Bachelorarbeit

Carsten Schulz

Evaluierung von Spamfilter-Mechanismen

Fakultät Technik und Informatik
Department Informatik

Faculty of Engineering and Computer Science
Department of Computer Science

Carsten Schulz

Evaluierung von Spamfilter-Mechanismen

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung
im Studiengang Angewandte Informatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr.-Ing. Martin Hübner
Zweitgutachter : Prof. Dr. Michael Neitzke

Abgegeben am 14.07.2008

Carsten Schulz

Thema der Bachelorarbeit

Evaluierung von Spamfilter-Mechanismen

Stichworte

Spam, Ham, Spamfilter, Evaluierung, Test, Vergleich, Antispam-Appliances

Kurzzusammenfassung

Die vorliegende Arbeit stellt ein Konzept zur Evaluierung von Spamfilter-Mechanismen vor. Desweiteren wird die Realisierung dieses Konzepts beschrieben. Für den Vergleich wurden Antispam-Appliances der Hersteller McAfee, Symantec, IronPort, SPONTS sowie CanIt herangezogen. Zusätzlich wurde die Filterqualität der Application Service Provider eXpurgate und SpamStopsHere sowie die Webmail-Anbieter WEB.DE und GMX einbezogen.

Carsten Schulz

Title of the paper

Evaluation of spam filter mechanisms

Keywords

spam, ham, spamfilter, evaluation, test, analysis, antispam appliances

Abstract

This assignment introduces a concept of spamfilter evaluation focusing on the implemented mechanisms as well as the realisation of this concept. For comparison antispam appliances developed by McAfee, Symantec, IronPort, SPONTS and CanIt were analysed. In addition the filter quality of the Application Service Provider eXpurgate and SpamStopsHere was prooved and the webmail provider WEB.DE and GMX were tested.

Inhaltsverzeichnis

Abbildungsverzeichnis **6**

1 Einleitung **7**
1.1 Zielsetzung und Gliederung . . . 8
1.2 E-Mail und SMTP . . . 9
1.3 UBE, UCE, Spam und Ham . . . 11
1.4 Wer sendet Spam und wie? . . . 13
1.5 Das Spam-Problem . . . 18

2 Grundlagen **20**
2.1 Verfahren zur Absendervalidierung . . . 20
2.1.1 Blacklists und Whitelists . . . 21
2.1.2 Absenderauthentifikation . . . 23
2.1.3 Challenge-Response-Verfahren . . . 25
2.2 Inhaltsfilter . . . 28
2.2.1 Prüfsummenverfahren . . . 29
2.2.2 Bayes-Filter . . . 30

3 Design **33**
3.1 Ansatz . . . 33
3.2 Aufbau und Ablauf . . . 34
3.3 Testkriterien . . . 37
3.3.1 Normierung . . . 38
3.3.2 Fragestellungen . . . 39

4 Realisierung **41**
4.1 Server . . . 41
4.2 Webserver . . . 42
4.3 E-Mail-Server . . . 43
4.4 Datenbank . . . 47
4.5 Spamfilter-Systeme . . . 48
4.5.1 Symantec Mail Security 8260 . . . 49
4.5.2 McAfee Secure Content Management-Appliance 3200 . . . 53

4.5.3 IronPort C10 Email Security Appliance 57
4.5.4 CanIt Anti-Spam Appliance 60
4.5.5 iku SPONTS Mail Security Appliance 63
4.5.6 eXpurgate 66
4.5.7 SpamStopsHere 68
4.5.8 WEB.DE und GMX 70

5 Auswertung **71**
5.1 Scan-Ergebnisse 72
5.1.1 Zuverlässigkeiten und Fehlerraten 73
5.1.2 WEB.DE und GMX 74
5.2 Weitere Tests und Beobachtungen 75
5.3 Spamfilter-Mechanismen 78
5.3.1 Der ASP-Ansatz 79

6 Zusammenfassung **80**

Literaturverzeichnis **82**

Abbildungsverzeichnis

1.1 Aufbau einer E-Mail . . . 9
1.2 Ablauf einer SMTP-Sitzung . . . 11
1.3 „bulletproof hosting“ . . . 15
1.4 Software zum Spammen . . . 16
1.5 Mailinglisten . . . 17

4.1 Symantec Mail Security 8260 . . . 49
4.2 Symantec Mail Security 8260 - Advanced Settings . . . 50
4.3 Symantec Mail Security 8260 - Sender Groups . . . 52
4.4 McAfee SCM 3200 . . . 53
4.5 McAfee SCM 3200 – Anti-Spam-Einstellungen . . . 55
4.6 McAfee SCM 3200 – Antispam-Regeln . . . 56
4.7 IronPort C10 . . . 57
4.8 IronPort C10 - Sender Groups . . . 59
4.9 CanIt Anti-Spam Appliance . . . 60
4.10 CanIt Anti-Spam Appliance - Administrator-Bereich . . . 61
4.11 CanIt Anti-Spam Appliance - Endnutzer-Bereich . . . 62
4.12 CanIt Anti-Spam Appliance - Bayes-Filter . . . 62
4.13 SPONTS-Appliance . . . 63
4.14 SPONTS-Appliance - UCE-Einstellungen . . . 65
4.15 eleven eXpurgate - Prinzip . . . 66
4.16 eleven eXpurgate - Portal . . . 67
4.17 SpamStopsHere - Portal . . . 68
4.18 SpamStopsHere - Domänen Optionen . . . 69

5.1 Zuverlässigkeiten und Fehlerraten . . . 73

1 Einleitung

Im Jahr 2007 gab es weltweit rund 1,4 Milliarden aktive E-Mail-Konten. Für die nächsten vier Jahre wird ein Anstieg auf bis zu zwei Milliarden Konten prognostiziert. Dies geht aus einer aktuellen Studie des Marktforschungsunternehmen Radicati Group hervor [The Radicati Group (2007)]. Die Firma Sophos meldete in ihrem regelmäßigen Security Report, dass 92,3 Prozent aller im ersten Quartal 2008 versendeten E-Mails als Spam klassifiziert wurden [Sophos (2008)]. IronPort erwartet für das Jahr 2008 zwischen 150 und 170 Milliarden Spam-Mails täglich [IronPort Systems, Inc. (2008c)]. Dies wäre eine Zunahme von etwa 40 Prozent im Vergleich zum bisherigen Rekordjahr 2007 mit einem Volumen von durchschnittlich 120 Milliarden unerwünschter Nachrichten pro Tag.

Die Zahlen machen deutlich, welche Belastungen auf Internet Service Provider oder auch größere Unternehmen mit entsprechend vielen E-Mail-Konten täglich zukommen. Gleiches gilt – wenn auch im geringeren Umfang – für den Privatanwender. Beschränkt man sich allein auf die Übertragungskosten, so lassen sich große Unterschiede zwischen kabelgebundenen und kabellosen Tarifen ausmachen. Gemäß Radicati lag die Anzahl mobil genutzter E-Mail-Konten 2007 bei ca. 25 Millionen. Jährlich wird hier ein Anstieg um 100 Prozent vorausgesagt. Ende 2011 sollen bereits 16 Prozent aller E-Mail-Konten weltweit über Mobilfunk erreichbar sein [The Radicati Group (2007)]. Da derzeit auf dem Mobilfunk-Markt hauptsächlich Zeit- und Volumentarife angeboten werden, kann es für den Nutzer spürbare finanzielle Auswirkungen haben, wenn neun von zehn zu übertragenden Nachrichten Spam sind.

Auf dem EU Spam Symposium 2006 nannte ein ehemaliger professioneller Spammer [Spammer-X (2006)] den Grund, weshalb jeder Einzelne auch morgen wieder eine Spam-Mail im Posteingang haben wird: Es wird immer Menschen geben, die die dort angepriesenen Produkte kaufen. Das bedeutet, dass das Spam-Problem von heute auf morgen vom Erdboden verschwunden wäre, wenn es niemanden mehr gäbe, der ein beworbenes Produkt kauft. So einleuchtend dies auch ist, so unwahrscheinlich ist es auch, dass es jemals eintreten wird.

Offensichtlich muss es andere Mittel und Wege geben, um die Spam-Flut wenigstens zu reduzieren. Neben einigen wenigen präventiven Maßnahmen der Spamvermeidung gibt es auf dem Markt eine Fülle an implementierten Mechanismen zur reaktiven Spambekämpfung. Diese finden sich teilweise in Freeware-Programmen für Privatanwender, teilweise in Anwendungen und Systemen für den professionellen Einsatz in Unternehmen. Eine Lösung für Unternehmen stellen die sogenannten *Appliances* dar. Dies sind eigens für spezielle Aufgaben konzipierte Systeme, in diesem Fall für die Filterung von E-Mails, die in der Regel vorkonfiguriert ausgeliefert werden. Oft spricht man auch von *Antispam-Appliances*. Sie werden in ein bestehendes Unternehmens-Netzwerk integriert und leisten so ihren Beitrag zur IT-Sicherheit.

1.1 Zielsetzung und Gliederung

Unerwünschte Werbe-E-Mails werden heutzutage hauptsächlich über Filter bekämpft. Dabei untersuchen Programme eine eingehende E-Mail auf typische Spam-Merkmale. In dieser Arbeit wird ein Konzept für einen Test vorgestellt, welcher die Güte von Filtersystemen verschiedener Hersteller (Antispam-Appliances) untersucht und vergleicht. Die Umsetzung dieses Konzepts in einer Experimentalumgebung stellt einen weiteren Abschnitt der Arbeit dar. Die aus dem Test gewonnenen Ergebnisse wurden analysiert und bewertet. Die Betrachtung der jeweils implementierten Spamfilter-Mechanismen stand hierbei im Vordergrund. Anhand der Erkennungsraten wurde versucht, den Nutzen der Mechanismen herauszustellen und zu bewerten. Ebenfalls wurde die Spamfilter-Güte zweier Webmail-Anbieter (WEB.DE und GMX) getestet. Die für den Test notwendigen Antispam-Appliances stellten die Hersteller in Form von Teststellungen zur Verfügung.

Nach notwendigen Begriffsdefinitionen und einer kurzen Erläuterung des Spam-Problems in einem einleitenden Kapitel folgt im Kapitel zwei eine Darstellung jener Spamfilter-Techniken, die von den untersuchten Filtersystemen angewandt wurden. Anschließend befasst sich Kapitel drei mit dem Konzept und Entwurf des Tests und definiert die für einen Vergleich herangezogenen Kriterien und Faktoren. Das vierte Kapitel zeigt implementierungsrelevante Details des Testaufbaus auf. Ebenfalls werden die Filtersysteme vorgestellt, die am Test teilgenommen haben. Hierbei werden hauptsächlich die eingesetzten Spamfilter-Verfahren betrachtet. Abschließend werden die gewonnenen Testergebnisse im Kapitel sechs aufgearbeitet und ausgewertet.

Ziel ist es, Spamfilter-Mechanismen zu identifizieren, die für ein Filtersystem geeignet, sinnvoll oder sogar unabdingbar sind. Gegebenenfalls sollen auch solche Mechanismen herausgestellt werden, die mehr versprechen, als sie letztendlich leisten.

1.2 E-Mail und SMTP

Eine auf elektronischem Weg in Rechnernetzwerken übertragene briefartige Nachricht nennt man *E-Mail* (engl.: electronic mail). E-Mail wird als der wichtigste und meistgenutzte Dienst im Internet angesehen.
Gemäß dem RFC 2822 [Resnick (2001)][1] besteht eine E-Mail aus Textzeichen („lines of characters“), die in *Header* und *Body* unterteilt werden. Der für die Übertragung wichtige Teil ist der Header. Er enthält zum einen alle Informationen, die zur Weiterleitung bzw. Auslieferung notwendig sind, und zum anderen Einträge über die bisherigen Zwischenstationen der E-Mail. Anhand eines Beispiels sollen mögliche Header-Einträge und ihre Bedeutung aufgezeigt werden:

From alex@mailgate.exam.ple Fri May 22 17:02:25 2005	Envelope Sender: gehört nicht zum Header, sondern ist Teil des SMTP-Dialogs.
Received: (qmail 12345 invoked by alias); Sat, 21 May 2005 13:51:40 +0000 Received: by server1 (Postfix, from userid 1000) id D344F45681; Sat, 21 May 2005 15:51:39 +0200 (CEST)	Mehrere "Received"-Zeilen zeigen den Weg, den die E-Mail vom Sender zum Empfänger genommen hat. Jeder Server, der die Mail weiterleitet, fügt seine Kennung und einen Zeitstempel am Anfang der E-Mail hinzu.
Date: Sat, 21 May 2005 15:51:37 +0200	Absendedatum
Subject: Der Sinn des Lebens	Betreff der E-Mail
Message-ID: <434571BC.807070@gmx.de>	Eindeutige Nummer dieser E-Mail
From: Alex Absender <alex@example.net>	Absenderadresse (auf diese Angabe ist kein Verlass)
To: Erwin Empfaenger <erwin@example.com>	Empfänger
Cc: ErwinsSekratariat <sekretariatqexample.com>	Eine Kopie der Mail wird auch an diese Adresse geschickt.
User-Agent: Mozilla Thunderbird 1.0.6	E-Mail-Programm des Absenders
In-Reply-To: <134535224@web.de>	Diese E-Mail ist eine Antwort auf die E-Mail mit dieser Message-ID.
	Eine Leerzeile, die das Ende des Headers markiert und damit die Kopfzeilen vom Nachrichtentext trennt.
Hallo Erwin, wir müssen einen Termin vereinbaren. Bis dann, Alex	Inhalt der Nachricht (Body)

Abbildung 1.1: Aufbau einer E-Mail
Reihenfolge und Syntax werden im RFC 2822 spezifiziert.

[1] RFC steht für Request for Comments. RFCs gelten als standardisierte Empfehlungen, die von der IETF (Internet Engineering Task Force), einer nicht-staatlichen, weltweit anerkannten Organisation, erlassen werden.

Die in blau hinterlegten Zeilen sind in typischen E-Mail-Programmen die für den Nutzer sichtbaren Teile einer E-Mail. Je nach Anwendung kann man aber auch die zusätzlichen Header-Informationen (rosa) angezeigt bekommen. Der gemäß RFC 2822 optionale Body einer E-Mail enthält die eigentliche Information. Zusätzlich können hier Signaturen, wie z.B. S/MIME[2] angehängt sein.

Für das Senden, Empfangen und Weiterleiten von E-Mails zeichnen sogenannte Mailserver verantwortlich. Direkt kommunizieren die Server über ihre jeweiligen Agenten, den MTAs (Mail Transfer Agents).

Vor dem Versenden wird die E-Mail zusätzlich mit einem virtuellen *Envelope* (Umschlag) versehen, auf welchem nochmals die Adressen von Absender und Empfänger stehen. Die Übertragung einer E-Mail von MTA zu MTA erfolgt schließlich über das *SMTP*-Protokoll[3], welches bereits 1982 in einem RFC spezifiziert wurde (RFC 821) [Postel (1982)]. SMTP findet sich im OSI-Referenzmodell in der Anwendungsschicht wieder.

Im Jahr 1995 wurde das Protokoll in Form von *ESMTP* (Extended SMTP) erweitert und in RFC 1869 [Klensin u. a. (1995)] spezifiziert. Beide Protokolle werden von heutigen MTAs unterstützt. Welches für die Übertragung genutzt wird, entscheidet das initiale Kommando des Sendes in der SMTP-Sitzung. Das Kürzel *HELO* steht für die Nutzung von SMTP, *EHLO* (Extended HELO) steht für die Nutzung von ESMTP.

Nachdem eine Verbindung zu einem anderen MTA aufgebaut wurde (standardmäßig auf Port 25), gestaltet sich der SMTP-Dialog wie folgt:

[2] MIME steht für Multipurpose Internet Mail Extensions und ist ein Kodierstandard, der die Struktur und den Aufbau von E-Mails und anderer Internetnachrichten festlegt. S/MIME ist eine Erweiterung und steht für Secure MIME, ein Standard für die Verschlüsselung und Signatur von MIME-gekapselter E-Mail durch ein asymmetrisches Kryptosystem.

[3] SMTP steht für Simple Mail Transfer Protokoll

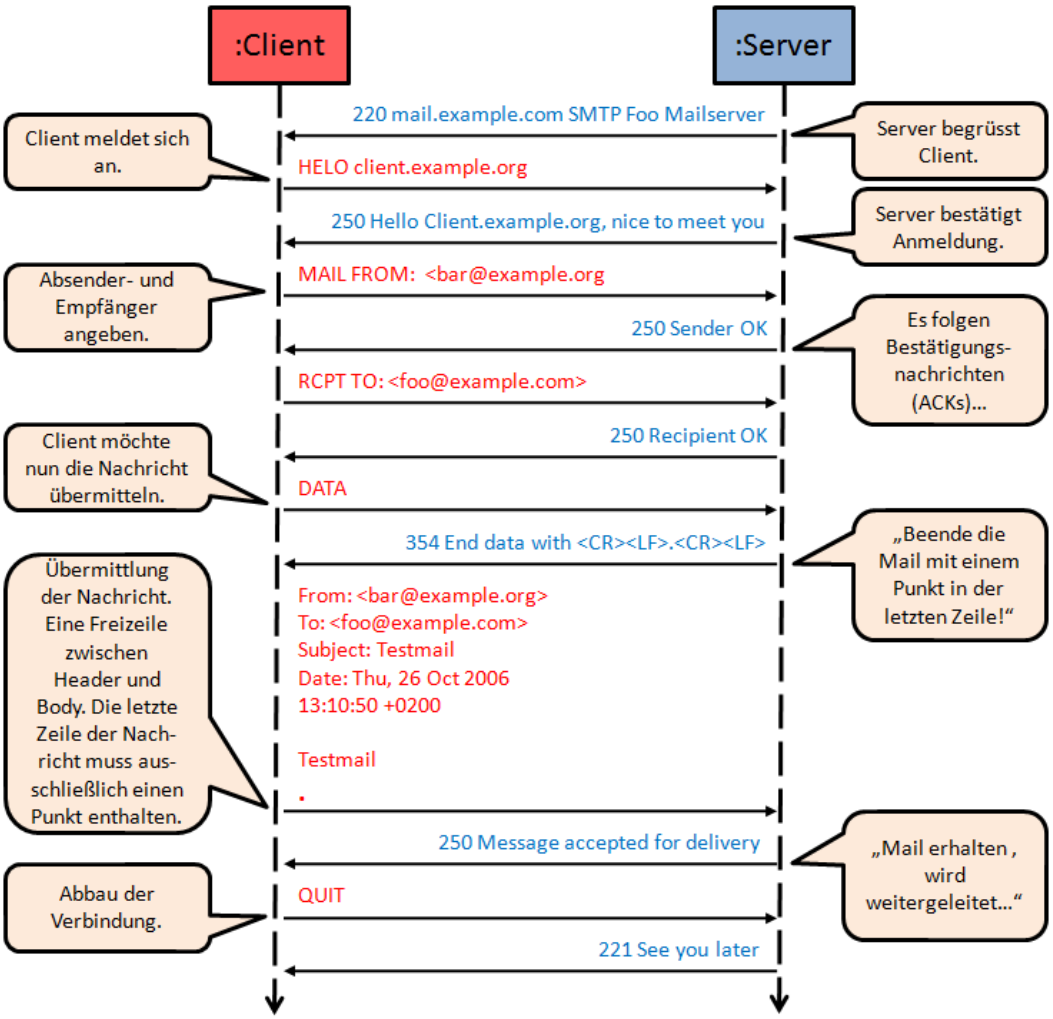


Abbildung 1.2: Ablauf einer SMTP-Sitzung

Wie später noch genauer erklärt wird, ist es einem MTA möglich, erkannte Spam-Nachrichten bereits im SMTP-Dialog abzuweisen. SMTP-Sitzungen können von den MTAs protokolliert werden. Dies ermöglicht dem Administrator eine sehr genaue Fehlersuche.

1.3 UBE, UCE, Spam und Ham

Bevor der Begriff „Spam“ genau eingeordnet werden kann, muß zunächst ein Oberbegriff von Spam, nämlich *UBE* geklärt werden. UBE steht für *Unsolicited Bulk E-Mail* (zu deutsch: unverlangte Massen-E-Mail) und damit für alle E-Mails, die massenhaft versendet werden und vom Empfänger nicht erwünscht sind. Dies bedeutet, dass die E-Mails an eine Vielzahl

von Empfängern gesendet werden, und dass das Einverständnis des Anwenders, solche E-Mails zu empfangen, in keinster Weise gegeben oder anzunehmen ist. Inhalt und Absender der E-Mail spielen dabei keine Rolle. So zählen beispielsweise Pishing-Mails genauso zu UBE wie Kettenbriefe oder die häufig von Freunden weitergeleiteten E-Mails lustigen Inhalts. Desweiteren fallen auch E-Mails in die UBE-Klasse, die religiöse, weltanschauliche oder politische Ideen, wie zum Beispiel rechtsextremistische Propaganda, verbreiten sollen.

Eine Untermenge von UBE ist *UCE* (*Unsolicited Commercial E-Mail*). Hier handelt es sich um all jene E-Mails, die einen kommerziellen Hintergrund haben. Diesmal spielt also der Inhalt eine wesentliche Rolle zur Abgrenzung von anderen UBEs. Mit UCE gleichzusetzen ist der Begriff ***Spam***. Die Begriffe sind synonym. Den Vorgang des Spamversendens an sich bezeichnet man als *Spamming* oder auch als *Spammen*, den Absender als *Spammer*.

Ursprünglich bezeichnete man mit Spam ein eher minderwertiges Dosenfleisch der US-Firma Hormel Foods[4]. „SPAM“ steht als Abkürzung für „**sp**iced h**am**“ (gewürzter Schinken). Die Firma akzeptiert die Abwandlung ihres eingetragenen Markenzeichens, erbittet sich jedoch, dass ihr Produkt in Großbuchstaben geschrieben wird, also „SPAM“.

Die Assoziation mit unerwünschten Nachrichten ist auf einen Sketch der britischen Comedy-Truppe Monty Python zurückzuführen. In diesem musste ein Gast in einem Restaurant feststellen, dass es kein Gericht ohne besagtes Dosenfleisch gab. Als dieser sich darüber beschweren wollte, startete eine Gruppe Wikinger am Nachbartisch ein Loblied auf SPAM und verhinderte dadurch jegliche weitere Form der Kommunikation.

Die erste E-Mail wurde 1971 über ein rudimentäres System verschickt. Die erste Spam-E-Mail wurde am 3. Mai 1978 im Arpanet[5] gesendet, als ein Ingenieur namens Gary Thuerk die Empfänger zu einer Produktpräsentation einlud. Den eigentlichen Startschuss gab allerdings ein amerikanischer Anwalt names Carter, der viele Internet-Newsgroups (Usenet) mit Werbung für seine Dienste überschwemmte. Die Reaktion der Internet-Gemeinde glich einem Aufschrei, und man verband diese Aktion schnell mit dem oben genannten Monty Python-Sketch. Dies prägte den Spam-Begriff so, wie man ihn heute kennt.

Typische Beispiele für Spam sind Angebote für Pharmazeutika, Software, Markenprodukte oder auch Finanzdienstleistungen. Es kann zwar nicht ausgeschlossen werden, dass die

[4] Hormel Foods Corporation: http://www.hormel.com und http://www.spam.com

[5] Das Arpanet (Advanced Research Projects Agency Network) wurde ab 1962 von einer kleinen Forschergruppe unter der Leitung des Massachusetts Institute of Technology und des US-Verteidigungsministeriums entwickelt und gilt als der Vorläufer des heutigen Internets.

Qualität der Produkte gut ist, meist handelt es sich jedoch um Fälschungen. Dies bestätigte sogar ein ehemaliger Spammer [Spammer-X (2006)]. Zudem kann man die Seriösität der Anbieter schon allein dadurch einschätzen, dass Spam als Werbemittel genutzt wird.
Häufig werden Links mit dem Ziel verschickt, den Empfänger der Mail auf eine Webseite zu locken. Hier hat man die Möglichkeit, die beworbenen Produkte zu kaufen. Anhand einer ID im Link kann dann nachvollzogen werden, dass der Anwender durch eine Spam-Mail auf die Seite gelangt ist. Spammer bekommen daraufhin in der Regel einen Anteil vom Kauferlös. Ebenfalls gibt es Webseiten, mit dem Ziel, durch möglichst viele Klicks mittels Bannerwerbung Geld zu verdienen.

Zur besseren Abgrenzung des Spam-Begriffs sei noch herausgestellt, dass E-Mails mit Viren, Trojanischen Pferden oder weiterer bösartiger Software nicht als Spam klassifiziert werden. Es ist auch nicht die Aufgabe von Spamfiltern, solche E-Mails zu erkennen oder auszusortieren. Diesen Schutz übernehmen Application-Level-Firewalls oder Antiviren-Programme.

Alle E-Mails, die kein Spam sind, bezeichnet man in Analogie zu SPAM als *Ham*. E-Mail-Filtersysteme haben demnach die Aufgabe, festzulegen, ob eine E-Mail Spam oder Ham ist. Es muss also entschieden werden, ob eine Nachricht unerwünscht bzw. unverlangt ist. Dies kann jedoch jeder Mensch individuell entscheiden. So ist es beispielsweise möglich, dass jemand es begrüßt, wenn ihm eine Möglichkeit der Bestellung von Pharmazeutika im Internet aufgezeigt wird. Es ist selbst für Menschen manchmal schwierig, E-Mails richtig zu klassifizieren. Diese Tatsache macht es der Software nicht gerade einfacher.

1.4 Wer sendet Spam und wie?

Schätzungen des Spamhaus Project [The Spamhaus Project (2008a)] besagen, dass es weltweit nur ca. 200 professionelle Spammer gibt, die für mindestens 80 Prozent des gesamten Spam-Aufkommens verantwortlich sind. Dies können einzeln agierende Personen oder kleine organisierte Gruppen sein, sogenannte *Spam Gangs*, die sich die Aufgaben teilen. In den 90er Jahren war es meist der Spammer selbst, der sich Produkte kaufte oder herstellte und diese dann via Spam anpries und schließlich verkaufte. Heutzutage haben Spammer in der Regel Abkommen mit Firmen, die dem Spammer pro verkauftem Artikel einen gewissen Anteil am Umsatz zukommen lassen.

Um Spam zu versenden, braucht es nicht viel: eine Internetverbindung, idealerweise eine DSL-Verbindung, spezialisierte Software, um mit geringem Zeitaufwand große Mengen von

Spam zu verschicken, und eine Mailingliste, also nichts anderes als eine Liste mit E-Mail-Adressen.

Oftmals wird eine analoge Einwählverbindung sogar bevorzugt, da diese ohne zeitliche Vorlaufzeit innerhalb von Minuten eingerichtet werden kann. DSL-Anschlüsse benötigen hingegen eine gewisse Vorlaufzeit und werden bei möglichen Beschwerden schnell gesperrt.

Die Motive eines professionellen Spammers dürften rein finanziell sein: Durch wenig Arbeit kann man verhältnismäßig viel Geld verdienen. Die Kosten, die ein Spammer hat, lassen sich gut abschätzen. Ein DSL-Anschluss mit Flatrate ist heutzutage schon für unter 20 Euro monatlich erhältlich. Professionelle Spammer nutzen zusätzlich das sogenannte *bulletproof hosting*, bei welchem es sich um Webserver handelt, die nie gesperrt werden. Das heisst, dass der Spammer einerseits seine Webseiten veröffentlichen und andererseits ungehindert Spam verschicken kann. Es werden keine Regulierungen oder Limitierungen vorgenommen, unabhängig vom Traffic auf der Domäne und unabhängig von der Anzahl eingehender Beschwerden. Eine einfache Suche nach „bulletproof hosting“ bei Google zeigt eine Vielzahl ominöser Anbieter solcher Dienste auf. Ein Beispiel für eine „all-inclusive“-Variante eines solchen Webhosting-Angebots wird in Abbildung 1.3 gezeigt. Ermöglicht wird dieses Vorgehen durch eine Gratwanderung der Anbieter entlang der Gesetzesgrenze. Oftmals werden Länder als Serverstandort ausgewählt, die keine oder nur eine unzulängliche Gesetzesregelung bezüglich der Internetnutzung aufweisen.

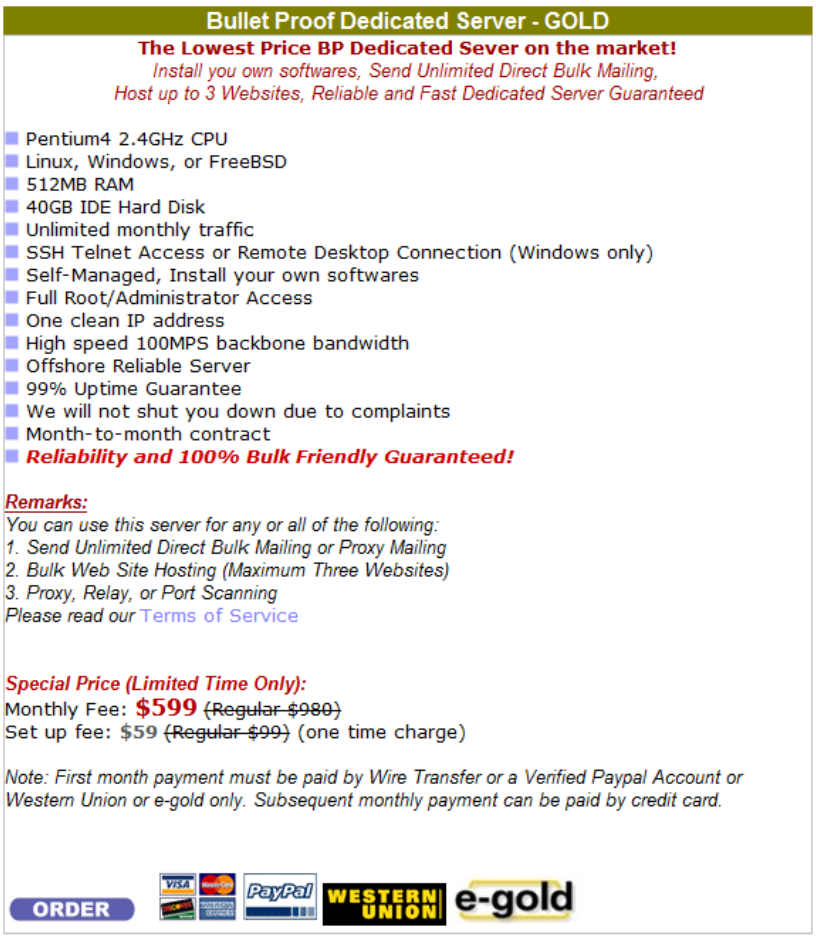


Abbildung 1.3: „bulletproof hosting“
Für $ 599 wird einem bei diesem Anbieter unlimitiertes E-Mail-Senden sowie die Gewissheit geboten, auch im Beschwerdefall nicht vom Netz genommen zu werden. [Sreenshot von http://www.bullet-proof-webhosting.com/#bulk-email-server, Stand: Mai 2008]

Schließlich benötigt der Spammer noch Software zum massenhaften Versenden von E-Mails. Diese wird entweder selbst programmiert, oder aber man bedient sich einfach im Internet. Sogar auf deutschen Download-Portalen wird man leicht fündig. So ergab eine Suche nach „bulk email“ im Downloadbereich der Seite von ZDNet.de[6] auf Anhieb vier verschiedene Tools zum Versenden von Bulk-E-Mails (siehe Abb.1.4). Hierzu sei angemerkt, dass die Suchergebnisse auf externe Webseiten verwiesen. Auch an E-Mail-Adressen kann man mit Leichtigkeit gelangen. Beispielsweise kommt man mit Hilfe der Suchfunktion des Filesharing-Programms eMule[7] relativ schnell an Archive mit mehreren Millionen E-Mail-Adressen, die teilweise sogar verifiziert sein sollen (siehe Abb. 1.5).

[6] http://www.zdnet.de
[7] http://www.emule.de/

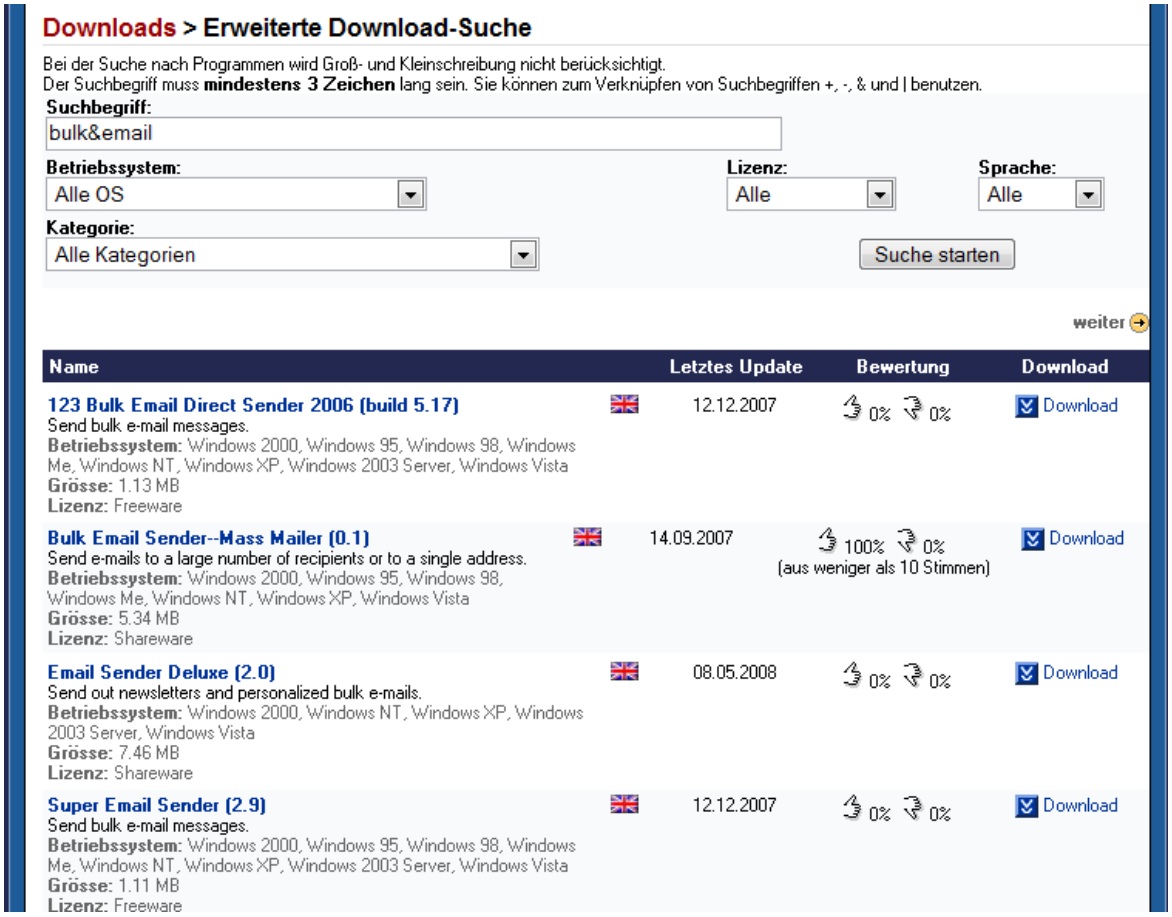


Abbildung 1.4: Software zum Spammen

Die Download-Suche auf ZDNet.de liefert ebenfalls Software zum Spammen. [Screenshot von http://www.zdnet.de/downloads/, Stand: Mai 2008]

Abbildung 1.5: Mailinglisten

Eine einfache Suche nach „spam email“ und schon kann man mit dem Filesharing-Programm eMule Archive mit angeblich verifizierten E-Mail-Adressen herunterladen. [Screenshot eMule, Stand: Mai 2008]

Den Aussagen eines ehemaligen Spammers [Spammer-X (2006)] zufolge kann man durch professionelles Spammen reich werden. Er nutzte die Worte „play the numbers game“ und meinte damit die Größenverhältnisse. Bei Millionen von versendeten Spam-Mails pro Tag muss nur ein Bruchteil der Empfänger ein Produkt kaufen, um Gewinn zu erzielen. Er illusstrierte dies an einem Beispiel: pro Tag verschickte eine Spam Gang bis zu 40 Millionen E-Mails, die für ein selbsthergestelltes („home made“) und somit gefälschtes Pharmazeutikum warb. Der entsprechende Link in der E-Mail wurde von 0,12 Prozent aller Empfänger angeklickt, was 48.000 Seitenaufrufen entsprach. Von diesen kaufte im Schnitt lediglich jeder Zweihundertste das beworbene Produkt. Es wurde ein Umsatz von $ 37.440 erzielt, wovon 50 Prozent ($ 18.720) direkt an die Spammer ging. Letztere hatten wöchentliche laufende Kosten von $ 11.030, die sich aus den Kosten für das bulletproof hosting ($ 230), für viertägigen Botnet-Zugang[8] ($ 6800) sowie für aktuelle Mailinglisten ($ 4000) errechneten. Folglich konnten die Spammer einen Gewinn von $ 7.390 erzielen. Rechnet man dies auf einen Monat auf, so wird deutlich, dass man durch Spammen weitaus mehr als sein täglich Brot verdienen kann.

1.5 Das Spam-Problem

So lästig und nervig Spam auch sein mag, man könnte trotzdem meinen, dass Spam ein zu verkraftendes Problem ist, und dass es im Internet weitaus schlimmere Gefahren gibt. Studien des Marktforschungsunternehmen Ferris Research kamen jedoch zu dem Ergebnis, dass Spam den Unternehmen weltweit Milliarden kostet [Ferris Research (2008)]. Konstatierte man den wirtschaftlichen Schaden im Jahr 2007 auf insgesamt 100 Milliarden Dollar, so geht man für das Jahr 2008 bereits von einer Summe von 140 Milliarden Dollar aus. Die Summe errechnet sich aus den Kosten für Antispam-Lösungen sowie den Produktionsausfällen der Mitarbeiter. Letztere bilden der Studie zufolge den weitaus größeren Anteil.

Wissenschaftler der Universität Hamburg fanden in einer empirischen Analyse unter 1000 Universitätsmitarbeitern heraus, dass die durch Spam verursachten Kosten pro Mitarbeiter im Jahr 2005 bei 447 Euro lagen [Clement u. a. (2008)]. Sie stellten ebenfalls fest, dass die Kosten auf IT-Ebene im Vergleich zu den Kosten auf Mitarbeiterebene zu vernachlässigen sind.

[8] Unter einem Botnet versteht man eine Gruppe von kleinen autonomen Computer-Programmen (Software-Bots). Betreiber illegaler Botnetze installieren diese auf netzwerkfähigen Computern ohne das Wissen der Inhaber und nutzen sie anschließend für ihre Zwecke, z.B. um Spam zu verschicken.

Dennoch entstehen den ISPs[9] Kosten, welche bedingt durch den zusätzlichen Traffic, der Nutzung der Mailserver- und Storage-Infrastruktur und dem zusätzlichen Personal, das für die Beseitigung und Vermeidung dieser Belastung notwendig ist, schnell anwachsen.

Berechnungen des Bundesamtes für Sicherheit in der Informationstechnik (BSI) haben gezeigt, dass für Großprovider finanzielle Kosten von 0,026 Cent pro Spam-Mail entstehen können [BSI (2005)]. Jährlich ergibt sich daraus für die ISPs eine zusätzliche Belastung von bis zu 1,43 Millionen Euro.

Anhand dieser Zahlen wird deutlich, dass es für eine Privatperson unbedeutend sein mag, wieviel Spam am Tag verschickt wird. Für Unternehmen und ISPs allerdings stellt Spam eines der größten IT-Probleme in der heutigen Zeit dar.

[9] Internet Service Provider

2 Grundlagen

In dieser Arbeit werden verschiedene Spamfilter-Systeme miteinander verglichen. Um eine qualifizierte Auswertung und Bewertung durchführen zu können, ist es notwendig, einen Blick hinter die Kulissen zu werfen. Sowohl hardware-basierte als auch die von Dienstleistern bereitgestellten Spamfilter bedienen sich diverser Antispam-Mechanismen. In diesem Kapitel werden zunächst die von den Systemen eingesetzten Verfahren und Techniken zur Spamfilterung vorgestellt. Ebenso werden Vor- und Nachteile der einzelnen Verfahren aufgezeigt. Eine individuelle und detailliertere Beschreibung der einzelnen Filtersysteme ist dem Kapitel vier zu entnehmen.

Im Wesentlichen klassifiziert man Antispam-Maßnahmen nach dem Zeitpunkt des Wirksamwerdens. So gibt es eine Reihe von Ansätzen, die bereits vor dem Versenden von Spam greifen sollen. Dies fängt bei der Rechtslage bezüglich des Versendens von Spam an und reicht bis zu dem Ansatz, E-Mails kostenpflichtig zu machen. Weiterhin existieren relativ neue Verfahren, die Spam während der Übertragung auf der Protokollebene, genauer gesagt während des Sende-Dialogs der Mailserver, aussortieren sollen. Methoden und Verfahren, die reaktiv, also nach dem Versenden einer Spam-Mail greifen, bilden eine weitere Klasse. Sie sind am Ende des Übertragungsweges auf der Empfängerseite angesiedelt. Da die Antispam-Appliances überwiegend dieser Klasse zuzuordnen sind, werden im folgenden ausschließlich solche Verfahren betrachtet. Bei Filtersystemen unterscheidet man zwischen herkunftsbasierten Verfahren und Inhaltsfiltern. Während sich erstere mit der Validierung des Absenders befassen, konzentrieren sich die Inhaltsfilter – wie der Name schon sagt – auf die E-Mail selbst.

2.1 Verfahren zur Absendervalidierung

Verfahren zur Absendervalidierung versuchen, die „kleine Designlücke“ von SMTP zu kompensieren: SMTP sieht keine Client-Authentifizierung vor, was bedeutet, dass theoretisch jeder SMTP-Client E-Mails an beliebig viele Mailserver senden kann [Ungerer (2007)]. Die hier vorgestellten Verfahren untersuchen und prüfen also den Ursprung einer E-Mail. Die dazu benötigten Informationen werden entweder aus dem SMTP-Dialog oder aus den Headern der E-Mail gewonnen. Sie beinhalten zum einen die IP-Adressen des sendenden Clients

und der passierten Mailserver und zum anderen die E-Mail-Adresse inklusive der Domäne des Absenders. Die oft auch als *reputationsbasiert* (engl.: reputation-based) bezeichneten Verfahren ermöglichen eine eventuelle Abweisung der E-Mail bereits während des SMTP-Dialogs. Der größte Teil aller Spam-Mails wird auf diese Weise abgewehrt.

2.1.1 Blacklists und Whitelists

Eine Möglichkeit, Absender zu blockieren, ist die Nutzung von sogenannten *Blacklists*. Dies können lokale, von einem Administrator selbst geführte Listen oder im Internet zentral oder verteilt geführte Listen sein. Als Oberbegriff liest man häufig *DNS-basierte Blacklists* (DNSBLs), da zur Abfrage dieser Listen die DNS-Technik genutzt wird. Ist ein Rechner oder eine ganze Domäne einmal durch Spamversand aufgefallen, so wird der jeweilige Namen und die jeweilige IP-Adresse in eine Blacklist eingetragen. Oft werden sogannte offene Mail-Relays oder mit Trojanern befallene Rechner (Zombie-Rechner), welche meist ohne Wissen des Anwenders massenhaft Spam verschicken, gelistet. Als offene Relays bezeichnet man Mailserver, die von jedem beliebigen Rechner (also auch außerhalb der zuständigen Domäne) E-Mails entgegen nehmen und an beliebige Adressen weiterleiten.

Eine sehr effektive Möglichkeit, dieses Verfahren zu nutzen, stellt die sogenannte *Realtime Blackhole List* (RBL) dar. RBLs sind Echtzeit-Blacklists im Internet, die von einem Mailserver abgefragt (DNS-Lookup) werden können, sobald die IP-Adresse des sendenden Mailservers bekannt ist, also noch vor dem SMTP-Dialog. Ist schließlich die IP-Adresse gelistet, so wird die Verbindung sofort wieder abgebaut und der SMTP-Dialog gar nicht erst gestartet. Optional kann der Mailserver meist noch eine Antwortnachricht im SMTP-Dialog (*Bounce*) generieren, aus der hervorgeht, dass der Sender in einer RBL gelistet ist. Neben kommerziellen DNSBLs findet man im Internet auch eine Vielzahl von kostenlosen Angeboten, welche meist durch kleinere Antispam-Organisationen bereitgestellt werden. Zu den bekanntesten Anbietern zählen Trend Micro's Email Reputation Services (früher RBL+, davor MAPS) [Trend Micro (2008)] sowie die von Spamhaus geführten SBL (Spamhaus Block List) und XBL (Exploit Block List) [The Spamhaus Project (2008b)].

Mitarbeiter der University of Tasmania erweiterten den Blacklist-Gedanken und entwickelten einen Ansatz names *Domain Specific Dynamic Blacklist* [Cook u. a. (2006)]. Sie führten die Logging-Einträge von Firewall, Intrusion Detection System und Mail Transfer Agent zusammen und stellten in einer Untersuchung fest, dass die von Spammern benutzten automatisierten Programme zum Versenden von Spam vorher meist eine „Probeverbindung" zum Netzwerk-Gateway aufbauen, bevor sie die eigentliche „Spam-Lawine" starten. Gleichermaßen sei es möglich, gewisse „Voranzeiger" von Zombie-Rechnern zu identifizieren.

Man entwickelte ein System, das ein Versenden von Spam-E-Mails sozusagen in real-time voraussagt und die entsprechende IP-Adresse unmittelbar vor dem Eintritt in das eigene Netz, also bereits am Gateway, blockt.

Das Gegenstück zu den Blacklists bilden die sogenannten *Whitelists*. Obwohl diese Listen auch – durch Dritte geführt – im Internet zur Verfügung gestellt werden, findet man sie viel häufiger im lokalen Einsatz vor. Öffentliche Whitelists enthalten Organisationen und Absender, die sich dafür verbürgt haben, keinen Spam zu versenden. Letzteres ist zwar ein löbliches Ziel, kann allerdings nie hunderprozentig ausgeschlossen werden, denkt man beispielsweise an einen unachtsamen Mitarbeiter, der einen noch nicht durch ein Virusprogramm erkennbaren Trojaner auf seinem Rechner hat. Steht ein Absender auf einer Whitelist, so wird dessen E-Mail ohne weitere Filterung zum Empfänger weitergeleitet. Vor allem im lokalen Einsatz ist es dem Adminstrator durch eine Whitelist möglich, die Spamfilterung individuell an das Unternehmen bzw. die Zielgruppe anzupassen.

Vor- und Nachteile

Vorteile von Black- und Whitelists sind die einfache Wartbarkeit und der sparsame Umgang mit Ressourcen [Ungerer (2007)]. Jedoch liegt in der Wartung auch gleichzeitig ein Problem dieser Techniken. Öffentliche Listen müssen stets aktuell gehalten werden. Wenn eine Organisation versehentlich einmalig Spam versendet (möglicher Weise durch einen befallenen Rechner) und so auf einer Blacklist landet, möchte sie natürlich schnellst möglich wieder von der Liste gestrichen werden. Betreiber öffentlicher RBLs bieten diesen Service zwar an, jedoch ist meist eine manuelle Bearbeitung einer solchen Anfrage notwendig, um Missbrauch zu vermeiden. Analog muss eine Organisation von einer Whitelist gestrichen werden, wenn sie plötzlich Spam versendet. Ernstzunehmende Anbieter von Blacklists verzichten generell auf den Eintrag eines großen Providers [Ungerer (2007)].

Der Einsatz von Blacklists lohnt sich. Da Spam bereits im SMTP-Dialog abgefangen wird, bleibt das eigene Netzwerk verschont. Durch die Benutzung von DNS-Abfragen ist der CPU-Overhead ebenfalls gering [Cook u. a. (2006)]. Jedoch benutzen Mailserver meist nicht nur eine Blacklist. Die Folge ist eine erhöhte Belastung des DNS-Systems. Eine Studie im Hause des MIT Computer Science and Artificial Intelligence Laboratory[1] im Jahr 2004 fand heraus, dass 14% aller eigenen DNS-Lookups Blacklist-Lookups waren [Jung und Sit (2004)].

Ein weiterer Nachteil einer Blacklist ist, dass ganze IP-Bereiche auf die Liste kommen könnten, obwohl nur ein einzelner Anwender Spam verschickt hat. Es besteht also die Gefahr,

[1] http://www.csail.mit.edu

dass ein größeres Unternehmen für eine gewisse Zeit nicht über E-Mail nach außen kommunizieren kann, was finanzielle Folgen nach sich ziehen könnte. So geschah es noch vor wenigen Jahren von Zeit zu Zeit, dass sogar E-Mail-Provider wie GMX kurzzeitig auf einer Blacklist standen und so viele Account-Inhaber vom Mailversand ausgeschlossen waren [Kuri (2005)]. Ebenso besteht die Gefahr der „Lähmung“ durch ein Konkurrenzunternehmen, indem man einfach dafür sorgt, dass die entsprechende Domäne auf einer Blacklist landet.

Blacklists werden meist von externen Anbietern geführt. Die Benutzung einer solchen Liste schließt demnach das Vertrauen in den Anbieter mit ein. Auf die korrekte und vor allem aktuell gehaltene Führung der Liste hat man keinerlei Einfluss. Gerade bei kostenlos nutzbaren Blacklists ist daher Vorsicht geboten. Der verantwortliche Administrator sollte regelmäßig kontrollieren, ob die Liste überhaupt noch geführt wird. Hier können Internetauftritte wie DNSBL Resource[2] hilfreich sein. Sie konzentrieren sich auf Neuigkeiten, insbesondere auf das „Absterben“ von Blacklists.

Aufgrund der aufgeführten Nachteile werden Blacklists häufig zusammen mit weiteren Antispam-Maßnahmen benutzt. So kann man spezifizieren, zu wieviel Prozent die Blacklist in die Gesamtentscheidung einfliessen soll. Generell sollten sich Administratoren und sonstige Entscheidungsträger immer überlegen, ob sie die Entscheidung über Spam und Ham in die Hände Dritter legen möchten. Wenn sie dies machen, muss sichergestellt werden, dass man dem Anbieter bezüglich Sorgfalt und Wartungsintervallen sowie schneller Erreichbarkeit im Notfall vertrauen kann.

Whitelists entlasten das übrige Spamfilter-System, da E-Mails von Absendern, die auf der Liste stehen, ohne weitere Prüfung direkt zum Empfänger weitergeleitet werden. Sollte ein Spammer jedoch eine auf einer Whitelist eingetragene Adresse erraten und diese als Absender-Adresse nutzen, so entwickelt sich dieser Vorteil schnell zu einem gravierenden Nachteil: jede Spam-Mail wird ungehindert an den anderen Filtersystem vorbei direkt zugestellt [Allman (2003)]. Aus diesem Grund sollten auch Whitelists – genau wie Blacklists – ausschließlich mit anderen Antispam-Mechanismen zusammen angewandt werden [Garcia u. a. (2004)].

2.1.2 Absenderauthentifikation

Eine große Gefahr, derer man sich bewusst sein muss, ist die des *Mail-Spoofings*. Mail-Spoofing bedeutet die Verfälschung oder Verschleierung der Identität des Absenders. Spammer nutzen dies, um ihre eigene Identität nicht preiszugeben und sich somit der Strafverfolgung zu entziehen. Mit Ausnahme der IP-Adresse des sendenden Mailservers können alle anderen Informationen über den Absender sowohl in der SMTP-Ebene als auch

[2] http://www.dnsbl.com

im Header gefälscht werden. Ermöglicht wird dies dadurch, dass das SMTP-Protokoll keine Integritäts- und Authentifikationsmaßnahmen vorsieht.

Sender Policy Framework

Um diesem Problem zu entgegnen, gibt es eine Reihe zusätzlicher Antispam-Mechanismen, die eine solche Absenderauthentifikation durchführen sollen. So wird beim *Sender Policy Framework (SPF)* in der DNS-Zone des Absenders ein sogenannter *Resource Record*[3] in Form einer txt- oder spf-Datei hinterlegt. Er enthält Informationen darüber, welche Rechner aus der jeweiligen Domäne E-Mails versenden dürfen. Der empfangende Mailserver kann ihn sich bereits während des SMTP-Dialoges, nachdem er die `MAIL FROM:`-Information bekommen hat, in der angegebenen Absenderdomäne einholen. Sofern diese einen SPF-Record hinterlegt hat, kann nun die IP-Adresse des sendenden Mailservers, gegebenenfalls unter Zuhilfenahme eines weiteren DNS-Lookups, mit den für die Domäne berechtigten Mailservern abgeglichen werden. Es wird also die Domäne des Absenders sowie die Berechtigung des Mailservers verifiziert. Schlägt dies fehl, so liegt ein DNS-Spoofing vor [The SPF Projekt (2008)]. SPF wurde 2003 entwickelt und im Jahr 2006 als RFC [Wong und Schlitt (2006)] spezifiziert. Derzeit befindet es sich allerdings noch im „Experimental“-Status .

Sender-ID

Ein sehr ähnliches Verfahren wurde von Microsoft entwickelt und wird mit den Namen *Sender-ID* bezeichnet [Microsoft (2006)]. Es stellt eine Mischform aus SPF und dem ebenfalls von Microsoft entwickelten *Caller-ID* dar. Nach heftigen Lizenzstreitigkeiten mit der Open-Source-Gemeinde erlangte Sender-ID im Jahr 2006 ebenfalls den „Experimental“-Status in Form des RFC 4406 [Lyon und Wong (2006)].

Sender-ID unterscheidet sich vom SPF dadurch, dass man hier eine verantwortliche *Purported Responsible Address* (PAR, zu deutsch in etwa: vorgegebene verantwortliche Adresse) benutzt, die sich auf ein einzelnes Postfach bezieht, welches die Übertragung der E-Mail initiert.

Ein großer Nachteil der beiden beschriebenen Techniken ist das Problem des Weiterleitens. Wenn der Empfänger seine E-Mails automatisch in ein anderes Postfach weiterleiten lässt und die sendende Domäne durch einen Record-Eintrag geschützt ist, so wird das Zielsystem nach der Weiterleitung annehmen, dass die Adressangaben gefälscht wurden, da die

[3] kleinste Informationseinheit im DNS

E-Mail von einem nicht-autorisierten Mailserver verschickt wurde. Abhilfe kann hier zum einen der Empfänger selbst schaffen, indem er beim Provider seines Zielsystems einen Whitelisting-Eintrag für E-Mails vom weiterleitenden System veranlasst. Bedingung wäre allerdings, dass jenes System dann die SPF-Prüfung vornimmt. Zum Anderen existiert eine Technik names *Sender Rewriting Scheme (SRS)*. Hier schreibt das weiterleitende System die Envelope-Absenderadressen der weitergeleiteten E-Mails auf seine Domäne um. Es zeichnet dann aber auch dafür verantwortlich, dass die ursprünglichen Absenderadressen verifiziert wurden und richtig sind. Eventuelle durch das Zielsystem veranlasste Bounce-Mails erreichen dann auch wieder das weiterleitende System. Genau genommen handelt es sich bei SPF und Sender-ID nicht wirklich um Antispam-Techniken, da sie nicht Spam bekämpfen, sondern lediglich die Absender verifizieren und Adressfälschungen aufdecken. Betrachtet man das in Kapitel eins erwähnte „bulletproof hosting“, so kann man desweiteren davon ausgehen, dass Spammer ebenfalls über SPF-verifizierbare Domänen verfügen.

DomainKeys

Mit *DomainKeys* gibt es ein weiteres Verfahren zur Authentifizierung von Absenderadressen. Das von Yahoo entwickelte und in RFC 4871 [Allman u. a. (2007)] standardisierte Verfahren bedient sich digitaler Signaturen, also dem *Public Key*-Verfahren. Hier wird eine E-Mail auf Senderseite mit einem privaten Schlüssel verschlüsselt und das Ergebnis als Signatur mit verschickt. Das Zielsystem kann sich den in der DNS-Zone hinterlegten zugehörigen öffentlichen Schlüssel herunterladen und damit die Nachricht verifizieren. Der DNS-Server wird also auch bei diesem Verfahren „zweckentfremdet“ und dient hier als Zertifizierungstelle. Obwohl DomainKeys keine wesentlichen Nachteile hat, tut es sich jedoch, genau wie die anderen Authentifizierungs-Mechanismen, in der Verbreitung sehr schwer. Ein Grund dafür liegt –wie auch bei SPF und Sender-ID – in der erforderlichen größeren Konfigurationsanpassung der Mailserver.

2.1.3 Challenge-Response-Verfahren

Spammer versenden ihre E-Mails in der Regel in sehr großen Mengen und mit gefälschten Absenderdaten. Das erfolgreiche Versenden von E-Mails gestaltet sich für den Spammer jedoch schwierig, sobald eine zusätzliche Aktion, z.B. die Bestätigung eines Links hinzukommt. Diese Aktion bzw. zusätzliche Aufgabe nennt man *Challenge*, die Antwort darauf wird als *Response* bezeichnet. Wenn nun gefälschte Absenderdaten benutzt wurden, kommt diese Challenge nie beim Spammer an, und die E-Mail wird nach einer festgelegten Zeitspanne vom Mailserver verworfen. Ist dies nicht der Fall, kommen theoretisch genauso viele

Challenges bei ihm an, wie E-Mails verschickt wurden. Letzteres dürfte nicht im Interesse eines Spammers liegen.

Challenge-Response-Verfahren können als eine Erweiterung des Whitelist-Verfahrens angesehen werden. Absender, die nicht gelistet sind, können durch die einmalige Beantwortung der Challenge auf die Whitelist gelangen und fortan das Spamfilter-System direkt durchschreiten [Pfleeger und Bloom (2005)]. Die Schwäche von Whitelists bezüglich unbekannter Absender wird auf diese Art kompensiert.

Besonders wirkungsvoll sind Challenge-Response-Verfahren gegen automatisierte Mail-Programme. Da sich Challenges von Mailserver zu Mailserver unterscheiden können, ist es den Programmen nicht möglich, automatisiert zu antworten. In der Zeit, in der ein Spammer persönlich mit der Beantwortung einer Challenge beschäftigt wäre, könnte er ebenso gut Spam an weitere E-Mail-Adressen versenden.

Problematisch ist es jedoch, dass oft „Unschuldige“ mit Challenges belästigt werden. Dies ist der Fall, wenn Absenderadressen gefälscht wurden. Existiert die Adresse zufällig, so trifft die Challenge auf den eigentlichen Inhaber dieses Postfachs und nicht auf den Spammer. Ein weiterer Nachteil ist die Gefahr von Deadlocks: Wenn zwei Parteien niemals zuvor miteinander kommuniziert haben, und beide Mailserver durch Challenge-Response-Verfahren vor Spam geschützt sind, würde eine E-Mail von A nach B wie vorgesehen eine Challenge auslösen. Diese Challenge-E-Mail würde jedoch bei B ebenfalls eine Challenge auslösen. Keine der Parteien hätte die Möglichkeit, eine entsprechende Antwort zu geben [Barracuda Networks (Datum unbekannt)]. Ähnlich verhält es sich bei automatischen Mailing-Listen, bei denen man sich anmelden kann. Diese Listen können auf keine Challenge antworten. E-Mails würden daher ebenfalls nicht ankommen. Beide Probleme könnte man dadurch lösen, dass man vor Beginn der Kommunikation den entsprechenden Gegenüber bzw. die Adresse der Mailing-Liste in seine eigene Whitelist einträgt [Cook u. a. (2006)].

Eine weitere Schwäche von Challenge-Response-Verfahren stellt die erhöhte Netzwerkbelastung dar. Wäre jeder Mailserver mit dieser Technik ausgestattet, so hätte man einen bis zu dreimal so hohen Netzwerkverkehr.

Greylisting

Eine Technik, die dem Challenge-Response-Verfahren zuzuordnen ist, nennt sich *Greylisting* [Lundgren IT (2008)]. Greylisting nutzt die meist relativ einfach gehaltene, auf den

Massenversand ausgelegte Implementierung der Bulkmailer-Software aus. Bevor eine SMTP-Session aufgebaut wird, sind dem empfangenden Mailserver bereits die IP-Adresse sowie die E-Mail-Adressen von Absender und Empfänger bekannt. Dieses Tripel wird in einer Datenbank abgelegt. Möchte nun ein Mailserver eine Verbindung aufbauen, prüft das Zielsystem in der Datenbank, ob ein solches Tripel bereits existiert. Falls nicht, wird eine Fehlermeldung zurückgeschickt (Code 451: „Try again later“). An diesem Punkt wirkt sich nun die Implemtierung der Software aus: Primitive Spam-Software ist nicht auf eine solche Fehlermeldung ausgelegt und wird die E-Mail verwerfen. Ein richtig konfigurierter MTA akzeptiert diese RFC-konforme Antwort und wird nach einer gewissen Zeitspanne (z.B. 15 Minuten) erneut eine SMTP-Verbindung aufbauen wollen. Nun ist das Tripel bereits bekannt, und die E-Mail wird zugestellt. Durch den Einsatz von Whitelists kann man auch hier gewissen Domänen oder Adressen ein ungehindertes Durchkommen ermöglichen.

Greylisting ist eine relativ aggressive Methode der Spam-Bekämpfung, da der erste SMTP-Dialog konsequent abgebrochen wird. Es ist aber durchaus wirkungsvoll. Problematisch wird es, wenn ein MTA schlecht oder falsch konfiguriert wurde. Dann kann es vorkommen, dass eine E-Mail komplett verworfen wird und ihr Ziel nie erreicht. Nachteilig ist die Zeitverzögerung, die davon abhängig ist, wie schnell der Sender einen neuen Versuch unternimmt. Dies betrifft jedoch lediglich die erste E-Mail. Alle weiteren E-Mails können zügig passieren, sofern das Tripel lange genug abgespeichert wird.

Nicht unerwähnt sollte bleiben, dass Spammer ihre Software anpassen können, um einen zweiten Sende-Versuch zuzulassen. So ist man zwar in der Lage, Greylisting zu überwinden, man verliert jedoch Zeit. Lässt man auf der Empfängerseite eine erneute Zustellung beispielsweise erst nach Ablauf einer Stunde zu, könnte diese Spam-Welle bereits von anderen Systemen identifiziert und womöglich in Blacklists aufgenommen worden sein. Dies würde bedeuten, dass man eine Stunde gewonnen hat und vor neuem Spam verschont geblieben ist. Somit ist der Einsatz von Greylisting trotz möglicher Softwareaktualisierungen auf Seiten des Spammers weiterhin denkbar.

Weitere Techniken, die das Challenge-Response-Verfahren benutzen, seien hier mit *ChoiceMail* [DigiPortal Software (2008)], einem Registrierungsverfahren und *SFM (Spam Free e-Mail Service* [University of Alberta (2008)], einem Verfahren, das dynamische Adressen benutzt, lediglich erwähnt.

2.2 Inhaltsfilter

Die bisher beschriebenen Mechanismen gewinnen die Informationen ausschließlich aus den Headern der E-Mails und den SMTP-Dialogen. Die folgenden Mechanismen konzentrieren sich auf den übrigen Teil der E-Mail, der `SUBJECT:`-Zeile und den Body. Der Einsatz der hier aufgeführten Verfahren bedeutet auch, dass die E-Mail bereits alle reputationsbasierten Antispam-Maßnahmen durchlaufen hat und bisher nicht abgewiesen wurde.

Der Vorteil von Inhaltsfiltern liegt in der individuellen Konfigurierbarkeit. So sind sie auch in solchen Unternehmen einsetzbar, die geschäftlich mit häufig bespammten Produkten zu tun haben. Hierzu zählen beispielsweise Pharmaunternehmen und Ärzte („Viagra“) oder Geschäfte, die mit Uhren handeln („Rolex“), um nur zwei Beispiele zu nennen.

Die wohl bekannteste Form der Inhaltsfilter ist der *Schlüsselwortfilter*, welcher den Body und die Betreff-Zeile der E-Mail auf solche Wörter untersucht, die in typischen Spam-Mails vorkommen. Dies sind beispielsweise die Namen der beworbenen Artikel oder aber Redewendungen wie „total free“, „letzte Chance“, „buy now“, etc. Man unterscheidet halbautomatische und vollautomatische Filter. Während erstere erkannte Spam-Nachrichten in einen dafür vorgesehenen Bereich aussortieren, löschen die vollautomatischen Filter die E-Mail sofort.

Spammer versuchten schnell, durch geschicktes Verändern der Wörter die Schlüsselwortfilter zu umgehen. So wurde beispielsweise aus dem Wort „VIAGRA“ das Wort „V1AGRA“ oder „V.I.A.G.R.A“. Der Phantasie sind dabei keine Grenzen gesetzt.

Weiterhin werden Spam-Mails zum Teil derart individuell verschickt, dass keine E-Mail der anderen gleicht. Hierzu braucht man lediglich andere Anreden im Text oder einfach eine eindeutige ID irgendwo in der E-Mail. Diese Ansätze versucht man mit dem sogenannten *Pattern Matching* (Mustererkennung) zu erkennen. Hierbei werden E-Mails miteinander verglichen und auf gleiche Muster hin untersucht. Diese Erkenntnis geht dann meist zusammen mit anderen Antispam-Verfahren in die Klassifizierung mit ein.

Eine andere Form der Inhaltsfilterung ist die *heuristische Inhaltsanalyse*, oft auch bezeichnet als *regelbasierende Filter*. Anhand von Regeln, welche meist in Form von regulären Ausdrücken angegeben sind, wird so jede E-Mail einer Prüfung unterzogen. Abschließend werden die erhaltenen Ergebnisse mit individueller Gewichtung aufsummiert. Entsprechend einem einstellbaren Schwellwert wird die Klassifizierung der E-Mail durchgeführt. Das Endergebnis kann, wie bei allen anderen Verfahren auch, optional wieder mit anderen

Antispam-Mechanismen zusammen betrachtet werden, um Spam von Ham zu unterscheiden. Es können unterschiedlichste Regeln aufgestellt werden, zum Beispiel das Herausfiltern aller E-Mails mit HTML-Code oder E-Mails mit bestimmten Wörtern. Ein Schlüsselwortfilter kann demnach durch eine einzige Regel angegeben werden. Andere Regeln könnten E-Mails mit überdurchschnittlich vielen Großbuchstaben aussortieren. Regelbasierende Filter waren bis zum Jahr 2002 der am meisten verbreitete Filtertyp. Sie wurden von den Bayes-Filtern (siehe weiter unten) abgelöst [Graham (2003)].

Ein Nachteil der regelbasierenden Filter ist, dass sie bereits durch geringfügige Variationen in der Spam-Mail nicht mehr greifen, da nicht alle Variationen von Wörtern mit Regeln abgedeckt werden können (z.B. „Free“ vs. „F*r*e*e“) [Cook u. a. (2006)].

2.2.1 Prüfsummenverfahren

Das sogannte verteilte Prüfsummenverfahren (engl.: *Distributed Checksum Clearinghouse, DCC*) [Rhyolite Software (2008)] ist ein Client-Server-Modell, bei dem der Client, z.B. ein einzelner Rechner mit einem Spamfilter-Programm oder ein SMTP-Server, eine Prüfsumme anhand der eingegangenen E-Mail errechnet und diese zu einem öffentlichen DCC-Server übermittelt. Die Prüfsumme kann zur eindeutigen Identifikation dieser E-Mail beitragen und bleibt bei kleinen Unterschieden in der E-Mail gleich. Wenn der Spammer also jede E-Mail mit einer anderen Anrede oder einer anderen zufälligen Zahl versieht, bleibt die errechnete Prüfsumme dennoch unverändert. Die DCC-Server zählen indes die Häufigkeit der gleichen Prüfsumme und erkennen so Spam-Mails vom selben Typ. Die Clients können sich diese Information schließlich abrufen und die Spam-Klassifizierung durchführen.

Eine leicht von DCC abweichende Implementierung des Prüfsummenverfahrens findet sich in *Vipul's Razor* [Prakash (2007)] wieder. Es unterscheidet sich dadurch, dass hier der Anwender selbst entscheidet, welche Prüfsummen an den Server gesendet werden. Die Server-Datenbank wird also ausschließlich mit den Prüfsummen manuell-gesichteter Spam-Mails gefüllt. Bei diesem Verfahren können Unstimmigkeiten auftreten, da es, wie in der Einführung beschrieben, durchaus unterschiedliche Auffassungen über die Klassifizierung als Spam oder Ham gibt. Es kann zudem nicht ausgeschlossen werden, dass ein unerfahrener Anwender diese Entscheidung treffen muss. Diesem Problem wurde Abhilfe geschaffen, indem jeder Anwender eine vom Server generierte ID zugewiesen bekommt, welche immer zusammen mit der Klassifizierung mitgeschickt wird. Der Server wägt die Entscheidungen aller Anwender miteinander ab und vergibt Erfahrungspunkte. So wird jedem Anwender ein gewisser Vertrauensgrad zugeordnet, der Einfluss auf die Gewichtung des Votings ausübt. Dies ist notwendig, da einfache Mehrheitsentscheidungen gefährlich sein können. Je nach

Branche des Unternehmens kann nicht immer davon ausgegangen werden, dass die Angestellten den nötigen Kenntnisstand über Spam haben. So ist es denkbar, dass anfangs die Anwender zwar voten, um ihren Vertrauensgrad aufzubauen, die Entscheidungen trifft jedoch zunächst ein erfahrener Administrator.

2.2.2 Bayes-Filter

Ein etwas anderer Ansatz der Spamfilterung geht in den Bereich der KI[4]. Da sich die Inhalte von Spam-E-Mails mit der Zeit regelmäßig ändern, braucht man eine Möglichkeit, auch die neueren Spam-Trends sofort zu erkennen. Hierbei bedient man sich der Statistik, genauer gesagt bei dem Jahrhunderte altem Wahrscheinlichkeitstheorem des Mathematikers *Thomas Bayes*. Die mit bedingten Wahrscheinlichkeiten arbeitenden sogenannten *Bayes-Filter* sollen vorhersagen, ob eine bisher unbekannte E-Mail als Spam zu klassifizieren ist oder nicht. Zum ersten Mal wurden die *naiven Bayes-Filter* von Microsoft Research [Dumais u. a. (1998)] sowie in den Arbeiten von Pantel und Lin [Pantel und Lin (1998)] propagiert.

Funktionsweise

Jede eingehende E-Mail wird in sogenannte *Tokens* (Zeichen oder auch kurze Zeichenketten) zerlegt. Anschließend wird jedem Token eine Spamwahrscheinlichkeit (engl.: *spamminess*) und gleichzeitig auch eine Hamwahrscheinlichkeit (engl.: *hamminess*) zugeordnet. In einer Datenbank – meist in Form eines Wörterbuchs – sind für bereits bekannte Token die Spamwahrscheinlichkeiten abgespeichert. Neuen Tokens wird eine initale Wahrscheinlichkeit zugeordnet, die meist knapp unter einem festgelegten Spam-Schwellenwert (z.B. 0,5) gewählt ist [Graham (2002)]. Z.B. würde eine Spamwahrscheinlichkeit von 0,4 bedeuten, dass das Token eher ein „HAM-Indikator“ ist, eine Wert von 0,6 würde dagegen bedeuten, dass es eher ein Spam-Indikator ist. Aus den Spamwahrscheinlichkeiten aller Token wird anschließend die Gesamt-Spamwahrscheinlichkeit der E-Mail berechnet. Es sei angemerkt, dass spamverdächtige Token die Gesamtwahrscheinlichkeit zwar erhöhen, nicht spamverdächtige Wörter verringern sie jedoch wieder [Graham (2003)]. Ein *naiver Bayes-Filter* kann dabei keine Zusammenhänge innerhalb des Textes herstellen. Er geht davon aus, dass jedes Token unabhängig voneinander auftritt.

Bevor Bayes-Filter einigermaßen zuverlässig genutzt werden können, müssen sie trainiert werden. Sie müssen eine bestimmte Anzahl an Spam- und Ham-E-Mails verarbeitet haben, um eine richtige Klassifizierung durchführen zu können. Im günstigsten Fall „füttert“ man

[4] künstliche Intelligenz

den Filter mit einigen hundert E-Mails, lässt ihn klassifizieren und überprüft anschließend jede einzelne Entscheidung manuell. Falsche Einstufungen müssen korrigiert werden. Das erste „Training“ wird meist mit einem Grunddatensatz von Ham- und Spam-E-Mails, welcher vom Entwickler bereitgestellt wird, durchgeführt. Die weitere Feinabstimmung kann bei aktuellen Verfahren mittels einer nachträglichen E-Mail-Markierung durch den Anwender selbst übernommen werden. Er kann auf eingefügte Links klicken (einen für Spam und einen für Ham) und so den Filter trainieren. Mit jeder verarbeiteten E-Mail lernt der Filter dazu. Die Filter benötigen anfangs unbedingt aktuelle persönliche Ham-E-Mails, um die Unterschiede zu lernen, und um später eine versehentliche Spam-Markierung einer eigentlichen Ham-Mail (genannt: *false positive*) zu vermeiden. Auch während der aktiven Phase muss ein Bayes-Filter weiter trainiert werden. Dadurch sind sie in der Lage, neue „Trends“ in die Statistik aufzunehmen und darauf zu reagieren. Ein gut trainierter und aktuell gehaltener Filter kann erstaunliche Filterergebnisse erzielen [Graham (2003)]. Dies bedarf jedoch einer im Verhältnis zu anderen Spamfilter-Techniken enormen Vorbereitungs- und Wartungsarbeit.

Folgende Illustration zeigt den Algorithmus eines typischen Bayes-Filters [Li und Zhong (2006)]:

- Stufe 1: Training
  - parse jede E-Mail in einzelne Tokens
  - berechne für jedes Token T die Spamwahrscheinlichkeit S wie folgt:
    $S[T] = \frac{C_{spam}(T)}{C_{ham}(T)+C_{spam}(T)}$
  - speichere die jeweiligen Werte in der Datenbank ab
- Stufe 2: Filterung
  - für jede Nachricht M:

    while (M not end) do
    scanne nächstes Token T_i
    frage $S(T_i)$ in der Datenbank ab
    berechne die Spamwahrscheinlichkeit der Nachricht wie folgt:
    $S(M) = \prod_{i=1}^{N} S[T_i]$
    berechne die Hamwahrscheinlichkeit H wie folgt:
    $H(M) = \prod_{i=1}^{N}(1 - S[T_i])$

  - berechne den allgemeinen Spam-Indikator I der Nachricht:
    $I(M) = f(S[M], H[M])$
    f ist eine filterabhängige Funktion, wie z.B. $I[M] = \frac{1+S[M]-H[M]}{2}$

– falls $I[M] > Schwellenwert$
  markiere Nachricht als Spam
  ansonsten
  markiere Nachricht als Ham

Ein Nachteil von Bayes-Filtern ist der verhältnismäßig hohe Ressourcenverbrauch. Herauszustellen sind hier vor allem die CPU-Last bei der Token-Analyse, die Speicherzugriffszeit zur Datenbankabfrage der Spamwahrscheinlichkeiten sowie der Speicherplatzbedarf der gehaltenen Datenbank selbst. Mitarbeiter der University of Georgia entwickelten ein Verfahren, das dieses Problem lösen soll [Li und Zhong (2006)]. Dieses Verfahren greift auf einen Hash-basierten Lookup zur schnelleren Datenbankabfrage sowie auf eine bessere Kompressionsmethode („lossy encoding“) zurück, um die Größe der Datenstruktur zu verkleinern.

Einen gravierenden Nachteil weist die Bayes-Filter-Technik jedoch bezüglich des Datenschutzes auf. Es ist eine Vertrauensfrage, ob ein Anwender eine persönliche Ham-Mail als solche markiert und dem Bayes-Filter zum Training schickt. Man muss sich vor Augen führen, dass bei entsprechender Infrastruktur, private E-Mails in der zentralen Datenbank abgelegt werden und theoretisch durch andere Personen gesichtet werden können. Ein Ansatz, bei dem persönliche Nachrichten so verschlüsselt werden, dass sie zwar für das Training noch geeignet sind, für Menschen aber nicht mehr zu lesen sind, wurde auf der Spam Conference 2006 [Zhong u. a. (2006)] vorgestellt.

Ein weiteres Problem, mit dem zu rechnen ist, ist das Einschlagen des Filters in eine falsche Richtung. Warum dies geschieht, kann nicht wie bei regelbasierenden Filtern nachvollzogen werden. Eine mögliche Ursache könnten die von Spammern vorgenommenen Angriffe auf Bayes-Filter sein. Hierbei wird durch die Benutzung ganz normaler Wörter versucht, die Statistik zu beeinflussen. Zusätzlich eingesetzte Verfahren zur *Rauschunterdrückung* können diesem jedoch entgegnen. Aufgrund der herausgestellten Gefahr, die von schlecht trainierten Filtern ausgeht, sollten Bayes-Filter ebenfalls nur im Verbund mit weiteren Antispam-Mechanismen eingesetzt werden.

3 Design

Nachdem die wesentlichen theoretischen Grundlagen in den vorangegangenen Kapiteln geklärt wurden, widmet sich dieses Kapitel dem Entwurf und der Spezifikation des Tests. Testaufbau und Testablauf werden ebenso beschrieben wie die Testkriterien, anhand derer die Ergebnisse miteinander verglichen werden. Die praktische Umsetzung dieses Konzepts sowie die implementierungsrelevanten Details zu den hier vorgestellten Schritten werden im folgenden Abschnitt vorgestellt.

3.1 Ansatz

Wissenschaftler der University of Waterloo veröffentlichten 2007 die Ergebnisse einer von ihnen durchgeführten Spamfilter-Evaluierung [Cormack und Lynam (2007)]. In dieser verglichen sie verschiedene Konfigurationen von insgesamt sechs verschiedenen Open Source Spamfiltern. Für die Studie zogen sie ein seit über 20 Jahren bestehendes E-Mail-Postfach, das einen regelmäßigen Spam-Eingang verzeichnete, heran. Sie entwickelten ein System, das jede eingehende E-Mail annimmt und anschließend parallel an alle Teststellungen weiterleitet. Auf diese Weise wurde sichergestellt, dass jeder Spamfilter mit genau den gleichen E-Mails konfrontiert wird. Dadurch wurde eine detaillierte und aufschlussreiche Auswertung ermöglicht. Im Schwerpunkt der Studie standen die Bayes-Filter und deren jeweilige Konfiguration. Ein Training der Bayes-Filter wurde durch manuelle Überprüfung jeder Filter-Entscheidung sichergestellt. Leider ist aus dieser Studie nicht ersichtlich, inwiefern die E-Mails nach der Weiterleitung noch als „aktuell“ bezeichnet werden können. Hierzu wäre eine detaillierte Betrachtung des weiterleitenden Systems notwendig.

In der vorliegenden Arbeit wurden die Filtersysteme mit „aktuellem“ Spam konfrontiert. Als aktueller Spam seien hier jene E-Mails bezeichnet, die ohne Zwischenfilterung in Echtzeit auf einen Spamfilter treffen. Ebenfalls erhielten die jeweiligen Testsysteme eine eigene IP-Adresse. Für eine festgelegte Domäne wurden sie im jeweiligen MX-Record[1] eingetragen und konnten so direkt mit anderen MTAs kommunizieren. Da der Kampf gegen Spam – wie im zweiten Kapitel beschrieben – bereits im SMTP-Dialog beginnt, ist auf diese Weise sichergestellt, dass Verfahren zur Absendervalidierung ungehindert zur Anwendung kommen können.

Im Gegensatz dazu würde eine einfache Weiterleitung von Spam aus anderen Postfächern zu einem verfälschtem Ergebnis führen. So wäre es möglich, dass in der Zeitspanne zwischen der ursprünglichen Sendezeit bis zum Eintreffen der E-Mail an einem Spamfilter die Absender-Domäne bereits in einer öffentlichen Blacklist aufgenommen wurde. Auch andere Verfahren zur Absendervalidierung würden bei einer Weiterleitung verfälschte Ergebnisse liefern. Das weiterleitende Postfach existiert schließlich wirklich, und die entsprechende Domäne dürfte in den meisten Fällen auch als vertrauenswürdig eingestuft werden. Auch ein Einsatz ganzer Spam-Archive[2], wie sie oft zum initalen Training von Bayes-Filtern benutzt werden, hätte die gleiche Auswirkung. Dies bekräftigt die Notwendigkeit der Verwendung von aktuellem (realtime) Spam.

3.2 Aufbau und Ablauf

Bevor man überhaupt eine Antispam-Appliance aufstellt, benötigt man notwendigerweise bespammte E-Mail-Konten. Daher ist der erste Schritt, sich Gedanken über die Art und Weise zu machen, wie und warum man Spam erhält.

Schritt 1: Publizieren von E-Mail-Adressen

Paradoxer Weise stellt sich hier nicht die Frage, wie Spammer an die E-Mail-Adressen kommen, sondern wie die Adressen zu den Spammern kommen. Es gäbe sicherlich die Möglichkeit, E-Mail-Postfächer zu eröffnen, und diese dann im Internet auf einschlägigen

[1] Der MX-Record, auch als MX Resource Record oder Mail Exchange Resource Record bezeichnet, ist ein Eintrag im DNS (Domain Name System), der sich auf den Dienst SMTP bezieht. Bevor ein sendender MTA Kontakt mit einem anderen Mailserver aufbauen kann, muss er dessen IP-Adresse kennen. Er fragt diese (also den MX-Eintrag) für die entsprechende Zieldomäne über DNS ab. Danach kann eine direkte SMTP-Sitzung augebaut werden.

[2] Unter http://www.spamlinks.net/filter-archives.htm werden beispielsweise sowohl Spam- als auch Ham-Archive zum Download angeboten.

Seiten der zwielichtigen Zone, bei Online-Gewinnspielen oder auch Foren und Newsgroups zu veröffentlichen. Diese Methode kommt jedoch dem „Anfordern“ von Spam sehr nahe. Um der genauen Definition treu zu bleiben und ausschließlich UCE zu bekommen, also unaufgeforderte kommerzielle E-Mails, wurde ein anderer Weg eingeschlagen. Hierzu wurden zu Beginn dieser Arbeit insgesamt zehn Domänen bei der *DENIC*[3] angemeldet. Pro angemeldeter Domäne wurden daraufhin vier miteinander verlinkte Webseiten veröffentlicht, die insgesamt vier E-Mail-Adressen enthielten. Drei davon sind in der normalen Browseransicht sichtbar, die vierte wurde in den Meta-Informationen, welche lediglich im Quelltext sichtbar sind, publiziert.

Ziel war es, dass Spammer durch sogenanntes *Harvesting* an die Adressen gelangen. Als Harvesting bezeichnet man den Vorgang der Quelltextanalyse von Webseiten mittels eigens dafür entwickelter Programme (genannt: Harvester oder Harvesting-Bots). Diese Programme ähneln dabei den Suchmaschinen-Crawlern, die Webseiten indizieren, mit dem Unterschied, dass hier E-Mail-Adressen „abgeerntet“ werden. Dabei spielt es keine Rolle, ob eine Adresse für Menschen in einem Browser sichtbar ist, oder ob die Adresse nur in den Meta-Informationen zu finden ist. Da der Quelltext analysiert wird, werden auch alle Adressen gefunden. Eine Möglichkeit, sich vor Webseiten-Harvesting zu schützen, wird von Eggendorfer aufgezeigt [Eggendorfer (2005)].

Eine mittlerweile als veraltet geltende präventive Schutzmaßnahme vor Harvestern ist das Einfügen von zusätzlichen Zeichen in die E-Mail-Adresse, welche es zu entfernen gilt, bevor man eine E-Mail an diese Adresse schicken möchte. So hat man die Möglichkeit, seine E-Mail-Adresse beispielsweise in der Form *vorname.name123@domain.de* mit dem Hinweis zu veröffentlichen, dass *123* vorher zu entfernen ist. Im gleichen Sinne ist auch die Verwendung des zusätzlichen Wortes *nospam* innerhalb der E-Mail-Adresse geläufig. Es bot sich an, die Wirkung dieser Maßnahme im Rahmen dieses Tests zu untersuchen. Folgendes E-Mail-Schema wurde für die auf der Webseite veröffentlichten Adressen gewählt:

	Adresse	**auf Webseite publiziert**
1	vorname.name@domain.de	ja
2a	vorname1.name1.nospam@domain.de	ja
2b	vorname1.name1@domain.de	nein
3a	vorname2.name2@nospam.domain.de	ja
3b	vorname2.name2@domain.de	nein
4	v.name@domain.de	ja (Meta)

[3] Die DENIC eG ist die zentrale Registrierungsstelle für alle Domains unterhalb der Top Level Domain .de. Siehe auch unter: http://www.denic.de/de/

Die jeweils erste Adresse auf einer Webseite war stets in der Form *vorname.name@domain.de* ohne weitere Besonderheiten. In der zweiten Adresse wurde das Wort *nospam* in den Namensteil aufgenommen. Obwohl klar sein sollte, dass es dieses E-Mail-Konto so nicht gibt, wurde es trotzdem eingerichtet (2a). Geht auf diese Adresse Spam ein, so ließe sich schlussfolgern, dass das Wort *nospam* schlicht ignoriert wird. Das dazu gehörige „offizielle“ Konto wurde ebenfalls eingerichtet, jedoch nirgens publiziert (2b). Wenn also auf diesem Konto E-Mails eingehen, könnte man daraus schließen, dass Spammer das Wort *nospam* entfernt haben. Die dritte veröffentlichte Adresse hat den *nospam*-Zusatz im Domänenteil der Adresse (3a). Analog exisiteren auch hier beide Postfächer. Die E-Mail-Adressen der Freemail-Accounts von WEB.DE und GMX wurden auf allen Internetseiten aller Domänen ausschließlich im Quelltext veröffentlicht (4).

Je öfter eine Webseite im Internet verlinkt ist, desto eher gelangen Spam-Harvester auf diese Seite. Daher wurden Freunde und Kommilitonen gebeten, die Webseiten für diesen Test von ihren Seiten zu verlinken. Analog erfolgte ein Aufruf in öffentlichen Newsgroups (de.admin.net-abuse.mail, news.admin.net-abuse.email), mit der Bitte, die Seiten zu verlinken. Auch in diversen Foren, Mailinglisten und Usegroups (Google Groups, Yahoo! Groups) konnten die Webseiten leicht veröffentlicht werden. Es wurden jedoch nie direkt E-Mail-Adressen, die für den Test genutzt werden sollten, in irgendeiner anderen Form veröffentlicht.

Nachdem die „Spamversorgung“ sichergestellt war, blieb die Frage nach dem Gegenstück: Um die Filtersysteme auch auf ihre Ham-Erkennung zu testen, wurden Ham-E-Mails benötigt. Dies stellte eine weitaus größere Herausforderung dar, als Spam zu bekommen. Eigene Recherchen sowie persönliche Gespräche auf dem EU Spam Sysmposium 2006 ergaben, dass im Internet keine brauchbaren frei zur Verfügung stehenden Ham-Archive existieren.

Eine denkbare Option wäre das Heranziehen bzw. Kopieren des E-Mail-Verkehrs einer größeren Organisation oder Firma gewesen. Dies wäre jedoch zum einen eine sehr große Vertrauensfrage, und zum anderen bezüglich des Datenschutzes sehr bedenklich gewesen. Schließlich nutzen viele Mitarbeiter ihre dienstlichen E-Mail-Konten auch privat. Es blieb also nichts anderes übrig, als sich geschickt einen E-Mail-Verteiler zu erstellen und manuell E-Mails zu verschicken. Um möglichst viele verschiedene Absender-Adressen zu haben, wurden hierfür nochmals Kommilitonen, Freunde und Bekannte – auch aus anderssprachigen Ländern – mit einbezogen. Ein Vorteil dieser Variante war, dass die Filterergebnisse Ham-Mail für Ham-Mail miteinander verglichen werden konnten.

Schritt 2: Einrichten eines E-Mail-Servers mit Datenbank-Anbindung

Mit dem Zeitpunkt der Veröffentlichung einer E-Mail-Adresse auf der Webseite musste das entsprechende E-Mail-Konto natürlich auch eingerichtet sein. Für diesen Zweck wurde zu Beginn dieser Arbeit ein E-Mail-Server (*Sendmail*) aufgesetzt und als zuständiger MTA für die 10 Domänen konfiguriert.

Um empfangene E-Mails automatisiert analysieren zu können, wurde ebenfalls eine Datenbank (*MySQL*) eingerichtet, welche die Nachrichten direkt vom E-Mail-Server weitergeleitet bekam. So waren in der Auswertungsphase sehr präzise Datenbank-Abfragen möglich.

Schritt 3: Einrichten und Konfigurieren der Appliances

Von der Publizierung einer E-Mail-Adresse bis zur ersten eintreffenden Spam-Mail braucht es eine gewisse Zeit. Daher musste eine „Anlaufzeit“ eingeplant werden, bevor man mit dem eigentlichen Test beginnen konnte. Ein solcher Zeitraum kann nie groß genug sein, denn je länger man wartet, desto mehr Spam kommt an, und desto mehr haben die Filtersysteme zu leisten. Aufgrund der beschränkten Zeit dieser Arbeit wurden nach ca. vier Monaten die Appliances in Betrieb genommen. Nach einer weiteren Woche, die zu Testzwecken verschiedener Konfigurationseinstellungen genutzt wurde, konnte der Test schließlich beginnen.

Schritt 4: Auswertung

Abschließend musste anhand der E-Mails in der Datenbank sowie den statistischen Daten der einzelnen Systeme eine sinnvolle Auswertung vorgenommen werden. Nach welchen Kritierien die Auswertung erfolgt ist, kann dem folgenden Abschnitt entnommen werden.

3.3 Testkriterien

Eine E-Mail, die einen Spamfilter passiert, kann von diesem grundsätzlich als Ham oder Spam klassifiziert werden. Einige Spamfilter bieten die Möglichkeit, im Bereich einer gewissen Unsicherheit die E-Mail als „Spam-verdächtig“ zu markieren. Diese Variante sei jedoch zunächst außer Acht gelassen und kommt gegebenenfalls in einer Einzelfallbetrachtung nochmals zum Tragen. Bleibt man bei den zwei Klassifizierungsmöglichkeiten, so ergeben sich daraus vier verschiedene Filter-Ergebnisse, die man in einem vergleichenden Test heranziehen kann: ein Spamfilter kann in seiner Entscheidung richtig oder falsch liegen. Richtig erkannte Spam-E-Mails bezeichnet man als **true positives**. Das Wort *positive*

bedeutet bei dieser Unterscheidung stets: „als Spam markiert“. Demnach werden falsch zugeordnete Spam-Nachrichten als ***false positives*** bezeichnet. Analog unterscheidet man richtig klassifizierte Ham-Mails als ***true negatives*** und falsch klassifizierte Ham-E-Mails mit **false negatives**. Folgende Übersicht verdeutlicht dies noch einmal.

E-Mail wurde klassifiziert als	Spam	und ist auch	Spam	-> true positive
E-Mail wurde klassifiziert als	Spam	ist aber	Ham	-> false positive
E-Mail wurde klassifiziert als	Ham	und ist auch	Ham	-> true negative
E-Mail wurde klassifiziert als	Ham	ist aber	Spam	-> false negative

Eine *false positive*-Klassifizierung ist von den beiden möglichen Fehleinschätzungen die weitaus schlimmere. Eine versehentlich ins Postfach geratene *false negative*-Nachricht kann schnell gelöscht werden, eine *false positive*-E-Mail hingegen kann unter Umständen für immer verloren gehen.

3.3.1 Normierung

Aufgrund der gewählten Art der Spamgewinnung konnte man nicht beeinflussen, wieviel Spam pro Domäne in die Postfächer gelangte. Die absoluten Zahlen gaben daher nur bedingt Auskunft über die Güte der Spamfilterung. Um die Ergebnisse der Filtersysteme miteinander vergleichen zu können, musste eine Normierung vorgenommen werden. Dies kann beispielsweise durch die Berechnung eines relativen Wertes erreicht werden.

In dieser Arbeit gilt folgende Normierung:

- **Ham-Fehlerrate** F_h**:** Die Anzahl falsch klassifizierter Ham-Mails dividiert durch die Anzahl aller empfangenen Ham-Mails.

$$F_h = \frac{\textit{false positives}}{\textit{false positives} + \textit{true negatives}}$$

- **Ham-Zuverlässigkeit** Z_h**:** Die Differenz aus 1 und der Ham-Fehlerrate.

$$Z_h = 1 - F_h$$

- **Spam-Fehlerrate F_s:** Die Anzahl falsch klassifizierter Spam-Mails dividiert durch die Anzahl aller empfangenen Spam-Mails.

$$F_s = \frac{\textit{false negatives}}{\textit{false negatives} + \textit{true positives}}$$

- **Spam-Zuverlässigkeit Z_s:** Die Differenz aus 1 und der Spam-Fehlerrate.

$$Z_s = 1 - F_s$$

Cormack und Lynam berechneten in ihrer Spamfilter-Evaluierung zusätzlich eine Gesamt-Fehlerrate [Cormack und Lynam (2007)]. Diese soll auch in diesem Test angegeben werden und errechnet sich wie folgt:

$$F_{sh} = \frac{\textit{false positives} + \textit{false negatives}}{\textit{true positives} + \textit{false positives} + \textit{true negatives} + \textit{false negatives}}$$

Anmerkung: Für eine prozentuale Darstellung können die errechneten Ergebnisse jeweils mit der Zahl 100 multipliziert werden.

3.3.2 Fragestellungen

Neben den oben genannten Testkriterien wurden weitere Fragestellungen aufgeworfen, auf die der Test Antworten geben sollte:

- Wie groß ist der Zeitraum zwischen der Publizierung der E-Mail-Adressen und dem Eingang der ersten Spam-Mail?
- Werden E-Mail-Adressen, die auf derselben HTML-Seite zu finden sind, gleich stark bespammt?
- Werden E-Mail-Adressen, die nur in den Meta-Informationen der Webseite zu finden sind, weniger stark bespammt?
- Welchen Einfluss haben „nospam“-Zusätze sowohl im Namen- als auch im Domänenteil der E-Mail-Adresse? Wird dieser Zusatz aus der Adresse gestrichen oder einfach übernommen?
- Welche Unterschiede gibt es zwischen den E-Mail-Providern, den Antispam-Dienstleistern und den Antispam-Appliances bezüglich der Fehlerraten?

- Werden ungefilterte Domänen auf Dauer stärker bespammt als gefilterte Domänen? (Anmerkung: Zwei der zehn Domänen verblieben über den gesamten Testzeitraum ungefiltert.)
- Welche Auffälligkeiten gibt es? Gibt es besonders viel bzw. besonders wenig bespammte Postfächer?
- Wie erklären sich extreme Werte bei den Test-Ergebnissen?
- Wenn eine gute Spam-Zuverlässigkeit vorliegt, liegt dann eine genauso gute Ham-Zuverlässigkeit vor?
- Welche Spamfilter-Mechanismen sind besonders wirkungsvoll?
- Worauf lassen sich gute bzw. schlechte Ergebnisse der verschiedenen Filtersysteme zurückführen?

Anhand der Fehler- und Zuverlässigkeitsraten und den Antworten zu diesem Fragenkatalog sollte abschließend versucht werden, eine Empfehlung über die Zusammenstellung von Spamfilter-Mechanismen für ein den zukünftigen Spam-Wellen gut gerüstetes System abzugeben.

4 Realisierung

Die folgenden Abschnitte befassen sich mit der detaillierten Umsetzung des in Kapitel drei beschriebenen Testkonzepts. Angefangen beim Aufsetzen eines sicheren Web- und E-Mail-Servers bis hin zur Struktur der Datenbank werden hier alle Schritte beschrieben, die nötig sind, um eine Experimentalumgebung für die Testdurchführung zu schaffen.

4.1 Server

Im Zentrum dieses Tests stand der Server, der die Webseiten gehostet hat, die E-Mails entgegen nahm und diese dann in eine Datenbank schrieb. Für den mit *Troubardix* benannten Server wurde ein Rechner gewählt, der über einen Intel Pentium 4 Prozessor mit 2,20 GHz, 512 MB Arbeitsspeicher sowie einer 60 GB großen Festplatte verfügte[1].

Als Betriebssytem wurde *Fedora Core 4*[2] gewählt, ein durch Red Hat gefördertes Linux. Der Kernel (Version 2.6.14.7) wurde handkompiliert und mit *grsecurity*-Patches[3] gehärtet. Auf Modul-Support wurde verzichtet, so dass Rootkits, die ab Kernel-Modul daherkommen, keine Chance hatten.

Da Troubardix im Serverraum des Instituts stand, bot sich für den externen Zugriff SSH an. SSH wurde vorkompiliert aus einem RPM (Version 4.2 Patchlevel 1) installiert. Zur Authentifizierung diente ein 3072-bit starker generierter RSA-Key. Der remote-root-Login wurde aus Sicherheitsgründen deaktiviert.

[1] Wie sich später herausstellte, entsprachen die Systemressourcen den erwarteten Belastungen. Lediglich bei den SQL-Abfragen in der Auswertungsphase wäre mehr Arbeitsspeicher angebracht gewesen. Die Größe der Datenbank bewegte sich in der Größenordnung von einigen Gigabytes, so dass häufig von der Festplatte in den Arbeitsspeicher nachgeladen werden musste. Dies führte zu teilweise relativ lang dauernden Datenbankabfragen.

[2] Fedora Core 4: http://download.fedora.redhat.com/pub/fedora/linux/core/4/i386/iso/

[3] grsecurity-Patches: http://www.grsecurity.org

Um das System zusätzlich zu schützen, wurde eine Firewall eingerichtet. Auf diese Weise konnten einerseits Angriffe abgewehrt und andererseits nicht gewünschter ausgehender Verkehr geblockt werden. Ausgehender E-Mail-Verkehr wurde mit Ausnahme des Institut-eigenen E-Mail-Servers vollständig gesperrt. So war es möglich, dass das System Warn- oder Status-E-Mails an den Administrator senden konnte. Die Sperrung beugte einem möglichen Massenversand von E-Mails, beispielsweise durch Konfigurationsfehler des E-Mail-Servers verursacht, vor. Bösartige Backdoor-Programme, vor denen ein System theoretisch nie hundertprozentig geschützt ist, können durch eine Firewall ebenfalls am Senden gehindert werden. Für diesen Zweck kam *iptables*[4] in der Version 1.3 zum Einsatz.

4.2 Webserver

Um die Webseiten zu hosten, wurde eine Minimal-Konfiguration des Apache HTTP Servers[5], Version 1.3 handkompiliert und installiert. Wie im Entwurf angesprochen, mussten zehn Domänen eingerichtet werden. Um die Übersicht zu wahren, empfahl es sich, eine Art Nummerierungsschema in die Domänennamen zu integrieren. Die Namenwahl erfolgte aufgrund einer spontanen Idee in plattdeutsch:

```
ersdesei.de
zweidesei.de
driddesei.de
vierdesei.de
fuenfdesei.de
sechsdesei.de
siebendesei.de
achdesei.de
neundesei.de
zehndesei.de
```

Durch die Verwendung namensbasierter virtueller Hosts erhielten alle Domänen die gleiche IP-Adresse wie der Server (137.193.63.230). Folgender Ausschnitt zeigt den Eintrag für die Domäne *ersdesei.de* in der Konfigurations-Datei des HTTP-Servers `/httpd/conf/httpd.conf`:

```
NameVirtualHost 137.193.63.230
<VirtualHost 137.193.63.230>
```

[4] iptables Firewall: http://www.iptables.org/projects/iptables/index.html
[5] Apache HTTP Server: http://httpd.apache.org/

```
    ServerAdmin web@ersdesei.de
    ServerName www.ersdesei.de
    ServerAlias ersdesei.de
    DocumentRoot /httpd/htdocs/site1
    CustomLog /httpd/logs/site1_access_log combined
    ErrorLog /httpd/logs/site1_error_log
</VirtualHost>
```

Um den Webserver von außerhalb erreichbar zu machen, mussten noch entsprechende Firewall-Einträge veranlasst werden. Selbiges gilt auch für die IP-Adressen der Appliances.

4.3 E-Mail-Server

Ebenfalls aus einem RPM-Paket heraus wurde der Open Source MTA *Sendmail*[6] installiert. Als finaler MTA war er dafür verantwortlich, die E-Mails per SMTP anzunehmen und in die Datenbank zu schreiben. Der MTA wurde so konfiguriert, dass er alle E-Mails annahm, sofern sie an eine der lokalen Domänen adressiert waren, also auch an nicht explizit eingerichtete Nutzer.

In den MX-Einträgen der Domänen standen bereits zu Beginn der Anlaufphase die IPs, die später dann den Appliances zugeordnet wurden. Bis zum Tag des Testbeginns nahm Troubardix die E-Mails der Domänen entgegen. Dies wurde durch entsprechende virtuelle IP-Zuweisungen umgesetzt.

Als die Appliances schließlich mit den vorgesehenen IP-Adressen online gingen, wurden vorher lediglich die virtuellen Einträge auf Troubardix entfernt. Durch diesen Trick mussten die MX-Einträge der Domänen im DNS nicht verändert werden, was durch DNS-Caching bedingt Verzögerungen mit sich gebracht hätte. Die Appliances konnten so von der ersten Minute an die E-Mails der zugewiesenen Domänen scannen.

Schließlich sorgten Weiterleitungseinträge dafür, dass jede E-Mail nach dem Scannen auch in der Datenbank auf Troubardix landete.

Ein Auszug aus der Sendmail-Konfigurations-Datei `/etc/mail/virtusertable` zeigt die Zuordnung von virtuellen User-Namen zu den E-Mail-Adressen für die erste Domäne :

[6] Sendmail MTA: http://www.sendmail.org/

```
#ersdesei.de
michael.mayer@ersdesei.de             domain1_user1
michael.maier.nospam@ersdesei.de      domain1_nospam_user2
michael.maier@ersdesei.de             domain1_user2
michael.mayr@nospam.ersdesei.de       domain1_user3_nospam
michael.mayr@ersdesei.de              domain1_user3
guestbook@ersdesei.de                 domain1_guestbook
m.maier@ersdesei.de                   domain1_user4
web@ersdesei.de                       domain1_web

# RFC2142 required users
info@ersdesei.de                domain1_info
marketing@ersdesei.de           domain1_marketing
sales@ersdesei.de               domain1_sales
support@ersdesei.de             domain1_support
abuse@ersdesei.de               domain1_abuse
noc@ersdesei.de                 domain1_noc
security@ersdesei.de            domain1_security
postmaster@ersdesei.de          domain1_postmaster
hostmaster@ersdesei.de          domain1_hostmaster
usenet@ersdesei.de              domain1_usenet
news@ersdesei.de                domain1_news
webmaster@ersdesei.de           domain1_webmaster
www@ersdesei.de                 domain1_www
uucp@ersdesei.de                domain1_uucp
ftp@ersdesei.de                 domain1_ftp

# all other users / Catch-All
@ersdesei.de                    domain1_other
@nospam.ersdesei.de             domain1_other_nospam
```

Neben den im Kapitel drei vorgestellten E-Mail-Adressen wurden weitere RFC2142-konforme Adressen definiert, die jede Domäne im Internet haben sollte [Crocker (1997)]. Alle nicht aufgeführten Adressen wurden dem `domain1_other`- bzw. dem `domain1_other_nospam`-User zugeordnet.

In der Sendmail-Konfigurations-Datei `/etc/mail/aliases` wurden lokale Weiterleitungen eingetragen. Die Weiterleitungen bestimmten, was mit einer E-Mail, die auf einem virtuellen User-Konto einging, geschehen sollte. Hier wurde festgelegt, dass jede E-Mail via Standardeingabe (STDIN) in ein PERL-Skript weiterzuleiten war. Das Skript teilte die Nachricht in Header und Body auf und sorgte dann für den Datenbankeintrag. Für jede eingehende E-Mail wurde das mit `count.pl` benannte Skript wie folgt aufgerufen:

Auszug aus `/etc/mail/aliases`:

```
#ersdesei.de
domain1_user1:          "|/etc/smrsh/count.pl user1@domain1"
domain1_nospam_user2:   "|/etc/smrsh/count.pl nospam_user2@domain1"
domain1_user2:          "|/etc/smrsh/count.pl user2@domain1"
domain1_user3_nospam:   "|/etc/smrsh/count.pl user3_nospam@domain1"
domain1_user3:          "|/etc/smrsh/count.pl user3@domain1"
domain1_user4:          "|/etc/smrsh/count.pl user4@domain1"
domain1_web:            "|/etc/smrsh/count.pl web@domain1"
```

```
domain1_guestbook:        "|/etc/smrsh/count.pl guestbook@domain1"

domain1_info:             "|/etc/smrsh/count.pl info@domain1"
domain1_marketing:        "|/etc/smrsh/count.pl marketing@domain1"
domain1_sales:            "|/etc/smrsh/count.pl sales@domain1"
domain1_support:          "|/etc/smrsh/count.pl support@domain1"
domain1_abuse:            "|/etc/smrsh/count.pl abuse@domain1"
domain1_noc:              "|/etc/smrsh/count.pl noc@domain1"
domain1_security:         "|/etc/smrsh/count.pl security@domain1"
domain1_postmaster:       "|/etc/smrsh/count.pl postmaster@domain1"
domain1_hostmaster:       "|/etc/smrsh/count.pl hostmaster@domain1"
domain1_usenet:           "|/etc/smrsh/count.pl usenet@domain1"
domain1_news:             "|/etc/smrsh/count.pl news@domain1"
domain1_webmaster:        "|/etc/smrsh/count.pl webmaster@domain1"
domain1_www:              "|/etc/smrsh/count.pl www@domain1"
domain1_uucp:             "|/etc/smrsh/count.pl uucp@domain1"
domain1_ftp:              "|/etc/smrsh/count.pl ftp@domain1"

domain1_other:            "|/etc/smrsh/count.pl undefined_user@domain1"
domain1_other_nospam:     "|/etc/smrsh/count.pl undefined_user@nospam.domain1"
```

Das PERL-Skript `count.pl` hatte zunächst die Aufgabe, eine E-Mail, die aus Header und Body besteht, so zu trennen, dass sie separat in die Datenbank eingetragen werden konnte. Für diesen Zweck wurde die reservierten Variable `$RS` genutzt, die für den Record-Seperator (RS) steht. Dieser wird beim Einlesen eines Absatzes verwendet. Durch die Zuweisung '`$RS = '';`' wird der Record-Seperator so eingestellt, dass ein Absatz bis zur nächsten Freizeile eingelesen wird. Gemäß dem SMTP-Standard in RFC2822 [Resnick (2001) werden Header und Body durch eine Freizeile voneinander getrennt. Wenn man nun die Standardeingabe (`STDIN`) auslesen lässt, unterbricht der RS das Einlesen beim Auffinden der ersten Freizeile, also dann, wenn der Header zu Ende ist. Der Header wird anschließend der Variablen `$header` zugewiesen. Die folgende While-Schleife liest die weiteren Absätze des Bodys solange ein, bis STDIN leer ist. Nun hat man den gesamten Body in der Variablen `$body`. Die zweite Aufgabe des Skripts besteht darin, den User-Namen, welcher dem Skript beim Aufruf mit übergeben wurde (`$ARGV[0]`), sowie `$header` und `$body` in die Datenbank einzutragen. Hierzu mussten zusätzlich Benutzername, Passwort und der Name der Datenbank spezifiziert werden.

Quellcode des erstellten PERL-Skripts `count.pl`:

```
#!/usr/bin/perl -w

use DBI;
use English;

# zum Login in die Datenbank benötigt:
$dbhost = 'localhost';
$dbname = 'spamcount';
$dbuser = 'spamtest';
```

```
$dbpass = 'password';

# STDIN (Standardeingabe) enthält gesamte E-Mail

if ($#ARGV == 0)
  {
    # alten RS merken:
    $oldRS = $RS;
    # RS auf 'leere Zeile' (Absatzmodus) setzen:
    $RS = '';
    # einlesen des ersten Absatzes(=Header):
    $header = <STDIN>;

    $body = '';
    # Rest der E-Mail in $body schreiben:
    while ( ! eof(STDIN) )
      {
        $body .= <STDIN>;
      }

    # RS zurücksetzen:
    $RS = $oldRS;

    # zur Datenbank verbinden:
    $dbc = DBI->connect("DBI:mysql:host=$dbhost:database=$dbname",
                        $dbuser, $dbpass) or
                        exit (0);
    # Usernamen, Header und Body in Datenbank schreiben:
    $dbc->do("INSERT INTO mails (an,header,body) VALUES (".
             $dbc->quote($ARGV[0]).",".
             $dbc->quote($header).",".
             $dbc->quote($body).");"
            );

    $dbc->disconnect;
  }
```

Zusammenfassend zeigt die folgende Übersicht die IP-Adressen der für die Domänen eingetragenen Mail-Server (MX-Records). Unter der MX-IP war dann die jeweils zugewiesene Appliance bzw. einer der beiden gewählten ASPs erreichbar.

Domänenname	Hostname	MX-Eintrag	MX-IP
ersdesei.de	www.ersdesei.de	mail.ersdesei.de	137.193.63.231
zweidesei.de	www.zweidesei.de	mail.zweidesei.de	137.193.63.232
driddesei.de	www.driddesei.de	mail.driddesei.de	137.193.63.233
vierdesei.de	www.vierdesei.de	mail.vierdesei.de	137.193.63.234
fuenfdesei.de	www.fuenfdesei.de	mail.fuenfdesei.de	137.193.63.235
sechsdesei.de	www.sechsdesei.de	mail.sechsdesei.de	137.193.63.236
siebendesei.de	www.siebendesei.de	MX von eXpurgate	externe IP
achdesei.de	www.achdesei.de	MX von SpamStopsHere	externe IP
neundesei.de	www.neundesei.de	mail.neundesei.de	137.193.63.239
zehndesei.de	www.zehndesei.de	mail.zehndesei.de	137.193.63.230

4.4 Datenbank

Wie bereits erwähnt, sollte jede empfangene E-Mail in einer Datenbank gespeichert werden. Dazu wurde die Open Source Datenbank MySQL[7] aus einem RPM installiert. Anschließend wurde eine Datenbank namens `spamcount` aufgesetzt. Eine mit `mails` benannte Tabelle wurde mit folgenden Feldern und Attributen versehen:

```
+--------+-------------------+----+-----+-----------------+--------------+
| Field  |Type               |Null| Key |Default          |Extra         |
+--------+-------------------+----+-----+-----------------+--------------+
| id     |bigint(20) unsigned|    | PRI |NULL             |auto_increment|
| an     |varchar(255)       |    |     |                 |              |
| header |longtext           |    |     |                 |              |
| body   |longtext           |    |     |                 |              |
| zeit   |timestamp          |YES |     |CURRENT_TIMESTAMP|              |
+--------+-------------------+----+-----+-----------------+--------------+
```

Ein Eintrag wurde ausschließlich durch das PERL-Skript veranlasst. Aus Sicherheitsgründen wurde dem dort spezifizierten User jedoch lediglich INSERT-Rechte gegeben. So wurden bereits eingetragene Daten vor Verändern oder Löschen geschützt.

Jede eingehende E-Mail erhielt als Primärschlüssel eine automatisch inkrementierte ID. Im `an`-Feld steht der Benutzername, der beim Aufruf des PERL-Skriptes mit übergeben wurde. Header und Body wurden entsprechend der vorgenommenen Trennung einzeln abgelegt. Zusätzlich erhielt jede E-Mail noch einen Zeitstempel, welcher die aktuelle Systemzeit beinhaltete.

[7] MySQL: http://www.mysql.de/

4.5 Spamfilter-Systeme

In diesem Abschnitt werden alle Systeme vorgestellt, die für den Test herangezogen wurden.

Folgende fünf Antispam-Appliances nahmen am Test teil:

- Symantec Mail Security 8260
- McAfee Secure Content Management-Appliance 3200
- IronPort C10 Email Security Appliance
- CanIT AntiSpam Appliance
- iKu SPONTS-Appliance

Zusätzlich dazu wurden zwei ASPs[8] sowie zwei Webmail-Provider getestet:

- eXpurgate
- SpamStopsHere
- WEB.DE
- GMX

Für den Zweck einer besseren Vergleichbarkeit wurden Appliances auf einem Leistungs-Level angefordert. Die Anbieter wurden gebeten, Geräte für die Größenordnung von 500 bis 1000 Nutzern zur Verfügung zu stellen. Leider war es den Herstellern teilweise nicht möglich, ein solches Gerät zu liefern. Symantecs Appliance lag beispielsweise weit über dem gewünschten Level. McAfee schickte indes statt dem Secure Messaging Gateway gleich ein komplettes Secure Internet Gateway. Für die zu testende Filterqualität spielt es jedoch keine Rolle, ob eine Appliance nun ein oder zwei Prozessoren hat.

Einige der gelieferten Systeme boten einen weitaus größeren Funktionsumfang als reine Spamfilterung. McAfee stellte beispielsweise einen „All-in-One-Scanner“ (siehe Produkthandbuch) bereit, der neben SMTP auch alle anderen gängigen Internet-Protokolle filtern konnte. Fast jede Appliance ließ sich auch für eine E-Mail-Content-Filterung konfigurieren, welche solche E-Mails in Quarantäne stellen konnte, die zwar kein Spam, aber dennoch aus diversen Gründen unerwünscht waren. Alle Funktionen, die nicht direkt mit der Spamfilterung zusammenhingen, wurden in diesem Test außer Acht gelassen.

[8] ASP steht für „Application Service Provider“ (zu deutsch: Anwendungsdienstleister). In diesem Fall handelt es sich um Unternehmen, die eine Spamfilterung über das Internet anbieten.

4.5.1 Symantec Mail Security 8260

Symantec stellte die „Mail Security 8260“-Appliance zur Verfügung. Sie war für den Einsatz in Unternehmen ab einer Größe von 1000 Nutzern konzipiert. Das Maximum liegt laut Hersteller bei ca. 10.000 Nutzern. Für eine größere Kapazität bietet Symantec die Möglichkeit, mehrere Appliances im Verbund zu betreiben. Dazu wird eine Appliance als Controlcenter eingerichtet und alle weiteren als Scanner. Hat man nur ein Gerät, so fungiert dieses als Controlcenter und Scanner.

Platziert wird die Appliance am Internet-Gateway, unmittelbar hinter der Firewall und vor dem eigentlichen Mailserver. Dort aufgestellt, scannt sie sämtlichen ein- und ausgehenden SMTP-Verkehr. Ein Einsatz als alleiniger Mailserver (also ohne einen dahinterliegenden MTA) ist nicht möglich. Optional ist es jedoch möglich, dass jeder Nutzer auf seinen Quarantäne-Bereich zugreifen kann, um E-Mails zu klassifizieren, die unter Spam- oder Viren-Verdacht stehen.

Abbildung 4.1: Symantec Mail Security 8260

Die 8260 wurde vorinstalliert ausgeliefert: Neben einem Red Hat Enterprise 3.0-Betriebssystem waren auch *„Symantec Antivirus“* sowie *„Symantec Brightmail Antispam“* bereits vorinstalliert. Zudem kam ein gehärteter Postfix-basierter MTA zum Einsatz.

Die Einrichtung der Appliance erfolgte mit der Unterstützung eines Assistenten. Hierbei konnte gewählt werden, ob eingehender und/oder ausgehender E-Mail-Verkehr gescannt werden soll. Um beides zu scannen, wird eine zweite IP-Adresse benötigt. Diese kann entweder in Form eines virtuellen Interfaces oder durch die Nutzung des zweiten Ethernet-Ports erfolgen. Für diesen Test wurde lediglich die eingehende E-Mail-Filterung aktiviert.

Symantecs Appliance bietet einen sogenannten „Traffic-Shaping“-Mechanismus auf TCP-Ebene, welcher es erlaubt, die Verbindungsgeschwindigkeit zu bremsen. Dadurch wird ein potenzieller spamversendender Mailserver, für den ein schnelles, massenhaftes Verschicken von E-Mails wichtig ist, in seiner Ausführung gebremst. Im Rahmen einer „E-Mail-Firewall“ können zusätzlich Schutzmechanismen gegen Spam-, Virus- und Directory-

Harvest-Attacken[9] aktiviert werden. Um stets auf dem aktuellen Stand zu bleiben, wird die 8260 regelmäßig mit Updates versorgt. Symantec verfügt über ein weltweites Netzwerk names „BLOC“ (Brightmail Logistics Operations Center), welches neue Spam-Wellen schnell erkennen soll. Unmittelbar nach dem Erkennen einer neuen Spam-Welle entwickelt man schnellstmöglich ein Spamfilter-Update, und stellt es als Download zur Verfügung. In der Tat erfolgte seitens Brightmail alle zehn Minuten eine Update-Anfrage. Inwiefern tatsächlich neue Updates installiert wurden, ließ sich aufgrund einer fehlenden Versionsnummern-Anzeige nicht nachvollziehen.

In der Dokumentation unerwähnt bleibt ein erweitertes Konfigurations-Menü. Aus dem „Settings“-Menü gelangt man mit der Tastenkombination `Shift+A` in ein Menü, wo weitere Einstellungen möglich sind. Hier kann unter anderem der SMTP-Willkommens-Text des MTAs verändert werden. Weiterhin kann zwischen verschiedenen Spam- und Virenfiltern (siehe Abb. 4.2) gewählt werden.

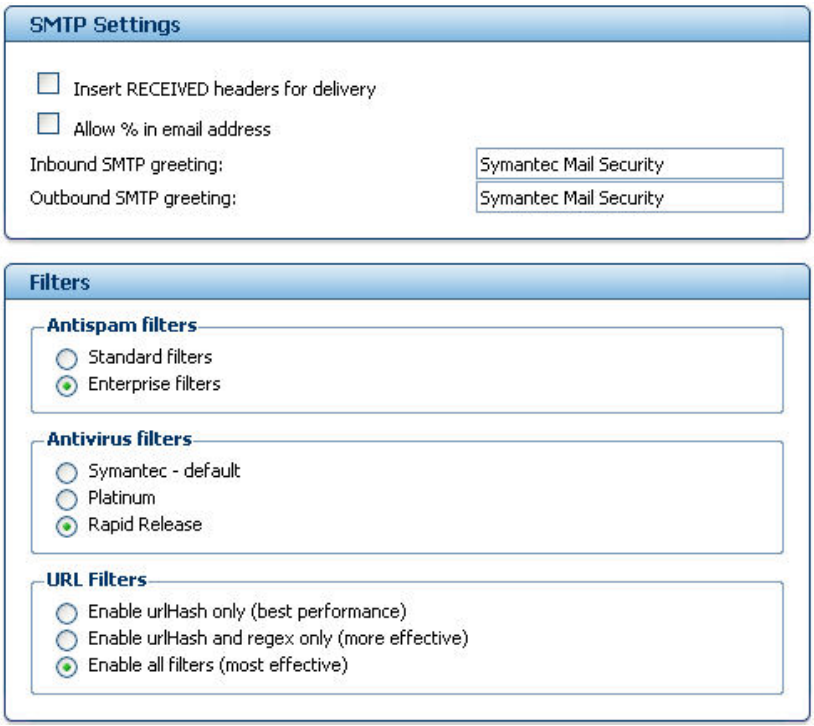


Abbildung 4.2: Symantec Mail Security 8260 - Advanced Settings
Mittels `Shift+A` gelangt man in ein verstecktes Konfigurations-Menü.

[9] Unter einer Directory-Harvest-Attacke versteht man das Ausprobieren von zufällig gewählten E-Mail-Adressen. Kommt daraufhin keine Unzustellbar-Nachricht zurück, so wird die entsprechende E-Mail-Adresse übernommen und kann fortan bespammt werden.

E-Mail-Filterung

Wie oben bereits erwähnt, setzt Symantec bei der Spamfilterung auf *„Brightmail Antispam“*, ein Programm, welches laut Herstellerangaben eine Spam-Erkennungsrate von 95 Prozent und eine *false positive*-Rate von weniger als eine pro eine Million Nachrichten haben soll [Symantec (Deutschland) GmbH (2008)].

Die Appliance verwendet folgende Spamfilter-Mechanismen:

- Sender Policy Framework (SPF)
- Sender Reputation Service (White- und Blacklists, IP-Sperrlisten)
- lokale White- und Blacklists
- URL Filter (URL-White- und Blacklists)
- heuristische Mustererkennung
- signatur- und hash-basierte Erkennung
- sprachbasierte Filterung
- Caller-ID

Einige Verfahren lassen sich optional aktivieren (z.B. SPF und Sprachfilterung). Andere Verfahren wiederum können manuell konfiguriert werden (z.B. White- und Blacklists inklusive der (Gegen-)Maßnahmen und Schlüsselwortfilter).

Jeder einzelne Filter liefert einen Wert zurück, der die Spam-Wahrscheinlichkeit angibt. Die Werte aller Filter gehen mit einer vorgegebenen nicht veränderbaren Gewichtung in die Gesamt-Spam-Wahrscheinlichkeit ein. Anhand eines einstellbaren Schwellenwertes kann man jedoch den Toleranzbereich bestimmen, ab welcher Punktezahl eine Spam-Klassifizierung bzw. ein Spam-Verdacht vorgenommen werden soll.

Um individuelle Filtereinstellungen zu treffen, unterstützt Brightmail Gruppenrichtlinien. Diese können aus einem unternehmenseigenen LDAP-Server[10] übernommen und regelmäßig synchronisiert[11] werden. Dadurch kann der Administrator die Filterstärke für die einzelnen Nutzergruppen unterschiedlich konfigurieren. Vor gleichem Hintergrund werden auch die E-Mail-Sender in Gruppen aufgeteilt. Abbildung 4.3 zeigt die Auswahlmöglichkeiten für die einzelnen Sender-Gruppen:

[10] LDAP steht für „Lightweight Directory Access Protocol“. Es erlaubt die Abfrage und die Modifikation von Informationen eines Verzeichnisdienstes. Die aktuelle Version ist in RFC 4511 (http://tools.ietf.org/html/rfc4511) spezifiziert.

[11] Die LDAP-Synchronisierung erfolgt hierbei ausschließlich vom LDAP-Server in Richtung Appliance.

Sender Groups	Enabled	Action
Blocked Senders (Domain-based) Senders from whom you never want to receive email	✓	Quarantine the message
Blocked Senders (IP-based) IP addresses from which you never want to receive email	✓	Delete the message
Blocked Senders (Third Party Services) Third party lists of IP addresses from which virtually all outgoing email is spam	✓	Reject SMTP Connection
Allowed Senders (Domain-based) Senders from whom you always want to receive email	✓	Deliver message normally
Allowed Senders (IP-based) IP addresses from which you always want to receive email	✓	Deliver message normally
Allowed Senders (Third Party Services) Third party lists of IP addresses from which virtually no outgoing email is spam	✓	Deliver message normally
Open Proxy Senders IP addresses that are open proxies used by spammers	✓	Reject SMTP Connection
Safe Senders IP addresses from which virtually no outgoing email is spam	✓	Deliver message normally
Suspected Spammers IP addresses from which virtually all outgoing email is spam	✓	Defer SMTP Connection

Abbildung 4.3: Symantec Mail Security 8260 - Sender Groups
Für jede Sender-Gruppe lässt sich eine individuelle Aktion definieren.

Für Spam-verdächtige E-Mails stehen mehrere Aktions-Optionen zur Verfügung: unverändert weiterleiten, in den Quarantäne-Bereich verschieben, in den Spam-Ordner des Nutzers verschieben, als Spam oder Spam-verdächtig markieren und weiterleiten oder löschen der E-Mail.

4.5.2 McAfee Secure Content Management-Appliance 3200

McAfee stellte sein „Secure Internet Gateway“ (SIG) in der Version 4.2 zur Verfügung, ein System aus der Familie der „Secure Content Management Appliances“ (SCM). Die Appliance wurde mit einem Red Hat Linux sowie diverser weiterer Software vorinstalliert ausgeliefert.

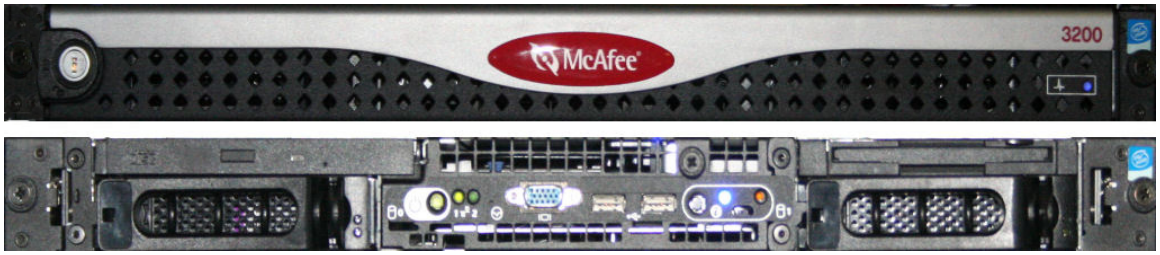


Abbildung 4.4: McAfee SCM 3200

Die SCM 3200 ist gemäß Hersteller für Netzwerke konzipiert, die 1000 Knoten nicht überschreiten. Installiert werden kann die Appliance am Internet-Gateway, also an den Eintrittspunkten des Netzwerkes. Jedoch sollte sie hinter einer Firewall platziert werden.

Bevor man die ersten Einstellungen vornimmt, muss man sich für einen von drei möglichen Betriebsmodi entscheiden: „Expliziter Proxy“, „Transparente Bridge“ oder „Transparenter Router“. Man muss wählen, ob der Netzwerkverkehr transparent gescannt werden soll, oder ob die beteiligten Systeme so umkonfiguriert werden, dass sie sämtliche zu filternde IP-Pakete explizit an die Appliance senden. Weiterhin muss entschieden werden, ob zwei Subnetze miteinander verbunden werden sollen (Bridge), oder ob eventuell ein Router für zwei Subnetze benötigt wird. Für diesen Test wurde – wie bei den anderen Appliances auch – der „Explizite Proxy“-Modus gewählt, d.h., dass die Appliance die IP des im DNS bereits eingetragenen Mailservers bekam.

Die Einrichtung der Appliance wird durch einen Assistenten begleitet, welcher sich nach dem ersten Login automatisch öffnet. Es folgen die üblichen Netzwerkeinstellungen sowie die Wahl des Betriebsmodus. Ebenso werden die Protokolle gewählt, für die sich die Appliance zuständig fühlen soll. Abschließend kann noch das Kennwort geändert und, wenn erwünscht, der SSH-Zugriff aktiviert werden.

E-Mail-Filterung

Voraussetzung zur E-Mail-Filterung ist die einmalige Aktivierung des SMTP-Protokolls und der Software „McAfee SpamKiller“. McAfee setzt bei der Behandlung von SMTP-Verkehr, wie auch bei den anderen Protokollen, auf Richtlinien. Hier lassen sich eine Reihe von Regeln definieren bzw. aktivieren, angefangen bei den lokalen White- und Blacklists bis hin zur Abwehr von Directory-Harvest- und DoS-Attacken.

SpamKiller verwendet neben regelbasierten Filtermethoden auch das Anti-Spam-Modul „SpamAssassin“[12]. Folgende Antispam-Mechanismen werden weiterhin verwendet:

- Integritätsanalyse (Absenderauthentifikation)
- IP-Reputationsfilterung (durch IP-Sperrlisten)
- heuristische Erkennung
- Inhaltsfilter (Schlüsselwortfilter, URL-Filter)
- regelbasierte White- und Blacklists (öffentlich und lokal)
- trainierbarer Bayes-Filter (in SpamAssassin integriert)

Anhand von Regeln kann der Administrator Faktoren (Schwellenwerte) und dazugehörige Aktionen definieren. So kann er beispielsweise einen Gesamtschwellenwert (siehe Abb. 4.5) festlegen, ab wann eine E-Mail als Spam markiert werden soll. Weiterhin kann er für jede Anti-Spam-Regel die Gewichtung bezüglich des Gesamtschwellenwertes bestimmen (siehe Abb. 4.6). Es können auch negative Werte eingetragen werden. Steht ein Absender zum Beispiel auf einer Whitelist, dann kann ein sehr hoher Negativ-Wert alle übrigen Spamfaktoren außer Kraft setzen und dadurch eine Spammarkierung verhindern.

Sollte eine E-Mail aufgrund einer Spam-Markierung im Quarantänebereich landen, so ist es jedem Benutzer möglich, sich in diesen einzuloggen und auf die entsprechende E-Mail zuzugreifen.

[12] SpamAssasin: http://spamassassin.apache.org

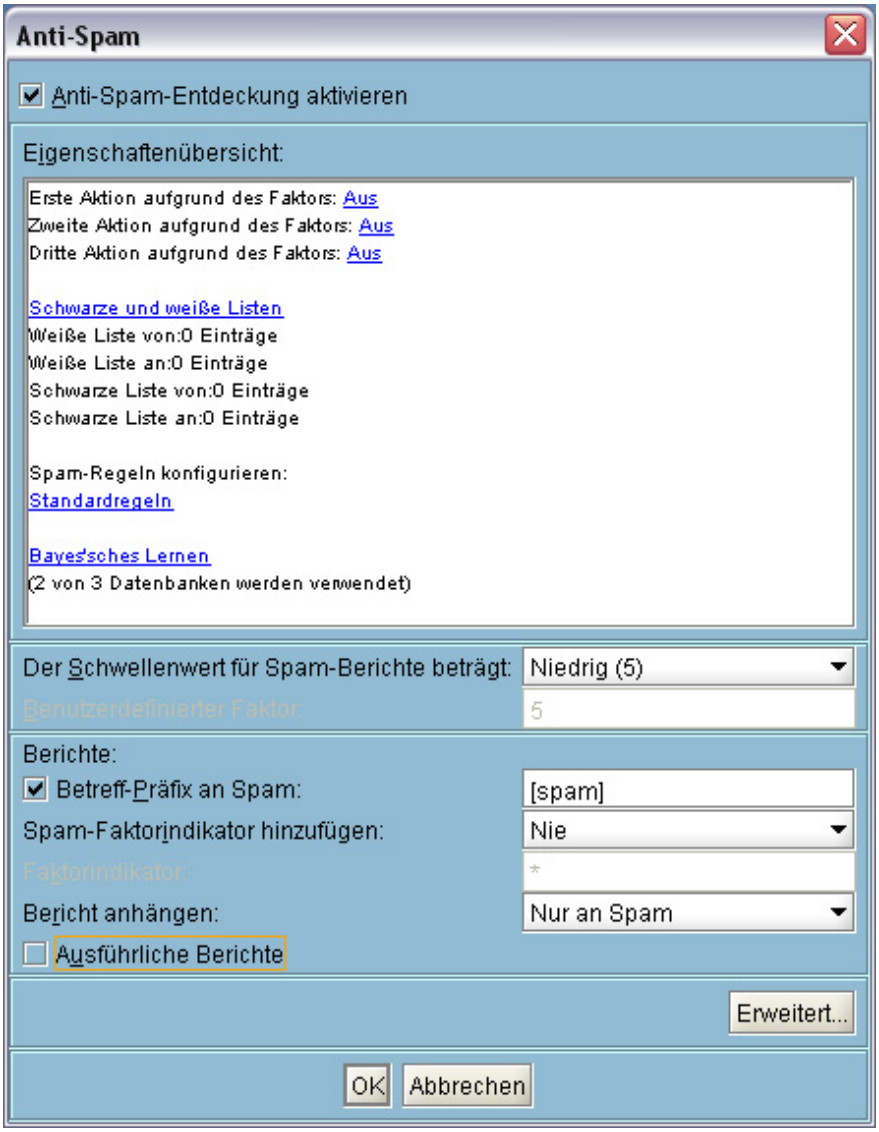


Abbildung 4.5: McAfee SCM 3200 – Anti-Spam-Einstellungen

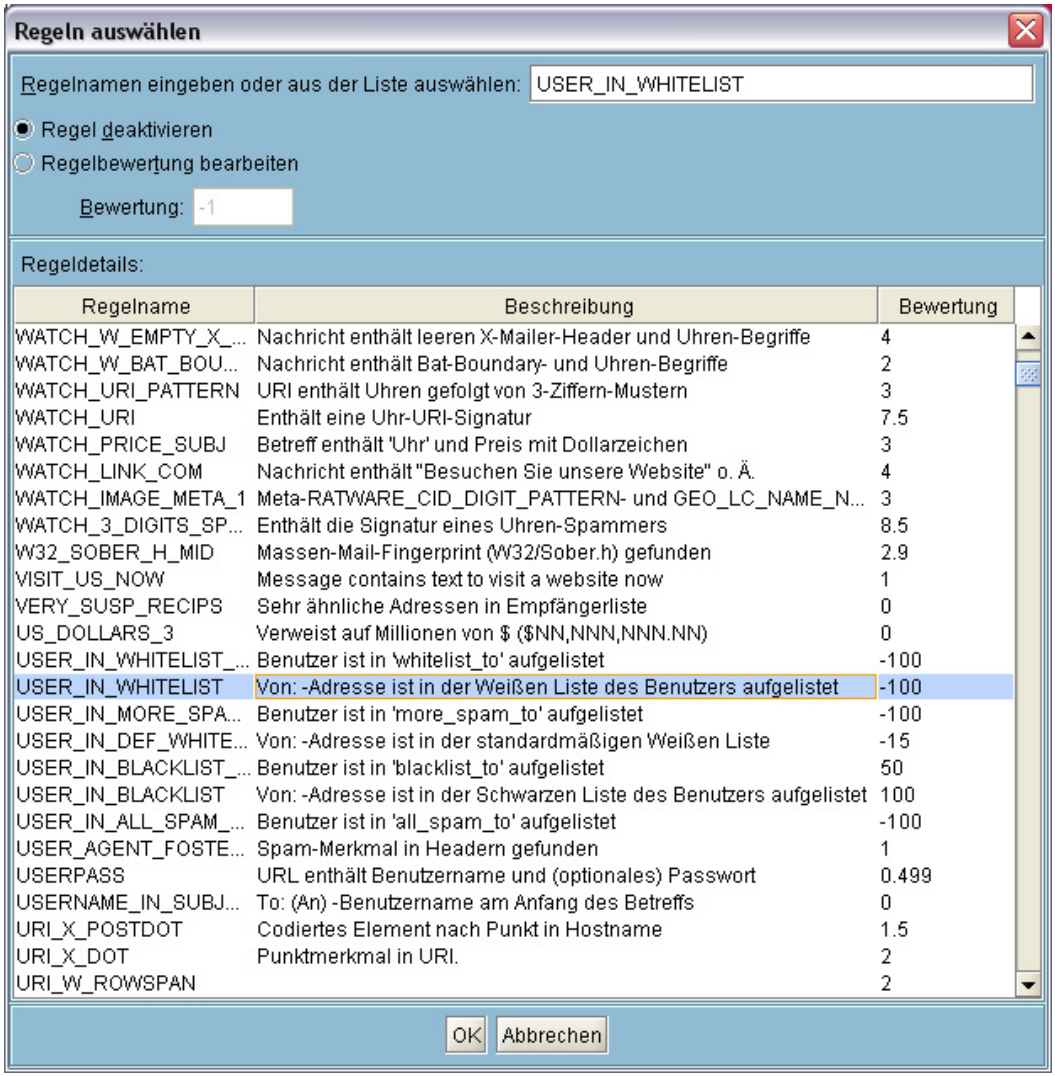


Abbildung 4.6: McAfee SCM 3200 – Antispam-Regeln

4.5.3 IronPort C10 Email Security Appliance

Die Firma IronPort stellte seine „C10 Email Security Appliance“ für den Test zur Verfügung. Konzipiert für kleine und mittelständische Unternehmen kann sie für bis zu 1000 Anwender eingesetzt werden. Mit „AsyncOS“ setzt man ein eigens für den in der Regel asynchronen E-Mail-Verkehr konzipiertes FreeBSD-basiertes Betriebssystem ein, welches eingehenden SMTP-Verkehr bis zu 10-mal effizienter abarbeiten soll als normale Unix-Systeme [IronPort Systems, Inc. (2008a)]. Mittels eines stapellosen Threading-Modells will man zudem über 10.000 gleichzeitige Verbindungen unterstützen. Auf eine Shell wurde hierbei gänzlich verzichtet. Dies begründet auch den durch eine Metallblende „verplombten“ Monitoranschluss an der Rückseite des Gehäuses.

Abbildung 4.7: IronPort C10
[Bildquelle: IronPort Systems, Inc.]

Installiert wird die Appliance genauso wie die der anderen Hersteller: Sie ist direkt hinter der Firewall platziert und nimmt als erster MTA die eingehenden E-Mails an. Die C10 lässt sich entweder als Bridge oder als expliziter Proxy einsetzen. Für erstere Variante stehen zwei Gigabit-LAN-Ports zur Verfügung. Wie bei allen anderen Appliances auch wurde jedoch die zweite Variante gewählt. Mit dem Netzwerk verbunden kann man nach dem Hochfahren des Servers auf das Webinterface zugreifen und die Erstkonfiguration mittels eines Assistenten durchführen. Neben den Netzwerkeinstellungen kann man über sogenannte „Listener“ einstellen, auf welchen Ports und für welche Domänen die C10 eingehende Pakete verarbeiten soll. Nachdem man die Weiterleitung auf den eigentlichen Mailserver konfiguriert hat, kann man schließlich noch die Einstellungen des Spamfilters vornehmen.

E-Mail-Filterung

IronPort verfügt über ein weltweites Netzwerk namens „SenderBase“. SenderBase ist IronPort zufolge das weltweit größte System zur Aufzeichnung und Analyse des aktuellen E-Mail-Verkehrs [IronPort Systems, Inc. (2008b)]. Auf Basis dieser Datenbank werden Upda-

tes entworfen, die schnellstmöglich die Systeme vor den neuen Gefahrenstufen im Internet schützen sollen.

Als Spamschutz ließen sich wahlweise entweder das IronPort-eigene Antispam-Modul oder Symantecs Brightmail aktivieren. Da Brightmail im Rahmen dieses Tests schon in der Symantec-Appliance zum Einsatz kam, wurde hier IronPorts Anti-Spam aktviert. Folgende Antispam-Mechanismen waren im Einsatz:

- Reputationsfilter (Reputation Filtering) durch IP-Sperrlisten
- Context Adaptive Scanning (Kontextbasiertes Scannen)
- Web-Reputation-Technologie (URL-Filterung)
- White- und Blacklists
- SPF
- Bildfilter
- Hash-basierte Muster-Erkennung

Die Einstellungen des Spamschutzes erfolgt in der C10 ebenfalls über Richtlinien. Der Administrator kann detailliert definieren, bei welcher Punktzahl welche Aktion stattfinden soll. Auch die einzelnen Schwellenwerte lassen sich für jeden konfigurierten Listener manuell festlegen. In Abbildung 4.8 sieht man, welche Auswirkungen die Reputations-Punktezahl, die von der SenderBase für jeden Absender übermittelt wird, auf die Einteilung der Listen hat. Wenn beispielsweise eine Absender-IP bereits häufig durch Spamversand aufgefallen ist, wird dieser IP von der SenderBase eine negative Punktezahl zugeordnet. Durch regelmäßiges Abfragen der SenderBase erfährt das System diese Punktezahl und ordnet den Absender in die Blacklist ein. Falls der Administrator es eingestellt hat, wird die E-Mail bereits im SMTP-Dialog abgewiesen. Für alle Regeln bietet IronPort Default-Werte an, so dass ein manuelles Konfigurieren theoretisch nicht nötig ist. Ist die SenderBase Reputation Score Null, so ist die IP-Adresse bisher nicht durch Spamversand aufgefallen. In diesem Fall wird die E-Mail der nächsten Spamerkennungs-Stufe übergeben, der Inhaltsanalyse.

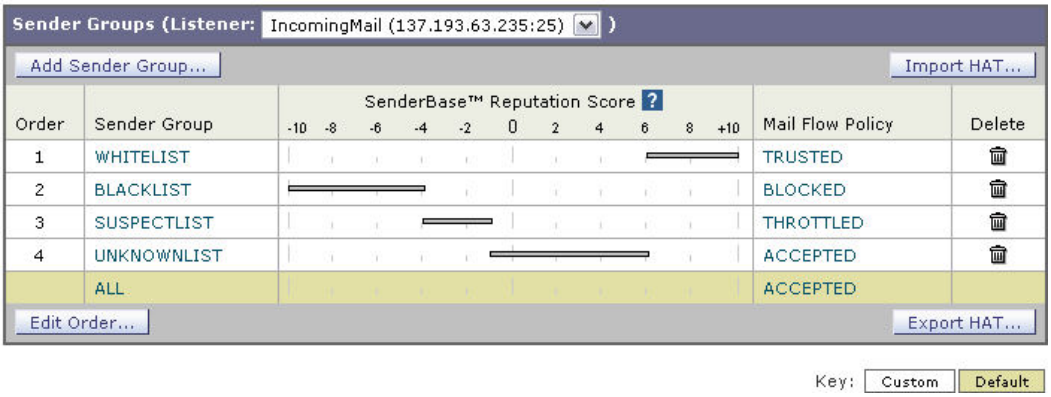


Abbildung 4.8: IronPort C10 - Sender Groups

Die SenderBase Reputation Score bestimmt, in welche Liste ein Absender eingetragen wird.

Für den ausgehenden E-Mail-Verkehr unterstützt die IronPort C10 zusätzlich das DomainKey-Verfahren.

4.5.4 CanIt Anti-Spam Appliance

Die CanIt Anti-Spam Appliance unterscheidet sich vom Ansatz her ein wenig von denen der bisher vorgestellten Antispam-Appliances. CanIt implementiert ein Delegationsverfahren, wie man es etwa von einem Verzeichnisdienst her kennt. So werden von einem Administrator zunächst globale Regeln zur Spamfilterung festgelegt (siehe Abb. 4.10). Anschließend kann er festlegen, welche Konfigurations-Möglichkeiten dem Enduser überlassen werden. Der Administrator delegiert also entsprechende Rechte an die Endnutzer.

Abbildung 4.9: CanIt Anti-Spam Appliance
[Bildquelle: www.pyramid.de]

Die Platzierung der Appliance im Netzwerk unterscheidet sich von den oben vorgestellten Systemen nicht. Gleiches gilt für die Inbetriebnahme.

Die E-Mail-Filterung erfolgt bei der CanIt stream-basiert, d.h. jeder Anwender hat seinen persönlichen Stream. Mit den entsprechenden Rechten kann er sich beispielsweise komplett aus der Spamfilterung herausnehmen oder aber seinen Scanner überdurchschnittlich streng einstellen. Das Verfahren setzt jedoch voraus, dass ein Endnutzer über Grundkenntnisse bezüglich Spam und Antispam-Mechanismen verfügt. Abbildung 4.11 zeigt den Quarantäne-Bereich eines Endnutzers. Im obigen Menü kann dieser seine eigenen Spamfilter-Regeln definieren bzw. anpassen. Ebenfalls kann die Aggressivität des Scanners eingestellt werden.

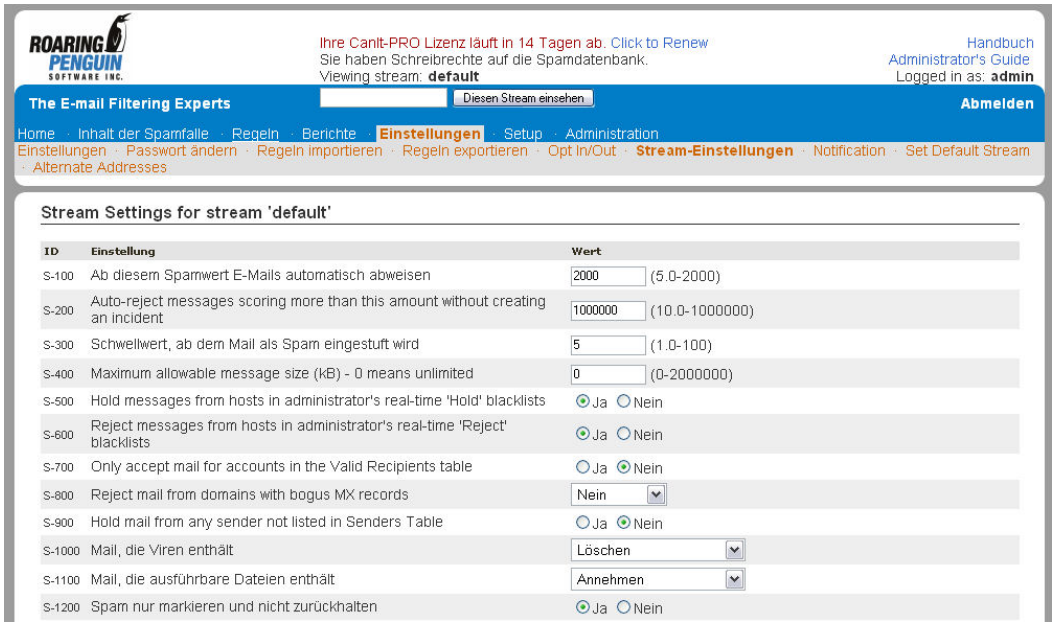


Abbildung 4.10: CanIt Anti-Spam Appliance - Administrator-Bereich
Der Administrator ist für die globalen Einstellungen verantwortlich und bearbeitet den Default-Stream.

E-Mail-Filterung

Bei der Konfiguration des Spamfilters wird dem Administrator sehr viel Flexibilität und Spielraum gewährt. Jede noch so kleine Regel lässt sich aktivieren oder deaktivieren. Die Schwellenwerte können ebenfalls individuell eingestellt werden. Einen besonderen Stellenwert nimmt der Bayes-Filter ein. Er lässt sich sowohl durch die Endnutzer als auch über das sogannte Roaring Penguin Training Network (RPTN) [Roaring Penguin Software Inc. (2008)] trainieren (siehe Abb. 4.12). Das Netzwerk tauscht weltweit Informationen mit Bayes-Filter anderer CanIt-Appliances aus.

Insgesamt werden folgende Spamfilter-Mechanismen verwendet:

- White- und Blacklists, Open Proxy Lists
- Greylisting
- regelbasierte Inhaltsfilter und Schlüsselwortfilter
- Absenderauthentifizierung durch SPF und Headeranalyse
- SpamAssassin (inkl. Bayesfilter)

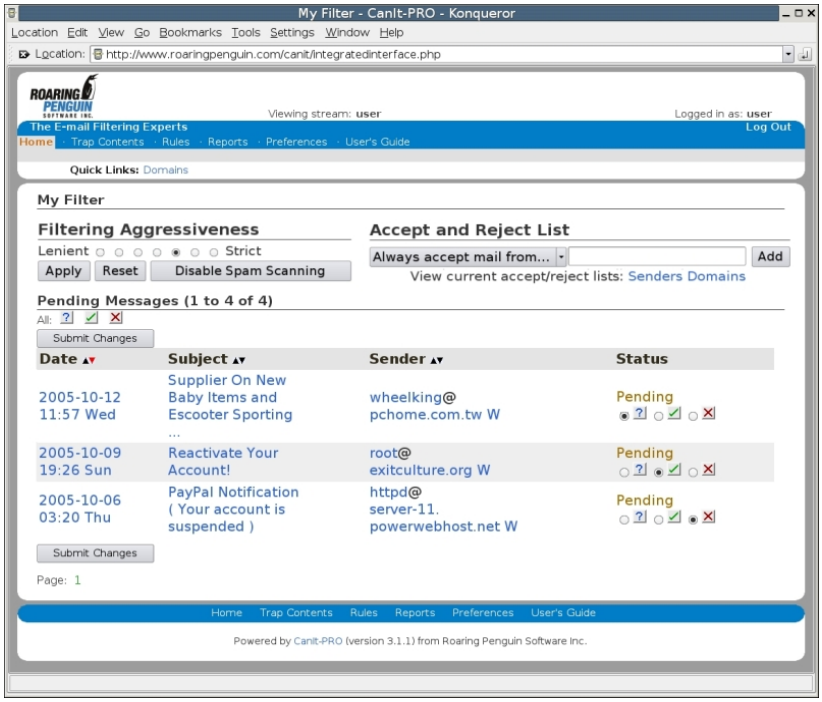


Abbildung 4.11: CanIt Anti-Spam Appliance - Endnutzer-Bereich
Hier kann ein Anwender die Spamfilterung seiner E-Mails sogar vollständig deaktivieren. [Bildquelle: www.roaringpenguin.com]

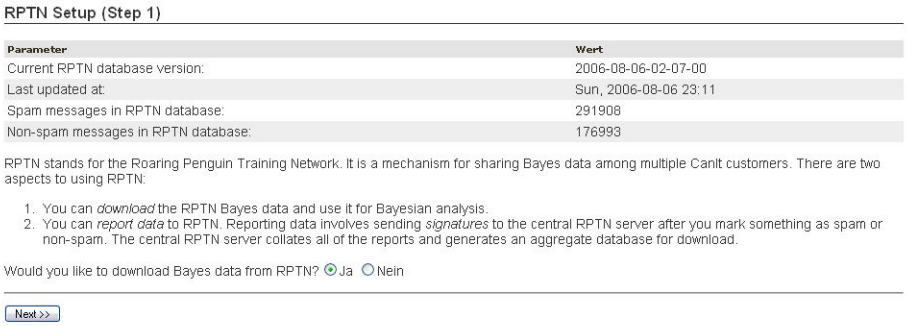


Abbildung 4.12: CanIt Anti-Spam Appliance - Bayes-Filter
Der Bayes-Filter kann durch das Roaring Penguin Training Network (RPTN) trainiert werden.

4.5.5 iku SPONTS Mail Security Appliance

Die Antispam-Lösung des deutschen IT-Unternehmens *iKu-Systemhaus AG* nennt sich SPONTS Mail Security Appliance und wurde 2004 erstmals auf den Markt gebracht.

Gemäß Hersteller soll pro Tag ein Maildurchsatz von 550.000 Nachrichten möglich sein [iKu GmbH & Co. KG (2008b)]. Als Betriebssystem wurde eine Debian-Linux-Distribution herangezogen. Ein root-Login ist hierbei möglich und für das Einspielen von Updates sowie Lizenzschlüssel sogar notwendig.

Abbildung 4.13: SPONTS-Appliance
Die SPONTS in einem Mini-ATX-Gehäuse.

Die SPONTS ist, wie alle anderen Appliances im Test auch, als SMTP-Proxy einzusetzen. Die Appliance muss die IP-Adresse des im MX eingetragenen Mailservers bekommen und leitet die empfangenen und gescannten E-Mails schließlich an den eigentlichen Mailserver weiter.

Gemäß iKu ist eine minimale Konfiguration in 10 Minuten möglich. Nachdem man in den Tabellen die lokalen Domänen eingetragen und den Backend-Server (eigentlicher Mailserver) definiert hat, ist die Box bereits für den Einsatz gerüstet.

Eine Protollierung aller Aktivitäten stellt iKu mit einer Journal- und Replayfunktion sicher. Im Journal sind alle E-Mails aufgeführt, die den MTA erreichten. Sogar abgewiesene Sendeversuche werden mit entsprechendem Grund für die Abweisung protokolliert. Die Replayfunktion ermöglicht es, bereits an den Backend ausgelieferte E-Mails erneut zu senden. Auf diese Weise kann keine E-Mail verloren gehen, falls der Backend mal ausfällt. Im Bereich Systeminfo wird dem Administrator zudem die Möglichkeit gegeben, die SMTP-Dialoge genau

nachzuvollziehen. Ein Download aller Protokolle im ZIP-Format erleichtert hierbei wesentlich die Arbeit. Eine weitere nützliche Option nennt sich UMS: Bei einem längeren Ausfall des Backend-Servers ist es möglich, die E-Mails über POP3 direkt bei der SPONTS abzuholen.

E-Mail-Filterung

Seit 2004 wirbt iKu mit dem sogenannten SPONTS-Effekt. Hierbei wird behauptet, dass man die E-Mail-Belastung auf das einzelne Postfach bezogen auf Dauer senken kann [iKu GmbH & Co. KG (2008b)]. Erreicht werden soll dies durch eine *550 user unknown*-Meldung im SMTP-Dialog, die auftritt, sobald eine Nachricht als Spam erkannt wurde. Dadurch soll erreicht werden, dass der Spammer denkt, das E-Mail-Konto gäbe es nicht mehr. Da man in den Konfigurationseinstellungen auch festlegen kann, mit welcher Begrüßung sich SPONTS im SMTP-Dialog melden soll, gibt es quasi keine Anhaltspunkte mehr, die SPONTS zu enttarnen. iKu geht davon aus, dass die genutzte E-Mail-Adresse langfristig aus der Mailingliste des Spammers verschwindet.

Ein eingehender Test zur Wirkung dieser Technik wurde von Eggendorfer durchgeführt [Eggendorfer (2004)]. Kritik übte er vor allem dadurch, dass das SMTP-Protokoll nur an gewissen Stellen eine *„550 user unknown“*-Meldung herausgeben kann. Wenn erst der Inhalt (DATA-Teil) der Nachricht übermittelt wurde, ist dies zum Beispiel nicht mehr möglich. Die Firma iKu reagierte und lässt in der aktuellen SPONTS-Version nun eine temporäre Fehlermeldung herausgegeben. Bei dem darauf folgenden Zustellversuch kann die SPONTS dann eine *„550 user unknown“*-Meldung ausgeben.

Zur Spamvermeidung verwendet SPONTS unter anderem die Software „SpamAssassin“. Folgende Mechanismen kommen dabei zum Einsatz:

- RBLs und Whitelists
- SMTP- und Domainüberprüfung des Absenders
- SPF
- RFC-Konformität
- Greylisting
- Inhaltsanalyse
- Bayes-Filter

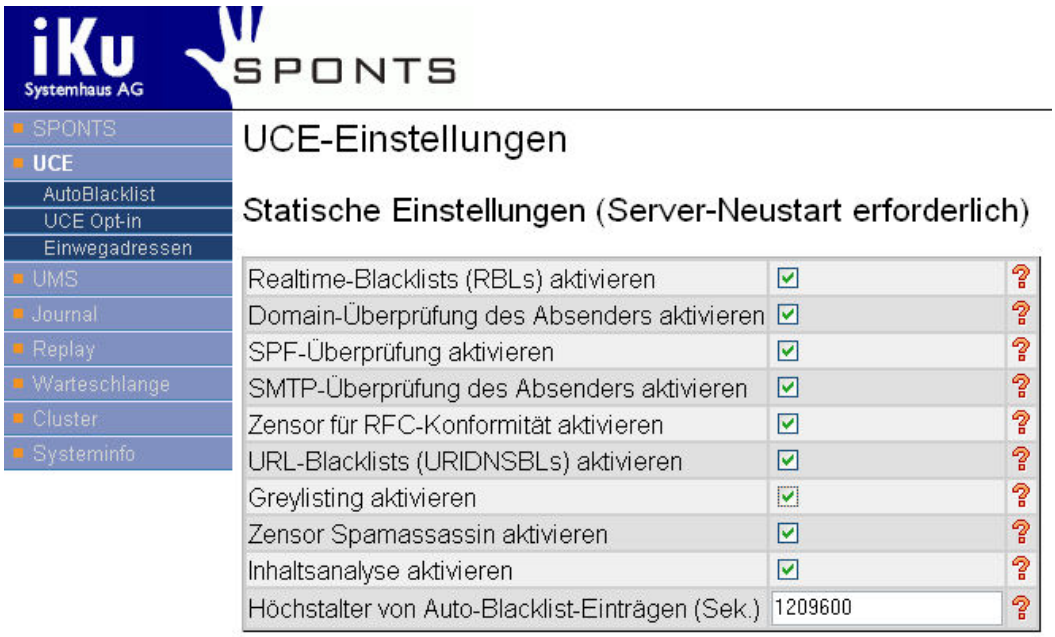


Abbildung 4.14: SPONTS-Appliance - UCE-Einstellungen

Alle Verfahren lassen sich individuell gewichten. Der Administrator kann sowohl einzelne Bewertungsfaktoren abändern, als auch ganze Mechanismen deaktivieren (siehe Abb. 4.14).

Eine anderes von iKu zum Patent angemeldetes Verfahren ist die Analyse des SMTP-Zeitverhaltens [iKu GmbH & Co. KG (2008a)]. Durch eine Analyse des typischen Zeitverhaltens eines MTAs soll die SPONTS seriöse Mailserver von Spammer-Software und Virenversendenden SMTP-Engines unterscheiden können.

4.5.6 eXpurgate

Alle oben beschriebenen Antispam-Systeme sind Appliances. Sie werden vor Ort in das Netzwerk integriert und filtern den E-Mail-Verkehr. Eine andere Möglichkeit der Spamfilterung wird durch sogenannte Application Service Provider (ASPs) angeboten. Diese bieten ebenfalls eine Filterung der E-Mails an, jedoch muss man dazu seinen gesamten E-Mail-Verkehr auf die Server der Dienstleister umleiten. Die Idee hinter der ASP-Variante ist, dass der ASP im DNS als zuständiger MX eingetragen wird und fortan alle E-Mails der Domäne annimmt. Nach einer Spamfilterung und Kategorisierung der Nachrichten werden diese dann weiter an den Domänen-eigenen Mailserver geschickt. Abbildung 4.15 verdeutlicht dieses Prinzip für „eXpurgate“, ein Dienst der Firma eleven [eleven (2008)].

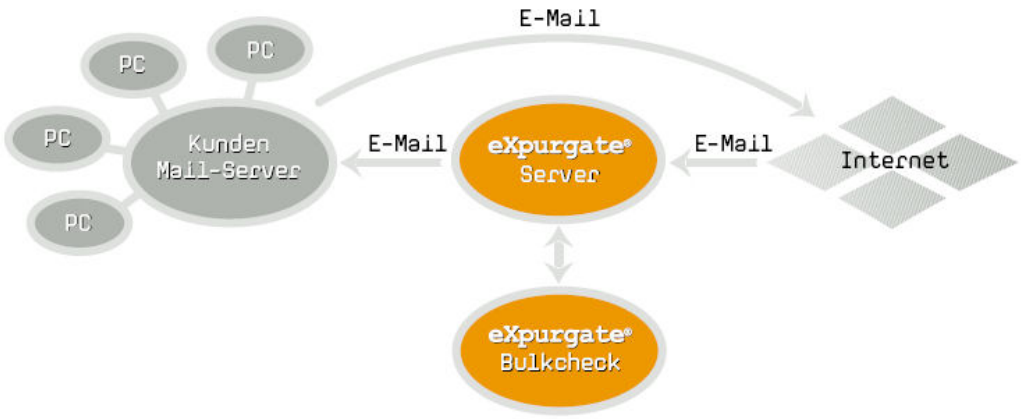


Abbildung 4.15: eleven eXpurgate - Prinzip
Nach einem Umweg über eleven's Server landen die E-Mails gefiltert beim eigenen Mailserver. [Bildquelle: www.eleven.de]

Zusammen mit weiteren Testverfahren nimmt eXpurgate folgende E-Mail-Kategorisierungen vor:

- Clean
- Spam
- Bulk
- Dangerous (z.B. E-Mails mit gefährlichem selbstausführenden Code)
- Dangerous.Virus (E-Mails mit Viren)
- Clean.Empty
- Bulk.Advertising

- Bulk.Porn

Die Einrichtung des eXpurgate-Dienstes dauert nur wenige Minuten. Es muss lediglich eine Kontaktperson benannt werden, sowie die Domäne(n), für die sich eXpurgate zuständig fühlen soll. Abschließend müssen nur die MX-Einträge der Domäne(n) auf die Mailserver von eXpurgate umgeschrieben werden (siehe Abb. 4.16).

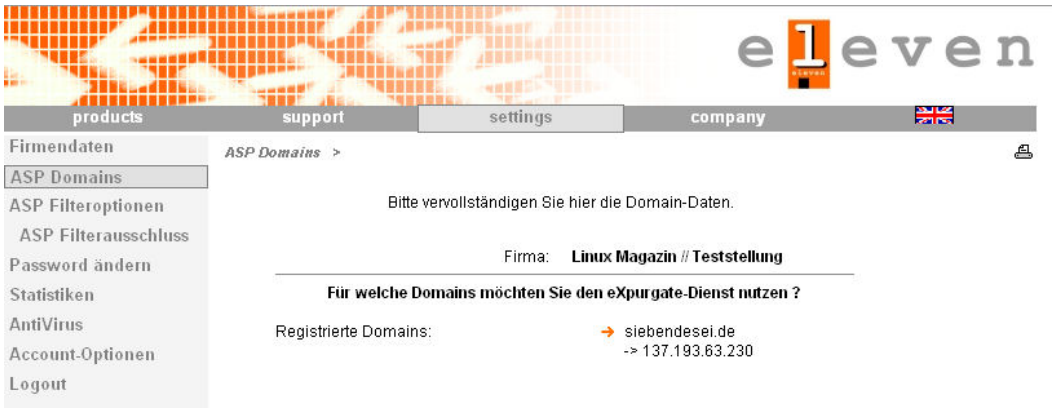


Abbildung 4.16: eleven eXpurgate - Portal
Im eXpurgate-Portal lassen sich u.A. die Domänen mit den zugehörigen Backends einstellen.

Über das Kundenportal ist es möglich, einzelne Accounts von der Filterung auszuschließen. Zusätzlich gelangt man von dort in den Statistikbereich, wo sich Informationen rund um den E-Mail-Verkehr abrufen lassen. In Form einer Excel-Datei können diese auch heruntergeladen werden.

Die von den Antispam-Appliances gewohnten Konfigurations-Optionen sind kaum vorhanden. Lediglich die Aktionen nach einer entsprechenden E-Mail-Klassifizierung können abgeändert werden. Man kann wählen, wie die E-Mails zu markieren sind, also welche Mitteilungen dem Header hinzugefügt werden sollen. Ein Mitspracherecht bei den verwendeten Antispam-Verfahren hat man nicht.

4.5.7 SpamStopsHere

Die Firma Greenview Data aus Kanada stellt ebenfalls einen ASP-basierten Spamfilter-Dienst zur Verfügung [Greenview Data Inc. (2008)]. Da das Prinzip das Gleiche ist wie bei eXpurgate, wird an dieser Stelle auf eine erneute Erläuterung verzichtet.

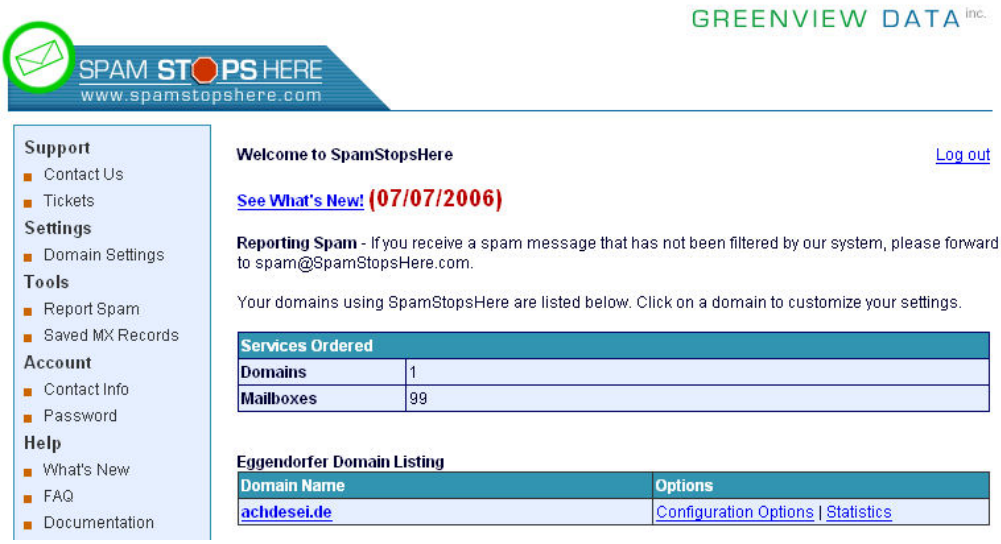
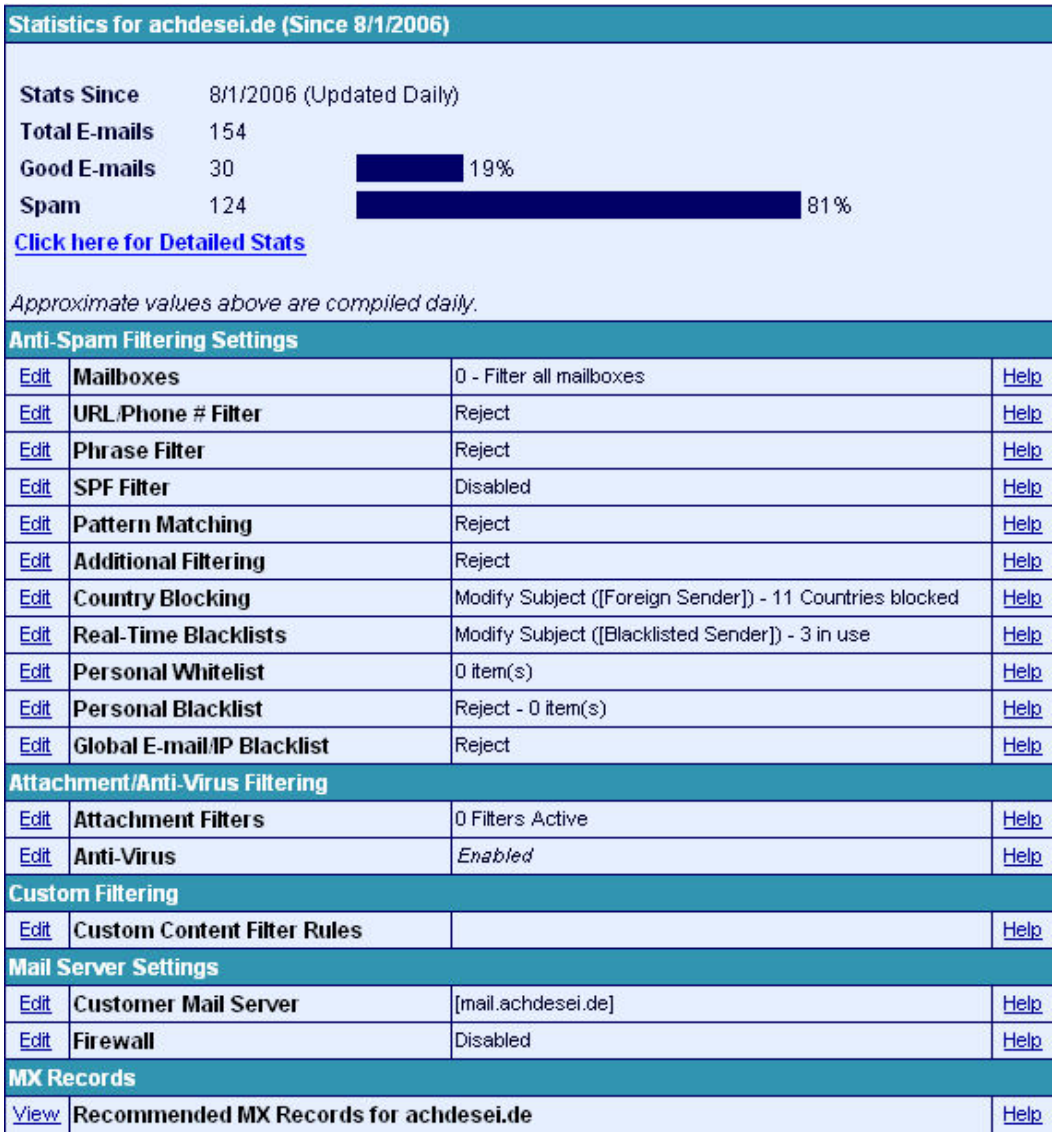


Abbildung 4.17: SpamStopsHere - Portal
Durch einen Klick auf Configuration Options gelangt man in den Konfigurationsteil der entsprechenden Domäne

Während sich bei eXpurgate lediglich Ausschluss-Accounts und Aktionen für Klassifizierungen definieren lassen, bietet SpamStopsHere individuelle Konfigurations-Optionen in den Bereichen Antispam-, Antivirus- sowie Contentfilterung an (siehe Abb. 4.18). So lässt sich der Spamfilter durch folgende Verfahren optimal an die Bedürfnisse des jeweiligen Unternehmens anpassen:

- Mailbox-OptIn: Man kann wählen, ob alle oder nur bestimmte Mail-Accounts gescannt werden sollen.
- *URL/Telefonnummern*: E-Mails mit unerwünschten URLs oder Telefonnummern werden herausgefiltert.
- *Phrasenfilter*: Filter für E-Mails mit für UCE typischen Sätzen und Redewendungen.
- *SPF*
- *Mustererkennung*
- *zusätzliche Inhaltsfilter*

- *Länder-Blockierung*: Absender aus bestimmten Ländern können geblockt werden.
- *RBLs*
- *lokale White- und Blacklists*
- *IP-Sperrlisten*



Statistics for achdesei.de (Since 8/1/2006)

Stats Since	8/1/2006 (Updated Daily)	
Total E-mails	154	
Good E-mails	30	19%
Spam	124	81%

Click here for Detailed Stats

Approximate values above are compiled daily.

Anti-Spam Filtering Settings			
Edit	Mailboxes	0 - Filter all mailboxes	Help
Edit	URL/Phone # Filter	Reject	Help
Edit	Phrase Filter	Reject	Help
Edit	SPF Filter	Disabled	Help
Edit	Pattern Matching	Reject	Help
Edit	Additional Filtering	Reject	Help
Edit	Country Blocking	Modify Subject ([Foreign Sender]) - 11 Countries blocked	Help
Edit	Real-Time Blacklists	Modify Subject ([Blacklisted Sender]) - 3 in use	Help
Edit	Personal Whitelist	0 item(s)	Help
Edit	Personal Blacklist	Reject - 0 item(s)	Help
Edit	Global E-mail/IP Blacklist	Reject	Help
Attachment/Anti-Virus Filtering			
Edit	Attachment Filters	0 Filters Active	Help
Edit	Anti-Virus	*Enabled*	Help
Custom Filtering			
Edit	Custom Content Filter Rules		Help
Mail Server Settings			
Edit	Customer Mail Server	[mail.achdesei.de]	Help
Edit	Firewall	Disabled	Help
MX Records			
View	Recommended MX Records for achdesei.de		Help

Abbildung 4.18: SpamStopsHere - Domänen Optionen
Wie bei einer Appliance können diverse Antispam-Verfahren aktiviert und angepasst werden.

4.5.8 WEB.DE und GMX

Bei den E-Mail-Providern WEB.DE und GMX wurde jeweils ein Account eingerichtet. Während eine normale E-Mail-Adresse insgesamt viermal auf den Webseiten zu finden ist, sind diese Adressen 40 mal (vier Webseiten pro Domäne multipliziert mit zehn Domänen) publiziert worden. Sie sind jedoch aussschließlich versteckt („hidden“) im Quelltext aufgeführt.

Um die Nachrichten von den Providern abzuholen, wurde *Fetchmail*[13]aus einem RPM heraus installiert. Fetchmail wurde so konfiguriert, dass er alle 20 Minuten die E-Mails aus den beiden Postfächern über POP3 abholt. Eine Abholung in kürzeren Abständen war bei WEB.DE nicht möglich.

Schließlich landen auch diese E-Mails in der Datenbank. Die Sendmail-Konfigurationsdateien wurden entsprechend ergänzt.

[13]Fetchmail: http://fetchmail.berlios.de/

5 Auswertung

In diesem Kapitel werden die Filterergebnisse der verschiedenen Systeme dargestellt. Desweiteren erfolgt eine Diskussion und Bewertung einzelner Resultate.

Vorher ist es jedoch notwendig, einige Besonderheiten und Rahmenbedingungen des Tests genauer zu erläutern. Bei allen Appliances wurden mittlere Spamfilter-Level eingestellt. Meistens waren diese standardmäßig schon vorkonfiguriert. Falls die Systeme Bayes-Filter aufwiesen, wurden diese zwar aktiviert, jedoch nicht manuell trainiert. Ein Training ist bekanntlich nur durch das Sichten jeder einzelnen E-Mail möglich, was den Aufwand dieser Arbeit gesprengt hätte. Weiterhin sei erwähnt, dass sich alle Spamfilter-Ergebnisse durch entsprechendes „Feintuning“ in den Filtereinstellungen theoretisch noch verbessern lassen.

Da die Testlizenzen meistens nur auf wenige Tage beschränkt waren, mussten des öfteren neue Lizenzschlüssel von den Herstellern angefordert werden. Zum Teil dauerte dies einige Tage. So kam es vor, dass bei den Appliances vorübergehend die Scannerfunktion deaktiviert war. Die E-Mails wurden in dieser Zeit ungescannt in die Datenbank weitergeleitet. Diese E-Mails waren identifizierbar und wurden bei der Beurteilung der Filtergüte ausgeschlossen.

Die Appliances boten jede für sich eigene Statistik-Funktionen in ihren Weboberflächen an. Diese Funktionen waren je nach Hersteller unterschiedlich implementiert. Während manche Systeme jeden SMTP-Verbindungsversuch als erfolgreiche Spam-Abwehr protokollierten, führten andere lediglich die tatsächlich gefilterten E-Mails auf. Diese Zahlen wurde bei der Bewertung daher ebenfalls nicht berücksichtigt. Gezählt wurden ausschließlich die E-Mails, die in der Datenbank eingetragen waren.

Der Ham-Versand erfolgte, wie im Kapitel drei bereits angedeutet, gleichmäßig und geschlossen. Ein Verteiler mit den E-Mail-Adressen der jeweiligen „ersten“ User einer Domäne wurde an alle Kommilitonen, Freunde und Bekannte geschickt. Diese sendeten daraufhin von all ihren E-Mail-Konten eine E-Mail an den kompletten Verteiler. Kleine Abweichungen in den absoluten Ham-Zahlen sind auf Tests vor dem eigentlichen Versenden zurückzuführen.

5.1 Scan-Ergebnisse

Die folgende Tabelle fasst die gescannten E-Mails der Domänen zusammen. Anhand der *true*-Spalten lassen sich die richtig klassifizierten E-Mails erkennen. Analog enthalten die *false*-Spalten die falsch zugeordneten Nachrichten. Die Ergebnisse von WEB.DE und GMX werden im Anschluss separat vorgestellt.

	gesamt	**true positives**	**false negatives**	**true negatives**	**false positives**
E-Mail ist:		Spam	Spam	Ham	Ham
markiert als:		Spam	Ham	Ham	Spam
McAfee	344	292	9	40	3
SPONTS	119	0	69	41	9
CanIt	229	167	17	38	7
IronPort	386	316	24	46	0
Symantec	653	537	69	47	0
eXpurgate	657	586	25	45	1
SpamStopsHere	434	366	26	42	0

Auf den ersten Blick stechen besonders die Symantec- und die IronPort-Appliance heraus. Beide Systeme haben keine *false positives* vorzuweisen. Es wurde also keine Ham-Mail falsch eingeordnet. Gleiches gilt für den ASP SpamStopsHere. Dies ist insofern eine beachtliche Leistung, dass die anderen Appliances genau die gleichen Ham-Mails zu scannen hatten.

Die SPONTS-Appliance war standardmäßig sehr aggressiv eingestellt. Daher führte eine Verletzung einer hochgewichteten Regel bereits zum Blocken einer E-Mail. Beispielsweise blockte die Appliance, wenn ein Absender auf einer RBL stand. Ebenfalls blockte sie bei einer Erkennung von aggressiven MTAs. Dies erklärt zum einen die im Vergleich zu den anderen Systemen geringe Gesamtzahl gefilterter E-Mails und zum anderen, dass keine *true positive*-Nachricht weitergeleitet wurde. Die Appliance hat solche Nachrichten gar nicht erst angenommen. Die neun weitergeleiteten *false positives* erklären sich durch ein relativ „mildes“ Zensoring des Antispam-Moduls SpamAssassin, welches zwar eine Spam-Markierung auslöste, eine Zustellung aber noch erlaubte. Obwohl im Journal der SPONTS-Box alle Zustellversuche von Spammern gelistet und auch abzählbar waren, machte ein Vergleich mit den anderen Appliances keinen Sinn. Erfahrungsgemäß sind es stets weitaus mehr Zustellversuche als eingehende E-Mails. Aus diesem Grund wurden die Zuverlässigkeiten bei der SPONTS nicht berechnet (siehe unten).

5.1.1 Zuverlässigkeiten und Fehlerraten

Wie im Test-Konzept beschrieben soll zusätzlich eine Normierung der Werte vorgenommen werden, um die Filterergebnisse besser vergleichen zu können. Dazu werden in der folgenden Tabelle die in Kapitel drei definierten Fehlerraten und Zuverlässigkeiten aufgeführt. Abbildung 5.1 zeigt die Systeme in einer graphischen Gegenüberstellung.

Anbieter	Spam-Zuverlässigkeit	Spam-Fehlerrate	Ham-Zuverlässigkeit	Ham-Fehlerrate
McAfee	97,0%	3,0%	93,0%	7,0%
SPONTS	-	-	82,0%	18,0%
CanIt	90,8%	9,2%	84,4%	15,6%
IronPort	92,9%	7,1%	100%	0%
Symantec	88,6%	11,4%	100%	0%
eXpurgate	95,9%	4,1%	97,8%	2,2%
SpamStopsHere	93,4%	6,6%	100%	0%

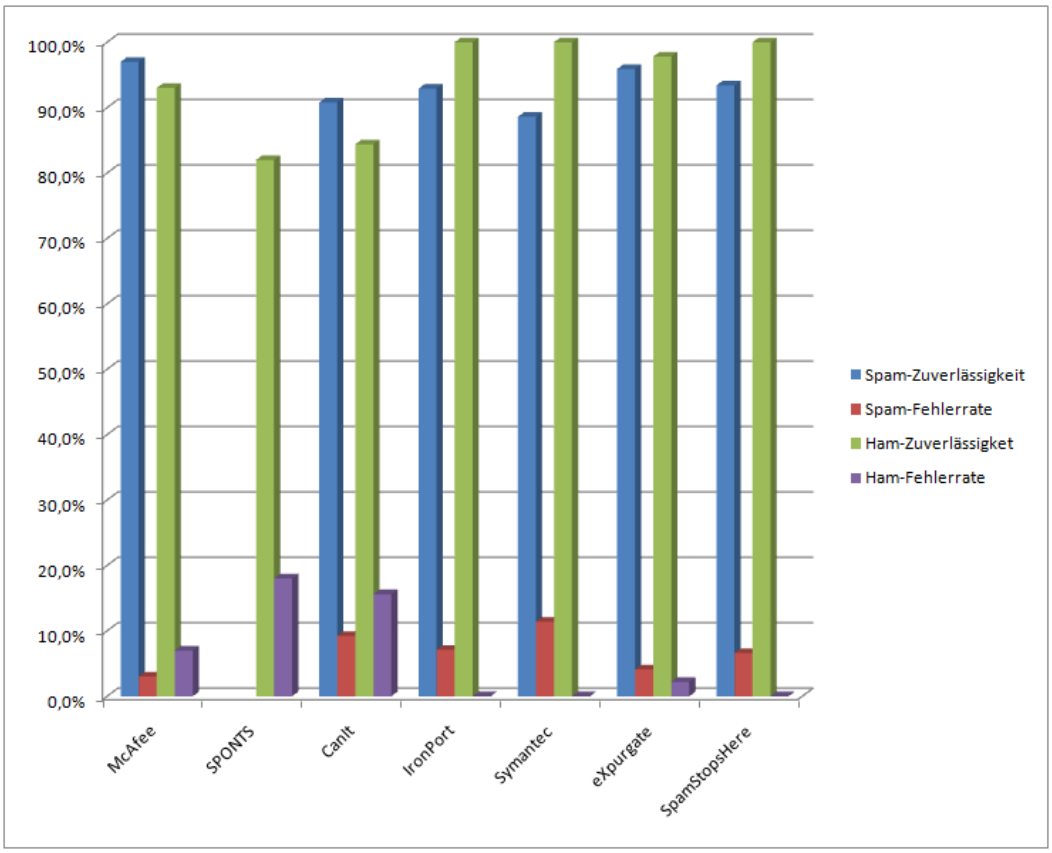


Abbildung 5.1: Zuverlässigkeiten und Fehlerraten

Kaum übersehbar ist neben den Ham-Zuverlässigkeiten von Ironport und Symantec auch die 97%ige Spam-Zuverlässigkeit der McAfee-Appliance. Die SPONTS- und die CanIt-Appliance weisen dagegen eine relatv hohe Ham-Fehlerrate auf.

Gesamt-Fehlerrate

Bei der Bewertung der Gesamt-Fehlerrate sollte man zusätzlich die einzelnen Fehlentscheidungen (*false positives* und *false negatives*) betrachten. Der Grund dafür sind die unterschiedlichen Konsequenzen. *False negatives* haben bei weitem nicht so tiefreichende Auswirkungen wie die *false negatives*, also der Verlust wichtiger E-Mails. In der folgenden Übersicht liegen Symantec und CanIt beispielsweise gleich auf. Die Ham-Zuverlässigkeit war hingegen bei Symantec 100% und bei CanIt lediglich 84,4%. Dies begründet die erforderliche Fallunterscheidung. Folgende Gesamt-Fehlerraten wurden von den Systemen erreicht:

Anbieter	Gesamt-Fehlerrate
McAfee	3,5%
SPONTS	-
CanIt	10,5%
IronPort	6,2%
Symantec	10,6%
eXpurgate	4,0%
SpamStopsHere	6,0%

Aus den Header-Protokollen der achten Domäne konnte entnommen werden, dass der ASP SpamStopsHere seine Ham-Fehlerrate von 0% wahrscheinlich durch die Art der Filterung erreichte. Anscheinend filterte man ausschließlich nach RBLs und Foreign Senders. Alle UCEs wurden aufgrund einer dieser beiden Filtertechniken markiert. Da die meisten der Ham-Versender Mail-Accounts aus seriös angesehen Domänen hatten, erklärt sich, weshalb keine Ham falsch eingeordnet wurde.

Ähnliche Werte erreichte der ASP-Dienst eXpurgate. Dies ist insofern interessant, da eXpurgate, wie in Kapitel vier angesprochen, dem Administrator keinerlei Freiheiten bei den Filtereinstellungen gewährt.

5.1.2 WEB.DE und GMX

Ein weiteres Ziel dieser Arbeit war es, die Spamfilter der Webmail-Anbieter GMX und WEB.DE in den Test zu integrieren. WEB.DE bot nach Anmeldung im WEB.DE-Club neben

einem normalen „3-Wege-Spamschutz“ zusätzlich einen „Premium Spam-Schutz“. Abgesehen von einigen wenigen Einstellungen, wie mit erkanntem Spam umzugehen sei, konnte man keine weiteren Optionen auswählen.

GMX dagegen erlaubte eine individuelle Spamfilter-Konfiguration. So wurden dem Nutzer persönliche Black- und Whitelists zur Verfügung gestellt. Weiterhin konnte man sich für die Nutzung einer GMX-internen Sperrliste entscheiden oder aber sogar für eine globale IP-Sperrliste.

WEB.DE hatte die Spamkategorisierung in Form einer Ordnereinteilung gelöst. So war der Posteingangsordner in „Freunde und Bekannte“, „Unbekannt“ und „Unerwünscht“ unterteilt. Letztlich entsprachen diese vom Prinzip her persönlichen White- und Blacklists, so dass theoretisch kaum eine Fehlentscheidung mehr getroffen werden konnte, wenn man seine Kontakte erst einmal bestätigt hatte. Ein Nachteil der Ordnernutzung für den Test war, dass nicht alle E-Mails an die Datenbank weitergeleitet werden konnten. Da WEB.DE E-Mails im „Unerwünscht“-Ordner nach 30 Tagen gelöscht hat, konnte man am Ende des Tests leider nicht mehr festgestellt, wieviele E-Mails insgesamt auf dem Account einliefen.

Im „Unbekannt“-Ordner befand sich keine UCE, was bedeutet, dass WEB.DE eine Spam-Zuverlässigkeit von 100 % erreichen konnte. Zwei Ham-Mails wurden fälschlicherweise in den „Unerwünscht“-Ordner verschoben, so dass eine Ham-Zuverlässigkeit von 94,7 % vorlag. Vier weitere Ham-E-Mails wurden mit einer erhöhten Spam-Wahrscheinlichkeit gekennzeichnet. Dennoch blieben sie im „Unbekannt“-Ordner.

GMX schreibt keine Ordner-Nutzung vor, so dass alle E-Mails via Fetchmail in die Datenbank gelangen konnten. GMX unterscheidet zwei Spam-Kategorien: „nicht Spam-verdächtig“ und „Spam-verdächtig“. In Spamverdacht gerieten 59 Nachrichten, wovon sieben Ham waren. Als unbedenklich wurden 37 E-Mails deklariert. Davon waren vier Nachrichten Spam. Dies entspricht einer Spam-Zuverlässigkeit von 92,9% und einer Ham-Fehlerrate von 17,5%.

5.2 Weitere Tests und Beobachtungen

Der Versuchsaufbau bot genug Flexibilität, um „Randerscheinungen“ zu beobachten. So wurde die Frage formuliert, wie lang der Zeitraum zwischen der Veröffentlichung einer E-Mail-Adresse und dem Empfang der ersten Spam-Mail sei. Der Zeitraum betrug 14 Tage.

Eine andere Fragestellung war, ob der Zusatz „nospam“ in einer E-Mail-Adresse eine Wirkung auf die Bespammung hat. Zur Erinnerung sei hier noch einmal kurz das verwendete E-Mail-Schema aufgeführt:

- User 1: *vorname.name@domain.de*
- User 2: *vorname2.name2.nospam@domain.de*
- User 2a: *vorname2.name2@domain.de* (nicht publiziert!)
- User 3: *vorname3.name3@nospam.domain.de*
- User 3a: *vorname3.name3@domain.de* (nicht publiziert!)
- User 4: *v.name@domain.de*

Während die User 1-3 offen auf der Webseite aufgeführt waren, stand User 4 lediglich im Quelltext. Die Adressen der User 2a und 3a wurden nie veröffentlicht. Addiert man die empfangenen E-Mails der User aller zehn Domänen zusammen, so ergibt sich folgendes Bild[1]:

User	E-Mails
User 1	1846
User 2	1549
User 2a	9
User 3	113
User 3a	1056
User 4	1784

User 1 wurde von allen E-Mail-Adressen am meisten bespammt. Dies könnte auf die Implementierung der Harvesting-Bots hindeuten. So wäre es denkbar, dass einige derart programmiert sind, dass sie die Webseite nicht bis zum Ende durchsuchen und eventuell nach dem Auffinden einer E-Mail-Adresse bereits wieder verlassen.

Das Gerücht, dass das Wort *nospam* innerhalb eines E-Mail-Namens Spam verhindert, wurde durch User 2 eindeutig widerlegt. Offensichtlich wurde der Zusatz schlichtweg ignoriert. Bestätigt wird dies durch die Anzahl der Mails von User 2a. Die neun verbleibenen E-Mails können dem Erraten von E-Mail-Adressen zugeschrieben werden.

Die User 3-Accounts unterlagen leider bis kurz vor Ende des Tests einem Konfigurationsfehler. Daher können die angegeben Werte nicht verwendet werden.

[1] Es handelt sich hierbei um die Anzahl aller eingegangen E-Mails, also auch jene, die vor dem eigentlichen Testbeginn mit den Filtersystemen, in der Anlaufphase, eingegangen sind.

Eine Überraschung stellt hingegen User 3a dar. Über eintausend empfangene E-Mails deuten eindeutig darauf hin, dass *nospam* aus dem Domänenteil der Adresse entfernt wurde.

Ebenfalls interessant sind die Ergebnisse für User 4. Dieser wurde ausschließlich im Meta-Quelltext (als „author mail“) publiziert. Entweder filtern Spammer gezielt nach solchen E-Mail-Adressen, weil davon auszugehen ist, dass Web-Administratoren regelmäßig ihre E-Mails abholen, oder die hohe Zahl hat mit der Reihenfolge der Anordnung auf der Webseite zu tun. Die „author mail“-Meta-Information stand relativ weit oben im Quelltext, noch vor dem öffentlich sichtbaren User 1. Dies würde den obigen Erklärungsversuch bekräftigen, dass Harvesting-Bots die Webseite nicht bis zum Ende scannen.

Auf dem GMX-Account gingen zusammengerechnet 96 E-Mails ein. Wie im dritten Kapitel erwähnt, wurden die E-Mail-Adressen der beiden Webmail-Accounts auf allen Webseiten, jedoch ausschließlich im Quelltext veröffentlicht. Verglichen mit den Zahlen der anderen Accounts erscheint 96 als sehr gering. Ein Grund dafür könnte sein, dass GMX, ähnlich wie die SPONTS-Appliance, viel Spam bereits im SMTP-Dialog abblockt bzw. gar nicht erst an das Postfach weiterleitet. Ein anderer Grund könnte die Art der Veröffentlichung sein. Die E-Mail-Adressen wurden als „hidden“ im Quelltext markiert. Zudem befanden sie sich im unteren Abschnitt des Quelltextes. Leider lässt sich dies aufgrund der fehlenden Gesamtzahlen bei WEB.DE nicht weiter belegen.

Eine andere Frage war, ob einige Dömanen stärker bespammt wurden als andere. In der Tat war eine unterschiedlich starke Bespammung zu registrieren (siehe folgende Tabelle[2]). Ein Muster ist dagegen nicht identifizierbar. Auffällig ist die Arbeitsweise der SPONTS-Appliance (driddesei.de), die, wie oben erwähnt, sehr viel UCE blockte.

Domäne	E-Mails
ersdesei.de	938
driddesei.de	316
vierdesei.de	700
fuenfdesei.de	628
sechsdesei.de	835
siebendesei.de	1194
achdesei.de	434
neundesei.de	992
zehndesei.de	634

[2] Auch hier handelt es sich sowohl um die E-Mails aus der Anlaufphase, als auch um die E-Mails aus der eigentlichen Testphase.

5.3 Spamfilter-Mechanismen

Zu Beginn dieser Arbeit wurde die Frage aufgeworfen, ob sich Spamfilter-Mechanismen identifizieren lassen, die besonders wirkungsvoll sind. Dies lässt sich anhand der absoluten und relativen Zahlen nicht beantworten. Die Spamfilter-Systeme, insbesondere die Appliances, boten jedoch Statistiken innerhalb ihrer Weboberflächen an. Diese wurden zwar für die Bewertung der Filtergüte – wie oben bereits erwähnt – nicht herangezogen, für die Beantwortung dieser Frage eignen sie sich jedoch sehr gut. Aus ihnen ging unter anderem hervor, durch welche Filter eine E-Mail abgewiesen bzw. klassifiziert wurden.

Eine Sichtung der jeweiligen Reports ergab, dass der Großteil aller Spam-Nachrichten (ca. zwei Drittel) durch absenderauthentifizierende Maßnahmen abgewehrt wurden. So wurden derartige Nachrichten entweder als Spam markiert oder bereits im SMTP-Dialog abgewiesen. Dies zeigt die besonders effektive Wirkung von Black- und Whitelists. Spamfilter-Techniken wie Greylisting oder SenderID hatten den Statistiken zufolge keinen Einfluss auf eine Klassifizierung. Dies lässt sich sicherlich durch die mangelnde Verbreitung und einer fehlenden breiten Akzeptanz erklären. Zwar werden die Techniken von den Herstellern implementiert, sie erfordern jedoch zusätzlich eine Umkonfigurierung des sendenen MTAs, was die Nutzung erheblich aufwendiger macht.

Voraussetzung für gute Filterergebnisse ist jedoch stets die Aktualität. Dies führt zu der Nutzung von globalen Spam-Erkennungs-Netzwerken, wie IronPort oder Symantec sie beispielsweise betreiben. Diese sind in der Lage, neue Spam-Wellen schnell zu erfassen und ermöglichen es den Betreibern, schnell zu reagieren und Updates für die Systeme zu veröffentlichen. Es wäre denkbar, dass die Aktualität der lokalen Listen den entscheidenen Unterschied in den erreichten Filterergebnissen gemacht hat. Ein großer, international vertretener Anbieter von Spamfilter-Systemen hat vermutlich auch bessere Möglichkeiten, die Aktualität seiner Systeme sicherzustellen, als ein rein nationaler Hersteller es kann.

Die Wirkung der Bayes-Filter konnte innerhalb dieses Testes nicht quantitativ nachgewiesen werden. Die Studie von Cormack und Lynam bestätigte jedoch, dass trainierte Bayes-Filter eine gute Waffe gegen Spam sind und in keinem Spamfilter-System fehlen sollten [Cormack und Lynam (2007)].

Es stellt sich die Frage, ob man ein Spamfilter-System allein aus Open Source-Komponenten zusammenstellen und mit diesem ähnlich gute Ergebnisse erzielen kann wie mit einem kommerziellen System. Das Antispam-Modul SpamAssassin[3] integriert in seiner aktuellen Version beispielsweise sowohl Blacklisten als auch diverse statistische Filter (u.a. Bayes). Abgesehen von einer fehlenden garantierten Zuverlässigkeit und fehlendem Support kann man diese Frage demnach durchaus mit ja beantworten. Fraglich bleibt allerdings die Filtergüte

[3] http://spamassassin.apache.org/

dieses Systems bezogen auf die soeben angesprochene Aktualität. Hier dürften kommerzielle Systeme aufgrund ihrer Erkennungs-Netzwerke einen Schritt voraus sein. Unberührt dessen erfordert ein „eigenes“ Spamfilter-System bei weitem mehr Wartungsaufwand als ein System, dass sich automatisch Updates einholt und durch einen professionellen Support gewartet wird.

5.3.1 Der ASP-Ansatz

Der ASP-basierte Ansatz stellt eine sehr gute Alternative zu lokalen Spamfilter-Systemen dar. Genereller Vorteil einer ASP-basierten Lösung ist die Abwendung von Gefahren für die eigene IT-Infrastruktur. Sowohl Pishing-Mails als auch Viren gelangen zumindest nicht über den SMTP-Weg in das Netzwerk des Unternehmens. Desweiteren spart man sich neben dem Konfigurationsaufwand auch jeglichen weiteren Wartungsaufwand, den beispielsweise ein Bayes-Filter dringend benötigt. Kosten für die Hardware (Neuanschaffung, Austausch) fallen ebenfalls nicht an. Schließlich können personelle Ressourcen in der IT-Abteilung anders verplant werden. Gerade für kleine bis mittelständische Unternehmen, bei welchen sich oft nur ein Angestellter für die IT verantwortlich sieht, ist die ASP-Lösung eine Alternative.

Ein weiterer Vorteil der zentralisierten Spamfilterung sei die breite Basis hervorgehoben: Da sämtliche E-Mails aller Kunden in Echtzeit eintreffen, ist man in der Lage, in kürzester Zeit Filter-Regeln so anzupassen, dass auch die neuesten Spam-Wellen abgewehrt werden können. Damit hat ein ASP bereits ein Netzwerk zur Verfügung, das Hersteller von Antispam-Appliances noch zusätzlich nebenher betreiben müssen (siehe IronPorts SenderBase oder Symantecs BLOC).

Nachteil einer solchen Lösung dürfte zweifelsfrei die Abgabe der Kontrolle an einen Dritten sein. So ist es eine Art Vertrauensfrage, ob man alle seine E-Mails über Systeme laufen lassen will, über die man selbst keine Kontrolle hat. Zum einen besteht die Gefahr des Missbrauchs und zum anderen die Gefahr des Abhandenkommens einer Nachricht. Missbrauch beugt eleven durch Verschlüsselung der E-Mails vor. So werden alle E-Mails durch einen „Kontrollsummen-Algorithmus“ verschlüsselt und erst dann anhand der Prüfsumme kategorisiert. Das bedeutet, dass der eigentliche Inhalt der E-Mail unangetastet bleibt.

Die erreichten Ergebnisse bestätigen die Wettbewerbsfähigkeit der ASP-Lösungen.

6 Zusammenfassung

Ziel dieser Arbeit war die Erstellung eines Konzepts zur Evaluierung verschiedener Spamfilter-Mechanismen. Weiterhin erfolgte eine Umsetzung dieses Konzepts sowie eine Auswertung der gewonnenen Erkenntnisse.

Nach der Einführung des Spam-Begriffes erfolgte die theoretische Aufarbeitung der geläufigsten Filter-Mechanismen. Anschließend wurde das Konzept ausgearbeitet und umgesetzt. Wesentliche Schritte waren dabei das Aufsetzen eines Webservers zur Publizierung von E-Mail-Adressen im Internet sowie das Vorbereiten eines E-Mail-Servers mit Datenbank-Anbindung zur Aufzeichnung aller eingehenden E-Mails. Eine wichtige Voraussetzung für den Test war die Arbeit mit Echtzeit-Spam. Spam-Archive wurden nicht genutzt.

Für die Evaluierung wurden fünf Antispam-Appliances, zwei Application Service Provider (ASPs) sowie zwei Webmail-Anbieter herangezogen, welche über einen Zeitraum von ca. sechs Wochen E-Mails filterten.

Die Evaluierung kam unter anderem zu dem Ergebnis, dass ca. zwei Drittel der Spam-Mails durch absenderauthentifizierende Maßnahmen erkannt werden konnten. Hierbei leisteten vor allem Realtime Blackhole Lists (RBLs) einen wirkungsvollen Dienst.

International agierende Hersteller von Spamfilter-Systemen unterhalten heutzutage meist große Netzwerke, aus denen sich aktuelle Trends der Spamversendung schnell erkennen lassen. Dies ist ein nicht zu unterschätzender Vorteil gegenüber kleineren Herstellern oder selbst zusammengestellten Systemen.

Die Ergebnisse zeigten weiterhin, dass ASPs eine sehr gute Alternative zur lokalen Installation von Spamfilter-Appliances darstellen. Ein Blick in die Referenzen der an dem Test beteiligten Anbieter zeigt, dass Firmen mittlerweile regen Gebrauch von dieser Art der Spamfilterung machen.

Obwohl manche Systeme sehr gute Werte bezüglich der Klassifizierung von E-Mails erreichen konnten, bleibt die Tatsache bestehen, dass man mit reaktiven Spamfilter-Mechanismen stets einen Schritt hinterher läuft. Eine präventive Spambekämpfung müsste jedoch weit vor der technischen Ebene beginnen. Denkbar wären hier unter anderem verschärfte gesetzliche Strafen für den Versand von Spam. Eine international gleiche Handhabung wäre dafür allerdings Grundvoraussetzung.

Ein denkbarer präventiver Ansatz auf technischer Ebene könnte beispielsweise bei der Unterbindung von Harvesting-Bots ansetzen. Dies wäre zum Beispiel durch einen breit angelegten Einsatz sogenannter „Honeypots“[1] möglich.

Da die in Spam beworbenen Produkte jedoch nach wie vor Käufer finden werden, bleibt im Moment nur die Gewissheit, dass auch morgen wieder eine Spam-Mail im Posteingang sein wird. . .

[1] Honeypots sind Webserver, die durch eine fortlaufende Generierung neuer temporärer Webseiten einen Harvester sozusagen „einfangen“.

Literaturverzeichnis

[Allman u. a. 2007] ALLMAN, E. ; CALLAS, J. ; DELANY, M. ; LIBBEY, M. ; FENTON, J. ; THOMAS, M.: *DomainKeys Identified Mail (DKIM) Signatures*. 2007. – URL http://www.ietf.org/rfc/rfc4871

[Allman 2003] ALLMAN, Eric: Spam, Spam, Spam, Spam, Spam, the FTC, and Spam. In: *ACM Queue* 1 (2003), Nr. 6, S. 62–69. – ISSN 1542-7730

[BSI 2005] BSI: *Antispam-Strategien – Unerwünschte E-Mails erkennen und abwehren*. 2005. – URL http://www.bsi.bund.de/literat/studien/antispam/antispam.pdf

[Clement u. a. 2008] CLEMENT, Michael ; PAPIES, Dominik ; BOIE, Harder-Johann: Kosten und Kostentreiber von unerwünschten Werbemails (Spam) – Eine empirische Analyse auf Provider- und Anwenderseite. In: *Zeitschrift für Betriebswirtschaft* 78 (2008), Nr. 4, S. 339–366. – ISSN 0044-2372 (Print) 1861-8928 (Online)

[Cook u. a. 2006] COOK, Duncan ; HARTNETT, Jacky ; MANDERSON, Kevin ; SCANLAN, Joel: Catching spam before it arrives: domain specific dynamic blacklists. In: *ACSW Frontiers '06: Proceedings of the 2006 Australasian workshops on Grid computing and e-research*. Darlinghurst, Australia, Australia : Australian Computer Society, Inc., 2006, S. 193–202. – ISBN 1-920-68236-8

[Cormack und Lynam 2007] CORMACK, Gordon V. ; LYNAM, Thomas R.: Online supervised spam filter evaluation. In: *ACM Trans. Inf. Syst.* 25 (2007), Nr. 3, S. 11. – ISSN 1046-8188

[Crocker 1997] CROCKER, D.: *RFC 2142 - MAILBOX NAMES FOR COMMON SERVICES, ROLES AND FUNCTIONS*. 1997. – URL http://www.ietf.org/rfc/rfc2142.txt

[DigiPortal Software 2008] DIGIPORTAL SOFTWARE, Inc.: *ChoiceMail*. 2008. – URL http://www.digiportal.com/choicemail.html

[Dumais u. a. 1998] DUMAIS, Susan ; PLATT, John ; HECKERMAN, David ; SAHAMI, Mehran: Inductive learning algorithms and representations for text categorization. In: *CIKM '98: Proceedings of the seventh international conference on Information and knowledge management*. New York, NY, USA : ACM, 1998, S. 148–155. – ISBN 1-58113-061-9

[Eggendorfer 2004] EGGENDORFER, Tobias: Spezialfilter. Anti-Spam-Appliance mit Langzeitwirkung. In: *Linux Magazin* (2004), Nr. 9

[Eggendorfer 2005] EGGENDORFER, Tobias: *No Spam!* Software & Support Verlag GmbH, 2005. – ISBN 3-935042-71-X

[eleven 2008] ELEVEN, Gesellschaft zur Entwicklung und Vermarktung von Netzwerktechnologien m.: *eXpurgate*. 2008. – URL http://www.eleven.de/

[Garcia u. a. 2004] GARCIA, F.D. ; HOEPMAN, J.-H. ; NIEUWENHUIZEN, J.van: Spam Filter Analysis. (2004), August. – URL citeseer.ist.psu.edu/article/fd04spam.html

[Graham 2002] GRAHAM, Paul: *A Plan for Spam*. 2002. – URL http://paulgraham.com/spam.html

[Graham 2003] GRAHAM, Paul: *Better Bayesian Filtering*. 2003. – URL http://paulgraham.com/better.html

[Jung und Sit 2004] JUNG, Jaeyeon ; SIT, Emil: An Empirical Study of Spam Traffic and the Use of DNS Black Lists. In: *Internet Measurement Conference*. Taormina, Italy, October 2004

[Klensin u. a. 1995] KLENSIN, J. ; FREED, N. ; ROSE, M. ; STEFFERUD, E. ; CROCKER, D.: *RFC 1869 - SMTP Service Extensions*. 1995. – URL http://www.ietf.org/rfc/rfc1869.txt

[Kuri 2005] KURI, Jürgen: *GMX erneut auf Antispam-Blockliste*. 2005. – URL http://www.heise.de/newsticker/meldung/64798

[Li und Zhong 2006] LI, Kang ; ZHONG, Zhenyu: Fast statistical spam filter by approximate classifications. In: *SIGMETRICS '06/Performance '06: Proceedings of the joint international conference on Measurement and modeling of computer systems*. New York, NY, USA : ACM, 2006, S. 347–358. – ISBN 1-59593-319-0

[Lyon und Wong 2006] LYON, J. ; WONG, M.: *Sender ID: Authenticating E-Mail*. 2006. – URL http://www.ietf.org/rfc/rfc4406

[Barracuda Networks Datum unbekannt] BARRACUDA NETWORKS: *An Overview of Spam Blocking Techniques*. Datum unbekannt. – URL http://www.barracudanetworks.com/ns/downloads/barracuda_spam_blocking_techniques.pdf

[Ferris Research 2008] FERRIS RESEARCH: *Press Conference: Why Today's Spam Filters Fail*. 2008. – URL http://www.ferris.com/2008/04/16/press-conference-why-todays-spam-filters-fail/

[Greenview Data Inc. 2008] GREENVIEW DATA INC.: *SpamStopsHere*. 2008. – URL http://www.spamstopshere.com/

[iKu GmbH & Co. KG 2008a] IKU GMBH & CO. KG: *Analyse des SMTP-Zeitverhaltens*. 2008. – URL http://www.sponts.de/timing.jsp

[iKu GmbH & Co. KG 2008b] IKU GMBH & CO. KG: *SPONTS*. 2008. – URL http://www.sponts.de/

[IronPort Systems, Inc. 2008a] IRONPORT SYSTEMS, INC.: *IronPort AsynchOS*. 2008. – URL http://www.ironport.com/de/technology/ironport_asyncos_operating_system.html

[IronPort Systems, Inc. 2008b] IRONPORT SYSTEMS, INC.: *IronPort SenderBase*. 2008. – URL http://www.ironport.com/de/technology/ironport_senderbase_network.html

[IronPort Systems, Inc. 2008c] IRONPORT SYSTEMS, INC.: *IronPort Threat Operation Center gibt Spam-Warnung für 2008*. 2008. – URL http://www.ironport.com/de/pdf/ironport_press_2008-03-04.pdf

[Lundgren IT 2008] LUNDGREN IT: *Greylisting*. 2008. – URL http://www.greylisting.org

[Roaring Penguin Software Inc. 2008] ROARING PENGUIN SOFTWARE INC.: *Roaring Penguin Training Network (RPTN)*. 2008. – URL http://www.roaringpenguin.com/files/rptn.pdf

[Symantec (Deutschland) GmbH 2008] SYMANTEC (DEUTSCHLAND) GMBH: *Symantec Brightmail AntiSpam*. 2008. – URL http://eval.symantec.com/mktginfo/de/de/enterprise/fact_sheets/DE-BrightMail_Anti_Spam.pdf

[The Radicati Group 2007] THE RADICATI GROUP: *Market Stats & Industry Commentary - Volume 4 Issue 3*. 2007. – URL http://www.radicati.com/news/market_stats/vol_4-3.asp

[The Spamhaus Project 2008a] THE SPAMHAUS PROJECT: *The 10 Worst ROKSO Spammers*. 2008. – URL http://www.spamhaus.org/statistics/spammers.lasso

[The Spamhaus Project 2008b] THE SPAMHAUS PROJECT: *SBL and XBL*. 2008. – URL http://www.spamhaus.org

[The SPF Projekt 2008] THE SPF PROJEKT: *Sender Policy Framework*. 2008. – URL http://www.openspf.org/

[Trend Micro 2008] TREND MICRO: *Email Reputation Services*. 2008. – URL http://us.trendmicro.com/us/products/enterprise/network-reputation-services

[University of Alberta 2008] UNIVERSITY OF ALBERTA: *Spam Free e-Mail Service*. 2008. – URL http://sfm.cs.ualberta.ca

[Microsoft 2006] MICROSOFT: *Sender ID Framework*. 2006. – URL http://www.microsoft.com/mscorp/safety/technologies/senderid/default.mspx

[Pantel und Lin 1998] PANTEL, Patrick ; LIN, Dekang: SpamCop: A Spam Classification & Organization Program. In: *Learning for Text Categorization: Papers from the 1998 Workshop*, AAAI Technical Report WS-98-05, 1998. – URL citeseer.ist.psu.edu/pantel98spamcop.html

[Pfleeger und Bloom 2005] PFLEEGER, Shari L. ; BLOOM, Gabrielle: Canning Spam: Proposed Solutions to Unwanted Email. In: *IEEE Security and Privacy* 3 (2005), Nr. 2, S. 40–47. – ISSN 1540-7993

[Postel 1982] POSTEL, Jonathan B.: *RFC 821 - Simple Mail Transfer Protocol*. 1982. – URL http://www.ietf.org/rfc/rfc821.txt

[Prakash 2007] PRAKASH, Vipul V.: *Vipul's Razor*. 2007. – URL http://razor.sourceforge.net

[Resnick 2001] RESNICK, P.: *RFC 2822 - Internet Message Format*. 2001. – URL http://www.ietf.org/rfc/rfc2822.txt

[Rhyolite Software 2008] RHYOLITE SOFTWARE, LLC: *Distributed Checksum Clearinghouse*. 2008. – URL http://www.rhyolite.com/anti-spam/dcc

[Sophos 2008] SOPHOS: *Sophos Security Threat Report, Q1*. 2008. – URL http://www.sophos.com/sophos/docs/eng/marketing_material/sophos-threat-report-Q108.pdf

[Spammer-X 2006] SPAMMER-X: *Inside The World of Spam: From the Eyes of a Spammer*. EU Spam Symposium, Maastricht. 2006. – URL http://spamsymposium.eu/files/Spammer-X.ppt

[Ungerer 2007] UNGERER, Bert: *Spam-Sperrung*. 2007. – URL http://www.heise.de/netze/artikel/print/90037

[Wong und Schlitt 2006] WONG, M. ; SCHLITT, W.: *Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, Version 1*. 2006. – URL http://www.ietf.org/rfc/rfc4408

[Zhong u. a. 2006] ZHONG, Zhenyu ; RAMASWAMY, Lakshmish ; LI, Kang: Towards a Ham Archive. In: *invited talk, MIT SPAM Conference, Boston, MA, Mar 28*, 2006

Versicherung über Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit im Sinne der Prüfungsordnung nach §24(5) ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 14.07.2008	
Ort, Datum	Unterschrift