



# **Part 24**

-

# **Mail Server**

*Version: 2019-12-10*

## Raspberry Pi Email Server using Citadel

This Raspberry Pi email server tutorial will take you through the steps on how to setup your very own email server that is both low cost and relatively versatile.

One of the many reasons to setup a personal server on the Pi is its low cost. The initial cost is low and the ongoing expenses is also relatively low due to the Raspberry Pi's low power usage. There are several other advantages to setting up a Raspberry Pi email server such as being able to utilize a custom domain name and having full control over your own personal communications. Since your emails are stored on your own personal server you will have no one scanning them so they can't try and present you with better advertisements. You can run this alongside an [Apache web server](#) for a complete setup.

Another great reason to setting up your own email server is that it's a great learning experience when it comes configuring everything correctly. It can be hard setting this up as there are quite a few things that can go wrong and if you're a beginner then troubleshooting may be a bit frustrating. The software itself is super easy to install but can be hard configuring it to work perfectly the first time so I have marked this project as intermediate difficulty. If you plan on sending email you will need to make sure your ISP is not blocking port 25. This is quite common but can usually be unblocked if requested.

### Installing the Citadel Email Server

There are a few ways to install a email server on the Raspberry Pi, one of the most complicated ways is to install Postfix and Dovecot separately then installing a web frontend. Each of these components will then have to be individually configured and can become quite time consuming to set up correctly.

However, we will be utilizing a software package called Citadel (<http://www.citadel.org/doku.php>). It is an easy to set up, powerful yet versatile mailing server. It heavily simplifies the process of setting up a email server and does not require extensive configuration to get it up and running. Best of all it is a 100% open source software.

1. First let's update our package repository by running the following command.

```
sudo apt-get update
```

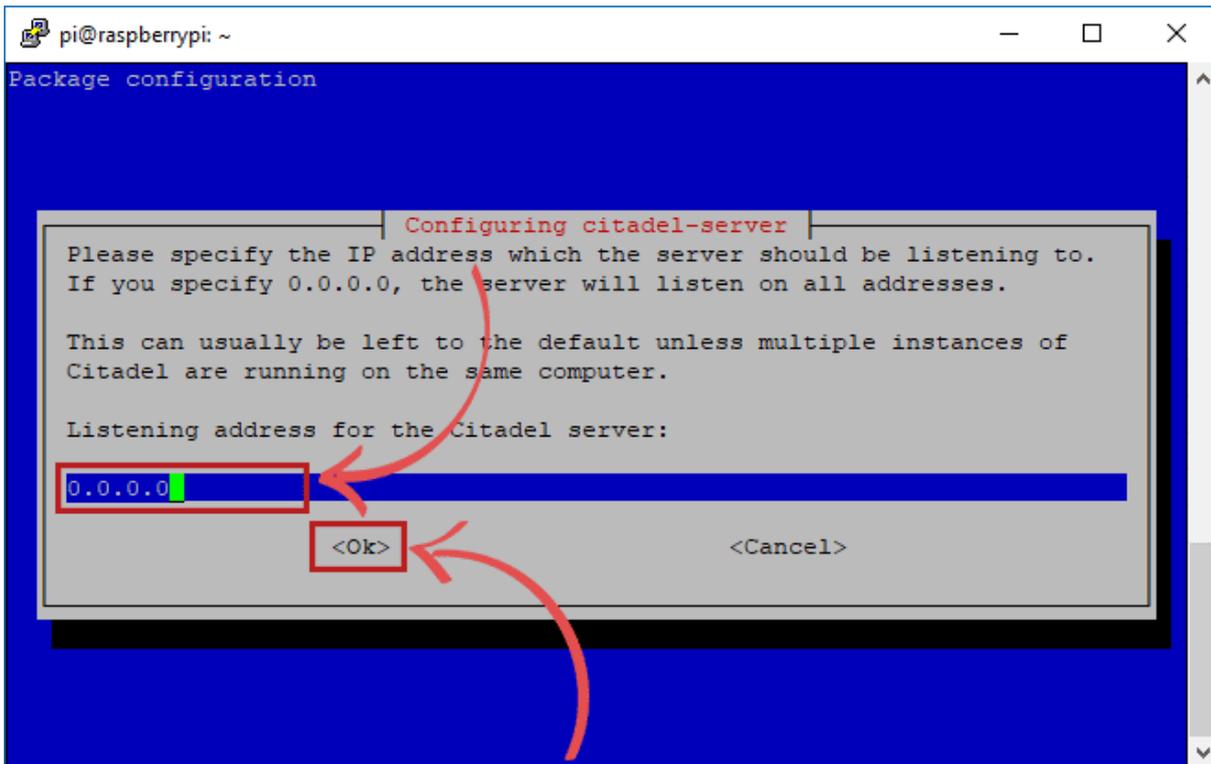
2. Now run upgrade so that you're updated and running on the latest packages.

```
sudo apt-get upgrade
```

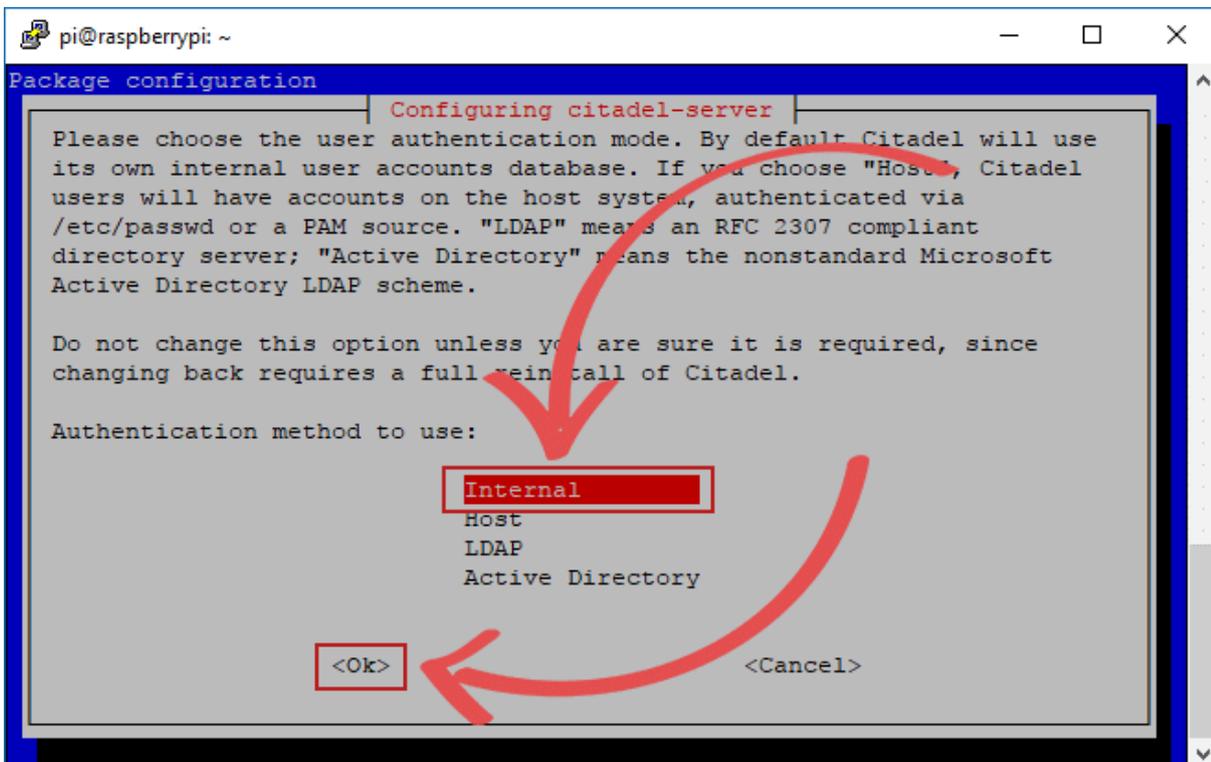
3. It's now time to start the installation process of Citadel, we can do that easily with the following command.

```
sudo apt-get install citadel-suite
```

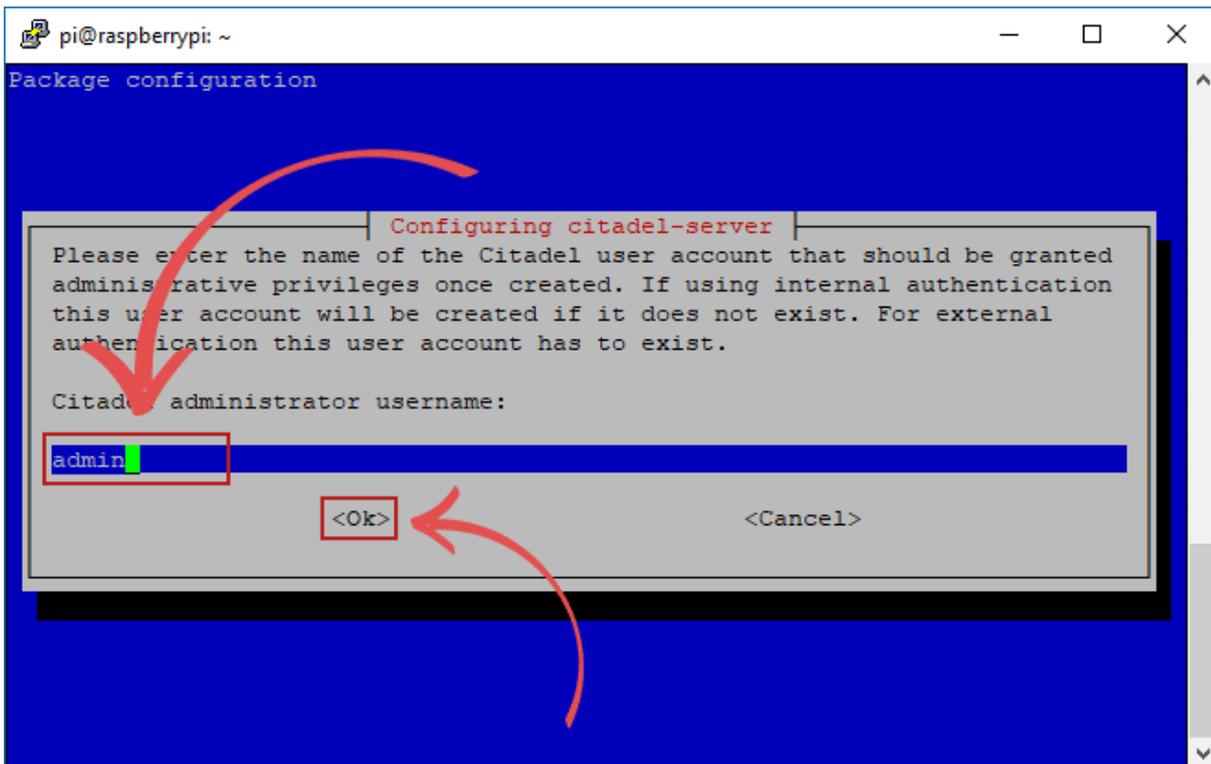
4. For this screen, leave the field set to 0.0.0.0 as this allows Citadel to listen on all incoming addresses. The only reason to change this would be if you are running multiple instances of Citadel. Simply press **enter** to continue.



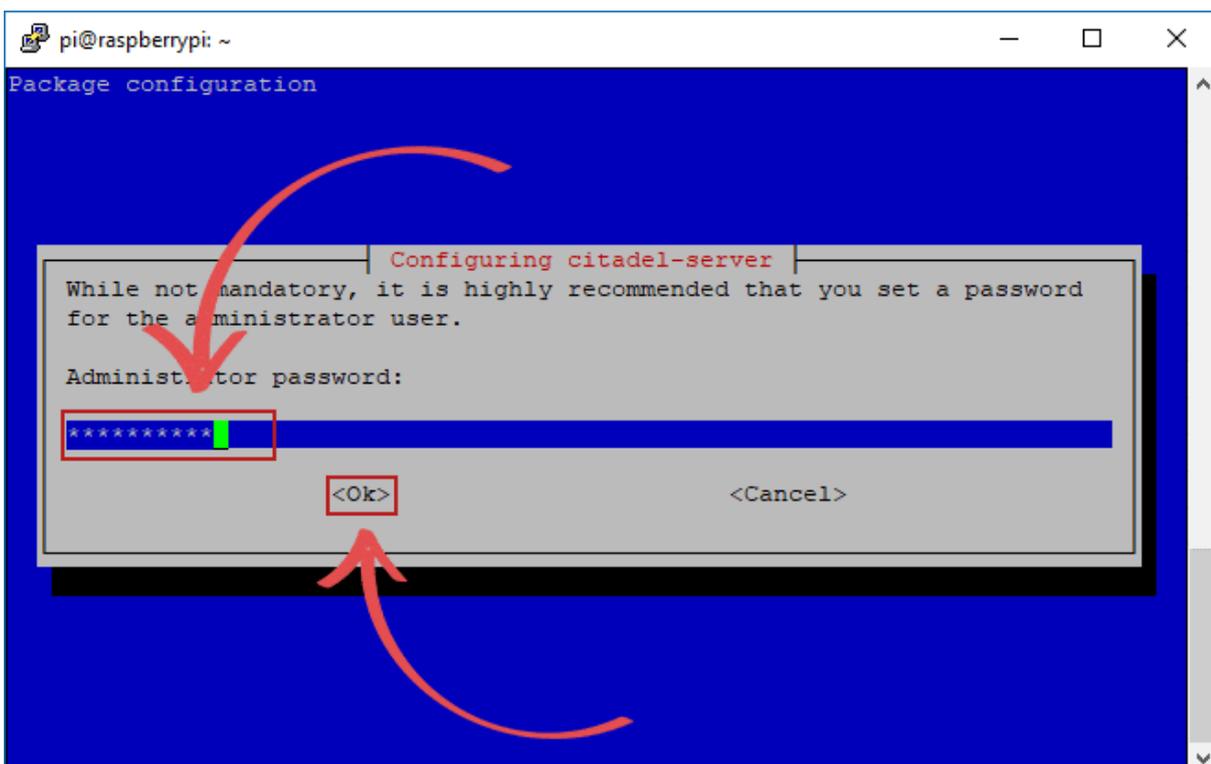
5. Now for the user authentication mode we're going to keep this set to **internal**. This allows Citadel to utilize its own internal user accounts database. Only change this if you have a need for the other three options. Press Enter to continue.



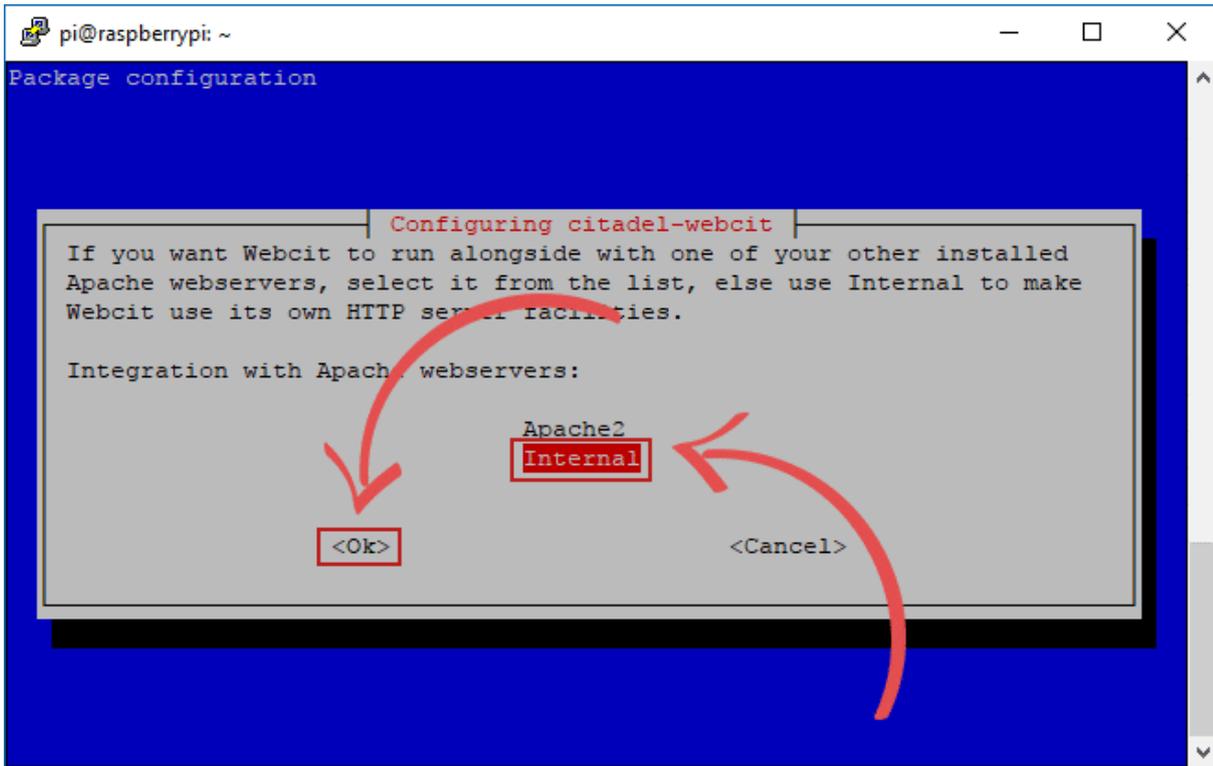
6. Here we get to set the name that we want for the administrator user, to keep this tutorial as simple as possible we will be just utilizing the default username **admin**. However, we recommend changing the username to something else since admin is usually really easy to guess and can lower the security of your server. Remember what you set here for later in the tutorial as this is what you will need to log into Citadels web interface. Press **enter** once you're happy with your choice.



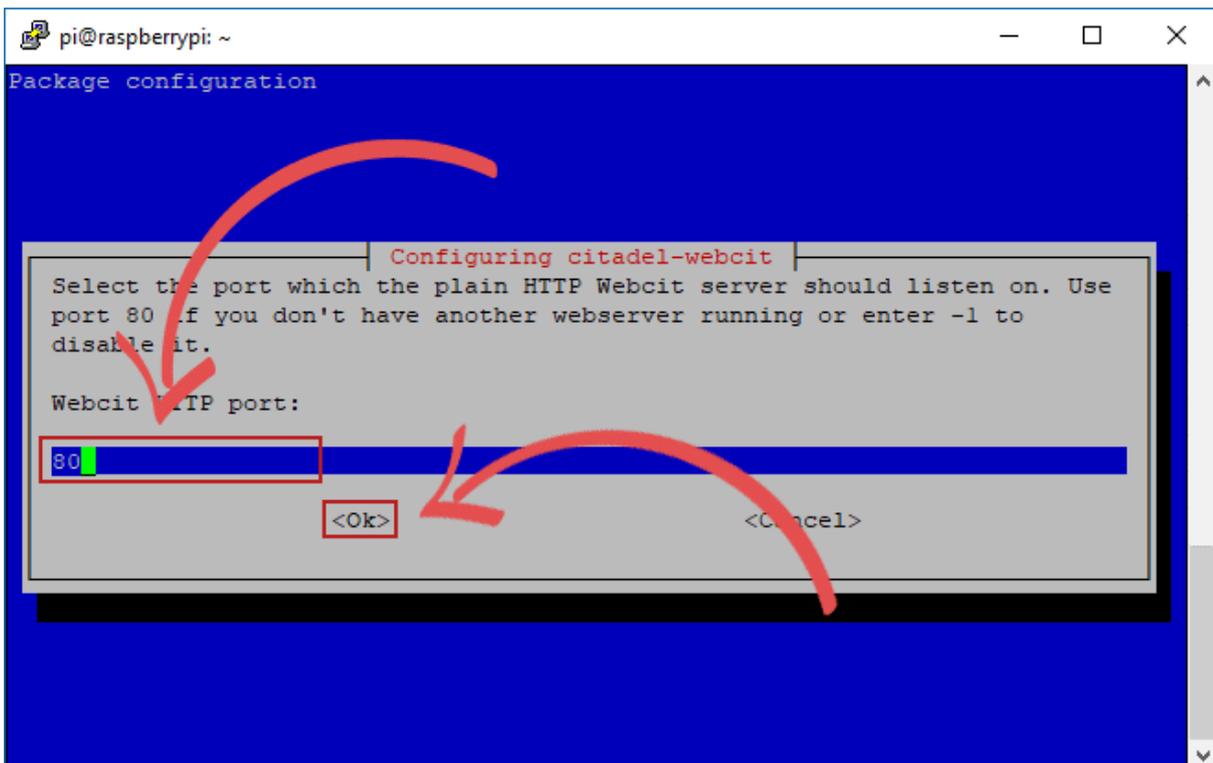
7. Now is the time to enter a password for the new administrator user, make sure you make this password difficult to guess.  
A secure password is crucial as anyone who gains access to your admin account could view all your emails and potentially damage your server's settings. It is even more crucial if you also plan on port forwarding port 80 to allow access to the web interface.  
Press Enter once you're happy with the password that you have entered.  
You will next be asked to verify this password, enter the same password again and press **enter**.



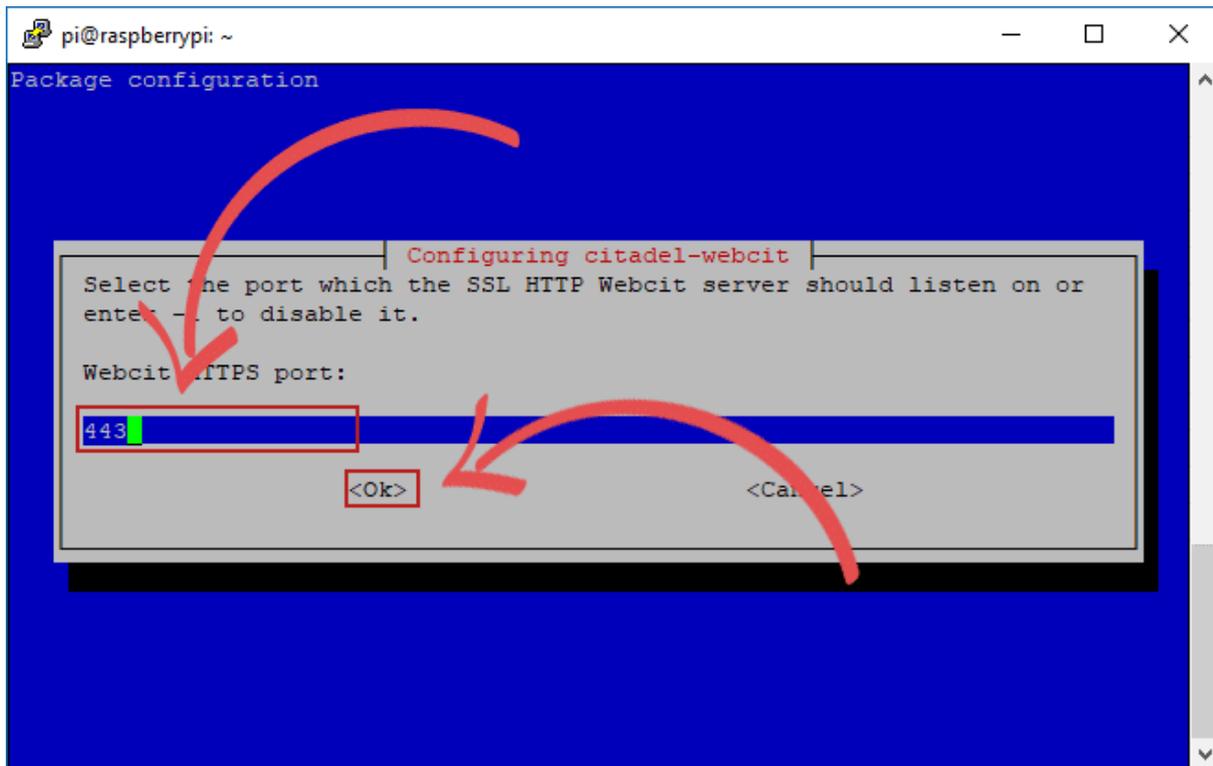
8. For the purposes of this tutorial we will be selecting internal to make Webcit utilize its own HTTP server facilities. However, if you're utilizing a Apache2 server already, then select Apache2. Use the arrow keys to select Internal then press **enter** to continue.



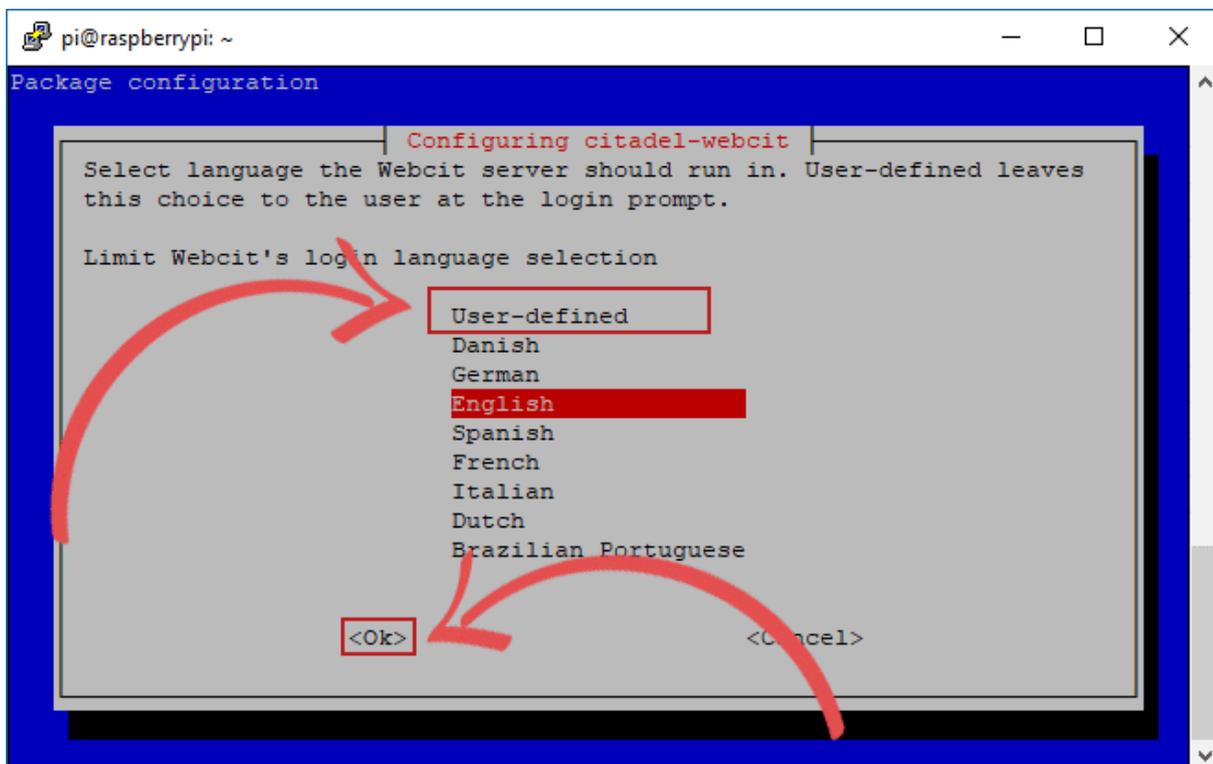
9. Now it will prompt you to select the port that you want to listen for a HTTP connection on, if you have no other webservers running it's safe to keep it set to 80. Otherwise try changing it to another port such as 8080. It's also important to know that some ISPs block port 80 so you will need to select a different one anyway. Press **enter** when you're happy with your choice.



10. Now it will prompt you to select the port that you want to listen for a HTTPS connection on, if you have no other webservers running it's safe to keep it set to 443. Otherwise try changing it to another port such as 4434. Press **enter** once you're happy with your choice



11. Now we can select whether we want to enforce a default language or let the user choose it on login. Easiest option is to just select "User Defined". When you're happy with your choice, press **enter**.



## Extra Steps

Sometimes the steps above will throw errors and not create the admin account we need to setup the email server correctly. If this is the case with you then you will need to do a few extra steps.

1. First we need to create the netconfigs folder and set the owner and group to citadel.

```
sudo mkdir /etc/citadel/netconfigs
sudo chown citadel:citadel /etc/citadel/netconfigs
```

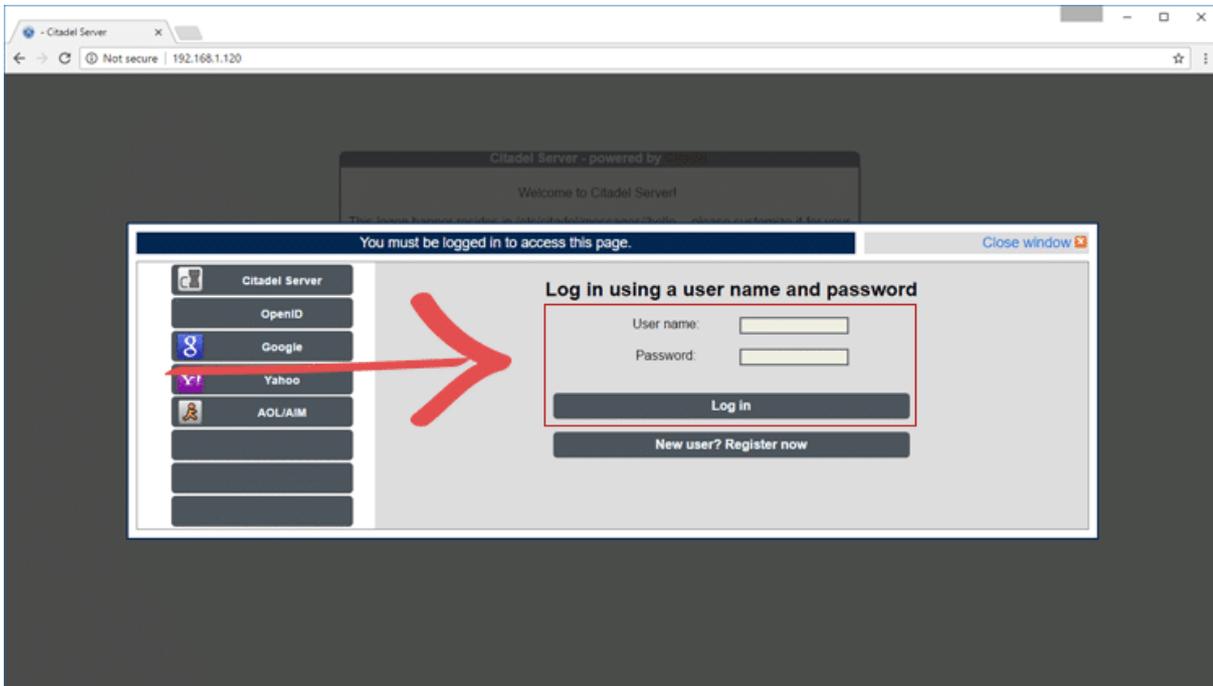
2. Next we need to restart Citadel so the previous changes are applied correctly. We also want to run setup again so that we can setup the admin account if it wasn't created correctly the first time.

```
sudo service citadel restart
sudo /usr/lib/citadel-server/setup
```

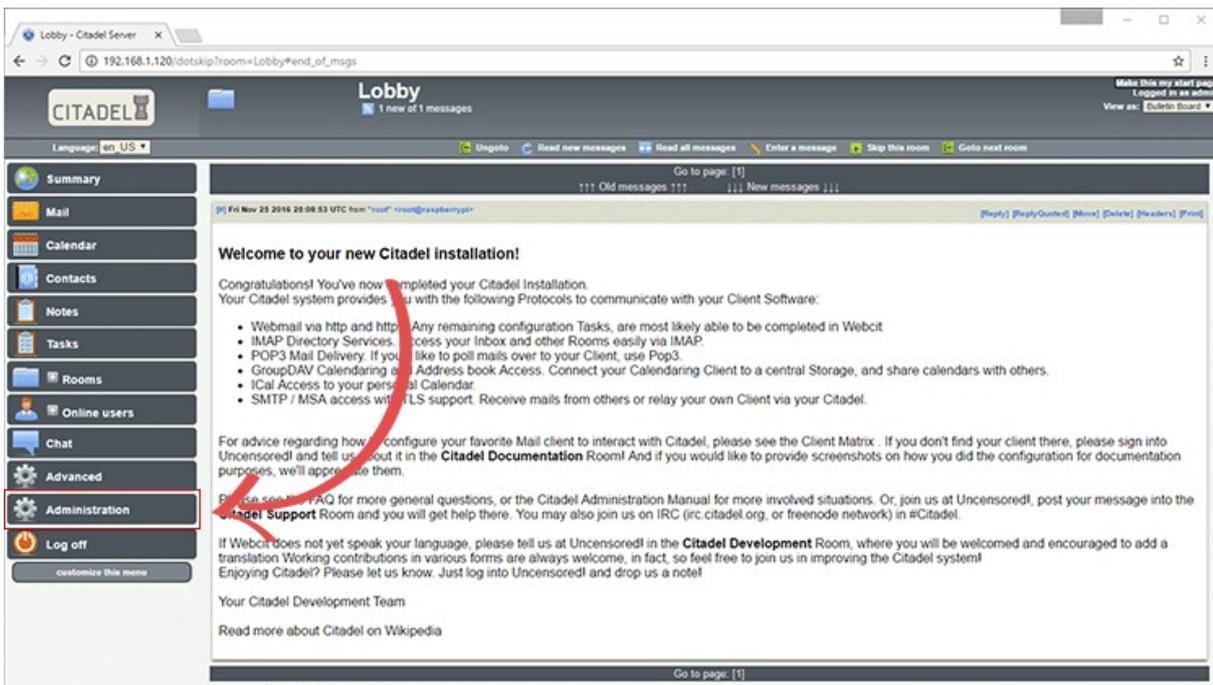
3. When you run the setup it will take you through a series of questions with the first being the name of admin. Set this to the username you wish to have for your admin account.
4. Next it will ask you to set a password for the administrator account. Set this to something secure so you're not easily hacked.
5. You can now skip through the rest of the questions. Simply press enter to skip the questions and not change anything.

## Setting up the Citadel Email Server

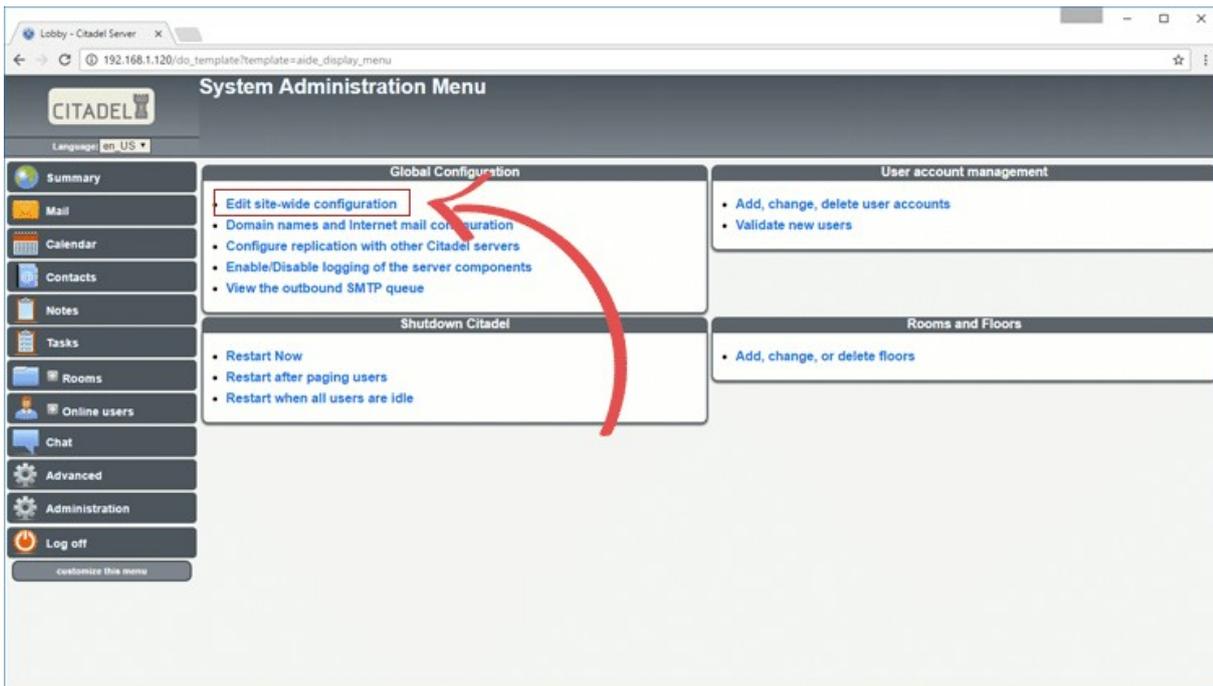
1. Now we need to load up Citadels web interface, to do this go to your Raspberry Pi's IP address in your favourite web browser. In my case, I would go to <http://192.168.1.105/>. If you don't have the Raspberry Pi's IP address on hand then you can type `hostname -I` into the Raspberry Pi's terminal to get your local IP address. Login by using the admin username and the admin password you set during the installation process.



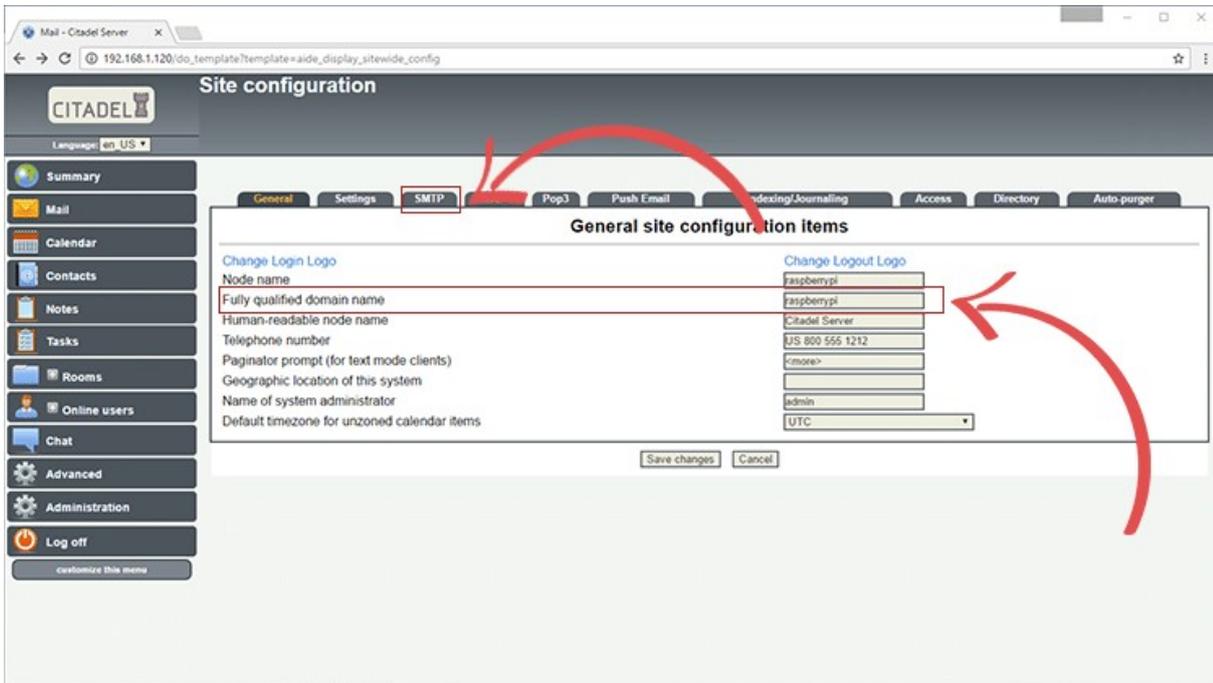
2. You should now be seeing the default dashboard, you can change the homepage by opening another page and clicking "Make this my start page" located in the top right hand corner. For now, just click on **administration** in the sidebar.



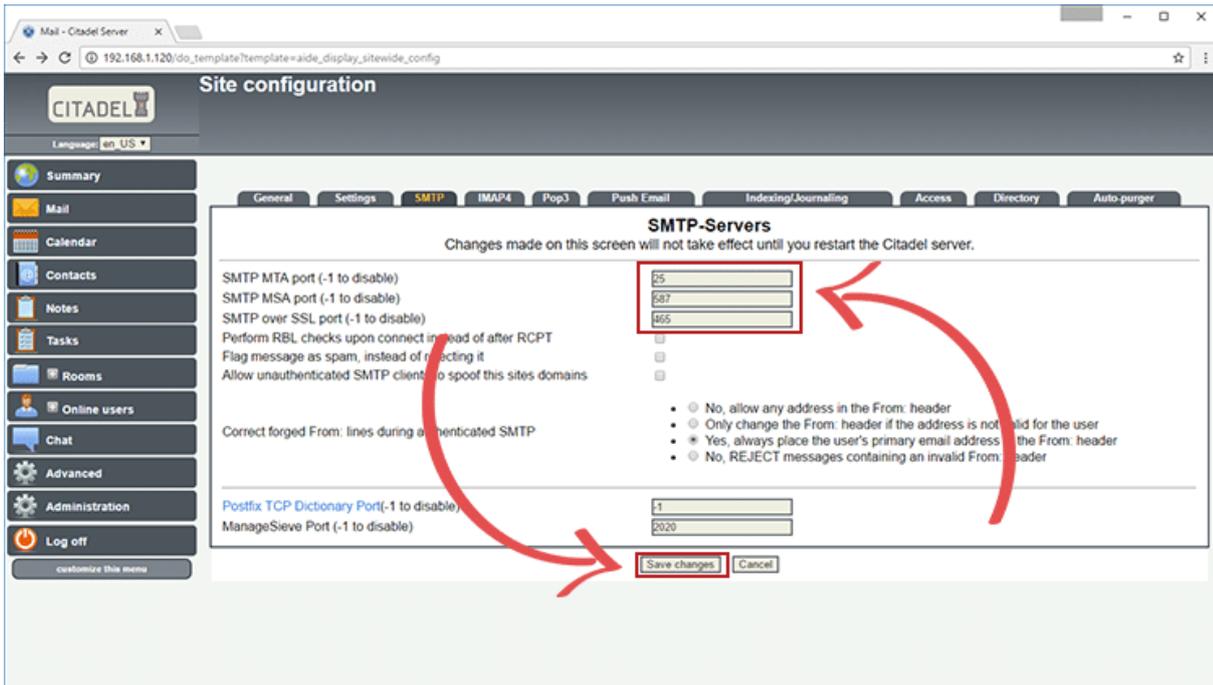
3. Now we are in the system administration menu, there's a wide variety of different options in here but what we are after is "Edit site-wide configuration", so click on that.



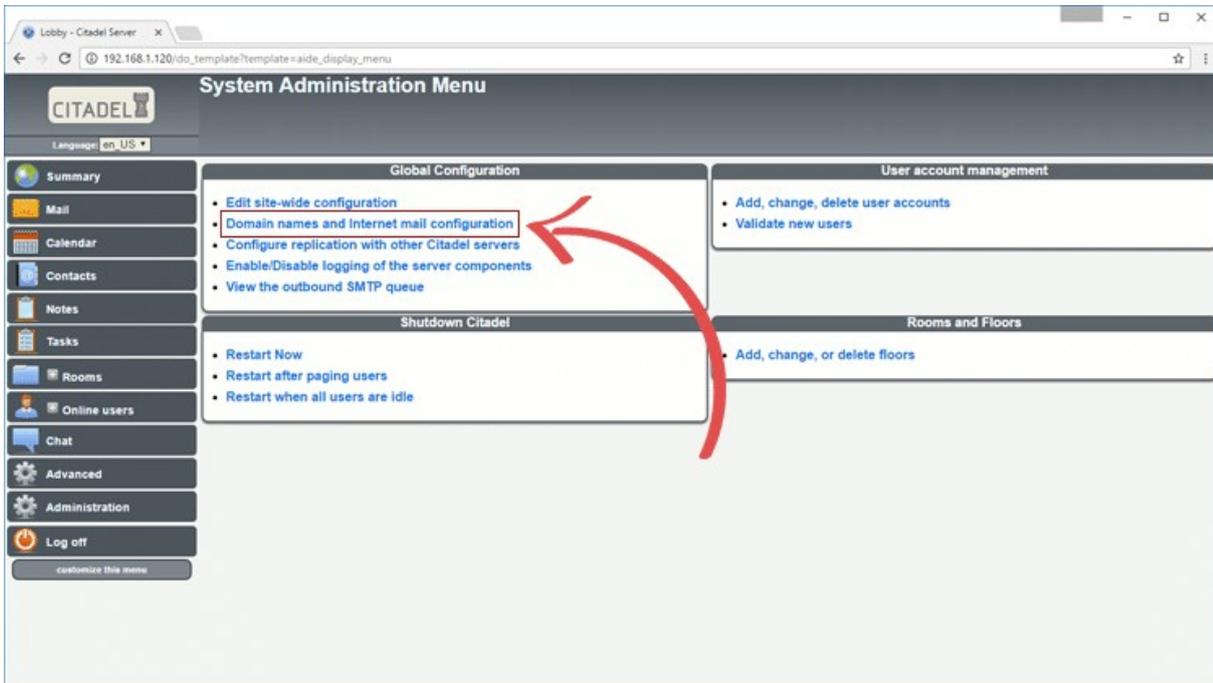
4. Now on this screen we will have to make a few changes to get everything working. Here we need to change the "Fully Qualified Domain Name" to the custom domain we plan to use for our email. In our case, we will set this to mail.pimylifeup.com. Once done, click on the "SMTP" tab.



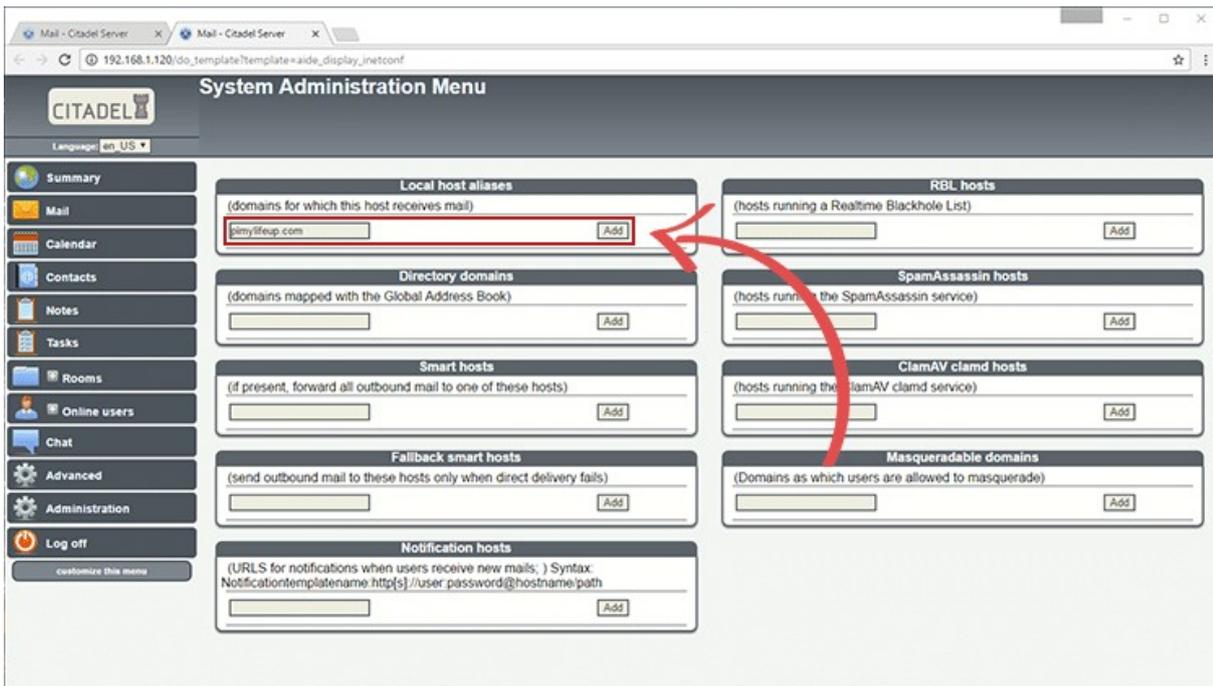
5. Now on this screen make note of the ports mentioned here, we will need to port forward these to allow the server to receive emails over SMTP. Once done, press the "Save Changes" button, then click on "Administration" in the sidebar.



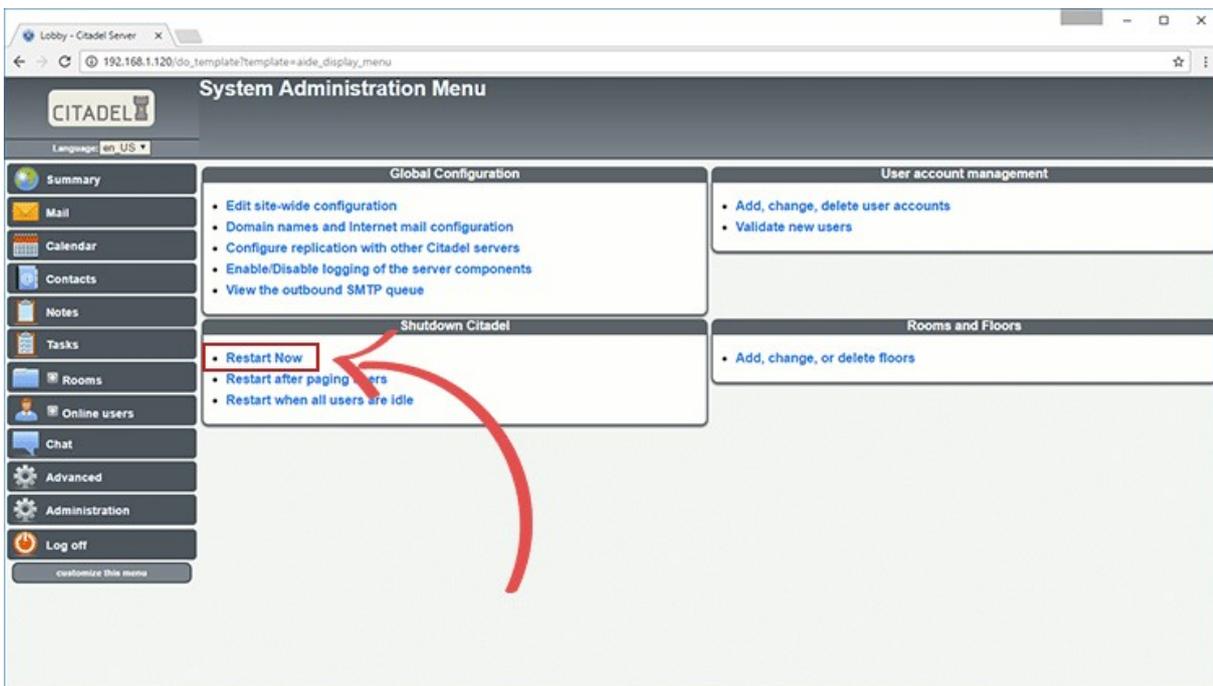
6. Now back in the "System Administration Menu", click on "Domain names and internet mail configuration". This will take us to the menu that we will be using to add additional domain names to Citadel.



7. Now on this screen, under "Local host aliases" type in the domain name you want to use then press the "Add" button. Once done, click back on "Administration" in the sidebar.



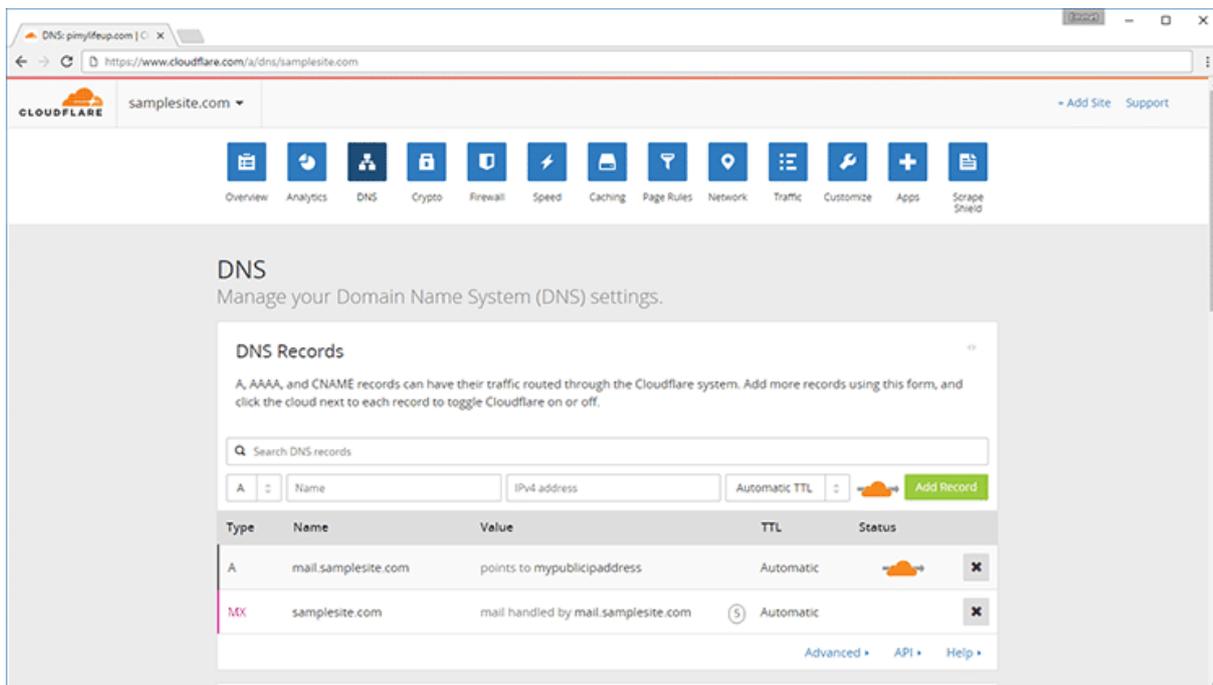
8. Now we are back on this screen, we need to click "Restart Now" under "Shutdown Citadel" this ensures that all our settings changes will now be loaded in.



## Setting up DNS for your Raspberry Pi Email Server

The easiest way to setup your DNS is to utilize a service like Cloudflare. Cloudflare offers a stable and redundant DNS service that can also be easily setup to also act as a dynamic dns service which is incredibly useful for anyone that is behind a dynamic IP address. An added advantage is that it also makes it incredibly easy to modify the DNS records, and updates are propagated much faster than most DNS services.

1. Go to your domain name on Cloudflare, then go to the DNS tab.
2. In here you need to add an A name record that points towards your network's public IP address. This will allow you to access your webmail client after you port forward. It is also needed so we can point the MX Record to it.
3. Now add an MX record that points towards your domain name that you set up as an A record in the previous step.



4. With that all done, you will now need to port forward the various ports from earlier. If you didn't make any changes then you will need to port forward the following ports 80, 25, 587 and 465. The last 3 of these ports should be your SMTP ports mentioned in the SMTP settings page from earlier.

If you're unsure on how to portforward on your router, you can try following our generic [port forwarding guide](#) that you can find on this website. Otherwise look up your router at the [port forward website](#).

Also, if you have a dynamic IP address you will need to [setup a dynamic DNS client](#), you can follow our guide on setting this up. Make sure you follow the steps on setting it up to work through Cloudflare.

## Setting up SpamAssassin for your Raspberry Pi Email Server

Now we will go through setting up Spamassassin, this is a tool that handles sifting through SPAM. It is a crucial tool to have installed on any private email server. It will add a bit more workload to the Raspberry Pi but it should be able to handle it just fine.

1. First let's install Spamassassin from the official packages by running the following command. The installation process of Spamassassin can take some time as it has to compile several things.

```
sudo apt-get install spamassassin
```

2. After Spamassassin has installed, we now need to make some changes to its configuration, run the following command to edit the first configuration file.

```
sudo nano /etc/spamassassin/local.cf
```

3. Update the lines mentioned below so that they match what we have written, if any of these lines have a # in front of it, then remove it. We will explain what each line does and why you should enable them.

This line makes Spamassassin modify the subject header of spam e-mails to include SPAM and the spam score that the Spamassassin system has assigned to it.

```
rewrite_header Subject [***** SPAM _SCORE_ *****]
```

This line tells Spamassassin to only modify the headers of an email and not make any changes to the actual body.

```
report_safe 0
```

Setting the required score low means you will initially get lots of false positives, but it will help you teach Spamassassin to know what emails are good and what emails are bad.

```
required_score 2.0
```

This next line sets Spamassassin to use a Bayesian filter, Bayesian is a way of estimating the probability of whether an email is Spam or not. It is a commonly used method that improves as the sample size increases.

```
use_bayes 1
```

The following line turns on the automatic learning for the Bayesian filtering.

```
bayes_auto_learn 1
```

Once you're all done, simply save and exit out of the file by pressing Ctrl+X and then Y and then pressing Enter.

4. Now there is one final file we need to edit to finish setting up Spamassassin, run the following command.

```
sudo nano /etc/default/spamassassin
```

The following line allows Spamassassin to work through systemctl, and means we can get it to boot at startup easily.

```
ENABLED=1
```

This line allows a cron job to automatically update the Spamassassin rules.

```
CRON=1
```

Saves and exit out of the file by pressing Ctrl+X and then Y.

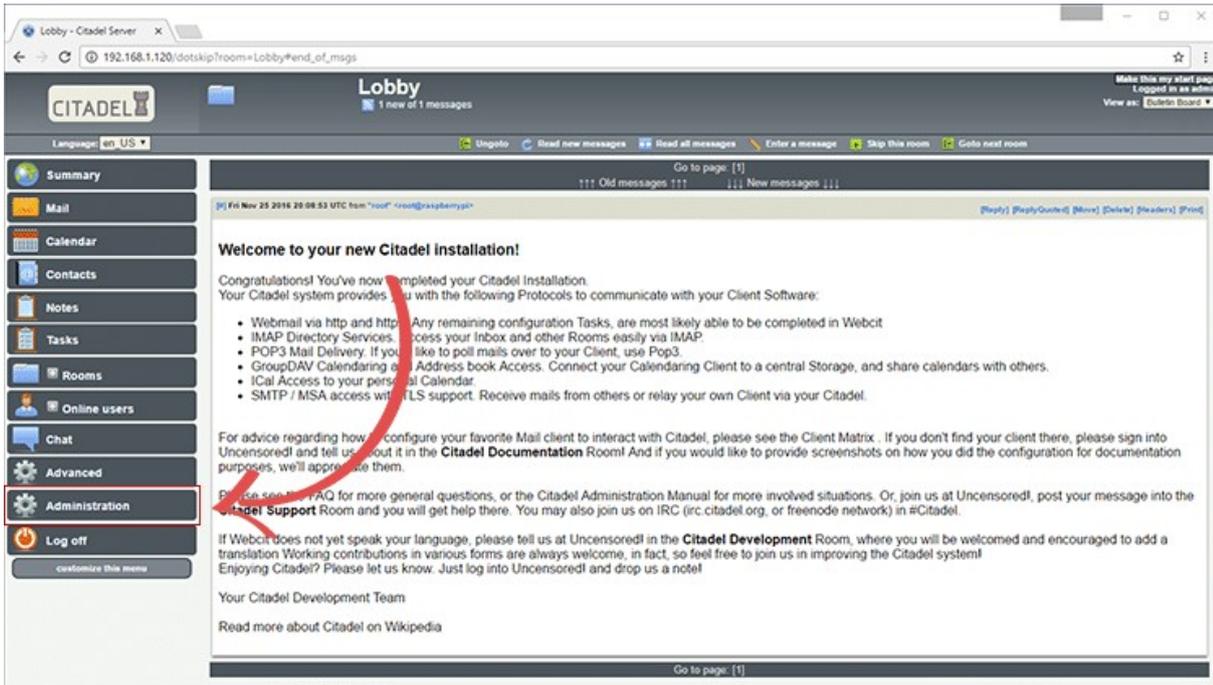
5. We can now start the Spamassassin daemon with the following command.

```
sudo service spamassassin start
```

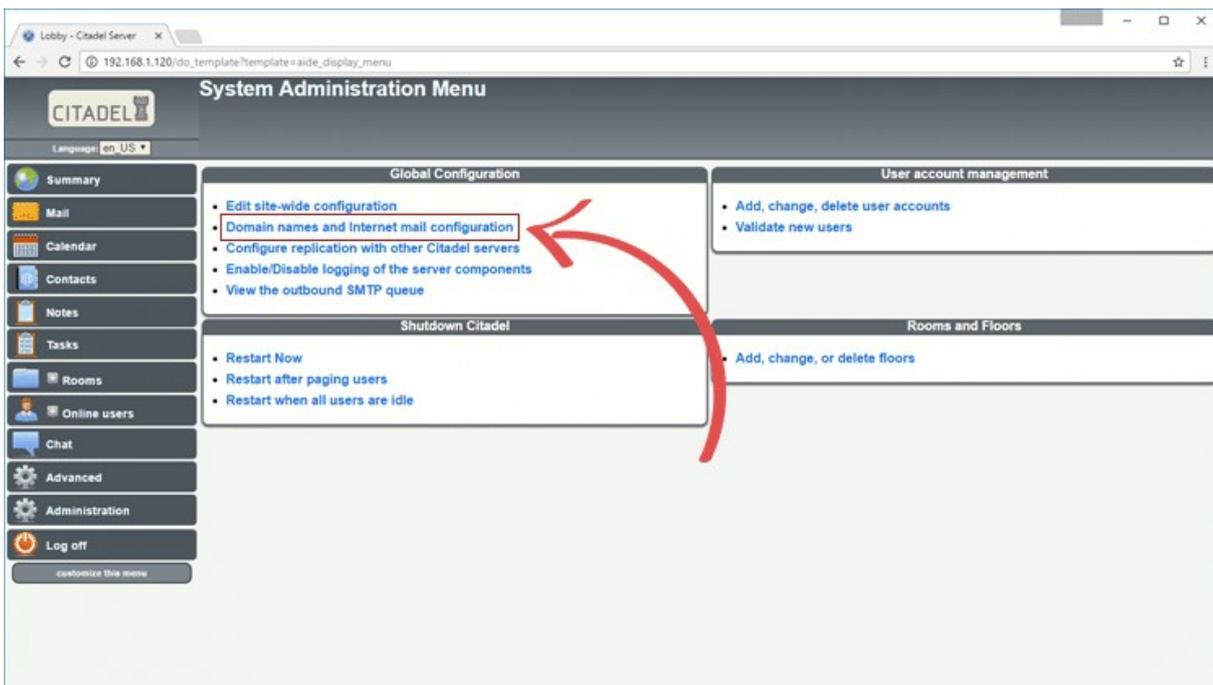
6. Now add Spamassassin to the init system with the following command so it automatically starts on boot.

```
sudo systemctl enable spamassassin
```

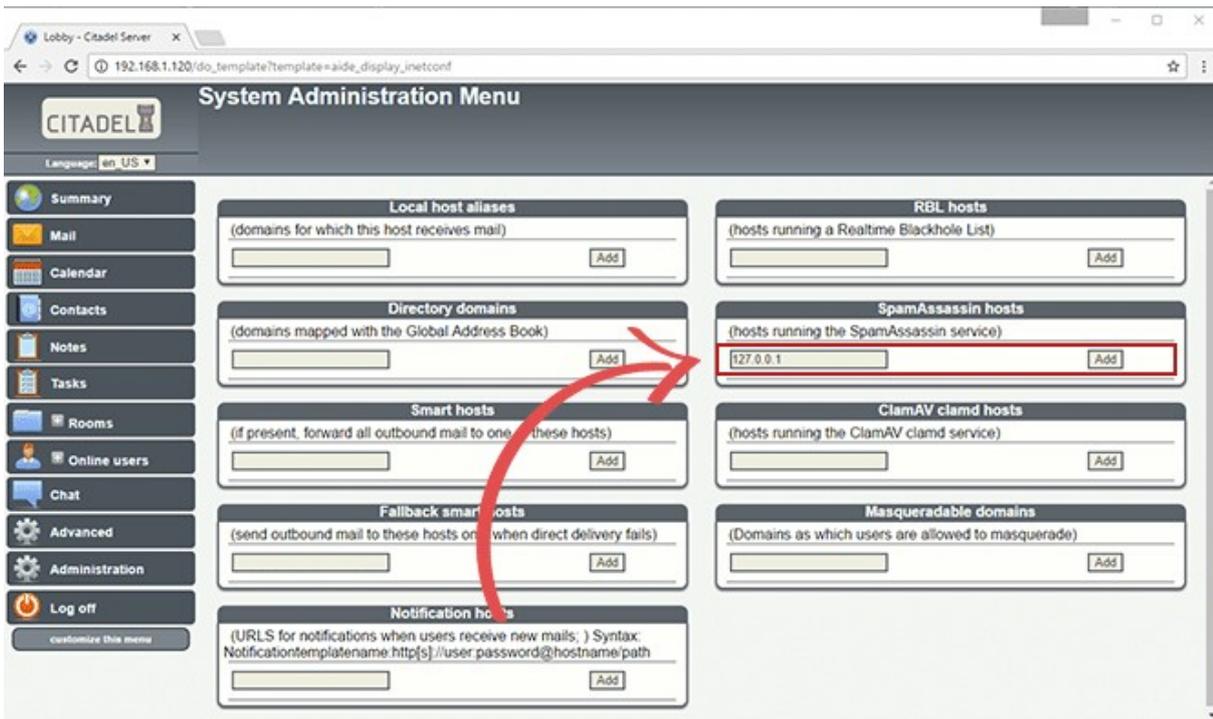
7. We now need to go to Citadels web interface and go back to the Administration screen.



8. Once we are back in here, we need to go to "Domain names and Internet mail configuration".



9. Now that we are in here, we need to type in the IP address, 127.0.0.1 under "SpamAssassin Hosts". Once you have typed it in, we can just press the "Add" button.



## Setting up ClamAV for Citadel

Now onto the final part of our Raspberry Pi email server installation guide, we are going to be setting up and installing the ClamAV software. This software will scan all your incoming emails for viruses and helps protect your Raspberry Pi from becoming infected.

1. Installing ClamAV is incredibly easy as it's already included in the Debian packages, we can just run the following command to install it.

```
sudo apt-get install clamav
```

2. Now we want to get ClamAV to download the latest version of its virus databases, we can do this by running the following command.

```
sudo freshclam
```

You may run into the error below, but we can deal with that:

ERROR: /var/log/clamav/freshclam.log is locked by another process

To deal with this error, we need to stop clamav, we can do this by running the following command. Then run sudo freshclam again.

```
sudo service clamav-freshclam stop
```

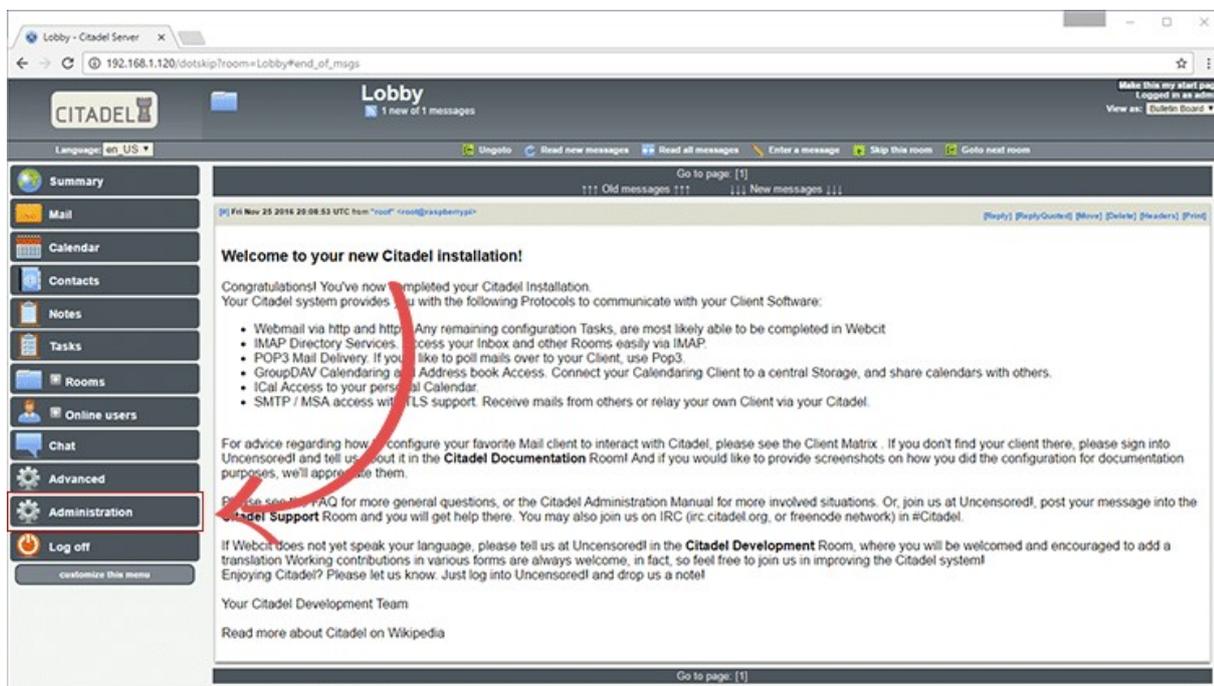
3. After the update has completed, run the following command.

```
sudo service clamav-freshclam start
```

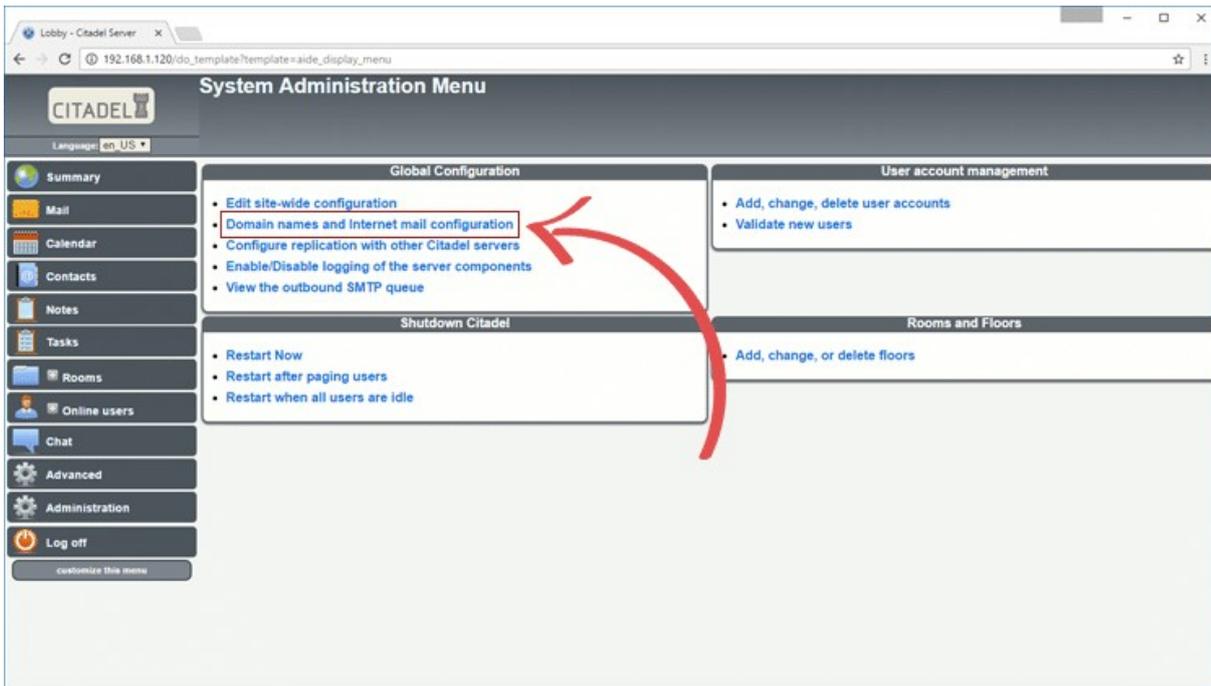
4. Now enable ClamAV with systemctl so its ensured to start on bootup.

```
sudo systemctl enable clamav-freshclam
```

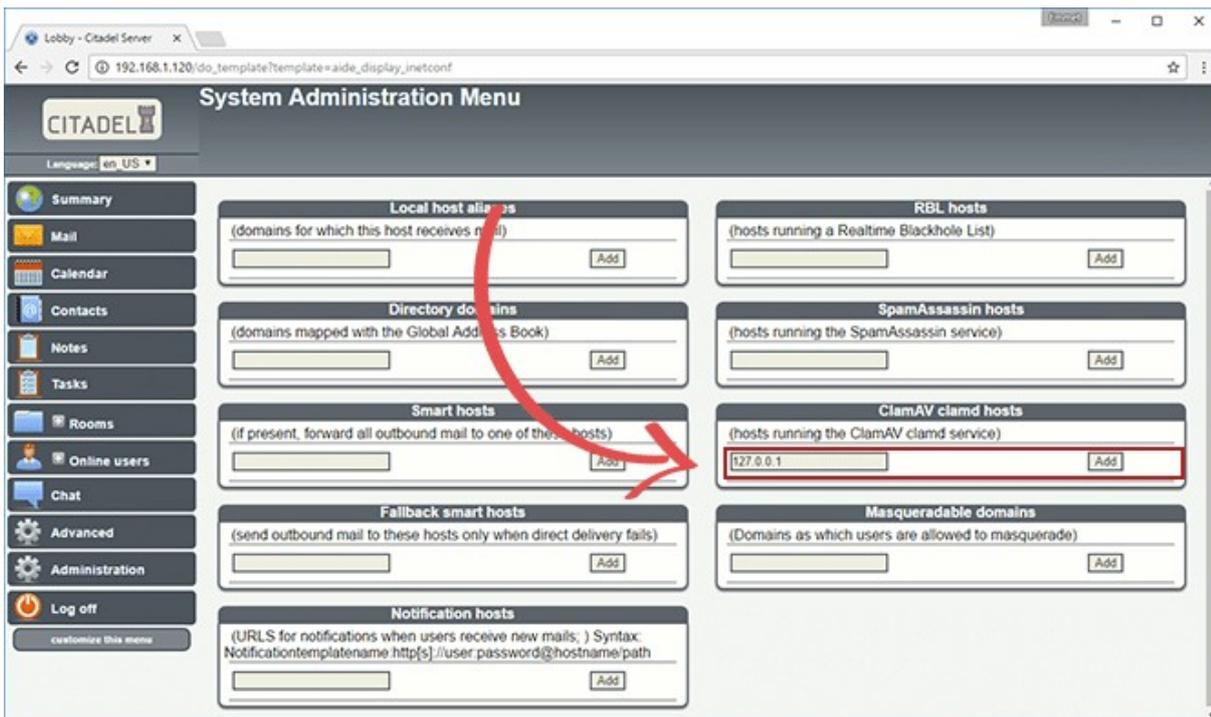
5. We now need to go to Citadels web interface and go back to the Administration screen.



6. Once we are back in here, we need to go to "Domain names and Internet mail configuration".



7. Now that we are in here, we need to type in the IP Address, 127.0.0.1 under "ClamAV clamd hosts". Once you have typed it in, we can just press the "Add" button.

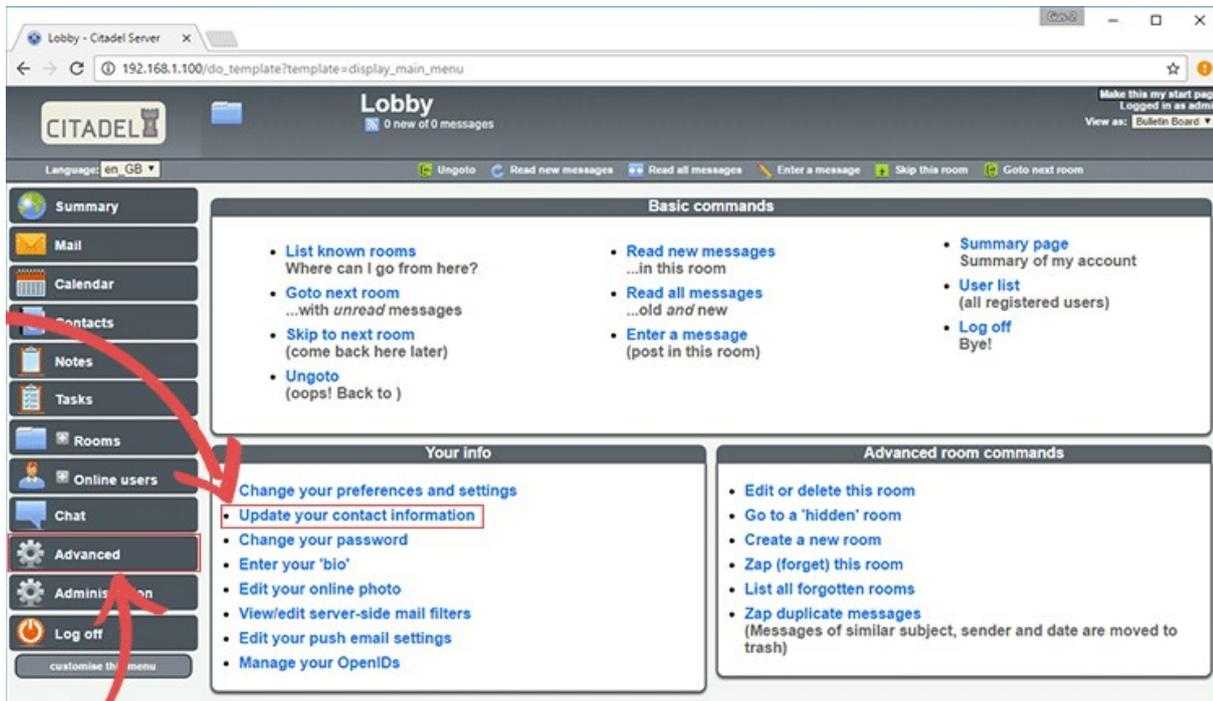


Hopefully by now you will have a fully functional email server that you're able to successfully connect to.

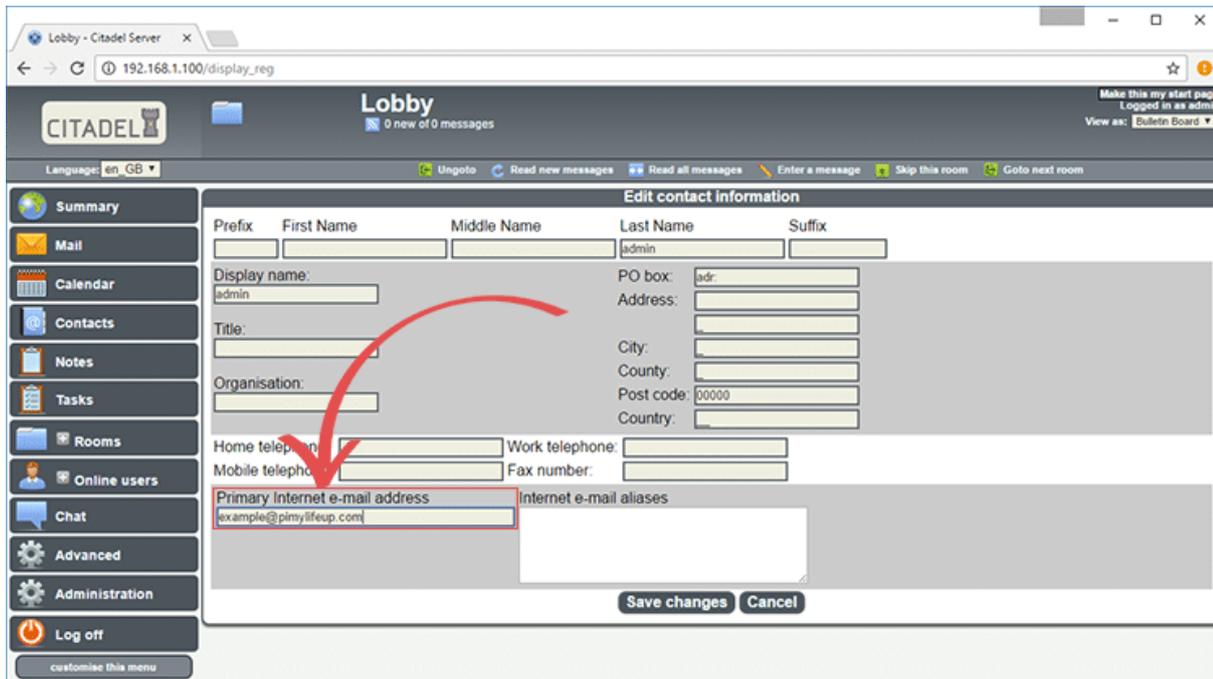
## Updating your User Account's Email Address

If you need to update your user accounts email address then this can be found in the advanced menu when you're logged into your chosen user.

1. Go to advanced and on this screen select "update your contact information".



2. Update your email address under "primary internet email address", you can also update other things such as your display name and other email addresses you want connected to this account (Internet email alias).



3. Once you're done select save changes.

## Troubleshooting

There are quite a few issues that you might run into whilst doing this tutorial with some being an easy fix and others a bit more difficult.

- Some ISPS will be blocking port 25 which means when you send emails they will fail. To get around this you will need to either get the port unblocked or setup citadel to use the ISP's SMTP server. You can find more information on [outbound email being stuck here](#).
- If you're not receiving email then this likely means your DNS has not been configured correctly. Go back to where your DNS is being managed and review the information.

I hope that this tutorial has shown you how to setup a Raspberry Pi email server and that you haven't run into any issues. It's certainly a great project for anyone who wishes to set up a cheap email server that you have full control over. If you have some feedback, tips or have come across any issues that you would like to share then please don't hesitate to leave a comment below.