

# S P A M

*- die technische Seite -*

*Jan-Ole Beyer <cosmic@cs.tu-berlin.de>  
Matthias Knoll <hanknolo@cs.tu-berlin.de>  
Kia Teymourian <kiat@cs.tu-berlin.de>*

*R. Gehring, K. Ishii, B. Lutterbeck:*

## **Information Rules**

Wintersemester 2001/2002

Fakultät IV: Elektrotechnik und Informatik  
Technische Universität Berlin



## ***Inhaltsverzeichnis***

<b>1.</b>	<b>Einführung</b>	<b>3</b>
<b>2.</b>	<b>Grundlagen der Kommunikation per E-Mail</b>	<b>3</b>
2.1	Simple Mail Transfer Protocol (SMTP)	3
2.2	Multipurpose Internet Mail Extension (MIME)	4
2.3	Post Office Protocol (POP)	4
2.4	Internet Message Access Protocol (IMAP)	4
<b>3.</b>	<b>Vorgehensweisen der Spammer</b>	<b>5</b>
3.1	Das Fälschen eines E-Mail-Headers per SMTP	5
3.2	Das Erkennen eines gefälschten Headers	6
3.3	Massmailer und Push-Tools	7
<b>4.</b>	<b>Adress-Sammler</b>	<b>7</b>
<b>5.</b>	<b>Maßnahmen gegen SPAM</b>	<b>9</b>
5.1	...von Seiten des Providers	10
5.1.1	Versand von SPAM	10
5.1.2	Empfang von SPAM	11
5.2	...von Seiten des Benutzers	11
5.2.1	Prävention	11
5.2.2	Filter	12
5.2.3	sonstiges	12
<b>6.</b>	<b>Schluss</b>	<b>13</b>
<b>7.</b>	<b>Quellenangaben</b>	<b>14</b>

# 1. Einführung

SPAM gehört mittlerweile wohl für fast jeden aktiven Nutzer zu den großen Übeln des Internets. In dieser Arbeit wollen wir einerseits auf die Grundlagen der E-Mail-Kommunikation, also auf die zugrundeliegenden Protokolle kurz eingehen, um ein allgemeines Verständnis von E-Mail zu schaffen. Dann werden wir auf die Techniken, die von Spammern angewendet werden eingehen, um dann vertieft auf Maßnahmen einzugehen, die sowohl serverseitig als auch vom Benutzer selbst angewendet werden können, um SPAM möglichst effektiv zu verhindern.

## 2. Grundlagen der Kommunikation per E-Mail

Elektronische Post funktioniert ähnlich der normalen ("gelben") Post. Ein Brief wird geschrieben und mit Absender und Empfänger versehen, um ihn dann an die elektronische Poststelle, den Mailserver, weiterzuleiten. Dieser ermittelt dann die Poststelle des Empfängers sowie den Weg dorthin und verschickt ihn auf diesem. Schließlich erreicht der Brief dann die Mailbox des Empfängers, zumeist mit einem Hinweis, dass neue Post vorhanden ist.

### 2.1 Simple Mail Transfer Protocol (SMTP)

SMTP ist die Abkürzung für "Simple Mail Transfer Protocol", ein weit verbreitetes Protokoll zum Versenden von E-Mails. Der SMTP-Dienst dient dazu, E-Mails an eine beliebige gültige E-Mail-Adresse zu senden. Der SMTP-Server (welcher den SMTP-Dienst bereitstellt), führt zwei wichtige Aufgaben aus: Erstens verifiziert (authentifiziert) er die Identität desjenigen, der auf den SMTP-Account zugreifen will, mittels Benutzername und Passwort. Als zweites macht er es möglich, E-Mails zu versenden, nachdem die Verifizierung vollzogen ist. Kann die E-Mail nicht zugestellt werden, sendet er sie mit einer Fehlermeldung wieder an den Absender zurück (der sogenannte *bounce*). Ist beispielsweise die Zieladresse falsch, wird die E-Mail vom SMTP-Server zurückgesendet mit dem Vermerk "Adresse unbekannt" ("address unknown") oder "Benutzer unbekannt" ("user unknown"). Es können auch andere Probleme auftreten, den Grund findet sich jedoch immer auf der zurückgekommenen E-Mail.

Ein Beispiel für SMTP-Nachricht (S bezeichnet den Mailserver, C den Client):

```
S: 220 Beta.GOV Simple Mail Transfer Service Ready
C: HELO Alpha.EDU
S: 250 Beta.GOV
C: MAIL FROM:<>
S: 250 OK
C: RCPT TO:<Jones@Beta.GOV>
S: 250 OK
C: RCPT TO:<Green@Beta.GOV>
S: 550 No Such User here
C: RCPT TO:<BrownGreen@Beta.GOV>
S: 250 OK
C: DATA
S: 354 Start mail input; end with <CR><LF>. <CR><LF>
C: ...sends body of mail message...
C: ...continues for as many lines as message contains
C:<CR><LF>.<CR><LF>
S:250 OK
C:QUIT
```

S: 221 Beta.GOV Service closing transmission channel

## 2.2 Multipurpose Internet Mail Extensions (MIME)

MIME ermöglicht es, neben ASCII-Text zum Beispiel Bild- oder Audio-Daten sowie Texte in anderen als ASCII-Zeichensätzen (auch gemischt) in definierten Formaten über das Internet auszutauschen. MIME ist nicht auf ein bestimmtes Transportprotokoll festgelegt und erfordert keinerlei Änderungen der Mailserver.

Die neuen Inhaltstypen müssen geeignet kodiert werden, damit die von den Transportsystemen gesetzten Bedingungen erfüllt werden. Im allgemeinen wird man die Kodierung so wählen, daß sich der Mail-Inhalt für das Mail-Transportprotokoll SMTP wie bisher als ASCII-Text darstellt. Für andere Protokolle, wie z.B. für *uucp* oder das erweiterte SMTP, kann auf eine solche 7-Bit-Kodierung verzichtet werden. Die verwendete Kodierung wird im Header in der Zeile *Content-Transfer-Encoding* festgehalten.

## 2.3 Post Office Protocol (POP)

POP3 ist die Abkürzung für "Post Office Protocol Version 3", ein weit verbreitetes Protokoll zur Benutzer-Authentifizierung und zum Herunterladen von E-Mails. Ein POP3-Account ist ein "Postfach" für eine E-Mail-Adresse. Nur mit zulässigem Benutzernamen und korrektem Passwort ist der Zugriff darauf möglich.

Damit der E-Mail-Client weiß, auf welchen POP3-Account er zugreifen soll, muß man ihm die Serveradresse (das "Postamt") und den POP3-Account (das "Postfach") auf diesem Server bekanntgeben. Erst diese Kombination von Server und POP3-Account macht die E-Mail-Adresse eindeutig (Im Internet gibt es keine gleichnamigen Serveradressen, und innerhalb eines Servers keine gleichnamigen Benutzerkennungen).

Der POP3-Dienst dient also dazu, E-Mails abzuholen, welche an einer E-Mail-Adresse mit POP3-Konto eingetroffen sind. Der POP3-Server (welcher den POP3-Dienst bereitstellt), führt zwei wichtige Aufgaben aus: erstens verifiziert (authentifiziert) er die Identität desjenigen, der auf den POP3-Account zugreifen will, mittels Benutzernamen und Passwort. Als zweites macht er es möglich, die eingetroffenen E-Mails herunterzuladen (abzuholen), nachdem die Verifizierung vollzogen ist.

## 2.4 Internet Message Access Protocol (IMAP)

IMAP ist neben POP3 ein weiteres (neueres) Protokoll, um E-Mail abzurufen. Es bietet einige neue Möglichkeiten. Während bei POP3 die E-Mail erst heruntergeladen werden muss, um sie zu bearbeiten, zu lesen oder zu löschen, können sie per IMAP direkt auf dem Mailserver verwaltet und gelesen werden, ohne sie vorher herunterladen zu müssen. Dies spart, insbesondere bei viel SPAM im Postfach, kostbare Downloadzeit.

Vorteile von IMAP sind neben der Einsparung von Downloadzeit natürlich auch für diejenigen offensichtlich, die ihre E-Mails an verschiedenen Orten bearbeiten, beispielsweise im Büro und zu Hause. Die E-Mail bleibt dann auf dem Server und kann auch von einem anderen Ort aus noch abgerufen und bearbeitet werden.

### 3. Vorgehensweisen der Spammer

#### 3.1 Das Fälschen eines E-Mail-Headers per SMTP

Sicherlich eine der größten Sicherheitsschwachstellen bei der Übertragung von E -Mails ist das bereits erwähnte *Simple Mail Transfer Protocol* (SMTP). So wie jeder Mailclient eine Verbindung zum dafür reservierten Port 25 eines Mailservers aufbaut, kann dies auch ein jeder "böswilliger" User tun, und mit entsprechenden Kenntnissen des SMTP beliebig g e-fälschte E -Mails versenden. Die Vorgehensweise dabei wird im Folgenden anhand eines kleinen Beispiels dargestellt.

Zunächst sucht man sich einen Server, der das sogenannte "open relaying" unterstützt, das bedeutet, ein Mailserver, der Mail von jedem User an jeden beliebigen User annimmt und zustellt. Zu diesem Server erstellt man dann eine Verbindung zum Port 25 (z.B. mittels "telnet") und kommuniziert mit ihm direkt im SMTP. Dies könnte in etwa wie in folgendem Beispiel aussehen:

(Der Servername und die Empfänger-E-Mail-Adresse wurden nachträglich geändert)

```
220 lotech.LOTECHDOM.LOCAL ESMTP Server (Microsoft Exchange
Internet Mail Service 5.5.2650.21) ready
helo mail.microsoft.com
250 OK
mail from: Bill@microsoft.com
250 OK - mail from <Bill@microsoft.com>
rcpt to: Matoffel@t-offline.de
250 OK - Recipient <Matoffel@t-offline.de>
data
354 Send data. End with CRLF.CRLF
To: Roberto
From: Rex
Subject: Fiesta Mexicana
Cc: Roy

Hossa!

.
250 OK
quit
221 closing connection
```

Nachdem man vom Server Bereitschaft signalisiert bekommen hat stellt man den eigenen Rechner mit dem Befehl "helo" vor. Hierbei ist zu erwähnen, dass man jedoch ohne Probleme jeden beliebigen Rechnernamen angeben kann, der zu einer gültigen Domain gehört (wie im Beispiel *mail.microsoft.com*). Im Anschluss daran kommt der sogenannte SMTP -Envelope, der die eigentlichen Absender- und Empfängerdaten enthält und mit den Befehlen "mail from:" und "rcpt to:" erstellt wird. Wiederum gilt, dass man bei den Absenderdaten nach Lust und Laune lügen kann, der Empfänger jedoch muss eine korrekte E -Mail-Adresse sein, da die Mail sonst logischerweise nirgendwo ankommen würde. Nach der Anweisung "data" folgen nun die eigentlichen Daten der E -Mail. Hierbei gilt, dass alle Informationen, deren erstes Wort durch einen Doppelpunkt abgeschlossen wird Header -Informationen sind. Nach einer Leerzeile folgt dann die zu übermittelnde Botschaft, deren Ende durch einen einzelnen Punkt auf einer eigenen Zeile signalisiert wird. Nun noch ein abschließendes "quit" und die

E-Mail ist versendet.

### 3.2 Das Erkennen eines gefälschten Headers

Nachdem wir nun gezeigt haben, wie einfach man mittels SMTP eine E-Mail fälschen kann, schauen wir uns nun den dazugehörigen Header an und erläutern daran, wie man den E-Mail-Header verstehen und gefälschte E-Mails erkennen kann.

```
Return-Path: <Bill@microsoft.com>
Received: from lotech.LOTECHDOM.LOCAL ([213.42.101.66]) by
mailin06.sul.t-online.de
        with esmtp id 16XOWf -0fc9tgC; Sun, 3 Feb 2002
16:25:53 +0100
Received: from mail.microsoft.com (pD9EB101F.dip.t -dialin.net
[217.235.16.31]) by lotech.LOTECHDOM.LOCAL with SMTP
(Microsoft Exchange Internet Mail Service Version 5.5.2650.21)
        id D9CLKWPP; Sun, 3 Feb 2002 19:22:03 +0400
To: Roberto
From: Rex
Subject: Fiesta Mexicana
Cc: Roy
```

Gehen wir also den Header von oben nach unten durch. Der Return-Path enthält die Absenderangabe aus dem SMTP-Envelope, kann also wie bereits erwähnt beliebig gefälscht sein. Sollte dies eine existierende E-Mail-Adresse sein, ist dies die Adresse, an die eine Antwort versendet wird, zumindest falls keine weitere Angabe mittels "Reply-To" gemacht wurde.

Die nun folgenden Zeilen sind die interessantesten und dienen am ehesten der Erkennung von gefälschten E-Mails. Die Received-Zeilen werden automatisch generiert, wenn ein Mail von einem *Message Transfer Agent* (MTA) empfangen wird, um zu dokumentieren, welche Orte die E-Mail durchlaufen hat. Es ist jedoch auch möglich, beim Versenden zusätzlich solche Zeilen einzufügen. Die oberste Zeile ist hierbei die vom eigenen Mail-Server hinzugefügte und somit vertrauenswürdig. Will man nun also herausfinden, ob die empfangene Mail gefälscht ist, geht man die Received-Zeilen von oben nach unten durch bis man zu nicht schlüssigen Angaben kommt. Die From-Passage macht Angaben über den Server, der die Nachricht gesendet hat. Der erste Name entspricht der Anmeldung über das *helo*-Kommando und muss wie bereits erwähnt nicht stimmen. Aus diesem Grund merkt sich der Empfänger auch die IP-Adresse des Senders und vermerkt sie in eckigen Klammern und gibt häufig auch gleich den zu dieser IP-Adresse gehörenden Namen an, sollte der erste nicht korrekt sein. Nach dem *by* findet man den empfangenden Mailserver, Angaben über das dort laufende Mailprogramm und Datum und Uhrzeit. Im Falle unseres Beispiels kann man erkennen das hinter dem zweiten Received die verifizierte IP-Adresse des Senders, die vom Mailserver festgestellt wird, zu einer anderen domain gehört, als die durch den Befehl "helo" angegebene. Also erkennen wir daran unseren kleinen Betrug.

Ein weiterer Anhaltspunkt, um gefälschte E-Mails zu erkennen, können auch die Zeitangaben sein. Sollten diese stark auseinander liegen, ist auch hier die Wahrscheinlichkeit, dass es sich um gefälschte Angaben handelt, sehr groß.

Die nächsten Zeilen sind die von uns nach dem "data" Kommando angegebenen und werden im E-Mail-Client des Empfängers in den entsprechenden Feldern dargestellt, so dass man sich, falls man den Header nicht untersucht, durchaus wundern kann, wie man denn überhaupt zu dieser Nachricht gekommen ist.

Da unser Beispiel-Header nicht sonderlich groß ist, hat er natürlich auch nicht alle Informationen, die man einem E-Mail-Header wiederfinden kann, abgedeckt. So findet man im No-

malfall auch noch eine Message-ID, die je nach Mail-Client das Sende-Datum, ein @ und den Absender in kodierter Form und den Namen des sendenden Servers enthält. Des Weiteren sind häufig Einträge zum MIME-Format, *Content-Type* und *Content-Transfer-Encoding* enthalten, welche Auskunft zur Interpretation des Body und zum richtigen Umgang mit Mail-Attachments geben. Es können außerdem sogenannte X-Einträge enthalten sein, die nicht standardisiert sind und die der sendende Mail-Client generiert. Beispielsweise findet man durch den Eintrag "X-Mailer" heraus, welchen Mail-Client der Sender verwendet, während "X-Sender" Angaben zum tatsächlichen Absender enthält. Über T-Online versendete E-Mails enthalten zum Beispiel beim "X-Sender" Eintrag die Kundennummer des Senders.

### 3.3 Massmailer und Push-Tools

Die vorgestellte Variante des E-Mail-Fälschens ist für Massenversendung natürlich nicht sonderlich praktikabel. Deshalb gibt es für solche Zwecke speziell entwickelte Programme, die sogenannten Push-Tools oder Massmailer. Diese werden auf diversen Webseiten legal kommerziell vertrieben, so sind wir bei unseren Nachforschungen zum Beispiel auf einen Anbieter gestoßen, der sogar den Quellcode seines Programms für 300\$ verkauft.

Diese Push-Tools haben ihren eigenen Mail-Server integriert und schaffen es bei entsprechender Bandbreite, bis zu 250.000 Mails pro Stunde zu versenden. Des Weiteren fälschen sie natürlich auch den E-Mail-Header und unterstützen jedes gängige Datenbankformat, so dass man nur eine entsprechende Liste mit E-Mail-Adressen benötigt. Wie solch eine Datenbank von E-Mail-Adressen zustande kommt, erläutern wir im folgenden Abschnitt.

## 4. Adress-Sammler

Es gibt eine Vielzahl von Möglichkeiten, wie Spammer (bzw. diejenigen, die ihnen Adress-Sammlungen verkaufen) an Adressen gelangen. Einige dieser Möglichkeiten wollen wir hier kurz vorstellen, um einen Überblick darüber zu geben, auf welche zum Teil kreative Art und Weise die eigene E-Mail-Adresse in die Hände von Spammern gelangt.

Da solche Adress-Sammlungen meist in einer Menge von hunderttausenden Adressen verkauft werden, werden die Adressen natürlich nicht mehr manuell, sondern durch automatisierte Programme. Im Allgemeinen durchsuchen diese Tools zum Beispiel Webseiten, folgen den Links, die sie darauf finden, und speichern "alles, was nach E-Mail-Adresse aussieht". Dies kann einerseits ein im HTML-Quelltext gefundenes "mailto:" sein, aber genauso kann auch einfach jedes Wort, das ein "@" beinhaltet gespeichert werden.

Auch die "robots.txt"-Datei, die normalerweise diesen sogenannten WebRobotern sagt, welche Seiten sie durchsuchen dürfen und welche nicht, nützt nichts, da die WebRoboter sich nur freiwillig nach ihr richten, was Adress-Sammler natürlich nicht tun.

Folgende Punkte zeigen einige der Möglichkeiten, im Internet an E-Mail-Adressen zu gelangen. Sie basieren hauptsächlich auf "How do spammers harvest email addresses?" von Uri Raz (<http://www.private.org.il/harvest.html>).

#### - UseNet Postings

Insbesondere durch Archive wie [groups.google.com](http://groups.google.com), in denen Newsgroups systematisch durchsucht werden können, ist das UseNet eine der wichtigsten Quellen für Adress-Sammler. Besonders interessant ist es, dass sich eine Sortierung nach Interessensgebieten vornehmen lässt, die die Adressen wertvoller macht.

- Mailing Listen
 

Es gibt mehrere Möglichkeiten, an die Adressen von Mailing Listen-Benutzern zu gelangen. Hierauf soll aber nicht näher eingegangen werden. Bei weiterem Interesse sei zum Beispiel auf oben genannten Text von Uri Raz verwiesen.
- Web Seiten
 

Hierauf wurde bereits weiter oben in Zusammenhang mit WebRobotern eingegangen.
- Gästebücher, Formulare
 

Die zumeist öffentlichen Gästebücher lassen sich genauso wie Web Seiten durchsuchen. Zu erwähnen ist allerdings, dass insbesondere manche Formulare, wie sie zum Beispiel häufig beim Download von Programmen, Treibern o.ä. ausgefüllt werden sollen, häufig lediglich der Adress-Sammlung dienen.
- ident daemon
 

Der ident daemon, der auf vielen UNIX-Rechnern vorhanden ist, dient normalerweise dazu, anderen Rechnern zu ermöglichen, einen Benutzer, der sich mit ihnen in Verbindung setzt, zu identifizieren. Da diese Identifikation den Usernamen des Benutzers enthält, können so auch häufig Rückschlüsse auf die E-Mail-Adresse gezogen werden.
- finger daemon
 

Manche finger daemons erlauben es einem Benutzer, über eine Eingabe wie john@host sämtliche Benutzer auszugeben, die den Namen John haben. Dies kann natürlich auch leicht dafür verwendet werden, Adresslisten zu erstellen.
- Web Browser
 

Bei Web Browsern gibt es mehrere interessante Möglichkeiten, an die E-Mail-Adresse des Benutzers zu gelangen.

Viele Browser übersenden bei einer anonymen FTP-Verbindung die E-Mail-Adresse als Passwort. Dies machen sich Adress-Sammler zunutze, indem sie eine Grafik auf ihrer Webseite über solch eine anonyme FTP-Verbindung (unbemerkt) laden lassen und das Passwort (sprich: die E-Mail-Adresse) speichern.

Eine weitere Möglichkeit ist die, über JavaScript den Browser zu veranlassen, eine E-Mail an die Adresse des Sammlers zu verschicken, wenn sich die Maus über einen bestimmten Teil der Seite bewegt.

Des Weiteren übersenden viele Browser mit dem HTTP\_FROM Header, der an alle Server versendet wird, auch die E-Mail-Adresse des Benutzers, die so leicht gespeichert werden kann.
- IRC, Chaträume
 

Insbesondere AOL-Chaträume sind ein beliebtes Ziel von Adress-Sammlern, da viele AOL-Kunden neu im Internet sind und somit "frische" Adressen verwenden. Auch hier lassen sich, wie bei Web Seiten und im UseNet, die Adressen über automatisierte Programme sammeln.
- AOL Profile
 

Aus den AOL-Profil-Listen lassen sich leicht automatisiert Adressen sammeln. Da hierin auch häufig Interessen der Benutzer stehen, lassen sich die Adressen auch gut nach eben diesen sortieren.
- NIC Einträge von Domain-Inhabern
 

Mittels beispielsweise des "whois"-Kommandos lassen sich die E-Mail-Adressen der Ansprechpartner einer Domain ermitteln, die zumeist aktuell sind und gelesen werden.

- "guessing & cleaning"

Dieser Punkt bedeutet, dass die Adress-Sammler "auf gut Glück" E-Mails verschicken, zum Beispiel im häufig verwendeten Format "vornname.nachname@isp.com" oder an Standard-Adressen (root@..., postmaster@..., hostmaster@...). Wenn keine Fehlermeldung zurückkommt, existiert die Adresse. Wahlweise kann man bei SMTP auch eine automatische Bestätigung anfordern, ob eine E-Mail korrekt übertragen oder gelesen wurde, so dass auf diese Weise existierende bzw. benutzte Adressen erkannt werden.

- White & yellow pages

Solche Adressverzeichnisse sind ein lohnendes Ziel für Adress-Sammler. Allerdings unterbinden viele ein derartiges Sammeln. Zu beachten ist, dass man sich nicht zwangsläufig selbst eingetragen haben muss. Hotmail beispielsweise trägt neue Benutzer automatisch im BigFoot-Verzeichnis ein.

- Zugang zum selben Computer

Wenn der Adress-Sammler Zugang zu einem von vielen Menschen benutzten Rechner hat, kann er auf vielfältige Art und Weise die Usernamen (und damit auch E-Mail-Adressen) der anderen Benutzer herausfinden, zum Beispiel unter UNIX durch die meist für alle lesbare Datei "/etc/passwd".

- vorheriger Besitzer der Adresse

Wenn man sich zum Beispiel bei AOL einen Screennamen aussucht, der schon einmal von jemandem benutzt wurde, kann es sein, dass dieser Name Spammer quasi schon vorher bekannt war. Das gilt natürlich auch für andere Provider.

- "social engineering"

Diese Technik beinhaltet unter anderem das Versenden von Kettenbriefen, um an Adressen zu gelangen. Wenn beispielsweise eine E-Mail mit dem Inhalt verschickt wird, dass jeder, der sie an alle Freunde weiterleitet und gleichzeitig eine Kopie an den ursprünglichen Absender schickt, kostenlose CDs bekommt, dann wird diese E-Mail sicherlich von recht vielen ernst genommen. Auf diese Weise lassen sich leicht komplette Adressbücher mit aktiven Adressen gewinnen.

Diese Punkte sollen das Bewusstsein dafür schärfen, auf welche vielfältige Weise Adress-Sammler vorgehen. Einige Punkte erlauben es, die Weitergabe der E-Mail-Adresse zu verhindern, andere aber auch nicht, und Adress-Sammler werden sich wohl immer neue Techniken überlegen, um ihr Ziel zu erreichen. Eine vollständige Geheimhaltung und gleichzeitige Verwendung einer Adresse ist wohl nicht möglich, aber zumindest ein größtmöglicher Schutz vor solchen Sammlern.

## **5. Maßnahmen gegen SPAM**

Im folgenden wollen wir auf Techniken eingehen, die eine Überfüllung des eigenen Postfachs zu verhindern helfen. Natürlich ist eine vollständige Unterdrückung von SPAM kaum möglich, und wenn doch, dann wohl kaum frei von Nachteilen.

Da sich aber eine ungewollte "Veröffentlichung" der Adresse kaum vermeiden lässt, wie wir in Teil 4 gesehen haben, muss das Haupt-Augenmerk, um SPAM zu vermeiden, wohl auf diese Seite gelegt werden.

## 5.1 ...von Seiten des Providers

Auf der Provider -Seite muss unterschieden werden zwischen einerseits der Verhinderung des Versands von SPAM und andererseits der Auslieferung von SPAM, wobei in beiden Fällen unbedingt verhindert werden muss, dass auch "normale" E-Mails dabei unterdrückt werden.

### 5.1.1 Versand von SPAM

Ein wichtiger Punkt, um gegen Spammer rechtlich vorgehen zu können, ist eine entsprechende *Acceptable Use Policy*, die einen nicht akzeptablen Gebrauch der Dienste eines Providers ausschließt. Dazu gehören zum Beispiel die Auflagen, den Service nicht gesetzeswidrig zu verwenden, nicht zu versuchen, illegal in andere Netze einzudringen und keine Mailbomben zu versenden, aber auch das Verbot, kommerzielle Nachrichten ohne Erlaubnis ins UseNet zu stellen oder Junkmail oder SPAM an Leute zu verschicken, die das nicht möchten.

Daneben gibt es natürlich auch einen technischeren Aspekt, den eigenen Mailserver gegen Spammer abzusichern, insbesondere das sogenannte *third party relaying*.

Das bedeutet, dass ein Aussenstehender (d.h. jemand, der nicht offizieller Nutzer des Mailservers ist) über eben diesen Mailserver E-Mails an andere Aussenstehende versendet. Diese sollte niemals möglich sein und gilt allgemein als Fehlkonfiguration des Mailservers.

Um das *third party relaying* zu verhindern, muss man, ganz allgemein, die Benutzung des Mailservers einschränken. Dazu gehören folgende Punkte:

- Die Empfänger-Adresse einer E-Mail gehört zur eigenen Domain
- Mails an die Empfänger-Domain werden grundsätzlich angenommen und weitergeleitet
- Die IP-Adresse des Clients ist bekannt und autorisiert (lokaler Benutzer, Standleitung, authentifizierte Wählleitung, SMTP-after-POP, SMTP-Authentifizierung, SMTP-per-TLS o.ä.)

Eine Anleitung, um *third party relaying* zu verhindern, findet sich unter <http://mail-abuse.net/tsi/ar-fix.htm>.

Eine weitere Möglichkeit, einen Spammer effektiv an seiner Arbeit zu hindern, ist das sogenannte *teergrubing*. Dies basiert darauf, dass der E-Mail-Versand über TCP/IP-Verbindungen erfolgt und dass höchstens 65.000 TCP/IP-Verbindungen gleichzeitig aufgebaut werden können, durch interne Beschränkungen meistens weniger.

*Teergrubing* ist nun der Versuch, den TCP/IP-Port, den der Spammer benutzt, auch nach der Mailauslieferung möglichst mehrere Stunden offenzuhalten, so dass die Leistungsfähigkeit des Spammers sinkt. Viele, dezentral organisierte Teergruben können einen Spammer somit komplett stoppen.

Das Offenhalten des Ports funktioniert durch die sogenannten Fortsetzungszeilen von SMTP (realisiert durch "-"), die bedeuten, dass der Host noch nicht fertig gesendet hat. Wenn er nun alle paar Minuten eine solche Fortsetzungszeile schickt, verbraucht das kaum Bandbreite, aber stoppt den Spammer.

Näheres zu Teergruben findet sich unter <http://www.iks-jena.de/mitarb/lutz/usenet/teergrube.html>

### 5.1.2 Empfang von SPAM

Auch der Empfang bzw. des Weiterleiten von SPAM durch den Mailserver kann natürlich verhindert werden. Da hierfür aber hauptsächlich verschiedene Möglichkeiten des Filterns in Frage kommen, sollte im allgemeinen darauf verzichtet werden, da Filtern auch immer die Möglichkeit beinhaltet, eigentlich gewünschte "gute" E-Mail eines Benutzers mit auszufiltern. Auf die verschiedenen Möglichkeiten wird in Punkt 5.2 näher eingegangen.

## 5.2 ...von Seiten des Benutzers

Auf Benutzerseite sollte man unterscheiden zwischen Prävention, Filterungstechniken und anderen Möglichkeiten, gegen Spammer vorzugehen. Durch eine gute Prävention läßt sich ein Großteil des SPAM schon dadurch verhindern, dass die eigene E-Mail-Adresse den Spammern gar nicht bekannt ist. Nichtsdestotrotz läßt sich, wie in Punkt 4 schon erwähnt, die E-Mail-Adresse niemals völlig "geheim" behalten, und somit sind ausserdem natürlich auch effektive Filter sinnvoll, um das Postfach vor Überfüllung zu schützen. Als drittes bleiben dann noch weitere Möglichkeiten, die sich sonst nirgendwo einordnen ließen.

### 5.2.1 Prävention

Auf präventive Maßnahmen sind wir teilweise in Abschnitt 4 eingegangen. An dieser Stelle möchten wir aber noch einmal konkret einzelne Punkte auflisten, die zu verhindern helfen, dass Spammer an eine E-Mail-Adresse herankommen.

Bei SPAM -Mails gibt es meistens eine E-Mail-Adresse am Ende, an die man eine E-Mail schicken sollte, um von der Adressliste entfernt zu werden. Das ist allerdings in der Realität nur selten der Fall. In den meisten Fällen dagegen dient eine Antwort des "Opfers" lediglich dazu, eine Adresse als benutzt (und damit wertvoller) zu verifizieren. SPAM sollte also auf *keinen* Fall beantwortet werden.

Auch Newsletter o.ä. dienen häufig dazu, Adressen zu sammeln, und das nicht nur für den speziellen Newsletter, sondern auch, um sie weiter zu verkaufen. Auf die meisten Newsletter sollte also, auch auf die von renommierten Unternehmen, verzichtet werden.

Wie schon in Abschnitt 4 erwähnt, gibt es WebRoboter, die das WWW nach E-Mail-Adressen durchsuchen. Es sollte also möglichst darauf verzichtet werden, die eigene E-Mail-Adresse im WWW zu veröffentlichen. Eine Möglichkeit, Besuchern trotzdem einen Kontakt per E-Mail zu ermöglichen, bieten Formulare, bei denen die eigene E-Mail-Adresse nicht offen auf der Seite steht.

Des weiteren erwähnten wir in Abschnitt 4 bereits Kettenbriefe und Hoaxes, also falsche Viruswarnungen. Auch diese sind, abgesehen davon, dass sie unsinnig sind, häufig dazu da, Adressen zu sammeln, insbesondere da kaum jemand solche E-Mails per *blind copy* weiterleitet. Sie sollten gelöscht und auf keinen Fall weitergeleitet werden.

Im allgemeinen sollten E-Mails an mehrere Personen, die sich gegenseitig nicht kennen, per BCC, also *blind copy*, verschickt werden, so dass die Adressen von einem eventuell auf der Liste vorhandenen Adress-Sammler nicht mißbraucht werden können.

Im UseNet hat es sich zum Teil eingebürgert (wo bei es zumindest im deutschsprachigen UseNet eher verpönt ist), seine Adresse zum Beispiel in der Form *name.nospam@isp.de* oder *name@isp.nospam.de* zu ändern. Näheres hierzu findet sich im Mini-FAQ "Falsche E-Mail-Adressen", das monatlich in folgenden News groups gepostet wird: *de.admin.net-*

*abuse.mail*, *de.newusers.questions*, *de.answers*, *news.answers* sowie im WWW zum Beispiel unter [www.gerlo.de/falsche-email-adressen.html](http://www.gerlo.de/falsche-email-adressen.html) zu finden ist.

Häufig wird auch beim Download von Treibern o.ä. die E-Mail-Adresse verlangt. Wenn man hier keine falsche Adresse angeben will, bleibt noch die Möglichkeit, die entsprechenden Dateien auf dem FTP-Server der jeweiligen Seite zu suchen. Meistens hat man auch hier Glück, und das, ohne die E-Mail-Adresse anzugeben.

Wenn man sich an diese Verhaltensregeln hält, sollte es denn Adress-Sammlern wesentlich erschwert werden, an die eigene E-Mail-Adresse zu kommen.

### 5.2.2 Filter

Neben dem Filtern durch den Mailserver direkt gibt es natürlich auch die (sinnvolle re) Möglichkeit, E-Mails als Benutzer selber filtern. Hierbei gibt es verschiedene Möglichkeiten, bei denen das Filtern ansetzen kann.

Eine Möglichkeit ist die, das Subject zu filtern. Spammer benutzen häufig mehrere Ausrufezeichen, Dollarzeichen, Wörter wie *sex*, *xxx* etc. Diese Art zu filtern kann aber auch leicht "normale" E-Mail ausfiltern, sofern auch von Freunden zum Beispiel mehrere Ausrufezeichen o.ä. verwendet werden.

Eine weitere Möglichkeit ist die, komplette Domains auszufiltern. Manche Domains sind dafür bekannt, spammer-freundlich oder zumindest -neutral zu sein. Auch diese Art des Filterns kann problematisch sein, falls "zulässige" E-Mail-Partner auch über solch einen Server E-Mail verschicken.

Des Weiteren lässt sich natürlich auch der Absender von SPAM "für die Zukunft" direkt filtern. Das ist aber selten von Erfolg gekrönt, da die meisten Spammer entweder, wie bereits erwähnt, falsche E-Mail-Adressen verwenden oder, falls sie richtig sind, diese nur einmalig benutzen.

Falls der Spammer den E-Mail-Header gefälscht hat, gibt es viele Möglichkeiten, dieses zu erkennen (zumindest falls er Fehler gemacht hat). Darauf wurde bereits kurz in Abschnitt 3 eingegangen. Im Gegensatz zu den anderen Filterungstechniken gibt es leider für diese Technik keine automatisierten Programme, das heißt, dass ein Filtern nach Header-Unstimmigkeiten noch manuell (was mit recht viel Aufwand verbunden ist) geschehen muss.

### 5.2.3 sonstiges

Es gibt natürlich noch andere Möglichkeiten neben Prävention und Filtern, aber diese ersten gehören wohl, zusammengenommen, zu den einfachsten und effektivsten.

Um die korrekte Adresse des Spammers herauszubekommen, kann man natürlich einerseits den Header manuell untersuchen. Einfacher ist es aber, den kompletten Header an die für Spam zuständige Adresse des eigenen Providers zu senden (meist *spam@...*, *abuse@...*), der sich dann weiter darum kümmert. Daneben gibt es auch noch Webseiten wie *spamcop.net*, die sich um diese sogenannten Spam-Reports kümmern.

Wenn man erst einmal einen Ansprechpartner oder eine Adresse des Spammers hat, gibt es in Deutschland auch die Möglichkeiten der Abmahnung oder der Anzeige. Da die meisten Spammer allerdings aus den USA heraus operieren, hat dieses Vorgehen selten Sinn.

Wenn man erst einmal ausprobieren will, ob die E-Mail-Adresse bei einem Newsletter o.ä. verantwortungsbewußt behandelt wird, bietet sich ein *limited life alias* an, eine E-Mail-

Adresse, die nur für eine bestimmte Zeit gilt, bei *mailexpire.com* zum Beispiel zwischen 12 Stunden und 3 Monaten.

Um Adress-Sammler effektiv an ihrer Arbeit zu hindern, kursiert im Internet ein kurzes Script, das man in den eigenen Webauftritt einfügen kann, zum Beispiel mit Hilfe eines für "normale" Besucher unsichtbaren Links. Es erstellt eine HTML-Seite mit einer großen Zahl unsinniger, zufälliger E-Mail-Adressen sowie einem Link auf sich selbst. Sollte ein automatisierter Adress-Sammler sich hierhin verirren, wird er quasi "eingefangen". Zu finden ist solch ein Script namens WPoison zum Beispiel unter <http://www.monkeys.com/wpoison>.

Im Abschnitt über Filterungstechniken sind wir schon auf das Filtern von ganzen Domains eingegangen. Unter anderem, um aktualisierte Informationen darüber automatisch zu erhalten, um effektiver filtern zu können, hat sich das Mail Abuse Prevention System (MAPS), eine non-profit-Organisation aus Kalifornien gegründet. MAPS sieht es als seine Aufgabe an, "das E-Mail-System des Internets vor dem Missbrauch der Spammer [zu] schützen." (freie Übersetzung der Autoren nach *mail-abuse.net*).

MAPS bietet verschiedene Dienste an. Unter anderem verwaltet MAPS die *Realtime Blackhole List* (RBL), eine Datenbank von Netzwerken, die sich gegenüber von Spammern freundlich oder neutral verhalten, oder die *Dial-Up User List* (DUL), eine Datenbank von dial-up-IP-Adressen von Providern, bei denen durch ein Übergehen des Mailservers leicht Massmailer verwendet werden können.

Neben diesen Diensten bietet MAPS noch mehrere andere Projekte. Näheres findet sich unter <http://www.mail-abuse.net>.

## **6. Schluss**

Wir hoffen, mit dieser Arbeit einen kurzen Überblick über die Techniken der Spammer und über Maßnahmen gegen sie gegeben zu haben. Für eine vertiefende Beschäftigung mit diesem Thema, insbesondere auch mit den Grundlagen der E-Mail-Kommunikation, möchten wir auf die unter den Quellenangaben angegebenen Internetseiten sowie das Buch von L. Hughes verweisen.

## **7. Quellenangaben**

computerbetrug.de. <http://www.computerbetrug.de>

Donnerhacke, Lutz, Teergruben FAQ, (Datum unbekannt).  
<http://www.iks-jena.de/mitarb/lutz/usenet/teergrube.html>

Gerlach, Carsten, Mini-FAQ: Falsche E-Mail-Adressen, 2001.  
[www.gerlo.de/falsche-email-adressen.html](http://www.gerlo.de/falsche-email-adressen.html)

Hochstein, Thomas, FAQ: E-Mail Adressen lesen und verstehen, 2001.  
<http://sites.inka.de/ancalagon/faq/headrfaq.html>

Hughes, Lawrence, Internet email: protocols, standards, and implementation.  
Artech House 1998

Lucke, Ken, Reading Email Headers, 1997.  
<http://www.stopspam.org/email/headers/headers.html>

Mail Abuse Prevention System.  
<http://mail-abuse.net>

McNamara, Rourke, Fake Mail FAQ, (Datum unbekannt).  
<http://www.geocities.com/SiliconValley/1947/Fakemail.htm>

Raz, Uri, How do spammers harvest email addresses?, (Datum unbekannt).  
<http://www.private.org.il/harvest.html>