Ubuntu: Postfix als Mailrelay

Eine Kurzanleitungen, um einen relayfähigen Mailoutserver zu bauen. Es handelt sich um einen Mailserver mit virtuellen Postfächern und Benutzern, d.h. es werden keine Systemuser benutzt. Es soll folgendes leisten können:

- Ausgehende Mails auf Viren überprüfen
- Loginverzögerung (Anvil)
- SMTP über Authentifizierung (Sasl über sasldb)
- Virtual users and domains
- Chroot
- DKIM-Signierung ausgehender Mails

Postfix-Installation und Konfiguration

```
# aptitude install postfix postfix-doc
```

Dann implementieren wir die Postfix-Programmkonfiguration. Zuvor definieren wir den „Mailname“, da Postfix und andere Programme darauf zugreifen. Ich setze einfach die Stammdomäne ein:

/etc/mailname

```
meine-maildomain.de
```

Definition
The file `/etc/mailname` is a plain ASCII configuration file, which on a Debian system contains the visible mail name of the system. It is used by many different programs, usually programs that wish to send or relay mail, and need to know the name of the system.

The file contains only one line describing the fully qualified domain name that the program wishing to get the mail name should use (that is, **everything after the @**).

/etc/postfix/main.cf

```
# See /usr/share/postfix/main.cf.dist for a commented, more complete
version
# Debian specific:  Specifying a file name will cause the first
# line of that file to be used as the name.  The Debian default
# is /etc/mailname.

myorigin = /etc/mailname
```

```
### Gerneral settings

#smtpd_banner = ESMTP $mail_name (Ubuntu)
# Hide the OS giving more security
smtpd_banner = ESMTP $mail_name
biff = no

# appending .domain is the MUA's job.
append_dot_mydomain = no

# Uncomment the next line to generate "delayed mail" warnings
delay_warning_time = 4h

readme_directory = /usr/share/doc/postfix


# TLS parameters
smtpd_tls_cert_file=/etc/ssl/certs/my-cert.pem
smtpd_tls_key_file=/etc/ssl/private/my-key.key
smtpd_tls_CAfile=/etc/ssl/sub.class1.server.ca.pem
smtpd_use_tls=yes
#smtp_tls_note_starttls_offer = yes
smtpd_tls_loglevel = 1
smtpd_tls_received_header = yes
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom
tls_random_prng_update_period = 3600s

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc package
for
# information on enabling SSL in the smtp client.


### System settings

myhostname = smtp.meine-maildomain.de
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
mydestination = hostname.mydomain.de, smtp.mydomain.de, localhost
relayhost =
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
html_directory = /usr/share/doc/postfix/html


## Outgoing mail settings
```

```
smtp_bind_address = 190.180.170.61
smtp_helo_name = smtp1.meine-maildomain.de
transport_maps = hash:/etc/postfix/transport


## Incoming mail settings

smtpd_reject_unlisted_sender = yes
smtpd_helo_required = yes
message_size_limit = 102400000


## Auth SASL settings

smtpd_sasl_auth_enable = yes
smtpd_sasl_type = cyrus
smtpd_sasl_path = smtpd
broken_sasl_auth_clients = yes
smtp_sasl_security_options = noanonymous
smtpd_sasl_local_domain =
smtpd_sasl_exceptions_networks = $mynetworks
smtpd_sender_login_maps = hash:/etc/postfix/sender_logins

smtpd_recipient_restrictions =
        check_recipient_access btree:/etc/postfix/access_recipient-rfc,
        reject_invalid_helo_hostname,
        reject_unauth_pipelining,
        reject_non_fqdn_sender,
        reject_non_fqdn_recipient,
        reject_unknown_sender_domain,
        reject_unknown_recipient_domain,
        permit_sasl_authenticated,
        permit_mynetworks,
        #reject_rbl_client zen.spamhaus.org,
        #reject_rbl_client ix.dnsbl.manitu.net,
        reject_unauth_destination,
        permit

smtpd_sender_restrictions =
  check_sender_access regexp:/etc/postfix/tag_as_originating.re
  permit_mynetworks
  permit_sasl_authenticated
  permit_tls_clientcerts
# For other mail use amavis filtering on port 10024 (skips DKIM
signing)
  check_sender_access regexp:/etc/postfix/tag_as_foreign.re


## Anvil

anvil_status_update_time = 1m
```

```
anvil_rate_time_unit = 10s
smtpd_client_event_limit_exceptions = 127.0.0.1
smtpd_client_connection_rate_limit = 5
smtpd_client_connection_count_limit = 15
#smtpd_client_message_rate_limit = 10
```

Im Anschluss erfolgt die Postfix-Dienstkonfiguration. Sie enthält spezielle Konfigurationen, um das Zusammenspiel zwischen Amavisd-new und Postfix herzustellen. Des Weiteren werden unterschiedliche Relay-IP-Adressen definiert sowie eine eigene IP für eingehende Mails.

Postfix läuft unter Ubuntu bereits in einem **Chroot**.

/etc/postfix/master.cf

```
[...]
pickup    fifo  n       -       -       60      1       pickup
        -o content_filter=
        -o receive_override_options=no_header_body_checks
[...]

# Incoming mail smtp.mydomain.de
190.180.170.60:smtp inet        n       -       -       -       -
smtpd
        -o content_filter=smtp-amavis:[127.0.0.1]:10024
        -o receive_override_options=no_address_mappings

# Outgoing mail backup ip, if default ip is blacklisted
smtp-backup unix        -       -       n       -       -       smtp
        -o smtp_helo_name=smtp2.meine-maildomain.de
        -o smtp_bind_address=190.180.170.62

# Amavisd-new
smtp-amavis unix        -       -       -     -       2  smtp
        -o smtp_data_done_timeout=1200
        -o smtp_send_xforward_command=yes
        -o disable_dns_lookups=yes
        -o max_use=20

127.0.0.1:10025 inet    n       -       -     -       -  smtpd
        -o content_filter=
        -o local_recipient_maps=
        -o relay_recipient_maps=
        -o smtpd_restriction_classes=
        -o smtpd_delay_reject=no
        -o smtpd_client_restrictions=permit_mynetworks,reject
        -o smtpd_helo_restrictions=
        -o smtpd_sender_restrictions=
        -o smtpd_recipient_restrictions=permit_mynetworks,reject
```

```
        -o smtpd_data_restrictions=reject_unauth_pipelining
        -o mynetworks=127.0.0.0/8
        -o strict_rfc821_envelopes=yes
        -o smtpd_error_sleep_time=0
        -o smtpd_soft_error_limit=1001
        -o smtpd_hard_error_limit=1000
```

Die Aliases-Datei sollte noch angepasst werden:

/etc/aliases

```
# See man 5 aliases for format
postmaster:    root
clamav: root
virusalert: root
root: postmaster@meine-maildomain.de
```

Diese wird mittels des Befehls newaliases aktiviert:

```
# newaliases
```

Logs nur nach /var/log/mail.log

/etc/rsyslog.d/50-default.conf

```
#  Default rules for rsyslog.
#
#                       For more information see rsyslog.conf(5) and
/etc/rsyslog.conf

#
# First some standard log files.  Log by facility.
#
auth,authpriv.*                 /var/log/auth.log
*.*;auth,authpriv.none,mail.none                -/var/log/syslog
[...]
```

Den Dienst neustarten:

```
# service rsyslog restart
```

Authentifizierungsmechanismus

Die Berechtigung für den Mailversand vergibt die Software SASL.

Folgende Pakete werden benötigt:

```
# aptitude install libsasl2-2 sasl2-bin libsasl2-modules
```

Die Anbindung haben wir bereits in der `main.cf` konfiguriert. Unser Server arbeitet mit einer SASL-DB des Sasl-Mechanismus.

Hier nochmals die wichtigsten Direktiven:

```
[...]
# Auth über Cyrus-SASL aktivieren
smtpd_sasl_auth_enable = yes

# Nicht standardkonforme Clients erlauben (Outlook & Co.)
broken_sasl_auth_clients = yes

# Nur User Mail versenden lassen, die in SASL-DB stehen
smtp_sasl_security_options = noanonymous

# Standard-Realm für kurze Benutzernamen (optional; erst ab SASL 2.1.19)
smtpd_sasl_local_domain =

# Authentifizierten Clients das Relayen erlauben (Schablone beachten!!)
smtpd_recipient_restrictions =
  ...
  permit_sasl_authenticated
  ...
[...]
```

Wenn SASL in einer Chroot-Umgebung laufen soll (zusammen mit Postfix), muss diese für SASL erst noch angelegt werden. Dies ist nicht Bestandteil des Pakets von Ubuntu.

```
# mkdir -p /var/spool/postfix/var/saslauthd
# chmod 755 /var/spool/postfix/var/run/saslauthd
# chgrp postfix /var/spool/postfix/var/run/saslauthd
# service saslauthd restart
```

SASL selbst muss noch eingestellt werden. Dies geschieht in der Datei `smtpd.conf`:

/etc/postfix/sasl/smtpd.conf

```
log_level: 3
pwcheck_method: saslauthd
auxprop_plugin: sasldb
mech_list: plain login
```

Wir haben definiert, dass Sasl über den SASL-Dämon angesprochen wird. Dieser greift auf eine Berkley-DB (SASLDB) zurück. Erlaubt sind Authenitifizierungen über Plain Text sowie Login.

Jetzt sollte der Dämon noch für den automatischen Start konfiguriert werden. Des Weiteren müssen

dem Dienst auch die Authentifizierungsmethoden bekannt sein.

Da wir Postfix im **Chroot** nutzen, müssen wir den Zugriff auf den Chroot-Käfig ebenfalls hinterlegen. Sonst kann SASLAUTHD nicht darauf zugreifen.

/etc/default/saslauthd

```
#
# Settings for saslauthd daemon
# Please read /usr/share/doc/sasl2-bin/README.Debian for details.
#

# Should saslauthd run automatically on startup? (default: no)
START=yes
[...]
# Example: MECHANISMS="pam"
#MECHANISMS="pam"
MECHANISMS="sasldb"
[...]
OPTIONS="-c -m /var/spool/postfix/var/run/saslauthd"
```

Amavisd-new als Mailscanner ausgehender Post

Folgende Pakete werden installiert:

```
# aptitude install amavisd-new clamav clamav-daemon
```

Die **Konfiguration** geschieht über mehrere Dateien. Hier zunächst ein Überblick über diese:

```
# ls -la /etc/amavis/conf.d/
drwxr-xr-x 2 root root  4096 Dec  9 21:42 ./
drwxr-xr-x 4 root root  4096 Dec  9 21:42 ../
-rw-r--r-- 1 root root  1653 Apr  8  2011 01-debian
-rw-r--r-- 1 root root   705 Aug 17  2011 05-domain_id
-rw-r--r-- 1 root root   429 Apr  8  2011 05-node_id
-rw-r--r-- 1 root root 20693 Apr  8  2011 15-av_scanners
-rw-r--r-- 1 root root   715 Aug 18  2011 15-content_filter_mode
-rw-r--r-- 1 root root  9669 Aug 21  2011 20-debian_defaults
-rw-r--r-- 1 root root   649 Apr  8  2011 21-ubuntu_defaults
-rw-r--r-- 1 root root   573 Apr  8  2011 25-amavis_helpers
-rw-r--r-- 1 root root  2130 Apr  8  2011 30-template_localization
-rw-r--r-- 1 root root  1567 Apr  8  2011 40-policy_banks
-rw-r--r-- 1 root root  1602 May 14  2012 50-user
```

Die wichtigste Datei ist die `50-user`. In ihr geben wir die Amavis-Instanzen an, die IP's und Ports auf denen der Dienst lauschen soll sowie die DKIM-Signierung:

```
use strict;

#
# Place your configuration directives here.  They will override those in
# earlier files.
#
# See /usr/share/doc/amavisd-new/ for documentation and examples of
# the directives you can use in this file
#

$myhostname = "smtp.meine-maildomain.de";

## DKIM signing
$enable_dkim_verification = 1; # enable DKIM signatures verification
$enable_dkim_signing = 1; # load DKIM signing code,

dkim_key('meine-maildomain.de', 'mail', '/var/lib/dkim/meine-
maildomain.de.pem');

@dkim_signature_options_bysender_maps = (
    { '.' => { ttl => 21*24*3600, c => 'relaxed/simple' } } );

# switch policy bank to 'ORIGINATING' for mail received on port 10026:
$interface_policy{'10026'} = 'ORIGINATING';
$policy_bank{'ORIGINATING'} = {  # mail originating from our users
  originating => 1,  # indicates client is ours, allows signing
  # force MTA to convert mail to 7-bit before DKIM signing
  # to avoid later conversions which could destroy signature:
  smtpd_discard_ehlo_keywords => ['8BITMIME'],
};

#@local_domains_maps = ( read_hash("$MYHOME/localdomains") );
@local_domains_maps = ('meine-maildomain.de');

$virus_admin = 'postmaster@meine-maildomain.de';         # due to D_DISCARD
default

@inet_acl = qw(127.0.0.1 [::1] 213.240.143.77 213.240.143.78);
$inet_socket_bind = '127.0.0.1';
$inet_socket_port = [10024,10026];

## Own policy, not default

#$policy_bank{'MYNETS'} = {
#   originating => 1,
#   };
$smtp_connection_cache_enable = 0;

#------------ Do not modify anything below this line -------------
1;  # ensure a defined return
```

In der `20-debian_defaults` ändere ich nur folgende Zeile, um die Produktversion zu verschleiern:

```
[...]
$X_HEADER_LINE = "$myproduct_name at $mydomain";
[...]
```

In der `15-content_filter_mode` muss folgende Zeile auskommentiert werden. Diese aktiviert den Virenscanner:

```
[...]
@bypass_virus_checks_maps = (
   \%bypass_virus_checks, \@bypass_virus_checks_acl,
\$bypass_virus_checks_re);
[...]
```

Berechtigungen setzen, damit Clamav auch auf die Mails von Amavis zugreifen kann:

```
# adduser clamav amavis
```

RAM-Disk für bessere Performance

/etc/fstab

```
[...]
tmpfs           /var/lib/amavis/tmp         tmpfs
defaults,size=128m,mode=755,uid=108,gid=117      0 0
```

Die uid sowie die gid müssen durch die entsprechenden Werte des amavis-Dienstusers ersetzt werden!

```
# id amavis
uid=108(amavis) gid=117(amavis) groups=117(amavis)
```

Anschließend müssen die Ordnerrechte noch entsprechend gesetzt werden:

```
# chown -R amavis:amavis /var/lib/amavis/tmp
```

DKIM

Den DKIM-Key erzeugen:

```
# mkdir -p /var/lib/dkim
# amavisd-new genrsa /var/lib/dkim/meine-maildomain.de.pem
```

Den Key testen und in die DNS-Zone eintragen:

```
# amavisd-new showkeys
; key#1, domain meine-maildomain.de, /var/lib/dkim/meine-maildomain.de.pem
mail._domainkey.meine-maildomain.de.       3600 TXT (
  "v=DKIM1; p="
  "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDUzgyCExvKoRWDGHb6CJIR4+lE"
  "bpcX0yp5GuryORz5S6LC3M7TCSWPi8z9vV1Peb9yddPQ2mPrQef2IqNKTsstrbXC"
  "CXVMoko8sXCoh/O5XDoLX1mXUqlcS9LcqJG+thF4W/43SsJyYoCEGdtokjD9nAMt"
  "03tOjr+dEDpW1Uk7pQIDAQAB")

# amavisd-new testkeys
TESTING#1: mail._domainkey.meine-maildomain.de          => invalid (public
key: not available)
```

Nun den Dienst neu starten und die Änderungen übernehmen:

```
# service amavis restart
```