

# Masqmail installieren und konfigurieren

Bernhard Walle  
<bernhard.walle@gmx.de>  
19. Januar 2005

*Masqmail* ist ein kompakter Mailserver, der speziell auf Einzelplatzrechner oder kleinere Netzwerke optimiert ist, die über keine dauerhafte Internetverbindung verfügen.

Folgender Artikel beschreibt die Installation und Konfiguration von *Masqmail* unter Linux. Es wird besonders auf die SuSE-Distribution eingegangen. Keinesfalls will dieser Artikel die bestehende Dokumentation ersetzen; vielmehr möchte ich sie ergänzen und auch eine deutschsprachige Anleitung zur Verfügung stellen.

Über Kommentare freue ich mich sehr, egal ob sie Lob, konstruktive Kritik oder Hinweise auf Fehler (egal ob inhaltliche, sprachliche oder einfach nur Rechtschreib- oder Tippfehler) beinhalten. Schreiben Sie mir einfach eine E-Mail!

Die Homepage von *Masqmail* befindet sich unter <http://masqmail.cx/masqmail/>. Hier kann auch die neueste Version heruntergeladen werden. Die Entwicklerversion 0.2.18 ist hinreichend stabil, um sie beispielsweise zu Hause einzusetzen<sup>1</sup>.

Ansonsten noch ein Hinweis: wenn Sie statt *Masqmail* lieber *Exim* verwenden (der Standard-MTA von Debian), finden Sie unter <http://www.linuxer.onlinehome.de/apps/exim.htm> eine sehr gute Anleitung wie das Zusammenspiel mit mehreren Usern und Accounts klappt.

## 1 Warum Masqmail?

### 1.1 Warum einen Mailserver?

Natürlich ist es auch in einer Linux-Umgebung möglich, ohne einen Mailserver auszukommen. Wenn man seine Mails beispielsweise mit *KMail*, *Netscape/Mozilla* oder *Sylpheed* – um nur einige populäre Programme zu nennen – bearbeitet, die die Kommunikation mit einem SMTP-Server beherrschen, braucht man keinen Mailserver auf seinem Rechner zu installieren.

Viele Linux-Programme setzen jedoch einen Mailserver bzw. das Programm `/usr/lib/sendmail` voraus, um Mails zu verschicken. Spätestens wenn man eigene Webanwendungen (CGI/Perl oder PHP) entwickelt, wird man um die Installation eines Mailservers nicht mehr herumkommen.

Außerdem ist ein Mailserver ein wichtiges Glied in der Kette *Sendmail* – *Fetchmail* – *Procmail* – *Mutt*, die so ziemlich jede Möglichkeit in Bezug auf Mails eröffnet. Vor allem auf die beispiellose Konfigurierbarkeit von *Mutt* und auf die überwältigenden Filtermöglichkeiten von *Procmail*<sup>2</sup> möchte ich nicht mehr verzichten.

---

<sup>1</sup> Wenn jede Software, die sich *stabil* nennt, so zuverlässig wie *Masqmail* arbeiten würde wäre ich hochzufrieden. Ich denke dabei nicht ausschließlich an Software aus Redmond sondern auch an Linux-Desktops, die ständig neue Features bekommen aber nie stabil werden – aber das ist ein anderes Thema.

<sup>2</sup> Dieses Programm eignet sich auch hervorragend zum automatischen Filtern von unerwünschten Werbemails. Ein guter Spamfilter kann unter <http://www.belwue.de/support/spamblock.html> heruntergeladen werden. Ein weiterer guter Filter, der zwar nicht auf Procmail-Regeln basiert aber über Procmail aufgerufen wird wäre *Spamassassin* (<http://www.spamassassin.org>).

## 1.2 Warum Masqmail?

Die Frage ist jetzt nun, warum man unter den Vielzahl der existierenden Mailserver ausgerechnet *Masqmail* einsetzen sollte. Die bekannten „großen“ Mailserver sind auf den Einsatz auf „richtigen“ Servern optimiert worden, d. h. auf Computern, die dauernd ans Internet angeschlossen sind und eine große Zahl an Benutzern und Mails zu verarbeiten haben.

Nun setzen aber immer mehr Menschen Linux auf dem Desktop ein und wollen auf der einen Seite nicht auf die oben angesprochene Flexibilität verzichten, auf der anderen Seite gibt es aber spezielle Anforderungen an einen Mailserver für solch einen Einsatz.

Folgende Gründe sprechen für *Masqmail*:

- *Masqmail* beherrscht einerseits die direkte Versendung, ist aber auf den Einsatz von SMTP-Relays (d. h. Weiterleitung der Mails an den SMTP-Server des Providers, wie dies auch bei Verwendung SMTP-fähiger Mailprogramme geschehen würde) optimiert. Es können mehrere SMTP-Relays konfiguriert werden, und zwar in Abhängigkeit vom Provider, bei dem man gerade eingewählt ist, oder aber vom Absender. Letzteres bietet eine flexible Möglichkeit, wenn man mehrere Mailadressen besitzt.
- *Masqmail* unterstützt SMTP-AUTH (SMTP mit Authentifizierung). Dieses Verfahren setzen immer mehr Provider und Freemailanbieter ein, um sich vor Spam zu schützen und die Notlösung SMTP-after-POP<sup>3</sup> langfristig ablösen.
- *Masqmail* stellt bei der Einlieferung von Mails fest, ob der Computer gerade online ist oder nicht. Ist der Computer online, so wird die Mail sofort ausgeliefert. Andernfalls wird die Mail in einer Warteschlange gestellt und ausgeliefert, wenn man sich das nächste mal beim Provider einwählt.
- *Masqmail* erlaubt es, Mails via POP3 beim Provider abzuholen. Ich werde nachfolgend dieses Verfahren beschreiben (im Unterschied zu früheren Versionen dieses Artikels), weil es eine sehr einfache Möglichkeit bietet, das SMTPafterPOPVerfahren zu nutzen.
- *Masqmail* kann sowohl direkt als auch über die Verwendung eines MDA (Mail Delivery Agent), beispielsweise *Procmal*, ausliefern. Diese Möglichkeit bietet aber so ziemlich jeder Mailserver.
- *Masqmail* ist einfach konfigurierbar. Während *Sendmail* für einen „einfachen Benutzer“ nur über spezielle Tools konfiguriert werden kann, ist *Masqmail* direkt über das Editieren weniger Konfigurationsdateien ohne große Probleme konfigurierbar. Nachfolgend werden die wichtigsten Konfigurationsmöglichkeiten beschrieben.
- *Masqmail* ist relativ klein und schnell. Auf einem normalen Einzelplatzcomputer lässt sich damit wesentlich schneller arbeiten als mit einem großen Mailserver wie *Sendmail*, *QMail*, *Exim* oder *Postfix*.
- *Masqmail* eignet sich aber ausdrücklich nicht als Ersatz der genannten großen Mailserver in einem mittelgroßen Netzwerk und für Computer, die dauerhaft ans Internet angeschlossen sind (mit Standleitung und fester IP-Adresse). Auf dies weist auch der Autor des Programmes auf

---

<sup>3</sup> Bei dieser Lösung wird der SMTP-Server meist für eine bestimmte IP-Adresse und einen bestimmten Absender eine gewisse Zeit freigeschaltet, nachdem man über das POP3-Protokoll seine Mails abgeholt hat. Vor allem bei Freemail-Anbietern wie GMX oder WEB.DE ist dieses Verfahren sehr verbreitet.

seiner Homepage ausdrücklich hin! Natürlich heißt dies nicht, dass Nutzer mit einer Flatrate auf Masqmail verzichten sollten.

## 2 Maßnahmen vor der Installation

### 2.1 Sendmail-Startskript

Soll *Masqmail* als lokaler Mailserver eingesetzt werden, an den Programme über den Port 25 ausliefern können sollen, dann muss es als Daemon (Hintergrundprozess) bei jedem Booten des Rechners gestartet und beim Herunterfahren wieder beendet werden. Dazu sind bestimmte Skripten (normalerweise in `/sbin/init.d`, `/etc/init.d` oder `/etc/rc.d` je nach Distribution und Version) erforderlich.

Nutzt man *Masqmail* auch zum Abholen seiner Post und verwendet man ein Mailprogramm, welches ohnehin kein SMTP kann sondern `/usr/lib/sendmail` (bzw. hier *Masqmail*) zum Verschicken seiner Mails verwendet, kann man auf das Starten von Masqmail als Daemon auch gut verzichten. Einerseits verzichtet man auf einen Hintergrundprozess andererseits erspart es Arbeit und verhindert auch eine potentielle Sicherheitslücke. Ich empfehle daher, *Masqmail* nicht als Daemon zu starten - wiederum im Gegensatz zu früheren Versionen dieses Artikels. Sollten Sie sich entscheiden, *Masqmail* nicht als Daemon laufen lassen zu wollen, müssen Sie trotzdem weiterlesen, weil es erforderlich sein kann das automatische Starten beim Hochfahren des Systems zu deaktivieren.

*Masqmail* wurde als Ersatz für *Sendmail* konzipiert und versteht daher die gleichen Kommandozeilenparameter - auch wenn noch nicht alle implementiert wurden und einige speziell für *Masqmail* hinzugefügt worden sind. Somit eignet sich das *Sendmail*-Startskript am besten für diese Aufgabe. Am einfachsten ist es somit, wenn zuvor *Sendmail* installiert wird (ist bei so ziemlich jeder Distribution enthalten) und dafür gesorgt wird, dass es bei jedem Start geladen wird. Wie das geht, steht im Handbuch der jeweiligen Distribution.

Bei SuSE vor 8.0 ist dazu in der Konfigurationsdatei `/etc/rc.config` die Variable `SMTP` auf den Wert `yes` zu setzen. Für spätere SuSE-Versionen verwendet man dazu den *Runlevel-Editor*, der über das *YaST2-Kontrollzentrum* aufgerufen wird. Hier wählt man unter *Runlevel-Eigenschaften* die Zeile `sendmail` aus und achtet darauf, dass es in den Runleveln 5 (Multiuser mit graphischer Oberfläche) und 3 (Multiuser mit Netzwerk) gestartet wird.

Außerdem muss noch die Variable `SENDMAIL_ARGS` auf den Wert `-bd -odb -q5m` gesetzt werden. Die Variable befindet sich ab SuSE 8.0 in der Datei `/etc/sysconfig/sendmail`, bei früheren Versionen der SuSE-Distribution in `/etc/rc.config.d/sendmail.rc.config`. Das `5m` steht für fünf Minuten, d. h. alle fünf Minuten wird die Warteschlange angearbeitet und lokale Mails werden verschickt. Normalerweise werden die Mails aber sowieso sofort verschickt, so dass diese Zeitspanne egal ist.

### 2.2 Benutzer und Gruppe anlegen

*Masqmail* arbeitet unter dem Usernamen `mail` und der Gruppe `trusted`. Zunächst sollte überprüft werden, ob die Benutzer bereits existieren. Bei einer Standardinstallation von SuSE 8.1 ist dies beispielsweise der Fall. Folgende Befehle führen diese Überprüfung durch:<sup>4</sup>

---

<sup>4</sup> # steht unter Unix für den Prompt von `root`, \$ hingegen steht für den Prompt eines beliebigen anderen Benutzers. Diese Konvention wird hier durchgehend verwendet und ist auch in anderen Dokumentationen üblich, auch wenn der

```
# grep ^mail /etc/passwd
mail:x:8:12:Mailer daemon:/var/spool/clientmqueue:/bin/false

# grep ^trusted /etc/group
trusted:x:42:bwalle
```

Falls eine ähnliche Ausgabe wie die oben gezeigte erfolgt, existiert der Benutzer bzw. die Gruppe. Nur wenn der Befehl nichts ausgibt ist davon auszugehen, dass Benutzer und/oder Gruppe angelegt werden müssen. Das Anlegen kann natürlich über *YaST* oder *YaST2* geschehen. Es reicht aber aus, folgende Kommandos auszuführen:

```
# groupadd -g 42 trusted
# useradd -u 42 -g 42 -d / -s /bin/sh -c "Mail Transfer Agent" mail
```

Anschließend sollten alle Benutzer, die Mails über *Masqmail* versenden können sollen und auch den Absender verändern dürfen, dieser neuen Gruppe hinzugefügt werden. Dies kann wieder über die Konfigurationstools der jeweiligen Distribution geschehen. Alternativ kann die Datei */etc/group* bearbeitet werden. Auch das Programm *usermod* eignet sich – allerdings müssen hierfür zunächst alle Gruppen herausgefunden werden, in denen der betreffende Benutzer Mitglied ist.

Beispiel:

```
$ groups bwalle
bwalle : users disk uucp dialout audio video

# usermod -G users,disk,uucp,dialout,audio,video,trusted bwalle
```

Es werden also alle Gruppen in der Liste nach *-G* übernommen und zusätzlich *trusted* angefügt. Wird nur die neue Gruppe angefügt, so ist der Benutzer anschließend ausschließlich Mitglied in dieser Gruppe, was zu unangenehmen Folgen führen kann!

Prinzipiell ist es möglich andere Benutzer- und Gruppennamen zu verwenden. Dies muss aber anschließend beim Ausführen von *configure* angegeben werden, z. B. *./configure --with-user=USER --with-group=GROUP*.

### 3 Masqmail kompilieren und installieren

Als erstes muss das Paket ausgepackt und in das neu erstellte Verzeichnis gewechselt werden:

```
$ tar xvfz masqmail-0.2.18.tar.gz
$ cd masqmail-0.2.18/
```

Nun wird das Programm *configure* ausgeführt, um eine Anpassung an das System vorzunehmen:

---

Standardprompt Ihrer Distribution möglicherweise anders aussieht.

```
$ ./configure --enable-auth
```

Der Parameter `--enable-auth` wird empfohlen, damit das SMTP-AUTH-Protokoll verwendet werden kann.

Nachdem das Skript ohne Fehlermeldungen ausgeführt wurde, wird die Kompilation gestartet. Sind Fehler aufgetreten, so sind höchstwahrscheinlich Programme oder Bibliotheken nachzuinstallieren. Beispielsweise benötigt *Masqmail* die *glib*-Bibliotheken, die einige wichtige Datenstrukturen in C bereitstellen, d. h. die Pakete *glib* und *glib-devel* müssen installiert sein<sup>5</sup>. Anschließend ist der Schritt zu wiederholen; vorher ist allerdings ggf. die erstellte Datei `config.cache` zu löschen.

```
$ make
```

Nachdem nun auch dieser Schritt ohne Fehlermeldungen erfolgt ist, wird das Programm schließlich auf dem Rechner installiert:

```
# make install
```

Abschließend ist es erforderlich, dass der alte MTA (Mail Transfer Agent; also *Sendmail*) gesichert wird und ein symbolischer Link auf `/usr/sbin/sendmail` und evtl. auch `/usr/lib/sendmail` angelegt wird. Normalerweise ist aber `/usr/lib/sendmail` bereits ein symbolischer Link auf `/usr/sbin/sendmail`, so dass dieser Schritt entfällt.

```
# mv /usr/sbin/sendmail /usr/sbin/sendmail.orig
# ln -s /usr/sbin/masqmail /usr/sbin/sendmail
# readlink /usr/lib/sendmail
/usr/sbin/sendmail
```

Nur wenn das letzte Kommando `/usr/sbin/sendmail` ausgibt, ist der Link richtig gesetzt. Wenn es auf Ihrem Rechner kein *readlink* gibt – dieses Programm gehört nämlich zum Paket *tetex* –, so kann der Link auch mit `ls -l` überprüft werden.

Abschließend muss der alte MTA zunächst als Daemon beendet werden, damit später (nach erfolgter Konfiguration) *Masqmail* gestartet werden kann:

```
# rcsendmail stop
Shutting down SMTP port: done
```

Je nach eingesetzter Linux-Distribution kann sich der Schritt unterscheiden.

Außerdem müssen noch für *Procmail* andere Rechte gesetzt werden. Der Grund liegt darin, dass unter *Sendmail* das Programm *Procmail* mit Root-Rechten aufgerufen wurde (*Sendmail* läuft mit Root-Rechten); dies ist jetzt nicht mehr der Fall.

---

<sup>5</sup> Je nach Linux-Distribution können sich die Paketnamen unterscheiden. Beispielsweise könnten die Headerdateien auch direkt in *glib* enthalten sein so dass kein Paket namens *glib-devel* existiert oder letzteres könnte *glibd* heißen (so der Fall bei älteren SuSE-Distributionen).

```
# chmod 6755 /usr/bin/procmail
```

Falls Sie SuSE verwenden, sollte noch folgender Eintrag in die Datei `/etc/permissions.local` vorgenommen werden. Damit wird verhindert, dass *SuSEconfig* bzw. *Check* die Rechte wieder zurücksetzt:

```
/usr/bin/procmail      root.root      6755
```

## 4 Masqmail konfigurieren

### 4.1 Beispielkonfigurationen kopieren

Im Unterverzeichnis `examples` befinden sich Beispielskonfigurationsdateien. Diese sind allerdings unvollständig und teilweise veraltet. Deshalb habe ich mir die Mühe gemacht, eine komplette Beispielkonfiguration aus meiner eigenen Konfiguration zu erstellen und dazu jeweils die erklärenden Abschnitte aus der (natürlich vollständigen und im übrigen sehr guten) Manualpage von *Masqmail* kopiert. Sie können sich diese als Paket unter <http://www.bwalle.de/config/masqmail-config.tar.gz> herunterladen.

Zunächst sollten Sie alle evtl. bestehenden alten Konfigurationsdateien von *Masqmail* löschen, dann das Archiv herunterladen, auspacken und anschließend die Dateien dorthin kopieren:

```
$ cd /tmp
$ wget http://www.bwalle.de/config/masqmail-config.tar.gz
$ tar xvfz masqmail-config.tar.gz
# rm -r /etc/masqmail
# mkdir /etc/masqmail
# cp /tmp/masqmail/* /etc/masqmail
# cd /etc/masqmail
# sh chperm
```

Der letzte Befehl ruft einen im Paket enthaltenen Dreizeiler auf, der automatisch die Rechte für die Konfigurationsdateien richtig setzt – und zwar sehr restriktiv, so dass nur *root* die Dateien lesen kann, was auf einem Multiusersystem verhindert, dass jeder Benutzer alle Mailpaßwörter lesen kann. Wenn Sie nicht alleine auf Ihrem System arbeiten ist dies sehr wichtig; das Skript sollte also nach jedem Anlegen einer Konfigurationsdatei in diesem Verzeichnis aufgerufen werden – dann reicht es allerdings, `./chperm` aufzurufen, da beim ersten Aufrufen das Skript auf „ausführbar“ gesetzt wird.

### 4.2 Globale Konfiguration

Die globale Konfiguration enthält alle Werte, die für das ganze System gelten. Sie wird in der Datei `/etc/masqmail/masqmail.conf` vorgenommen. Eine ausführliche Dokumentation zu allen Variablen dieser Datei findet man in der Manpage *masqmail.conf(5)*, die mit dem Kommando `man 5 masqmail.conf` aufgerufen werden kann.

Zunächst muss der Hostname gesetzt werden. Um diesen herauszufinden, führt man am besten `hostname -f` aus. Das Ergebnis trägt man in die Konfigurationsdatei ein (bzw. man ändert den Wert in der Vorlage, die man eben kopiert hat):

```
host_name = "hugo.local"
```

Außerdem benötigt *Masqmail* eine Liste aller Hosts, die als lokal zu betrachten sind. Normalerweise ist das das Ergebnis von `hostname -f`, `hostname` und noch `localhost`:

```
local_hosts="hugo;hugo.local;localhost"
```

Unter `local_nets` trägt man ebenfalls den Hostnamen ein, am Anfang mit einem Stern, gefolgt von einem Punkt. Die Variable erweitert die `local_hosts`-Variable und macht v. a. bei Netzwerken Sinn. Wichtig ist, dass alles, was nicht in `local_hosts` und `local_nets` enthalten ist, als nicht lokal betrachtet wird. Nur Mails an diese „Domains“ werden als lokal betrachtet.

```
local_nets="*.hugo.local"
```

Als nächstes muss *Masqmail* wissen, welche Verbindungen er akzeptieren soll. Dazu gehört auch der Port (für SMTP normalerweise immer 25). Wieder ist die Ausgabe von `hostname` erforderlich:

```
listen_addresses="hugo:25;localhost:25"
```

Die folgenden Variablen können unverändert bleiben:

```
spool_dir="/var/spool/masqmail"  
mail_dir="/var/spool/mail"  
use_syslog=false  
log_dir="/var/log/masqmail"  
do_queue=false
```

Im Folgenden wird *Masqmail* mitgeteilt, wie es erkennen soll, ob man gerade online ist oder nicht. Ich empfehle, dies mit einer temporären Datei zu machen, die angelegt wird, wenn man online geht und wieder gelöscht wird, wenn die Verbindung zum Provider getrennt wird (dazu später mehr). Folgende Einstellungen sind dazu notwendig:

```
online_detect=file  
online_file="/tmp/connect_route"
```

Schließlich soll *Masqmail* für die Zusammenarbeit mit *Procmail* konfiguriert werden. Dazu sind folgende Einstellungen erforderlich:

```
mbox_default="mda"  
mda="/usr/bin/procmail -Y -d ${rcpt_local}"  
alias_file = "/etc/aliases"  
alias_local_caseless = true
```

Als letztes sind die \*.route-Dateien einzutragen. Für jeden SMTP-Server (ob vom Provider oder von Freemail Providern wie GMX) ist dazu eine eigene Datei erforderlich. Masqmail eignet sich auch, um mehrere Provider bei der Einwahl zu unterscheiden. Im Folgenden wähle ich als Beispielprovider T\_Online. Ich werde nur einen Provider aber mehrere Routen konfigurieren. Ich wähle mich also immer über T-Online ein, verwende aber mehrere (zwei) SMTP-Server zum Verwenden meiner Mails. Statt T\_Online kann jede andere Zeichenkette (am besten ohne Leerzeichen) verwendet werden; es ist auch egal, wie der Provider jetzt tatsächlich heißt.

Die Konfiguration eignet sich auch, wenn man sich zwar tatsächlich über mehrere Provider einwählt, trotzdem aber immer die gleiche Mailkonfiguration verwenden möchte. Natürlich kann man auch nur eine Route angeben.

```
connect_route.T_Online = "/etc/masqmail/GMX.route;/etc/masqmail/T_Online.route"
```

Außerdem sollen die Mails auch abgeholt werden. Die Konfiguration kann wieder in Abhängigkeit vom Provider erfolgen, bei dem man gerade eingewählt ist:

```
online_gets.T_Online = "/etc/masqmail/GMX.route;/etc/masqmail/T_Online.get"
```

Somit ist die globale Konfiguration beendet. Der MTA kann jetzt gestartet werden. Es ist aber sinnvoller, zuerst die eingetragenen Route-Dateien zu erstellen und zu editieren.

### 4.3 Route-Dateien konfigurieren

Wie oben bereits beschrieben wird für jeden SMTP-Server eine eigene .route-Datei benötigt, die auch in der globalen Konfigurationsdatei eingetragen werden muss.

In meiner Beispielformatung werde ich den Server in Abhängigkeit vom Absender festlegen. Wenn ich mit meiner GMX-Adresse Mails verschicke, verwende ich den Server mail.gmx.net, bei Mails mit meiner T-Online-Adresse den Server smtp.t-online.de.

Damit *Masqmail* die verschiedenen Absender unterscheiden kann, muss der sog. Return-Path richtig gesetzt sein. Bei seinem Mailprogramm muss man einfach den From:-Header richtig setzen und es außerdem anweisen, dass auch der Envelope-From richtig gesetzt werden soll. Bei *Mutt* setzt man dazu die Variable `envelope_from` in der Konfigurationsdatei `$HOME/.muttrc` oder auch `$HOME/.mutt/muttrc`. Bei Programmen, die nicht über *Sendmail* sondern über SMTP versenden, sollte dies automatisch geschehen. Ansonsten weisen Sie Ihr Mailprogramm an, beim Versenden mit *Sendmail* den `-f`-Parameter zu benutzen.

Als erstes die kommentierte Konfigurationsdatei `/etc/mailsqmail/GMX.route`:

```

# Das Protokoll ist natuerlich SMTP
protocol = "smtp"

# Hier wird der Mailserver eingetragen (Port 25 ist Standard bei
# SMTP und sollte unveraendert bleiben)
mail_host = "mail.gmx.net:25"

# Nun werden die Absenderadressen angegeben, bei denen ueber diesen
# Server Mails verschickt werden sollen
allowed_return_paths = "*@gmx.de"

# Nun folgt der Authentifizierungsmechanismus
auth_name = "cram-md5"

# Username und Passwort bei SMTP-Auth
auth_login = "2363897"
auth_secret = "SECRET"

# Wenn SMTP-after-POP verwendet wird, dann hier den Pfad zur zugehoerigen
# Konfigurationsdatei eintragen und die Kommentarzeichen entfernen
##pop3_login = "/etc/masqmail/GMX.get"

```

Die Konfiguration für T-Online-Server ist ähnlich, nur dass hier keine Authentifizierung stattfindet (diese erfolgt bereits durch die Einwahl) und daher die drei Angaben `auth_name`, `auth_login` und `auth_secret` fehlen.

Weitere Optionen findet man in der Manpage *masqmail.route(5)*.

#### 4.4 Get-Dateien konfigurieren

Das Abholen der Post über POP3 wird über die `.get`-Dateien konfiguriert. Für jeden Server ist dabei genauso wie bei den `.route`-Dateien eine eigene Datei erforderlich.

Hier wieder die kommentierte Beispielkonfiguration für GMX durch die Datei `/etc/masqmail/GMX.get`:

```

# Das Protokoll ist natuerlich POP3
protocol = "pop3"

# Hier der Mailserver
server = "pop.gmx.net"

# Dann Benutzername und Passwort
user = "2360897"
pass = "SECRET"

# Lokale Adresse, an die ausgeliefert werden soll (also der Benutzer,
# der die Post erhalten soll)
address = "bwall@localhost"

```

Die Konfiguration für T-Online erfolgt natürlich analog dazu.

Weitere Informationen finden Sie in der Manpage *masqmail.get(5)*.

Falls Sie sich dafür entschieden haben, *Masqmail* als Daemon zu starten, dann ist jetzt der richtige Zeitpunkt, dies erstmalig durchzuführen, ansonsten überspringen Sie diesen Schritt einfach:

```
# rcsendmail start
  Initializing SMTP port. (sendmail)                               done
```

#### 4.5 Internetwahl konfigurieren

Dieser Schritt ist erforderlich, damit alle Mails verschickt werden, wenn man sich beim Provider einwählt. Außerdem muss die temporäre Datei angelegt und gelöscht werden. Nur so kann *Masqmail* die Mails sofort verschicken, wenn man gerade online ist.

Normalerweise sind die Einwahlskripten in */etc/ppp* so konfiguriert, dass keine Modifikationen in den Skripten selbst vorgenommen werden müssen sondern benutzerspezifische Befehle in den Dateien *ip-up.local* bzw. *ip-down.local* eingetragen werden können. Lesen Sie ggf. das Handbuch Ihrer Linux-Distribution wie vorgegangen werden muss.

Dann trägt man in die Datei */etc/ppp/ip-up.local* (es kann sein, dass diese Datei noch nicht existiert) folgendes ein:

```
# Onlinestatus setzen
echo -n T_Online > /tmp/connect_route
chmod 644 /tmp/connect_route

# Queue abarbeiten
/usr/sbin/masqmail -go T_Online -qo T_Online

# als Daemon starten
/usr/sbin/masqmail -go 5m T_Online
```

Beachten Sie, dass das *T\_Online* mit dem übereinstimmen muss, das in der Datei */etc/masqmail/masqmail.conf* eingetragen wurde.

Zunächst wird die temporäre Datei, die *Masqmail* zum Feststellen des Onlinestatus benötigt, angelegt und mit passenden Rechten versehen. Schließlich wird die Queue abgearbeitet, d. h. es werden neue Mails geholt und die Post, die sich während der Offlinezeit in der Warteschlange angesammelt hat, wird verschickt. Schließlich starten wir *Masqmail* als Daemon<sup>6</sup>, so dass alle fünf Minuten nach neuer Post geschaut wird. Statt *5m* können Sie auch eine andere Zeitspanne eintragen. Wenn Sie nur einmal Post holen möchten, wenn Sie online gehen, brauchen Sie diese Zeile überhaupt nicht.

Schließlich legt man die Datei */etc/ppp/ip-down.local* mit folgendem Inhalt an:

---

<sup>6</sup> Dies hat nichts damit zu tun, was vorher mit „als Daemon starten“ bezeichnet wurde. Vorher ging es darum, *Masqmail* als Mailserver laufen zu lassen der an Post 25 lauscht. Hier soll einfach alle fünf Minuten Post abgeholt werden.

```
rm /tmp/connect_route
killall masqmail
```

Dadurch wird die Datei für die Erkennung der Onlineverbindung gelöscht – sie besteht ja nicht mehr – und gleichzeitig wird der *Masqmail*-Daemon, der alle fünf Minuten nach neuen Mails schaut, beendet.

Nun muss noch sichergestellt werden, dass *ip-up.local* und *ip-down.local* die richtigen Rechte besitzen, d. h. sie müssen ausführbar sein (sonst werden sie ggf. nicht abgearbeitet):

```
chmod 755 ip-up.local
chmod 755 ip-down.local
```

Schließlich ist die eigentliche Konfiguration von *Masqmail* beendet!

## 5 Procmail

*Procmail* bietet viele Möglichkeiten – sehr lehrreich sind die Manpages *procmail(1)*, *procmailrc(5)* und *procmailx(5)*.

Für's erste reicht eine einfache *\$HOME/.procmailrc* mit folgendem Inhalt:

```
PATH=/bin:/usr/bin:
MAILDIR=$HOME/Mail
DEFAULT=$MAILDIR/inbox
LOGFILE=$HOME/.procmail.log
```

## 6 Sonstiges

Der Einfachheit halber sollte man dafür sorgen, dass die Mails an Root an den User, unter dem man normalerweise gerade arbeitet, weitergeleitet werden. Dazu ist in der Datei */etc/mail/aliases* folgender Eintrag vorzunehmen:

```
root:          bwallo
```

In manchen Distributionen (wie SuSE 8.1) lässt sich eine entsprechende Option auch schon direkt bei der Installation einrichten.