# Setting up a mail server on Ubuntu 16.04

Help and Support

Help Desk

Homepage

Attachments

Capture.PNG

SERVER MANAGEMENT

Managed Service Provider | Network Solutions | Information Systems | IT Projects | www.serman.ee

## Overview

> This article is written by Server Management Inc. IT systems administrator Kristo Rood- 07.12.2017, Tartu, Estonia.

## Need help configuring your device? Contact us info@serman.ee

## Prerequisities

1. A linux server with (atleast) Ubuntu 16.04.2
2. 3-4 hours of time

# 1. Postfix Configuration

## Setting up a hostname

Before installing postfix you have to set your hostname to a FQDN (Fully qualified domain name). You can do this by doing:

```
sudo hostnamectl set-hostname fqdn
```
for example

```
sudo hostnamectl set-hostname mail.yourdomain.com
```

## Configuring date and time

For postfix to time emails and logs correctly you need your server to have the correct timezone. You can check if your servers time matches with your computers time by doing:

```
date
```
which will output something like this:

```
Tue Nov 28 11:33:39 EET 2017
```
if you need to change the timezone, you can use this command:

```
sudo dpkg-reconfigure tzdata
```
and select the correct timezone for your are from the list.

## DNS Setup

For this to work you need to set up 3 dns records.

### MX Record

MX record @ mail.yourdomain.com
Where mail.yourdomain.com is your FQDN.

### A record

mail.yourdomain.com 1.1.1.1
Where mail.yourdomain.com is your FQDN, and 1.1.1.1 is the ip to your server.

### PTR record

For reverse lookups you need to set up a PTR Record

1.1.1.1 mail.yourdomain.com
Where 1.1.1.1 is the ip to your server, and mail.yourdomain.com is your FQDN.

## Installing Postfix

To install postfix run the following commands:

sudo apt-get update sudo apt-get install postfix -y
During installation you will be asked to select a configuration type:

1. No configuration means the installation process will not configure any parameters.
2. Internet Site means using Postfix for sending email to other MTAs and Receiving email from other MTAs.
3. Internet with smarthost means using postfix to receive email from other MTAs, but using another smart host to relay emails to the recipient.
4. Satellite system means using smart host for sending and receiving email.
5. Local only means emails are transmitted only between local user account.

In this case we will choose **Internet Site**.

Next you should enter your domain name, bare in mind that this should be a single FQDN (without a node name) e.g. yourdomain.com. This means your mail address will be user@yourdomain.com.

To check if postfix is listening on port 25 use the following command:

sudo netstat -lnpt
To check if port 25 is open on your firewall or hosting provider

sudo apt install nmap sudo nmap server-ip
To send a test email use:

echo "test" | sendmail username@gmail.com

# 2. Setting up DKIM, SPF & DMARC

## Setting up SPF

### Setting up records

First you have to set up a DNS TXT record like this:

TXT @ "v=spf1 mx -all"
Use dig to find out if the DNS record has updated:

dig yourdomain.com txt

### Configuring Policy Agent

First install policy packages by doing:

sudo apt install postfix-policyd-spf-python
As the next step open up Postfix master configuration by doing:

sudo nano /etc/postfix/master.cf
And then add the following lines to the end of it:

policyd-spf unix - n n - 0 spawn user=policyd-spf argv=/usr/bin/policyd-spf
After that edit the main Postfix configuration:

sudo nano /etc/postfix/main.cf

And then add the following lines to the end of it:

```
policyd-spf_time_limit = 3600
smtpd_recipient_restrictions =
    reject_unauth_destination,
    check_policy_service unix:private/policyd-spf
```

Restart Postfix to apply the changes:

```
sudo systemctl restart postfix
```

# Setting up DKIM

Firstly install OpenDKIM and its tools by doing:

```
sudo apt-get install opendkim opendkim-tools
```

Then add the postfix user to the opendkim group:

```
sudo gpasswd -a postfix opendkim
```

Then edit the configuration file:

```
sudo nano /etc/opendkim.conf
```

And make it look like the following:

```
# This is a basic configuration that can easily be adapted to suit a standard
# installation. For more advanced options, see opendkim.conf(5) and/or
# /usr/share/doc/opendkim/examples/opendkim.conf.sample.
# Log to syslog
Syslog          yes
# Required to use local socket with MTAs that access the socket as a non-
# privileged user (e.g. Postfix)
UMask           002
# Sign for example.com with key in /etc/mail/dkim.key using # selector '2007' (e.g. 2007._domainkey.example.com) #Domain          example.com #KeyFile       /etc/mail/dkim
# Commonly-used options; the commented-out versions show the defaults.
Canonicalization    relaxed/simple
Mode            sv
SubDomains      no
AutoRestart     yes
AutoRestartRate     10/1M
Background      yes
DNSTimeout      5
SignatureAlgorithm      rsa-sha256
# Always oversign From (sign using actual From and a null From to prevent
# malicious signatures header fields (From and/or others) between the signer
# and the verifier.  From is oversigned by default in the Debian pacakge
# because it is often the identity key used by reputation systems and thu
# somewhat security sensitive.
OversignHeaders     From
# List domains to use for RFC 6541 DKIM Authorized Third-Party Signatures # (ATPS) (experimental)
#ATPSDomains        example.com
#OpenDKIM user # Remember to add user postfix to group opendkim
UserID              opendkim
# Map domains in From addresses to keys used to sign messages
KeyTable            /etc/opendkim/key.table
SigningTable        refile:/etc/opendkim/signing.table
# Hosts to ignore when verifying signatures
ExternalIgnoreList  /etc/opendkim/trusted.hosts
InternalHosts       /etc/opendkim/trusted.hosts
```

## Configuring signing table & trusted hosts

First make a directory for the configurations and add appropriate permissions:

```
sudo mkdir -p /etc/opendkim/keys
sudo chown -R opendkim:opendkim /etc/opendkim
sudo chmod go-rw /etc/opendkim/keys
```

Next create the signing table:

```
sudo nano /etc/opendkim/signing.table
```

And populate it with the following:

```
*@yourdomain.com default._domainkey.yourdomain.com
```

Where yourdomain.com is your domain name.

Next up make the key table

```
sudo nano /etc/opendkim/key.table
```

And populate it with the following:

```
default._domainkey.yourdomain.com yourdomain.com:default:/etc/opendkim/keys/yourdomain.com/default.private
```

Where yourdomain.com is your domain name.

Finally add trusted hosts:

```
sudo nano /etc/opendkim/trusted.hosts
```

And add the following to the file:

```
127.0.0.1 localhost *.yourdomain.com
```

Where yourdomain.com is your domain name.

## Generating keys

First make a directory for the keys like so:

```
sudo mkdir /etc/opendkim/keys/yourdomain.com
```

Then use the included OpenDKIM tool to generate keys:

```
sudo opendkim-genkey -b 2048 -d yourdomain.com -D /etc/opendkim/keys/yourdomain.com -s default -v
```

Lastly change permissions to the key

```
sudo chown opendkim:opendkim /etc/opendkim/keys/yourdomain.com/default.private
```

### Adding a DNS record for DKIM

First get the public key:

```
sudo cat /etc/opendkim/keys/yourdomain.com/default.txt
```

From there you can copy everything in the parentheses, **but make sure to remove whitespaces and quotes**.

Then in the DNS records manager make a configuration like this

```
TXT default._domainkey "v=DKIM1;k=rsa;p=your_key"
```

To test the key:

```
sudo opendkim-testkey -d yourdomain.com -s default -vvv
```

### Connecting DKIM setup to Postfix

First create a directory for the OpenDKIM socket file and add appropriate permissions to it:

```
sudo mkdir /var/spool/postfix/opendkim sudo chown opendkim:postfix /var/spool/postfix/opendkim
```

Then edit the socket configuration file.

```
sudo nano /etc/default/opendkim
```

Uncomment the first SOCKET line and replace it with the following line.

```
SOCKET="local:/var/spool/postfix/opendkim/opendkim.sock"
```

Next edit the postfix configuration file

```
sudo nano /etc/postfix/main.cf
```

And add the following lines:

```
# Milter configuration
# OpenDKIM
milter_default_action = accept
milter_protocol = 2
smtpd_milters = local:/opendkim/opendkim.sock
non_smtpd_milters = local:/opendkim/opendkim.sock
```

Then restart opendkim and postfix:

```
sudo service opendkim restart
sudo service postfix restart
```

### Testing

To test the current configuration and to see if it fully works, you can use the port25 verifier like so:

```
echo "test" | sendmail check-auth@verifier.port25.com
```

If everything works correctly the result should be something like this:

```
=== Summary of Results ===
SPF check: pass
DKIM check: pass
SpamAssassin check: ham
```

## Setting up DMARC

To set up DMARC you just need to add a new DNS record like this:

```
v=DMARC1; p=none; rua=mailto:reports@yourdomain.com
```

# 3. Dovecot & TLS

## Generating a certificate

Firstly we need to install certbot:

```
sudo apt-get install letsencrypt
```

Next generate the certificate

```
sudo letsencrypt certonly --agree-tos --email your-email-address -d mail.yourdomain.com
```

## Configuring Postfix

To use an email client we need to enable submission of emails. First open the master.cf file

```
sudo nano /etc/postfix/master.cf
```

And uncomment or add the following lines **(make sure you leave whitespace before -o)**

```
submission inet n - y - - smtpd
 -o syslog_name=postfix/submission
 -o smtpd_tls_security_level=encrypt
 -o smtpd_tls_wrappermode=no
```

```
 -o smtpd_sasl_auth_enable=yes
 -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
 -o smtpd_recipient_restrictions=permit_mynetworks,permit_sasl_authenticated,reject
 -o smtpd_sasl_type=dovecot
 -o smtpd_sasl_path=private/auth
```

Next add the certificates to postfix for that open the main.cf file:

```
sudo nano /etc/postfix/main.cf
```

Then edit or add these TLS parameters:

```
smtpd_tls_cert_file=/etc/letsencrypt/live/mail.yourdomain.com/fullchain.pem
smtpd_tls_key_file=/etc/letsencrypt/live/mail.yourdomain.com/privkey.pem
smtpd_use_tls=yes
smtp_tls_security_level = may
smtp_tls_loglevel = 1
smtpd_tls_loglevel = 1
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache
smtpd_tls_security_level=may
smtpd_tls_protocols = !SSLv2, !SSLv3
```

Lastly reload postfix

```
sudo postfix reload
```

# Setting up Dovecot

Firstly install dovecot by doing:

```
sudo apt install dovecot-core dovecot-imapd
```

Then open the dovecot configuration file

```
sudo nano /etc/dovecot/dovecot.conf
```

And add the following line

```
protocols = imap
```

# Configuring Mailbox

Open the mailbox configuration file:

```
sudo nano /etc/dovecot/conf.d/10-mail.conf
```

Then uncomment or add the folloing line:

```
mail_privileged_group = mail
```

Next add permissions to Dovecot

```
sudo gpasswd -a dovecot mail
```

# Configuring encryption

First open the auth config file:

```
sudo nano /etc/dovecot/conf.d/10-auth.conf
```

Then change the following lines (which changes the login format to user@yourdomain.com):

```
disable_plaintext_auth = yes
auth_username_format = %n
```

Next open the ssl config file:

```
sudo nano /etc/dovecot/conf.d/10-ssl.conf
```

Then edit the following lines like this (picture of configuration in attachment)

# Add authentication between Dovecot and Postfix

Open the following file:

```
sudo nano /etc/dovecot/conf.d/10-master.conf
```

And change the service auth section of the file to look like the following:

```
service auth {
    unix_listener /var/spool/postfix/private/auth {
      mode = 0660
      user = postfix
      group = postfix
    }
}
```

# Auto create mail folders

To edit mail folders open the mailboxes configuration file:

```
sudo nano /etc/dovecot/conf.d/15-mailboxes.conf
```

To auto create a folder, add this inside the corresponding section

```
auto = create
```

For example:

```
mailbox Trash {
    auto = create
    special_use = \Trash
 }
```

Then restart Dovecot:

```
sudo service dovecot restart
```

# Configuring Email client

To configure a desktop email client you need to set up the following settings in the client:

| | Protocol | Server hostname | Port | SSL |
|---|---|---|---|---|
| Incoming | IMAP | mail.yourdomain.com | 993 | SSL/TLS |
| Outgoing | SMTP | mail.yourdomain.com | 587 | STARTTLS |