OBFUSCATING CLEARTEXT

Cleartext is a bad thing, I have said so before (see here), any protocol which is cleartext is a bad thing as far as I am concerned. But I also know that sometimes you just do not have a choice, after all not everyone is as paranoid as me (remember just because you do not catch them, does not mean they are not out to get you!), so what do you do then? Well one very nice answer is to use *stunnel*, this application creates a SSL-enabled server which forwards the data to another port, it can also run under windows - but we will look at the linux side of things here. In this article we will look at using this to secure POP3..

What do you need?
Well you need *openssl* and -of course- *stunnel*, once you download it you do the normal *./configure; make; make install* dance. We then create a folder for use (if it does not already exist) */etc/stunnel*. Then we can create a certificate file for use by *stunnel*, a *.pem* file. You can use these commands to create one if you do not already have your own certificate..

```
openssl req -new -x509 -days 3650 -nodes -out /etc/stunnel/stunnel.pem -keyout /etc/stunnel/stunnel.pem
chmod 0600 /etc/stunnel/stunnel.pem
```

Now we need to create a configuration for *stunnel* to use, try this as a starter */etc/stunnel/stunnel.conf*..

```
cert = /etc/stunnel/stunnel.pem
#foreground = yes
#debug = 7

[pop]
accept = 995
connect = 110
```

You can see we specify the certificate file first, for testing purposes you may want to uncomment the next two lines until you know *stunnel* works 100%. By uncommenting the lines you get *stunnel* to run in the foreground and do lots of logging. The *[pop]* section is very simple, the port to accept connections on, and then the connect port to send those incoming connection to. You can have multiple sections, for example you could have one for *smtp*. Once thats done you run it like so..

```
stunnel /etc/stunnel/stunnel.conf
```

If all goes well, when you do a *netstat -natlp* you should see your new listening port. You can get this to run on startup anyway you want. Now that the server is ready, you can change your email clients to use SSL connections with POP3, by default the POP3 SSL port is *995*, if you have made it something else you may need to specify it. And thats it, your POP3 cleartext traffic is now encrypted. Pretty much all email clients should be able to do POP3 over SSL.

Final Words
Two things to remember when doing this. First, this will encrypt the traffic only, it does not help further secure a weak password. Secondly, your mail logs will show any POP3 logins using this as coming from *localhost*, so if that is a problem for your specific setup be careful. *Stunnel* has many other options for you to try out, as always, have fun and learn.