If you are reading this, you have an extra PC, NAS, or raspberry pi that you are looking to make into a email server. Unfortunatelly, email servers are notoriously difficult to create. Fortunately enough, scripts have been created that more or less automate this process and stream lined a lot of it. Particularly, I nabbed a script off Luke Smith that installs and configures it for you. Unfortunately (again), if you are setting this server up in a residential area with residential email service you may have some difficulties outside of setting up the email server battling with your ISP (Internet Service Provider) due to blocked ports. Fortunately (again), there is a simple work around.

Conceptually there is not that much to do, which is what this blog post is going to be about. The specifics will be left to the script, so if you are interested in learning how that works go read through that. It isn't earth shattering, but there are enough intricies that I am going to stick to borrowing the script when needed. What it boils down to is: install postfix, install dovecot, install spamassasin, install OpenDKIM, configure postfix to utilize SMTP, configure dovecot to connect to postfix and allow external connection for IMAP, link spamassasin, and hook up OpenDKIM. If you didn't get all that, don't worry. Each section will break down the components.

The issue I came accross while setting up my email server is that my ISP (Internet Service Provider) was blocking port 25. Although I had a working email server, it was rendered useless because I couldn't send/receive email outside of my local network. This just requires some external help and minor tweaks on your end to get things moving smoothly.

I'll also preface this by saying that the script README says that it is configured for VPS (Vitual Private Servers), but it worked great on my little 300$ hp pavillion makeshift server running Debian 10. Now to talk about each component.

Email Server Components

Postfix

As stated by Wikipedia:

> Postfix is a free and open-source mail transfer agent (MTA) that routes and delivers electronic mail.

Or in other words, this is the guy that will send/receive mail to and from the outside world. It allows for SMTP with TLS (which is what is enabled with this script), and allows for email to be encrypted (again if you want to know the specifics look at the script). This program is also the one who will send your mail down the chain to be saved once it has been grabbed. It has to work with Dovecot to know how to format it so you can access it with your email client and Spamassasin to know what sort of mail to keep and reject.

Dovecot

> Dovecot is an open source IMAP and POP3 email server for Linux/UNIX-like systems

Basically, this is what you interface with to access your email through your email client. Again, it allows for encrypted connection and what not. It is also responsible for the structure of you mail folders. Say you have an Inbox, Junk, Sent, and Drafts folder. That is laid out with Dovecot. It also needs to know about Postfix in order to know where to grab it's mail from.

Spamassasin

What it does is pretty self evident by the name. To be safe though here is a description via Wikipedian

> It uses a variety of spam-detection techniques, including DNS and fuzzy checksum techniques, Bayesian filtering, external programs, blacklists and online databases.

OpenDKIM

> OpenDKIM is an open source implementation of the DKIM (Domain Keys Identified Mail) sender authentication system proposed by the E-mail Signing Technology Group (ESTG), now standardized by the IETF.

In other words, it just signs your email in a way to show that is valid in such a way that big email servers (Like Gmail, Outlook, etc.) will accept your mail instead of tossing it to the side as spam.

Problem with Residential email servers

Port 25 is blocked by a **lot** of ISPs. Why does that matter? Well, port 25 is what mail servers use to transfer email. That means that your email server is only good locally. There are two things you need to do in order to fix this problem:

1. You need to borrow an SMTP server
2. You need an external service to re-route your email to another port

Maybe it isn't the best answer since you are probably trying to gain more independence and privacy on the internet, but it works amazingly.

Borrowing SMTP server

The quick and easy way to get it up and running is to borrow Google's STMP servers. If you have an account, then you are all set to get this done. This also should work if your ISP supplies an SMTP server. You will just need the credentials for your ISP email account.

First, create a file such as /etc/postfix/sasl_passwd. In that file you are going to type out:

```
[mail.example.com]:587 user@example.com:passwd
```

where mail.example.com will be smtp.gmail.com if you are using gmail. Then supply your email address and password as shown. For the sake of security, and how almost all examples include, is to make this file only accessible to read or write via root. To do so type:

```
chmod 0600 /etc/postfix/sasl_passwd
```

Then create a database for postfix to access by typing:

```
postmap /etc/postfix/sasl_passwd
```

Now to configure postfix to read that map. Open /etc/postfix/main.cf and find/create the lines as follows:

```
relayhost                  = [mail.example.com]:587
smtp_sasl_auth_enable      = yes
smtp_sasl_password_maps    = hash:/etc/postfix/sasl_passwd
smtp_sasl_security_options = noanonymous
smtp_tls_CApath            = /etc/ssl/certs
smtp_use_tls               = yes
```

Restart postfix via systemctl restart postfix and try sending an email to yourself.

Reroute incoming mail

If you are in a residential area, you most likely need a DDNS (Dynamic Domain Name System), personally I use Dynu. It is free, and easy to set up. The only thing you need to do here is create an account, go to Dynamic DNS Service, click the add button, use one of the given DNS names or type in your own. If you opted to use your own domain name on the website you got your domain name from in your "External Hosts" record type in the ip address given to you by Dynu.

First thing you need to do is to create a rollernet.us account. Go to mail derives, select SMTP Redirection (Direct Connection) (I believe that is the one that I am using. At any rate, for free accounts, only one of the SMTP redirections work). Type in your domain, and under your destination server, type your domain name again. Under port, type something like 2525 (this is the port we are going to be redirecting mail to). Now you need to set permissions for what is allowed through. In the "Mail Services" base page there is a hyper link labeled "valid user table", click that. For all domain names given, ensure that the “Default Action” is “Allow”. This will allow all data incoming on port 25 be moved to port 2525.

Going back to the website you got your Domain name from. Under “Email Services” create two new entries with the “Points to” options filled out as `mail.rollernet.us` and `mail2.rollernet.us`.

There are two ways you can make sure that you receive mail from this untraditional port in postfix. You can either redirect port 2525 back to port 25 on your router, or you can add the line

```
2525 inet n - y - - smtpd
    -o content_filter=spamassassin
```

After you complete all these configurations, give everything some time for all settings to take effect (those set for the DDNS and your domain name mostely). You should now be able to send and receive email from you home server! If you have any questions about the specific setup process, I’d recommend watching Luke Smith’s Video about setting up an email server with his script. Again, all these changes are built off of the changes made by that script.

Thank you for reading, and happy emailing!