition* Band 1, Januar 2000, S. 34–40.

[M. R01] G. Tsudik und C. Landwehr M. Reed, P. Syverson. Towards an Analysis of Onion Routing Security. *Designing Privacy Enhancing Technologies*, 2001, S. 96–114.

[oSou81] Information Sciences Institute University of Southern California. Transmission Control Protocol. RFC 793, September 1981.

[R. S02] L. Korba R. Song. Review of Network-Based Approaches for Privacy. *Proceedings of the 14th Annual Canadian Information Technology Security Symposium, Ottawa, Ontario, Canada*, Mai 2002, S. 13–17.

[Syve03] P. Syverson. Onion Routing for Resistance to Traffic Analysis. *DARPA Information Survivability Conference and Exposition* Band 2, 2003, S. 108ff.

Abbildungsverzeichnis

Tor und JAP: Umsetzung von Anonymitätstechniken

Mario Strefler

Kurzfassung

Im Folgenden wird die Umsetzung von Anonymitätstechniken anhand der im Internet tatsächlich eingesetzten Anonymitätsdienste Tor und JAP beschrieben. Nach dem Vorstellen der Entwurfsziele, der Funktionsweise und der Protokolldetails werden mögliche Angriffe auf die beiden Protokolle betrachtet. Abschließend werden rechtliche Probleme beim Einsatz von Anonymisierern in Deutschland erläutert und die Vor-und Nachteile beider Dienste verglichen.

1 Einleitung

1.1 Motivation

Im Folgenden sollen Möglichkeiten der Anonymisierung von Internet-Verbindungen vorgestellt werden. Anonymität bedeutet Schutz der Identität und kann aus verschiedenen Gründen gewünscht werden. Nicht nur Journalisten und ihre Informanten möchten gerne anonym bleiben, auch Sicherheitsbehörden möchten im Internet beispielsweise Internetforen verdeckt beobachten können. Patienten würden gerne im Internet Informationen zu einer Krankheit einholen oder an Selbsthilfegruppen teilnehmen, ohne dabei diese persönlichen Informationen der Öffentlichkeit preiszugeben. Neben diesen legitimen und legalen Interessen wie Meinungsfreiheit, Pressefreiheit, dem Schutz des Rechts auf informationelle Selbstbestimmung oder Strafverfolgungsinteressen gibt es auch Menschen, die Anonymität wünschen weil sie illegale Absichten hegen. Dadurch wird ein Gegensatz zwischen dem Recht auf Anonymität und der Verhinderung bzw. Aufklärung von Verbrechen erzeugt. Diese Problematik tritt auch bei der Umsetzung von Verschlüsselungssystemen auf und ist ein tiefergehendes Problem das den Rahmen dieser Arbeit sprengen würde. Deshalb soll es hier nicht weiter behandelt werden.

Es sollen nun Aspekte der Anonymität im Internet betrachtet werden. Kommunikation über paketvermittelte Netzwerke ist in der Regel nicht anonym, da jeder Teilnehmer eine netzweit eindeutige Adresse besitzt, die zur Kommunikation benutzt wird. Selbst wenn der Inhalt verschlüsselt ist, muss die Adresse lesbar sein, da sonst das Paket nicht zugestellt werden kann.

Im heutigen Internet wird ein gewisser Grad an Anonymität dadurch erreicht, dass Benutzer typischerweise Mitglieder kleinerer Netze sind, die durch einen Gateway mit dem Internet verbunden sind. Dadurch benutzen aus der Sicht eines Knotens im Internet alle Mitglieder eines kleineren Netzes die Adresse ihres Gateways; individuelle Kommunikation wird somit in der Masse versteckt. Diese Art der Anonymität ist jedoch nicht sehr sicher, denn Pakete werden von einem Gateway nur weitergeleitet. Selbst wenn die Datenpakete verschlüselt sind und es damit nicht möglich ist, aus den Nutzdaten Rückschlüsse auf den Sender zu ziehen, kann ein Angreifer, wenn er den Gateway kontrolliert oder Netzwerkverkehr im lokalen Netz oder im Internet belauschen kann, die ein- und ausgehenden Paketströme korrellieren und somit einzelnen Nutzern zuordnen.

Aus demselben Grund bieten auch anonyme Proxies keinen besonders starken Schutz. Desweiteren stehen jedem Benutzer nur eine begrenzte Anzahl von Gateways zur Verfügung, die alle zur selben Domain gehören. Dies ermöglicht das Erstellen von Profilen anhand der im Internet sichtbaren Gateway-Adresse.

Die Anonymisierung einer Kommunikationsbeziehung umfasst mehrere Maßnahmen. In den übermittelten Daten selbst können Informationen enthalten sein, die Rückschlüsse auf die Identität des Senders oder Empfängers zulassen, daher sollte die komplette Kommunikation verschlüsselt ablaufen. Ende-zu-Ende-Verschlüsselung muss aber zwischen den Endpunkten vereinbart werden und kann nicht Teil eines Anonymisierungsdienstes sein, denn Ende-zu-Ende Verbindungen stellt der Anonymisierungsdienst nur innerhalb des zu diesem Dienst gehörigen Netzwerkes bereit. Nicht nur die Daten, auch die verwendeten Protokolle können Informationen enthalten, die dazu beitragen können, die Identität des Kommunikationspartners zu enthüllen. Beispielsweise werden in HTTP-Nachrichten [FGMF+99] oft das Betriebssystem, der Webbrowser und die vorher besuchten Seiten eines Benutzers mit übertragen. Cookies können ebenfalls Informationen enthalten, mit Hilfe derer man Benutzer wiedererkennen kann. Ein weiteres Problem sind Seiteneffekte wie das Nutzen von nicht anonymisierten Protokollen durch anonymisierte Protokolle. Das Aufrufen einer Seite über einen anonymisierten Webbrowser kann beispielsweise eine nicht anonymisierte Namensauflösung bei einem DNS-Server auslösen, wenn der Browser die eingegebene alphanumerische Adresse selbst bei einem DNS-Server in eine IP-Adresse auflösen lässt.

Gewünscht ist also Anonymität gegen einen möglichst starken Angreifer, die selbst dann noch gewährleistet ist, wenn der Angreifer alle Knoten im Netz kontrolliert. Es wird klar werden, dass dies nicht möglich ist. Sicherheit kann aber gewährleistet werden, wenn mindestens einer der an der Anonymisierung beteiligten Knoten nicht kompromittiert ist.

1.2 Related Work

Neben dem noch zu behandelnden Onion Routing [SyGR97] gibt es noch weitere Ansätze, um im Internet Anonymität zu erreichen. Es gibt spezielle Ansätze, die nur bestimmte Dienste ano-nymisieren: Anonyme Remailer (Mixmaster [MCPS03], Mixminion [DaDM02]) ermöglichen das anonyme Versenden von Email, indem jeder Remailer nur den (verschlüsselten) Text der Nachricht, nicht aber den Header weitersendet. Werden Nachrichten über mehrere Remailer geschickt, kennt jeder Remailer nur seinen Vorgänger und seinen Nachfolger in der Kette. Anonyme Peer-to-Peer Publishing Netzwerke (Freenet [CSWH00]) ermöglichen das verteilte, anonyme Speichern von Daten. Bei Freenet ist geplant, Anfragen vor der Verarbeitung durch ein Onion-Netzwerk zu leiten, um sie zu anonymisieren. Tarzan [FrMo02] ist ein Peer-to-Peer Netz, das sich noch im Entwicklungsstadium befindet und Anonymisierung auf der IP-Schicht erreichen will.

1.3 Gliederung

In dieser Arbeit sollen nun zwei Ansätze vorgestellt werden, die den Anspruch haben, die oben erwähnte Forderung nach Sicherheit auch gegen starke Angreifer erfüllen zu können und die auch beide in der Praxis eingesetzt werden. Tor [DiMS04] ist ein Overlay-Netz, das eine Weiterentwicklung des Onion-Routing Protokolls darstellt. In Abschnitt 2 wird Tor vorgestellt. Zuerst wird Tor mit Onion Routing verglichen und die Entscheidungen beim Entwurf von Tor betrachtet. Tor ermöglicht sowohl die Anonymisierung von Benutzern als auch von Diensteanbietern; die Protokolldetails für beide Möglichkeiten werden behandelt, um abschließend mögliche Angriffe und Abwehrmaßnamhen zu betrachten. Abschnitt 3 behandelt JAP [Kö04], ein Projekt der Universität Dresden, das auf sogenannte Mix-Kaskaden setzt. Hier

kann der Bentzer nur zwischen mehreren fest vorgegebenen Mix-Folgen wählen. Wieder werden Funktionsweise und Protokolldetails betrachtet, einige Angriffe besprochen und zusätzlich noch rechtliche Probleme erörtert, die beim Betrieb von Anonymisierungsdiensten entstehen. Abschliessend werden in Abschnitt 4 Tor und JAP verglichen und bewertet.

2 Tor

Im nächsten Abschnitt wird eine kurze Einführung in Onion Routing gegeben, danach werden in Abschnitt 2.2 die Entwurfsentscheidungen für Tor erläutert und in 2.3 und 2.4 das Protokoll für die Anonymisierung von Verbindungen und von Diensten vorgestellt. In 2.5 werden das Angreifermodell und mögliche Angriffe erörtert.

2.1 Onion Routing

Tor ist eine Weiterentwicklung des Onion Routing Protokolls. Onion Routing hat diesen Namen, weil es ein Application Layer Routing Protokoll ist, bei dem alle Daten die über das Netzwerk verschickt werden, mehrfach verschlüsselt sind. Für jeden Router auf dem Weg werden die Daten mit dessen Schlüssel verschlüsselt; diese Verschlüsselungschichten liegen wie die Schalen einer Zwiebel (*Onion*) übereinander. Ein Onion Routing-Netzwerk besteht aus mehreren über verschlüsselte TCP-Verbindungen vollvermaschten *Core Onion Routern* (COR), die Daten empfangen, entschlüsseln und weiterleiten. Durch das Entschlüsseln sind die Daten, die den COR verlassen, nicht mehr den eingehenden Verbindungen zuzuordnen.

2.2 Entscheidungen bei der Entwicklung von Tor

2.2.1 Ziele

Außer den technischen Anforderungen an das Tor-Protokoll gibt es noch weitere; diese sollen nun besprochen werden. Da Tor versucht, die Verbindungen eines Benutzers in der Masse aller Verbindungen zu verstecken, ist die Anzahl der Benutzer ein wichtiger Faktor für den Grad an Anonymität, der gewährleistet werden kann. Je mehr Verbindungen an einem Knoten ein- und ausgehen, desto schwieriger ist es für einen Angreifer, sie zu korrellieren. Moderate Systemanforderungen und einfache Bedienbarkeit sind daher wichtig für die Sicherheit des Systems. Tor sollte keine Änderungen an bereits installierten Programmen benötigen, auf möglichst vielen Betriebssystemen laufen, keine allzu hohen Latenzen verursachen und muss mit der Bandbreite auskommen, die ein Benutzer zur Verfügung stellen kann. Ein einfaches Design erleichtert das Führen formaler Sicherheitsbeweise, ein überschaubares Programm erleichtert das Finden von Sicherheitslücken und erhöht die Stabilität.

Im Zuge der Vereinfachung wurden einige Eigenschaften nicht berücksichtigt. Tor hat zum Beispiel keine echte Peer-to-Peer Umgebung, denn Peer-to-Peer Systeme weisen einige noch ungelöste Probleme auf. Durch unterschiedliche Verzögerungen auf den einzelnen Kanälen können unterschiedliche Sichten des Netzwerkes entstehen, die ein Angreifer zu seinem Vorteil nutzen kann. Wird zum Beispiel ein OR A neu in das Netz aufgenommen, so wissen am Anfang nur wenige Knoten davon. Baut nun ein OP einen Circuit durch A, so gehört er zu mit Sicherheit zu der kleinen Menge an Knoten, die die Information über A bereits haben. Gegen Ende-zu-Ende Timing Angriffe ist Tor nicht vollständig sicher, da dazu sehr hoher Aufwand nötig ist. Näheres dazu in 2.5.2.

Tor stellt eine anonyme Verbindung bereit, sichert aber keine Protokolle ab. Es werden andere Programme benötigt, um benutzerspezifische Informationen aus den höheren Protokollen zu

entfernen. Diese Modularisierung des Entwurfs ist eine Folge des Designzieles der Einfachheit: Protokollspezifische Funktionen für jedes mögliche Anwendungsprotokoll in Tor zu integrieren ist zu aufwändig; das führt dazu, dass Protokolle, die nicht so häufig benutzt werden und daher von Tor nicht anonymisiert werden, nicht mit Tor zusammenarbeiten könnten. Dank der Trennung von Verbindungsanonymität und Datenanonymität können beliebige Programme ihren eigenen Protokollanonymisierer nutzen und ihre Verbindung über Tor aufbauen. Außerdem wird die Tatsache, dass ein Benutzer eine Verbindung zu einem Tor-Netzwerk unterhält, nicht verborgen.

2.2.2 Neuerungen im Vergleich zu früherem Onion Routing

Der alte Onion Routing Entwurf wies einige Schwächen auf; einige von ihnen versucht Tor zu beheben. Dafür wurden einige Neuerungen in den Entwurf eingebracht:

1. Perfect forward secrecy: Beim ursprünglichen Onion-Entwurf konnten Angreifer Pakete speichern und später die OR auf dem Weg nacheinander zwingen, sie zu entschlüsseln. Bei Tor gibt es einen neuen Algorithmus, um Pfade im Netz zu erzeugen. Dabei werden Session Keys ausgehandelt, so dass nach dem Löschen der Session Keys ein entschlüsseln der Pakete unmöglich ist.

2. Standardisierte Schnittstelle: Durch die Trennung der Anonymisierung von Verbindung und transportierten Daten kann Tor eine einheitliche Schnittstelle (SOCKS) verwenden. Onion Routing versuchte für jedes Programm eine eigene, die Protokolldaten anonymisierende Schnittstelle anzubieten und konnte dadurch weniger Programme unterstützen.

3. Keine Verkehrsformung: Onion Routing setzte Umordnung, Padding und Verkehrsformung ein, um einen besseren Schutz zu erreichen. Neuere Ergebnisse und Erfahrungen [BaGS01] zeigen, dass der erreichte Schutz den dafür nötigen Aufwand an Ressourcen, sowohl an Zeit als auch an Verkehr, möglicherweise nicht rechtfertigen. Bis ein beweisbar sicheres und effizientes Verfahren gefunden ist um diese Techniken einzusetzen, werden sie im Interesse der Einfachheit und der Ressourcenschonung nicht implementiert.

4. Mehrere TCP-Verbindungen teilen sich einen Circuit: Onion Routing öffnete für jede TCP-Verbindung einen neuen Circuit; dies zieht jedesmal das Aushandeln eines Schlüsselpaares nach sich und kostet damit Zeit. Mehrere Verbindungen über einen Circuit zu multiplexen erhöht die Effizienz des Protokolls.

5. Neue Circuit-Topologie: Verkehrsströme müssen nicht den ganzen Circuit durchlaufen, sondern können an jedem beliebigen Knoten auf dem Weg das Netzwerk verlassen.

6. Staukontrolle: Bei Onion Routing gab es keinen Mechanismus zur Staukontrolle. Tor benutzt Ende-zu-Ende ACKs zur dezentralen Staukontrolle unter Wahrung der Anonymität.

7. Directory Server: Onion Routing flutete Informationen über den Zustand des Netzes, wie die Adressen teilnehmender OR oder den Zustand der Verbindungen, durch das gesamte Netzwerk. Tor benutzt vertrauenswürdige Knoten als Directory Server, die Informationen über Router und deren Zustand bereithalten.

8. Variable Exit Policies: Der Betreiber eines Routers kann selbst bestimmen zu welchen Servern und Ports er Verbindungen erlauben will. Dies erhöht die Zahl der Freiwilligen, die Ressourcen für Tor bereitstellen.

9. Ende-zu-Ende Integritätsschutz: Tor verifiziert die Integrität der transportierten Daten, bevor sie das Netzwerk verlassen. Dies erschwert Angriffe, die Daten verändern, um sie besser verfolgen zu können: Ein Angreifer könnte in einem Paket im Datenteil Bits verändern. Diese Veränderung hat innerhalb des Tor-Netzes keine Auswirkungen, da der Datenteil von den OR nur entschlüsselt wird. Am Rand des Netzwerkes könnte man jedoch Pakete beobachten, deren Daten nicht dem verwendeten Protokoll entsprechen, wie z.B. Kontrollzeichen in Klartextprotokollen.

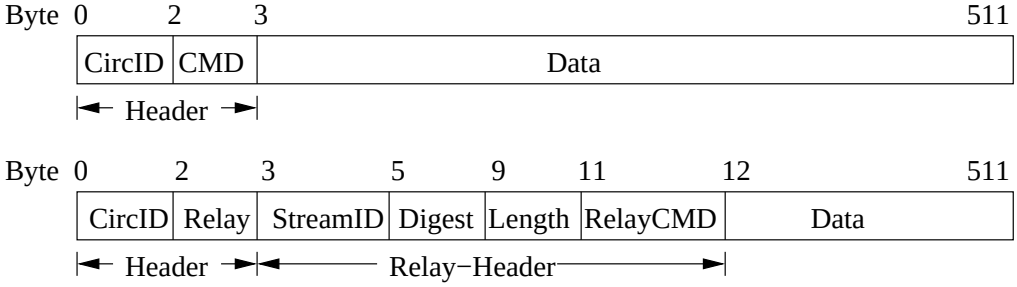
10. Anonyme Dienste: Über Rendezvous-Punkte können Server ihre Dienste anonym anbieten.

2.3 Realisierung anonymer Verbindungen

Es soll nun die Funktionsweise von Tor näher betrachtet werden. In einem Tor-Netzwerk gibt es zwei Arten von Knoten: Onion Router und Onion Proxies. Onion Router (OR) sind untereinander über TLS-Verbindungen vollvermascht verbunden und bilden das Kernnetz. Sie besitzen mehrere Schlüssel: einen Identitätsschlüssel, der über einen längeren Zeitraum gültig ist und nur benutzt wird um TLS-Zertifikate und den *router descriptor* (Datenpaket, das die Schlüssel eines Routers, seine Strategien zum Aufbau von Verbindungen zu Knoten außerhalb des Tor-Netzes und Netzwerkdaten wie seine Verbindungsgeschwindigkeit enthält) zu signieren und einen Onion-Schlüssel, der benutzt wird um Nachrichten an den OR zu entschlüsseln und Sitzungsschlüssel auszuhandeln. Für jede TLS-Verbindung, die ein OR zu einem anderen unterhält, muss er außerdem noch einen Verbindungsschlüssel verwalten. Onion Proxies (OP) sind die Programme, die bei den Tor-Benutzern lokal auf ihren Rechnern laufen. Sie leiten die TCP-Verbindungen eines lokalen Anwendungsprogrammes in das Tor-Netzwerk und bauen Circuits zu einem Endpunkt im Netz auf. Innerhalb dieses Circuits gibt es für jede TCP-Verbindung, die ein OP unterhält, einen *Stream*, der die Daten dieser Verbindung transportiert.

2.3.1 Aufbau einer Zelle

Die Datenübertragung in Tor-Netzwerken erfolgt über Pakete, die Zellen genannt werden. Eine Zelle hat eine Größe von genau 512 Byte und besteht aus einem Header und dem Payload. Es gibt zwei Arten von Zellen: Kommando- und Relayzellen. Relayzellen besitzen einen zweiten Relayheader im Datenfeld.



Die einzelnen Felder haben folgende Bedeutung:

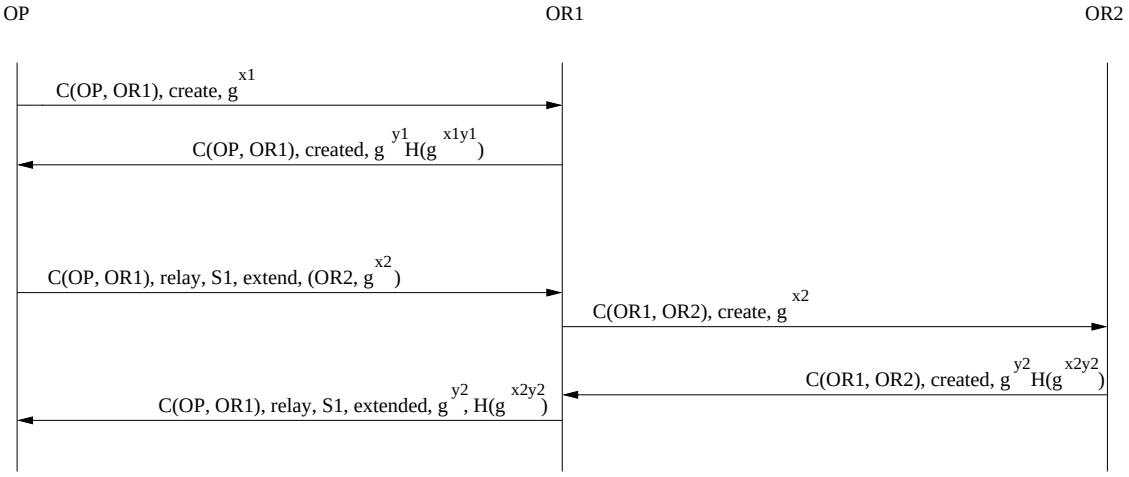
- **CircID**: Gibt den Circuit an zu dem die Zelle gehört.
- **CMD**: Beschreibt den Zweck der Zelle: Sie ist entweder eine Relayzelle oder eine Kommandozelle vom Typ *Padding*, *Create*, *Created* oder *Destroy*.
- **StreamID**: Gibt den Stream innerhalb des Circuits an zu dem die Relayzelle gehört.

- **Digest**: Prüfsumme: Aus dem beim Streamaufbau ausgehandelten Schlüssel wird ein Startwert berechnet, auf den bei jedem neuen Paket ein Hash der Daten addiert wird.
- **Length**: Länge der Nutzdaten.
- **RelayCMD**: Der Typ der Relayzelle: *Data*, *Begin*, *End*, *Teardown*, *Connected*, *Extend*, *Extended*, *Truncate*, *Truncated*, *Sendme* oder *Drop*.

2.3.2 Circuits

Bei Tor können mehrere Streams denselben Circuit nutzen. Der Aufbau eines Circuits kann wegen der dabei benötigten public-key Operationen und Verzögerungen im Netzwerk mehrere Zehntelsekunden dauern. Deshalb werden Circuits nicht erst bei Bedarf aufgebaut, sondern es gibt immer mindestens einen Circuit, der zur Übertragung von Daten genutzt werden kann. Circuits werden jede Minute über einen anderen Pfad neu aufgebaut.

Ein OP baut einen Circuit inkrementell auf: Zuerst wählt er einen Pfad und eine Circuit-ID $C_{(OP,OR_1)}$. Dann sendet er eine mit dem Onion Key von OR_1 verschlüsselte *create*-Kommandozelle an den ersten Knoten OR_1 auf dem Pfad. Darin ist die erste Hälfte g^{x_1} eines Diffie-Hellman Schlüsselaustausches enthalten. OR_1 antwortet mit einer *created*-Kommandozelle, die seine Hälfte g^{y_1} des Schlüsselaustausches und einen Hash KH des dabei ausgetauschten Schlüssels enthält. KH wird mitgeschickt um sicherzustellen, dass beide Endpunkte denselben Schlüssel berechnet haben. In der gegenwärtigen Implementierung [DiMa] ist KH der SHA-1 Hash [EaJo01] von $g^{x_1 y_1}|00$. SHA-1 hat eine Ausgabelänge von 20 Byte. An den Schlüssel wurde ein 0-Byte angehängt, weil aus ihm noch weitere Werte erzeugt werden. Der Startwert für den Digest der Zellen, die der OP sendet, ist das SHA-1 von $g^{x_1 y_1}|01$, der Startwert für den Digest der Zellen auf dem Rückweg ist SHA-1($g^{x_1 y_1}|02$). Analog zur Berechnung von KH ist 01 das Byte mit dem Wert 1, 02 ist das Byte mit dem Wert 2. Auf dieselbe Weise werden auch zwei symmetrische Schlüssel gewonnen, einer für jede Richtung: Die Hashwerte von $g^{x_1 y_1}|03$ und $g^{x_1 y_1}|04$ werden konkateniert. Die ersten 16 Byte bilden den Schlüssel für den Datenstrom vom OP zum OR, die nächsten 16 Byte den Schlüssel für den Datenstrom vom OR zum OP. Die restlichen 8 Byte werden verworfen. Diese Werte berechnen OP und OR_1 lokal. Sie können sich sicher sein, dabei zum selben Ergebnis zu kommen, weil sie sich mittels KH davon überzeugt haben, denselben Wert für $g^{x_1 y_1}$ zu haben.



Um den Circuit zu erweitern, stößt der OP jetzt bei OR_1 dieselbe Prozedur an. Dazu sendet der OP eine *extend*-Relayzelle an OR_1, die im Payload die Adresse von OR_2 und die mit dem Onion Key von OR_2 verschlüsselte erste Hälfte g^{x_2} eines DH-Schlüsselaustausches mit OR_2 enthält. OR_1 kopiert diese erste Hälfte und sendet sie mit einer *create*-Kommandozelle

mit der Circuit-ID $C_{(OR_1,OR_2)}$ an OR_2. Aus der von OR_2 zurückgesendeten *created*-Zelle kopiert OR_1 die zweite Hälfte des Schlüsselaustausches und den Hash des Schlüssels in eine *relay extended*-Zelle und sendet diese an den OP. Nachdem der OP und OR_2 über ihren gemeinsamen Schlüssel zwei symmetrische Schlüssel ausgetauscht haben, besteht der Circuit nun aus dem OP, OR_1 und OR_2. Um den Circuit um einen weiteren Knoten zu erweitern, sendet der OP wieder eine *relay extend*-Zelle mit dem neu aufzunehmenden Knoten an den letzten Knoten des Circuits, der daraufhin wieder eine *create*-Zelle schickt. dadurch kennt jeder OR auf dem Circuit lediglich seinen Vorgänger und seinen Nachfolger.

Da der OP die Zellen jeweils mit dem Onion Key des neu hinzukommenden ORs verschlüsselt, ist dieser implizit authentisiert. Der OR hingegen weiß nicht, wer den Circuit öffnen will. Da der OR den Hash des Schlüssels zurück sendet, weiß der OP, dass die Nachricht vom richtigen OR kommt.

Um eine Relayzelle an einen bestimmten OR A auf dem Circuit zu schicken, verschlüsselt ein OP, der alle OR auf dem Circuit kennt, eine Zelle mit den symmetrischen Schlüsseln aller ORs auf dem Weg zu A. Wenn A die Zelle erhält, wird er nach dem Entschlüsseln feststellen, dass der Digest einen gültigen Wert besitzt und das in der Relayzelle angegebene RelayCMD ausführen.

Um einen Circuit abzubauen, sendet der OP eine *destroy*-Kontrollzelle an den ersten Knoten auf dem Circuit. Dieser beendet alle zu diesem Circuit gehörigen Streams und sendet eine *destroy*-Zelle an den nächsten Knoten.

Alternativ kann der OP auch eine *truncate*-Relayzelle an einen OR auf dem Circuit schicken. Dieser sendet eine *destroy*-Zelle an den nächsten OR auf dem Pfad und antwortet dem OP mit einer *truncated*-Relayzelle. Auf diese Weise kann der OP den Weg des Circuits ändern, ohne dies den dazwischenliegenden ORs mitzuteilen. Falls ein OR auf einem Circuit ausfällt, sendet der auf dem Pfad vor ihm liegende eine *truncated*-Zelle an den OP, der dann den übriggebliebenen Circuit wie eben beschrieben wieder erweitern kann.

2.3.3 Streams

Wenn ein Anwendungsprogramm eine TCP-Verbindung öffnen möchte, gibt es den Verbindungswunsch über die SOCKS-Schnittstelle an den lokalen OP weiter. Der OP wählt den neuesten offenen Circuit (einen solchen gibt es immer, weil Circuits auf Vorrat und nicht bei Bedarf angelegt werden), wählt einen der ORs auf dem Circuit als *Exit Node* und sendet diesem eine *begin*-Relayzelle mit einer zufällig gewählten StreamID. Der *Exit Node* antwortet mit einer *connected*-Relayzelle an den OP, der via SOCKS die Anwendung vom erfolgreichen Verbindungsaufbau informiert. Die Daten des TCP-Streams werden in *data*-Zellen gepackt und über das Tor-Netzwerk an den *Exit Node* geschickt. Dabei enthält das Paket für jeden OR auf dem Weg einen Header und einen Relay-Header.

Der Streamabbau erfolgt analog zum TCP-Verbindungsabbau über einen zwei-Wege Handshake. Der OP sendet eine *end*-Relayzelle an den *Exit Node*, dieser antwortet ebenfalls mit einer *end*-Zelle. Hat ein Enpunkt eine *end*-Zelle gesendet, aber noch keine empfangen, so ist der Stream in einem halbgeschlossenen Zustand. Für den Fall eines Verbindungsabbruchs sendet der Vorgänger des ausgefallenen Knotens eine *teardown*-Zelle an den OP.

Die transportierten Daten müssen nicht nur vor unbefugtem Lesen, sondern auch vor Veränderung geschützt werden. Ist dies nicht gegeben, sind mehrere Angriffe denkbar. Wenn ein Angreifer raten kann, welche Daten eine Zelle enthält, könnte er sie manipulieren und beobachten, an welchen Stelle nicht wohlgeformte Pakete das Tor-Netz verlassen und auf diese Weise eine Zuordnung herstellen. Da die Verbindungen zwischen den ORs mittels TLS

geschützt sind, sind sie vor Veränderung durch einen Angreifer außerhalb des Circuits geschützt. Vor Veränderungen durch einen OR auf dem Circuit werden Nachrichten durch das Digest-Feld im Header der Relayzelle geschützt. Beim Aufbau des Circuits wurde bereits durch Hashen des Diffie-Hellman Schlüssels ein gemeinsames Geheimnis D berechnet. Wenn einer der Endpunkte jetzt eine Zelle sendet, berechnet er aus dem alten D durch Addition des SHA-1-Hashwertes der gesendeten Daten ein neues D und schickt die ersten vier Bytes dieses Wertes im Digest-Feld mit.

Um zu verhindern, dass im Overlay-Netzwerk Verbindungen überlastet werden, muss dort eine zusätzliche Staukontrolle eingeführt werden. Jeder OR hat für jeden Circuit und jeden Stream zwei Fenster: Das *Packaging Window* gibt die Anzahl der Zellen an, die er aus Richtung des OPs weiterschickt, das *Delivery Window* enthält die Anzahl der Pakete, die er noch zum OP zurückzuschicken bereit ist. Jedesmal wenn eine Zelle verschickt wird, wird das Fenster des jeweiligen Circuits und des Streams für die entsprechende Richtung dekrementiert. Der Wert des Fensters wird durch *Sendme*-Relayzellen wieder erhöht, die ein Knoten schickt, wenn er nicht überlastet ist. Diese Zellen tragen die Stream-ID des Streams zu dem sie gehören. Ist die Stream-ID Null, so betrifft die Nachricht den Circuit.

2.3.4 Directory Server

Um über ein Tor-Netzwerk Daten übertragen zu können, muss ein OP oder OR Informationen über der Zustand des Netzwerkes haben: Dazu zählen die Adressen der ORs, die Teil des Netzwerkes sind und der Zustand der Verbindungen zwischen ihnen. Diese Informationen werden von Directory Servern (DS) vorgehalten, die ihre Informationen von den einzelnen ORs bekommen. Die ORs senden periodisch signierte Zustandsberichte an jeden Directory Server, die daraus einen signierten Zustandsbericht des ganzen Netzes (ein *Directory*) erstellen. Danach sendet jeder DS sein Directory an jeden anderen DS und die DS einigen sich auf ein gemeinsames Directory. Alle mit diesem Directory einverstandenen DS signieren es. Jeder OP hat eine vorgegebene Liste von Directory Servern, von denen er regelmäßig Directories abruft. Ein OR akzeptiert ein Directory nur, wenn es mindestens von einer Mehrheit der DS signiert wurde. Zur Zeit werden drei Directory Server betrieben.

Ziel dieses Verfahrens ist es, eine einheitliche Sicht des Netzwerkes zu schaffen. Könnte ein Angreifer den Directory Servern unterschiedliche Informationen zuspielen und damit unterschiedliche Sichten des Netzwerkes erzeugen, könnte er den unterschiedlichen Wissensstand der einzelnen Knoten ausnutzen oder möglicherweise sogar das Netz partitionieren, indem er jedem Directory Server eine andere Teilmenge der OR schickt. Diese einheitliche Sicht könnte auch mit nur einem DS erreicht werden; die Vervielfältigung dieser Informationen soll den Netzausfall bei Ausfall des einzigen DS verhindern und die Anfragen der OR nach Directories auf mehrere DS verteilen. Gleichzeitig wird es einem Angreifer erschwert eigene OR in das Netz einzubringen, denn dafür müsste er eine Mehrheit der DS kompromittieren.

2.4 Realisierung anonymer Dienste

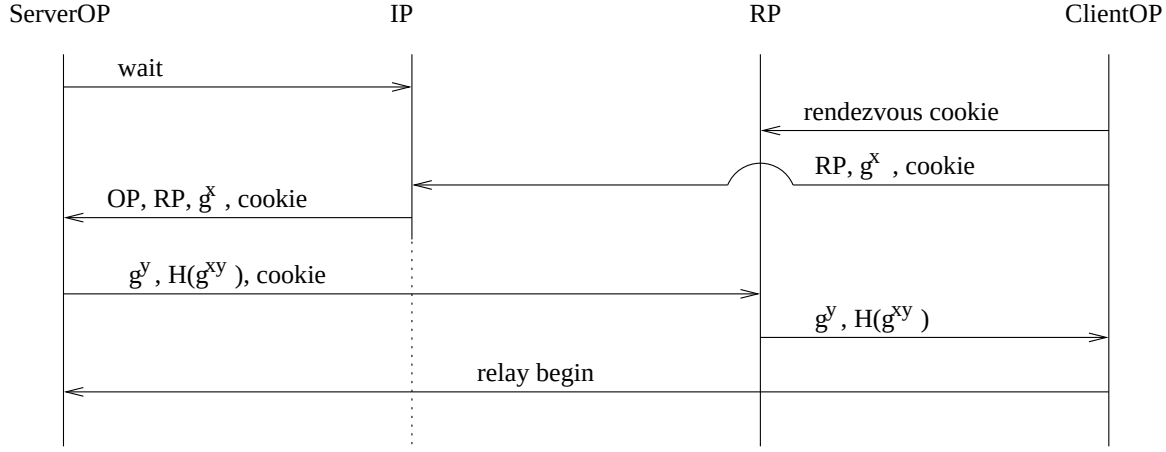
Tor bietet nicht nur Clients, sondern auch Servern die Möglichkeit anonym zu bleiben. Dabei kann ein Server innerhalb eines Tor-Netzwerks *Introduction Points* anlegen, die es einem Client ermöglichen mit ihm Kontakt aufzunehmen. Der Client wählt einen *Rendezvous Point* (RP), baut zu ihm eine anonyme Verbindung auf und teilt dem Introduction Point den RP mit. Der Introduction Point teilt dem Server diese Information mit und der Server baut eine anonyme Verbindung zum RP und damit zum Client auf.

2.4.1 Ziele

Anonyme Dienste sollen mehrere Eigenschaften haben: Damit der Dienst eine möglichst hohe Verfügbarkeit hat, soll das Dienstangebot des Servers auch dann gewahrt bleiben, wenn ORs ausfallen. Ausserdem muss der Server eingehende Anfragen filtern können, so dass ein Angreifer ihn nicht durch den Aufbau vieler Verbindungen lahmlegen kann. Das Angebot soll bentzerfreundlich sein, also soll eine Anwendung, die einen versteckten Dienst nutzen will, dafür nicht geändert werden müssen. Als für anonyme Dienste speziellen Punkt kommt noch hinzu, dass ein Rendezvous Point nicht für Dienste verantwortlich gemacht werden darf, die er selbst nicht anbietet. Wenn ein von ihm vermittelter Dienst illegal ist, merkt der RP nichts davon, weil er die Daten, die er weiterleitet, nicht lesen kann.

2.4.2 Rendezvous Points

Der Server generiert ein Langzeit-Schlüsselpaar, das den angebotenen Dienst identifiziert und wählt mehrere Introduction Points. Der OP des Servers kennt die IP und den Port des angebotenen Dienstes, den öffentlichen Schlüssel und eine Strategie für die Client-Autorisation. Der OP baut einen Circuit zu jedem Introduction Point auf und wartet auf Nachrichten von ihnen. Der OP publiziert anonym eine signierte Nachricht, die den öffentlichen Schlüssel, eine Gültigkeitsdauer und eine Liste der Introduction Points enthält. Diese Nachricht schickt er an alle Directory Server. Der angebotene Dienst selbst bleibt unmodifiziert und weiß nicht einmal, dass er anonymisiert wird. Der Client besorgt sich von den DS die Introduction Points des Servers. Er wählt einen OR als RP, baut eine Verbindung zu ihm auf und übermittelt ihm einen Zufallswert, anhand dessen er den Server erkennen kann. Dann baut er eine Verbindung zu einem Introduction Point auf und übermittelt ihm die Adresse des RP, den Zufallswert und den ersten Teil eines DH-Schlüsselaustausches. Der Introduction Point leitet diese Nachricht an den Server weiter. Falls der Server sich entscheidet, eine Verbindung mit dem Client zuzulassen, baut er eine Verbindung zum RP auf, schickt den Zufallswert, die zweite Hälfte des DH-Austausches und einen Hash des damit vereinbarten Schlüssels. Der RP verbindet die beiden Circuits. Der Client schickt eine *begin*-Relayzelle zum Server und öffnet damit einen Stream über den die TCP-Verbindung laufen kann.



2.5 Angriffe

2.5.1 Angreifermodell

Bei der Analyse von Tor wird ein Angreifer angenommen, der einen Teil des Netzwerkes beobachten kann. Er kann einen Teil des Netzes und auch einen Teil der OR kontrollieren

und hat auch Kontrolle über einen Teil der Verbindungen, auf denen er beliebig Verkehr erzeugen, verändern, abfangen oder verzögern kann.

2.5.2 Passive Angriffe

Passive Angriffe beinhalten nur das Beobachten, nicht aber das Verändern des Netzverkehrs. Beobachtet ein Angreifer einen bestimmten OP, so kann er daraus Rückschlüsse ziehen, selbst wenn er den Inhalt des Pakete nicht lesen kann. Interaktive Anwendungen haben andere Verkehrsmuster als beispielsweise der Download großer Dateien. Verkehr an der Netzgrenze ist nur verschlüsselt, wenn das Anwendungsprotokoll eine Verschlüsselung ausgehandelt hat. HTTP-Pakete in und aus dem Netzwerk heraus enthalten Daten im Klartext, die benutzt werden können, um Benutzer zu identifizieren. Um dies zu verhindern müssen Anonymisierer für die einzelnen Anwendungsprotokolle eingesetzt werden. Der Angreifer kann Vermutungen bezüglich der Kommunikationsendpunkte bestätigen, indem er Schwankungen der Datenrate und zeitliche Abstände zwischen Datenströmen beobachtet. Eine mögliche Maßnahme dagegen ist das Betreiben eines OR auf dem selben Rechner, auf dem auch der OP läuft. Dadurch ist es nicht möglich zu unterscheiden, welche Pakete von dem Rechner selbst stammen und welche nur weitergeleitet werden. Es werden immer mehr Pakete gesendet als empfangen, da durch *Padding*-Zellen falls nötig künstlicher Verkehr erzeugt wird.

2.5.3 Aktive Angriffe

Aktive Angriffe beinhalten das Verändern oder Unterdrücken ausgetauschter Daten und Angriffe auf die Knoten eines Netzwerkes mit dem Ziel diese auszuschalten oder zu übernehmen. Im Folgenden werden die häufigsten Angriffe kurz besprochen.

Wenn ein Angreifer einen OP kontrolliert, kann er die komplette Kommunikation mithören, bevor sie verschlüsselt und anonymisiert wird. Deshalb ist es wichtig, vor der Installation eines OP die digitale Signatur zu überprüfen. Wenn ein Angreifer einen OR unter seiner Kontrolle hat, kennt er die Zuordnung von eingehenden und ausgehenden Circuits, die durch diesen OR verlaufen. Wenn er mehrere OR kontrolliert, sei es durch Einbringen eigener OR in das Netz, die Übernahme bereits im Netz befindlicher Knoten oder einer Kombination daraus, kann er versuchen Netzwerkverkehr an sich zu ziehen, indem er andere OR gezielt überlastet und damit ihre Verbindungsqualität senkt. Dagegen hilft die bereits besprochene Staukontrolle. Weiter könnte er bei passierenden Paketen Bits kippen und am Rand des Netzwerkes nach Paketen mit sinnlosem Inhalt suchen. Wird ein feindlicher OR als Endpunkt eines Circuits gewählt und sind die Datenströme des Protokolls nicht authentifiziert, so kann der OR die gesendeten Daten nach Informationen durchsuchen oder die Rolle des eigentlichen Kommunikationspartners übernehmen bzw. einen Man-in-the-Middle Angriff ausführen.

Ein Angreifer kann über das Tor-Netzwerk illegale Aktivitäten durchführen und damit versuchen das Netzwerk über sozialen oder juristischen Druck zumindest zu verkleinern.

2.5.4 Angriffe gegen Directory Server

Ein Tor-Netzwerk kann mehrere hundert OR enthalten, besitzt in der Regel aber nur eine geringe Anzahl an Directory Servern. Ein Angreifer kann dies ausnutzen und die DS angreifen. Wenn er eine Mehrheit der DS lahmlegt oder zumindest davon abhält, sich auf eine Sicht des Netzes zu einigen, ist effektiv das komplette Netz lahmgelegt, da Directories von der Mehrheit der DS signiert werden müssen, um akzeptiert zu werden. Kann er eine Mehrheit der DS kontrollieren, so kann er bestimmen, welche OR Teil des Netzes sein dürfen. Damit kann er

eigene OR in das Netz einfügen oder durch Verkleinerung des Netzes die Möglichkeiten der OP bei der Pfadwahl einschränken. Er kann auch versuchen die DS davon zu überzeugen, OR unter Kontrolle des Angreifers in das Netz aufzunehmen. Zum gegenwärtigen Zeitpunkt gibt es ausser dem IP-Bereich keine Möglichkeit, mehrere OR demselben Eigentümer zuzuordnen.

2.5.5 Angriffe gegen versteckte Dienste

Möglichkeiten speziell einen versteckten Dienst anzugreifen, beinhalten das kompromittieren oder lahmlegen von Introduction Points oder Rendezvous Points, falls diese bekannt sind. Um effektiv zu sein, muss der Angreifer möglichst viele Introduction Points treffen.

2.6 Bewertung

Tor versucht einen Kompromiss zwischen Anonymität und Benutzbarkeit zu finden. Es kann kein System geben, das alle Benutzer zufriedenstellt: Höhere Anonymität hat ihren Preis in Form von höherem Verkehrsaufkommen und längeren Wartezeiten. Benutzer mit unterschiedlichen Ansprüchen an den Grad der Anonymität und unterschiedlicher Bereitschaft dafür Einbußen an Verbindungsqualitäten hinzunehmen, benötigen unterschiedliche Systeme. Tor hat ein relativ starkes Angreifermodell und dennoch einen geringen Overhead und erlaubt auch Verbindungen für interaktive Protokolle wie SSH oder Instant Messaging, die schnelle Reaktionszeiten benötigen. Im Tor-Netzwerk sind zur Zeit ca. 700 Router aus Ländern der ganzen Welt im Einsatz. Der Client ist leicht zu installieren, es gibt eine gute Dokumentation und das Netz funktioniert meistens ohne allzu grosse Verzögerungen. Dadurch besitzt Tor eine breite Nutzerbasis, die Voraussetzung für wirkungsvolle Anonymität ist.

3 JAP

JAP ist ein Anonymisierungsdienst, der nach einem etwas anderen Prinzip funktioniert als Tor. Bei JAP sendet der lokal installierte JAP-Client Daten über eine von mehreren fest vorgegebenen Mix-Kaskaden, anstatt sich den Pfad durch das Netz selbst aufzubauen. Der Benutzer erhält über einen *InfoService* genannten Dienst Rückmeldung über den Zustand der einzelnen Kaskaden und den Grad seiner Anonymität.

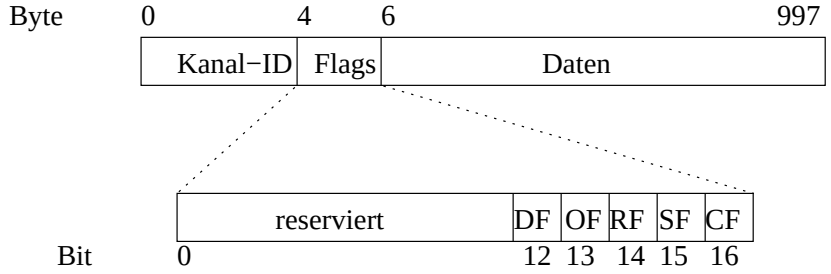
3.1 Funktionsweise

Bei JAP gibt es analog zum Tor-Kernnetz eine Menge von Mix-Kaskaden. Diese bestehen aus über TCP verbundenen einzelnen Mixen. Der letzte Mix einer Kaskade ist über Proxies mit dem Internet verbunden. Mixe leisten die eigentliche Anonymisierung. Sie sammeln Pakete um sie umkodiert (entschlüsselt) und umsortiert wieder ausgeben. Dadurch wird eine Zuordnung von ausgehenden zu eingegangenen Paketen verhindert. Zusätzlich wird Dummy-Traffic erzeugt. Das Verfahren ist sicher, solange wenigstens ein Mix in der Kaskade unkorrumpiert ist. Eine Mix-Kaskade besteht aus einem fest vorgegebenen Pfad von Mixen, die in keiner anderen Kaskade vorkommen. Der JAP-Client baut also nicht selbst einen Pfad, sondern wählt lediglich eine der Mix-Kaskaden aus, um seine Daten zu übermitteln. Diese starre Struktur bedeutet auch, dass das Kernnetz relativ stabile Mitgliedschaften besitzt: Neue Mixe werden nur aufgenommen, wenn sie sich beim Verwalter des Projekts anmelden und eine Selbstverpflichtungserklärung unterzeichnen. Die Einhaltung dieser Erklärung wird vom Unabhängigen Landeszentrum für Datenschutz in Kiel überprüft.

3.2 Realisierung einer anonymen Verbindung

3.2.1 Aufbau eines Paketes

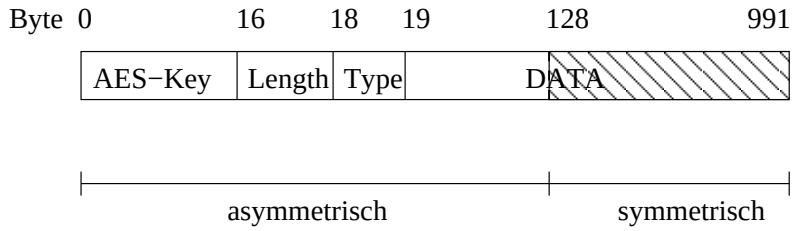
Die Datenübertragung erfolgt über Mix-Pakete, die jeweils einem Mixkanal zugeordnet sind. Die Mix-Pakete besitzen eine Größe von 998 Byte und haben folgenden Aufbau:



Dabei beinhaltet das Feld Kanal-ID die eindeutige Bezeichnung eines Kanals. Es sind folgende Flags definiert:

- das Open-Flag: wird vom JAP-Client gesetzt, um einen neuen Kanal zu öffnen
- das Close-Flag: wird gesetzt, um einen Kanal zu schließen
- das Dummy-Flag: wird gesetzt, wenn das Paket Dummy-Traffic enthält
- das Channel-Suspend-Flag: wird vom ersten Mix gesetzt und signalisiert dem letzten Mix in der Kaskade, dass er keine Daten mehr schicken soll
- das Channel-Resume-Flag: wird vom ersten Mix gesetzt, um dem letzten Mix mitzuteilen, dass er die Datenübertragung fortsetzen kann

Der Rest des Flag-Feldes ist für zukünftige Flags reserviert. Bei gesetztem Open-Flag sind die ersten 128 Byte des Datenteils mit dem öffentlichen Schlüssel des Mixes, an den das Paket gerichtet ist, mit RSA verschlüsselt und enthalten als erstes einen 16 Byte langen AES-Schlüssel, der für das ab diesem Zeitpunkt eingesetzte schnellere AES verwendet wird. 128 Byte ist die Mindstblockgröße des eingesetzten RSA-Verfahrens, deshalb wird ein Teil der Nutzdaten asymmetrisch, der Rest symmetrisch verschlüsselt. Die ersten 3 Byte der Nutzdaten enthalten den Payloadheader: 2 Byte codieren die tatsächliche Länge der Nutzdaten, das nächste Byte bestimmt den Typ des Payloads und damit den Proxy der Kaskade an den die Daten gesendet werden.



3.2.2 Der Mix-Kanal

Ein JAP-Client öffnet zuerst eine normale TCP-Verbindung zu einem ersten Mix in einer Kaskade und wartet auf Anfragen von Anwendungsprogrammen. Wenn eine solche Anfrage kommt, erzeugt er ein erstes Mix-Paket mit gesetztem Open-Flag, um einen Kanal zu öffnen. Die Payload dieses Pakets wird folgendermaßen erzeugt: Der Client kennt die Länge der

Kaskade und für jeden Mix in der Kaskade dessen öffentlichen Schlüssel. Zuerst wählt der Client eine zufällige Kanal-ID id_1 und für jeden Mix M_i in der Kaskade einen 128-Bit AES-Schlüssel k_i und trägt das Tupel $(id_1, k_1, \cdots, k_n)$ in seine Kanal-Tabelle ein. Dann stellt er die Nutzdaten für den letzten Mix in der Kaskade zusammen: Er konkateniert k_n, den Payload-Header und die Nutzdaten und füllt den Rest mit Zufall auf. Die ersten 128 Byte dieses Paketes werden mit dem öffentlichen Schlüssel von M_n verschlüsselt, der Rest mit k_n. So wird auch für die anderen Mixe in der Kaskade verfahren, wobei jeweils das Ergebnis der letzten Runde die Nutzdaten für den nächsten Mix sind.

Wenn nun M_1 dieses Paket empfängt, überprüft er, ob das Open-Flag gesetzt ist. Da dies der Fall ist, entschlüsselt er die ersten 128 Byte mit seinem geheimen Schlüssel, speichert k_1 und entschlüsselt den Rest des Paketes mit k_1. Aus diesen Nutzdaten löscht er seinen symmetrischen Schlüssel, hängt 16 Byte Zufall an und schickt ein Paket mit gesetztem Open-Flag und zufällig gewählter Kanal-ID id_2 an M_2, den nächsten Mix in der Kaskade. Das Tripel (id_1, id_2, k_1) speichert er in seiner Kanal-Tabelle.

Ist das *open*-Flag nicht gesetzt, so handelt es sich um ein Datenpaket, das durch die Kaskade durchgereicht und dabei natürlich von jedem Mix entschlüsselt wird. Der letzte Mix kann den Datenteil lesen und findet darin eine Anfrage an einen Rechner im Internet. Diese sendet er, nimmt die Antwort entgegen, verschlüsselt sie mit seinem Schlüssel und schickt die über die Kaskade an den JAP-Client. Der Client entschlüsselt für jeden Mix auf der Kaskade mit dessen Schlüssel und erhält dann die Antwort auf seine Anfrage.

3.3 Angriffe

Es sollen nun einige Angriffe beschrieben werden. Bei einem Replay-Angriff zeichnet der Angreifer eine Nachricht an einen Mix auf und sendet sie später noch einmal an denselben Mix. Dieser wird sie genau wie die erste entschlüsseln. Damit kann der Angreifer, der alle von diesem Mix gesendeten Daten überwacht, den Datenstrom nach einem doppelten Paket durchsuchen. Damit hat er das zu einem eingehenden Paket gehörige ausgehende Paket gefunden. Um diesen Angriff zu verhindern, speichert jeder Mix die Hashwerte aller Pakete, die ihn innerhalb eines Zeitabschnittes passieren. Erkennt er ein Paket als doppelt, so muss er es verwerfen.

Kennt ein Angreifer die Größe der Sendewarteschlange eines Mixes, so kann er versuchen, sie mit eigenen Paketen fast vollständig zu füllen. Von diesen kennt er den Weg, den sie beim Verlassen der Kaskade nehmen, so dass es nur noch wenige Möglichkeiten gibt, welche eingehenden Pakete welchen Weg genommen haben könnten.

Da es nur wenige Mixe gibt, könnte ein Angreifer versuchen möglichst viele eigene Mixe in ein Netzwerk aufnehmen zu lassen und damit möglicherweise sogar eine eigene Kaskade besitzen. Damit könnte er alle durch seine Kaskade laufenden Nachrichten zuordnen.

3.4 Rechtliche Probleme

Beim JAP-Projekt gab es bereits Interessenskonflikte zwischen den Betreibern und den Strafverfolgungsbehörden. Im Juni 2003 erbat das hessische Landeskriminalamt (LKA) bei der TU Dresden, einem der Betreiber des JAP-Dienstes, Auskunft über die IP-Adresse, die einem JAP-Benutzer zu einem Zeitpunkt in der Vergangenheit zugeordnet war. Es gibt in diesem Zusammenhang zwei Möglichkeiten der Strafverfolger die Anonymität aufzuheben. Die erste ist ein Auskunftsanspruch nach § 100g und § 100h StPO, der die Offenlegung von im normalen Betrieb gesammelten Daten erzwingen kann. Das Unabhängige Landeszentrum für Datenschutz (ULD), ein Partner im JAP-Projekt, teilte dem LKA mit, dass diese Daten nicht ermittelt werden könnten, da keinerlei Daten gespeichert würden.

Daraufhin erkundigte sich ein Mitarbeiter des LKA, ob eine Überwachung der Zugriffe auf eine bestimmte Seite im Internet in der Zukunft möglich sei. Er wurde darüber informiert, dass eine Überwachung möglich gemacht werden könne, es dafür allerdings eines richterlichen Beschlusses bedürfe. Diese Aussage bezieht sich auf § 100a und § 100b der StPO, der die Anordnung der Überwachung und Aufzeichnung von Daten regelt. Diese Anordnung ist nur dann zu erwirken, wenn der begründete Verdacht einer in diesem Paragraphen aufgeführten Straftat vorliegt. Im Juni 2003 meldete sich ein Beamter des BKA mit derselben Frage und wurde auf die entsprechenden Paragraphen hingewiesen. Daraufhin ging dem ULD ein richterlicher Beschluss zu, der eine Auskunft über die Besucher eines bestimmten Forums anordnete, dies allerdings auf der Grundlage von § 100g und § 100h. Auf deren Grundlage besteht aber nur die Pflicht zur Auskunft über Daten, die für den normalen Gebrauch gespeichert werden. Das ULD legte gegen diesen Beschluss Beschwerde ein; da eine Beschwerde aber keine aufschiebende Wirkung hat, musste der Beschluss dennoch umgesetzt werden. Da sich die komplette Kaskade unter Kontrolle der TU Dresden befand, konnte eine Rückverfolgungsfunktion implementiert werden. Der Quellcode für JAP ist allerdings öffentlich zugänglich, so dass die Funktion entdeckt wurde und in Foren und Newsgroups Diskussionen auslöste. Über laufende Ermittlungsverfahren dürfen jedoch keine Informationen veröffentlicht werden, so dass von Seite des ULD keine Stellungnahme erfolgen konnte.

Am 11. Juli 2003 setzte das Landgericht Frankfurt am Main den Beschluss des Amtsgerichtes auf die Beschwerde hin aus. Das Landgericht hob den Beschluss des Amtsgerichtes im September 2003 auf und folgte in ihrer Begründung den Ausführungen des ULD. Davon unberührt bleibt die Möglichkeit einer Anordnung nach § 100a StPO; da alle Mixe des Netzes im Einzugsbereich der deutschen Gerichte liegen, besteht prinzipiell also immer noch die Möglichkeit, dass Nutzer von JAP überwacht werden.

4 Zusammenfassung und Fazit

Onion Routing benutzt ein Netzwerk, bei dem möglichst viele Benutzer Router beisteuern sollen; damit wird die Last verteilt. Dadurch dass Benutzer ihre eigenen Circuits aufbauen und Daten diese Circuits an jeder beliebigen Stelle verlassen können, soll die Zuordnung von Circuits zu Benutzern erschwert werden. JAP stellt mehrere Mix-Kaskaden bereit. Dadurch, dass weniger Wege zur Auswahl stehen, sollen möglichst viele Daten den gleichen Weg nehmen und dadurch das Unterscheiden zwischen ihnen erschwert werden. Mixe werden nicht von Privatpersonen betrieben, da sie eine hohe Verfügbarkeit und Bandbreite haben müssen. Über den Nutzen von Umordnen und Padden gibt es unterschiedliche Meinungen. [BaGS01] bemerken, dass in der aktuellen Version von Freedom auf Verkehrsformung verzichtet wird, da der Nutzen im Vergleich zum Aufwand zu gering ist. [CaLy05] haben sogar bewiesen, dass es ein Onion Routing Modell gibt, das im Universal Composability Framework [Cane01] sicher ist; die Anonymität ist also auch ohne Verkehrsformung sogar unter der schlechtestmöglichen Netzwerksituation gewahrt. Dazu wird ein spezielles Padding eingesetzt, bei dem Tags das korrekte Verhalten der Router garantieren. Im praktischen Einsatz hat Tor gegenüber JAP mehrere Vorteile: Durch die Größe des Tor-Netzes kann Tor eine schnelle Verbindung bereitstellen, bei der man kaum Verzögerungen bemerkt. Eine Verbindung via JAP hat eine spürbare Verzögerung gegenüber einer nicht anonymisierten Verbindung. Das liegt zum einen an dem hohen Grad der Netzauslastung, zum anderen daran, dass Mixe Nachrichten sammeln um sie umzusortieren. Des weiteren unterstützt JAP nur die Protokolle HTTP, HTTPS und FTP; als Grund wird das hohe Missbrauchspotential von peer-to-peer Programmen, Instant Messaging, SSH und Email angegeben. Tor hat sich zur Aufgabe gemacht die Anonymisierung möglichst vieler Protokolle, die über TCP laufen zu ermöglichen.

Literatur

[BaGS01] Adam Back, Ian Goldbarg und Adam Shostack. Freedom Systems 2.1 Security Issues and Analysis. White paper, Zero Knowledge Systems, Inc., May 2001.

[CaLy05] Jan Camenish und Anna Lysyanskaya. A Formal Treatment of Onion Routing. In Victor Shoup (Hrsg.), *Proceedings of CRYPTO 2005*. Springer-Verlag, LNCS 3621, August 2005, S. 169–187.

[Cane01] Ran Canetti. Universally Composable Security: A New Paradigm for Cryptographic Protocols. In *Proc. 42nd IEEE Symposium on Foundations of Computer Science (FOCS)*, 2001, S. 136–145.

[CSWH00] Ian Clarke, Oskar Sandberg, Brandon Wiley und Theodore W. Hong. Freenet: A Distributed Anonymous Information Storage and Retrieval System. In *Proceedings of Designing Provacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability*, July 2000, S. 46–66.

[DaDM02] George Danezis, Roger Dingledine und Nick Maethewson. Mixminion: Design of a Type III Anonymous Remailer Protocol. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, May 2002.

[DiMa] Roger Dingledine und Nick Mathewson. Tor Protocol Specification. http://tor.eff.org/cvs/tor-stable/doc/tor-spec.txt.

[DiMS04] Roger Dingledine, Nick Mathewson und Paul Syverson. Tor: The Second-Generation Onion Router. In *Proccedings of the 13th USENIX Symposium*, August 2004.

[EaJo01] D. Eastlake und P. Jones. US Secure Hash Algorithm 1 (SHA1). Internet Engineering Task Force: RFC 3174, September 2001.

[FGMF+99] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach und T. Berners-Lee. Hypertext Transfer Potocol: HTTP/1.1. Internet Engineering Task Force: RFC 2616, June 1999.

[FrMo02] Michael J. Freedman und Robert Morris. Tarzan: A Peer-to-Peer Anonymizing Network Layer. In *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS 2002)*, November 2002.

[Kö04] Stefan Köpsell. AnonDienst - Design und Implementierung, Januar 2004.

[MCPS03] Ulf Möller, Lance Cottrell, Peter Palfrader und Len Sassaman. Mixmaster Protocol — Version 2, July 2003.

[SyGR97] Paul Syverson, David Goldschlag und Michael Reed. Anonymous Connections and Onion Routing. In *Proceedings of the 1997 IEEE Symposium on Security and Privacy*. IEEE CS Press, May 1997, S. 44–54.

Anonyme Routing-Verfahren in Ad-hoc-Netzen

Pascal Birnstill

Kurzfassung

Forschungsergebnisse zum Thema „Sicherheit in mobilen Ad-hoc-Netzwerken“ wurden bereits zahlreich veröffentlicht. Nur wenige davon betrachten allerdings den Aspekt der Anonymität, der in einigen Einsatzszenarien mobiler Ad-hoc-Netze essentiell ist. Diese Ausarbeitung stellt drei ausgewählte Routingprotokolle zur Etablierung von Anonymität in mobilen Ad-hoc-Netzwerken vor und diskutiert diese Ansätze im Hinblick auf erreichbare Anonymität, Skalierbarkeit und Robustheit gegenüber Angriffen.

1 Einleitung

Mobile Ad-hoc-Netzwerke sind im Vergleich zu drahtgebundenen Netzen wesentlich leichter angreifbar. Diese Tatsache geht unmittelbar einher mit den Eigenschaften, die von mobilen Ad-hoc-Netzwerken gefordert werden. Neben dem drahtlosen Medium, das grundsätzlich für jedermann abhörbar ist, müssen kooperative Algorithmen ausgeführt werden, welche die dynamische Topologie unterstützen. Dies macht wiederum eine zentrale Überwachung des Netzes unmöglich. Ein Knoten, der Daten durch das Netz zu einem Zielknoten übertragen möchte, ist also mangels Infrastruktur darauf angewiesen, dass seine Daten von anderen Knoten weitergeleitet werden, bis diese das gewünschte Ziel erreicht haben [ZWKB+04].

Diese Eigenschaften mobiler Ad-hoc-Netze führen unmittelbar auf die Fragestellung, wie Kommunikation unter solchen Rahmenbedingungen geschützt werden kann, weisen aber zugleich auch auf deren Komplexität hin. Protokolle, die Sicherheit im Sinne von Vertraulichkeit, Integrität und Authentizität für die Kommunikation in mobilen Ad-hoc-Netzen gewährleisten, wurden bereits zahlreich veröffentlicht. Für das Problem der Anonymität, das im nächsten Abschnitt präziser formuliert wird, existieren dagegen bisher nur wenige Lösungsvorschläge. Drei dieser anonymen Routingprotokolle werden im Kern dieser Ausarbeitung beschrieben und analysiert.

1.1 Problemstellung

In einigen Einsatzszenarien mobiler Ad-hoc-Netzwerke genügt es nicht, dass ein mithörender Angreifer die Kommunikation nicht entschlüsseln oder unentdeckt manipulieren kann. Sind die Identitäten der Knoten nicht geschützt, so kann bereits die Information, dass Kommunikation zwischen diesen Knoten stattfindet, für einen Angreifer wertvoll sein. Kann ein Angreifer allerdings Paketflüsse vom Sender zum Ziel (über mehrere Hops hinweg) verfolgen und den Umfang der Kommunikation beobachten oder Regelmäßigkeiten (sogenannte *Kommunikationsmuster*) erkennen, so ist er auch ohne Kenntnis der Identitäten in der Lage, Rückschlüsse auf die Wichtigkeit bzw. Funktion der Knoten zu ziehen. Stellt man sich als Einsatzszenario beispielsweise eine geheime Operation auf fremdem Territorium vor, so stößt man auf ein weiteres Problem. Ein passiver Angreifer sollte aus abgehörten Übertragungen keine Hinweise

auf die geographische Position der kommunizierenden Knoten, also auf deren Entfernung in Hops und deren Mobilität, gewinnen können [KHSG05], [ZWKB+04].

Alle diese Forderungen sind untrennbare Teilaspekte des Oberbegriffs der *Anonymität* (in den Abschnitten 2.1.1 bis 2.1.3 genauer definiert) und sollen von den in dieser Ausarbeitung betrachteten Routingprotokollen erfüllt werden.

1.2 Gliederung

Im folgenden Abschnitt 2 werden einige Grundlagen zusammengefasst, die dem besseren Verständnis dieser Ausarbeitung dienen sollen. Der wichtigste Punkt ist dabei das *Ad-hoc On-demand Distance Vector* Routingprotokoll (AODV), dessen Struktur sich in den anschließend beschriebenen Protokollen für anonymes Routing wiederfindet. Das erste dieser Protokolle ist *Anonymous On Demand Routing (ANODR)*, das in Abschnitt 3 in drei Varianten beschrieben wird, die sukzessive die wichtigsten Kriterien beim Design anonymer Routingprotokolle für mobile Ad-hoc-Netzwerke erläutern. Abschnitt 4 befasst sich mit dem *Anonymous Secure Routing* Protokoll (ASR), das eng mit ANODR verwandt ist, jedoch stärkere Anonymität verspricht. Das letzte in dieser Ausarbeitung vorgestellte Routingprotokoll ist schließlich *Secure Distributed Anonymous Routing (SDAR)* in Abschnitt 5, das im Unterschied zu ANODR und ASR einen vertrauensbasierten Ansatz verfolgt.

2 Grundlagen

Dieser Abschnitt stellt eine Zusammenstellung von Definitionen und Kurzfassungen grundlegender Konzepte dar, auf die an den entsprechenden Stellen in dieser Ausarbeitung verwiesen wird.

Zunächst wird in 2.1 der Begriff der Anonymität definiert. 2.2 beschreibt typische Angriffe auf mobile Ad-hoc-Netze. Anschließend werden in 2.3 anhand des *Ad-hoc On-demand Distance Vector* Routingprotokolls (AODV) beispielhaft die Grundzüge des Routings in mobilen Ad-hoc-Netzwerken erklärt. 2.4 führt die für das Verständnis der anonymen Routingprotokolle erforderlichen kryptographischen Grundlagen ein. Schließlich wird in 2.5 das Prinzip des *MIX-Net* bzw. *Onion Routing* beschrieben, welches in ANODR und teilweise auch in SDAR eingesetzt wird.

2.1 Definition von Anonymität

In den folgenden Abschnitten sollen einige Definitionen den Begriff der Anonymität untergliedern und handhabbar machen.

2.1.1 Identity Privacy

Der Begriff *Identity Privacy* impliziert zwei Forderungen. Einerseits muss der Schutz der Identitäten des Quell- und des Zielknotens gegenüber allen anderen Knoten des Netzwerks gewährleistet werden. Andererseits sind auch die Identitäten der Hops auf einer Route gegenüber der Quelle, dem Ziel und den übrigen Knoten zu schützen [ZWKB+04].

2.1.2 Location Privacy

Für *Location Privacy* existieren zwei Abstufungen. *Weak Location Privacy* ist gegeben, wenn kein Knoten die exakten Positionen von Quelle und Ziel kennt, ausgenommen diese beiden Knoten selbst. Können andere Knoten, insbesondere solche, die auf der Route zwischen Quelle und Ziel liegen, keine Informationen über ihre Entfernung - die Anzahl der Hops - zur Quelle oder zum Ziel gewinnen, so ist *Strong Location Privacy* gewährleistet [ZWKB+04].

2.1.3 Route Anonymity

Route Anonymity ist gegeben, wenn die folgenden Forderungen erfüllt sind: Ein passiver Angreifer kann einen Paketfluss weder zur Quelle noch zum Ziel verfolgen, unabhängig davon, ob sich er sich auf der Route oder außerhalb befindet. Weiterhin können Angreifer außerhalb der Route keine Informationen über die Route bzw. deren einzelne Hops gewinnen [ZWKB+04].

2.2 Passive vs. aktive Angriffe

Grundsätzlich wird bei Angriffen auf mobile Ad-hoc-Netzwerke zwischen passiven und aktiven Attacken unterschieden.

Ein *passiver* Angreifer hört den Netzwerkverkehr ab und versucht mit analytischen Methoden Paketflüsse zu verfolgen, um Routen aufzudecken und Hinweise auf die Identitäten anderer Knoten zu gewinnen. Da ein solcher Angreifer kein feststellbar bösartiges Verhalten zeigt, also beispielsweise das Routingprotokoll korrekt ausführt, ist er nahezu unmöglich zu entdecken. Passive Angriffe sind also meist Angriffe auf die in den Abschnitten 2.1.1, 2.1.2 und 2.1.3 beschriebenen Aspekte der Anonymität.

Ein Beispiel für einen passiven Angriff ist die sogenannte *Message Size* Attacke. Dabei versucht ein Angreifer ein Paket bzw. dessen Funktion anhand der Größe zu identifizieren und durch das Netz zu verfolgen [BEKXK04].

Dagegen verfolgt ein *aktiver* Angreifer meist das Ziel, Datenübertragungen zu stören oder gar ganz zu unterbrechen. Dazu kann er beispielsweise Pakete löschen oder in großer Zahl replizieren.

Ein häufiger aktiver Angriff ist z.B. eine *Denial-of-Service*-Attacke (DoS). Bei DoS geht es grundsätzlich darum, einen Dienst durch Überlastung auszuschalten. Spezielle Formen von DoS sind *Distributed DoS*, d.h. koordinierte DoS-Attacken durch eine größere Anzahl von Angreifern, z.B. Bot-Netze, oder *One-to-Multiple DoS*, also ein einzelner Angreifer, der mehrere Hosts angreift [ZWKB+04].

Im Kontext von Routingprotokollen für mobile Ad-hoc-Netze dient eine passive Attacke oft dazu, eine aktive Attacke vorzubereiten, so kann z.B. eine aufgedeckte Route durch eine DoS-Attacke auf einen einzelnen Knoten unterbrochen werden [KHSG05].

2.3 AODV: Ad-hoc On-demand Distance Vector Routing [Perk03]

Ad-hoc On-demand Distance Vector Routing (AODV) ist ein reaktives Routingprotokoll, d.h. Routen werden erst bei Bedarf ermittelt (*Route Discovery*) und nicht periodisch aktualisiert. Stattdessen führt jeder Knoten eine Sequenznummer, die in Route Discovery Phasen inkrementiert wird und somit als Indikator für die Aktualität von Routen dient.

Möchte ein Sender S mit einem Ziel Z kommunizieren und besitzt S noch keine bzw. keine gültige Route zu Z, so initiiert der Knoten S eine Route Discovery Phase, indem er das

Netz mit *Route Request (RREQ)* Paketen flutet. Vereinfacht dargestellt, enthält ein solches RREQ-Paket die IP-Adresse und die aktuelle Sequenznummer von S, eine global eindeutige Broadcast-ID, die IP-Adresse und die letzte bekannte Sequenznummer von Z. Ein Knoten M, der ein RREQ empfängt, aktualisiert in seiner Routingtabelle die Route zurück zu S, sodass ein *Reverse Path* zu S aufgebaut wird. Falls $M = Z$, so generiert M ein *Route Reply (RREP)* Paket und sendet dieses per Unicast zurück an S. Ist $M \neq Z$, so leitet M das RREQ per Broadcast an seine Nachbarn weiter. Um zu verhindern, dass M ein RREQ mehrfach verarbeitet und weiterleitet, werden die Broadcast-ID und die IP-Adresse von S bei M zwischengespeichert, sodass Duplikate erkannt und verworfen werden können.

Beim Weiterleiten eines RREP-Pakets zurück zum Sender S aktualisiert wiederum jeder Hop seine Route zum Ziel Z, um einen *Forward Path* aufzubauen. Erreicht das RREP schließlich S, so kann S direkt mit dem Senden von Daten an Z beginnen. Es ist möglich, dass S noch ein RREP mit einer aktuelleren Route (höhere Sequenznummer von Z) oder einer Route mit niedrigerem Hopcount zu Z empfängt. In einem solchen Fall kann S seine Routinginformationen aktualisieren und fortan die neue Route benutzen; dies ist allerdings implementierungsabhängig.

Der Vollständigkeit halber sei erwähnt, dass der AODV-Standard noch einen zweiten Modus für die Route Discovery Phase vorsieht. Hier kann ein RREQ auch von einem Knoten M, $M \neq Z$, mit einem sogenannten *Gratuitous RREP* beantwortet werden, falls M eine aktuelle Route zu Z (also eine gültige Route mit einer Sequenznummer von Z größer oder gleich der Sequenznummer von Z im RREQ) kennt.

Eine Route gilt als aktiv, solange Daten zwischen Sender S und Ziel Z fließen. Sobald die Route nicht mehr benutzt wird, „altern“ die Routinginformationen in den Knoten. Ist ein vorgegebener Timeout erreicht, so werden die Einträge ungültig und können aus den Routingtabellen verdrängt werden. Um den Ausfall eines Hops auf einer aktiven Route festzustellen sieht AODV zwei Möglichkeiten vor. Einerseits können Quittungen auf der Sicherungsschicht verwendet werden, d.h. wenn ein Knoten für übertragene Pakete keine Quittungen mehr empfängt, so wiederholt er das Senden des ersten unbestätigten Pakets bis zu einem definierten Grenzwert. Wird dieser Grenzwert schließlich überschritten, so wird ein *Route Error* Paket an den sendenden Endknoten der Route (also S oder Z) geschickt. Bei der anderen Lösung muss jeder Knoten, der ein Paket weitergeleitet hat, anschließend für eine vorgegebene Zeit auf dem Kanal lauschen und auf einen Übertragungsversuch des nächsten Hops warten (*Passive Acknowledgement*). Wird eine unterbrochene Route weiterhin benötigt, so muss eine neue Route Discovery Phase initiiert werden, um eine alternative Route zu ermitteln.

Das AODV-Protokoll ist ein weit verbreitetes Routingprotokoll für mobile Ad-hoc-Netze, das keine Anonymität für teilnehmende Knoten unterstützt. Dennoch wurde es hier einführend vorgestellt, da die Protokolle, die im Kern dieser Ausarbeitung betrachtet werden, das Konzept von AODV aufgreifen und dieses um Mechanismen zur Etablierung von Anonymität erweitern.

2.4 Kryptographische Grundlagen

Die folgenden Abschnitte beschreiben die kryptographischen Konzepte, auf welche die anschließend betrachteten anonymen Routingprotokolle zurückgreifen.

2.4.1 Symmetrische Kryptographie

Die symmetrische Kryptographie basiert auf einem gemeinsamen Geheimnis (*Shared Secret*) zwischen zwei Kommunikationspartnern, welches sowohl zum Ver- als auch zum Entschlüs-

seln verwendet wird. Weit verbreitete symmetrische Verfahren sind beispielsweise der *Data Encryption Standard (DES)* und der *Advanced Encryption Standard (AES)*.

Symmetrische Algorithmen sind effizient zu berechnen, allerdings wird ein Verfahren zum sicheren Austausch bzw. zur Etablierung des Schlüssels benötigt (z.B. der *Diffie-Hellman-Schlüsselaustausch*).

2.4.2 Asymmetrische Kryptographie

Bei asymmetrischen Verschlüsselungsverfahren erzeugt jeder Benutzer ein Paar von Schlüsseln, einen öffentlichen (*Public Key*) und einen privaten Schlüssel (*Private Key*). Der Public Key dient der Verschlüsselung von Daten, deren Entschlüsselung ausschließlich mit dem passenden Private Key möglich ist. Folglich muss ein Benutzer seinen Public Key allen anderen Benutzern, mit denen er vertraulich kommunizieren möchte, zugänglich machen, während der Private Key geheim bleibt. Beispiele für asymmetrische Verfahren sind *RSA* und *El-Gamal*.

Grundsätzlich gilt, dass die Ausführung asymmetrischer Algorithmen im Vergleich zu symmetrischen Verfahren deutlich mehr Rechenleistung bzw. Rechenzeit erfordert. Deshalb benutzen sogenannte *hybride* Verfahren asymmetrische Kryptographie um Schlüsselmaterial für symmetrische Algorithmen auszutauschen.

Im Kontext digitaler Signaturen werden private Schlüssel auch zum Verschlüsseln eingesetzt. Eine solche Signatur kann dann entsprechend mit dem Public Key des Erzeugers überprüft werden.

2.4.3 Trapdoor-Hashfunktionen

Das kryptographische Konzept der *Trapdoor-Hashfunktionen* (dt. Geheimtür) kombiniert effizient berechenbare Hashfunktionen mit asymmetrischer Kryptographie (siehe Abschnitt 2.4.2). Kryptographische Hashfunktionen haben die Eigenschaft, dass es „praktisch unmöglich“ ist, zu einem gegebenen Hashwert ein Urbild zu finden. Eine Trapdoor-Hashfunktion berechnet aus einer kurzen Eingabe I (z. B. eine Zufallszahl) und einem Private Key eine Trapdoor und ist mit Kenntnis des entsprechenden Private Keys - der *Trapdoor Information* - effizient umkehrbar. Die Trapdoor-Hashfunktion kann also aus der Trapdoor und dem Private Key wieder I berechnen. Der Besitzer des Private Keys verfügt schließlich mit der Kenntnis von I über einen global nachprüfbaren Beweis für das Öffnen der Trapdoor [KoHo03].

2.5 MIX-Net und Onion-Strukturen

David Chaums *MIX-Net*-Konzept basiert auf der Idee, Anonymität des Senders zu etablieren, indem Pakete auf zufälligen Routen durch ein Netz von MIXes zum Empfänger geroutet werden. Dabei wird angenommen, dass zwischen allen teilnehmenden Knoten bereits vertrauliche Kommunikation, gestützt auf symmetrische oder asymmetrische Verschlüsselung, etabliert wurde. Möchte ein Sender S eine Nachricht m über einen MIX M an einen Empfänger Z senden, so sieht das Paket von S an M folgendermaßen aus:

$$\{Z, N_S^1, \{m, N_S^0\}_{PK_Z}\}_{PK_M}$$

Dabei steht N_X^i für von Knoten X berechnete Zufallszahlen (Nonces), während $\{...\}_{PK_X}$ bedeutet, dass der Inhalt der Klammern mit dem Public Key des Knotens X verschlüsselt ist.

M entschlüsselt die äußere Verschlüsselung, entfernt die Nonce und leitet das Paket an Z weiter. Entsprechend hat ein Paket, das über $n+1$ MIXes weitergeleitet werden soll, die Form

$$\{M_n, N_S^n, \{...\{M_1, N_S^2, \{Z, N_S^1, \{m, N_S^0\}_{PK_Z}\}_{PK_{M1}}\}_{PK_{M2}}\}...\}_{PK_{Mn+1}}$$

Eine solche kryptographische Struktur wird *Onion* (dt. Zwiebel) genannt. Jeder weiterleitende MIX kann genau eine „Schale“ dieser Onion entfernen und kennt somit nur den nächsten Hop auf der Route. Somit ist also bereits eine abgeschwächte Form von Identity Privacy (s. Abschnitt 2.1.1) gegeben. Route Anonymity (s. Abschnitt 2.1.3) kann dagegen ohne zusätzliche Mechanismen nicht gewährleistet werden, da Pakete mittels einer Message Size Attacke (s. Abschnitt 2.2) durch das Netz verfolgt und Routen somit aufgedeckt werden können [KoHo03].

3 ANODR: Anonymous On Demand Routing for Mobile Adhoc Networks

Das Design des ANODR-Protokolls [KoHo03] verfolgt das Hauptziel, Schutz gegenüber passiven Angreifern zu bieten. Es wird also angenommen, dass potentielle Angreifer selbst unentdeckt bleiben wollen, um Datenübertragungen zu analysieren und Routen und Identitäten von Knoten aufzudecken. Deshalb wurde Robustheit gegenüber aktiven Attacken (s. Abschnitt 2.2) nicht explizit gefordert.

Grundsätzliche Voraussetzung für ANODR ist, dass zwei Knoten, die anonym miteinander kommunizieren möchten, auf bereits ausgetauschtes symmetrisches oder asymmetrisches Schlüsselmaterial (s. Abschnitte 2.4.1, 2.4.2) zurückgreifen können.

Analog zu AODV (s. Abschnitt 2.3) zerfällt das Routing auch bei ANODR in die beiden Phasen *Anonymous Route Discovery* und *Anonymous Route Maintenance*. Ziel der Anonymous Route Discovery Phase ist es, während der Suche nach einer Route zu einem gewünschten Ziel (und einer Reverse Route zurück zum Sender), zufällige *Route Pseudonyme* für die einzelnen Hops zu etablieren. Diese Route Pseudonyme werden anstelle der Identitäten oder Adressen der Knoten als Routinginformationen benutzt. In der Anonymous Route Maintenance Phase geht es darum, inaktive Routen aus den Routingtabellen zu entfernen und Ausfälle aktiver Routen bzw. einzelner Hops zu erkennen.

3.1 Anonymous Route Discovery

In der Anonymous Route Discovery Phase wird das Netz wie bei AODV mit RREQ-Paketen geflutet. Diese setzen sich bei ANODR wie folgt aus einer global eindeutigen Sequenznummer seq, einer nur für den gewünschten Empfänger Z zu öffnenden Trapdoor tr_Z (s. Abschnitt 2.4.3) und einer kryptographischen Onion (Definition s. Abschnitt 2.5, Aufbau s. Abschnitte 3.1.1 - 3.1.3) zusammen:

$$[RREQ, seq, tr_Z, onion]$$

Wie die Trapdoor tr_Z zu erzeugen ist, bleibt der Implementierung überlassen, hängt allerdings in erster Linie vom verfügbaren Schlüsselmaterial ab. Unter der Annahme, dass jeder Knoten den Public Key PK_Z von Z kennt, kann die Trapdoor tr_Z wie in Abschnitt 2.4.3 beschrieben aus PK_Z und einer Zufallszahl N_S berechnet werden. Z kann schließlich mit Hilfe seines Private Keys die Trapdoor-Hashfunktion umkehren und N_S in das RREP-Paket (s. unten) einfügen. Ein Knoten, der ein RREP empfängt und tr_Z zwischengespeichert hat, ist nun in der Lage, die Authentizität des RREP-Pakets zu überprüfen, indem er tr_Z nachrechnet.

Die Onion-Struktur dient der Etablierung der bereits erwähnten Route Pseudonyme und ist für den realisierbaren Grad der Anonymität ebenso maßgeblich wie für die Effizienz und Skalierbarkeit des Routingprotokolls. Um diesen Sachverhalt zu verdeutlichen, werden im Folgenden drei Varianten von ANODR diskutiert, die sich hinsichtlich der verwendeteten kryptographischen Onion unterscheiden. Die ersten beiden Ansätze können dabei als Zwischenschritte bei der Entwicklung des endgültigen Entwurfs des ANODR-Protokolls verstanden werden, welche die wichtigsten Designentscheidungen begründen.

Das Problem, in jedem Knoten global eindeutige Sequenznummern zu erzeugen, ist im Allgemeinen schwierig zu lösen. In [KoHo03] wird vorgeschlagen, diese Sequenznummern mittels einer Hashfunktion und einem global eindeutigen Initialwert zu berechnen. Da bei ANODR davon ausgegangen wird, dass jeder Knoten zum Schutz seiner Identität ein global eindeutiges *Identity Pseudonym* besitzt, kann dieses zur Berechnung der ersten Sequenznummer eines Knotens verwendet werden. Eine neue Sequenznummern wird entsprechend berechnet, indem die Hashfunktion auf die letzte verwendete Sequenznummer angewendet wird. Weitere Verfahren zur Etablierung global eindeutiger Sequenznummern werden in [MoCa02] beschrieben.

3.1.1 Ansatz mit asymmetrischer Kryptographie: *ANODR-PO*

ANODR-PO (Public key protected Onion) verfolgt den intuitiven Ansatz, das in Abschnitt 2.5 beschriebene Konzept des MIX-Nets mit asymmetrischer Kryptographie auf mobile Adhoc-Netze zu übertragen.

Empfängt ein Knoten M_i ein RREQ-Paket mit einer bereits bekannten Sequenznummer, so wird das Paket verworfen. Andernfalls wird die PO des RREQ-Pakets erweitert, indem das Identity Pseudonym des Incoming Hops der Onion vorangestellt und das Resultat mit dem eigenen Public Key verschlüsselt wird. Schließlich wird das RREQ per Broadcast weitergeleitet.

Erreicht ein RREQ das gewünschte Ziel Z, so ist die enthaltene PO eine geeignete Onion um eine Reverse Route zurück zum Sender zu aufzubauen. Dazu öffnet Z die Trapdoor tr_Z und generiert ein RREP-Paket, das sich aus einem zufällig zu wählenden Route Pseudonym N für den nächsten Hop, dem Beweis pr_Z (im oben beschriebenen Beispiel wäre dies die von S gewählte Zufallszahl N_S) für das Öffnen der Trapdoor und der mit dem RREQ empfangenen Onion zusammensetzt

$$[RREP, N, pr_Z, onion].$$

Um passive Angriffe grundsätzlich aufwändiger zu machen, werden bei ANODR (im Unterschied zu AODV) auch RREP-Pakete per Broadcast weitergeleitet.

Empfängt ein Knoten M_i ein RREP, so entschlüsselt er die äußerste Schale der enthaltenen Onion mit seinem Private Key. Falls der Schlüssel passt, findet M_i im ersten Feld des Resultats sein eigenes Identity Pseudonym vor und erkennt daran, dass er sich auf der gesuchten Route befindet. Entsprechend vergibt M_i ein Route Pseudonym N' für den nächsten Hop, trägt in seiner Routingtabelle ein, dass eingehende Pakete mit dem Route Pseudonym N künftig unter dem Route Pseudonym N' weiterzuleiten sind (und umgekehrt) und ersetzt im RREP-Paket vor dem Weiterleiten N durch N'. Knoten die beim Entschlüsseln der Onion feststellen, dass sie nicht auf der Route liegen, verwerfen RREP-Pakete.

Der offensichtlichste Nachteil dieses Ansatzes besteht in der Verwendung asymmetrischer Kryptographie. Asymmetrische Verfahren erfordern grundsätzlich sehr viel mehr Rechenleistung als symmetrische Verfahren, was beim Einsatz leistungsschwacher Geräte (z.B. PDAs, Sensorknoten) zu erheblichen Verzögerungen führt [KoHo03].

3.1.2 Ansatz mit symmetrischer Kryptographie: *ANODR-BO*

Der zweite Ansatz ersetzt die bisher verwendete PO-Struktur durch eine Onion mit symmetrischer Verschlüsselung, die aufgrund der höheren Geschwindigkeit, mit der RREQ- und RREP-Pakete durch das Netz fließen, auch als *Boomerang Onion (BO)* bezeichnet wird.

Empfängt ein Knoten M_i ein RREQ-Paket mit einer gültigen Sequenznummer, so erweitert er die BO analog zum vorhergehenden Ansatz, indem er das Identity Pseudonym des Incoming Hop voranstellt. Die Verschlüsselung erfolgt allerdings für jedes RREQ mit einem neuen, zufällig generierten Secret Key $K_{i,seq}$. Entsprechend entschlüsselt M_i bei einem empfangenen RREP-Paket die äußerste Schale der Onion mit dem passenden Geheimnis $K_{i,seq}$ (eindeutige Zuordnung durch global eindeutige Sequenznummer) und leitet es ansonsten unverändert weiter.

Mit dieser Modifikation lässt sich das Problem rechenintensiver asymmetrischer Kryptographie lösen; ANODR-BO kann mittels geeigneter symmetrischer Verschlüsselungsverfahren auch auf leistungsschwachen Geräten ausgeführt werden. Die beiden bisherigen Ansätze haben jedoch ein Problem gemeinsam. Beim Weiterleiten eines RREQ-Pakets muss jeder Knoten gegenüber seinen direkten Nachbarn sein Identity Pseudonym offenbaren, damit dieses in die nächste Schale der Onion eingefügt werden kann. Somit ist die Forderung nach Identity Privacy und Route Anonymity (s. Abschnitte 2.1.1 und 2.1.3) nur eingeschränkt erfüllt. Entsprechend ist das Ziel des dritten und letzten Ansatzes, die Verwendung der Identity Pseudonyme der Knoten im Routingprotokoll zu vermeiden.

3.1.3 Ansatz mit Trapdoor-Hashfunktionen: *ANODR-TBO*

ANODR-TBO (*Trapdoor Boomerang Onion*) zeigt, wie mit Hilfe von Trapdoor-Hashfunktionen eine Anonymous Route Discovery Phase realisiert werden kann, die gänzlich ohne die Identity Pseudonyme der Knoten auskommt. Dazu wird eine Onion aus symmetrischen Trapdoors verwendet, die wie folgt aufgebaut wird: Ein Knoten M_i, der ein RREQ-Paket mit gültiger Sequenznummer empfängt, erweitert die TBO, indem er die bestehende Onion zusammen mit einer Zufallszahl N (N ist hier kein Route Pseudonym, diese werden nach wie vor erst mit den RREP-Paketen transportiert) mit einem zufällig generierten Secret Key $K_{i,seq}$ verschlüsselt. Die resultierende Schale der Onion ist somit eine Trapdoor und die Kombination aus N und $K_{i,seq}$ die zugehörige Trapdoor Information, die nur M_i selbst kennt. Folglich kann die TBO eines RREP-Pakets immer nur vom nächsten Hop auf der Reverse Route - also dem Erzeuger der Trapdoor - korrekt geöffnet werden, sodass das RREP auch nur von diesem Knoten per Broadcast weitergeleitet werden kann.

Auch unter dem Gesichtspunkt der erforderlichen Rechenleistung für die kryptographischen Operationen stellt ANODR-TBO nochmals eine Verbesserung dar, da die Knoten beim Weiterleiten der RREQ-Pakete nur Trapdoor-Hashfunktionen berechnen müssen. Mehr dazu in Abschnitt 3.3. Der Aufbau der Routen bzw. die Etablierung der Route Pseudonyme verläuft vollkommen analog zu den beiden bisherigen Ansätzen.

3.2 Anonymous Route Maintenance

Analog zu AODV werden Routen auch bei ANODR nur für ein begrenztes Zeitfenster als gültig angenommen und entsprechend nach einem Timeout on-demand neu ermittelt.

Die Erkennung ausgefallener Hops basiert bei ANODR auf der Annahme einer bestätigten Kommunikation mit einem vordefinierten Grenzwert von Sendewiederholungen. Wird dieser Grenzwert überschritten, so sucht der Sender in seiner Routingtabelle alle Einträge N', die

über den ausgefallenen Hop N weiterzuleiten sind, markiert diese Route Pseudonyme als ungültig und verschickt per Broadcast RERR-Pakete vom Format $[RERR, N']$. Ein Knoten, der ein RERR-Paket empfängt, versendet entsprechend weitere RERR-Pakete mit den zu N' korrespondierenden Route Pseudonymen N'', sodass letztlich jeder Knoten seine direkten Nachbarn über ungültige Routen informiert.

3.3 Bewertung von ANODR

Zur Evaluation des Protokolls wurden alle drei oben beschriebenen Entwürfe von ANODR implementiert [KoHo03]. Die Simulationsumgebung bestand aus 50 mobilen Knoten mit einer Funkreichweite von je 250m auf einer Fläche von 1500m x 300m. Zur Ermittlung von Vergleichsdaten diente eine Implementierung des AODV-Protokolls.

Im Ergebnis fallen ANODR-PO und ANODR-BO bei der Ende-zu-Ende-Verzögerung stark gegenüber ANODR-TBO und AODV ab. Dies wird damit begründet, dass die Größe der Onion-Struktur bei Multi-Hop-Routen schnell wächst und entsprechend zu Verabeitungsverzögerungen in den Knoten führt. Analog dazu ist die Performanz von ANODR-TBO stark von der eingesetzten Kryptographie abhängig. So wird in dieser Simulation mit Hilfe des auf ellyptischen Kurven basierenden *ECAES* eine Performanz erreicht, die AODV deutlich näher kommt [KoHo03]. Ob eine ANODR-TBO-Implementierung im praktischen Einsatz skaliert ist bisher nicht nachgewiesen worden und wird in mehreren Quellen angezweifelt [KHSG05], [ZWKB$^+$04].

Darüber hinaus bringt eine Verwendung kryptographischer Onions wie bei ANODR die Gefahr eines Message Size Angriffs (s. Abschnitt 2.2) mit sich, d.h. ein (passiver) Angreifer kann von der Größe der Onion in einem empfangenen RREQ-Paket auf die Anzahl der Hops zum Sender schließen. Beträgt diese Entfernung nur einen Hop, so wird dies besonders problematisch, da der Sender dann mittels einer gerichteten Antenne lokalisierbar ist [ZWKB$^+$04]. Location Privacy (s. Abschnitt 2.1.2) ist damit nur eingeschränkt erfüllt.

4 ASR: Anonymous Secure Routing in Mobile Ad-hoc Networks

Das *Anonymous Secure Routing* Protokoll (ASR) [ZWKB$^+$04] ist eng verwandt mit ANODR und wurde entwickelt, um die in Abschnitt 3.3 genannten Probleme bei ANODR zu lösen. Dazu wird beispielsweise statt der in [ZWKB$^+$04] kritisierten kryptographischen Onion ein One-Time-Pad eingesetzt. Außerdem soll ASR Sicherheit gegenüber aktiven Attacken bieten. In der folgenden Beschreibung des ASR-Protokolls wird angenommen, das zwischen Sender S und Ziel Z bereits ein gemeinsames Geheimnis existiert.

4.1 Anonymous Route Discovery

Die Route Discovery Phase beginnt auch bei ASR mit dem Versenden von RREQ-Paketen per Broadcast. Diese haben hier das Format

$$[RREQ, seq, K_{shared}(Z, K_{Session}, U_0), K_{Session}(seq, END), PK_{i-1}, U_{i-1}]$$

mit

- $seq \rightarrow$ eine global eindeutige Sequenznummer

- $K_{shared} \rightarrow$ das vorausgesetzte gemeinsame Geheimnis von Sender S und Ziel Z
- $K_{Session} \rightarrow$ der Session Key der aktuellen Route Discovery Phase
- $U_0 \rightarrow$ eine von S gewählte Zufallszahl
- $END \rightarrow$ ein vom Protokoll definierter Code dafür, dass das RREQ Z erreicht hat
- $PK_{i-1} \rightarrow$ der Public Key eines one-time Schlüsselpaars, das der vorhergehende Hop M_{i-1} erzeugt
- $U_{i-1} \rightarrow$ eine Zahl, die M_{i-1} berechnet

Dabei kennzeichnet die Schreibweise $K(...)$ eine symmetrische Verschlüsselung mit dem Shared Secret K.

Die von S gewählte Zufallszahl U_0 und die in jedem Knoten M_i weiter berechnete Zahl U_{i-1} ermöglichen das Mitführen des Hopcounts, ohne diesen jedoch für Knoten $M_i \neq Z$ offenzulegen. Somit kann S eine maximale Entfernung für eine zu ermittelnde Route festlegen, während weiterleitende Knoten ohne Kenntnis von U_0 keine Informationen über ihre Entfernung zu S gewinnen können. Z hingegen kann U_0 entschlüsseln und damit aus U_{i-1} den Hopcount berechnen. Der genaue Algorithmus, nach dem U_i in jedem Knoten neu berechnet wird, ist nicht Gegenstand dieser Ausarbeitung, wird aber in [ZWKB+04] detailliert beschrieben.

Empfängt ein Knoten M_i ein RREQ, so überprüft er zunächst anhand der Sequenznummer, ob bereits ein entsprechender Eintrag in seiner Routingtabelle existiert. Falls ja, wird das Paket verworfen, ansonsten versucht M_i, $K_{shared}(Z, K_{Session}, U_0)$ zu entschlüsseln. Scheitert dies, so ist $M_i \neq Z$, d.h., das RREQ wird per Broadcast weitergeleitet. Zuvor speichert M_i noch die Routinginformationen, bestehend aus Sequenznummer, PK_{i-1} und $K_{Session}(seq, END)$, berechnet aus U_{i-1} die nächste Zahl U_i und ersetzt PK_{i-1} und U_{i-1} durch PK_i bzw. U_i.

Falls M_i $K_{shared}(Z, K_{Session}, U_0)$ entschlüsseln kann, so folgt daraus entsprechend $M_i = Z$. Anhand der Werte von U_0 und U_n überprüft Z, ob die Länge der gefundenen Route kleiner oder gleich dem von S gewünschten maximalen Hopcount ist. Ist die Route zu lang, so verwirft Z das RREQ, ansonsten wird ein RREP-Paket generiert.

Zu diesem Zeitpunkt kennt also jeder Knoten M_i auf der Route den Public Key PK_{i-1} des nächsten Knotens auf dem Reverse Path zu S und Z kennt die Länge aller Routen zwischen S und Z, welche die von S vorgegebene maximale Länge nicht überschreiten.

Das Format der RREP-Pakete ist bei ASR

$$[RREP, \{T_{i+1}\}_{PK_i}, T_{i+1}(seq, K_{Session})]$$

mit

- $T_{i+1} \rightarrow$ eine von M_{i+1} gewählte Zufallszahl, die (nach der Route Discovery Phase) als Secret Key zwischen M_i und M_{i+1} dient
- $K_{Session} \rightarrow$ der Beweis dafür, dass der Zielknoten Z den dritten Teil des RREQ-Pakets $K_{shared}(Z, K_{Session}, U_0)$ entschlüsselt hat

Die Schreibweise $..._{PK_X}$ kennzeichnet dabei eine asymmetrische Verschlüsselung mit dem Public Key PK_X des Knotens X, $K(...)$ wieder eine symmetrische Verschlüsselung mit dem Shared Secret K.

Jeder Knoten, der ein RREP empfängt, versucht zunächst $\{T_{i+1}\}_{PK_i}$ mit seinem Private Key zu entschlüsseln. Passt der Schlüssel nicht, so liegt der Knoten nicht auf der Route und das

RREP wird verworfen. Andernfalls handelt es sich um den Knoten M_i (den nächsten Knoten auf dem Reverse Path), der nun das Geheimnis T_{i+1} kennt und damit den hinteren Teil des RREP-Pakets entschlüsseln kann. M_i schlägt die enthaltene Sequenznummer in seiner Routingtabelle nach und verwirft das RREP, falls kein entsprechender Eintrag vorhanden ist. Existiert dagegen ein Eintrag, so bleibt zu prüfen, ob das RREP authentisch ist, d.h. ob es tatsächlich von Z stammt . Dazu wird $K_{Session}(seq, END)$ (aus der Routingtabelle) mit dem Schlüssel $K_Session$ aus dem RREP-Paket nachgerechnet und anschließend verglichen; bei Ungleichheit wird das RREP verworfen.

Ist das RREP gültig, so berechnet M_i einen zufälligen Schlüssel T_i und ergänzt seine Routinginformationen zu der gegebenen Sequenznummer mit T_i und T_{i+1}. Dann werden $\{T_i\}_{PK_{i-1}}$ und $T_i(seq, K_{Session})$ erzeugt, um damit das RREP-Paket zu aktualisieren, das schließlich per Broadcast weitergeleitet wird.

Ist das RREP beim Sender eingetroffen, so verfügt jeder Knoten M_i auf der Route über je ein gemeinsames Geheimnis mit dem nächsten Knoten auf dem Reverse Path (T_{i+1}) und dem nächsten Knoten auf dem Forward Path (T_i) zwischen S und Z.

4.2 Anonymous Data Transmission und Route Maintenance

Im Unterschied zu ANODR werden bei ASR während der Anonymous Route Discovery Phase keine Route Pseudonyme (s. Abschnitt 3.1) etabliert. Stattdessen wird bei der Datenübertragung für jeden Hop eine ähnlich kleine Information erzeugt, die als TAG bezeichnet und vollkommen analog zu den Route Pseudonymen benutzt wird.

Seien M_i und M_{i+1} benachbarte Knoten auf einer Route mit dem gemeinsamen Geheimnis T_{i+1} und $H(K, I)$ eine schnelle HMAC-Funktion mit einem Schlüssel K und einer Eingabe I als Parameter. Dann erzeugt M_i den TAG_i für ein Datenpaket von M_i zu M_{i+1} als $[N, H(T_{i+1}, N)]$, wobei N eine von M_i beliebig initialisierte Zahl ist, die mit jedem auf dieser Route verschickten oder empfangenen Paket inkrementiert wird.

Wie die Route Pseudonyme bei ANODR werden die $TAGs$ auch verwendet, um Informationen über ausgefallene Hops zu verteilen. Ein entsprechendes RERR-Paket hat demnach das Format $[RERR, TAG]$.

4.3 Bewertung von ASR und Vergleich mit ANODR

Bei ASR kann ein Angreifer aus abgefangenen RREQ-Paketen keine Rückschlüsse auf die Entfernung zum Sender ziehen, da sich die Länge der einzelnen Felder in einem RREQ während des Weiterleitens nicht verändert. Somit ist das Problem, das bei ANODR durch die stetig wachsende kryptographische Onion (s. Abschnitt 3.3) gegeben ist, gelöst, d.h., Location Privacy (s. Abschnitt 2.1.2) ist hier auch für den Sender gewährleistet [ZWKB+04].

Wie in Abschnitt 3 bereits erwähnt, war Robustheit gegenüber aktiven Attacken bei ANODR kein primäres Designziel. Entsprechend kann ein One-to-Multiple-DoS-Angriff (s. Abschnitt 2.2) auf ANODR relativ leicht gefahren werden, indem ein Angreifer das Netz mit gefälschten RREQ- oder RREP-Paketen flutet und somit empfangende Knoten mit der Berechnung kryptographischer Operationen beschäftigt. Diese Attacke soll bei ASR nicht mehr funktionieren, da für RREQs der kryptographische Rechenaufwand bei der Verarbeitung minimiert wurde und für RREPs der in Abschnitt 4.1 beschriebene Mechanismus zur Authentifizierung des Ziels eingebaut wurde [ZWKB+04].

Insgesamt ist ASR also als Optimierung des ANODR-Protokolls hinsichtlich der Aspekte Anonymität und Robustheit gegenüber aktiven Angriffen einzuordnen [KHSG05]. Die Frage nach

einer Verbesserung bei Skalierbarkeit und Performanz ist hingegen bislang unbeantwortet, da keine entsprechenden Simulationsergebnisse vorliegen.

5 SDAR: Secure Distributed Anonymous Routing for Mobile Ad-hoc Networks

Das Design des *Secure Distributed Anonymous Routing* Protokolls (SDAR) [BEKXK04] greift wie ASR einige Konzepte von ANODR auf. Die Weiterentwicklungen bei SDAR zielen aber weniger auf Optimierung der Effizienz ab, sondern vielmehr darauf, möglichst sichere und zuverlässige Routen zu finden. Um dies zu erreichen, wird ein vertrauensbasierter Ansatz gewählt. Wie Vertrauen unter anonymen Rahmenbedingungen etabliert werden kann, folgt in den nächsten Abschnitten. Zusätzlich ist in SDAR die Option integriert, Nutzdaten verschlüsselt zu übertragen.

5.1 Trust und Community Management

Der Ansatz, den das SDAR-Protokoll verfolgt, um potentiell schädliche oder bösartige Knoten zu identifizieren und schließlich isolieren zu können, basiert auf der Einführung von Vertrauensstufen (*Trust Levels*). Die Vertrauensstufe eines Knotens M_i ist ein Wert, der aus dessen Verhalten in der Vergangenheit resultiert; solange M_i das Protokoll korrekt ausführt, wird seine Vertrauensstufe inkrementiert, bei Fehlverhalten dekrementiert. Berechnet wird die Vertrauensstufe von M_i von jedem seiner direkten Nachbarknoten (völlig unabhängig voneinander). Daraus folgt, dass die Knoten im Netzwerk identifizierbar sein müssen. Um dennoch einen Kompromiss mit der erwünschten Identity Privacy (s. Abschnitt 2.1.1) zu erzielen, werden die Identitäten der Knoten wie bei ANODR durch Identity Pseudonyme geschützt, welche auch vom zugrundeliegenden Transportprotokoll als Adresse verwendet werden sollten.

Nach der Definition von SDAR bildet ein Knoten M_i (in dieser Funktion als *Central Node* bezeichnet) zusammen mit seinen Nachbarknoten, die einen Hop von M_i entfernt sind, eine sogenannte *Community*. Die Zusammensetzung einer solchen Community kann sich allerdings durch die Mobilität der Knoten jederzeit ändern. Deshalb lauscht M_i auf HELLO-Nachrichten, die jeder Knoten periodisch per Broadcast sendet. Eine HELLO-Nachricht enthält das Identity Pseudonym und den Public Key ihres Absenders; der Empfänger M_i speichert diese Informationen ab, falls er sie nicht bereits besitzt. Somit kann M_i neue Nachbarn in seine Community aufnehmen und Knoten, von denen innerhalb eines bestimmten Zeitfensters kein HELLO eingeht, aus der Liste seiner Nachbarn streichen.

Für jede Community unterteilt der Central Node M_i seine Nachbarn in Abhängigkeit von deren Vertrausensstufe in drei Klassen ein. Außerdem generiert M_i zwei Schlüssel, einen *High Trust Level Community Key (HTLCK)* und einen *Medium Trust Level Community Key (MTLCK)*. Den HTLCK teilt M_i der Klasse der Knoten hoher Vertrauensstufe mit, den MTLCK den Klassen der Knoten hoher und mittlerer Vertrauensstufe, wobei die Übertragung dieser Community Keys durch die in der Community verteilten Public Keys der Knoten geschützt ist. Mit der Klasse der Knoten niedriger Vertrauensstufe wird kein Schlüssel geteilt.

Entdeckt der Central Node M_i einen neuen Nachbarknoten (HELLO-Nachricht mit unbekanntem Public Key), so setzt M_i dessen Vertrauensstufe initial auf einen Wert, der geringfügig über dem Schwellenwert zwischen den Klassen der Knoten niedriger und mittlerer Vertrauensstufe liegt. Der neue Nachbar befindet sich also zunächst in der Klasse der Knoten mittlerer Vertrauensstufe und erhält entsprechend den MTLCK von M_i - geschützt durch seinen Public Key. Solange M_i das Protokoll korrekt ausführt, wird seine Vertrauensstufe bei seinen Nachbarknoten inkrementiert, bei bösartigem Verhalten (s. Abschnitt 5.2) dekrementiert. Falls

ein Knoten in eine andere Klasse auf bzw. absteigt oder die Community verlässt, muss der betroffene Community Key (eventuell auch beide Community Keys) erneuert werden.

5.2 Erkennung bösartigen Verhaltens von Knoten

An dieser Stelle bleibt noch zu klären, wie ein Knoten bösartiges Verhalten eines Nachbarknotens erkennen kann, wobei bei SDAR zwischen zwei Arten bösartigen Verhaltens von Knoten unterschieden wird. Verwirft ein Knoten ein Paket, welches er nach den Vorgaben des Protokolls weiterleiten müsste, so wird dies als *Malicious Dropping* bezeichnet, während unter *Malicious Modification* das ebenfalls unzulässige Ändern eines weiterzuleitenden Pakets zu verstehen ist. Generell kann negatives Verhalten von Knoten festgestellt werden, indem ein Knoten nach dem Senden eines Pakets noch für eine bestimmte Zeit auf dem Kanal lauscht und überprüft, ob seine Nachbarn das Paket korrekt weiterleiten. Entsprechend muss auch der tatsächliche Zielknoten einer Übertragung Kopien der empfangenen Pakete weiterleiten, um seine Anonymität zu schützen und seine Vertrauensstufe nicht zu gefährden (vgl. 5.5).

5.3 Vertrauensbasiertes Routing

Das in den letzten beiden Abschnitten beschriebene Trust und Community Management wird nun eingesetzt, um Einfluss auf die Sicherheit und Zuverlässigkeit einer Route zu nehmen.

Initiiert ein Sender S eine Route Discovery Phase (analog zu den bisher betrachteten Protokollen), so kann S eine Vertrauensstufe für die Hops der zu ermittelnden Route vorgeben, die nicht unterschritten werden darf. S verschlüsselt also das RREQ-Paket mit dem zur geforderten Vertrauensstufe gehörigen Schlüssel (MTLCK oder HTLCK) und schränkt somit den Kreis der an der Weiterleitung beteiligten Nachbarn auf entsprechend vertrauenswürdige Knoten ein. Diese verfahren beim Verarbeiten des RREQ analog.

5.4 Route Discovery Phase

Wie im letzten Abschnitt bereits erwähnt, wird eine Route Discovery Phase auch bei SDAR durch das Versenden von RREQ-Paketen per Broadcast gestartet. Ein solches RREQ eines Senders S, der eine Route zu einem Ziel Z benötigt, setzt sich bei SDAR zunächst aus den vier Teilen

$$[RREQ, TRUST\text{-}REQ, OPK;$$

$$\{\{ID_Z, K_{Session_S}, PL_S\}_{PK_Z};$$

$$P_S;$$

$$\{ID_S, PK_S, OPK, OSK, SessionID_S, Sign_S\}_{K_{Session_S}}\}_{HTLCK/MTLCK}]$$

zusammen mit

- $TRUST\text{-}REQ \rightarrow$ die von S geforderte Vertrauensstufe (HIGH, MEDIUM oder LOW)
- $OPK \rightarrow$ ein von S generierter one-time Public Key
- $ID_X \rightarrow$ das Identity Pseudonym des Knotens X
- $K_{Session_X} \rightarrow$ ein vom Knoten X generierter Session Key (symmetrisch)
- $PL_S \rightarrow$ die Länge des von S eingefügten Paddings

- $P_S \rightarrow$ ein von S eingefügtes Padding
- $PK_Z \rightarrow$ der Public Key des Zielknotens Z
- $OSK \rightarrow$ der zu OPK korrespondierende one-time Private Key
- $SessionID_X \rightarrow$ eine von Knoten X zufällig gewählte Session-ID
- $Sign_X \rightarrow$ eine von Knoten X mit dessen Private Key erzeugte Signatur (ein kryptographischer Hash zur Integritätssicherung)

Der vordere Teil des RREQ-Pakets ist unverschlüsselt; hier sind die für die Verarbeitung bzw. Weiterleitung des RREQs nötigen Informationen enthalten. Der one-time Public Key OPK dient dabei einerseits zum Verschlüsseln von Routinginformationen (s. unten, Verarbeitung von RREQ-Paketen), die von weiterleitenden Knoten an das RREQ angehängt werden, und andererseits als eindeutiger Identifikator des RREQs. Falls S die Vertrauensstufe HIGH oder MEDIUM fordert, werden alle weiteren Teile des RREQ-Pakets zusätzlich mit dem entsprechenden Community Key (HTLCK bzw. MTLCK) verschlüsselt. Der zweite Teil des RREQ-Pakets ist mit dem Public Key PK_Z des Zielknotens Z verschlüsselt, sodass nur Z selbst erkennen kann, dass er das Ziel einer gesuchten Route ist. Es wird davon ausgegangen, dass S den Schlüssel PK_Z besitzt oder beispielsweise bei einer CA (Certification Authority) bekommen kann. Die Verteilung von Public Keys im Netz wird vom SDAR-Protokoll nur für die unmittelbare Nachbarschaft (durch das Community Management, s. Abschnitt 5.1) übernommen. Der folgende Teil des RREQs, ein Padding zufälliger Länge, soll verhindern, dass ein passiver Angreifer von der Größe des Pakets auf dessen Hopcount schließen kann, und somit Location Privacy (s. Abschnitt 2.1.2 gewährleisten. verschleiert die tatsächliche Größe des Pakets und verhindert somit, dass ein passiver Angreifer das Paket anhand seiner Größe identifizieren und verfolgen kann (Message Size Angriff, vgl. 3.3). Die Analyse von RREQ-Paketen stellt also keine Bedrohung für die Route Anonymity (s. Abschnitt 2.1.3) dar. Da nur Z die Länge des Paddings und den Session-Key $K_{Session_S}$ aus dem zweiten Teil des RREQs entschlüsseln kann, sind die Informationen im vierten Teil des Pakets ebenfalls nur Z zugänglich. Mit dem enthaltenen Private Key OPK kann Z schließlich die am Ende des RREQ-Pakets gesammelten Routinginformationen (s. unten, Verarbeitung des von RREQ-Paketen) entschlüsseln.

Die Verarbeitung von RREQ-Paketen verläuft ähnlich zu den bisher betrachteten Protokollen. Empfängt ein Knoten M_i ein RREQ, so überprüft er zunächst anhand des OPK, ob er dieses Paket bereits erhalten hat und verwirft es gegebenenfalls. Handelt es sich um ein neues RREQ-Paket der Vertrauensstufe LOW, so kann es auf jeden Fall verarbeitet werden, bei Vertrauensstufe HIGH oder MEDIUM sucht M_i in seiner Liste nach dem entsprechenden Community Key. Falls dieser nicht vorhanden ist, verwirft M_i das RREQ, ansonsten wird es mit dem Community Key entschlüsselt. Als nächstes versucht M_i, den zweiten Teil des RREQ-Pakets, $\{ID_Z, K_{Session_S}, PL_S\}_{PK_Z}$, mit seinem Private Key zu entschlüsseln, um herauszufinden, ob $M_i = Z$ ist.

Falls $M_i \neq Z$, werden Routinginformationen der Form

$$\{ID_{M_i}, K_{Session_{M_i}}, SessionID_{M_i}, Sign_{M_i}\}_{OPK}$$

an das RREQ angehängt. Somit wächst die Größe des Pakets also linear mit jedem Hop (vgl. Abschnitt 5.5). Anschließend wird das RREQ-Paket (bis auf den vordersten Teil) je nach geforderter Vertrauensstufe mit dem entsprechenden Community Key des Knotens M_i verschlüsselt und per Broadcast weitergeleitet. Die für M_i zu speichernden Routinginformationen bestehen aus der $SessionID_{M_i}$, dem Identity Pseudonym des Vorgängerknotens (aus dem Transportprotokoll) und dem Session-Key $K_{Session_{M_i}}$.

Ist $M_i = Z$, so kann Z den zweiten Teil des RREQ-Pakets entschlüsseln. Mit der Länge des Paddings PL_S und dem Session-Key $K_{Session_S}$ kann somit auch der vierte Teil des RREQs entschlüsselt werden, der den one-time Private Key OSK enthält. Mit Hilfe von OSK erhält Z schließlich Zugang zu den Session-Keys $K_{Session_{M_i}}$ und den Identity Pseudonymen ID_{M_i} der Knoten M_i auf der ermittelten Route. Mit diesen Session-Keys erzeugt Z analog zu ANODR eine kryptographische Onion (s. Abschnitte 2.5 bzw. 3) für den Reverse Path, die in das Route Reply Paket eingefügt wird. In der innersten Schale dieser Onion, die mit dem Session-Key $K_{Session_S}$ des Senders S verschlüsselt ist, befinden sich die Session-IDs $SessionID_{M_i}$ und die Session-Keys $K_{Session_{M_i}}$ aller Knoten M_i auf der Route in der Reihenfolge des Forward Paths, sowie ein Padding.

Erreicht das RREP-Paket schließlich den Initiator S der Route Discovery Phase, so hat S mit den Session-Keys $K_{Session_{M_i}}$ der Knoten M_i die Möglichkeit, die zu übertragenden Nutzdaten in eine kryptographische Onion einzubetten und vertraulich zum Ziel Z zu übertragen.

5.5 Bewertung von SDAR

Bei näherer Betrachtung des Designs von SDAR fällt ein grundlegendes Problem auf: In Abschnitt 5.2 wird erwähnt, dass ein Knoten auch Kopien von Paketen, welche für ihn selbst bestimmt sind, weiterleiten muss. Dies ist einerseits erforderlich, um nicht von einem potentiellen Angreifer als Ende einer Route erkannt werden zu können; andererseits würde das Nichtweiterleiten dieser Pakete von den Nachbarknoten - der Community des Knotens - als Malicious Dropping interpretiert werden und zu einer Dekrementierung der Vertrauensstufe führen. Demnach würden Pakete unendlich oft weitergeleitet werden, was zwangsläufig zu einer Überlastung des Netzwerks führen muss. Ohne einen Mechanismus, der es einem Knoten ermöglicht, Duplikate oder veraltete Pakete zu verwerfen, ohne seine Vertrauensstufe zu gefährden, kann das Protokoll somit sicherlich nicht implementiert und praktisch eingesetzt werden.

Im Hinblick auf die durch SDAR etablierte Anonymität ist noch der Apekt der Route Anonymity (s. Abschnitt 2.1.3) zu diskutieren. Durch das in Abschnitt 5.4 beschriebene Padding zufälliger Länge ist lediglich die initiale Größe der RREQ- bzw. RREP-Pakete zufällig. Durch das Anhängen der Routinginformationen an das RREQ bzw. Hinzufügen einer weiteren Schale zur kryptographischen Onion beim RREP in jedem weiterleitenden Knoten wächst die Paketgröße allerdings linear, sodass Paketflüsse mittels einer Message Size Attacke verfolgbar sind. Route Anonymity wird somit von SDAR nicht gewährleistet.

Die Robustheit gegenüber aktiven Attacken basiert bei SDAR auf der Angriffserkennung durch das Trust und Community Management (s. Abschnitt 5.1). DoS-Angriffe können damit allerdings nicht unterbunden werden [BEKXK04].

Unter dem Aspekt der Effizienz betrachtet, lässt sich kaum ein Grund finden, weshalb SDAR besser skalieren sollte als ANODR. Die RREQ- und RREP-Pakete können ebenfalls sehr groß werden, sodass mit Verarbeitungsverzögerungen in den Knoten zu rechnen ist. Ein praktischer Einsatz von SDAR ist somit nur in Szenarien denkbar, in denen die Zuverlässigkeit und Sicherheit der Routen bedeutend wichtiger ist als Effizienz bzw. die vom Protokoll unterstützte Mobilität.

6 Related Work

Ein weiteres Protokoll, welches hier aus Platzgründen nicht detailliert beschrieben werden kann, wurde in [ZhLL05] unter dem Namen *MASK* veröffentlicht. Es basiert auf einem so-

genannten *Neighbourhood Authentication* Protokoll, mit dessen Hilfe sich benachbarte Knoten paarweise authentifizieren und Schlüssel austauschen können, ohne dabei ihre Identität aufzudecken. Die verfügbaren Simulationsergebnisse dokumentieren die Effizienz des MASK-Protokolls nur im Vergleich mit AODV, zeigen aber bereits, dass auch hier die Verarbeitungsverzögerung in den Knoten mit zunehmender Länge der Routen zum Problem wird.

7 Zusammenfassung & Fazit

In dieser Ausarbeitung wurden die anonymen Routingprotokolle ANODR, ASR und SDAR betrachtet. Die Analyse hat gezeigt, dass lediglich das Design des ASR-Protokolls den Forderungen nach Identity Privacy, Strong Location Privacy und Route Anonymity in vollem Umfang gerecht wird. Simulationsergebnisse existieren bisher nur für ANODR, diese deuten allerdings darauf hin, dass auch dessen „Weiterentwicklung“ ASR nicht die nötige Effizienz erreichen kann, um hohe Mobilität zu unterstützen [KHSG05]. Der vertrauensbasierte Ansatz des SDAR-Protokolls erscheint stellenweise noch nicht ganz ausgereift, zeigt aber eine weitere Richtung auf, welche die Forschung beim Entwurf anonymer Routingprotokolle ebenfalls vefolgt.

Die Entwicklung eines Routingprotokolls für mobile Ad-hoc-Netzwerke, das Anonymität mit akzeptabler bis guter Effizienz auch in Szenarien mit höherer Mobilität vereinigt, ist nach wie vor eine große Herausforderung. Das Optimierungspotential liegt dabei höchstwahrscheinlich weniger beim Entwurf effizienterer Mechanismen zur Etablierung von Anonymität als bei den kryptographischen Verfahren, auf denen diese Mechanismen aufsetzen, wie beispielsweise die deutlich divergierenden Simulationsergebnisse der ANODR-Implementierungen mit unterschiedlicher Kryptographie zeigen [KoHo03], [KHSG05].

Literatur

[BEKXK04] A. Boukerche, K. El-Khatib, L. Xu und L. Korba. SDAR: A Secure Distributed Anonymous Routing Protocol for Wireless and Mobile Ad Hoc Networks. IEEE, LCN '04, 2004.

[KHSG05] J. Kong, X. Hong, M. Y. Sanadidi und M. Gerla. Mobility Changes Anonymity: Mobile Ad Hoc Networks Need Efficient Anonymous Routing. IEEE, ISCC'05, 2005.

[KoHo03] J. Kong und X. Hong. ANODR: ANonymous On Demand Routing with Untraceable Routes for Mobile Ad-hoc Networks. ACM, 2003.

[MoCa02] G. Montenegro und C. Castelluccia. Statistically Unique and Cryptographically Verifiable (SUCV) Identifiers and Adresses. Network and Distributed System Security Symposium(NDSS), 2002.

[Perk03] C. Perkins. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561 (Experimental), 2003.

[ZhLL05] Y. Zhang, W. Liu und W. Lou. Anonymous Communications in Mobile Ad Hoc Networks. IEEE, 2005.

[ZWKB+04] B. Zhu, Z. Wan, M. S. Kankanhalli, F. Bao und R. H. Deng. Anonymous Secure Routing in Mobile Ad-Hoc Networks. IEEE, LCN'04, 2004.

Datenschutz für standortbasierte Dienste

Tobias Heitfeld

Kurzfassung

Zum Beginn dieser Seminararbeit wird zunächst ein Überblick vermittelt, was unter dem Begriff der standortbasierten Dienste zu verstehen ist. Diese Dienste werden dann in den Kontext des E-Commerce eingeordnet. Nach dieser ersten Annäherung werden dann die einzelnen Bestandteile von standortbasierten Diensten vorgestellt und näher erläutert. Danach wird das Zusammenwirken dieser Komponenten dargelegt. Es werden auch einige Beispiele für konkrete Anwendungsgebiete, die auf standortbasierten Diensten beruhen angesprochen. Nachdem die Möglichkeiten von standortbasierten Diensten aufgezeigt wurden werden auch die Gefahren für den Datenschutz erwähnt.Es bestehen natürlich auch zahlreiche Risiken wenn eine neue Technologie eingesetzt wird. Zur Vermeidung oder Abschwächung dieser Gefahren werden sowohl technische als auch juristische Mittel vorgestellt und auf ihre Tauglichkeit geprüft. Am Ende dieser Ausführungen wird schließlich mit einem Ausblick in die Zukunftsmöglichkeiten ein Fazit gezogen.

1 Einleitung

Der Begriff der standortbasierten Dienste ist ein recht neues Konzept, das andeutet, dass die Anwendungen in Verbindung mit der geographischen Position des Nutzers einen Dienst erbringen. Mit der Weiterentwicklung der mobilen Kommunikation repräsentieren diese Anwendungen ganz neue Anforderungen an die Technik. Mittlerweile ist schon deutlich geworden, dass das tägliche Leben zukünftig von standortbasierten Diensten, die auf Computern, PDAs, Mobiltelefonen und so weiter integriert sein werden, beeinflusst wird. Dennoch ist es eine sehr komplexe Aufgabe, die Benutzer solcher Anwendungen mit zusätzlichen Informationen zu ihrer aktuellen Position zu versorgen. Bedenkt man dabei alleine die Vielzahl der möglichen Anwendungen und alle grundsätzlichen Voraussetzungen die erfüllt werden müssen. Das sind zum Beispiel die Existenz von Standards, effiziente Rechnerleistung und nicht zuletzt die Mensch-Maschine-Kommunikation. Im Rahmen dieser Arbeit geht es jedoch nur um die Standortbestimmung und die verschiedenen Möglichkeiten von standortbasierten Diensten um damit die Problematik des Datenschutzes zu verdeutlichen.

2 Definition von standortbasierten Diensten

Wie schon in Kapitel 1 erwähnt sind standortbasierte Dienste ein relativ neues Konzept. Dieses lässt sich auch daran erkennen, dass es in der Literatur nicht nur eine und auch keine konkrete Definition für diesen Begriff gibt. Zunächst werden an dieser Stelle leichte Abwandlungen, Synonyme beziehungsweise englischsprachige Ausdrücke für standortbasierte Dienste vorgestellt. Die am weitesten verbreiteten Ausdrücke sind standortbasierte oder standortbezogene Dienste, die auch im englischen Sprachraum als location based und location dependent services vertreten sind. Es werden jetzt zwei mögliche Definitionen für standortbasierte Dienste vorgestellt, um daraus eine auf diese Seminararbeit zugeschnittene zu erarbeiten.

> Unter Location Based Services (LBS) versteht man ortsgebundene Dienste eines Mobilfunkanbieters, die auf den Aufenthaltsort des Nutzers abgestimmte Informationen liefern, so über Einkauf- und Freizeitmöglichkeiten, über Sehenswürdigkeiten, regionale Angebote und den öffentlichen Verkehr, aber auch für die Standortbestimmung von Kindern durch ihre Eltern oder den Notruf für hilfsbedürftige Personen.[ITWi06]

> Location services can be defined as services that integrate a mobile device's location or position with other information so as to provide added value to the user. [ScVo04]

Aus diesen beiden Definitionen kann man schlussfolgern, dass ein standortbasierter Dienst in Abhängigkeit der Position des mobilen Endgerätes, das der Benutzer bei sich trägt, Dienste mit zusätzlichen Informationen, die sich auf seine Position beziehen, in Anspruch nehmen kann, oder diese ihm angeboten werden können. Es lassen sich bei dieser Art der Informationsversorgung immer drei Schritte erkennen. Zunächst muss der aktuelle Standort ermittelt werden, danach werden die standortspezifischen Informationen zusammengestellt und aufbereitet und zuletzt muss der Dienst dem Nutzer bereit gestellt werden. Weiterhin ist es auch noch denkbar, die Dienste abhängig von der Uhrzeit der Anforderung und dem jeweiligem Benutzer zu machen und dadurch eine weitere Einbindung in den Kontext zu erreichen.[Zobe01]

3 Anwendungsgebiete von standortbasierten Diensten

Es sind drei hauptsächliche Anwendungsgebiete für standortbasierte Dienste zu identifizieren. Diese sind der militärische und staatliche Sektor, der Katastrophen- und Notfall-Sektor und der kommerzielle Sektor. Der erste Sektor lässt sich leicht dadurch erklären, dass wie viele andere neue Technologien auch die standortbasierten Dienste aus der militärischen Nutzung entstanden sind und dort auch noch heute genutzt werden. Das Potenzial für die Nutzung im zweiten Sektor und insbesondere bei Notrufen bietet sich dadurch, dass es vielen Anrufern nicht möglich ist ihre exakte Position zu beschreiben.Um ihren Notruf mit ihrer Position an die nächste Einsatzzentrale weiterzugeben bietet sich ein standortbasierter Notruf-Dienst an. Doch die meiste Bewegung und Vielfalt an Anwendungen findet man naturgemäß im kommerziellen Bereich. Hierauf wird später noch einmal genauer eingegangen und auch einige Möglichkeiten genauer vorgestellt.

4 Einordnung in den Kontext

Um weiter zu präzisieren, was standortbasierte Dienste sind und wie sie sich in das Umfeld der neuen Technologien einfügen wird jetzt eine Abgrenzung zu ähnlichen Anwendungen vorgenommen. Alle Arten wirtschaftlicher Tätigkeiten, die auf Basis elektronischer Verbindungen getätigt werden lassen sich dem E-Business zuordnen. Der E-Commerce unterscheidet sich nur gering vom E-Business, denn hier werden die wirtschaftlichen Tätigkeiten als Handelstransaktionen präzisiert. Eine weitere Verfeinerung stellt das M-Business dar, das analog zum E-Business definiert ist aber jedoch nur auf mobilen Endgeräte aufbaut. Wiederum kann man hier auch äquivalent den M-Commerce eingrenzen [PiRW03]. Die standortbasierten Dienste sind somit als ein Bestandteil des M-Business und des M-Commerce zu betrachten, da hier ausschließlich mobile Endgeräte zum Einsatz kommen um dann einen standortbasierten Dienst anbieten zu können. Diese Abgrenzung ist in Abbildung 1 noch einmal graphisch aufbereitet[Zobe01].

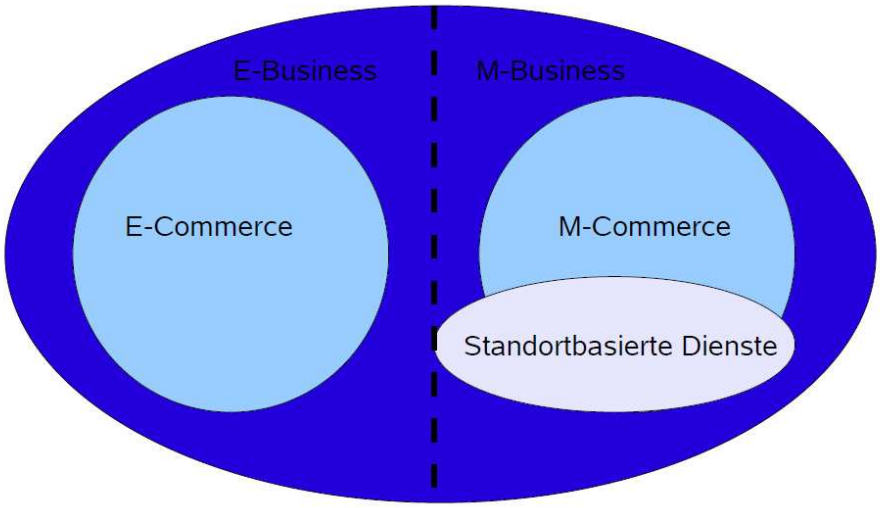


Abbildung 1: Einordnung in den Kontext.

5 Bestandteile von standortbasierten Diensten

Um standortbasierte Dienste den Nutzen zugänglich zu machen sind verschiedene Infrastrukturelemente notwendig. Diese fünf Basiskomponenten werden hier vorgestellt und in Abbildung 2 graphisch in Zusammenhang gebracht.[ScVo04]

1. *Mobiles Endgerät:* Dieses ist das Werkzeug für den Benutzer, damit er den gewünschten Dienst anfordern kann. Das Ergebnis kann dann in Form von Text, Bildern, Sprache und so weiter geliefert werden. Mögliche Geräte hierfür sind zum Beispiel Mobiltelefone, PDAs, Notebooks und ähnliche tragbare Geräte.

2. *Kommunikationsnetzwerk:* Das mobile Netzwerk ist die zweite Komponente, die benötigt wird um Benutzerdaten und Dienstanforderungen von dem mobilen Endgerät zum Dienstanbieter und die daraus resultierenden Informationen zurück zu übertragen. Meistens handelt es sich hierbei um einen Mobilfunkanbieter.

3. *Positionierungskomponente:* Um einen standortbasierten Dienst anbieten zu können muss natürlich die Position festgestellt werden. Dieses kann entweder automatisch über das mobile Kommunikationsnetzwerk oder mittels des Global Positioning System geschehen oder der Benutzer spezifiziert seinen Aufenthaltsort manuell. Die anderen möglichen Varianten, wie zum Beispiel WLAN oder Bluetooth, werden hier nicht betrachtet.

4. *Dienst- und Anwendungsanbieter:* Dieser Anbieter stellt eine Menge von Diensten und Anwendungen bereit und ist dafür verantwortlich, dass die Anfragen auch verarbeitet werden. Diese Aufgaben umfassen zum Beispiel das Festlegen einer Route, das Suchen von Geschäften abhängig von der Position, das Bereitstellen von spezifischen Informationen, für die sich der Benutzer interessiert und so weiter.

5. *Daten- und Inhaltsanbieter:* Der Dienst- und Anwendungsanbieter wird nicht alle Informationen, die er dem Benutzer zugänglich macht, selbst speichern und auf dem aktuellen Stand halten. Deshalb werden geographische Basisinformationen und Informationen zu bestimmten Orten naturgemäß bei den darauf spezialisierten Unternehmen nachgefragt.

Für eine genauere Erörterung des Ablaufes sei an diese Stelle auf ein späteres Kapitel 8 verwiesen.

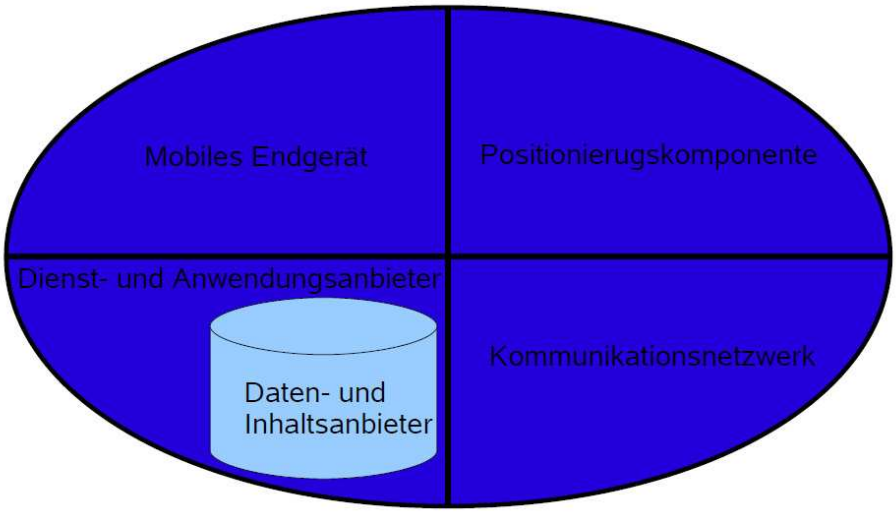


Abbildung 2: Zusammenhang der Infrastrukturelemente.

6 Anwendungstypen

Grundsätzlich kann man zwischen zwei verschiedenen Typen von standortbasierten Diensten unterscheiden. Diese Differenzierung ist abhängig davon, ob der Benutzer die Daten, die geliefert werden angefordert hat oder nicht. Diese zwei verschiedenen Möglichkeiten werden Pull- und Push-Dienste genannt.[Küpp05]

- *Pull-Dienste:* Bei dieser Variante werden dem Benutzer Informationen geliefert, die er explizit angefordert hat. Das ist vergleichbar mit dem Aufruf eine Website, indem man die Adresse eingibt. Eine weitere Unterscheidung kann hier in funktionale Dienste, das Rufen eines Taxis und informationelle Dienste, die Suche nach einen nahen Restaurant getroffen werden.
- *Push-Dienste:* Dieser Typ liefert dem Benutzer Informationen, die er nicht oder nur indirekt angefordert hat. Solche Dienste werden aktiviert, wenn zum Beispiel ein bestimmtes Gebiet betreten wird oder der Benutzer sich dort eine gewisse Zeitspanne aufhält. Ein mögliches Szenario für einen indirekten Dienst ist das Zusenden von Informationen über Veranstaltungen in der Stadt in der sich der Benutzer zur Zeit befindet. Ein nicht angeforderter Dienst könnte das Zusenden von Werbung in einem bestimmten Bereich eines Einkaufszentrum sein. Da bei dieser Art der Dienst meistens nicht in Verbindung mit früheren Anforderungen des Benutzers steht, ist es schwieriger diesen zu realisieren, denn die Wünsche und Bedürfnisse des Benutzers müssen vom System erkannt werden.

7 Techniken zur Lokalisierung

Wie schon zuvor im Kapitel 5 erwähnt kann man die Lokalisierungsverfahren in zwei verschiedene Verfahren unterteilen. Diese Differenzierung ist abhängig davon, ob für die Lokalisierung das mobile Kommunikationsnetzwerk oder das Global Positioning System verwendet wird. Weiterhin lässt sich auch noch eine Unterscheidung zwischen endgeräte- und netzwerkbasierten Verfahren unterscheiden. Es werden jetzt einige der am meisten verbreiteten Verfahren vorgestellt um diese dann zu vergleichen.

7.1 Netzbasierte Techniken

- *cell of origin (coo):* Bei dieser Technik wird die Position des Benutzers durch die Zelle des mobilen Kommunikationsnetzes, in der er sich befindet, festgelegt. Die Position wird durch den Mittelpunkt der jeweiligen Zelle gekennzeichnet. Allerdings haben diese Zellen unterschiedliche Größen und die genaue Position des Benutzers ist innerhalb dieser Zelle unbekannt. Dieses Verfahren ist in Abbildung 3 noch einmal graphisch verdeutlicht.

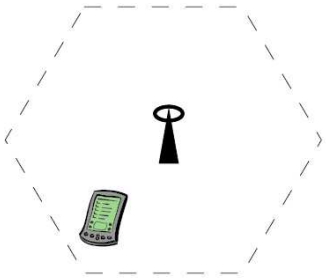


Abbildung 3: cell of origin.

- *angle of arrival (aoa):* Die Position des Benutzer wird durch eine Kreuzpeilung in einem Antennenfeld ermittelt. Hierbei wird die Richtung aus der das Signal das Endgerätes bezüglich einer Bezugsrichtung bei mindestens zwei verschiedenen Basisstationen gemessen. Da die Position der Basisstationen bekannt ist kann die Position des Endgerätes in Abängigkeit von Störeinflüssen recht einfach und genau berechnet werden. Verdeutlicht wird dieses Verfahren durch Abbildung 4.

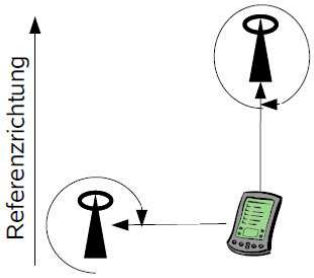


Abbildung 4: angle of arrival.

- *time difference of arrival (tdoa):* Bei diesem Verfahren wird zu Ermittlung der Position des Benutzers der Laufzeitunterschied des Signals vom Endgerät zu mindestens drei verschiedenen Basisstationen gemessen. Hierbei ist es jedoch essentiell, dass die Basisstationen über genau synchronisierte Uhrzeiten verfügen. Da auch hier die Positionen der Basisstationen bekannt sind, kann durch Berücksichtigung der Signalgeschwindigkeit und des Laufzeitunterschiedes die Position des Endgerätes berechnet werden. Die Abbildung 5 stellt dieses graphisch dar.

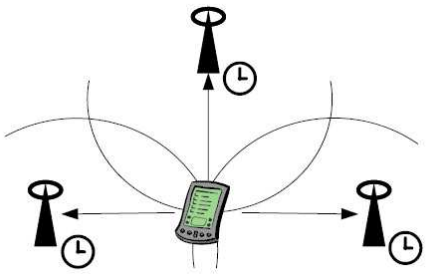


Abbildung 5: time difference of arrival.

- *time of arrival (toa):* Hierbei werden nicht die Laufzeitunterschiede des Signals zu den Basisstationen zur Berechnung genutzt, sondern die Laufzeiten des Signals selbst. Bei diesem Verfahren ist es jedoch nicht wichtig das mindestens drei Basisstationen wie sich überlappen, sondern es ist schon mit weniger möglich. Allerdings ist es dadurch auch nicht so genau wie das vorherige. Dieses Verfahren ist in Abbildung 6 graphisch verdeutlicht.

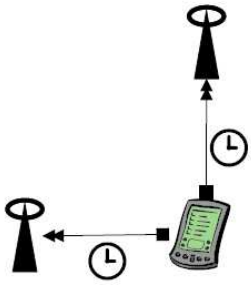


Abbildung 6: time of arrival.

7.2 Endgerätbasierte Techniken

- *enhanced oberved time difference (eotd):* Bei dieser Technik empfängt das Endgerät mindestens von drei Basisstationen, die über synchrone Uhrzeiten verfügen, so genannte „time-stamps“. Die Festellung der Laufzeitdifferenz erfolgt in dem Endgerät, das dann entweder selbst die Position ermittelt oder die Daten weiterleitet. Die Abbildung 7 schematisiert dieses Verfahren nochmal.

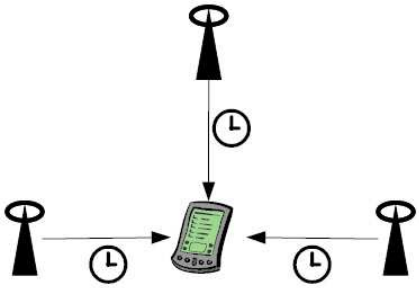


Abbildung 7: enhanched observed time difference.

- *(assistant) global positioning system ((a)gps):* Die Basis für diese Technik bilden 24 Satelliten in einer geostationären Umlaufbahn. Für die Positionsbestimmung muss immer Sichtkontakt zu mindestens drei Satelliten bestehen. Es wird immer zur gleichen Zeit ein Pseudo-Zufallscode vom Endgerät und den Satelliten erzeugt, die ihn an das Endgerät senden und durch die Ermittlung der Laufzeitdifferenz der einzelnen Signale kann dann die Position des Endgerätes bestimmt werden. Um eine Verbesserung zu erreichen werden zusätzlich noch stationäre Global Positioning System Empfänger aufgestellt, die dann zusätzlich zu den Satelliten den gleichen Pseudo-Zufallscode an das Endgerät schicken. Verdeutlicht wird dieses Verfahren in Abbildung 8.

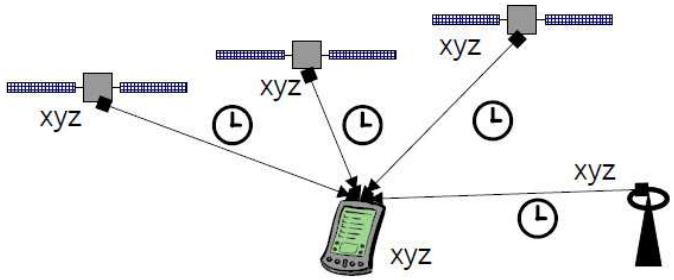


Abbildung 8: (assistant) global positioning system.

- *differential global positioning system (dgps):* Um Umwelteinflüsse auszugleichen, wird bei dieser Methode ein zusätzlicher Satellit benötigt um die Position von einer fest installierten Referenzstation zu ermitteln. Die Abweichung von der festgestellten Position der Referenzstation und der tatsächlichen geht dann in die Berechnung der Position des Endgerätes mit ein. Die Abbildung 9 veranschaulicht dieses Verfahren nocheinmal graphisch.

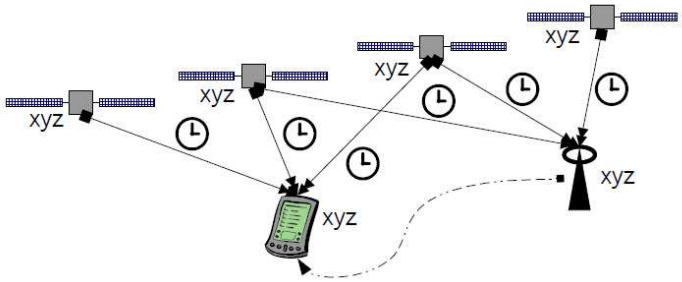


Abbildung 9: differential global positioning system.

	coo	aoa	tdoa	toa	eotd	gps	agps	dgps
Genauigkeit min - max	100m-30km	100-200m	ca. 70m	ca. 125m	50-125m	10-50m	10-50m	5m
Netzaufwand	gering	mittel	mittel	hoch	mittel	gering	mittel	hoch
Gerätemodifikation	nein	nein	nein	nein	ja	ja	ja	ja
Sichtkontakt	nein	nein	nein	nein	nein	ja	ja	ja

Tabelle 1: Vergleich der Lokalisierungsverfahren

Wie aus der Tabelle 1 leicht zu erkennen ist sind die Lösungen, die das Global Positioning System verwenden, nicht unbedingt immer anwendbar, denn sie sind zum Beispiel in einem Kaufhaus nur bedingt einsatzfähig, da der Sichtkontakt zu den Satelliten nicht vorhanden ist. Außerdem müssten die schon vorhandenen Endgeräte modifiziert werden um diesen Standard nutzen zu können. Je nachdem, welchen Dienst die Anwendung erbringen soll, kann ein anderes Verfahren, das ohne Endgerätmodifikation auskommt, ausgewählt werden.

8 Spezifische Anwendungen

Natürlich lassen sich auch hier, wie bei allen anderen elektronischen Handelsformen, die Anwendungen den typischen Bereichen zuordnen. Da die Möglichkeiten jedoch so vielfältig sind, werden nur einige repräsentative Anwendungen für die jeweilige Klasse genannt.

- *Business to Business:* Im Bereich der Unternehmen sind für verschiedene Teilbereiche - wie zum Beispiel die Logistik oder die Sicherheit - diverse Anwendungen denkbar. Zum einen ist hier das Flottenmanagement zu nennen, das durch den Einsatz von standortbasierten Diensten weitgehend automatisiert werden könnte. Weiterhin ist auch das Verfolgen von Warenlieferungen oder gestohlenen Gütern möglich.

- *Consumer to Consumer:* Die Benutzer haben auch vielfältige Möglichkeiten standortbasierte Dienste untereinander zu nutzen. Man denke alleine daran Blind Dates in Abhängigkeit der aktuellen Position möglich zu machen. Weiterhin ist es auch möglich die Standorte von Kindern herauszufinden oder Freunde zu treffen. Eine ganz andere Möglichkeit sind Spiele in der Realität, die die umgebende Umwelt mit einbeziehen.

- *Business to Consumer:* Die wahrscheinlich größte Masse der Anwendungen verteilt sich naturgemäß auf den Bereich zwischen Unternehmen und Kunden. Alleine die Möglichkeiten, die mit Navigationsdiensten oder Auskünften denkbar sind kennen fast keine Begrenzung. Weiterhin ist es auch möglich gezielt Werbung oder Sonderangebote zukommen zu lassen.

9 Typischer Ablauf

Um später aufzeigen zu können welche Datenschutzprobleme auftreten können, wird jetzt kurz der Ablauf eines typischen standortbasierten Dienstes dargestellt. Wie zuvor schon in Kapitel 2 angedeutet, werden mehrere Schritte durchlaufen. Zunächst muss der standortbasierte Dienst durch das Endgerät initialisiert werden. Es ist hierbei unerheblich ob es sich um einen Pull- oder Push-Dienst handelt. Wenn eine Funktion aktiviert ist muss die aktuelle Position des Benutzers mit einem der vorher beschriebenen Verfahren 7 netzwerk- oder endgerätebasiert ermittelt werden. Je nachdem wie dies geschehen ist wird entweder vom Lokalisierungsdienst oder vom Endgerät die Position über das Kommunikationsnetzwerk an ein Gateway übermittelt, das als Schnittstelle zum Internet fungiert. Der Dienst- und Anwendungsanieter erhält die Anfrage und leitet diese an die verschiedenen Daten- und Inhaltsanbieter weiter, um alle Informationen, die für den Benutzer relevant sind zu sammeln. Schließlich werden die Inhalte aufbereitet und dem Benutzer zurück übermittelt. Dieser ganze Ablauf ist in Abbildung 10 noch einmal schematisch dargestellt.

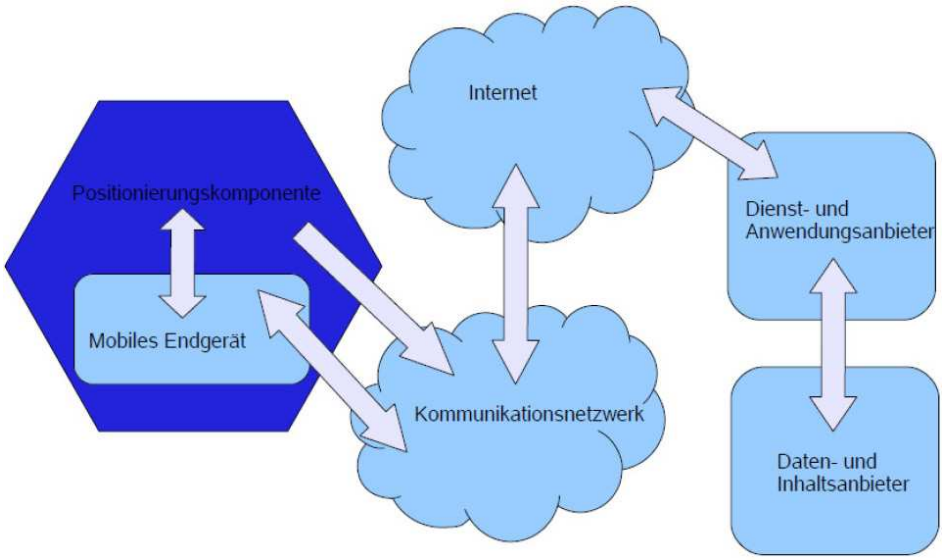


Abbildung 10: typischer Ablauf.

10 Datenschutzproblematik

In diesem Abschnitt soll dargestellt werden, was für die Anbieter von standortbasierten Diensten möglich ist. Zur Vereinfachung werden hier alle schon vorhandenen Einschränkungen nicht betrachtet. Das Szenario das hier entwickelt wird ist also der schlimmste Fall.

10.1 Datensammlung

Zunächst soll festgestellt werden, welche Daten überhaupt vom Anbieter gesammelt werden können. Mehrere persönliche Daten sind dem Anbieter bereits im Voraus bekannt. Dazu gehören zum Beispiel alle im Vertrag fixierten Punkte wie Name, Adresse, Kontoverbindung, Alter, Tarif und Vertragslaufzeit, um nur die häufigsten zu nennen. Wenn die angebotenen Dienste genutzt werden, fallen noch weitere Daten an. Einige leicht zu gewinnende sind zum Beispiel die Rechnungshöhe aufgegliedert in die einzelnen Leistungen, alle einzelnen genutzten Dienste und deren Dauer und das Zahlungsverhalten des Kunden. Ein wenig schwieriger ist es die Position des Benutzers zu verfolgen, da hier sehr viele Daten anfallen. Jedoch ist dieses auch ohne weiteres denkbar.

10.2 Datenerhebung

Da es den Anbietern von standortbasierten Diensten nicht möglich ist, alle anfallenden Daten auf unbegrenzte Zeit zu speichern, muss ein Weg gefunden werden, dem jeweiligem Benutzer ein Profil mit seinen Präferenzen zuzuordnen. Dabei sind zum Beispiel folgende Möglichkeiten denkbar. Daten die nur einmal auftauchen werden ohne weitere Komprimierung gespeichert. Sobald aber viele Daten auftauchen und sich auch oft wiederholen müssen verschiedene Kriterien festgelegt werden, um eine Auswahl über die zu speichernden Daten treffen zu können. Für das Beispiel der genutzten Dienste sind dieses eventuell nur Dienste, die mehrmals innerhalb eines bestimmten Zeitraums genutzt wurden. Es werden vielleicht auch nur Positionen des Benutzers gespeichert, die herausragend sind oder oft hintereinander oder zu bestimmten Uhrzeiten aufgesucht werden.

10.3 Datenverarbeitung

Aus der Vielzahl der gesammelten Daten allein lässt sich noch kein großer Nutzen ziehen. Zunächst müssen Verbindungen hergestellt werden um die Präferenzen der Benutzer zu erkennen. Zum Beispiel können Rückschlüsse von den genutzten Diensten an bestimmten Standorten gezogen werden. Wenn genug Verbindungen hergestellt werden konnten und die Datendichte groß genug ist, kann mit dem so erstellten Profil der Benutzer beeinflusst werden. Es können ihm zum Beispiel gezielt die Dienste angeboten werden, die er an bestimmten Standorten öfter genutzt hat, wenn er sich in der Nähe befindet. Weiterhin besteht die Möglichkeit, dem Benutzer standortbasierte Informationen zukommen zu lassen obwohl er diese gar nicht wünscht. Als ein Szenario ist hier Werbung für bestimmte Produkte des Anbieters zu nennen.

10.4 Datennutzung

Die Anbieter von standortbasierten Diensten können schon Nutzen aus den Daten der Benutzer ziehen, jedoch ist dies kein Vergleich zu anderen Unternehmen. Restaurants könnten zum Beispiel Benutzern von standortbasierten Diensten, die sich mittags oder abends in der Nähe aufhalten, Sonderangebote zukommen lassen. Internetversandhäuser könnten anhand der Aufenthaltsorte ihrer Kunden spezifische Kataloge für diese zusammenstellen. Alle Unternehmen könnten sich umgehend über ihre Mitarbeiter informieren, indem sie ihre Gewohnheiten betrachten, die Aufenthaltsorte analysieren, die genutzten Dienste anschauen und noch vieles mehr. Letztendlich ist jeder Benutzer für alle anderen zu lokalisieren und zu verfolgen. Es ist also eine totale Überwachung möglich.

11 Schutzmechanismen

Das in dem vorherigen Abschnitt 10 aufgezeigte Szenario ist jedoch den Benutzern nicht zuzumuten. Daher müssen eine Reihe von Schutzmaßnahmen ergriffen werden um ihre Privatsphäre zu schützen. Es sollen hier nun einige verschiedene juristische und technische Möglichkeiten angeführt und erläutert werden.

11.1 Juristische Schutzmechanismen

Zunächst werden hier einige juristische Aspekte betrachtet. Um schon einen deutlich besseren Privatspärenschutz für die Benutzer erreichen dürfen die Anbieter von standortbasierten Diensten nicht alle anfallenden Daten unbegrenzt speichern. Grundsätzlich dürfen nur Standortdaten erhoben werden, wenn das Einverständnis des Benutzers vorliegt.

- *Europäisches Recht* Die Richtlinie 2002/58/EG setzt den Maßstab für alle Mitgliedsstaaten der Europäischen Union. Hierbei handelt es sich hauptsächlich um die Verarbeitung personenbezogener Daten und den Schutz der Privatsphäre in der elektronischen Kommunikation. Daher wird sie auch als Datenschutzrichtlinie für elektronische Kommunikation bezeichnet. Dennoch ist dies kein direkt geltendes Recht und muss von jedem Mitgliedsstaat individuell in den nationalen Rechtsrahmen eingearbeitet werden.

- *Deutsches Recht* In diesem Rechtsrahmen muss zunächst unterschieden werden, ob es sich bei einem standortbasierten Dienst um einen Telekommunikationsdienst gemäß der Telekommunikationsdatenschutzverordnung (TDSV) oder einen Teledienst gemäß des Teledienstsdatenschutzgesetzes (TDDSG) handelt. Die erste regelt den Umgang mit

den persönlichen Daten, die bei dem Übermittlungsvorgang bei einer Telekommunikationsverbindung anfallen. Das zweite regelt bei der Inanspruchnahme von Telediensten die Nutzung der persönlichen Daten. Mediendienste im Sinne des Mediendienste-Staatsvertrag (MDStV) sind hier jedoch nicht zu betrachten, da sie nicht auf Individualkommunikation sondern auf Verteilangeboten an die Allgemeinheit beruhen. Um jedoch genau festlegen zu können ob es sich bei standortbasierten Diensten um Telekommunikationsdienste oder um Teledienste handelt müssen zwei verschiedene Vertragsfälle unterschieden werden. Diese sind abhängig davon zwischen welchen Parteien ein Vertrag zu Stande gekommen ist [Hell02].

  - *Standortdaten nach dem TDSV* In diesem Szenario bietet der Betreiber des mobilen Kommunikationsnetzwerkes dem Kunden selbst einen standortbasierten Dienst an und ist damit selbst der Vertragspartner. Daher handelt es sich hierbei um einen Telekommunikationsdienst. Betrachtet man diesen Fall genauer, dann ist die Standortkennung gemäß §6 TDSV historisch bedingt nur eine grobe Lokalisierung. Ob genauere Positionsbestimmungen mit den in Kapitel 7 genannten Lokalisierungsverfahren ebenfalls darunter fallen ist zweifelhaft. Daher ist für die Sammlung, Erhebung und Verarbeitung dieser Standortdaten das Einverständnis des Benutzers notwendig.
  - *Standortdaten nach dem TDDSG* In diesem Szenario bietet ein von dem Betreiber des mobilen Kommunikationsnetzwerkes unabhängiger Dritter den standortbasierten Dienst an. Dieser Vertragspartner unterliegt jedoch nicht, wie im vorherigen Szenario, dem Fernmeldegeheimnis, das im siebten Teil des TKG geregelt ist. Daher handelt es sich hierbei nicht um einen Telekommunikationsdienst, sondern um einen Teledienst. In diesem Fall gilt gleichermaßen für den Betreiber des mobilen Kommunikationsnetzwerkes und den Anbieter von standortbasierten Diensten, dass die Standortdaten des Benutzers nur verarbeitet werden dürfen, um den Dienst erfüllen zu können. Da dieses im Allgemeinen der Fall ist und der Benutzer auch sein Einverständnis dazu gegeben hat, ist dieses ohne weiteres möglich. Jedoch müssen diese Daten dann nach der vollständigen Erbringung des standortbasierten Dienstes wieder gelöscht werden (§96 TKG).

- *Recht auf informationelle Selbstbestimmung* Jeder Deutsche hat das Recht, die Veröffentlichung und Verwendung der über ihn gespeicherten Daten selbst zu bestimmen. Dieses Recht ist ein aus der allgemeinen Handlungsfreiheit (Art. 2 Abs. 1 GG) in Verbindung mit der Menschenwürde (Art. 1 Abs. 1 GG) abgeleitetes Recht. Die zu speichernden Verbindungsdaten dürften in diesen Bereich fallen, da sich die gesammelten Daten zum Beispiel durch Telefonnummer und Anschrift einer bestimmten Person zuordnen lassen. Problematisch ist hier, dass der einzelne Bürger über die gesammelten Daten keine Verfügungsmacht hat und auch keine Kontrolle besitzt, wofür seine Daten verwendet werden. Im so genannten „Volkszählungsurteil“ des Bundesverfassungsgerichts, dass als Meilenstein für den Datenschutz gilt, wurde 1983 schon die Gefahr erkannt, dass personenbezogene Daten zu einem teilweise oder weitgehend vollständigen Persönlichkeitsbild zusammengefügt werden, ohne dass der Betroffene dessen Richtigkeit und Verwendung zureichend kontrollieren kann. Es kann im Endeffekt sogar zu einer Falschdarstellung der jeweiligen Person kommen, da man zwischen dem Endgerät mit dem ein Dienst in Anspruch genommen wird und dem letztendlichen Benutzer unterscheiden muss. So kann die Mitnutzung eines Endgerätes nicht auf den Nutzer selbst, sondern lediglich auf die Person, die das Endgerät mitbenutzt hat, zurückgeführt werden.[Albe05]

- *Vorratsdatenspeicherung* Bereits 1996 gab es erste Diskussionen über Mindestspeicherfristen im Telekommunikationsgesetz, die aber abgelehnt wurden. Nach den Terroranschlägen 2001 wurde ein Gesetzesentwurf zur Verbesserung des strafrechtlichen Instru-

mentariums für die Bekämpfung des Terrorismus und der organisierten Kriminalität, der erste Vorschläge für eine Speicherung von Verbindungsdaten in der Telekommunikation und im Bereich der Teledienste beinhaltete, abgelehnt. Ein weiterer Vorschlag im Jahre 2002 bekräftigte die Forderung nach einer umfangreichen Vorratsdatenspeicherung, in dem das Abhören und Aufzeichnen des außerhalb einer Wohnung nichtöffentlich gesprochenen Wortes als entscheidendes Ermittlungsinstrumentarium benannt wurde. Allerdings wurde auch dieser Vorschlag wegen der Unverhältnismäßigkeit des Eingriffs abgelehnt. Parallel zu den Diskussionen über die Novellierung des Telekommunikationsgesetzes wurde 2002 die Telekommunikations-Überwachungsverordnung verabschiedet, die die bisherige Fernmeldeverkehr-Überwachungs-Verordnung ablöste. Ansonsten ist das Erheben und Verwerten von Standortdaten nur dann erlaubt, wenn diese Daten technisch erforderlich sind, um den standortbasierten Dienst zu ermöglichen oder abzurechnen. Auch danach wurde über eine eventuelle Einführung einer Vorratsdatenspeicherung im Rahmen der anstehenden Novellierung des Telekommunikationsgesetzes, dass letztendlich am 2004 verabschiedet wurde, diskutiert. Die Bundesregierung wies jedoch daraufhin, dass die Möglichkeit der Einführung einer Mindestspeicherungsfrist für Telekommunikationsverkehrsdaten Gegenstand ausführlicher Diskussionen im Deutschen Bundestag und im Bundesrat gewesen ist und das – in Übereinstimmung mit der Auffassung der Bundesregierung – eine Einigung des Inhalts zustande kam, eine solche Verpflichtung im Rahmen der Novellierung des Telekommunikationsgesetzes nicht einzuführen. Jedoch besteht nach der Richtlinie 2002/58/EG eine Umsetzungsfrist bis zum Jahre 2007. Da die Richtlinie nicht ausreichend harmonisiert ist, liegt es an den Mitgliedsstaaten für sich eine geeignete Speicherfrist zwischen sechs Monaten und zwei Jahren festzulegen. Ebenso ist es Definitionssache der Länder, was als schwere Straftat zu verstehen ist und somit einen Zugriff auf die Daten rechtfertigt. Als eine Alternative zu der Vorratsdatenspeicherung wird das amerikanische Quick-Freeze-Verfahren angesehen.[Schü05]

11.2 Technische Schutzmechanismen

An dieser Stelle werden einige technische Möglichkeiten aufgezeigt den Datenschutz bei standortbasierten Diensten zu gewährleisten. Diese Verfahren bieten eine größere Sicherheit für den Benutzer, da es dem Anbieter der Dienste nicht möglich ist die zugehörigem Standortinformationen weitergehend zu nutzen. Die folgenden Techniken werden in der Reihenfolge der Komplexität vorgestellt.

- *Abschalten des Endgerätes* Die wohl einfachste und effektivste Möglichkeit der ständigen Ortung durch den Anbieter von standortbasierten Diensten zu entziehen, ist es das Endgerät nur für die tatsächliche Benutzung einzuschalten. Zu diesem Zeitpunkt erhält der Anbieter allerdings die Information über die Position und den gewünschten Dienst. Genau so wenig effektiv ist es, das Endgerät zwar eingeschaltet zu haben aber auf die Nutzung der standortbasierten Dienste zu verzichten, da trotzdem eine Bestimmung des Standortes möglich ist.

- *Abschalten der standortbasierten Dienste* Im Gegensatz zu der Nichtnutzung der standortbasierten Dienste ist das komplette Abschalten sehr sinnvoll. Wenn der Anbieter von standortbasierten Diensten nicht die Möglichkeit hat den Standort zu ermitteln, kann er auch keinen Dienst erbringen und keine sensiblen persönlichen Daten speichern. Ob dieses jedoch möglich ist hängt von dem jeweiligem Endgerät und dem verwendeten Lokalisierungsverfahren ab. Betrachtet man zum Beispiel UMTS oder GSM so ist eine Standortbestimmung schon alleine technisch unumgänglich.

11.2.1 privacy enhancing technologies

Unter diesem Begriff versteht man Technologien, die dazu beitragen die Privatsphäre des Benutzers zu schützen. Dieses können dann Bestandteile des Endgerätes sein, Protokolle für den Ablauf der Kommunikation oder bestimmte Regeln, an die sich alle Beteiligten bei standortbasierten Diensten zu halten haben.

- *Pseudonyme* Bei der Benutzung von standortbasierten Diensten sollte darauf geachtet werden, dass die Standortdaten sich nicht einem bestimmten Benutzer zuordnen lassen. Weiterhin sollte es auch nicht möglich sein aus mehreren zusammenhängenden Daten die Indentität zu erraten. Daher werden für die Benutzung von standortbasierten Diensten Pseudonyme verwendet, die der Benutzer vor jeder Anwendung auswählen kann. Ein wenig fortgeschrittener ist es, wenn Richtlinien eingesetzt werden um diesen Pseudonymwechsel zu vollziehen. Dabei kann es jedoch sein, das dieser Algorithmus durchschaut wird und anhand der Standortdaten die verschieden Pseudonyme miteinander in Verbindung gebracht werden können. In der Abbildung 11 ist dieses noch einmal graphisch dargestellt.

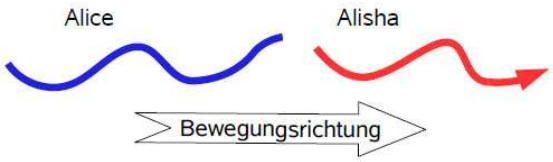


Abbildung 11: Pseudonymwechsel.

- *Camouflage* Eine weitere Möglichkeit um das Erstellen von Positionsnetzwerken zu verhindern ist es falsche Informationen zu liefen. hier kann zwischen zwei verschiedenen Verfahren unterschieden werden, die jedoch kombinierbar sind.

  1. *zeitliches Verstecken* Bei dieser Technik werden die Zeitintervalle in denen es möglich ist die Position des Endgerätes festzustellen geregelt. Es ist also nicht mehr möglich die ganze Zeit das Endgerät zu verfolgen und es kann somit auch kein genaues Netzwerk über die Bewegungen des Benutzers erstellt werden. Dieses ist noch einmal in der Abbildung 12 als Graph dargestellt.

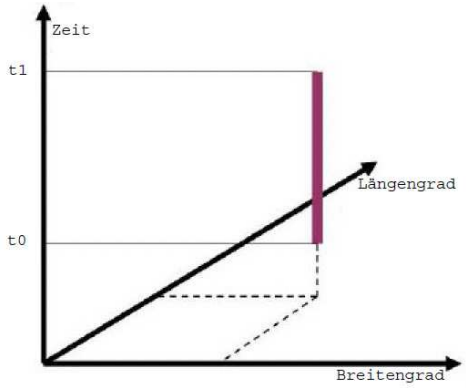


Abbildung 12: zeitliches Verstecken.

2. *räumliches Verstecken* Bei dieser Technik wird die Genauigkeit mit der das Endgerät zu orten ist, festgelegt. Dieses ist natürlich nur in dem Rahmen möglich wie es die gewünschte Anwendung zulässt. Auch hier ist es nicht mehr möglich ein genaues Netzwerk der Bewegungen des Benutzers aufzustellen, da die Standortdaten einfach zu ungenau sind. Dieses ist noch einmal in der Abbildung 13 als Graph dargestellt.

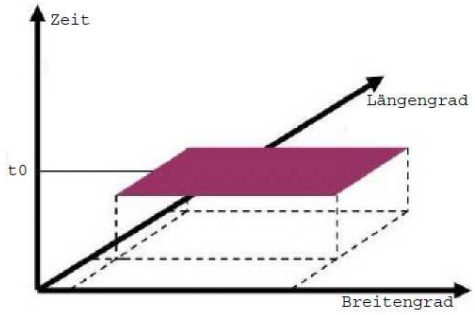


Abbildung 13: räumliches Verstecken.

- *MIX-Zoning* Um das Problem, das bei einem Pseudonymwechsels autreten kann besser zu lösen und damit dann sicherzustellen, dass er von einem externen Betrachter nicht nachvollzogen werden kann, wird diese Technik eingesetzt. In Verbindung mit einem Camouflageverfahren wird hier der Pseudonymwechsel von mehreren Benutzern zur gleichen Zeit und in der gleichen Standortumgebung vollzogen. Damit ist es für die externen Betrachter nicht mehr möglich die Standortdaten eindeutig zuzuordnen. Da es jedoch möglich ist, dass sich nicht genug Benutzer dort befinden, werden virtuelle Pseudonyme zur Hilfe genommen. Wenn ein Benutzer versucht sein Pseudonym zu wechseln wird sichergestellt, dass seine Alternative oder ein virtuelles Pseudonym seinen Weg kreuzt. Das Problem hierbei besteht darin für alle virtuellen Pseudonyme ein realistisches Bewegungsmuster zu erzeugen, das den Benutzer nicht kompromittiert. In der Abbildung 14 ist noch einmal graphisch dargestellt, wo die Pseudonymwechsel vollzogen werden und die Abbildung 15 stellt die Sicht eines externen Betrachters dar.

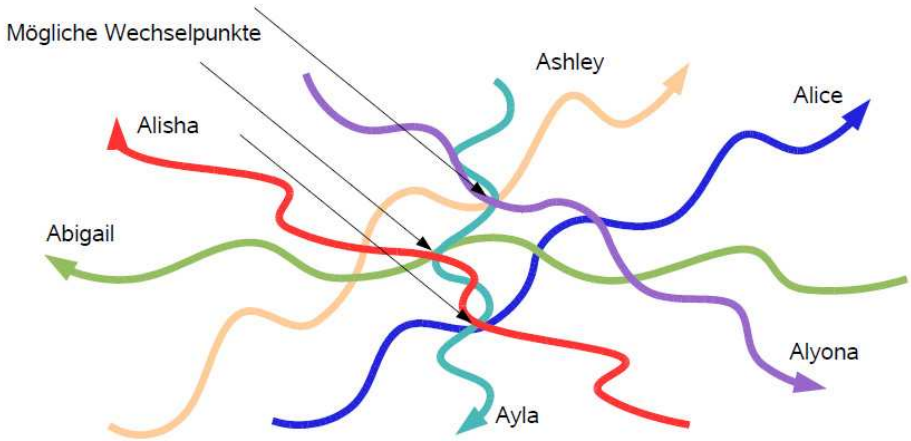


Abbildung 14: Pseudonymwechsel.

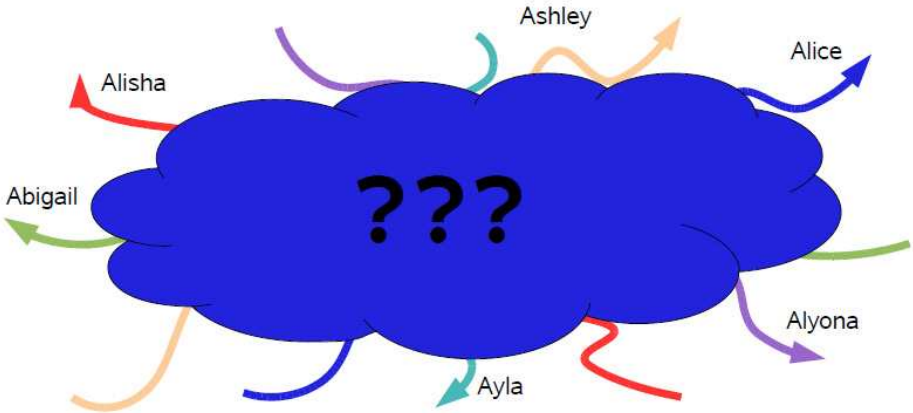


Abbildung 15: externe Sicht.

11.2.2 PRIME-Project

Nicht nur im Bereich der standortbasierten Dienste sondern auch für alles anderen digitalen Medien ist es wichtig die Privatsphäre der Benutzer zu schützen. Dieses Ziel hat sich das Projekt Privacy and Identity Management for Europe (PRIME) gesetzt. Um die Privatsphäre des Benutzers von standortbasierten Diensten zu schützen ist es sinnvoll den Anbietern solcher Dienste nicht direkt Zugang zu den sensiblen Standortdaten zu gewähren. Um dieses Ziel zu erreichen, kann eine weitere Entität, der Intermediär, eingebunden werden. Dieser erhält dann die Standortdaten des Benutzers und gibt diese innerhalb von vorher festgelegten Parametern an den Dienstanbieter weiter. Alle diese Transaktionen werden protokolliert und somit ist es ist für den Benutzer jederzeit möglich nachzuvollziehen welche Daten von ihm bekannt sind. Eine weitere mögliche Funktion ist die Beglaubigung von bestimmten Angaben des Benutzers. Bei dieser Architektur ist es allerdings sehr problematisch, dass dem Intermediär alle Standortdaten und angeforderten Dienste bekannt sind. Zwar ist es den Anbietern von standortbasierten Diensten nicht mehr direkt möglich die Standortdaten zu nutzen, aber ob sich der Intermediär an seine Verschwiegenheit hält, ist eine völlig andere Frage. Aus diesem Gründen kann gesagt werden, dass das PRIME-Project nicht unbedingt die beste Lösung ist, da hier die Teilnehmer nicht uneingeschränkt über ihrer Standortdaten verfügen können.[Fede01]

12 Ausblick

Betrachtet man heute die immer schneller voranschreitende Entwicklung der standortbasierten Dienste, dann ist es zu erwarten, dass in naher Zukunft diese Anwendungen zu dem täglichem Leben dazugehören. In Verbindung mit weiteren neuen Technologien, Weiterentwicklungen in der Lokalisierung, den Endgeräten und der Datenübertragung sind auch noch weitere Anwendungen möglich. Allerdings birgt dieses Szenario, das eng mit dem Ubiquious Computing verknüpft ist, gerade für Datenschützer extreme Gefahren. Viele Benutzer sehen auch nicht die Gefahren für ihre Privatsphäre, die durch die Sammlung ihrer Standortdaten entstehen.

13 Fazit

Betrachtet man abschließend das Thema des Datenschutzes für standortbasierte Dienste, so ist es auffällig, dass wie bei allen neuen Technologien sehr viele Gefahren vorhanden sind. Um diese Gefahren zu beseitigen und das Risiko der Benutzer zu minimieren stellen sich sowohl Politik und Wirtschaft viele Herausforderungen. Zur Zeit befinden wir uns auf dem richtigen Weg um den Datenschutz der Benutzer auch für zukünftige Anwendungsszenarien zu gewährleisten.

Literatur

[Albe05] Marion Albers. *Informationelle Selbstbestimmung*. Nomos-Verl.-Ges. 1. Auflage, 2005.

[Fede01] Hannes Federrath. *Designing privacy enhancing technologies*. Springer. 2001.

[Hell02] Stafanie Hellmich. *Location Based Services - Datenschutzrechtliche Anforderungen*. Multimedia und Recht. 2002.

[ITWi06] ITWissen. location based service. Website, 5 2006. http://www.itwissen.info/definition/lexikon/.

[Küpp05] Axel Küpper. *Location-based services*. Wiley. 2005.

[PiRW03] Arnold Picot, Ralf Reichwald und Rolf T. Wigand. *Die grenzenlose Unternehmung*. Gabler. 5. Auflage, 2003.

[Schü05] Raimund Schütz. *Kommunikationsrecht*. Beck. 2005.

[ScVo04] Jochen Schiller und Agnès Voisard. *Location-based services*. Elsevier. 2004.

[Zobe01] Jörg Zobel. *Mobile-Business und M-Commerce*. Hanser. 2001.

Abbildungsverzeichnis

Tabellenverzeichnis

Datenschutz und WWW

Marcel Czink

Kurzfassung

Diese Arbeit befasst sich mit aktuellen Problemen des Datenschutzes im WWW. Der Datenschutz ist in einer Vielzahl einzelner Gesetze geregelt – dem BDSG, TDDSG, MDStV u.a. Um etwas Licht in das Dunkel zu bringen, wird auf einzelne Anwendungen des WWW und ihr Verhältnis zum Datenschutzrecht eingegangen. Am Anfang werden Grundbegriffe des Datenschutzrechts erklärt und die relevanten Gesetze vorgestellt. Den ersten Schwerpunkt bildet „Cookies und Datenschutz“, anschließend wird die datenschutzkonforme Gestaltung von Webseiten und deren Inhalte genauer betrachtet. Welche Daten von einem Diensteanbieter gespeichert werden, behandelt dann der folgende Teil. Zum Schluss wird noch auf Datenschutzkonzepte und deren technologischen Grundlagen eingegangen. Es wird vor allem P3P betrachtet. Zum Schluss werden noch Ausblicke auf zukünftige Gesetze bzw. derer Umsetzung gezeigt.

1 Einleitung

1.1 Datenschutz / BDSG

Das Bundesdatenschutzgesetz (BDSG) ist die deutsche Umsetzung der EU-Datenschutzrichtlinie 95/46/EG. Das Gesetz trat am 14. Januar 2003 in Kraft. Das Gesetz gliedert sich in einen *Allgemeinen Teil*, einen Teil für die *Datenverarbeitung öffentlicher Stellen*, und in einen Teil der *Datenverarbeitung nicht-öffentlicher Stellen*. Im weiteren ist vor allem die Datenverarbeitung nicht-öffentlicher Stellen von Belang. Es gibt noch eine Vielzahl anderer Gesetze, die den Datenschutz spezifisch für bestimmte Bereiche konkretisieren, man spricht hier von der Subsidarität des BDSG, d.h. diese Spezialgesetze sind vorrangig anzuwenden. Beispiele, die uns im folgenden noch begegnen werden, sind das Telekommunikationsgesetz (TKG), das Teledienstedatenschutzgesetz (TDDSG) und der Mediendienste-Staatsvertrag (MDStV). Das BDSG kann somit als ein „Auffanggesetz“ gesehen werden [WoGe05, 3.1.1]. Grundsätzliche zum Datenschutz, dass auch in die EU-Datenschutzrichtlinie und damit in das BDSG einfloss, wurde vom BVerfG in der ständigen Rechtssprechung entwickelt. Hervorzuheben ist das in der Verfassung verankerte *Recht auf informationelle Selbstbestimmung (Art. 2 Abs. 1 i.V.m. Art. 1 Abs. 1 GG)*. Das BDSG beinhaltet ein Verbotsprinzip mit Erlaubnisvorbehalt, d.h. grundsätzlich sind alle Formen der Erhebung, Verarbeitung und Nutzung *personenbezogener Daten* nicht zulässig, mit einigen Ausnahmen (§ 4 Abs. 1 BDSG):

- Einwilligung des Betroffenen
- Erlaubnis durch das BDSG
- Erlaubnis durch andere Gesetze und Rechtsvorschriften

1.1.1 personenbezogene Daten

Damit das BDSG und damit Datenschutzrecht als solches einschlägig ist, müssen personenbezogene Daten vorliegen. Liegen keine personenbezogene Daten vor, fällt auch der Umgang mit diesen Daten nicht unter das Datenschutzrecht. Der Begriff der „personenbezogenen Daten“ wird in § 3 Abs. 1 BDSG definiert als „Einzelangaben über *persönliche* oder sachliche Verhältnisse einer *bestimmten* oder *bestimmbaren* natürlichen Person“ . Voraussetzung ist also, dass Daten einer Person zugeordnet werden können. Bestimmbar bedeutet, dass Informationen mit Zusatzwissen einer Person zugeordnet werden können [WoGe05, 3.3.2.2]. Ab wann eine „Einzelangabe“ eine datenschutzrechtliche Relevanz hat, beantwortet das BVerfG mit der Formel: „Es gibt kein belangloses Datum “ [Bund83]. Es geben praktisch alle Daten Auskunft über den Betroffenen. Nur wenn sich nicht auf die Person schließen lässt, hat das bloße Datum keinen datenschutzrechtlichen Wert. Um die Bestimmbarkeit einer Person auszuschließen, gibt es zwei Möglichkeiten, die in § 3 Abs. 6 bzw. Abs. 6a BDSG definiert werden:

- *Anonymisieren* § 3 Abs. 6 BDSG: Bei erfolgter Anonymisierung kann der Personenbezug nicht mehr hergestellt werden und das BDSG ist somit nicht einschlägig. Kann der Personenbezug nur mit unverhältnismäßigen Aufwand wieder hergestellt werden, so ist es streitig, ob es dem Anwendungsbereich des BDSG unterliegt [WoGe05, 3.3.2.4].
- *Pseudonymisieren* § 3 Abs. 6a BDSG: Der Name und andere Identifikationsmerkmale werden durch ein Pseudonym ersetzt. Der Personenbezug ist dauerhaft nicht ausgeschlossen und auch nicht gewollt. Das Pseudonym ist grundsätzlich nicht mehr personenbezogen, wenn es vom Nutzer selber generiert wurde [WoGe05, 3.3.2.3].

Mit beiden Techniken kann man somit die Anwendung des BDSG von Anfang an ausschließen. Das BDSG unterscheidet zwischen der Erhebung, Verarbeitung und dem Nutzen personenbezogener Daten. Das *Erheben* ist das Beschaffen von Daten über den Betroffenen (§ 3 Abs. 3 BDSG); die Daten sollten wenn möglich direkt beim Betroffenen erhoben werden, d.h. nicht bei Dritten erhoben werden, dies entspricht dem geforderten „Direkterhebungsgrundsatz“ und ist in § 4 BDSG geregelt. Das *Verarbeiten* ist in § 3 Abs. 4 Nr. 1 - Nr. 5 näher bestimmt. So fallen das Speichern, Verändern, Übermitteln, Sperren und Löschen personenbezogener Daten unter diesen Begriff. Wichtig ist vor allem das Übermitteln, das ein Bekanntgeben gespeicherter oder durch Datenverarbeitung gewonnener personenbezogener Daten an Dritte darstellt (§ 3 Abs. 4 Nr. 3). *Nutzen* personenbezogener Daten ist jede andere Verwendung von Daten, und stellt damit den sog. Auffangtatbestand dar (§ 3 Abs. 5 BDSG). *Nutzen* personenbezogener Daten ist z.B. das Veröffentlichen von Daten, allerdings nur, wenn das Veröffentlichen keine Übermittlung i.S.d. § 3 Abs. 3 BDSG ist [WoGe05, 3.4.4]. Die Abgrenzung zwischen Übermittlung und Veröffentlichung ist in vielen Fällen schwierig, jedoch ist das Veröffentlichen von Daten auf Websites – die Zugänglichmachung von Daten – keine Übermittlung und fällt somit unter den Tatbestand des *Nutzen*. Der EuGH erklärt in einem Urteil vom 6. November 2003 bezüglich der Richtlinie 95/46/EG: „Die Handlung, die darin besteht, auf einer Internetseite auf verschiedene Personen hinzuweisen und diese entweder durch ihren Namen oder auf andere Weise, etwa durch Angabe ihrer Telefonnummer oder durch Informationen über ihr Arbeitsverhältnis oder ihre Freizeitbeschäftigungen, erkennbar zu machen, stellt eine ganz oder teilweise automatisierte Verarbeitung personenbezogener Daten“ dar [Geri03]. Da die Veröffentlichung personenbezogener Daten ohne einen besonderen Schutz regelmäßig weltweit abrufbar ist, könnte eine Übermittlung personenbezogener Daten in ein Drittland i.S.d. Art. 25 95/46/EG vorliegen. Wäre jede Veröffentlichung gleichzeitig eine Übermittlung in Drittländer, so müsste jedes Land, von dem aus die Website abgerufen werden kann, ein angemessenes Schutzniveau aufweisen (Art.25 Abs.1 95/46/EG). Der EuGH

sieht in seinem Urteil in der Zugänglichmachung keine Übermittlung von Daten in ein Drittland, wenn der Ort der Zugänglichmachung sich in einem Mitgliedsstaat befindet [Geri03]. Die Vorschriften werden gemäß Art. 4 Abs. 1 lit. a (95/46/EG) vom jeweiligen Mitgliedsstaat angewandt, in Deutschland ist somit das BDSG einschlägig.

Im weiteren wird das Nutzen und Verarbeiten personenbezogener Daten der Einfachheit halber als *Verwenden* bezeichnet.

1.1.2 Einwilligung

Mit der Einwilligung des Betroffenen ist die Verarbeitung, Erhebung und Nutzung von Daten zu seiner Person zulässig. Die Einwilligung ist im § 4a BDSG geregelt und muss folgende Bedingungen für ihre Wirksamkeit erfüllen:

- *die freie Entscheidung des Betroffenen (§ 4a Abs. 1 Satz 1 BDSG)* „Die freie Entscheidung des Betroffenen verträgt insbesondere keine gezielte Beeinflussung durch wirtschaftliche Begünstigungen (Koppelungsgeschäft) bzw. Drohung der Leistungsverweigerung.“ [WoGe05, 4.2.1]

- *informierte Entscheidung des Betroffenen (§ 4a Abs. 1 Satz 2 und Satz 3 BDSG)* Der Betroffene muss über den Zweck der Erhebung und über die anschließende Verwendung der Daten im Vorab informiert werden. Eine Blankovollmacht, d.h. eine Einverständniserklärung über die Erhebung, Verarbeitung und Nutzung der Daten zu jedem erdenklichen oder nicht hinreichend spezifizierten Zweck scheidet mangels hinreichender Bestimmtheit aus.

- *Form der Einwilligung* Die Form der Einwilligung bedarf grundsätzlich der Schriftform (§ 4a Abs. 1 Satz 3). Eine andere Art der Einwilligung ist möglich insbesondere wenn Eilbedürftigkeit oder die Gefahr einer Zweckverfehlung besteht [WoGe05, 4.2.1.1]. Die elektronische Form der Einwilligung ist unter Bezug auf § 126 Abs. 3 BGB möglich. [Simi03, §4a Rn.38]

1.1.3 TDG und MDStV / TDDSG und BDSG

Die im Teledienstegesetz (TDG) geregelten Dienste dienen vor allem der Individualkommunikation (§ 2 Abs. 2 TDG). Wichtigster Vertreter hier sind „Angebote zur Information und Kommunikation, soweit die redaktionelle Gestaltung zur Meinungsbildung für die Allgemeinheit nicht im Vordergrund steht“ (§ 2 Abs. 2 Nr. 2) Darunter fallen namentlich Datendienste wie Verkehrs-, Wetter-, Umwelt-, und Börsendaten, hierzu zählen aber auch Einzelwerbeangebote über Waren und Dienstleistungen sowie sonstige Angebote und Anzeigen. Mediendienste hingegen sind an die „Allgemeinheit gerichteten Informations- und Kommunikationdienste“ (§2 Abs. 1 Satz 1 MDStV); auf sie ist der MDStV anzuwenden. An die „Allgemeinheit gerichtet“ bedeutet, dass die redaktionelle Gestaltung zur Meinungsbildung für die Allgemeinheit im Vordergrund steht (§ 2 Abs. 4 Nr. 3 TDG). Dazu gehören unter anderem auch Online-Zeitungen – also die elektronische Presse – sowie redaktionell aufbereitete Homepages. Das Teledienstedatenschutzgesetz (TDDSG) regelt den Datenschutz bei den Telediensten. Wie bereits erwähnt, gilt das BDSG subsidiär; es enthält ergänzende Regelungen. Grundsätzlich gehen dem BDSG auch die einzelnen Landesdatenschutzgesetze vor. Da aber eine große Ähnlichkeit zwischen den LDSGs und dem BDSG besteht, bezieht sich diese Arbeit nur auf das BDSG.

1.1.4 3-Schichten-Modell

Das Verhältnis der Gesetze untereinander ist leider etwas kompliziert, da es kein einheitliches Datenschutzgesetz gibt. Der Einfachheit halber kann man ein Schichtenmodell erstellen, indem man jeder technischen Gruppe von Kommunikationsvorgängen ein Gesetz zuordnet [Schl04, S.731]. Auf Schicht 1 ist die Kommunikation rein technischer Natur. Es entstehen TK-Nutzungsdaten und TK-Bestandsdaten, wie z.B. IP-Adressen oder Internet-Telefonnummer. Für dieser der Telekommunikationsanbietern vorbehaltenen Schicht ist dann auch das TKG einschlägig. Die nächste Schicht – Schicht 2 – stellt Internetdienste zur Verfügung. Hier kommunizieren Personen über das Internet miteinander. Das Anbieten von Informationsdiensten, wie Online-Zeitungen, aber auch interaktive Dienste wie Online-Banking, gehören zur Schicht 2. Für die Schicht 2 sind das TDG, TDDSG und der MDStV einschlägig. Daten die erhoben werden sind Nutzungsdaten, Bestandsdaten, Abrechnungsdaten. In der Schicht 3 stehen hauptsächlich die Inhalte der Kommunikation im Vordergrund – die Inhaltsdaten. Für die Schicht 3 ist das BDSG einschlägig. Man könnt auch sagen, sämtliche Daten, die nicht der Schicht 1 und Schicht 2 zugeordnet werden können, fallen unter das BDSG.

1.2 Nutzungsarten

Es gibt drei verschiedene Nutzungsarten des Internets, die auch verschiedene datenschutzrechtliche Probleme aufwerfen. [uMdK00, 2.1]

- *Reine Nutzung von Informationen durch einen Internetzugang*
- *Bereitstellung von Informationen*
- *Bereitstellung von Informationen und Interaktion mit Personen*

2 Datenschutz und WWW

Es werden im folgenden Abschnitt WWW-Technologien im Lichte des Datenschutzes betrachtet. Cookies und Nutzerprofile sind hier die wichtigsten Technologien. Auch Webseiten müssen die Anforderungen des Datenschutzes erfüllen. Es wird die Zulässigkeit von Inhalts-, Bestands- und Nutzungsdaten geprüft.

2.1 Cookies und Nutzerprofile

2.1.1 Was sind Cookies?

Cookies sind kleine Dateien, „die von einem Web-Server erzeugt, an einen Web-Browser, der mit diesem Server eine Verbindung aufgebaut hat, gesendet und auf dem Rechner des Nutzers abgelegt werden“ [Scha02a, Rn.178]. Cookies sind eine Entwicklung der Firma Netscape und wurden ursprünglich für den Netscape Navigator benutzt. Microsoft und andere Hersteller von Browsersoftware portierten Cookies, so dass sie heute eine enorme Verbreitung im Internet haben. Mit einem Cookie können mehrere Daten gespeichert werden: Zum Einen werden im Cookie selber Werte wie Gültigkeitsdauer, Gültigkeits-Pfad, Domäne oder Version gespeichert [KrMo00] die rein technischer Natur sind. Es kann aber auch eine einzigartige Identifikationsnummer vergeben werden. Auf dem Server können nun Informationen wie z.B. abgerufene Websites, Login-Name, Passwörter, Beginn und Ende einer Session und vieles andere mehr gespeichert und dem Nutzer über diese Identifikationsnummer zugeordnet werden.

2.1.2 Die Gefahr bei Cookies

Damit besteht die Gefahr, dass das Verhalten der Nutzer studiert werden kann und Nutzerprofile angelegt werden können. Sobald der Nutzer unmittelbar mit Hilfe der Cookies identifiziert werden kann, liegen personenbezogene Daten vor. Meldet man sich zum Beispiel bei Amazon an, so wird ein Cookie gesetzt. Beim nächsten Besuch kann man nun über die Funktion „Mit 1-Click kaufen“ fast ganz ohne eine zusätzliche Bestätigung einkaufen, da man über das Cookie identifiziert wurde. Unangenehm kann es für den Benutzer werden, wenn seine besuchten Webseiten ebenso gespeichert werden wie seine Anmeldeten. So könnte man ein Profil des Nutzers erstellen und z.B. Werbung auf ihn abstimmen. Das Zusammenführen der Identifikationsnummer mit zusätzlichen Informationen über den Nutzer birgt die Gefahr des „gläsernen Anwenders“ .

2.1.3 Zulässigkeit bei Cookies

„Die Verwendung von Cookies ist – abhängig von dem jeweils verwendeten Konzept – datenschutzrechtlich relevant“ [IhDi03, S.351]. Werden mit Hilfe von Cookies personenbezogene Daten erfasst, muss eine Reihe von Formalien und Bedingungen beachtet werden. Umgekehrt gilt natürlich, dass Cookies datenschutzrechtlich irrelevant sind, wenn sie keinen Personenbezug herstellen können. Bei Personenbezug gilt auch hier, dass entweder eine Einwilligung des Nutzers oder eine Erlaubnis durch Gesetz vorliegen muss (§ 3 Abs. 1 TDDSG). Die Einwilligung sollte explizit vom Nutzer gegeben werden, die bloße Voreinstellung des Browsers reicht dabei nicht aus [IhDi03, S.351]. Der Server hat den Nutzer vor Beginn der Datenverarbeitung über Art, Umfang, Zweck und Ort der Verarbeitung zu informieren (§ 4 Abs. 1 TDDSG). Diese Unterrichtung muss vor Beginn der Übertragung erfolgen, d.h. bevor das Cookie gesetzt wird [IhDi03, S.351]. Zusätzlich muss der Inhalt der Unterrichtung für den Nutzer jederzeit abrufbar sein (§ 4 Abs. 2 Nr. 3 TDDSG). Der Nutzer hat ein Recht auf Widerruf seiner Einwilligung, auf dieses Recht muss der Betreiber des Servers hinweisen (§ 4 Abs. 3 TDDSG). Die Einwilligung an sich muss durch eine bewusste, eindeutige Handlung des Nutzers erfolgen (§ 4 Abs. 2 TDDSG), ein abzuhakendes Kontrollkästchen reicht hierbei aus [IhDi03, S.351]. Der Diensteanbieter hat noch eine Vielzahl weitere technischer Maßnahmen zu realisieren, wie z.B. die Möglichkeit jederzeitigen Abbrechens der Verbindung durch den Nutzer, Löschung und Sperrung der Daten, Trennung der Abrechnungsdaten, usw (§ 4 Abs. 5 TDDSG, § 18 Abs. 4 MDStV).

2.1.4 Zulässigkeit bei Nutzerprofilen

An Nutzerprofile werden noch strengere Maßstäbe gesetzt. Ein Nutzerprofil ist eine Datensammlung von Informationen über einen Nutzer, die das Surf- und Kaufverhalten des Nutzers aufzeichnen um mit statistischen und mathematischen Methoden ein möglichst realitätsnahes Bild der Persönlichkeit des Nutzers zu entwerfen [Kühl06, F,S.26]. Die Nutzerprofile sind für die Marktforschung, Werbung oder zur bedarfsgerechten Gestaltung von Websites nützlich oder gar erforderlich (§ 6 Abs. 3 TDDSG). Grundsätzlich sind die Nutzerprofile mit Einwilligung des Nutzers möglich, das bereits besprochene Koppelungsverbot muss explizit beachtet werden (§ 3 Abs. 4 TDDSG). Die Einwilligung kann schriftlich oder elektronisch erfolgen (§ 4 Abs. 2 TDDSG). Sind Daten erhoben worden, so dürfen sie nur zu dem angegebenen Zweck benutzt werden, eine Zweckentfremdung ist nur mit einer weiteren Einwilligung oder durch eine gesetzliche Erlaubnis möglich (§ 3 Abs. 3 TDDSG).

Das Anlegen von Nutzerprofilen unter Verwendung von Nutzungsdaten ist zulässig, wenn folgende Voraussetzungen erfüllt sind: (§ 6 Abs. 3 TDDSG):

- *das Anlegen eines Nutzerprofils ist nur unter Verwendung von Pseudonymen zulässig* (§ 6 Abs. 3 Satz 1 TDDSG). Es gibt verschiedene Arten von Pseudonymen [WoGe05, 3.3.2.3]:
  - das *selbstgenerierte Pseudonym* wird vom Benutzer selbst vergeben. Nur er kann seine wahre Identität aufdecken
  - *Referenz-Pseudonyme* sind Listen anhand derer man den Personenbezug herstellen kann, es existieren so genannte Referenzlisten.
  - *Einweg-Symbole* können nur einmal benutzt werden, bei mehrfacher Benutzung kann eine Identifikation möglich sein (z.B. TAN-Listen)

  Es wird nur das Anlegen von Nutzerprofilen erlaubt, für die Übermittlung müssen die Nutzerprofile entweder anonymisiert werden (§ 6 Abs. 5 Satz 4 TDDSG) oder die Übermittlung ist zum Zwecke der Abrechnung und Ermittlung des Entgelts notwendig (§ 6 Abs. 5 Satz 1 und Satz 2 TDDSG)
- das Nutzerprofil wird zum Zwecke der Marktforschung, Werbung oder zur bedarfsgerechten Gestaltung von Websites angelegt. Eine weitergehende Verwendung ist nicht zulässig (§ 6 Abs. 3 Satz 1 1. HS TDDSG).
- ausdrücklicher Hinweis auf das Widerspruchsrecht und das Nicht-widersprechen des Nutzers (§ 6 Abs. 3 Satz 1 2. HS und Satz 2 TDDSG). Auf das Widerspruchsrecht ist vor der Übermittlung hinzuweisen, so dass der Nutzer eine informierte Entscheidung treffen kann.
- das Nutzerprofil darf nicht mit Daten des Träger des Pseudonyms zusammengeführt werden (§ 6 Abs. 3 Satz 3 TDDSG). Anhand der Pseudonymen kann kein Dritter auf die Person schließen. Persönliche Daten sind somit getrennt von den Pseudonymen zu verwahren.

2.2 Inhaltsdaten und Gestaltung von Websites

2.2.1 Inhaltsdaten

Inhaltsdaten sind eigentlich die Daten, die keine Bestands-, Verkehrs-, Nutzungs- oder Abrechnungsdaten sind. Also Daten, für die keine speziellere Regelung wie das TKG oder TDDSG greift. Inhaltsdaten fallen somit unter das BDSG. Im Vordergrund stehen hier die Inhalte der Kommunikation. Im BDSG ist die Erhebung, Verarbeitung und Nutzung personenbezogener Daten ohne Einwilligung nicht zulässig, es sei denn, das BDSG oder ein anderes Gesetz erlaubt das ausdrücklich. Auf diese so genannten Erlaubnistatbestände des BDSG wird folgenden näher eingegangen. Man unterscheidet bei nicht-öffentlichen Stellen grundsätzlich zwischen der Erhebung und Verwendung für eigene Zwecke und der *geschäftsmäßigen* Datenerhebung und -speicherung zum Zweck der Übermittlung (§ 29 und § 30 BDSG). Im Fall der geschäftsmäßigen Datenerhebung und -speicherung zum Zweck der Übermittlung bilden die Daten die Ware oder Dienstleistung, während bei der Verarbeitung für eigene Zwecke die Daten nur Hilfsmittel für den eigentlichen Geschäftszweck sind [WoGe05, 4.4.3.1]. Drei relevante Erlaubnistatbestände bei Erhebung und Verwendung für eigene Zwecke werden näher erläutert:

- Die Erhebung und Verwendung von Daten für eigene Zwecke ist zulässig, wenn es der *Zweckbestimmung eines Vertragsverhältnisses* dient (§ 28 Abs. 1 Satz 1 Nr. 1 BDSG). Der Zweck ist ein gemeinsamer Zweck, er wird von beiden Parteien des Vertrages verfolgt [WoGe05, 4.4.2.2.1]. Zum Beispiel wäre die Gewinnung von Werbeadressen nur ein einseitiger Zweck.

- Die Erhebung und Verwendung von Daten für eigene Zwecke ist zulässig, wenn das berechtigte Interesse der verantwortliche Stelle das schutzwürdige Interesse des Betroffenen überwiegt (§ 28 Abs. 1 Satz 1 Nr. 2 BDSG). Es ist kein rechtliches Interesse von Nöten, ein berechtigtes Interessen kann schon darin bestehen, typischen Geschäftsrisiken vorzubeugen [WoGe05, 4.4.2.2.2]. Das schutzwürdige Interesse des Betroffenen ist im Lichte der informationellen Selbstbestimmung (Art. 2 Abs. 1 i.V.m. Art. 1 Abs. 1 GG) auszulegen. Beide Interessen sind gegeneinander abzuwiegen.
- Die Erhebung und Verwendung von Daten für eigene Zwecke ist zulässig, wenn die Daten öffentlich zugänglich sind und das schutzwürdige Interesse des Betroffenen am Ausschluss der Verarbeitung oder Nutzung nicht überwiegt (§ 28 Abs. 1 Satz 1 Nr. 3 BDSG). Allgemein zugängliche Daten sind Daten aus Printmedien, Rundfunk, Fernsehen, Filmen, Videos, CD-ROM, Telefonbücher. Daten und Informationen von Websites sind ebenfalls öffentlich zugängliche Daten [WoGe05, 4.4.2.2.3]. Grundsätzlich reicht die *Entnehmbarkeit*. Im Rahmen einer Güterabwägung muss das schutzwürdige Interesse des Betroffenen überprüft werden.

2.2.2 Sensible Daten

Ein besonderes Interesse gilt noch den sog. sensitiven oder sensiblen Daten. Im Gesetz werden sie als „besondere Arten von Daten“ bezeichnet. Es sind restriktive Anforderungen zu beachten. Man sollte sich die Einwilligung des Betroffenen einholen, da ein Vorrang der Einwilligung besteht [Simi03, § 28 R.321]. Ein Anforderungskatalog befindet sich in § 28 Abs. 6 und Abs. 7 BDSG. Sensible Daten sind vor allem solche (§ 3 Abs. 9 BDSG): Angaben über

- rassische und ethnische Herkunft,
- politische Meinungen,
- religiöse und philosophische Überzeugungen,
- Gewerkschaftszugehörigkeit,
- Gesundheit,
- Sexualleben.

Die Angabe, dass sich eine Person den Fuß verletzt hat und partiell krankgeschrieben ist, ist bereits ein sensibles Datum. So urteilte der EuGH in seinem Urteil vom November 2003 [Geri03].

2.3 Datenschutzanforderungen und datenschutzgerechte Gestaltung von Websites

Werden personenbezogene Daten erhoben, so muss die verantwortliche Stelle genannt und über den Zweck der Erhebung, Verarbeitung oder Nutzung informiert werden (§ 4 Abs. 3 BDSG). Bei Telediensten, die *geschäftsmäßig* betrieben werden (§ 6 TDG) sowie bei den Mediendiensten, gilt die Informationspflicht. Diese Informationspflicht soll für Transparenz sorgen, denn der Nutzer möchte natürlich wissen, mit wem er es zu tun hat. Da die verantwortliche Stelle genannt werden muss, hat die Informationspflicht auch eine Relevanz im Datenschutz. Medien- bzw. Teledienste sind durch folgenden Informationen zu kennzeichnen (§ 6 TDG, § 10 MDStV):

- Name und Anschrift
- elektronische Kontaktadresse (eMail)
- bei einer behördlichen Zulassungspflicht, die Angaben zur Behörde
- zugehöriger Registereintrag und Registernummer
- u.U. Angaben zum Beruf
- u.U. Umsatzsteueridentifikationsnummer

Bei nicht geschäftsmäßigen Mediendiensten reicht der Name und die Anschrift aus. Besondere Informationspflichten gibt es noch für „kommerzielle Kommunikation“ bei den Telediensten (§ 7 TDG). Für Websites reicht ein *Impressum* auf der jeweiligen Seite oder ein Link auf dieses aus. Wie bereits erwähnt, muss der Nutzer über die Erhebung und Verwendung personenbezogener Daten informiert werden. Das ist im § 4 Abs. 1 TDDSG bzw. im § 18 Abs. 1 MDStV geregelt und umfasst folgende Pflichten:

- Informationen über die Art der Erhebung und Verwendung
- Informationen über den Umfang der Erhebung und Verwendung
- Informationen über den Zweck der Erhebung und Verwendung
- Informationen über die Verarbeitung seiner Daten in Drittstaaten
- Unterrichtung, dass bei einem automatisierten Verfahren der Nutzer später identifiziert werden kann
- Jederzeitige Abrufbarkeit der Unterrichtung durch den Nutzer

Die Unterrichtung sollte in ausreichender Schriftgröße, deutlich sichtbar auf der Seite angebracht sein oder über einen Link auf diese zugänglich sein. Nicht ausreichend ist ein allgemeiner Hinweis auf die AGBs, oder ein Hinweis, dass personenbezogene Daten erhoben oder verwendet werden, oder ein Hinweis, dass die Datenschutzbestimmungen eingehalten werden [Scha02b, S.9]. Das BDSG gibt dem Betroffenen ein Auskunftsrecht (§ 34 Abs. 1 BDSG), dass es ihm ermöglicht, über

- zu seiner Person gespeicherte Daten (§ 34 Abs. 1 Satz 1 Nr. 1 BDSG),
- die Empfänger, die Daten erhalten haben (§ 34 Abs. 1 Satz 1 Nr. 2 BDSG),
- sowie über den Zweck der Speicherung (§ 34 Abs. 1 Satz 3 Nr. 3 BDSG)

informiert zu werden. Das TDDSG konkretisiert das Auskunftsrecht im § 4 Abs. 7 TDDSG weiter, indem der Betroffene *unentgeltlich und unverzüglich* Auskunft über seine Person oder zu seinem Pseudonym gespeicherten Daten Auskunft zu geben hat. Generell gilt für das Auskunftsrecht, dass es nicht abdingbar ist (§ 6 Abs. 1 BDSG). Werden personenbezogene Daten automatisch verarbeitet, so kann der Betroffene auch Auskunft über den logischen Aufbau der Datenverarbeitung verlangen (§ 6a Abs. 3 BDSG). Besteht der Betroffene auf eine schriftliche Auskunft, so muss sie ihm auch schriftlich erteilt werden. Ansonsten kann man sie auch elektronisch erteilen, z.B. durch Authentifizierung über das Web oder durch eine E-Mail an den Betroffenen [Scha02b, S.17]. Auch die Weitervermittlung zu einem anderen Diensteanbieter muss dem Nutzer angezeigt werden. Unter einer Weitervermittlung zu einem anderen Diensteanbieter sind z.B. externe Links zu verstehen. Um das zu ermöglichen, reicht

z.B. die Einblendung eines Hinweistextes oder das Erscheinen eines Hinweistextes beim Zeigen des Mauscursors auf den externen Link, oder das Kennzeichnen von Bannerwerbung durch den Hinweis „Anzeige“ [Scha02b, S.19]. Eine Einwilligung des Nutzers kann folgendermaßen realisiert werden [Scha02b, S.13]:

- double opt in
  - der Nutzer aktiviert ein Kontrollkästchen
  - der Nutzer erhält eine Bestätigungs-Email in der die Einwilligung schriftlich fixiert ist
  - der Nutzer sendet die Bestätigungs-Email zurück und erklärt damit, dass er einverstanden ist
- confirmed opt in
  - der Nutzer aktiviert ein Kontrollkästchen
  - der Nutzer erhält eine Bestätigungs-Email, in der die Einwilligung schriftlich fixiert ist und auf sein Widerspruchsrecht hingewiesen wird

Rechtlich zweifelhaft und eventuell unwirksam ist Folgendes:

- nur ein Hinweis auf die AGBs, denn ein Hinweis ist nur eine Information und keine Einwilligung des Nutzers
- Einräumung eines Widerspruchsrecht (opt-out Lösung) ist ebenso keine Einwilligung
- Einblenden einer Einwilligungserklärung ohne eine Bestätigung des Nutzers ist auch noch keine Einwilligung
- fehlender Hinweis auf die Freiwilligkeit der Angaben des Nutzers ermöglicht keine informierte Entscheidung des Betroffenen

2.3.1 Verantwortlichkeit

Falls es zu einem Verstoß gegen das Datenschutzrecht kommt, ist die Frage wer für den Verstoß verantwortlich ist, relevant. Die Verantwortlichkeiten werden in den §§ 8-11 TDG sowie in den §§ 6-9 MDStV geregelt. Man kann sie grundsätzlich in folgender Übersicht darstellen:

- für *eigene Informationen und eigene Inhalten* ist der Anbieter selber verantwortlich; man spricht von einem *Content-Provider* (§ 8 Abs. 1 TDG bzw. § 6 Abs. 1 MDStV). Damit sind zunächst selbst erstellte Informationen auf der eigenen Homepage gemeint. Tageszeitungen oder Börseninformationsdienste sind Beispiele von Content-Providern.
- für *fremde Informationen und fremde Inhalte* ist der Anbieter, der sie zur Nutzung speichert, der sog. *Service-Provider* grundsätzlich nicht verantwortlich, wenn er
  - keine Kenntnis von der rechtswidrigen Handlung hat,
  - und sie bei Kenntnisnahme die Information unverzüglich entfernt oder den Zugang gesperrt haben.

  (§ 11 TDG bzw. § 9 MDStV) Der Service Provider bietet nur seine Ressource als Dienst an. Zum Beispiel stellt 1und1 Speicherplatz zu Verfügung; in diesem Fall ist 1und1 nur Service-Provider.

- für *fremde Informationen und fremde Inhalte*, die von einem Anbieter in einem Kommunikationsnetz übermittelt werden oder zu dem sie einen Zugang vermitteln – sog. *Access-Provider* – ist der Anbieter grundsätzlich nicht verantwortlich, wenn er
  - die Vermittlung nicht veranlasst hat,
  - den Adressaten der übermittelten Information nicht ausgewählt hat,
  - die übermittelten Informationen nicht ausgewählt oder verändert hat.

  (§ 9 Abs. 1 TDG bzw. § 7 Abs. 1 MDStV) Access-Provider sind Anbieter, die außer dem bloßen technischen Kommunikationsvorgang keine weiteren Informationen bereithalten. Ein Beispiel wäre die Telekom. Access-Provider spielen auf der TK-Ebene eine große Rolle.

2.3.2 Konsequenzen bei einem Verstoß gegen das Datenschutzrecht

Wer gegen das Datenschutzrecht verstößt, kann von der Strafverfolgungsbehörden belangt und / oder privatrechtlich zur Rechenschaft gezogen werden. Im § 43 BDSG ist ein umfangreicher Katalog aufgeführt. Wer vorsätzlich oder fahrlässig gegen einen der genannten Punkte verstößt, handelt *ordnungswidrig*. Eine Ordnungswidrigkeit ist z.B. das fahrlässige Erheben oder Verarbeiten personenbezogener Daten, die nicht allgemein zugänglich sind (§ 43 Abs. 2 Nr. 1 BDSG). Ordnungswidrigkeiten werden nach dem § 43 Abs. 1 BDSG mit bis zu 25000 Euro, nach dem § 43 Abs. 2 BDSG mit bis zu 250000 Euro geahndet. Ein Verstoß gegen das Datenschutzrecht ist dann eine Straftat, wenn zusätzlich zum § 43 Abs. 2 BDSG ein Entgelt genommen wird oder die Absicht besteht, sich oder andere zu bereichern oder jemand anderen zu schädigen (§ 44 BDSG). Die Straftat kann mit bis zu 2 Jahren Gefängnis oder Geldstrafe geahndet werden. Privatrechtlich kann der Geschädigte aus § 823 Abs. 2 BGB i.V.m. § 7 BDSG Schadensersatz verlangen, wenn das Gesetz oder andere datenschutzrelevante Rechtsvorschriften verletzt wurden. Die Pflicht zum Schadensersatz entfällt, wenn die verantwortliche Stelle die gebotene Sorgfalt beachtet hat.

2.4 Protokollierung / Logs

Nachdem wir im vorherigen Kapital auf die Inhalte und deren Gestaltung näher eingegangen sind, wollen wir uns nun mit Bestands- und Nutzungsdaten und deren Erhebung und Verarbeitung befassen.

2.4.1 Bestandsdaten

Der Begriff der Bestandsdaten ist im § 5 Satz 1 TDDSG und im § 19 Abs. 1 Satz 1 MDStV geregelt. Dabei handelt es sich um Daten, die auch dann *ohne Einwilligung* erhoben, verarbeitet und genutzt werden dürfen, wenn sie für die Begründung, inhaltliche Ausgestaltung oder Änderung eines Vertragsverhältnis erforderlich sind. Es ergibt sich also rein aus dem Zweck des jeweiligen Vertragsverhältnisses, welche Daten zu den Bestandsdaten zurechnen sind. Sie sind unabhängig von der Nutzung des Dienstes und werden regelmäßig für einen längeren Zeitraum gespeichert. Beispiele für Bestandsdaten sind:

- Personalien
- Bankverbindung
- Kreditkarteninformationen

- Zugangskennungen

Werden Bestandsdaten mit den Daten aus dem Nutzungsvorgang verbunden, so handelt es sich um Nutzungsdaten. Die Aufzählung der Erlaubnistatbestände ist abschließend, wie es aus dem Wortlaut des Gesetzes („nur“) hervorgeht. Der Zweckbindungsgrundsatz gilt natürlich auch hier, d.h. eine Zweckentfremdung – Erhebung, Verarbeitung, Nutzung von personenbezogenen Daten ohne Einwilligung zu einem anderen Zweck als der vom Vertragsverhältnis – ist grundsätzlich nicht zulässig.

2.4.2 Nutzungsdaten

Der Begriff Nutzungsdaten wird im § 6 Abs. 1 Satz 1 TTDSG und § 19 Abs. 2 Satz 1 MDStV legaldefiniert. Es handelt sich um personenbezogene Daten, die ohne Einwilligung erhoben, verarbeitet und genutzt werden dürfen, wenn dies für die Inanspruchnahme oder zur Abrechnung des jeweiligen Dienstes erforderlich ist. Im Satz 2 werden Nutzungsdaten aufgezählt; die Aufzählung ist nicht abschließend („insbesondere“) :

- Merkmale zur Identifikation des Benutzers (lit. a)
- Angaben über Beginn und Ende sowie über den Umfang der jeweiligen Nutzung (lit. b)
- Angaben über die vom Nutzer in Anspruch genommenen Dienste (lit. c)

Nutzungsdaten sind immer personenbezogen, da sie immer einer Person zugeordnet werden können. Typische Nutzungsdaten, die für das WWW relevant sind, sind folgende [Scha02a, Rn. 426]:

- Systemdaten (z.B. IP-Adresse des Nutzers)
- Identifikationsdaten (z.B. Zugangsdaten, Passwörter des Nutzers)
- Standort der genutzten Ressource (z.B. URL)
- Nutzungsart der Ressource (z.B. lesen, schreiben, löschen)
- jedwede Information des Benutzers, die ausgelesen werden kann (z.B. über den Browser) wie z.B. Betriebssystem, Seriennummern von Software oder Hardwarekomponenten, Browsertyp, usw.
- spezifische, vom Dienst abhängige Information (Eingabe bei Eingabemasken)

Zur „Inanspruchnahme oder zur Abrechnung eines Dienstes“ dürfen Nutzungsdaten erhoben und verwendet werden. Daneben dürfen Nutzungsdaten zum „Zwecke der Werbung, Marktforschung und bedarfsgerechten Gestaltung des Dienstes“ zur Erstellung von Nutzerprofilen verwendet werden (§ 6 Abs. 3 TDDSG bzw. § 19 Abs. 4 MDStV). Die Nutzungsdaten müssen entweder unter Pseudonymen verwendet oder komplett anonymisiert sein. Daneben gibt es noch den Tatbestand der Mißbrauchsaufklärung. Im § 6 Abs. 8 TDDSG bzw. § 19 Abs. 9 MDStV wird geregelt, dass wenn es tatsächlich dokumentierte Anhaltspunkte gibt, die darauf hinweisen, dass ein Nutzer das Entgelt überhaupt nicht oder nur teilweise entrichtet, der Diensteanbieter Nutzungsdaten erheben und über die Frist von 6 Monaten (Abs. 7 TDDSG bzw. Abs. 8 MDStV) speichern darf. Schaar [Scha02a, Rn. 435] zählt Beispiele für zulässige Speicherung und Verarbeitung von Nutzungsdaten auf:

- Erhebung und Verarbeitung der Nutzungszeiten und Nutzeridentifikation nur im Rahmen des jeweiligen Tarifs zum Zwecke der Abrechnung
- Verbindung der Nutzungszeiten mit Bestandsdaten nur zum Zwecke der Abrechnung
- Speicherung von dynamischen IP-Adressen während des Nutzungsvorgangs
- Verwendung von Nutzungsdaten und Nutzerprofilen zum Zwecke der Marktforschung und Werbung nur unter Pseudonym

2.4.3 Abrechnungsdaten

Abrechnungsdaten sind Nutzungsdaten, die über das Ende des Nutzungsvorgangs hinaus gespeichert und verarbeitet werden dürfen, wenn es zum Zwecke der Abrechnung erforderlich ist (§ 6 Abs. 4 TDDSG bzw. § 19 Abs. 5 MDStV). Die Erforderlichkeit bezieht sich auf den Grundsatz der Datensparsamkeit und Datenvermeidung (§ 3a BDSG), d.h. auch nur wirklich für die Abrechnung relevante Daten dürfen gespeichert und verarbeitet werden. Der Anbieter, der Zeitpunkt, die Dauer, die Art, der Inhalt und die Häufigkeit bestimmter von einem Nutzer in Anspruch genommener Dienste dürfen nicht erkennbar sein, mit der Ausnahme, dass das vom Nutzer verlangt wird (Einzelnachweis) (§ 6 Abs. 6 TDDSG bzw. § 19 Abs. 7 MDStV).

2.4.4 Protokollierung und Logs

Zusammenfassend kann man sagen, dass die Erhebung, Verarbeitung und Nutzung der Daten eng an den eigentlichen Zweck gebunden ist. So sind Bestandsdaten an das Vertragsverhältnis gebunden und Nutzungsdaten an den Nutzungsvorgang. Da Daten immer nur dann für einen Zweck erhoben, verarbeitet und genutzt werden dürfen, wenn sie dafür erforderlich sind, müssen die Daten regelmäßig nach Wegfall des Zweckes gelöscht werden; Bestandsdaten bei Beendigung des Vertrages, Nutzungsdaten bei Beendigung des Nutzungsvorgangs, Abrechnungsdaten nach der Abrechnung. Bezüglich Nutzungsdaten ermöglicht § 9 BDSG eine darüber hinausgehende Verwendung, wenn sie zur Gewährleistung der Datensicherheit innerhalb der Stelle erforderlich sind. Der Umfang dieser Maßnahmen ist eng auszulegen. Die Daten müssen vor Zugriffe Dritter sicher aufbewahrt werden. Wie oben besprochen, dürfen Daten auch zur Missbrauchsbekämpfung protokolliert werden. Aber auch hier in engen Grenzen.

3 Datenschutzkonzepte

3.1 Selbstdatenschutz

Der Selbstdatenschutz stellt Konzepte dar, wie man sich selber gegenüber den Gefahren bezüglich Datensicherheit und Datenschutz schützen kann. Es handelt sich insoweit um einen Selbstschutz der Nutzer. Wichtig dafür ist zunächst die Aufklärung der Nutzer – und zwar auf beiden Seiten, also Benutzer und Anbieter – über die Gefahren. Nur wenn der Nutzer ein Bewusstsein dafür entwickelt, wird er auch bereit sein, sich aktiv zu schützen. Mit der zunehmenden Verbreitung des Internets rückten auch die Gefahren hinsichtlich Datenschutz und Datensicherheit immer mehr ins Zentrum von Nachrichten und Berichterstattungen. Mit der zunehmenden Sensibilisierung der Benutzer kann sich der Einsatz von datenschutzfördernden Techniken für den Anbieter als Wettbewerbsvorteil erweisen. Jedenfalls lässt sich mit der Gewährleistung von Datenschutz und Datensicherheit das Vertrauen der Benutzer in den Anbieter stärken. Es werde mehrere Möglichkeiten aufgezählt, wie man auf freiwilliger Basis Datenschutz gewährleisten kann.

3.1.1 Privacy Policies

Privacy Policies (Privacy Statements) stellen ein ziemlich einfache Möglichkeit dar, Transparenz im Datenschutz zu gewährleisten. Sie sind eine Erklärung des Diensteanbieters, wie er mit den personenbezogenen Daten des Nutzer umgeht. Man könnte auch von einer „freiwilligen Selbstverpflichtung“ sprechen. Für diesen ist das eine Hilfe, ob er Daten von sich Preis geben will oder nicht. Die OECD hat einen Privacy-Statement-Generator entwickelt, der bei der Gestaltung von Privacy Policies hilfreich sein kann:

http://www.oecd.org/sti/privacygenerator.

3.1.2 P3P

P3P ist ein vom W3C entwickelter Industriestandard zum Austausch von Datenschutzinformationen. P3P steht für Platform for Privacy Preferences. Dem Nutzer soll es ermöglicht werden, durch von ihm festgelegte Datenschutzpräferenzen, die Verwendung seiner Daten durch den Web-Seiten-Betreiber zu kontrollieren. Das wird von den meisten gängigen Browsern unterstützt oder mit einem Plug-In ermöglicht. P3P gehört zu den so genannten Privacy-Enhancing Technologies (PET). Die PETs ermöglichen Datenschutz durch technische Verfahren. Damit wird Datenschutz gleich in die Technik integriert oder durch technische Mittel gefördert. Der Nutzer legt dafür seine eigene Datenschutzpräferenzen fest, wie er z.B. mit Cookies umgehen möchte. Der Serverbetreiber legt bei sich seine Privacy Policies an, in denen er beschreibt, was bei ihm mit den Daten des Nutzers geschieht. Bei einem Aufruf der Website durch den Nutzer, werden die Präferenzen des Nutzers mit der Privacy Policy verglichen. Weichen beide voneinander ab, erscheinen Warnhinweise auf Seite des Nutzers, der dann in den Vorgang eingreifen kann. Sind beide identisch, läuft der Vorgang ohne Unterbrechung weiter. Folgende Darstellung veranschaulicht das Konzept von P3P:

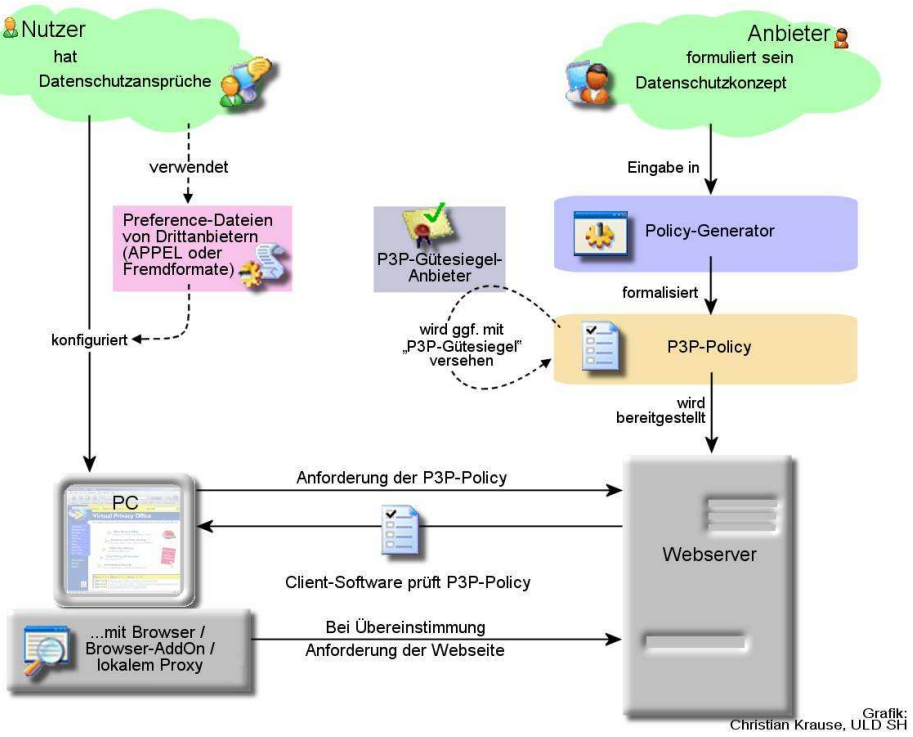


Abbildung 1: Funktionsweise P3P. [Krau]

Rechtlich gesehen ist allerdings fraglich, ob die Festlegung der Datenschutzpräferenzen auch eine Einwilligung darstellt [Scha02a, Rn.936].

3.1.3 Privacy-Enhancing Technologies (PET)

Privacy-Enhancing Technologies lassen sich in verschiedene Gruppen einteilen:

- Schutz der Identität des Nutzers, mit Hilfe von Anonymitätsdienste oder Dienste zur Pseudonymisierung des Nutzers wie z.B. einen identity protector, der Pseudonyme generieren und verwalten kann
- Schutz der Herkunft der Daten, ebenso mit Hilfe von Anonymitätsdienste oder Dienste zur Pseudonymisierung wie z.B. durch einen Proxy (http://www.anonymizer.com/)
- Schutz des Inhalts der Daten, z.B. mit Hilfe von Datenverschlüsselungsprogramme zur sicheren Kommunikation

3.2 Datenschutzaudit

Datenschutzaudit ist die Prüfung des Datenschutzes durch eine dritte, objektive Instanz vor. Somit wird also überprüft, ob die freiwillige Selbstregulierung des Anbieters gewissen Ansprüchen genügt. Werden die Maßstäbe der Prüfungsstelle erfüllt, erhält der Anbieter ein Gütesiegel. Es ist allerdings auch klar, dass die Qualität des Gütesiegels stark von der Prüfungsstelle, nämlich hinsichtlich der Intensität und Maßstäbe der Prüfung, abhängt. Um ein aussagekräftiges Gütesiegel zu schaffen, hat der Gesetzgeber die Idee des Datenschutzaudits im § 9a BDSG verankert. Im Satz 2 wird für die näheren Anforderungen an die Prüfung, und Bewertung, das Verfahren sowie die Auswahl und Zulassung der Gutachter auf ein „besonderes Gesetz“ verwiesen. Dieses Umsetzungsgesetz zum Datenschutzaudit gibt es bisher leider noch nicht. Trotzdem gibt es vielversprechende Umsetzungen des Datenschutzaudits, wie z.B. durch das Land Schleswig-Holstein: http://www.datenschutzzentrum.de/guetesiegel/index.htm.

4 Ausblick

4.1 Vorratsdatenspeicherung

Die Richtlinie 2006/24/EG zur Vorratsdatenspeicherung von Daten ist am 3. Mai 2006 in Kraft getreten und muss bis zum 15. September 2007 im nationales Recht umgesetzt werden. Die Richtlinie wurde zum Zweck der Ermittlung und Verfolgung von schweren Straftaten beschlossen. Die Vorratsdatenspeicherung betrifft gemäß Art. 1 Abs.1 die Anbieter öffentlich zugänglicher elektronischer Kommunikationsdienste oder Betreiber eines öffentlichen Kommunikationsnetzes. Grundsätzlich werden nur Inhalts- und Standortdaten auf Vorrat gespeichert, *keine Inhaltsdaten* (Art. 1 Abs. 2 bzw. Art. 5 Abs. 2). Die auf Vorrat gespeicherten Daten müssen mindestens zwischen 6 und höchstens 24 Monate gespeichert werden (Art. 6), je nach nationaler Umsetzung. Die folgenden Daten sind für das WWW relevant und müssen entsprechend auf Vorrat gespeichert werden:

- zur Rückverfolgung und Identifizierung der Quelle einer Nachricht benötigte Daten betreffend Internetzugang, Internet-E-Mail und Internet-Telefonie (Art. 5 Abs. 1 lit. a Nr. 2 (i)-(iii)):

  - die zugewiesene(n) Benutzerkennung(en),
  - die Benutzerkennung und die Rufnummer, die jeder Nachricht im öffentlichen Telefonnetz zugewiesen werden,
  - der Name und die Anschrift des Teilnehmers bzw. registrierten Benutzers, dem eine Internetprotokoll- Adresse (IP-Adresse), Benutzerkennung oder Rufnummer zum Zeitpunkt der Nachricht zugewiesen war;

  Mit der Speicherung der IP-Adresse und der Einwahldaten – wie z.B. die t-online Benutzerkennung – kann jeder Nutzer identifiziert werden.

- zur Identifizierung des Adressaten einer Nachricht benötigte Daten betreffend Internet-E-Mail und Internet-Telefonie (Art. 5 Abs. 1 lit. b Nr. 2 (i)-(ii)):

  - die Benutzerkennung oder Rufnummer des vorgesehenen Empfängers eines Anrufs mittels Internet-Telefonie,
  - die Namen und Anschriften der Teilnehmer oder registrierten Benutzer und die Benutzerkennung des vorgesehenen Empfängers einer Nachricht;

- zur Bestimmung von Datum, Uhrzeit und Dauer einer Nachrichtenübermittlung benötigte Daten betreffend Internetzugang, Internet-E-Mail und Internet-Telefonie (Art. 5 Abs. 1 lit. c Nr. 2 (i)-(ii)):

  - Datum und Uhrzeit der An- und Abmeldung beim Internetzugangsdienst auf der Grundlage einer bestimmten Zeitzone, zusammen mit der vom Internetzugangsanbieter einer Verbindung zugewiesenen dynamischen oder statischen IP-Adresse und die Benutzerkennung des Teilnehmers oder des registrierten Benutzers,
  - Datum und Uhrzeit der An- und Abmeldung beim Internet-E-Mail-Dienst oder Internet- Telefonie-Dienst auf der Grundlage einer bestimmten Zeitzone;

- zur Bestimmung der Art einer Nachrichtenübermittlung benötigte Daten betreffend Internet-E-Mail und Internet-Telefonie: Anspruch genommene Internetdienst (Art. 5 Abs. 1 lit. d);

- zur Bestimmung der Endeinrichtung oder der vorgeblichen Endeinrichtung von Benutzern benötigte Daten betreffend Internetzugang, Internet-E-Mail und Internet-Telefonie (Art. 5 Abs. 1 lit. e Nr. 3 (i)-(ii)):

  - die Rufnummer des anrufenden Anschlusses für den Zugang über Wählanschluss,
  - der digitale Teilnehmeranschluss ein anderer Endpunkt des Kommunikationsvorgangs.

Nach Ablauf der Frist müssen die Daten vernichtet bzw. gelöscht werden (Art. 7 lit. d). Es ist natürlich klar, dass nun jedem Internet-Anschluss die Online-Zeit, IP-Adresse und der Name und die Anschrift des Besitzers zugeordnet werden kann. Die Vorratsdaten werden für die Strafverfolgungsbehörden dann interessant, wenn es noch entsprechendes Zusatzwissen gibt: der Eintrag der IP-Adresse in Web-Protokollen, Anmelde-Protokollen für Dienste und ähnlichem. Jeder Benutzer hinterlässt somit Spuren, die von den Behörden verfolgt werden können. Und das betrifft Kriminelle genauso wie normale Bürger. Ob es verhältnismäßig ist, alle Internetnutzer unter Generalverdacht zu stellen, ist fraglich.

4.2 Telemediengesetz

Das geplante Telemediengesetz soll die Regelungen des MDStV, TDG und TDDSG in einem Gesetz vereinen. Damit trägt der Gesetzgeber der Konvergenz der Medien Rechnung. Die aktuelle Fassung ist vom 14. Juni 2006 und wurde noch nicht von der Bundesregierung im Bundestag eingereicht. Allerdings zeichnet sie jetzt schon der Widerstand einiger Interessenverbände ab, die eine Aufweichung des Datenschutzes sehen. Der Entwurf der Bundesregierung befindet sich hier:

http://www.bmwi.de/BMWi/Redaktion/PDF/M-O/elgvg-elektronischer-gesch_C3_A4ftsverkehr-vereinheitlichungsgesetz,property=pdf,bereich=bmwi,sprache=de,rwb=true.pdf

5 Fazit

Durch den Wandel zur Informationsgesellschaft werden in vielen Prozessen Daten erhoben und verwendet. Für Firmen und Unternehmen sind personenbezogen Daten von großem Wert, erlauben sie doch perfekt abgestimmte Marketingstrategien. Mit der Verbreitung von Informationstechnolgien steigt aber auch die Sensibilität der Nutzer bezüglich dem Umgang mit ihren Daten. Der Datenschutz wird auch in Zukunft ein immer größere Rolle spielen. Sich mit dem Datenschutz auseinander zu setzen hat zweierlei Vorteile: Durch den richtigen Umgang mit personenbezogenen Daten kann das Vertrauen des Verbrauchers in das eigene Unternehmen gestärkt werden und eventuell rechtliche Probleme vermieden werden.

Literatur

[Bund83] Bundesverfassungsgericht. BVerfGE 65, 1 - Volkszählung. 1983. Urteil des Ersten Senats vom 15. Dezember 1983 auf die mündliche Verhandlung vom 18. und 19. Oktober 1983 - 1 BvR 209, 269, 362, 420, 440, 484/83 in den Verfahren über die Verfassungsbeschwerden.

[Geri03] Europäischer Gerichtshof. in der Rechtssache C-101/01. 2003. Richtlinie 95/46/EG - Anwendungsbereich - Veröffentlichung personenbezogener Daten im Internet - Ort der Veröffentlichung - Begriff der Übermittlung personenbezogener Daten in ein Drittland - Meinungsfreiheit - Vereinbarkeit eines weiter gehenden Schutzes personenbezogener Daten nach den Rechtsvorschriften eines Mitgliedstaats mit der Richtlinie 95/46.

[IhDi03] R. Ihde und M. Dittmann. Cookies im Internet. *Datenverarbeitung, Steuer, Wirtschaft, Recht: DSWR* (12), 2003, S. 350–352.

[Krau] C. Krause. http://www.datenschutzzentrum.de/selbstdatenschutz/p3p/grafiken/p3p_flow_1.jpg.

[KrMo00] D. Kristol und L. Montulli. HTTP State Management Mechanism. RFC 2965, oct 2000.

[Kühl06] J. Kühling. Vorlesung Datenschutzrecht, 2006.

[Scha02a] P. Schaar. *Datenschutz im Internet*. Beck. 2002.

[Scha02b] P. Schaar. Datenschutzgerechte Gestaltung von Websites. 2002. http://privcom-datenschutz.de/Pr%e4sentationen/DS-gerechte%20Websites.pdf.

[Schl04] S. Schleipfer. Das 3-Schichten-Modell des Multimediadatenschutzrechts. *Datenschutz und Datensicherheit* (28), 2004, S. 727–733.

[Simi03] S. Simitis. *Kommentar zum Bundesdatenschutzgesetz*. Nomos-Verlagsgesellschaft. 5. Auflage, 2003.

[uMdK00] Arbeitskreise Technik und Medien der Konferenz der Datenschutzbeauftragten des Bundes und der Länder. OH Internet. In *Orientierungshilfe zu Datenschutzfragen des Anschlusses von Netzen der öffentlichen Verwaltung an das Internet*, 2000. http://www.lfd.m-v.de/download.html.

[WoGe05] H. Wolgemuth und J. Gerloff. *Datenschutzrecht*. Leuchterhand. 2005.

Abbildungsverzeichnis

Datenschutz im eCommerce - Elektronische Bezahlsysteme

Yorck Frhr. v. Mirbach

Kurzfassung

Die Landschaft der elektronischen Bezahlverfahren im eCommerce hat sich in der letzten Dekade signifikant verändert. Keines der Systeme der Pionierphase hat eine entscheidende Verbreitung gefunden, und von den Systemen der zweiten Generation haben nur einige wenige einen ausreichenden Kundenstamm aufbauen können. Derzeit befinden wir uns an der Schwelle der dritten Generation elektronischer Bezahlverfahren. In dieser Arbeit werden die verbliebenen Systeme der zweiten Generation und die Vertreter der dritten Generation eingehend betrachtet. Neben der Funktionsweise und Handhabbarkeit der einzelnen Verfahren wird insbesondere auch auf die Notwendigkeit des Datenschutzes eingegangen. Abschließend wird ein Ausblick gegeben, welche Systeme den Anforderungen der Zukunft genügen und damit auch mittelfristig Bestand haben werden.

1 Einleitung

Der eCommerce in Deutschland hat ungebrochene Zuwachsraten und wächst im zweistelligen Prozentbereich. Nach einer gemeinsamen Studie des Bundesverbandes Informationswirtschaft, Telekommunikation und neue Medien e.V. (BITKOM), des Bundeswirtschaftsministeriums (BMWi) sowie des Bundesverbandes der Deutschen Industrie (BDI) ist der Umsatz im eCommerce in Deutschland im Jahr 2005 um 58% auf 321 Milliarden Euro angewachsen [Maga06]. Durch den Fokus auf elektronische Bezahlverfahren, ist für diese Arbeit besonders der Umsatz im B2C-E-Commerce von Interesse, der vom Hauptverband des Deutschen Einzelhandels für das Jahr 2006 auf 16,3 Milliarden Euro geschätzt wird [dDeu05]. Im europäischen Vergleich ist Deutschland damit führend beim Umsatz im Internethandel. Durch diese Größenordnung wird klar, welch ein Potential, gerade hier in Deutschland, nach wie vor für elektronische Bezahlverfahren existiert, da die Zahlungen im B2C–Bereich nahezu ausschließlich im Internet initiiert werden.

Durch die Nutzung elektronischer Bezahlverfahren kann eine schnelle und komfortable Abwicklung gewährleistet werden. Dies ist gerade im Bereich der digitale Güter wichtig, da ein Medienbruch durch Einsatz traditioneller Bezahlsysteme unzweckmäßig ist. Zusätzlich können die traditionellen Systeme kein adäquates Angebot für niederpreisige Artikel im Mikropaymentbereich ($< €5.00$) machen.

Ein weiterer Pluspunkt der hier betrachteten Verfahren ist der deutliche sparsamere Umgang mit Daten im Vergleich zu den traditionellen Verfahren. Die persönlichen Daten müssen für Zahlungen nicht mehr beim einzelnen Internethändler angegeben werden, sondern entfallen entweder komplett oder müssen nur zentral beim Zahlungssystemanbieter hinterlegt werden.

Dennoch werden die traditionellen Bezahlverfahren noch sehr häufig für die Abwicklung von Zahlungen im Internet verwendet. Dies lässt sich zum einen auf die große Fülle von Anbietern elektronischer Bezahlverfahren zurückführen; denn wer weiß schon, für welches Zahlungssystem man sich entscheiden soll, wenn auf jeder Händlerseite ein anderes Zahlungssystem beworben wird. Darüber hinaus sind die einzelnen Zahlungssysteme nicht kompatibel, so

dass es oftmals nicht ausreicht, sich nur bei einem Zahlungssystem anzumelden. Diese Hürde schreckt gerade unerfahrenere Internetnutzer ab, sich überhaupt auf die innovativen Systeme einzulassen.

Persönliche Daten im Internet preiszugeben, ist ein sehr sensibles Thema, gerade wenn es um Bankdaten geht. Dieser Punkt erhält eine besondere Bedeutung in Anbetracht der Zahlen, die das Institut für Wirtschaftspolitik und Wirtschaftsforschung (IWW) der Universität Karlsruhe in seiner Online-Umfrage IZV8 ermittelt hat [KrLS06]. Nach dieser Studie haben schon 7,6% der Online-Nutzer negative Erfahrungen bei Zahlungen im Internet gemacht. Solche Zahlen schüren die Furcht vor Datenmissbrauch und verringern die Akzeptanz, Daten im Internet preiszugeben.

Ein weiterer Grundsatz, der eine Vorraussetzung dafür ist, dass elektronische Bezahlverfahren akzeptiert werden, ist die Einfachheit der Benutzung. Gerade Systeme der ersten Generation haben diesen Grundsatz verletzt.[1] An dieser Stelle soll kurz der Fall „Secure Electronic Transaction“ (SET) angeschnitten werden, da dieses System in der einschlägigen Literatur nach wie vor oft genannt wird, obwohl es ein bedeutender Fehlschlag war. Die Idee als solche bestach sicherlich durch ihr Gesamtkonzept. Durch die beiden Initiatoren Mastercard und Visa standen zusätzlich noch entsprechend kompetente Partner dahinter. Das System sollte die Kreditkartenzahlung durch Einsatz von digitalen Signaturen und Verschlüsselungsmechanismen extrem sicher machen, womit die Bedenken auf Kundenseite bezüglich der Nutzung von Kreditkarten im Internet aufgehoben werden sollten. Der auschlaggebende Punkt für das grandiose Scheitern dieser Methode war die übermäßige Komplexität der Benutzung für den Internetnutzer.[2] Verallgemeinernd folgt als Resultat, dass sich jedes aktuelle elektronische Bezahlverfahren zu allererst an der Einfachheit der Benutzung orientieren muss.

Der nachfolgende Teil dieser Arbeit ist in drei Hauptabschnitte aufgeteilt. Im ersten Teil werden die einzelnen Verfahren dargestellt und beschrieben, im zweiten Teil folgt die datenschutzrechtliche Betrachtung und zum Abschluss erfolgt ein Ausblick, welche Verfahren sich mittelfristig, für welche Bereiche durchsetzen werden.

In der Betrachtung der einzelnen Verfahren werden zuerst einmal die Besonderheiten elektronischer Bezahlverfahren im allgemeinen betrachtet, bevor die Systeme im einzelnen erörtert werden. Die Gliederung der Bereiche ist nach dem Zeitpunkt der tatsächlichen Begleichung der Rechnung vorgenommen worden. Die vorgestellten Verfahren sind in die folgenden Bereiche aufgeteilt:

- **PrePaid:** Bei Systemen dieser Gruppe wird ein Guthaben bei dem Anbieter gekauft und kann im Anschluss beliebig verbraucht werden.
- **Pay-on-Demand:** Bei dieser Systemgruppe wird der Betrag im Moment des Bezahlens vom Kunden beglichen.
- **PostPaid:** Hier begleicht der Kunde seine Kaufbeträge gegenüber dem Anbieter des Zahlungssystems kumuliert über einen gewissen Zeitraum.
- **Pay-as-you-like:** Bei diesen Systemen kann der Kunde im Einzelfall wählen, welche der drei o.g. Bereiche er nutzen möchte. Diese Gruppe stellt die Systeme der dritten Generation dar.

Analog zu den gewählten Bereichen wurden die Systeme ausgewählt, die in ihrer Gruppe die vielversprechenste Zukunft zu haben scheinen. Im einzelnen sind dies:

[1] [PáNP05] stellt das Versagen der Systeme der ersten Generation eindrucksvoll dar.
[2] Für eine genaue Betrachtung des Systems siehe [Schw04]

- PaySafeCard
- GeldKarte
- Click&Buy
- PayPal

Der zweite Teil der Arbeit konzentriert sich auf die datenschutzrechtliche Betrachtung. Auch hier werden zuerst die allgemeinen Besonderheiten für elektronische Bezahlverfahren dargestellt und dann eine knappe Einordung in die Rechtsgrundlagen vorgenommen. Anschließend folgt die genaue Betrachtung der einzelnen Verfahren.

Zum Schluss werden die einzelnen Verfahren gesondert bewertet und es wird eine Aussage darüber getroffen, für wen und für welchen Zweck welche Verfahren als sinnvoll erscheinen. Zusätzlich werden die Zukunftschancen der einzelnen Systeme betrachtet und eine Aussage getroffen, welche der vorgestellten Syteme uns auch noch in ein paar Jahren begegnen werden.

2 Elektronische Bezahlsysteme

2.1 Besonderheiten elektronischer Bezahlsysteme

Die Arbeit behandelt elektronischen Bezahlverfahren. Wie der Studie des IWW zu entnehmen ist (Abbildung 1) wird der Großteil der Zahlungen im Internet jedoch immer noch mit den traditionellen Zahlungsverfahren durchgeführt.

1.9 Welche Zahlungsmethoden kennen Sie oder haben Sie schon beim Einkaufen oder Bestellen über das Internet benutzt?

N=(12518-13234), Angaben in vH der Teilnehmer, Mehrfachnennungen möglich

		IZV8	G1	G2	Erf1	Erf2	Erf3	Newsleser	Gewinnspieler
Mobiltelefon	Bekannt und verwendet	1,5	1,6	1,3	2,1	1,4	1,5	1,2	1,5
	Bekannt, nicht verwendet	20,0	19,6	20,5	14,9	19,8	20,7	20,3	20,4
	Unbekannt	78,5	78,8	78,2	82,9	78,8	77,9	78,5	78,1
Vorausbezahlte Systeme	Bekannt und verwendet	9,0	10,5	5,9	7,5	9,2	9,1	5,5	3,7
	Bekannt, nicht verwendet	28,3	25,2	30,3	22,5	25,9	31,6	31,5	25,1
	Unbekannt	62,7	64,4	63,9	70,0	64,9	59,4	63,0	71,1
Inkasso-/Billing-Verfahren	Bekannt und verwendet	29,6	35,7	22,9	21,1	29,5	30,9	23,4	11,6
	Bekannt, nicht verwendet	37,4	32,7	45,5	25,3	33,7	42,6	45,4	36,9
	Unbekannt	33,0	31,6	34,6	53,5	36,9	26,3	31,3	51,5
Kreditkarte	Bekannt und verwendet	47,3	48,5	46,0	23,7	41,7	56,0	50,3	25,7
	Bekannt, nicht verwendet	33,2	31,2	35,3	32,0	34,9	32,1	35,7	38,9
	Unbekannt	19,5	20,3	18,7	44,3	23,4	11,9	14,0	35,4
Bezahlen per E-Mail	Bekannt und verwendet	18,1	18,3	18,0	8,1	16,2	22,8	19,7	13,7
	Bekannt, nicht verwendet	35,7	32,3	39,3	25,8	35,5	38,9	40,6	38,7
	Unbekannt	46,1	49,3	42,7	66,1	48,3	38,3	39,6	47,7
Elektronische Lastschrift vom Händler initiiert	Bekannt und verwendet	53,2	51,2	55,4	26,9	45,8	64,2	61,9	35,2
	Bekannt, nicht verwendet	30,1	30,6	29,6	38,4	34,0	25,1	27,2	36,9
	Unbekannt	16,6	18,2	15,0	34,7	20,2	10,7	10,9	27,9

Abbildung 1: Bekanntheitsgrad und Nutzung von Bezahlsystemen [KrLS06]

Besonders auffällig ist hierbei die weit verbreitete Nutzung der elektronischen Lastschrift und der konventionellen Kreditkartenzahlung. In Abgrenzung zu den konventionellen Verfahren, wird im Rahmen dieser Arbeit ein elektronisches Bezahlsystem wie folgt definiert: „*Ein elektronisches Bezahlsystem entkoppelt die Zahlung vom Nutzer. Darüber hinaus findet die Zahlung ohne Medienbruch vollständig im Internet statt.*“ Durch diese Definition wird klar, dass

die elektronische Lastschrift nicht als elektronsches Bezahlverfahren gilt, da dem Inhalteanbieter sowohl Name und Zahlungsinformationen des Kunden übermittlet werden, womit diese Informationen gekoppelt bleiben. Die Kreditkartenzahlung hingegen verletzt den Grundsatz, dass die Zahlung ohne Medienbruch stattfinden soll, da das Clearing, der Abgleich mit der Kreditkartenzentrale, nicht über das Netz abläuft. Somit ist auch die Kreditkartenzahlung kein elektronisches Bezahlverfahren im Sinne dieser Arbeit.

Hervorzuheben sind einige Besonderheiten der elektronischen Verfahren:

- Datensparsamkeit durch Pseudonymität bzw. Anonymität. Dadurch ist eine direkte Nachverfolgbarkeit des Kunden durch den Händler nicht gegeben.
- Durch die vollständige Abwicklung im Netz und den Fokus der Systemanbieter sind die Verfahren tauglich für Zahlungen im Micropayment-Bereich.
- Die Zahlungsgeschwindigkeit ist deutlich höher als bei den traditionellen Verfahren.

Die technische und funktionelle Betrachtung der einzelnen Verfahren wird aufgrund der folgenden Kriterien vorgenommen:

Ablauf: Hier wird beschrieben, wie der Bezahlvorgang mit dem jeweiligen System abläuft, welche Parteien in welcher Form beteiligt sind und wie die Interaktion im einzelnen abläuft. Diese Darstellung wird sowohl grafisch, als auch in Textform vorgenommen.

Höhe der Zahlungen: In welchem Bereich können Zahlungen mit diesem System vorgenommen werden? Dadurch kann geklärt werden, für welche Zahlungen das System besonders geeignet ist.

Einfachheit der Bedienung: Wie wird der Aufwand bewertet, den ein Nutzer bei dem gewählten Verfahren hat? Dieser Punkt ist, wie schon oben genannt, der entscheidende Punkt dafür, dass ein System überhaupt überlebensfähig ist.

Multi–Nationalität: Welche Verbreitung besitzt das angesprochene System? Das ist zum einen wichtig, wenn es um die Akzeptanz des Systems bei Händlern geht und zum anderen, wenn es darum geht, ob das System andere Währungen als den Euro verarbeiten kann.

Nutzung: In welchem Umfang wird das System heute genutzt? Welcher Umsatz und welche Nutzerzahlen liegen dem System zugrunde? Dadurch wird ausgedrückt, wie etabliert das System bereits ist.

Sicherheit: Wie sicher ist das System und welche Sicherheitsmechanismen werden genutzt? Ein Bezahlsystem muss sicher sein, denn sobald Zweifel an der Sicherheit des Systems aufkommen, hat dies gravierende Folgen und führt zwangsweise zu Kundenverlusten. Hierbei ist allerdings anzumerken, dass ein elektronisches Bezahlsystem nur so sicher sein kann wie der Rechner über den ein Nutzer es bedient! Ohne elementare, persönliche Sicherheitssysteme, wie eine *„personal firewall“* sowie ein *„aktiver Viren- und Trojanerschutz“* ist jede Form der elektronischen Bezahlung inhärent unsicher. Auf diese Aspekte wird in dieser Arbeit nicht weiter eingegangen.

Partner: Welche Partner sind an dem Verfahren beteiligt? Für die Durchsetzung und den Betrieb eines solchen Systems sind starke Partner nötig, die durch ihren Namen und ihre Finanzkraft das System stützen.

2.2 PrePaid – PaySafeCard

Die PaySafeCard ist ein System, dass auf dem Konzept einer PrePaid-Karte beruht und seit dem Jahr 2001 in Deutschland verfügbar ist. [AG06]

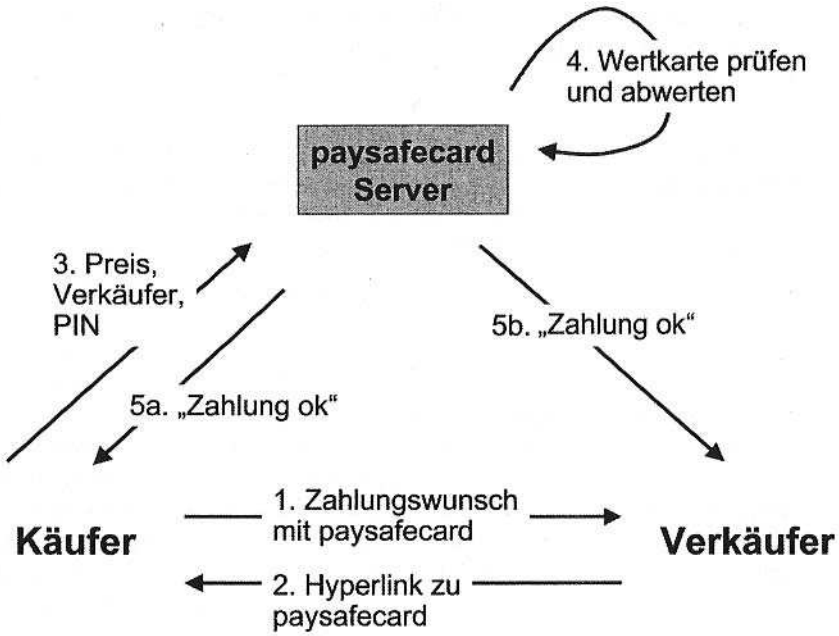


Abbildung 2: Schematischer Ablauf bei Nutzung der PaySafeCard [RoBG03]

Ablauf: Die PaySafeCard gibt es in zwei Varianten: Zum einen als Karte ähnlich einer Telefonkarte und zum anderen als POS-PIN Ausdruck[3]. Beides wird von stationären Händlern verkauft. Das gekaufte Guthaben steht dem Nutzer ab diesem Zeitpunkt voll zur Verfügung. Wenn der Nutzer nun eine Zahlung im Internet tätigen möchte, wählt er die entsprechende Option bei dem Internethändler aus und wird von diesem an den PaySafeCard-Server geroutet. Auf den zwei Varianten der Karte befindet sich jeweils eine 16-stellige PIN. Die Eingabe dieser PIN beim PaySafeCard-Server ist ausreichend für die Bezahlung. Für die weitere Nutzung kann für die Karte ein spezielles Passwort vergeben werden, damit wird die Sicherheit für die aktivierte Karte weiter erhöht. Der Zahlungsvorgang selbst wird über den Intermediär PaySafeCard-Server durchgeführt, der die Daten der Karte prüft und danach die Zahlung gegenüber dem Kunden und dem Händler bestätigt. Abbildung 2 stellt den Ablauf schematisch dar.

Höhe der Zahlungen: Die PaySafeCard gibt es in verschiedenen Werten (€10, €25, €50, €100), zusätzlich ist es möglich, mehrere Karten miteinander zu verbinden, so dass zum einen alte Karten aufgebraucht und zum anderen höhere Beträge bezahlt werden können. Mit der PaySafeCard können durch das vorhandene Guthaben, Kleinstbeträge im unteren Centbereich beglichen werden.

Einfachheit der Bedienung: Die Bedienung der PaySafeCard ist einfach, die Eingabe der PIN und des eventuellen Passwortes sind ausreichend für die Bezahlung. Als Manko bleibt, dass die Karte im stationären Handel gekauft werden muss.

Multi–Nationalität: Die PaySafeCard ist derzeit in Östereich und Deutschland präsent und damit ausschliesslich in der Eurozone.

[3] Ausdruck der erforderlichen Informationen im Handel über einen Rechnungsdrucker

Nutzung: Die PaySafeCard wird von ca. 2.000 Web-Shops akzeptiert und im Jahr 2005 wurden €60 Millionen mit Hilfe der PaySafeCard umgesetzt. Durch die Anonymität des Verfahrens können keine Aussagen über die Anzahl der Nutzer gemacht werden.[4]

Sicherheit: Durch die Begrenzung der Kartenhöhe auf maximal €100 und die statistisch extrem geringe Chance eine 16-stellige Zahl *korrekt zu erraten*, ist das Verfahren als ausreichend sicher einzustufen.

Partner: Die PaySafeCard besitzt die Commerzbank, die BAWAG und die IBM als Partner, womit Sie durch namenhafte Partner gestützt wird.

2.3 Pay-on-Demand – GeldKarte

Die Geldkarte ist ein Gemeinschaftsprojekt aller deutschen Banken und Sparkassen und nutzt eine pyhsische Chipkarte zur Bezahlung. [RoBG03]

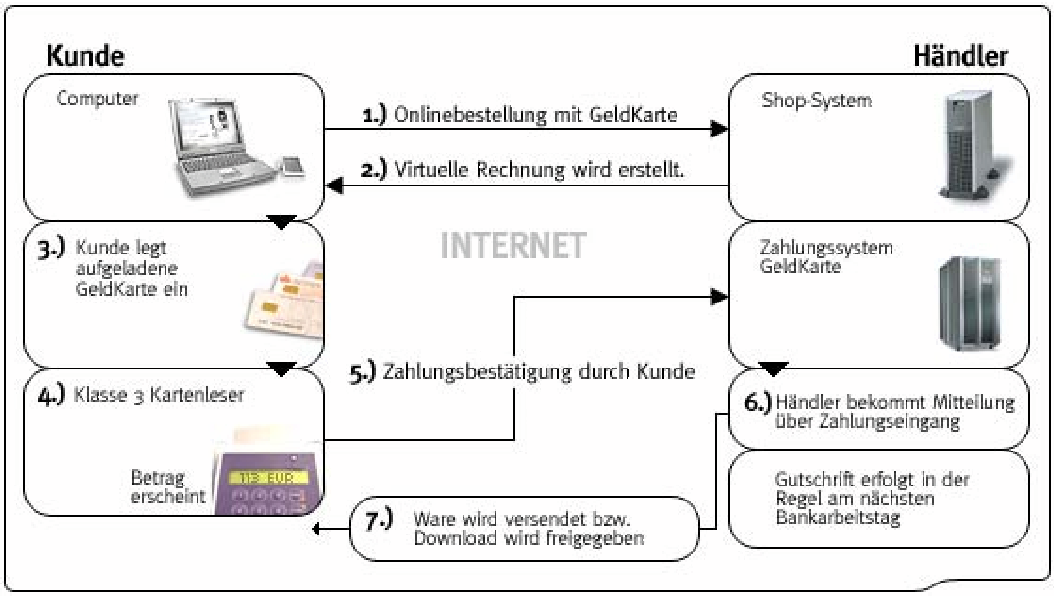


Abbildung 3: Schematischer Ablauf bei Nutzung der GeldKarte [Geld06]

Ablauf: Die GeldKarte ist in nahezu alle deutschen EC-Karten integriert und hat Ihre Hauptfunktion im Bereich des stationären Handels. Durch den Einsatz spezieller Chipkartenlesegeräte (Kartenleser Stufe 3) ist der Einsatz auch für die Bezahlung im Internet möglich. Die Chipkarte wird an einem entsprechenden Ladeterminal in der Offline-Welt aufgeladen. Bei einem Bezahlvorgang wird die Option GeldKarte gewählt, der Händler übermittelt dann eine entsprechende digitale Form der Rechnung, die an den Kartenleser weitergeleitet wird. Durch die Eingabe der Geldkarte und der dazugehörigen PIN wird die Zahlung auf der Kundenseite initiiert und dann an einen Bankserver übertragen. Dieser bestätigt die Zahlung gegenüber dem Händler und fungiert damit ebenfalls als Intermediär für die Zahlung. Abbildung 3 stellt den Ablauf schematisch dar.

Höhe der Zahlungen: Die GeldKarte kann bis zu einer Höhe von €200 aufgeladen werden und kann durch das gespeicherte Guthaben ebenfalls für Kleinstbeträge genutzt werden.

[4]Diese Daten stammen aus einem Experteninterview mit der PaySafeCard.com Wertkarten AG

Einfachheit der Bedienung: Bevor die Karte genutzt werden kann, müssen zuerst einmal ein kleines Softwareprogramm und der Kartenleser installiert werden, danach wird die Zahlung wie in Geschäften über den Kartenleser vorgenommen.

Multi–Nationalität: Die GeldKarte gibt es ausschliesslich in Deutschland und eine Ausweitung darüber hinaus erscheint kaum wahrscheinlich.

Nutzung: In Deutschland sind momentan ca. 63 Millionen GeldKarten im Umlauf, die im Internet bisher hauptsächlich zur Alterskontrolle genutzt werden. Für den Jugendschutz existieren schon 130.000 Akzeptanzstellen im Internet. Durch die hohen Anschaffungskosten der speziellen Lesegeräte für die Bezahlung bieten bisher nur 7 Web-Shops die Nutzung der GeldKarte für das elektronische Bezahlen an. [5]

Sicherheit: Durch die Trennung der Zahlung vom eigentlichen PC über das Lesegerät ist die Zahlung mit der GeldKarte als besonders sicher anzusehen.

Partner: Die GeldKarte ist ein Gemeinschaftsprojekt der deutschen Banken und Sparkassen. Hierbei muss allerdings darauf verwiesen werden, dass die Nutzung im stationären Handel Hauptaugenmerk der Partner ist.

2.4 PostPaid – Click&Buy

Das Zahlungssystem Click&Buy der Firma Firstgate ist ein Abrechungsservice, der zwischen Kunde und Händler vermittelt. [RoBG03]

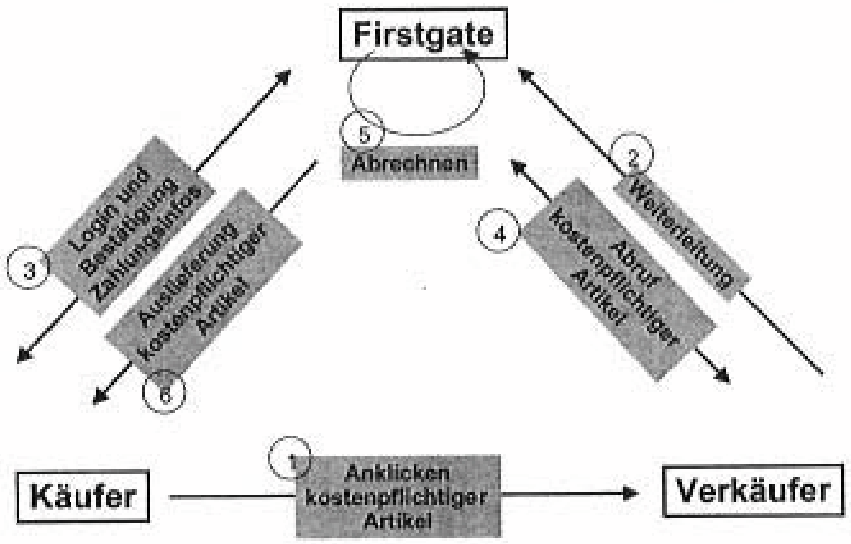


Abbildung 4: Schematischer Ablauf bei Nutzung von Click&Buy [RoBG03]

Ablauf: Anders als die bisher vorgestellten Verfahren findet die komplette Zahlungskette bei dem Zahlungssystem Click&Buy der Firma Firstgate im Internet statt. Der Nutzer registriert sich einmalig auf der Homepage von Click&Buy und gibt dort seine Konto- oder Kreditkarteninformationen an. Diese Daten werden benötigt, um die vom Nutzer getätigten Zahlungen später abzurechnen. Beim Interneteinkauf wird nach der Wahl der Zahlungsoption Click&Buy der Nutzer direkt an den Server von Click&Buy weitergeleitet, dort wird der Nutzer aufgefordert seine registrierte E-Mail-Adresse und sein

[5] Diese Daten stammen aus einem Experteninterview bei der EURO Kartensysteme GmbH

Passwort einzugeben. Nach der Bestätigung dieser Daten wird der Download der digitalen Ware gestartet. Firstgate übernimmt damit die Verrechnung und Distribution von digitalen Waren für Ihre Händlerkunden. Abbildung 4 stellt den Ablauf schematisch dar.

Höhe der Zahlungen: Durch den Fokus auf digitale Güter bewegt sich die typische Höhe der Zahlungen in einem Bereich von €0,1 bis €50, da jedoch eine langfristige Bindung der Kunden mit hinterlegten Bankdaten besteht, sind grundsätzlich auch höhere Zahlungen möglich.

Einfachheit der Bedienung: Nach der Registrierung muss nach Auswahl der Zahlungsoption die hinterlegte E-Mail-Adresse und das dazugehörige Passwort eingegeben werden und die Zahlung ist erfolgt.

Multi–Nationalität: Click&Buy ist international weit verbreitet und in Europa, Amerika und Asien vertreten. Damit können unterschiedliche Währungen problemlos über das System abgerechnet werden.

Nutzung: Mit ca. 6.000 Webshops und 6 Millionen registrierten Kunden ist das System schon voll etabliert und hat eine gute Marktposition.

Sicherheit: Solange die Benutzerdaten nicht kompromittiert werden, ist eine gute Sicherheit gegeben, da alle Zahlungsinformationen über den Server der Firma Firstgate laufen. Durch die langfristige Bindung können gestohlene Nutzerdaten allerdings einen hohen Schaden anrichten. Es ist allerdings möglich ein „spending limit“ für die Abrechnungsperiode zu setzen, womit diese Gefahr verringert wird.

Partner: Obwohl Click&Buy keine stützenden Partner hat, konnte es sich sehr gut im Markt etablieren und behaupten.

2.5 Pay-as-you-like – PayPal

PayPal, das Tochterunternehmen von Ebay, bietet einen umfassenden Zahlungsservice an. [Wiki06] [NiWe02a]

Ablauf: PayPal ermöglicht Zahlungen an E-Mail-Adressen von einem PayPal-Konto. Um Transaktionen durchführen zu können, müssen jedoch beide Parteien bei PayPal registriert sein. Für den Internetkäufer stellt sich der Bezahlvorgang wie folgt dar: Er wählt die Zahlungsoption PayPal und wird daraufhin an die PayPal-Seite weitergeleitet. Dort gibt er seine registrierte E-Mail-Adresse und das Passwort ein, die Zahlungsinformationen des Händlers sind schon weitestgehend eingetragen, und er muss dies nur noch bestätigen, um den Vorgang abzuschließen. Die Zahlung geht dann innerhalb von Sekunden beim Händler ein. Dem Kunden bieten sich vorrangig zwei Möglichkeiten die Bilanz seines PayPal-Kontos auszugleichen: Zum einen Post-Paid über die Angabe einer Kreditkarte und zum andern PrePaid über die vorherige Aufladung des Kontos per Überweisung.

Höhe der Zahlungen: Mit PayPal sind auch Zahlungen im Micropayment Bereich möglich; der Schwerpunkt liegt jedoch derzeit noch auf höheren Zahlungen.

Einfachheit der Bedienung: Nach Auswahl der Zahlungsoption muss die registrierte E-Mail-Adresse und das entsprechende Passwort eingegeben werden, danach müssen die vorgegebenen Daten überprüft werden, um die Zahlung abzuschliessen. Wirklich überzeugend wird diese Methode dadurch, dass mit PayPal auch normale Geldüberweisungen auf diese Weise durchgeführt werden können.

Multi–Nationalität: PayPal ist besonders weit verbreitet und in über 55 Nationen verfügbar.

Nutzung: Mit über 100 Millionen registrierten Nutzern hat sich PayPal weltweit durchgesetzt. Anzumerken ist allerdings, dass es vor allem in Amerika eine weite Verbreitung gefunden hat, da dort eine Bundesstaaten übergreifende Überweisung von den Banken nicht angeboten werden. [NiWe02b] Durch die enge Bindung an das Mutterunternehmen und die entsprechenden Auktionstransaktionen ist es nicht ganz klar wie viele der 100 Millionen Nutzer das erweiterte Angebot von PayPal nutzen. Dennoch gibt es keinen Zweifel an der besonders starken Marktposition von PayPal.

Sicherheit: Auch hier ist der Bezahlvorgang nur durch eine Kombination von Benutzername und Passwort geschützt. Da das System i.d.R. für Zahlungen höherer Beträge genutzt wird, ist dies bedenklich.

Partner: PayPal wurde 1999 selbstständig gegründet, ist aber seit 2002 eine 100%ige Tochtergesellschaft von eBay.

3 Datenschutzrechtliche Betrachtung

3.1 Allgemeines zum Datenschutz in elektronischen Bezahlsystemen

Für die Nutzer ist Datenschutz im eCommerce essentiell. Das Bewusstsein, dass jeder Schritt in einem Web-Shop nachverfolgt werden kann und im Regelfall auch wird, hat zu einer Skepsis gegenüber dem eCommerce im Allgemeinen geführt. Dem Internetnutzer wird die Vollständigkeit und Wirksamkeit dieser Methoden vor Augen geführt, wenn er Web-Shops häufig besucht. Als prominentes Beispiel ist hier der Internet-Buchhändler Amazon zu nennen. Durch gelegentliche Einkäufe und sporadische Besuche der Seite gelingt es dem Portal oftmals schon erschreckend genau auf die Präferenzen des Nutzers ausgewählte Produkte zu präsentieren. Das hat natürlich für beide Seiten auch einen positiven Aspekt, der Nutzer kann ohne große Suche für ihn passende Produkte kaufen, und der Händler erhöht damit seinen Umsatz. Doch offensichtliche Methoden der Auswertung der personenbezogenen Daten wie diese schüren die Furcht vor den Methoden im Hintergrund, Methoden, die vor dem Nutzer verborgen bleiben. Durch die Vielzahl der Informationen, die durch das Surfen im Internet generiert werden, ist der Internetnutzer nicht mehr Herr über seine Daten. Er kann im Einzelfall nicht bestimmen, ob Daten genutzt werden.

Vor diesem Hintergrund ist der Datenschutz bei Bezahlsystemen im Internet besonders wichtig. Hier können datenschutzrechtliche Aspekte den Unterschied über Erfolg und Mißerfolg ausmachen. Die Daten, die ein Nutzer für Zahlungen offenlegt, gehören für den Nutzer zu den sensibelsten Daten überhaupt. Somit können Bezahlsysteme, die dem Kunden gegenüber klar vermitteln, dass seine Daten sicher sind bzw. seine Daten zur Zahlung gar nicht erst benötigt werden, eine vertrauensbildende Maßnahme für den Internethandel insgesamt sein. Ein Web-Shop kann somit seine Akzeptanz durch den Einsatz geeigneter Zahlungsmethoden erhöhen. Dadurch wird die Wahl von datensparsamen Bezahlsystemen zu einem entscheidenden Wirtschaftsfaktor im eCommerce.

Den Anforderungen an Datensparsamkeit werden elektronische Bezahlsysteme in sehr viel besserem Maße gerecht, als dies die traditionellen Systeme tun. Bei elektronischen Bezahlverfahren werden Daten, wenn überhaupt, nur beim Anbieter des Systems hinterlegt. Wenn der Nutzer bei einem Web-Shop einkauft, der dieses System unterstützt, muss er seine Zahlungsdaten nicht gegenüber dem Internethändler offenlegen. Die Zahlung wird über den Systemanbieter abgewickelt. Zu beachten ist hierbei, dass die Wahl eines elektronischen Bezahlsystems

nicht bedeutet, dass der Händler keine persönliche Daten vom Kunden benötigt. Wenn es sich bei der Ware um physische Produkte handelt, müssen diese zum Kunden gelangen. Dafür benötigt der Internethändler wenigstens die Lieferanschrift des Kunden, wahrscheinlich ergibt sich auch hier schon eine Profilbildung im Web-Shop. Anders verhält es sich, wenn digitale Waren gehandelt werden. Hierbei ist es nicht notwendig, dass der Händler persönliche Daten über den Kunden sammelt. Durch die Wahl eines elektronischen Bezahlsystems kann der Kunde gegenüber dem Händler/Anbieter anonym bleiben.

3.1.1 Rechtlicher Rahmen

Bevor im weiteren datenschutzrechtlich auf die einzelnen Verfahren eingegangen wird, soll noch der rechtliche Rahmen abgesteckt werden, in dem sich elektronische Bezahlsysteme bewegen. Grundsätzlich gibt es drei Gesetzeswerke die für den Datenschutz in diesem Bereich in Frage kommen können: Das Bundesdatenschutzgesetz (BDSG), das Teledienstedatenschutzgesetz (TDDSG) sowie der Mediendienste-Staatsvertrag (MDStV). Zur Entscheidung, welches dieser Gesetze für elektronische Bezahlverfahren gilt, muß die Unterscheidung getroffen werden, ob es sich hierbei um *Teledienste*, oder *Mediendienste* handelt. Das Teledienstegesetz (TDG) definiert Teledienste in seinem § 2 als „*elektronische Informations- und Kommunikationsdienste, die für eine individuelle Nutzung von kombinierbaren Daten wie Zeichen, Bilder oder Töne bestimmt sind und denen eine Übermittlung mittels Telekommuniktation zugrunde liegt.*“ Der MDStV definiert Mediendienste in seinem § 2 Abs. 1 als „*das Angebot und die Nutzung von an die Allgemeinheit gerichteten Informations- und Kommunikationsdiensten...*“ Diese beiden Definitionen unterscheiden sich im wesentlichen durch den Zielsetzung der Individualnutzung bzw. der Nutzung durch die Allgemeinheit. Hieraus wird ersichtlich, dass es sich bei elektronischen Bezahlverfahren um Teledienste handelt, da hierbei eine Individualnutzung definitiv gegeben ist und für Teledienste ist das TDDSG zuständig. Somit greift zu allererst das TDDSG, Aspekte die dort nicht betrachtet werden sind durch das BDSG abgedeckt. [RoBG03]

Deutsches Recht gilt nach dem Sitzlandprinzip erst einmal nur für Unternehmen, die Ihren Sitz auch in Deutschland haben. Durch die Allgemeine Datenschutz-Richtlinie der EG (DSRL) ist europäisches Recht jedoch vereinheitlicht, so dass hier keine besonderen Probleme auftreten. Da alle angesprochenen Unternehmen wenigstens eine Niederlassung in einem europäischen Staat haben, ist eine weitere Betrachtung in Bezug auf das Ausland hier nicht notwendig. [6]

3.1.2 Bewertungskriterien

Für die datenschutzrechtliche Betrachtung der Verfahren, wird ab jetzt nur noch der Erwerb digitaler Güter betrachtet. Wie bereits erörtert, besteht für einen Händler/Anbieter dieser Waren eigentlich kein Grund mehr, Daten jenseits der Zahlungsabwicklung über den Nutzer zu sammeln. Darüber hinaus ist es zur Bewertung der Datensicherheit sinnvoll zu betrachten, welche Parteien welche Daten einsehen können. In diesem Szenario können zwei Parteien unterschieden werden: Der Anbieter der digitalen Waren und der Betreiber des Zahlungssystems. Für diese Parteien gilt es zu klären, welche Art des Zugriffes sie auf Daten besitzen, ob es sich um personenbeziehbare, pseudonyme oder anonyme Daten handelt. [7]

[6] Dennoch sei an dieser Stelle angemerkt, dass die DSRL eine Übertragung personenbezogener Daten in Staaten außerhalb der EU nicht erlaubt, wenn dort nicht ähnliche Datenschutzgesetze gelten. Da dies beispielsweise für die Vereinigten Staaten gilt, wurde im Jahr 2000 die Datenschutz-Vereinbarung *Safe Harbor* verabschiedet. Durch diese Vereinbarung ist es möglich, dass amerikanische Unternehmen, die sich zu den Grundsätzen von *Safe Harbor* verpflichten, personenbezogene Daten in die USA übertragen können. [Fink02]

[7] Personenbeziehbar sind Daten dann, wenn Zusatzwissen verfügbar ist, um die Daten einer bestimmten Person direkt zuzuordnen. Als anonym gelten Daten, wenn es „*die Wahrscheinlichkeit, dass diese der Person*

Zur Bewertung der einzelnen Verfahren ergeben sich somit die folgenden Beurteilungskriterien:

Datenmenge: Welche Daten werden für die Nutzung des Zahlungsdienstes und für die einzelnen Zahlung gesammelt, und wo werden sie gespeichert?

Dateneinsicht: Auf welche Daten des Nutzers hat der Anbieter der digitalen Waren Zugriff, und auf welche Daten kann der Betreiber des Zahlungssystems zugreifen?

Art des Zugriffs: Wie stellt sich die Art des Zugriffs für die beiden Parteien dar. Handelt es sich um anonyme, pseudonyme, oder personenbeziehbare Daten?

Gefahr des Datenmissbrauchs: Wie hoch ist die Gefahr des Datenmissbrauchs einzuschätzen, sowohl vom Waren- als auch vom Zahlungsanbieter?

3.2 PaySafeCard

Datenmenge: Da die PaySafeCard anonym im stationären Handel gekauft wird, fallen hierbei keine personenbezogenen Daten an. Für die eigentliche Bezahlung wird ebenso kein Personenbezug hergestellt, da nur die PIN geprüft wird. Als einzige Daten fallen die Nutzungsdaten an über die der Nutzer jedoch nicht identifiziert werden kann.

Dateneinsicht: Keine der beiden Parteien hat Zugriff auf personenbezogene Daten. Hierdurch wird die enorme Datensparsamkeit der PaySafeCard unterstrichen.

Art des Zugriffs: Es werden keine personenbezogenen Daten generiert, womit es sich für Inhalte- und Zahlungssystemanbieter um anonyme Daten handelt.

Gefahr des Datenmissbrauchs: Durch die enorme Datensparsamkeit ist ein Datenmissbrauch bei der Verwendung der PaySafeCard nicht zu befürchten.

3.3 GeldKarte

Datenmenge: Bei der Aufladung der GeldKarte wird vom Konto des Besitzers der entsprechende Betrag abgebucht und auf den Chip der GeldKarte übertragen. Hierbei kennen die kartenausgebenden Banken zwar die personenbezogenen Daten des Besitzers, aber die Abrechnung erfolgt über die , von den Banken unabhängigen, Evidenzzentralen, in denen Schattensalden mit Bezug auf die Kartennummern geführt werden. In den Evidenzzentralen fallen somit Daten über die einzelnen Transaktionen an, während bei den kartenausgebenden Banken Daten über die Be- und Entladung der GeldKarten anfallen. [RoBG03]

Dateneinsicht: Der Anbieter der digitalen Ware bekommt nur mitgeteilt, dass die Zahlung eingegangen ist. Der Zahlungsanbieter führt über die Evidenzzentralen eine Transaktionsübersicht.

Art des Zugriffs: Für den Händler stellen sich die Daten als anonym dar, während der Zahlungsanbieter pseudonym mit den Waren arbeitet.

Gefahr des Datenmissbrauchs: Durch die strikte Trennung der personenbezogenen Daten von den Transaktionsdaten ist die Gefahr des Datenmissbrauchs als sehr gering einzustufen.

---

zugeordnet werden können, so gering ist, dass sie nach der Lebenserfahrung oder dem Stand der Wissenschaft praktisch ausscheidet.“ Pseudonyme Daten sind anonyme Daten, für die es jedoch eine Zuordnungsregel gibt, nach der sie personenbestimmbar sind. [Scho03]

3.4 Click&Buy

Datenmenge: Durch die langfristige Bindung werden beim Anlegen eines Benutzerkontos Adress- und Bankdaten abgefragt, verifiziert und gespeichert. Jede einzelne Transaktion wird festgehalten und ist für den Nutzer einsehbar.

Dateneinsicht: Gegenüber dem Händler ist das Verfahren sehr datensparsam, aber der Payment Server von Firstgate hat vollen Zugriff auf persönliche Daten und Transaktionsdaten.

Art des Zugriffs: Daraus ergibt sich ein anonymer Datenumgang für den Anbieter der digitalen Waren und ein personenbeziehbarer Datenumgang für den Betreiber des Zahlungssystems.

Gefahr des Datenmissbrauchs: Erfreulich ist zwar dass auch bei diesem Verfahren der Händler keinen Zugang auf persönliche Daten hat, dafür ist dies jedoch beim Payment Server möglich. Da dies vom Umfang allerdings in etwa dem Maß an Datenumgang entspricht, den ein Kunde einem Web-Shop-Betreiber bei Zahlung per elektronischer Lastschrift gewährt, ist die Gefahr des Missbrauchs bei solch einem spezialisierten Unternehmen immer noch als gering einzustufen.

3.5 PayPal

Datenmenge: Auch bei PayPal werden durch die langfristige Bindung und die damit verbundene Profilbildung Adress- und Bankdaten gespeichert. Zusätzlich werden ebenfalls Transaktionsdaten gespeichert und können später vom Nutzer eingesehen werden.

Dateneinsicht: Auch bei Nutzung von PayPal hat der Händler keine Einsicht in persönliche Daten, doch auch hier kann der Anbieter des Systems gleichzeitig auf persönliche Daten und Transaktionsdaten zugreifen.

Art des Zugriffs: Es folgt wieder, dass der Händler einen anonymen Datenumgang und der Systemanbieter einen personenbeziehbaren Datenumgang pflegt.

Gefahr des Datenmissbrauchs: Durch die Abschirmung der personenbezogenen Daten und eine nahezu gleiche Datenmenge wie beim oben genannten System Click&Buy ist das Risiko des Datenmissbrauchs grundsätzlich als gering einzustufen. Allerdings muss dazu gesagt werden, dass es aus datenschutzrechlicher Sicht bedenklich ist, dass sich keines der beiden, in Amerika ansässigen, Mutterunternehmen (*eBay*, *PayPal*) der PayPal Europe Ltd. zu den Grundsätzen der Safe Harbor Vereinbarung bekannt hat. Dies wird dadurch verstärkt, dass Zweifel darüber aufgekommen sind wo die Daten der Nutzer tatsächlich gespeichert werden.[Nödl05]

4 Ausblick

Nachdem die einzelnen Verfahren von Ihrer funktionellen und datenschutzrechtlichen Seite betrachtet wurden, folgt nun die Gesamtbewertung. In Abbildung 5 sind die Verfahren noch einmal gegenübergestellt. Auf der horizontalen Achse ist dabei der Zeitpunkt der Zahlung und auf der vertikalen Achse die Höhe der Zahlung abgetragen. Die Bewertung wird nach funktionellen, wirtschaftlichen und datenschutzrechtlichen Gesichtspunkten vorgenommen.

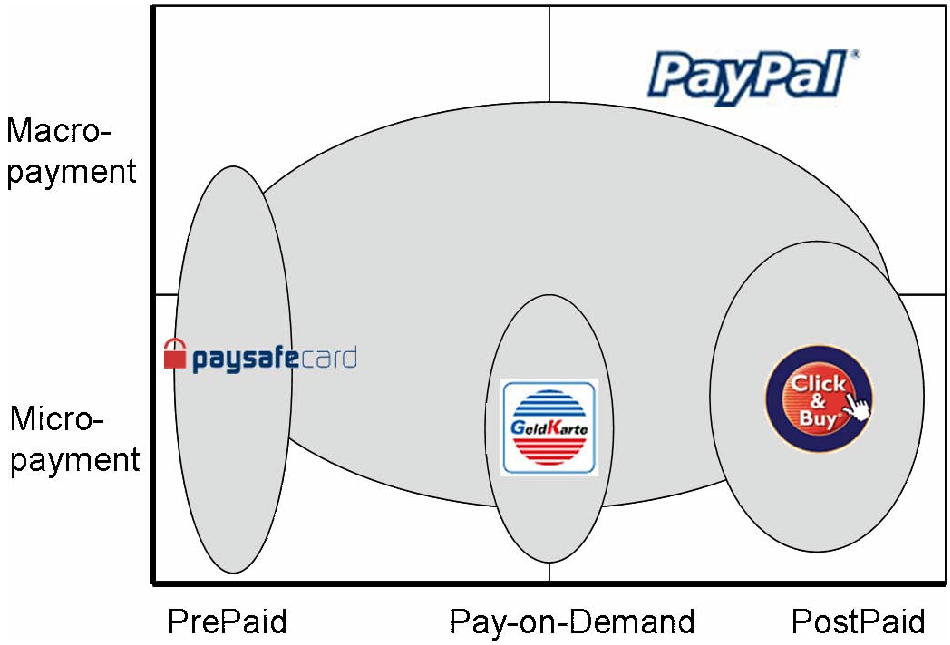


Abbildung 5: Gegenüberstellung der behandelten Verfahren

PaySafeCard: Der bestechende Vorteil der PaySafeCard ist eindeutig die signifikante Datensparsamkeit, die durch ihre Verwendung erzielt wird. Kein anderes Verfahren kann ein ähnlich gutes Ergebnis in diesem Bereich erzielen. Durch den hohen Umsatz, der über die PaySafeCard erzielt wird, ist die Verbreitung der Karte momentan gesichert. Es bleibt jedoch abzuwarten wie hoch der Stellenwert des Datenschutzes für die Nutzer wirklich ist, denn auch wenn die Bezahlung selbst problemlos und unkompliziert abläuft, ist der physische Kauf der Karte ein klarer Minuspunkt.

GeldKarte: Der Einsatz der GeldKarte für das Bezahlen im Internet gewährt zwar einen hohen Datenschutz, aber der wirklichen Verwendung stehen einige Hürden im Weg. Das Hauptproblem sind die hohen Anschaffungskosten der Lesegeräte. Dadurch bedingt ist auch die Zahl der Akzeptanzstellen im Internet sehr klein. Für den Jugendschutz wird sie zwar heute schon intensiv genutzt, aber hierfür können auch deutlich einfachere Lesegeräte genutzt werden. Daher wird die Ausweitung von der Nutzung für den Jugendschutz hin zum elektronischen Bezahlen nicht ohne weiteres erfolgen können.

Click&Buy: Auch wenn dieses Verfahren nicht so datensparsam ist wie die vorhergehenden, bietet es doch einen sehr hohe Datenschutz im Vergleich zu traditionellen Bezahlverfahren. Gerade für den Bereich der Micropayments ist dies eine entscheidender Vorteil. Die Profilbildung erschwert zwar den Einstieg in die Benutzung, sichert dafür aber eine langfristige Kundenbindung, wie von der hohen Nutzerzahl abzuleiten ist.

PayPal: PayPal ist der Branchenprimus im Markt für elektronische Bezahlverfahren. Die schiere Zahl von 100.000.000 Benutzern ist beeindruckend und zeigt, dass das Verfahren schon viele Nutzer durch seinen Leistungsumfang und seine einfache Bedienbarkeit überzeugt hat. Kritisch bleibt, dass hohe Beträge ohne gesonderte Sicherheitsmechanismen wie PIN- und TAN-Nummern versendet werden können. Außerdem wirft die Verschachtelung der Unternehmensstrukturen unweigerliche Fragen bezüglich des Datenschutzes und der Datenverwendung auf.

Aus der Bewertung ergeben sich die folgenden Resultate für die Benutzung der einzelnen Verfahren:

Die GeldKarte, der lange Zeit ein großes Potential für die Bezahlung im Internet nachgesagt wurde, hat es nicht geschafft sich zu etablieren und wird dies, durch die gegebenen Hürden, aller Wahrscheinlichkeit nach auch nicht in Zukunft schaffen. Sie ist, wie viele andere Systeme der zweiten Generation elektronischer Bezahlverfahren schon vor ihr, an der Komplexität der Bedienung gescheitert.

Die PaySafeCard erscheint für den Nutzer sinnvoll, der gelegentlich digitale Waren im Internet kauft und eine hohe Anforderung an den Datenschutz stellt. Daher erscheint es sehr wahrscheinlich, dass dieses System sich weiter durchsetzen wird, da die Sensibilität für den Datenschutzes in Zukunft eher noch zunehmen wird.

Das Verfahren Click&Buy macht Sinn für Nutzer, die häufig im Internet digitale Waren einkaufen und einen hohen Anspruch in Bezug auf ein benutzerfreundliches System haben.

Durch das breite Leistungsspektrum könnte PayPal sich zum alleinigen Zahlungverfahren im Internet durchsetzen. Das selbstgesteckte Ziel von PayPal ist nach Aussage des Gründers Peter Thiel die „*Weltherrschaft im Zahlungsverkehr*“ [NiWe02a]. Ob dieses Ziel erreicht wird, kann zum jetzigen Zeitpunkt noch nicht bewertet werden. Fest steht, dass PayPal einen deutlich schwereren Stand auf dem europäischen Markt hat als auf seinem Heimatmarkt in den Vereinigten Staaten. Die Marktführerschaft von PayPal erscheint jedoch mittelfristig nicht gefährdet.

Grundsätzlich folgt daraus, dass die Entscheidung des Nutzers für ein bestimmtes elektronisches Bezahlverfahren von seinem Anspruch an das Verhältnis zwischen Datenschutz und Einfachheit der Bedienung abhängt. Je mehr Wert auf die Bedienung gelegt wird, desto eher wird er sich für ein System mit langfristiger Bindung entscheiden.

Die generelle Entscheidung für ein elektronisches Bezahlsystem erscheint jedoch in jedem Fall als sinnvoll, da nur noch einer einzigen Stelle im Internet Vertrauen entgegengebracht werden muss. Dadurch ist es nicht mehr notwendig bei jedem Online-Einkauf erneut Zahlungsdaten zu hinterlegen.

Literatur

[AG06] PaySafeCard.com Wertkarten AG. PaySafeCard Website. *www.paysafecard.com/de*, July 2006.

[dDeu05] Hauptverband des Deutschen Einzelhandels (HDE). E-Commerce Umsatz 2006. *www.einzelhandel.de*, December 2005.

[Fink02] Simon Fink. Datenschutz zwischen Staat und Markt. Diplomarbeit, Universität Konstanz, November 2002.

[Geld06] GeldKarte. GeldKarte Website. *www.geldkarte-online.de*, July 2006.

[KrLS06] Malte Krüger, Kay Leibold und Dominik Smasal. Internet Zahlungssysteme aus Sicht der Verbraucher - Ergebnisse der Online-Umfrage IZV8. Technischer Bericht, Universität Karlsruhe (TH), March 2006.

[Maga06] Manager Magazin. Internethandel - Deutschland übernimmt Spitzenposition. *www.manager-magazin.de*, June 2006.

[NiWe02a] Robert Nitschke und Jürgen Weiß. PayPal - The new world currency? Whitepaper, August 2002.

[NiWe02b] Robert Nitschke und Jürgen Weiß. PayPal, das neue Microsoft des Zahlungsverkehrs? Whitepaper, August 2002.

[Nödl05] Jens Nödler. Der rechtliche Rahmen von Zahlungen mittels PayPal. Seminararbeit, January 2005.

[PáNP05] Róbert Párhonyi, Lambert J.M. Niewenhuis und Aiko Pras. Second generation micropayment systems: lessons learned. In *Proceedings of The Fifth IFIP conference on e-Commerce, e-Business, and e-Government (I3E 2005)*, October 2005.

[RoBG03] Alexander Roßnagel, Jürgen Banzhaf und Rüdiger Grimm. *Datenschutz im Electronic Commerce*. Verlag Recht und Wirtschaft. 1. Auflage, 2003.

[Scho03] Philip Scholz. *Datenschutz beim Internet-Einkauf*. Nomos Verlagsgesellschaft. 1. Auflage, 2003.

[Schw04] Christian Schwartze. Sichere Finanztransaktionen im WWW, January 2004.

[Wiki06] Wikipedia. Paypal. *http://de.wikipedia.org/wiki/Paypal*, May 2006.

Abbildungsverzeichnis