E-Mail unter Unix

Marcel Holtmann
Universität Bielefeld - Technische Fakultät
AG Rechnernetze und verteilte System
marcel@rvs.uni-bielefeld.de

27. Mai 1999

E-Mail unter Unix

- Die Geschichte der Elektronischen Post
- Wie funktioniert E-Mail?
- Sendmail - Das Allround-Talent
- Installation und Konfiguration von Sendmail
- Spamming und Anti-Relaying
- Procmail und Majordomo
- SMTP after POP

Referenzen

1) Die Sendmail-Bibel: Bryan Costales mit Eric Allmann
2nd Edition, O'Reilly

2) Sendmail-Konfiguration: Ralf Hildebrandt
c't 9'99: S.210ff

3) www.sendmail.org

4) RFC821: *Simple Mail Transfer Protocol*
RFC822: *Standard for the Format of ARPA Internet Text Messages*

Was ist eine Mail

- Das Gegenstück zur E-Mail ist die Deutsche Bundespost
- Man schreibt einen Brief - unterschreibt diesen und steckt ihn in einen Breifumschlag. Auf den Umschlag schreibt man unten links die Empfänger-Adresse und oben rechts klebt man eine Briefmarke
- Danach wird der Brief in den Postkasten eingeworfen. Ist der Empfänger in der Nachbarschaft, so wird der Brief von dem lokalen Postamt zugestellt. Wenn der Brief woanders hin muß, so wird er zu dem passenden Postamt transportiert und von diesem dann ausgeliefert

Sendmail und die Deutsche Bundespost

- Das Unix-Programm Sendmail ist ein Postzentrum (Auslieferung) und ein Postamt bzw. Briefkasten (Absenden) für die elektronische Post
- Elektronische Post wird normalerweise innerhalb von Sekunden zugestellt - Briefe bei der Post brauchen mindestens einen Tag
- Ein Adresswechsel wird innerhalb von Sekunden vollzogen und E-Mails können in der ganzen Welt zugestellt werden. Zieladressen werden dynamisch und „on-demand“ abgefragt. Nach Namensänderungen wird die Post immer noch korrekt ausgeliefert
- E-Mail kann verschiedene Transportmedien benutzen. Was damit gleich zu setzen wäre, wenn die Bundespost den Service von UPS nutzen würde um ihre Briefe auszuliefern

E-Mail ist keine Briefpost

- Es werden zwar gerne Analogien zwischen Briefpost und E-Mail gezogen, dennoch sind die beiden Bereiche weit von einander entfernt
- Begrifflichkeiten wie Umschlag und Zieladressen sind aber von ähnlicher Bedeutung
- Die meisten Funktionen, Namen oder Begriffe wurden aber an der Briefpost angelehnt

Die Geschichte der E-Mail

- Die ersten „E-Mail's“ gab es so ca. um 1978 mit dem Programm `delivermail` (später `sendmail` von Eric Allman)

- 1980 wurde das ARPAnet von NCP (Network Control Protocol) nach TCP/IP umgestellt und die Rechneranzahl stieg von 256 auf über eine Millarde

- Zur Mailauslieferung wurde immer noch UUCP oder FTP benutzt

- Um eine Vereinheitlichung des E-Mail versenden und der Auslieferung zu erhalten wurde das *Simple Mail Transfer Protocol* (SMTP) entworfen

Die Geschichte der E-Mail

- Durch die Hierachie der Domainnamen und die TCP/IP-Protokoll-Suite änderten sich auch die E-Mail Adressen

  `benutzer@domainname.topleveldomain`

- Die Kommunikation mit den Serverprogrammen erfolgte über den Port 25 eines Rechners

- Alle Rechner sprechen mit TCP/IP und SMTP ein und dieselbe Sprache

- Das Programm Sendmail beherrscht dennoch weiterhin UUCP und lokale Auslieferung von E-Mail

Was ist nun eine E-Mail?

- Eine E-Mail besteht aus drei Teilen. Dem *Header*, dem *Body* und dem *Envelope*
- In dem Header ein E-Mail stehen die Daten über Absender, Datum etc.
- Der Body ist der Inhalt der E-Mail mit einem Subject
- Ein Envelope ist bei E-Mails die Empfängeradresse oder -adressen, denn im Gegenzug zu einem Brief kann eine E-Mail ohne Mehraufwand an unterschiedliche Personen geschickt werden
- Die Empfängeradresse wird normalerweise bei den Daten für den Header mit untergebracht, dennoch wird ein Briefkopf von einem Umschlag unterschieden

Der Aufbau einer E-Mail

- Eine E-Mail ist eine reine ASCII-Datei
- Jede E-Mail beginnt mit dem Wort `From` und kennzeichnet hiermit den Absender und das Absendedatum

```
From you@Here.US.EDU  Fri Dec 13 08:11:44 1996
Received: (from you@localhost) by Here.US.EDU (8.8.4/8.8.4)
       id AA04599 for you; Fri, 13 Dec 96 08:11:44 -07000
Date: Fri, 13 Dec 96 08:11:43
From: you@Here.US.EDU (Your Full Name)
Message-Id: <9631131611.AA0214@Here.US.EDU>
To: you
```

Der Aufbau einer E-Mail

- Danach folgen Paramter der Form $Name$`:`, die Eigenschaften wie Absender und Datum aufführen. Die Daten in der `From` Zeile sind beim Versenden der E-Mail hinzugefügt worden, wobei die Paramter wie Datum etc. beim Erstellen der Nachricht erzeugt werden

- Der Körper der E-Mail besteht aus der Zeile `Subject:`, eine Leerzeile und dem Text der Nachricht

```
Subject: a test

This is a one line message
```

Der Aufbau einer E-Mail

- Der Umschlag einer E-Mail ist normalerweise nur die Zeile `To:`, also die Empängeradressen

```
To: friend1, friend2@remote
```

- Im weiteren und wie auch in vielen anderen Büchern werden die Begriffe Header und Envelope unter dem deutschen Stichwort Umschlag benutzt

- Manchmal wird auch nur die Zeile `From`. . . als Header bezeichnet, da diese vom Mailprogramm beim Aufruf bzw. Versenden der E-Mail hinzugefügt wird

Das Postamt in der elektronischen Welt

- In der elektronischen Welt gibt es einen MTA und einen MUA
- Ein MTA ist der *Mail-Transfer-Agent*, der für die Weiterleitung der E-Mail zuständig ist
- Der MTA ist aber zugleich Briefkasten, Postbote und Verteilerzentrum
- Für die Auslieferung der E-Mail verwendet der MTA meistens andere Dienste des Systems. Meistens in Form von Programmen wie `mail`, `procmail` etc. Eine Auslieferung an andere Postzentren wird mit Hilfe von SMTP bewerkstelligt
- mit dem MUA (*Mail-User-Agent*) werden die Briefe geschrieben und dann in den Briefkasten geworfen, d.h. dem MTA übergeben

Der Briefkasten an der Haustür

- Elektronische Post muß nun ja auch ausgeliefert werden und manchmal sammeln sich ein paar Briefe in einem Postkasten

- Die Auslieferung übernimmt der *Mail Deliver Agent* MDA. Ein typischer MDA ist das Programm `mail` unter BSD Unix und `mailx` unter SysV

- In Unix System werden die Mails im *mbox*-Format in einer Datei abgelegt, die sich entweder unter `/var/mail/`*user* oder im Homeverzeichnis des Benutzers befindet (hier dann meistens ~`/Mail/Inbox`)

Das *mbox*-Format

- Beim mbox-Format werden die Mails hintereinander abgespeichert. Die Reihenfolge entspricht der Auslieferungsfolge

- Die Datei hat das Format von aufeinander folgenden E-Mails, die durch eine Leerzeile getrennt sind

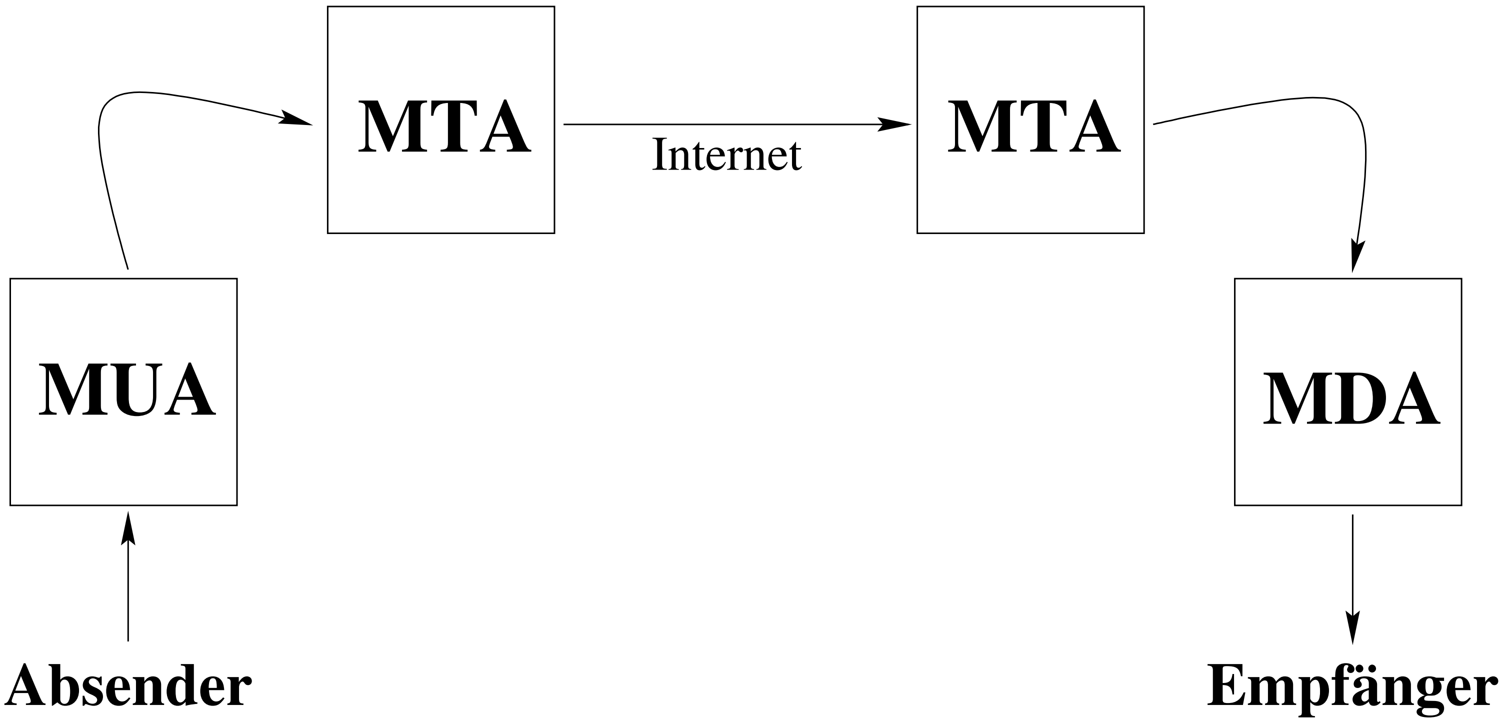
```
From ...
To: test

Blah Blah

From ...
To: irgendwem

Hallo
```

Der Weg einer E-Mail



Ein einfache E-Mail mit `mail`

- Mit dem Programm `mail` kann man schnell und einfach eine E-Mail verschicken

```
> mail marcel@rvs.uni-bielefeld.de
From: holtmann@uni-bielefeld.de
To: marcel@rvs.uni-bielefeld.de
Subject: Eine Mail an mich selbst

Hallo Marcel,

ich schreibe mir jetzt eine Mail
.
```

Was macht das Programm `mail`?

- Eigentlich tut das Programm nichts anderes als die Empfängeradresse (Argument) und den E-Mail-Text (mit Header-Informationen) an den MTA zu übergeben

- Der MTA *sendmail* hat aber neben dem Eingabeinterface über SMTP auch eine direkte Eingabe per Kommandoaufruf

```
> /usr/lib/sendmail marcel@rvs.uni-bielefeld.de

...

.
```

Eine E-Mail über SMTP

```
> telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 localhost ESMTP Sendmail 8.8.8/8.8.8; Thu, 20 May 1999 08:40:54 +0200
Helo localhost
250 localhost Hello holtmann@localhost [127.0.0.1], pleased to meet you
Mail From: holtmann@uni-bielefeld.de
250 holtmann@uni-bielefeld.de... Sender ok
Rcpt To: marcel@rvs.uni-bielefeld.de
250 marcel@rvs.uni-bielefeld.de... Recipient ok (will queue)
```

Eine E-Mail über SMTP

```
Data
354 Enter mail, end with "." on a line by itself
Subject: Eine Mail

Hallo ...
.
250 IAA06493 Message accepted for delivery
Quit
221 merlin.holtmann.net closing connection
Connection closed by foreign host.
```

Die E-Mail beim Empfänger

```
From mholtman@TechFak.Uni-Bielefeld.DE Thu May 20 09:48 MET 1999
Received: from gemma.TechFak.Uni-Bielefeld.DE
        (gemma.TechFak.Uni-Bielefeld.DE [129.70.136.103])
        by mailhost.rvs.uni-bielefeld.de (8.9.2/8.9.1)
        with ESMTP id JAA17853
        for <marcel@rvs.uni-bielefeld.de>;
        Thu, 20 May 1999 09:48:33 +0200 (MET DST)
Received: from beo.TechFak.Uni-Bielefeld.DE
        (beo.TechFak.Uni-Bielefeld.DE [129.70.133.33])
        by gemma.TechFak.Uni-Bielefeld.DE (8.9.1/8.9.1/TechFak/)
        with SMTP id JAA24343
        for <marcel@rvs.uni-bielefeld.de>;
        Thu, 20 May 1999 09:48:31 +0200 (MET DST)
```

Die E-Mail beim Empfänger

```
From: Marcel Holtmann <mholtman@TechFak.Uni-Bielefeld.DE>
Received: by beo.TechFak.Uni-Bielefeld.DE (SMI-8.6/pk970604A)
        id JAA21766; Thu, 20 May 1999 09:48:30 +0200
Date: Thu, 20 May 1999 09:48:30 +0200
Message-Id: <199905200748.JAA21766@beo.TechFak.Uni-Bielefeld.DE>
To: marcel@rvs.uni-bielefeld.de
Subject: Eine Mail an mich selbst
Content-Type: text
Content-Length: 20

Hallo Marcel,
...
```

Zentralisierung

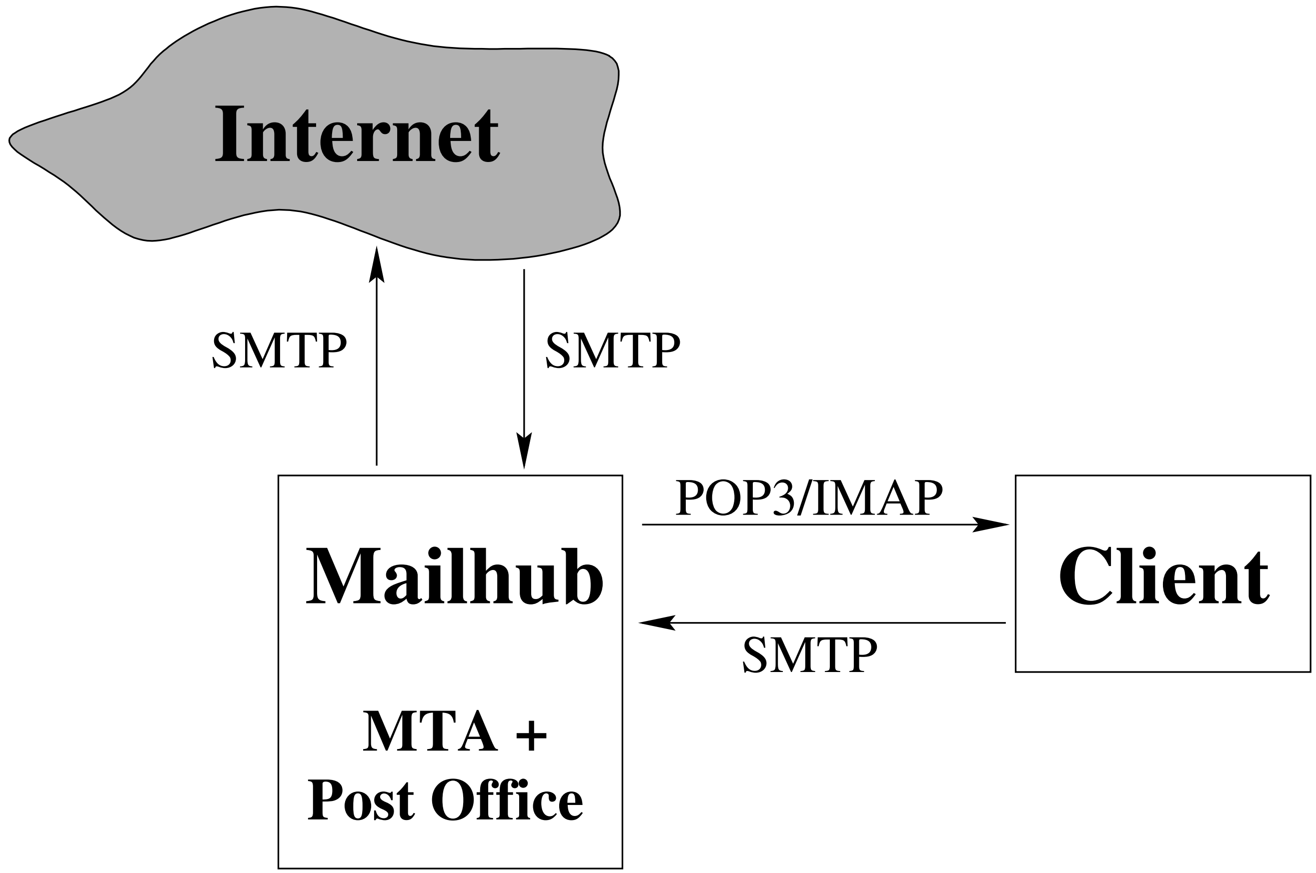
- Auf jedem neuen Unix-System ist eine Version von Sendmail installiert. Teilweise wurde der Orginal-Source erweitert oder modifiziert

- Für größere Netzwerke lohnt sich die Installation eines zentralen Mailserver, der über SMTP erreichbar ist

- Die einzelnen Workstations nehmen dann nur noch Mail an und leiten diese an den Mailserver weiter

- Für die Abholung der E-Mails von einem Mailserver kommen dann die Post Office Protokolle *POP3* und *IMAP* in Frage

Zentralisierung



Das Programm Sendmail

- Sendmail ist ein *Mail Transfer Agent* von Eric Allman
- Es ist das Standardmailsystem unter Unix
- Über 70% der Rechner im Internet laufen mit Sendmail
- Die Konfiguration von Sendmail ist eines der größten Probleme für einen System Administrator
- Bezugsquelle ist `http://www.sendmail.org`

Installation von Sendmail

- Die aktuelle Version von Sendmail ist 8.9.3 vom 4. Februar 1999
- `ftp.sendmail.org/pub/sendmail/sendmail.8.9.3.tar.gz`
- Die Datei muß entpackt werden und in dem Verzeichnis `sendmail-8.9.3/src` wird dann `./makesendmail` aufgerufen
- In dem Verzeichnis `obj.$OS.$OSVER.$MACH` befindet sich dann die Datei `sendmail`, welche dann nach `/usr/lib` oder `/usr/sbin` kopiert werden sollte

Konfiguration von Sendmail

- Neben dem Sendmail-Programm wird auch noch eine Konfigurationsdatei benötigt: „Die `sendmail.cf`“
- Sendmail kann komplett über diese Datei gesteuert werden
- Sämtliche Aktionen (z.B. Posteingang, Postauslieferung etc.) werden in Abhängigkeit von dieser Datei ausgeführt
- Das Erstellen dieser Datei von Hand ist sehr kompliziert und Fehleranfällig

Die Datei `sendmail.cf`

```
#########################
#   Format of headers   #
#########################

H?P?Return-Path: <$g>
HReceived: $?sfrom $s $.$?_($?s$|from $.$_)
        $.by $j ($v/$Z)$?r with $r$. id $i$?u
        for $u; $|;
        $.$b
H?D?Resent-Date: $a
H?D?Date: $a
...
```

Und die *Rewriting Rules*

```
########################################
###   Ruleset 0 -- Parse Address   ###
########################################

S0
...

R$* < @ > $*              $@ $>Parse0 $>3 $1              user@ => user
R< @ $=w . > : $*         $@ $>Parse0 $>3 $2              @here:... -> ...
R$- < @ $=w . >           $: $(dequote $1 $) < @ $2 . >   dequote "foo"@here
R< @ $+ >                 $#error $@ 5.1.3 $: "User address required"
R$* $=O $* < @ $=w . >    $@ $>Parse0 $>3 $1 $2 $3        ...@here -> ...
R$-                       $: $(dequote $1 $) < @ *LOCAL* >        dequote "foo"
R< @ *LOCAL* >            $#error $@ 5.1.3 $: "User address required"
...
```

Warum keine einfachere Konfiguration?

- Die Konfigurationseinträge sind so ausgelegt, das sie schnell geparst werden können

- Durch die komplexen Ersetzungregeln kann man Sendmail dazu bringen, alles zu tun, was man möchte

- Man kann die Auslieferung für jeden Benutzer unterschiedlich händeln (z.B. Lokal, POP3, SMTP, Weiterleitung oder Zurückweisung)

- Man kann den *Envelope* von ankommenden und abgehenden Mails so modifizieren, daß ein einheitliches Aussehen existiert (z.B. Masquarading, Account $\leftrightarrow$ E-Mail Interferenzen etc.)

Erstellen einer Konfiguration

- Die `sendmail.cf` wird normalerweise nicht mehr von Hand erstellt oder modifiziert
- Man läßt sich die Datei mit Hilfe von Makros und dem GNU-Tool `m4` generieren
- Die einzige Arbeit ist dann noch das Erstellen eines Makros, welches meistens unter 10 Zeilen lang ist
- Die fertige Konfigurationsdatei wird dann nach `/etc` bzw. `/etc/mail` kopiert

Eine einfache Makro-Datei

```
VERSIONID(`@(#)mail.mc    8.10 (Bielefeld) 20/05/1999')
OSTYPE(linux)
DOMAIN(generic)
MAILER(local)
MAILER(smtp)
```

```
> cd sendmail-8.9.3/cf/cf
> m4 ../m4/cf.m4 file.mc > file.cf
```

Das Starten von Sendmail

- `/usr/lib/sendmail -bd -q30m`
- Sendmail wird mit `-bd` im Daemon-Modus gestartet
- Verbindungen auf Port 25 (SMTP) werden von Sendmail verarbeitet
- Nichtzustellbare E-Mails werden 30 Minuten in die Queue gelegt, bevor ein erneuter Sendeversuch unternommen wird
- Sendmail kann nun Mails entgegennehmen und lokal sowie auch über SMTP ausliefern bzw. weiterleiten

Sendmail und Aliase

- Die Aliase werden in Sendmail mit der Datei `/etc/mail/aliases` konfiguriert
- Für den Account Postmaster <u>muß</u> ein Alias vorhanden sein (RFC 822)
- Aliase dienen der Weiterleitung von E-Mails
- Einfache Mailinglisten können mit Aliasen realisiert werden
- Selbst der Mailverteiler Majordomo nutzt die Alias-Funktion von Sendmail

Ein Beispiel

```
# Following alias is required by the mail protocol, RFC 822
# Set it to the address of a HUMAN who deals with the mail problems.
Postmaster: root

# Alias for mailer daemon; returned messages from our MAILER-DAEMON
MAILER-DAEMON: postmaster

# Aliases to handle mail to programs or files, eg news or vacation
nobody: /dev/null
root:   holtmann

# Alias for distribution list, members specified here:
staff: ladkin,holtmann

# Alias for distribution list, members specified elsewhere:
keyboards: :include:/home/holtmann/keyboards.list

# Alias for a person, so they can receive mail by several names:
ladkin: ladkin@cs.mit.edu
marcel: holtmann
```

Der umgekehrte Alias

- Genau wie eingehende Mail, kann auch ausgehende Mail einen anderen Absender bekommen

- Das Umschreiben der Absenderadressen erfolgt mit Hilfe der Datei `/etc/mail/genericstable`

```
holtmann   marcel@rvs.uni-bielefeld.de
```

- So kann z.B. mit Hilfe der Datei `genericstable` und `aliases` eine E-Mail Adresse mit einen anderen Account verbunden werden

Unix-Account:	`holtmann`
E-Mail Adresse:	`marcel@...`

Relaying

- Ein Mail-Hub muß E-Mails von seinen Clients annehmen können und weiterleiten
- Theoretisch muß ein MTA von jedem Rechner auf der Welt eine E-Mail per SMTP annehmen und diese weiterleiten oder lokal ausliefern
- Relaying ist ein „Internet-Service“ um die Ausfallsicherheit von Systemen zu gewährleisten
- Über den DNS können Fallback-Mailserver angegeben werden, die die Auslieferung übernehmen. Diese Server können sich in einem ganz anderen Netz befinden

Gefahr des Relaying

- Jeder Benutzer kann *jeden* Mailhub als Absender für seine E-Mails benutzen
- Versender von Werbe-Mails (SPAM-Mails) können einfach Mails absetzen und benutzen dann die Rechnerresourcen des Server-Betreiber
- Die Hilfsbereitschaft im Internet zum Weiterleiten von E-Mails ist derzeit auf gleich **NULL** gesunken
- Sendmail ab der Version 8.8.x kommt mit der Voreinstellung, die Relaying verbietet
- Sendmail müssen explizit die Hosts angegeben werden, von denen es Mails zum Versenden annehmen soll

Konfiguration des Relaying

- Man kann mit dem Makro `FEATURE(‘relay_entire_domain’)` alle Clients einer DNS-Domain erlauben über den Mailhub die Post zu versenden
- Mit `FEATURE(‘relay_based_on_MX’)` wird allen Rechnern, die den Server als Fallback-MX angegeben haben, das Relaying erlaubt
- Mit Hilfe der Datei `/etc/mail/access` wird das Relaying für spezielle IP’s, Domainnamen, E-Mail Adressen etc. erlaubt bzw. verweigert
- Das Makro `FEATURE(‘access_db’)` sagt Sendmail, das es diese Datei abfragen soll

Format der Datei `access`

```
spam@aol.com              REJECT
peter@cyberspam.com       RELAY
cyberspam.com             REJECT
192.168.212               REJECT
client1.net.com           RELAY
```

- Mit `FEATURE(‘rbl’)` kann man die *Realtime Black List* einbinden, die alle Spam-Adresse bzw. IP’s enthält und Online erreichbar ist

- **Was passiert, wenn man durch einen dummen Fehler selber auf diese Liste kommt?**

Probleme des Anti-Relaying

- Wenn man sich über einen Service-Provider einwählt kann man nur dessen Server zum Mailversenden benutzen

- Bei Notebook's mit wechselnden IP's muß man jedesmal die Einstellungen des MUA neu konfigurieren

- Man weiß nie, was mit der E-Mail Adresse auf einem fremden MTA gemacht wird

Ich will aber immer meinen Mailserver benutzen!

Die Frage:

Wann darf ein Client ein Relaying durchführen?

Das Problem:

Eine Authentifizierung ist im SMTP nicht vorgesehen

Die Lösung: „SMTP after POP“

- Jeder Benutzer der seine E-Mails abgeholt hat kann auch E-Mails versenden

- Nach einer erfolgreichen POP-Anfrage kann der Benutzer für ein bestimmtes Zeitintervall (z.B. 10 min) von der IP relayen

- Der Server wird für einen Zeitraum und für eine bestimmte IP geöffnet

- Der mögliche Mißbrauch wird auf ein Minimum reduziert und ist für Spammer unattraktiv

Die Implementierung

- Dynamic Relay Authorization Control (DRAC)
  `http://mail.cc.umanitoba.ca/drac/`

- Das Paket DRAC ist auf den MTA Sendmail abgestimmt und kann mit jedem POP oder IMAP-Server eingesetzt werden, sofern dessen Source-Code vorliegt

- Patches für den Cyrus IMAP-Server und den QPopper von Qualcomm sind vorhanden

Das Konzept von DRAC

- Sendmail wird angewiesen eine weitere Relay-Datei zu benutzen (`/etc/mail/dracd`)
- Auf dem gleichen Rechner wird der Daemon `rpc.drac` gestartet (am besten mit Sendmail zusammen)
- Diesem Daemon kann nun mit Hilfe von RPC mitgeteilt werden, wenn ein Client eine zeitlich begrenzte Relayerlaubnis erhalten soll
- DRAC trägt diese Erlaubnis in die Datei `/etc/mail/dracd` ein

Modifikation der POP oder IMAP-Server

- In dem Source-Code des jeweiligen Server wird ein RPC-Aufruf eingebaut, der die Daten an den DRACD sendet
- Der Aufruf muß nach jeder erfolgreichen Anfrage erfolgen
- Die Mailserver müssen so gesichert werden, das nur von gewissen Rechnern der RPC-Aufruf akzeptiert wird
- DRAC ist ein Ansatz zur Lösung des Problems, weitere Alternativen sind Mailserver mit SSL