UNIVERSITY OF APPLIED SCIENCES

Department of Computer Science

Seminar

IT Sicherheit

Remailer: Typ I bis III

Eingereicht am:

14. November 2015

Eingereicht von:

Mervyn McCreight

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einführung

1.1 Motivation

Bis zur Entwicklung der Typ-0 Remailer[1] lag der Fokus des allgemeinen Interesses darin, den Inhalt einer Nachricht, die über das Internet mit anderen Personen ausgetauscht wird, vor dem Einfluss dritter Personen zu schützen. Dies gelang durch Verschlüsselung des Nachrichteninhaltes mit Hilfe verschiedener gängiger Verschlüsselungsverfahren, die von Zeit zu Zeit immer effektiver wurden. Unter anderem ausgelöst durch die Einführung der Vorratsdatenspeicherung[2] entwickelte sich das Interesse, auch den Absender und/oder Empfänger einer Nachricht vor Außenstehenden zu schützen, um einen Eingriff in die Privatsphäre zu verhindern. Zu diesem Zweck wurden sogenannte *Remailer* entwickelt. Das Ziel von Remailern ist es, Nachrichten und deren Austausch zu entpersonalisieren, sodass Anonymität für Absender und Empfänger erreicht wird.

Die folgenden Kapitel dieser Ausarbeitung werden einen Einblick in das Design der Remailertypen I bis III bieten. Außerdem wird beleuchtet, inwiefern diese verschiedenen Remailertypen nicht nur zeitlich, sondern auch designtechnisch aufeinander eingewirkt haben. Zudem werden die Prinzipien, mit deren Hilfe die verschiedenen Protokolle Anonymität gewärleisten, erläutert und die unterschiedlichen Typen im Bezug auf ihre Sicherheit und Zuverlässigkeit analysiert.

1.2 Sitzungsmodell

Im Rahmen dieser Ausarbeitung wird auf ein einheitliches Sitzungsmodell zurückgegriffen. Dieses Sitzungsmodell modelliert allgemein die Ziele und Eigenschaften, die für den Anwender eines Remailers und dessen Nachricht gelten. Weiterhin umfasst das Modell auch einen potentiellen Angreifer, für den definiert wird, über welches Wissen und welche Fähigkeiten er verfügt.

Das Sitzungsmodell besteht im Wesentlichen aus drei verschiedenen Personen:

Alice möchte eine Nachricht versenden. Für außenstehende Personen soll nicht ersichtlich sein, an wen diese Nachricht gesendet wird.

Bob ist der Empfänger der Nachricht. Für außenstehende Personen soll nicht ersichtlich sein, von wem die Nachricht gesendet wurde.

Eve ist ein Angreifer. Sie möchte die Ziele von Bob und Alice gefährden, also die Anonymität von Alice und Bob aufheben.

[1] Nym-Remailer

[2] verpflichtet unter anderem Internetprovider, den Datenverkehr ihrer Kunden zu protokollieren und über einen bestimmten Zeitraum zu speichern.

Eve stehen dabei eine Vielzahl von Fähigkeiten zur Verfügung. Genauer definiert kann Eve:

- das gesamte Netzwerk beobachten
- den vollständigen Traffic einsehen
- beliebige Pakete abfangen
- beliebige Pakete modifzieren
- beliebige Pakete versenden

Eve versucht, mithilfe der ihr gegebenen Mittel einer Nachrichtenkommunikation einen Absender und einen Empfänger zuzuordnen. Gelingt ihr dies in dem gegebenen Szenario, hat sie die Anonymität von Alice und Bob bezogen auf deren Nachrichtenaustausch aufgehoben.

2 Typ-I Remailer

Anfang der 90er Jahre beschloss eine Interessensgruppe mit dem Namen „Cypherpunk[1]“, das Prinzip eines Remailers aufzugreifen und zu verbessern. Sie entwickelten das Cypherpunk-Remailer Protokoll, das als Typ-I Remailer klassifiziert wird. Ziel der Entwicklung war es, die Unsicherheiten des Typ-0 Remailers zu beseitigen. Anders als bei dem Typ-0 Remailer, der ein pseudonymisierender Remailer ist, handelt es sich beim Typ-I Remailer um einen anonymisierenden Remailer. Das Ziel einer Anonymisierung ist das Verändern personenbezogener Daten in der Art, dass es unmöglich ist, diese Daten einer Person zuzuordnen.[2] Die Anonymisierung bezieht sich in diesem Fall auf den Absender. Jede Information über diesen soll versteckt werden, sodass ein Rückschluss unmöglich ist.[3] Folgerichtig bietet ein anonymisierender Remailer deutlich mehr Geheimnisschutz als ein pseudonymisierender Remailer.

2.1 Mix Netzwerke

Die technische Umsetzung des Cypherpunk-Remailers wurde von der Idee der Mix Netzwerke von David Chaum beeinflusst. Ein Mix-Netzwerk ermöglicht anonyme Kommunikation innerhalb eines Netzwerkes. Der Sender soll gegenüber dem Empfänger verborgen bleiben.[4] Ein Mix-Netzwerk besteht aus einer beliebig großen Menge an Mixen M. Ein Mix in einem Mix-Netzwerk ist üblicherweise ein Server, der von beliebigen Personen betrieben werden kann. Ein Mix fungiert in einem Mix-Netzwerk als Nachrichtenübermittler. Er versendet empfangene Nachrichten derart modifiziert weiter, dass sie nicht mehr auf angenommene Nachrichten zurückzuführen sind. Durch diese Eigenschaft wird Senderanonymität gewährleistet.[5]

2.2 Nachrichtenaustausch

Einige Konzepte des Mix-Netzwerkes wurden aufgegriffen, um den Typ-I Remailer zu entwickeln. Die wichtigsten Aspekte sind das Verschleiern des Nachrichtenweges durch das Senden über beliebige Knoten eines Netzwerks, sowie die schichtenweise Verschlüsselung einer Nachricht.[6] Zur Verschlüsselung der Nachrichten wird das PGP-Verfahren verwendet. Es handelt sich hierbei um ein asymmetrisches Verschlüsselungsverfahren, sodass ein Schlüsselpaar bestehend aus einem privaten und einem öffentlichen Schlüssel benötigt wird. Nachrichten, die über das Typ-I Remailer Protokoll versendet werden, durchlaufen dementsprechend mehrere Remailer. Möchte Alice eine Nachricht N

[1] der Name entspringt dabei einem Wortspiel aus „Cipher“ (engl. für Chiffre) und „Cyberpunk“. Er verdeutlicht das Ziel der Gruppe, über Verschlüsselungstechniken Datenschutz in der elektronischen Datenverarbeitung zu erlangen.
[2] vgl. § 3 Abs. 6 BDSG.
[3] vgl. S. 151 [HF13].
[4] In Mix Netzwerken bleibt zusätzlich noch der Empfänger dem Sender unbekannt. Diese Eigenschaft ist für die weitere Betrachtung dieser Ausarbeitung jedoch nicht wesentlich und wird dementsprechend nicht betrachtet.
[5] für vollständige Informationen über Mix-Netzwerke vgl. [Cha81].
[6] vgl. S. 84 [SS13].

an Bob übermitteln, sucht sie sich aus einer gegebenen Menge von Cypherpunk-Remailern eine endliche Teilmenge $C = (C_1, C_2, ..., C_n)$ an Remailern aus, über die die Nachricht Schritt für Schritt an Bob übertragen wird. Jeder Remailer verfügt für die PGP-Verschlüsselung über je einen öffentlichen Schlüssel E_C und einen privaten Schlüssel D_C. Im Folgenden wird eine Nachricht N, die mit einem öffentlichen Schlüssel E_x verschlüsselt wurde, als $E_x(N)$ bezeichnet. Für die selektierte Teilmenge an Remailern definiert Alice eine Routingreihenfolge $A = (A_1, A_2, ..., A_n, A_{Bob})$. Der letzte Eintrag A_{Bob} ist notwendig, da der letzte Remailer in der Kette die Nachricht an Bob übermitteln muss. Alice verschlüsselt ihre Nachricht zusammen mit den entsprechenden Routing-Informationen nacheinander mit den öffentlichen Schlüsseln E_{C_x} der selektierten Remailer, in rückwärtiger Reihenfolge der Routingordnung, beginnend mit dem letzten Remailer C_n. Dieses Verfahren wird als „schichtenweise Verschlüsselung“ bezeichnet.

$$N' = (A_1, E_{C_1}(A_2, E_{C_2}(...(A_n, E_{C_n}(A_{Bob}, E_{Bob}(N)))))) \tag{2.1}$$

Anschließend initialisiert sie das Versenden der Nachricht, indem sie $E_{C_1}(\ldots)$ an A_1 schickt.

Eine Nachricht, die einen Typ-I Remailer erreicht, enthält, nach Entschlüsselung mithilfe des eigenen privaten Schlüssels D_C, folgende Informationen:

- eine Adresse A_i
- eine (verschlüsselte) Nachricht $E_j(...)$

Der Adressinformation A_i entnimmt der Remailer, an wen die Nachricht $E_j(...)$ weitergeleitet werden soll. Vor dem Weiterleiten der Nachricht entfernt der Remailer alle Informationen, die auf den Sender der Nachricht schließen lassen. Essentiell ist, dass ein Remailer durch die Adressinformation A nur die Adresse des direkten Nachfolgers erhält. Außer Alice ist niemandem der vollständige Pfad des Nachrichtenverlaufs durch das Remailer-Netzwerk bekannt.

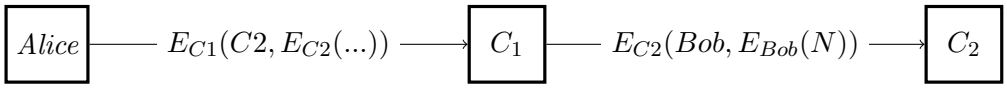

Abbildung 2.1: Schichtenweise Entschlüsselung der Nachricht am Beispiel mit zwei Remailern

Auf diese Weise wird erreicht, dass selbst der Betreiber eines Remailers die Anonymisierung des Absenders nicht aufheben kann. Dieses Verfahren wird fortgeführt, bis die Nachricht über den letzten Remailer C_n der Routingordnung den eigentlichen Empfänger Bob erreicht.[7]

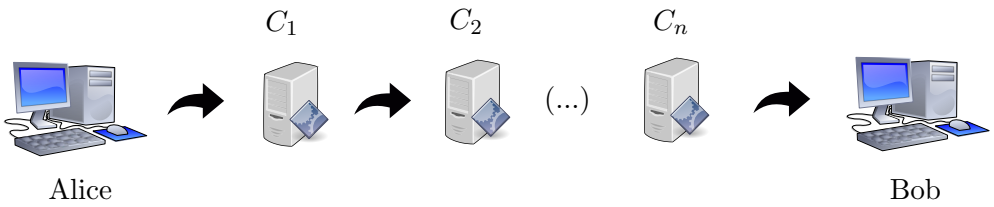

Abbildung 2.2: Exemplarische Nachrichtenübertragung mit Cypherpunk-Remailern

Als Absender ist Bob lediglich die Adresse des Remailers C_n bekannt. Der ursprüngliche Absender der Nachricht, Alice, bleibt verborgen. Daraus folgt, dass es Bob in diesem Protokoll nicht möglich ist, dem Absender einer Nachricht zu antworten.

[7] vgl. S. 72-77 [Kub07].

2.3 Analyse

2.3.1 Sicherheit

Trotz vielversprechender Ansätze sind Cypherpunl-Remailer nicht sicher.[8] Geht man davon aus, dass es Eve als potentiellen Angreifer möglich ist, den vorhandenen Traffic zu analysieren,kann sie den Pfad einer Nachricht nachvollziehen (Traffic-Analyse). Diese Möglichkeit besteht im Wesentlichen aus zwei Gründen:

- Ein Remailer ändert die Größe einer Nachricht nur minimal.
- Ein Remailer leitet die Nachricht nach Empfang sofort weiter.

Die geringfügige Änderung der Nachrichtengröße (lediglich die Routing-Informationen entfallen), ermöglicht eine Zurordnung zwischen eingehenden und ausgehenden Nachrichten. Die sofortige Weiterleitung einer einkommenden Nachricht erleichtert diese Zuordnung ebenfalls. Diese Kenntnisse sind ausreichend um den Verlauf einer Nachricht anhand des Sendezeitpunkts und der Größe zu verfolgen.

Zudem erkennt ein Cypherpunk-Remailer nicht, ob eine Nachricht, die er empfängt, von ihm bereits empfangen und bearbeitet wurde. Das ermöglicht Replay-Angriffe, bei denen Eve eine Nachricht, die sie von Alice abgefangen hat, beliebig häufig in das Remailer-Netzwerk einspielt, um den Nachrichtenverlauf zu analysieren und so auf den Empfänger der Nachricht zu schließen.

2.3.2 Zuverlässigkeit

Der gewählte Aufbau des Cypherpunk-Remailer Protokolls führt außerdem dazu, dass der Absender einer Nachricht nie weiß, ob der Empfänger die Nachricht tatsächlich empfangen hat. Ist einer der gewählten Remailer, über den die Nachricht weitergeleitet werden soll, defekt oder nicht zu den anderen Remailern kompatibel, bricht die Übertragungskette an dieser Stelle und die Nachricht wird nicht zugestellt. Dadurch, dass zu dem Zeitpunkt der Kette weder Sender noch Empfänger bekannt sind, ist es nicht möglich, das Fehlverhalten zu signalisieren.

[8] sicher insofern, dass die Anonymität eines Absenders gewährleistet wird.

3 Typ-II Remailer

3.1 Motivation

Das Mixmaster-Remailer Protokoll ist seit 1995 verfügbar. Die Motiviation und gleichzeitig das Ziel der Entwicklung der Typ-II Remailer war, die Schwächen der Typ-I-Remailer Generation zu beseitigen. Die wesentlichen Konzepte zur Umsetzung und Verbesserung wurden von Lance Cottrell in seiner Ausarbeitung „Mixmaster and remailer attacks“ erarbeitet. Darin legt er die Schwächen der Cypherpunk-Remailer offen und analysiert, wodurch diese entstehen, und erarbeitet konkrete Vorschläge, wie die vorhandenen Sicherheitslücken zu schließen sind. Er stellt dar aus welchem Grund Cypherpunk-Remailer nicht zuverlässig verhindern, dass eine Verbindung zwischen eingehenden und ausgehenden Nachrichten an einem Knoten im Remailer-Netzwerk hergestellt werden kann.[1] Damit entwarf und implementierte Lance Cottrell das erste Design des Mixmaster Protokolls.[2]

3.2 Funktionsweise

Das Mixmaster-Remailer Protokoll basiert, analog zum Cypherpunk-Remailer Protokoll, auf einem Netzwerk von Remailern, ähnlich einem Chaum'schen Mix-Netzwerk. Das Verfahren, nach dem eine Nachricht die verschiedenen Knoten des Netzwerks traversiert, bleibt größtenteils identisch. Auch hier wird ein asymmetrisches Verschlüsselungsverfahren verwendet, auf dessen Basis eine Nachricht entsprechend der öffentlichen Schlüssel der Remailer schichtenweise verschlüsselt wird. Der Pfad einer Nachricht durch das Netzwerk muss zum Zeitpunkt der schichtenweisen Verschlüsselung vollständig bekannt sein. Weiterhin manipuliert ein Remailer nach dem Empfangen und Entschlüsseln einer Nachricht den Nachrichtenheader, um den Absender der Nachricht unkenntlich zu machen. Auf diese Weise wird jede Art von absenderbezogenen Informationen entfernt und die Nachricht anonymisiert.

Bisher sind alle Schritte identisch mit dem Cypherpunk-Remailer Protokoll. Um die Schwächen der Typ-I Remailer zu entfernen, bedurfte es der Einführung zusätzlicher Sicherheitsmaßnahmen, die im Folgenden erläutert werden.

Da die Aufbereitung einer Nachricht für das Mixmaster-Remailer Protokoll durch die zusätzlich eingeführten Sicherheitsmechanismen sehr komplex geworden ist, existiert Client-Software für das Mixmaster-Remailer Protokoll, die das Vorbereiten einer Nachricht für einen Anwender übernehmen. Da Mixmaster-Remailer nur noch Nachrichten akzeptieren, die genau dem spezifizierten Protokoll entsprechen, ist die Verwendung einer Client-Software notwendig. Möchte Alice eine Nachricht über Mixmaster-Remailer an Bob senden, muss sie anders als bei Cypherpunk-Remailern die Vorarbeit nicht manuell durchführen. Stattdessen verwendet sie die Client-Software.

[1] vgl. S. 276 [Ora01].
[2] vgl. [mix08] - zuletzt aufgerufen am 17.10.2015.

3.2.1 Chunks

Im Mixmaster-Remailer Protokoll werden Nachrichten in gleichgroße Blöcke[3] aufgeteilt (beispielsweise 20 kB groß). Entstehen dabei einer oder mehrere Blöcke, die nicht die gewünschte Größe haben, werden diese mit zufällig generierten Daten aufgefüllt. Anstelle der vollständigen Nachricht werden die verschiedenen gleichgroßen Blöcke einer Nachricht über das Remailer-Netzwerk verteilt. Zusammengehörende Blöcke müssen nicht denselben Pfad durch das Netzwerk nehmen. Es ist vorteilhaft, wenn die Blöcke über unterschiedliche Remailer in dem Netzwerk verteilt werden. Wichtig ist, dass der letzte Remailer, der die Nachricht an den Empfänger überträgt, für alle Blöcke einer Nachricht identisch ist. Nur dieser Remailer ist dazu in der Lage, die vollständige Nachricht wiederherzustellen, sofern er alle der Nachricht zugehörigen Blöcke empfangen hat. Anschließend leitet er die Nachricht an den Empfänger weiter.

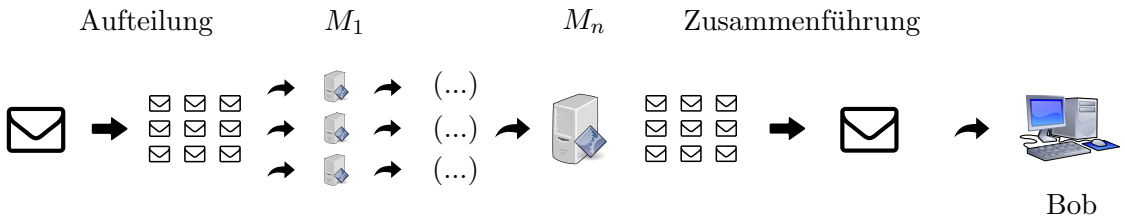

Abbildung 3.1: Exemplarische Nachrichtenübertragung mit Mixmaster-Remailern

Dieses Verfahren sorgt dafür, dass eine Nachricht nicht mehr anhand ihrer Größe durch das Remailer-Netzwerk verfolgt werden kann. Alle Nachrichten, die innerhalb des Remailer-Netzwerks übertragen werden, sind von gleicher Größe und Eve identisch. Eine Zuordnung ist auf diese Weise nicht mehr möglich.

3.2.2 Pool

Anders als bei den Cypherpunk-Remailern werden einkommende Nachrichten bei den Mixmaster-Remailern nicht sofort zum Zeitpunkt des Eintreffens weitergeleitet. Stattdessen speichert ein Mixmaster-Remailer seine einkommenden Nachrichten in einem Nachrichtenspeicher zwischen. Dieser Nachrichtenspeicher wird auch Nachrichtenpool genannt. Darin werden einkommende Nachrichten gesammelt. Wichtig ist, dass die Reihenfolge der Nachrichtenspeicherung hierbei keinem festen Schema folgt. Einkommende Nachrichten werden in zufälliger Reihenfolge im Nachrichtenpool abgelegt. Für jeden Remailer ist ein Größenschwellwert für den Nachrichtenpool individuell konfigurierbar. Zu dem Zeitpunkt, an dem die Größe des Nachrichtenpools diesen Schwellwert übersteigt, werden in ihm befindliche Nachrichten in zufälliger Reihenfolge an ihren entsprechenden Empfänger weitergeleitet.

Vorstellbar ist, dass ein Remailer nie genügend Nachrichten empfängt um seinen Nachrichtenpool ausreichend zu füllen. Damit die bis dahin im Nachrichtenpool gesammelten Nachrichten nicht blockiert werden, wird nach Ablauf eines individuell festlegbaren Zeitintervalls der Nachrichtenpool um zufällig generierte Pseudonachrichten erweitert, sodass der Größenschwellwert überschritten wird. Anschließend werden alle in ihm befindlichen Nachrichten, inklusive der Attrappen, weitergeleitet.

Auf diese Weise kann Eve eine Nachricht nicht mehr über eine zeitliche Zuordnung verfolgen. Eine Verbindung zwischen Empfangszeitpunkt und Absendezeitpunkt einer Nachricht an einem Remailer kann nicht hergestellt werden. Durch die gleichartigen Nachrichten[4] ist es einem Angreifer zusätzlich

[3] auch "chunks".
[4] gleichartig bezogen auf ihre Größe.

nicht möglich, überhaupt eine Verbindung zwischen einer einkommenden und einer ausgehenden Nachricht herzustellen.

3.2.3 Signatur

Weiterhin wurde beim Mixmaster-Remailer die Überprüfung der Integrität einer im Remailer-Netzwerk verschickten Nachricht als zusätzlicher Sicherheitsmechanismus eingeführt. Dazu wird in einem verschlüsselten Header der Nachricht eine Signatur übertragen. Mithilfe der Signatur prüft ein Remailer den Inhalt der Nachricht auf Manipulation oder vollständige Fremdeinführung. In der Konsequenz ist Eve außerstande, manipulierte Nachrichten in das Remailernetzwerk einzuspielen, um das Verhalten auf diese Nachrichten zu analysieren.

3.2.4 Identifikation

Zusätzlich zur Signatur enthält eine Nachricht in einem weiteren verschlüsselten Header eine eindeutige Identifikation, üblicherweise eine Nummer. Über diese ID kann ein Remailer eine Nachricht eindeutig erkennen und ermitteln, ob er dieselbe Nachricht mehrfach empfängt. Hat ein Mixmaster-Remailer eine Nachricht bereits empfangen und sie in seinem Nachrichtenpool abgelegt, oder sogar weitergeleitet, wird er jede weitere Nachricht mit der selben Identifikation ignorieren. Durch diesen Sicherheitsmechanismus hat Eve nicht mehr die Möglichkeit, die bei Cypherpunk-Remailern denkbaren, Replay-Angriffe erfolgreich zu fahren. Fängt Eve eine Nachricht von Alice an einen Remailer ab, und versucht durch mehrmaliges Absenden der Nachricht an einen Remailer das Verhalten zu analysieren, schlägt der Angriff fehl, da mehrfach gesendete Nachrichten ignoriert werden.

3.3 Sicherheitsanalyse

Das Mixmaster-Remailer Protokoll heute als sicher. Dies spiegelt sich auch in der aktuellen Benutzung von Remailern wieder. Mixmaster-Remailer sind das aktuell am meisten verwendetet Remailer-Protokoll.

Aus theoretischer Sicht sind auch Mixmaster-Remailer nicht in der Lage vollständige Anonymität zu garantieren. Eve ist es mittels einer Flooding-Attacke möglich, eine Nachricht vom Sender durch das Remailer-Netzwerk bis zum Empfänger zu verfolgen. Dabei fängt Eve die Nachricht von Alice an den ersten Remailer-Netzwerkknoten ab und hält diese zurück. Anschließend überflutet[5] sie das Remailer-Netzwerk solange mit eigenen Nachrichten, bis die Nachrichtenspeicher der einzelnen Remailer aussschließlich mit Eves Nachrichten gefüllt sind. Nun ist jeder Nachrichtenverkehr im Remailer-Netzwerk auf Nachrichten von Eve zurückzuführen. Eve ist in der Lage, jede dieser Nachrichten als ihre zu identifizieren, da sie den Empfänger ihrer eigenen Nachricht kennt. Ist dieser Status erreicht, versendet Eve die von Alice abgefangene und zurückgehaltene Nachricht. Da Eve alle anderen Nachrichten im Remailer-Netzwerk kennt, kann sie die Nachricht von Alice über das gesamte Netzwerk bis zum Empfänger Bob zu verfolgen. Eve muss nur auf eine Nachricht warten, die nicht an den Empfänger ihrer Nachricht versendet wird. Der Empfänger dieser Nachricht muss der Empfänger der Nachricht von Alice, also Bob, sein.

Ein solcher Angriff ist in der Praxis nicht umsetzbar, da davon auszugehen ist, dass Mixmaster-Remailer zu jeder Zeit auch von fremden Personen verwendet werden. Zusätzlich müsste Eve nicht nur alle Remailer des Netzwerks kennen, sondern auch deren individuellen Nachrichtenpoolgrößen.

[5] daher der Name "Flooding-Attacke".

All diese Fakten zu erlangen ist für Eve in der Praxis kaum realisierbar. Dadurch kann Eve zu keinem Zeitpunkt den Zustand herbeiführen, in dem nur ihre Nachrichten im Remailernetwerk vorhanden sind.

4 Nym Server

4.1 Motivation

In den vorherigen Kapiteln wurden die Typ-I und Typ-II Remailer vorgestellt. Bei beiden handelt es sich um anonymisierende Remailer. Die Konsequenz ist, dass die Identität des Senders zwar verborgen bleibt, es auf diesem Weg jedoch unmöglich ist einem Absender einer Nachricht zu antworten. Um diese Funktionalität anbieten zu können, wurden Nym Server entwickelt.

Ein Nym Server ist ein Pseudonym-Server, der es seinen Benutzern erlaubt, ein Pseudonym zu hinterlegen. Die Benutzer sind über ihr Pseudonym auf dem Nym Server für andere Personen erreichbar, ohne dass andere Personen die reale Identität, die sich hinter dem Pseudonym verbirgt, erfahren.

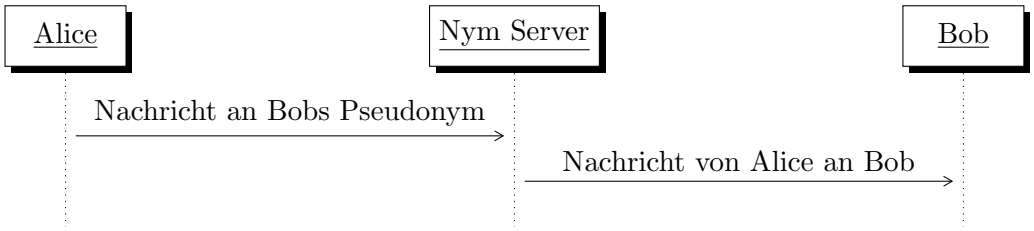

Abbildung 4.1: Senden einer Nachricht über einen Nym-Server (vereinfacht)

Beim Typ-0 Remailer war es bereits möglich, einem Absender einer Nachricht auf diese zu antworten. Grund dafür ist, dass der Typ-0 Remailer ebenfalls pseudonymisierend war. Er ersetzte die Absenderadresse durch ein Pseudonym und besaß eine Zuordnungstabelle zwischen Pseudonymen und tatsächlichen Identitäten. Dadurch war es einem Typ-0 Remailer möglich, Nachrichten, die an ein Pseudonym gesendet wurden, an die tatsächliche Adresse weiterzuleiten. Der große Nachteil dieser Technik ist, dass einem Betreiber eines Typ-0 Remailers eine Zuordnung zwischen verwendeten Pseudonymen und Identität vorliegt. Einerseits stellt diese Zuordnung eine Schwachstelle dar, da ein Angreifer dazu in der Lage ist diese Zuordnung ebenfalls zu erhalten. Andererseits macht sich der Betreiber selbst angreifbar. So kann er beispielsweise durch staatliche Behörden dazu gezwungen werden, die Identität eines Pseudonyms preiszugeben. Per definitionem handelt es sich daher bei Typ-0 Remailern nicht um Nym-Server.[1]

4.2 Umsetzung mit Hilfe von Typ-I Remailern

Das Ziel von Nym-Servern ist, Pseudonymisierung zu gewährleisten. Dabei soll der Betreiber des Nym-Servers keine Kenntnis von den tatsächlichen Identitäten, die hinter den Pseudonymen stehen, haben. Die Grundidee ist, dieses mit Hilfe des Typ-I-Remailer Protokolls zu erreichen.

[1] vgl. [nym].

Im Unterschied zu Typ-0 Remailern, die eine direkte Zuordnung zwischen Pseudonym und Identität speichern, beschränken sich Nym Server auf die Speicherung von Nyms. Ein Nym besteht aus folgenden Informationen[2]:

- ein öffentlicher Schlüssel
- ein Reply-Block
- ein Pseudonym

Der Reply-Block beinhaltet einen Pfad durch ein Typ-I Remailernetzwerk zum eigentlichen Empfänger, der sich hinter dem Nym verbirgt.

Möchte Bob ein Pseudonym bei einem Nym Server erstellen, muss er zuerst einen Reply-Block erzeugen. Anschließend schickt er diesen Reply-Block zusammen mit einem gewünschtem Pseudonym und einem öffentlichen Schlüssel an den Nym Server. Der Nym Server legt beim Empfang dieser Nachricht ein Nym mit den empfangenen Daten an.[3] Wichtig ist, dass Bob die Nachricht anonymisiert an den Nym-Server sendet, damit dem Nym-Server zu keinem Zeitpunkt die wahre Identität von Bob bekannt ist. Nun ist Bob für Alice über das angegebene Pseudonym für Alice erreichbar.

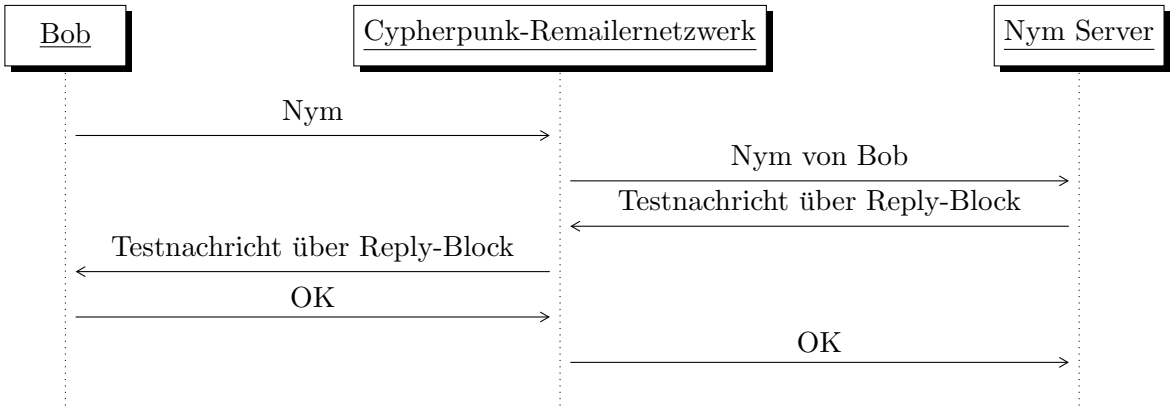

Abbildung 4.2: Registrierung eines Nyms

Zur Vereinfachung des Diagramms befindet sich der erste Remailer von Bobs Reply-Block in der obigen Abbildung im selben Remailernetzwerk, über das Bob seine Nymregistrierungsdaten anonymisiert an den Nym Server überträgt.

Möchte Alice Bob eine Nachricht zukommen lassen, schickt sie die Nachricht an Bobs Pseudonym auf dem entsprechenden Nym-Server. Empfängt der Nym-Server eine Nachricht, die an ein ihm bekanntes Pseudonym gerichtet ist, verschlüsselt er die Nachricht mit dem im Nym angegebenen öffentlichen Schlüssel und reicht sie zusammen mit dem hinterlegten Reply-Block an den ersten Remailer der im Reply-Block angegebenen Kette von Remailern weiter. Für die eigentliche Zustellung der Nachricht sind die im Reply-Block angegebenen Remailer zuständig. Die Nachricht wird entsprechend des Typ-I-Remailer Protokolls an Bob übertragen.[4] Erhält Bob die Nachricht, entschlüsselt er sie mit seinem privaten Schlüssel und verfügt nun über die Nachricht.

[2] vgl. S. 25 [Loe09]
[3] vgl. S. 25 [Loe09].
[4] vgl. S. 26 [Loe09].

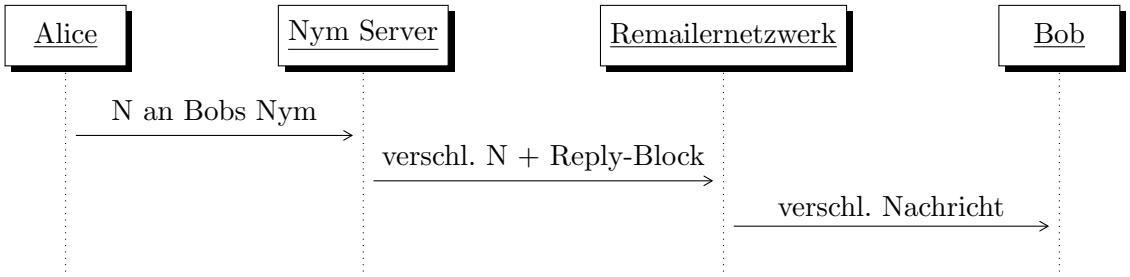

Abbildung 4.3: Weiterleitung einer Nachricht an ein Pseudonym

Nur dem letzten Remailer des Reply-Blocks ist die echte Adresse von Bob bekannt. Da der Reply-Block selbst schichtenweise verschlüsselt ist, kann der Block selbst nur im Verlauf des Pfades durch das Netzwerk entschlüsselt werden. Der Nym Server kann den Block nicht entschlüsseln. Somit hat der Nym Server nicht die Möglichkeit die wahre Identität von Bob zu erfahren.

Die sicherheitsbezogenen Schwächen dieses Verfahrens korrespondieren mit denen des Typ-I Remailer Protokolls, da dieses verwendet wird um die Anonymität von Bob gegenüber dem Nym Server und Eve zu gewährleisten.

Eine Lösung über das Typ-II Remailer Protokoll ist nicht möglich. Der Grund liegt im Mixmaster-Remailer Protokoll vorhandene Integritätscheck der zu übertragenden Nachrichten. Das Design des Mixmaster-Protokolls erfordert zum Zeitpunkt der schichtenweisen Verschlüsselung für die Erstellung der Signaturen, die Kenntnis der zu versendenden Nachricht. Im genannten Nym Server Verfahren ist die Nachricht jedoch erst nach der Erstellung des Reply-Blocks bekannt.

5 Typ-III Remailer

Die als Typ-II klassifizierten Mixmaster-Remailer gelten in der Praxis als sicher. Trotzdem wurde das Mixminion-Remailer Protokoll entworfen. Mixminion-Remailer stellen den Typ-III der Remailerklassifizierung dar. Die signifikanten Neuerungen, die mit dem Mixminion-Remailer Protokoll eingeführt wurden, waren:

- das Antworten auf anonyme Nachrichten
- verschlüsselte Kommunikation zwischen den Remailern (TLS statt SMTP)
- Einführung einer Verzeichnisserverstruktur

Die Möglichkeit auf eine empfangene anonyme Nachricht zu antworten, ist der entscheidende Vorteil der Mixminion-Remailer. Kein anderes anonymisierendes Remailer Protokoll (Typ-I und Typ-II) unterstützt dieses Feature nativ.

Die Mixminion-Remailer sind, äquivalent zu den vorherigen Remailer Protokollen, eine Weiterentwicklung des vorherigen Protokolls. So bilden die Mixmaster-Remailer die technische Grundlage für die Entwicklung des Mixminion Protokolls.

5.1 Funktionsweise

5.1.1 SURBs

Die Single-Use-Reply-Blocks[1] wurden eingeführt, um dem Empfänger das Antworten auf eine empfangene anonyme Nachricht zu ermöglichen. Dabei soll die Anonymität des Senders erhalten bleiben.

Betrachtet man die Lage basierend auf den vorherigen Remailer-Protokollen, kann Bob nicht auf eine über ein Remailer-Netzwerk versandte Nachricht von Alice antworten. Das liegt daran, dass ihm lediglich die Identität des letzten Remailers bekannt ist, nicht jedoch die von Alice.

Um Bob ein Antworten zu ermöglichen, erstellt Alice einen Single-Use-Reply-Block und hängt diesen an die Nachricht an. Der SURB befindet sich dabei in verschlüsselter Form im Header der Nachricht. Ein SURB enthält zwei wichtige Informationen:

- die E-Mail Adresse von Alice in verschlüsselter Form
- einen Pfad durch das Remailer-Netzwerk zu Alice

[1] kurz: SURBs.

Wichtig ist, dass der SURB für den Empfänger nicht entschlüsselbar ist. Bob ist weiterhin nicht in der Lage die Identität von Alice zu entschlüsseln. Dazu ist lediglich der entsprechende Mixminion-Remailer fähig. Bob kann jedoch den SURB dazu verwenden, um Alice auf ihre anonyme Nachricht zu antworten. Dazu hängt Bob seine Antwortnachricht an den SURB an und verschickt dieses Konstrukt an den ersten Remailer des im SURB vorhandenen Remailerpfades. Der letzte Remailer dieses Pfades kann die Adresse von Alice entschlüsseln und die Antwort an Alice weiterleiten.

Bob kann pro empfangenen SURB nur eine Nachricht an Alice schicken. Das liegt darin begründet, dass ein SURB nur einfach verwendbar ist.[2] Möchte Bob Alice eine weitere Nachricht schicken, muss er auf eine weitere Nachricht von Alice warten, die einen SURB enthält, um diesen für eine erneute Antwort zu verwenden. Nach der einmaligen Verwendung eines SURBs werden alle weiteren Nachrichten, die mit Hilfe dieses SURBs versendet werden, wie Duplikate betrachtet und verworfen.

5.1.2 Nachrichten

Typisierung

Bisher existierte in den Remailer-Protkollen nur eine Art von Nachricht – eine anonyme Nachricht von Alice an Bob. Für die Hinzunahme von Antwortnachrichten, die gesondert behandelt werden müssen, ist die Einführung einer Typisierung von Nachrichten für das Mixminion-Remailer Protokoll unerlässlich. Hierbei wird zwischen drei Arten von Nachrichten unterschieden:[3]

1. normale Nachrichten
2. direkte Antworten über SURBs
3. anonyme Antworten

Eine normale Nachricht entspricht einer anonymisierten Nachricht von Alice an Bob, wie sie aus den bisherigen Protokollen bekannt ist. Bei einer direkten Antwort über einen SURB gibt Bob bei der Antwort seine Identität preis. Da er nicht weiß, ob er wirklich Alice antwortet – denkbar wäre auch, dass Alice die Adresse eines Dritten in dem SURB angegeben hat – ist es wünschenswert, dass Bob ebenfalls die Möglichkeit hat bei einer Antwort anonym zu bleiben.[4] Hierfür existiert die dritte Art einer Nachricht, bei der die Identität von Bob ebenfalls verborgen wird.

Ununterscheidbarkeit

In den vorherigen Kapiteln wurde darauf hingewiesen, dass es für die Gewährleistung der Sicherheit unerlässlich ist, dass der Datenverkehr innerhalb des Remailernetzwerks für Eve transparent erscheinen muss. Eve darf nicht dazu in der Lage sein, durch die Analyse des Datenverkehrs eine Verbindung zwischen einkommenden und ausgehenden Nachrichtenvekehr herstellen zu können.

Dieser Zustand muss auch nach der Einführung verschiedener Nachrichtentypen erhalten bleiben. Die Nachrichtentypen müssen daher nach außen hin ununterscheidbar sein. Das ist wichtig, damit Eve keine Rückschlüsse bezüglich der Art des Nachrichtenverkehrs über die Unterscheidung der Nachrichtentypen ziehen kann. Diese Transparenz wird dadurch gewährleistet, dass alle drei

[2] daher SINGLE-USE-Reply-Block.
[3] vgl. S. 4 [DDM].
[4] das ist nur ein beispielhafter Grund. Logischerweise könnte Bob auch aus anderen Gründen anonym bleiben wollen.

Nachrichtenarten strukturell identisch aufgebaut sind. Sie verfügen über gleichgroße Header- und Bodygrößen.[5]

Strukturen der unterschiedlichen Typen

Die Nachrichten die in einem Mixminion-Remailer Netzwerk verschickt werden, sind ingesamt 32kB groß. Durch diese Gleichförmigkeit der Größe liefert eine Analyse des Traffic keinen Aufschluss darüber, welche Arten von Nachrichten gerade in dem Remailer-Netzwerk ausgetauscht werden.

Jede Nachricht im Mixminion-Protokoll besteht grundlegend aus den Komponenten

- Header
  - primärer Header
  - sekundärer Header
- Body (Payload)

Signifikant ist, dass der Header sich in zwei Teile aufteilt – dem primären und dem sekundären Header. Dabei haben der primäre und der sekundäre Header jeweils eine Größe von 2kB und der Body eine Größe von 28kB. Der Body einer Nachricht enthält, unabhängig vom Typ der Nachricht, die eigentlich zu übertragende Information; im letzten Schritt also die Nachricht des Senders an den Empfänger. Die Typisierung einer Nachricht wird durch den Inhalt des primären und sekundären Headers unterschieden.

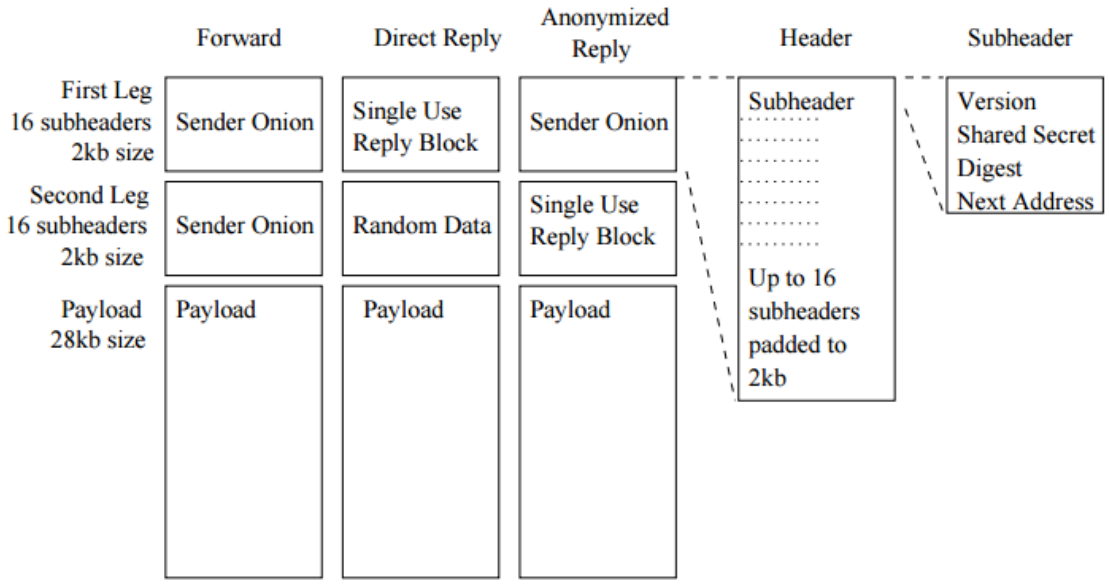

Abbildung 5.1: Die Struktur von Nachrichten im Mixminion-Protokol

Source: S. 4 [DDM]

Die obige Abbildung beschreibt den Inhalt des primären und des sekundären Headers in den verschiedenen Nachrichtenfällen. Man sieht, dass sowohl der primäre als auch der sekundäre Header im Fall einer normalen Nachricht, als auch der primäre Header im Falle einer anonymen Antwort, Pfadinformationen durch das Netzwerk beinhalten. Ein solcher Pfad ist in Subheader aufgeteilt.

[5] Body wird häufig auch Payload genannt.

Jeder Subheader entspricht einem Datum für den nächsten Remailer des Pfades und beinhaltet im wesentlichen drei Daten:

- ein Master-Secret für die Erstellung eines symmetrischen Schlüssels für den Aufbau einer verschlüsselten Verbindung zum nächsten Remailer im Pfad
- eine Adresse des nächsten Remailers im Pfad
- eine Prüfsumme zum Überprüfen der Integrität des Rests des Headers

Die Pfade des primären und sekundären Headers ergeben, im Falle einer normalen Nachricht, zusammen den Gesamtpfad der Nachricht durch das Remailernetzwerk zum Empfänger. Dabei entspricht der Pfad im primären Header dem ersten Teil, der Teil im sekundären Header dem zweiten Teil des Pfades. Sowohl der primäre als auch der sekundäre Header können sich maximal aus 16 dieser Subheader zusammensetzen. Demzufolge ist die maximale Länge der Route einer normalen Nachricht im Mixminion-Protokoll 32.

Bei einer direkten Antwort von Bob ist der SURB im primären Header der, der von Alice zum Antworten übermittelt wurde. Da in diesem alle Informationen enthalten sind, die benötigt werden um die Nachricht bis zu Alice durchzustellen, befinden sich im sekundären Header in diesem Fall keine signifikanten Informationen. Um den gleichartigen Schein zu wahren, wird der sekundäre Header mit Platzhalterdaten gefüllt.

Bei einer anonymen Antwort befindet sich der SURB von Alice im sekundären Header. Im primären Header befindet sich ein von Bob spezifizierter Pfad durch das Netzwerk, der vor dem Pfad des SURBs durchlaufen wird. Dadurch bleibt die Identität von Bob dem Empfänger der Antwort verborgen.[6]

5.1.3 Verzeichnisserver

Eine weitere Neuerung im Mixminion-Protkoll ist die Einführung von Verzeichnisservern. Ein Verzeichnisserver trägt für jeden Mixminion-Remailer im Netzwerk drei unmittelbar relevante Informationen:[7]

- Die Existenz eines Remailers
- Den aktuellen Schlüssel des Remailers
- Den aktuellen Status des Remailers

Alle drei Eigenschaften werden den Verzeichnisservern von den Remailern selbst mitgeteilt. Wichtig ist, dass es nicht nur einen Verzeichnisserver pro Netzwerk gibt. Pro Netzwerk gibt es mehrere Verzeichnisserver. Diese verschiedenen Verzeichnisserver müssen dauerhaft synchronisiert sein, damit sichergestellt ist, dass sie die gleichen Daten bezüglich des Remailernetzwerks verteilen (Redundanz). Dadurch wird auch verhindert, dass nicht funktionstüchtige Remailer weiter von einem Benutzer in deren Nachrichtenpfad eingebaut werden. Dies würde dazu führen, dass eine Nachricht nie ihren Empfänger erreichen würde. Des weiteren haben die Verzeichnisserver die Aufgabe, sich dauerhaft gegenseitig zu verifizieren. Dadurch wird verhindert, dass ein Angreifer einen manipulierten Verzeichnisserver einspielt, um beispielsweise alle Daten nur über bestimmte Remailer laufen zu lassen.[8] Bei dieser Vorgehensweise wird davon ausgegangen, dass nicht alle Verzeichnisserver manipuliert sind, da ansonsten die gegenseitige Verifikation und Synchronization hinfällig wäre.

[6] vgl. S. 4 [DDM].
[7] vgl. S. 8 [DDM].
[8] vgl. S. 9 [DDM].

5.2 Ablauf

Möchte Alice eine Nachricht an Bob senden, benötigt sie zunächst alle nötigen Informationen vom Verzeichnisserver. Von diesem erhält sie zu jedem Remailer im Netzwerk dessen Status und aktuellen Schlüssel. Die schichtweise Verschlüsselung der Nachricht geschieht äquivalent zum Mixmaster Protokoll. Zusätzlich wird der sekundäre Header mit der Prüfsumme des Nachrichtenbodies verschlüsselt. Anschließend sendet sie die Nachricht an den ersten Remailer im Pfad.

Empfängt nun ein Remailer eine Nachricht, wird zunächst die Integrität der Daten über die im Subheader angegebene Prüfsumme überprüft. Anschließend baut er mit Hilfe des ebenfalls im Subheader angegebenen Master-Secrets eine gesicherte Verbindung (TLS) zum nächsten Remailer im Pfad auf und überträgt die Nachricht an diesen, nachdem er sie entsprechend der schichtweisen Verschlüsselung für seinen Teil entschlüsselt hat. Danach wird die gesicherte Verbindung wieder aufgelöst.

Die Stelle, an der der gesamte Pfad des primären Headers durchlaufen ist, wird "CrossoverPunkt genannt. An diesem Punkt wird der sekundäre Header mit Hilfe der Prüfsumme des Nachrichtenbodies entschlüsselt und der primäre mit dem sekundären Header vertauscht[9]. Wurde eine Nachricht in der Zwischenzeit in irgendeiner Form manipuliert, ändert sich die Prüfsumme des Nachrichtenbodies und der sekundäre Header ist nicht wiederherstellbar. In diesem Fall wird die Nachricht an diesem Punkt verworfen[10]. Dadurch werden Attacken, die ein manipulieren der Nachricht benötigen, verhindert. Nachdem der sekundäre Header erfolgreich entschlüsselt und entsprechend vertauscht wurde, wird wie im vorherigen Ablauf weiter verfahren, bis die Nachricht letztendlich zu Bob weitergeleitet wird.

5.3 Sicherheitsanalyse

In der Theorie handelt es sich bei dem Mixminion-Protokoll um das sicherste der drei hier betrachteten Remailer-Protokolle. Beim Design wurden neuste Forschungsergebnisse und durch die älteren Remailer-Protokolle gesammelten Erfahrungen genutzt, um sich gegen bekannte typische Angriffe gegen Remailer zu schützen[11]. Ebenso wurden viele Mängel der früheren Remailer-Protkolle herausgearbeitet und beseitigt.

Das Mixminion Protokoll ist jedoch nie in einem vollständigen Zustand implementiert worden. Es ist seit je her nie über die Beta-Phase der Implementierung hinaus gekommen. So existieren unter Umständen noch Fehler in der Implementierung, die die Sicherheit des Systems gefährden. Außerdem laufen viele Mixminion-Remailer aufgrund ihrer unfertigen Implementierung noch mit Debug-Einstellungen, sodass Aktivitäten geloggt werden. Dieser Zustand gewährleistet keine Sicherheit, insofern wird das Protokoll in der Praxis nicht aktiv genutzt. Dadurch konnten bisher auch keine praktischen Erfahrungen bzgl. der Sicherheit dieses Remailer-Protokolls gesammelt werden.

[9] dieser Vorgang wird als swap operation bezeichnet. Vgl. S. 4-5 [DDM]
[10] vgl. S. 5 [DDM]
[11] vgl. S. 5ff [DDM]

6 Zusammenfassung

In den vorausgegangenen Kapiteln wurde die Funktionsweise der Protokolle der verschiedenen Remailertypen I - III betrachtet. Die Protokolle wurden insbesondere aus sicherheitstechnischer Sicht analysiert. So wurde für jeden Remailertyp behandelt, ob er gegen gängige Angriffsarten resistent ist, und welche Sicherheitsmechanismen entwickelt wurden, um eine Immunität der verschiedenen Angriffsarten zu gewährleisten. Da diese drei Remailertypen auf dem Prinzip der Mixnetzwerke basieren, wurden Angriffstypen gewählt, die Mixnetzwerke potentiell gefährden können.

Die verschiedenen Remailer-Protokolle unterscheiden sich im Umfang der Funktionalitäten, die das Protokoll nativ zur Verfügung stellt. Das Typ-I und Typ-II Protokoll beschränkt sich auf die reine Funktionalität eines Remailers. Es spezifiziert, wie eine Nachricht anonymisiert übertragen werden kann. Im Typ-III Protokoll ist zusätzlich die Möglichkeit des Antwortens auf anonymisierte Nachrichten enthalten. Dies war für das Typ-I Protokoll nur über einen zusätzlichen Nym-Server möglich. Des weiteren ist ein Verzeichnis, in dem alle verwendbaren Remailer des entsprechenden Typs inklusive aller zur Benutzung notwendigen Daten aufgelistet sind, nicht Teil des Typ-I und Typ-II Protokolls. Ein Verzeichnis muss manuell gepflegt werden. Das Typ-III Protokoll umfasst auch die automatisierte Pflege eines Verzeichnisses.

Übersicht der Features

Feature	**Cypherpunk**	**Mixmaster**	**Mixminion**
Anonymisiertes Versenden von Nachrichten	√	√	√
Antworten auf anonymisierte Nachrichten	×	×	√
Verzeichnis	×	×	√

Tabelle 6.1: Fähigkeiten der Remailertypen I-III

Im Kontext der Ausarbeitung ist herausgearbeitet, dass die Entwicklung der verschiedenen Remailertypen hauptsächlich fehlergetrieben war. So wurde ein folgendes Protokoll entwickelt, indem die Schwächen des vorherigen Protokolls analysiert wurden und versucht wurde, aufbauend auf das vorhergehende Protkoll diese Fehler zu beheben. So bauen diese drei Remailertypen alle auf einer gemeinsamen Designbasis auf. Im Laufe der Entwicklung der Typen wurde diese Basis durch inkrementelle Erweiterungen stetig verbessert.

Aktive Attacken

Angriffsart	Cypherpunk	Mixmaster	Mixminion
Manipulation eines Remailers	√	√	√ \ ?
Wiedereinspielung von Nachrichten (Replay)	×	√	√ \ ?
Manipulation einer Nachricht (Tagging)	×	√	√ \ ?

Passive Attacken

Angriffsart	Cypherpunk	Mixmaster	Mixminion
Traffic Analyse	×	√	√ \ ?

Tabelle 6.2: Mögliche Angriffe auf Typ-I bis Typ-III Remailer mit Resistenzangabe

Man sieht, dass das Cypherpunk-Remailer Protokoll Schwächen gegen diverse Angriffe aufweist. Das Mixmaster- und das Mixminion-Protokoll gelten beide als theoretisch sicher. Das Mixminion-Protokoll ist dem Mixmaster-Protokoll, da es sich um eine Weiterentwicklung handelt, theoretisch sicherheitstechnisch überlegen. Es wendet aktuellere sicherheitsrelevante Forschungsergebnisse an, wie die Verwendung von verschlüsselten und verifizierten Verbindungen, wodurch das Abfangen von Paketen durch Dritte verhindert werden soll. Da die praktische Implementierung des Mixminion-Protokolls jedoch seit 2007 stagniert, und der jetzige Stand einer frühen Betaphase entspricht, ist das Mixminion-Protokoll in der Praxis nicht verwendbar. Der jetzige Stand der Implementierung beinhaltet noch zahlreiche Programmierfehler und sicherheitsgefährende Speicherung von Debug-Logs. Eine sichere Anonymisierung gemäß des Designs kann nicht gewährleistet werden. Daher ist aktuell das Mixmaster-Remailer Protokoll das meistverwendete Verfahren.

Ein weitere Betrachtungspunkt ist die Zuverlässigkeit eines Remailers. Wenn Alice eine Nachricht an Bob schickt, muss für Alice sichergestellt sein, dass der Empfänger Bob die Nachricht auch tatsächlich erhält. Bei allen drei Remailertypen handelt es sich um Protokolle, denen ein Verbund von Remailern zur anonymisierten Versendung einer Nachricht zugrunde liegt. In jedem Fall ist es essentiell und gegeben, dass einem Knoten zu keiner Zeit der gesamte Pfad der Nachricht bekannt ist. Ein Knoten kennt nur den unmittelbaren Vorgänger und Nachfolger des Pfads. Aus diesem Grund ist es einem Knoten im Netzwerk nicht dazu in der Lage dem ursprünglichen Absender Alice ein Fehlschlagen der Zustellung mitzuteilen. Aus diesem Grund ist es wichtig, dass die Übertragung der Nachricht durch das Netzwerk fehlerfrei zugesichert werden kann.

Zuverlässigkeitsanalyse

Szenario	Cypherpunk	Mixmaster	Mixminion
Ausfall eines Remailers im Pfad	×	×	√ \ ?
Manipulation eines Remailers im Pfad	×	×	√ \ ?

Tabelle 6.3: Zuverlässigkeit des Nachrichtenzustellvorgangs der Typ-I bis Typ-III Remailer

Die Tabelle zeigt, dass sowohl im Cypherpunk, als auch im Mixmaster- Protokoll eine Zustellung einer Nachricht zum Zeitpunkt der Erstellung nicht zugesichert werden kann. In beiden Fällen ist der Grund dafür das manuell organisierte Verzeichnis der Remailer. Beim Mixminion-Protokoll wird durch die automatische und dauerhafte Synchronisierung der Remailer mit einem Verzeichnisserver der Ausfall eines Remailerknotens durch das Protokoll selbst festgestellt und verhindert, dass ein Klient diesen Remailer in seinen Pfad aufnimmt. Da das Mixminion Protokoll nie vollständig implementiert wurde, ist diese Zusage rein theoretischer Natur.

Literaturverzeichnis

[Cha81] David L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24:84–88, 1981. http://www.freehaven.net/anonbib/cache/chaum-mix.pdf.

[DDM] G. Danezis, R. Dingledine, and N. Mathewson. Mixminion: Design of a type iii anonymous remailer protocol. http://www.mixminion.net/minion-design.pdf.

[HF13] P. Horster and D. Fox. *Datenschutz und Datensicherheit: Konzepte, Realisierungen, Rechtliche Aspekte, Anwendungen.* DuD-Fachbeiträge. Vieweg+Teubner Verlag, 2013.

[Kub07] J. Kubieziel. *Anonym im Netz: Techniken der digitalen Bewegungsfreiheit.* Open Source Press, 2007.

[Loe09] K. Loesing. *Privacy-enhancing Technologies for Private Services.* Schriften aus der Fakultät Wirtschaftsinformatik und Angewandte Informatik der Otto-Friedrich-Universität Bamberg. University of Bamberg Press, 2009.

[mix08] Mixmaster protocol manpage. http://mixmaster.sourceforge.net/manpage.html, 2008.

[nym] Nym-server definition. https://www.techopedia.com/definition/1696/nym-server.

[Ora01] A. Oram. *Peer-to-Peer: Harnessing the Power of Disruptive Technologies.* O'Reilly Media, 2001.

[SS13] J. Sambleben and S. Schumacher. *Informationstechnologie und Sicherheitspolitik: Wird der dritte Weltkrieg im Internet ausgetragen?* Books on Demand, 2013.