



## qmail - ein sicherer und schneller sendmail-Ersatz

Dr. Erwin Hoffmann

fehcom

<http://www.fehcom.de>

E-Mail - SMTP - ist sicherlich neben dem Web-Protokoll HTTP die verbreitetste Anwendung im Internet. Der Dokumentenaustausch via E-Mail und Internet stellt zusätzlich zu Telefon und Fax für viele Firmen ein zentrales Rückgrat ihrer Kommunikation dar. Neben der allgemeinen Verfügbarkeit des Systems kennzeichnet vor allem der Durchsatz, d.h. die Anzahl der zu verarbeitenden E-Mails pro Zeiteinheit, ein entsprechendes System aus. Als langfristigen Schnitt kommen hierbei auf einen Mitarbeiter eine gesendete und eine empfangene E-Mail pro Arbeitstag. Der Autor dieses Artikels empfängt - da er auf mehreren Mailingslisten ist - etwa 50-100 E-Mails pro Tag und sendet zwei. Das Anwendungsprofil entscheidet also über die effektive Zahl. Grosse Internet Service Provider (ISP) legen ihre Systeme so aus, dass diese zwischen 0,5 und 2 Millionen E-Mails pro Tag verkraften können. Neben Verfügbarkeit und Durchsatz kommt noch das Kriterium der Sicherheit hinzu. Unter Sicherheit wird in diesem Zusammenhang verstanden, dass das System

- nicht angreifbar ist, z.B. durch ein fehlerhaftes Parsing von E-Mail-Adressen und Interpretation als Kommando,
- gegen DoS-Attacken weitgehend immun ist,
- die Verarbeitung von E-Mails als ausführbare Skripte mit Systemrechten ausschliesst,
- hinreichende Massnahmen zur Unterdrückung von Spam-E-Mail bereit stellt sowie
- evtl. mittels Virens Scanner die E-Mails vor der Auslieferung überprüfen kann.

Der weit verbreitete UNIX SMTP Mail Transfer Agent (MTA) sendmail [1] mit seinen (damaligen) Sicherheitslücken war für Dan J. Bernstein - Assistant-Professor am Department für Mathematics, Statistics und Computer Science an der University of Illinois in Chicago - Grund genug eine eigene, sichere und schnelle Implementierung eines SMTP-MTA zu kreieren: qmail [2].

### SMTP-MTAs

Bevor wir uns mit dem Aufbau und der Arbeitsweise von qmail beschäftigen, will ich erst einmal auf die prinzipielle Struktur eines SMTP-MTA einge-

hen (Abbildung 1). Aufgabe eines SMTP-MTA, ist es

- (Empfang:) E-Mails von SMTP-Clients in einem Serverprozess zu empfangen,
- (Ablage:) die E-Mails lokal abzulegen und
- (Versand:) lokal erstellte E-Mails an andere SMTP-Server zu verschicken bzw.
- (Relaying:) empfangene E-Mails an andere MTAs weiterzuleiten.

SMTP ist bekanntlich keine Benutzer-zu-Benutzer-Anwendung sondern eine Rechner-zu-Rechner-Kommunikation. Neben dem SMTP-Client und -Server kommt dem Message Store (MS) zentrale Bedeutung zu. Im MS werden E-Mails temporär aufbewahrt (Queuing) bevor sie lokal ausgeliefert oder im Sendeprozess verschickt werden. SMTP-Server und Client arbeiten beim Versenden und teilweise auch beim Empfang von E-Mails eng mit dem Domain Name System (DNS) zusammen. Neben dem Lookup der IP-Adresse über den FQDN (Full Qualified Domain Name) im A-Record, beinhaltet das DNS auch die Informationen über SMTP-Zielsysteme als sog. Mail Exchange (MX-) Records.

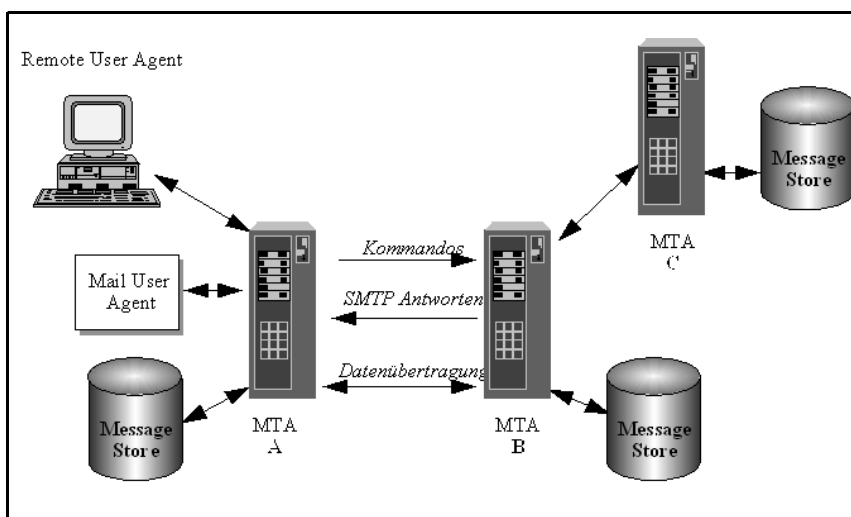


Abbildung 1: Aufbau und Kommunikation von SMTP-MTAs.

Die Kommunikation zwischen SMTP-Client und SMTP-Server besteht aus Protokollelementen (HELO/EHLO etc.) und der eigentlichen Datenübertragungsphase. E-Mails werden bis zur Weiterleitung in einem Message-Store (MS) abgelegt. Mittles eines Mail-Delivery-Agents (MD) wird die E-Mail entweder lokal in ein Postfach zugestellt oder aber an den SMTP-Sendeprozess überstellt. Auf die E-Mails in den Postfächer greifen (lokale) Mail-User-Agents (MUAs) oder Remote-User-Agents (RUAs) zu.

Bei einer typischen E-Mail-Adresse “<Vorname.Nachname@Host.Domain.tld>” ist der SMTP-Server für die Auswertung der Adresse rechts vom “@” verantwortlich. Der SMTP-Server muss daher lediglich Kenntnis über seine IP-Adresse und seinen Hostnamen besitzen. Hingegen ist es Aufgabe des Ablageprozesses (Message Delivery) zu entscheiden, ob der Empfänger lokal vorliegt. Der Ablageprozess muss (qmail-local) über geeignete Systemaufrufe feststellen, ob der Empfänger (links vom “@”) dem System bekannt ist, bzw. über ein lokales E-Mail Postfach verfügt. Dies geschieht unter Unix üblicherweise über die Auswertung der /etc/passwd Datei. Unter sendmail sind die Postfächer typischerweise als sog. Mailboxen unter /var/mail oder /var/spool/mail organisiert. D.h. alle Anwender haben in diesem allgemeinen Verzeichnis eine personalisierte Mailbox, in der die E-Mails als konkatinierte Datei aufbewahrt werden.

## E-Mail Clients

Der Zugriff auf die lokalen Postfächer geschieht entweder unter Unix über einen lokalen Mail User Agent (MUA) wie mail, emacs oder mutt, oder erfolgt von einem abgesetzten Client aus mittels des Protokolls POP3 (Post Office Protocol) bzw. IMAP4 (Internet Mail Access Protocol). Typische Remote User Agents (RUA) sind z.B. der Netscape Mailer, Eudora, und Microsoft Outlook. Der MUA/RUA ist letztlich für die korrekte Interpretation (und Erstellung) der E-Mail verantwortlich.

Bekanntlich kann eine E-Mail in die Bestandteile SMTP-Envelope, E-Mail-Header, RFC822-Body und den Attachments (BinHex, MIME, PGP, etc.) gegliedert werden. Während ein MUA den Aufbau des lokalen Postfachs kennen muss, ist dem RUA dessen Struktur egal, da der Zugriff hierauf über den POP3- bzw. IMPA-Server erfolgt. Als Sicherheitsmassnahme kommen hierbei die Angabe von Postfachinhaber und Passwort in Betracht, wobei diese streng zu trennen sind von System-Accounts. In der Regel wird z.B. ein POP3-Nutzer einen gleichnamigen System-Account besitzen; muss er aber nicht.

## Aufbau von qmail

Im Gegensatz zu sendmail entspricht qmail der oben dargestellten Arbeitsteilung in vorbildlicher Weise. Wie in Abbildung 2 dargestellt, wird jeder dieser Vorgänge über einen eigenen Unix-User verwirklicht, der im System nur mit minimalen Rechten verankert ist. In der Datei /etc/passwd wird qmail mit folgenden Usern und Unix-Shells eingerichtet:

- ❑ alias\*:7790:2108:qmail Alias  
User:/var/qmail/alias:/bin/true
- ❑ qmaild\*:7791:2108:qmail Daemon  
:/var/qmail:/bin/true
- ❑ qmail\*:7792:2108:qmail  
splogger/var/qmail:/bin/true
- ❑ qmailp\*:7793:2108:qmail-  
getpw:/var/qmail:/bin/true
- ❑ qmailq\*:7794:2107:qmail

Queue:/var/qmail:/bin/true

- ❑ qmail\*:7795:2107:qmail-  
remote:/var/qmail:/bin/true
- ❑ qmails\*:7796:2107:qmail-  
send:/var/qmail:/bin/true

qmail nutzt dabei eine Gruppe *qmail* (gid = 2107) und eine mit *nofiles* (gid = 2108). qmail verlangt nur an einer Stelle root-Rechte: Wenn nämlich die E-Mail vom zentralen MS in das lokale Postfach via qmail-local transferiert werden muss. Der Prozess qmail-spawn ruft hierbei den Auslieferungsagenten qmail-local mit den effektiven User-Rechten des Besitzers auf, wobei diese Besitzrechteänderung natürlich nur von root vorgenommen werden kann.

Aufgrund dieses minimalistischen Ansatzes reagiert qmail kritisch auf Fehler bei den Besitzrechten. Zwar können

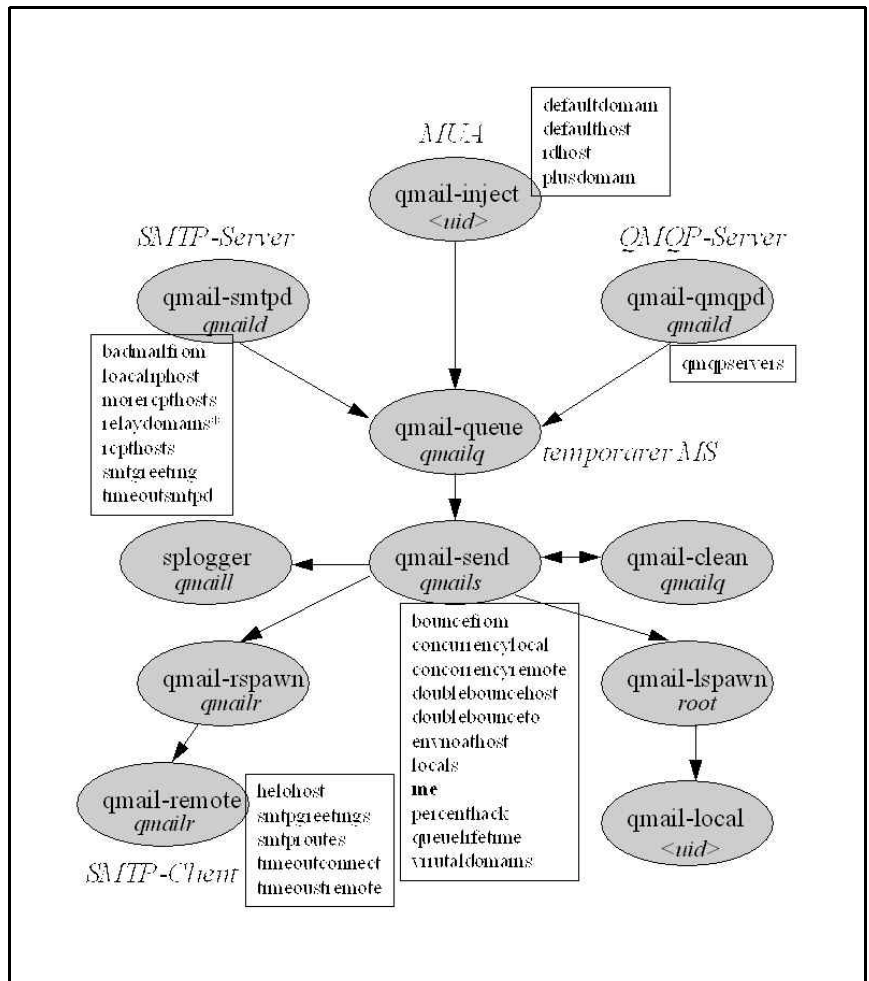


Abbildung 2: Die qmail-Prozesse.

Die Pfeile geben die Verarbeitungsrichtung an. Angegeben ist jeweils die qmail-id, unter der Prozess läuft sowie in den Kästen die zum Prozess gehörenden Kontrolldateien.

die oben dargestellten uids und gids zum Installationszeitpunkt beliebig gewählt werden; ändert man im Laufzeitsystem die Rechte an den diesen qmail-Usern zugeordneten Dateien (z.B. auch der Queue), führt dies unweigerlich zu Ausführungsfehlern.

## qmail-Entwicklung und -Umfang

qmail liegt heute in der Version 1.03 vor und wird als tarball oder aber als fertiges RPM installiert. Dan J. Bernstein hat 1995 die erste qmail Version 0.70 veröffentlicht und die aktuelle Version 1998 zur Verfügung gestellt. Seit dieser Zeit haben sich natürlich Ergänzungen des (E)SMTP-Standards ergeben, die durch die Gemeinschaft der qmail-Anwender in qmail per Patches eingepflegt werden. qmail ist zwar eine frei verfügbare, distributierbare und ergänzbare Software, bleibt jedoch (geistiges) Eigentum von D.J. Bernstein. Dies setzt enge Grenzen z.B. hinsichtlich von Ports zu anderen Unix-Derivaten.

Neben qmail ist D.J. Bernstein intensiv involviert der Entwicklung von UCSPI-Software [3] (Unix Client-/Server Programming Interface). Hierzu zählen vor allem der gerne mit qmail eingesetzte tcpserver, dnscache und andere Produkte. qmail bringt seinen eigenen Mailinglisten Manager mit: ezmlm (Easy Mailing List Manager, kurz Easy-Mail) [4]. In Form der Erweiterung ezmlm-idx ist dieser sehr leistungsfähig und verfügt zudem über Web-basierte Konfigurations- und Archiv-Funktionen.

qmail bringt von Hause aus einen ESMTP-Client und Server mit. Hinzu kommen entsprechende Funktionen für das von D.J. Bernstein entwickelte "Quick Mail Queuing"-Protokoll (QMQP). Als lokale E-Mail Auslieferungsagenten fungiert qmail-lokal sowie der qmail-pop3d-Server. qmail ist mittels eines Patches IPv6-fähig [5]. Wer will, kann sich neben qmail 1.03 aus folgendem Fundus bedienen:

fastforward: sendmail kompatible Alias-Verwaltung

IMPA4-, POP3-Server: vpopmail, qpopper

LDAP-Unterstützung

MySQL-Support

TLS/OpenSSL-Anbindung mittels STARTTLS-Kommandos

POP-before-SMTP und ATRN

tcpserver statt inetd oder xinetd

dnscache statt bind-client

ezmlm-idx Mailinglisten-Manager (statt Majordomo)

procmail Einbindung

Spam-Filter [6] mit RBL-Unterstützung

Virensanner Plug-In Scan4Virus für McAfee, Trend, AMAViS und andere

## qmail-Installation

Derzeit gibt es qmail als tarball und als Red Hat RPM. D.J. Bernstein verzichtet bei qmail auf den - für ihn unsicheren - stdio-Einsatz. qmail lässt sich auf allen bekannten Unix-Plattformen kompilieren und einsetzen.

Die einzelnen Schritte: Zunächst ist das tar-file qmail\_1-03\_tar.gz geeignet zu entpacken, z.B. unter /tmp. Es lohnt einen Blick in die Dokumente INSTALL\*, BLURB\* sowie FAQ zu werfen. Anschliessend empfiehlt es sich, mittels su nach root zu wechseln, das Verzeichnis /var/qmail anzulegen und die o.g. qmail-User im System einzurichten. Mit make setup check wird qmail kompiliert, die Binaries sowie alle Dateistrukturen erzeugt und ein erster rudimentärer Check durchgeführt. Hierbei wird auch die Existenz der qmail-User überprüft. Mittels des Skripts ./config wird bereits ein prinzipiell lauffähiges qmail-System konfiguriert. Jetzt beginnt die eigentliche Arbeit. Der Administrator sollte sich spätestens zu diesem Zeitpunkt folgende Punkte auf seine Checkliste geschrieben haben:

Über welchen Superdaemon (tcpserver, xinetd, inetd) soll qmail-

smtpd und qmail-pop3d in Zukunft gestartet werden?

Sollen die Anwender das Standard-Mailbox-Format oder das von D.J. Bernstein bevorzugte Maildir-Format bekommen (oder ggf. eine gemischte Konfiguration)?

Wie ist qmail in die System-Startup-Skripte einzugliedern?

Wie administrierte ich in Zukunft das qmail-System? Welche Syslog-Funktionen und Auswertungen sollen vorgenommen werden?

Die Antworten auf diese Fragen sind sehr individuell und qmail macht hier keine Vorgaben. D.J. Bernstein lehnt den inetd aus Performance-Gründen ab und empfiehlt seinen tcpserver. Gute Erfahrungen habe ich auch mit dem xinetd gemacht [7]. tcpserver und xinetd erlauben ein explizites Filtern von SMTP-Verbindungen per IP-Adresse. Ferner ermöglichen sie umfangreiche Logging-Funktionen. Während allerdings xinetd wie inetd die Server-Applikationen per Argumentenlisten aufruft, nutzt tcpserver Standard-Unix Pipe-Verfahren.

qmail-local muss wissen, wie es die E-Mails lokal ausliefern soll, d.h. entweder im Mailbox- (mbox) oder im Maildir-Format. In jedem Fall liegt das Postfach bei qmail im lokalen Heimatverzeichnis, also nicht wie bei sendmail unter /var/mail/<userid>, sondern unter \$HOME/mbox oder \$HOME/Maildir. Dieses Information wird im Startup-Skript von qmail eingetragen.

qmail installiert sich mit einer Reihe von vorkonfigurierten Startup-Skripten unter seinem Hauptverzeichnis /var/qmail. Hiervon braucht in der Regel nur eines ausgewählt werden. Unter SuSE-Linux muss das geeignete qmail-Startup-Skript nach /sbin/init.d kopiert werden bzw. ersetzt mittels eines Symlink das Standard sendmail-Skript. Dies ist natürlich vom jeweiligen Unix-System abhängig; bietet aber einem erfahrenen Systemadministrator keine Schwierigkeiten.

Unix-Systeme werden z.Z. noch

kaum mit qmail als Default-MTA ausgeliefert. Sofern bereits z.B. eine sendmail-Implementierung (ggf. auch das alte binmail) vorliegt, muss diese disabled werden. Oft sind sendmail-Aufrufe in MUAs integriert. Um deren Funktionalität zu gewährleisten, verfügt qmail über einen sendmail wrapper. Dieser ersetzt über einen Symlink die Standard sendmail-Datei.

Wesentlich anders stellt sich die Situation bei der Administration von qmail dar. D.J. Bernstein macht hierzu keine Aussagen. Meiner Erfahrung nach empfiehlt es sich aber, einen dedizierten qmail Administrations-Account (z.B. User qmailadmin in der Gruppe qmail) einzurichten, der für die qmail-Administration zuständig ist. Nicht nur, dass unter dessen \$HOME-Verzeichnis, das aktuell installierte qmail-System ausgepackt und evtl. gepatcht wird. Durch das Setzen der Environment-Variablen für die qmail man-pages und für die Executables ist auch eine komfortable Verwaltung möglich. Es empfiehlt sich, die qmail-Konfiguration mittels des Kommandos `/var/qmail/bin/qmail-showctl` in eine Datei umzuleiten, um die aktuellen Konfigurationsparameter festzuhalten. Unter diesem Account sind auch die notwendigen Skripte installiert, um die per splogger aufgezeichnete qmail-Aktivitäten im Syslog auszuwerten. Mittels des Skripts `qmailanalog` kann in jedem Fall eine genaue Auswertung der qmail-Aktivität erfolgen.

An dieser Stelle sind wir wieder bei einer kritischen Verzweigung. qmail vermeidet den Standard Syslog-Daemon, sondern verwendet über seine `splogger` ein unabhängiges Verfahren. D.J. Bernstein hat hierzu - wie oft - seine eigene Vorstellung. Der Syslog-Daemon scheint unter Last und unter den Bedingungen umfangreicher I/Os konzeptionelle Schwächen aufzuzeigen, die durch Nutzung entsprechender Applikationen aus dem Bündel der UCSPI-Software vermieden werden können. Wie aus Abbildung 2 hervorgeht, beinhaltet dies jedoch im wesentlichen den qmail-Sendeprozess. `qmail-smtpd` Aktivitäten werden in der Regel nicht mitgeschrieben bzw.

müssen über `tcpserver/xinetd` in den Maillog eingepflegt werden.

## Das qmail-Verzeichnis

qmail richtet sich unter im Verzeichnis `/var/qmail` mit einigen Unterverzeichnissen bequem ein:

- `alias` - reserviert für den qmail Alias-User
- `bin` - Ablage der qmail-Binaries
- `boot` - hier finden sich einige vorkonfigurierte Boot-Skripte
- `control` - Konfigurationsverzeichnis für qmail (Abbildung 2)
- `doc` - Installations- und andere Dokumente
- `man` - qmail man-pages
- `queue` - tiefverzweigtes qmail-Queue-Verzeichnis
- `users` - spezielle qmail-User, ausserhalb von `/etc/passwd`

Es ist zu beachten, dass diese Struktur nicht geändert werden sollte. qmail versteht sich als *systembezogener* Dienst. qmail initialisiert während der Installation speziell Dateien in seinem Queue-Verzeichnis. Diese können nicht zwischen den Systemen ausgetauscht werden. Ich empfehle auch immer, die qmail-Binaries auf dem lokalen System zu erzeugen; was auch für Perl-Skripte gilt. Erst nach erfolgreichem "make" weiss ich, dass eine ausführbare Datei auch wirklich läuft!

## qmail-Queue

Die Systemabhängigkeiten gelten umsomehr für die qmail-Queue. qmail legt hier nämlich temporäre Dateien unter ihrer inode-Kennung ab. Im Verlaufe des E-Mail-Verarbeitung "wandert" eine E-Mail - nach SMTP-Envelope und Nachricht in unterschiedliche Dateien getrennt - über verschiedene Verarbeitungsstadien in der Queue. Der Status der Verarbeitung drückt sich insbesondere durch ein "T" als Tag-Label im E-Mail-Envelope aus.

Die qmail-Queue arbeitet hocheffizi-

ent und stellt im Grunde genommen das eigentliche Kernelement von qmail dar. D.J. Bernstein hat grossen Aufwand in die Realisierung eines schnellen Verarbeitungsalgorithmus für temporäre Daten (Dateien) in einer Datenbank gesteckt [9]. Die qmail-Queue ist äusserst robust und braucht in der Regel auch keine Pflege, da Inkonsistenzen über `qmail-clean` beseitigt werden. Die Verknüpfung des (temporären) Dateinames einer in Verarbeitung befindlichen E-Mail mit dem inode, macht allerdings ein Backup der Queue nahezu unmöglich. Nahezu deshalb, weil natürlich findige qmail-Anwender entsprechende Hilfsprogramme entwickelt haben und zur Verfügung stellen.

## qmail-Konfiguration

Die modulare Arbeitsweise von qmail drückt sich auch in den in Abbildung 2 genannten Konfigurationsdateien aus. Diese gelten jeweils für das entsprechende qmail-Modul. Dies macht die Pflege einerseits flexibler, andererseits komplexer. Flexibilität wird erreicht, indem z.B. Änderungen an den Einstellungen der Daemon-Prozesse, wie z.B. `qmail-smtpd` "on-the-fly" wirksam sind. Komplexer ist es deshalb, weil durchaus widersprechende Einstellungen möglich sind.

So kann z.B. mittels der wichtigen Konfigurationsdatei "rcpthosts" festgehalten werden, für welche FQDN E-Mail-Adressen (RCPT TO:) sich `qmail-smtpd` verantwortlich sieht. Es versteht sich von selbst, dass hierunter auch der lokale Hostname selbst zu subsumieren ist (muss aber nicht). Dieser ist aber in der Regel in "locals" einzutragen; muss aber nicht. Dafür sorgt nämlich schon die absolut notwendige Datei "me" aus der qmail seinen eigenen Hostname bezieht, der natürlich unabhängig sein kann vom systemweiten hostname ! Keinesfalls darf jedoch in "locals" eine virtuelle Domäne stehen, für die qmail E-Mail zu empfangen hat. Diese muss in "virtualdomains" eingetragen sein und in - richtig ! - "rcpthosts" oder "morercpthosts". Hierbei ist der Systeme-

madadministrator angewiesen, in der modularen qmail-Arbeitsweise zu denken. Monolithische Konfigurationsdateien à la sendmail.cf haben bei qmail keine Entsprechung.

## DNS und IP-Adressen in qmail

Prinzipiell können in E-Mail-Adressen auch IP-Adressen vorkommen, z.B. <vorname.nachname@[hostip]>. Generell wird aber von den entsprechenden SMTP-RFCs ein FQDN-Lookup vorgeschrieben. SMTP verwendet bekanntlich folgende Adresssyntax im Envelope:

- Empfänger: RCPT TO:  
<user@domain.tld>
- Sender: MAIL FROM:  
<user@myhost.domain.tld>

Um eine E-Mail erfolgreich zustellen zu können (Sendeprozess), muss der jeweilige MTA folgendes überprüfen:

- Ist das Empfängersystem über einen entsprechenden MX-Eintrag als SMTP-MTA qualifiziert (DNS MX-Lookup)?
- Welche IP-Adresse besitzt das Empfängersystem (DNS A-Lookup)?

Im Gegenzug, das heisst beim Empfang einer E-Mail, hat das Zielsystem festzustellen:

- Bin ich als Zielsystem angesprochen (über die Auswertung des Domain-Teils von RCPT TO:)?
- Soll ich die E-Mail stellvertretend für meine Domäne bzw. für andere Zielsysteme/Domänen annehmen (wie dies bei qmail z.B. in "locals" und "virtualdomain" hinterlegt werden kann)?
- Soll ich vielleicht diese E-Mail an ein anderes Zielsystem weiterleiten (relays), falls z.B. eine SMTP-Route der Form  
<zielsystem@relay2@relay1>  
vorliegt?

Zusätzlich kann noch folgendes auftreten:

- Es kann beim Empfangsprozess eine optionale Überprüfung des Absenders vorgenommen werden, ob die Absenderadresse auf einen Rechner mit MX-Record zeigt.
- Beim Sendeprozess lässt der DNS-Lookup durch eine explizite SMTP-Route in der qmail-Konfigurationsdatei "smtproutes" umgehen.

qmail holt sich die IP-Adressen über Umgebungsvariablen, die ihm das Hilfsprogramm tcp-env bereit stellt. Der SMTP-Empfangsprozess qmail-smtpd ist mit tcp-env im Aufruf zu verknüpfen (z.B. beim xinetd). tcp-env kann auch noch optional das TCP/UDP-Protokoll ident aufrufen. Dies führt in einer Firewall-Konfiguration regelmässig zu Schwierigkeiten, da das ident-Port 113 in der Regel nicht freigeschaltet ist, und qmail dann bei der Verbindungsaufnahme (üblicherweise 60 Sekunden) warten muss, bis der ident-Dienst terminiert.

## E-Mail-Ablage

Es ist klar, dass das Maildir-Verfahren prinzipiell das geeignetere E-Mail Depot darstellt; allerdings unterstützen "alte" MUAs wie mail dieses Ablageformat nicht. Um dies zu umgehen bieten sich neuere E-Mail-Clients wie mutt [9] an. Andererseits ist qmail hier wieder so flexibel, dass pro User entschieden werden kann, in welcher Form seine E-Mail lokal abgelegt werden soll Entweder kann - on-the-fly - mittels des qmail-Hilfsprogramms maildir2mbox die in einem Maildir abgelegte E-Mail in eine Mailbox transferiert werden. Alternativ kann einem einzelnen Benutzer statisch - unabhängig von der systemweiten qmail-Einstellung - eine Mailbox zugewiesen werden. Dann ist allerdings der POP3-Zugriff via qmail-pop3d perdu. Für viele Anwender aber eine akzeptable Work-Around. Eines tut qmail aber in keinem Fall: E-Mail an root ist Tabu. Der root-User bekommt per qmail keine E-Mail zugesandt. root sollte nur per su genutzt werden und E-Mails an root sind an einen vordefinierten Alias-User umzuleiten.

Beim Einrichten neuer User im Unix-System ist darauf zu achten, dass diese möglichst automatisch das entsprechende Maildir bzw. die Mailbox eingerichtet bekommen. Hilfreich hierzu ist die Einbindung des qmail-Kommandos "maildirmake", dass allerdings nicht von root, sondern mit den effektiven User-Rechten aufzurufen ist. Ggf. sind die entsprechenden User-Templates (z.B. unter /etc/skel) anzupassen, da ansonsten prinzipiell eine sendmail-kompatible mbox unter /var/mail erzeugt wird. Alternativ empfiehlt es sich, die Standard-Unix "adduser" bzw. "useradd" Kommandos durch Shell-Skripte zu ersetzen, die dann das entsprechende "maildirmake" sowie die Erzeugung des geeigneten ".qmail"-Files im entsprechenden \$HOME-Verzeichnis mit beinhalten. Dies gilt besonders für die SuSE-Distribution, wo üblicherweise "yast" zum Einsatz kommt.

Der Zugriff auf die Postfächer wird über systemweite Environment-Variablen geregelt, die in den Profil-Skripten gesetzt sind. Nimmt ein lokaler E-Mail-Client dies nicht an, kann ggf. ein Symlink gesetzt werden. In jedem Fall sind bereits sendmail gefüllte Mbox-en zu übertragen. Mittels des Hilfsprogramms mbox2maildir, das qmail-Anwender bereit gestellt haben, lässt sich der bestehende alte Inhalt in das neue Format übertragen.

## Links:

- [1] sendmail:  
<http://www.sendmail.net/>
- [2] qmail: <http://www.qmail.org/>
- [3] UCSPI: <http://cr.yip.to/ucspi-tcp.html>
- [4] Ezmlm: <http://www.ezmlm.org/>
- [5] Fujiwara:  
<http://www.rcac.tdi.co.jp/fujiwara>
- [6] Spamcontrol:  
<http://www.fehcom.de/qmail.html>
- [7] Synack Communications:  
<http://synack.net/>
- [8] cdb: <http://cr.yip.to/cdb.html>
- [9] Mutt: <http://www.trash.net/~thomasb/mutt/>