安装和配置 Postfix

Rei 写于 22 Apr 2015

Postfix 是一个 MTA（Mail Transfer Agent），可以用来收发邮件。开发网站多少都需要收发邮件的功能，例如邮件验证、找回密码等。

配置邮件系统过程比较复杂，而且需要很多维护工作，如果发送量不大，可以先用 Mailgun，Mandrill 等第三方邮件发送服务，在遇到以下情况的时候，再考虑自建邮箱系统：

- 第三方服务费用太高。
- QQ 邮箱对第三方服务拒信率高。
- 邮件排队时间过长，发送不及时。

自建邮件系统可以处理这些问题，有更多优化空间。

系统需求

Ubuntu LTS 14.04 。

设置 hostname

事先设置好 hostname 的话，Postfix 可以自动配置好很多参数，节省时间。假设你的网站域名是 `example.com`，要搭建独立的邮件服务器（推荐），就把主机名设置为 `mail.example.com`。

```
# echo 'mail.example.com' > /etc/hostname
# hostname -F /etc/hostname
```

安装 Postfix

```
# apt-get install postfix
```

安装过程会弹出设置窗口，全部回车确认既可。

基本配置

Postfix 的配置文件位于 `/etc/postfix` 文件夹。先看 `main.cf` 文件，有几个重要的配置。如果事先设置了正确的 hostname，那么这些配置已经自动设置好了。

myhostname

```
myhostname = mail.example.com
```

myhostname 让 Postfix 知道自己主机的名字。

myorigin

```
myorigin = /etc/mailname
```

myorigin 的值存放在另一个文件中，打开这个文件可以看到一行内容 mail.example.com。

在通过 Postfix 发送邮件的时候，如果 From 字段不完整，例如 From: user，Postfix 会根据 myorigin 的值将地址补全为 From: user@mail.example.com。

* 发邮件的时候 From 字段是可以随意指定的。

mynetworks

```
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
```

mynetworks 指定了本地网络的 IP 段，默认只包含主机自己。

smtpd_relay_restrictions

```
smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_
```

smtpd_relay_restrictions 指定了 Postfix 在作为邮件发送方的时候，只接受通过以下规则的发信请求：

- permit_mynetworks 允许 mynetworks 包含的主机发信。
- permit_sasl_authenticated 允许通过 SASL 身份验证的主机发信。
- defer_unauth_destination 不符合其它转发规则的时候拒绝发信。

mydestination

```
mydestination = mail.example.com, localhost.example.com, localhost
```

mydestination 指定了 Postfix 在收到这些域名地址作为目标地址的邮件时，作为接收方收下邮件。

如果收到的邮件既不符合转发规则，又不符合接收规则，则会拒绝收信。

由于我希望这台服务器能接受主域名 example.com 的邮件，所以将这个配置修改为：

```
mydestination = example.com, mail.example.com, localhost.example.com, localhost
```

重载 Postfix：

```
# service postfix reload
```

为了检查安装情况，现在先做一些测试。

测试一：发邮件

用 sendmail 命令给自己的邮箱发送一封空邮件

```
# sendmail youremail@gmail.com
.
```

输入第一行的时候，sendmail 会等待输入邮件内容，此时直接输入一个 . 结束输入，这会产生一个空邮件。

登录你的邮箱，如无意外可以在垃圾邮件箱找到这封邮件。这说明 Postfix 已经具有发送能力。

测试二： 收邮件

由于还没有配置 DNS，其它邮件服务商还无法识别这部主机，先在另一台主机用 telnet 进行测试，假设邮件服务器的 IP 是 192.168.33.10：

```
$ telnet 192.168.33.10 25
Trying 192.168.33.10...
Connected to 192.168.33.10.
Escape character is '^]'.
220 mail.example.com ESMTP Postfix (Ubuntu)
MAIL FROM: youremail@gmail.com
250 2.1.0 Ok
RCPT TO: root
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
text
.
250 2.0.0 Ok: queued as 651FE22162
QUIT
Connection closed by foreign host.
```

加粗部分是需要输入的内容。

在邮件服务器上检查信件：

```
# tailf /var/mail/root
```

大概会看到这样的内容：

```
From youremail@gmail.com  Wed Apr 22 16:13:33 2015
Return-Path: <youremail@gmail.com>
X-Original-To: root
Delivered-To: root@mail.example.com
Received: from unknown (unknown [192.168.33.1])
        by mail.example.com (Postfix) with SMTP id 651FE22162
        for <root>; Wed, 22 Apr 2015 16:13:13 +0000 (UTC)

text
```

Postfix 默认使用 mbox 格式将系统用户的邮件存放到 /var/mail 目录下。

MX 记录

如果你的邮件服务器已经部署到公网上，要用来接收其它邮件服务商发来的邮件，那么需要到域名的 DNS 服务器进行修改。

首先给邮件服务器设置 A 记录（自行替换为真实 IP）：

```
mail.example.com. IN A 192.168.33.10
```

然后给主域名设置 MX 记录：

```
example.com. IN MX 10 mail.example.com.
```

在本地测试更新状况：

```
$ dig example.com mx
```

如果返回了正确的 MX 记录，则邮件服务器已经可以被识别，但不同服务商所用的 DNS 更新状况不一样，全部生效也许要过一段时间。

在本地测试邮件服务器是否能收到邮件：

```
$ sendmail root@example.com
.
```

aliases

用登录 ssh 的方式收邮件不方便，我们可以使用 aliases 功能将邮件转发到自己的个人邮件地址。

打开 /etc/aliases 文件，目前应该是这样：

```
# See man 5 aliases for format
postmaster: root
```

将 root 作为别名，转发到个人邮件地址：

```
# See man 5 aliases for format
postmaster: root
root: youremail@gmail.com
```

这样以 root@example.com 作为目的地址的邮件将会转发到 youremail@gmail.com。

别名支持多个地址，所以一个简易的邮件列表可以这样实现：

```
# See man 5 aliases for format
postmaster: root
root: youremail@gmail.com
support: youremail@gmail.com, another@gmail.com
```

这样以 support@example.com 作为目的地址的邮件将会转发到 youremail@gmail.com 和 another@gmail.com 。

修改 /etc/aliases 文件后，需要运行一条命令让它生效：

```
# newaliases
```

SASL 身份验证

目前 Postfix 只能为本地应用发送邮件，还不接收为远程应用发送邮件。如果你的应用跟 Postfix 装在同一个服务器，那么无需身份验证既可发送邮件；而如果不在同一个服务器，则需要配置某种验证方式验证发信者的身份。

一个方法是配置 mynetworks，把应用服务器纳入本地网络，以 192.168.33.11 为例：

```
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128 192.168.33.11
```

mynetworks 的值还支持掩码，如果应用服务器位于同一个网段，这种方法比较方便。

我比较习惯帐号密码的方式验证，在 Postfix 可以使用 SASL 模块实现帐号密码验证。SASL 模块支持多种帐号密码储存方式，这里只介绍用独立数据库文件（sasldb2）存放的方式。

安装 SASL

首先安装一些工具用于创建 SASL 专用的帐号密码。

```
# apt-get install sasl2-bin
```

创建 SASL 帐号密码

```
# saslpasswd2 -c -u example.com postmaster
# cp -a /etc/sasldb2 /var/spool/postfix/etc/
```

* Postfix 的 smtpd 进程默认使用了 chroot，所以要把数据库文件拷贝到 /var/spool/postfix/etc 目录内。

输入后会提示输入两次密码，创建成功后帐号为 postmaster@example.com，密码为所输入的密码。

为了让 Postfix 能读取这个文件，把 postfix 加到 sasl 组，并设为只读：

```
# gpasswd -a postfix sasl
# chmod 640 /var/spool/postfix/etc/sasldb2
```

检查现有的帐号：

```
# sasldblistusers2 -f /var/spool/postfix/etc/sasldb2
```

配置 Postfix

在 /etc/postfix/main.cf 添加以下内容：

```
# SASL
smtpd_sasl_auth_enable = yes
smtpd_tls_auth_only = yes
```

这些配置打开 SASL 登录，并且只允许 TLS 安全传输的情况下进行验证。之所以强制 TLS，是因为默认的 PLAIN 校验传输的几乎是明文密码。

新建文件 /etc/postfix/sasl/smtpd.conf，添加内容：

```
pwcheck_method: auxprop
```

这个配置指定使用数据库文件读取帐号密码信息。

修改配置后，重载 Postfix 让配置生效：

```
# service postfix reload
```

测试登录

首先准备 SASL PLAIN 验证需要的帐号密码字符串，输入以下命令：

```
printf '\0%s\0%s' 'postmater@example.com' '123456' | openssl base64
```

实际中替换你需要的帐号密码，输出结果即为登录字符串。

由于设置了强制 TLS 登录，用 telnet 就不那么方便了，这时候可以用以下命令连接：

```
openssl s_client -connect mail.example.com:25 -starttls smtp
```

这条命令打开到服务器的 smtp 连接，并且完成 starttls 过程，之后界面跟 telnet 类似。完成校验的命令如下：

```
auth plain AHBvc3RtYXRlckBleGFtcGxlLmNvbQAxMjM0NTY=
235 2.7.0 Authentication successful
```

看到 `Authentication successful` 即为通过校验。现在远程主机可以通过帐号密码登录，使用 Postfix 发信了。

SPF 记录

SPF 记录是一种通过 DNS 记录，验证邮件发送主机的 IP 是否可信的方法。之所以需要额外的验证方式，是因为 Email 的发送地址很容易伪造，没有有效 SPF 记录的邮件很可能被归为垃圾邮件。后面的 DKIM 也是验证邮件可信度的一种方式。

配置 SPF 记录分为设置发信和校验来信两种情况。

设置发信 SPF

发信的 SPF 记录不需要 Postfix 设置，而完全在 DNS 上。

为自己的域名添加一条 TXT 记录：

```
example.com IN TXT "v=spf1 mx ~all"
```

这条记录表示域名自身 MX 记录指向的主机为可信主机，~all 表示除此以外的主机为软拒绝。

SPF 记录的语法规则可以查阅 http://www.openspf.org/SPF_Record_Syntax 。

校验来信 SPF

Postfix 需要安装一个组件以支持 SPF 校验。

```
# apt-get install postfix-policyd-spf-perl
```

添加 Postfix policy：

```
# postfix-add-policy spfcheck nobody /usr/sbin/postfix-policyd-spf-perl
```

在 /etc/postfix/main.cf 添加以下内容：

```
# SPF
smtpd_recipient_restrictions = permit_mynetworks permit_sasl_authenticated check_pol
spfcheck_time_limit = 3600
```

重载配置：

```
# service postfix reload
```

现在 Postfix 每收到一封邮件都会进行 SPF 校验，然后在邮件头部加入类似 Received-SPF: pass 的信息。

DKIM 签名

DKIM 是另一种验证邮件有效性的方法。跟 SPF 不同，DKIM 在 DNS 公开一个公钥，然后用私钥对自己的邮件进行签名。收信者查询发信方域名获得公钥，然后校验签名是否有效。

为 Postfix 添加 DKIM 支持需要用到 opendkim 这个包。

安装 opendkim

```
# apt-get install opendkim opendkim-tools
```

配置 opendkim

打开 /etc/opendkim.conf，添加以下内容：

```
Domain example.com
KeyFile /etc/mail/dkim.key
Selector mail
```

Domain 为自己的域名，KeyFile 为域名对于的私钥，Selector 为公钥存放的主机名（这里设置为 mail._domainkey.example.com）。

打开 /etc/default/opendkim，添加以下内容：

```
SOCKET="inet:8891@localhost" # listen on loopback on port 8891
```

这里让 opendkim 的守护进程监听 8891 端口，用于和 Postfix 通信。

* 你应该配置防火墙禁止外部访问白名单以外的端口。

生成密钥和配置 DNS

生成密钥：

```
# mkdir /etc/mail
# cd /etc/mail
# opendkim-genkey -s mail -d example.com
# cp mail.private dkim.key
```

查看 mail.txt 文件，里面有 dkim 的公钥，将它添加为 DNS TXT 记录，类似于：

```
mail._domainkey.example.com. IN TXT "v=DKIM1; k=rsa; p=PpYHdE2tevfEpvL1Tk2dDYv0pF28/
```

重启 opendkim：

```
# service opendkim restart
```

配置 Postfix

打开 /etc/postfix/main.cf，添加以下内容：

```
# DKIM
smtpd_milters = inet:localhost:8891
```

```
non_smtpd_milters = inet:localhost:8891
```

这为 Postfix 加上了 DKIM 的过滤器，发信的时候签名，收信的时候校验。

重载 Postfix：

```
# service postfix reload
```

现在 Postfix 会自动为发出的邮件签名，校验收到的邮件。

总结

现在已经利用 Postfix 搭建了一个邮件服务器，我们可以使用它收发网站邮件。但搭建完毕只是第一部，接下来还需要观察送达情况。最重要的一点，不要发送垃圾邮件，不要让网站有漏洞让用户发送垃圾邮件，否则会拉低所有邮件的评分，导致正常邮件也发不出去。

要深入了解 Postfix 的使用，推荐书籍：《Postfix 权威指南》。看完这本后就可以看官方文档：http://www.postfix.org/documentation.html 。