

# STunnel

## STunnel

Where to get the source:

- <http://mike.daewoo.com.pl/computer/stunnel/>
- <http://www.stunnel.org/>
  
- stunnel-3.8p4 is the latest version

# STunnel

You'll also need:

- SSL Library
  - OpenSSL <http://www.openssl.org/>
  
- PThreads
  - pth <http://www.gnu.org/software/pth/pth.html>

## BE CAREFUL!

The web sites are surprisingly lacking information on some key issues:

- What are X509 Certificates
- How does one get a Certificate
- How should Certificates be handled
- What type of security do they provide

There are a few sources of information

- <http://www.ultranet.com/~fhirsch/Papers/wwwj/article.html>
- [http://www.ultranet.com/~fhirsch/Papers/cook/ssl\\_cook.html](http://www.ultranet.com/~fhirsch/Papers/cook/ssl_cook.html)

## What is it?

- A tool to encrypt and (sometimes) authenticate TCP connections.
- A tool to encrypt existing TCP services without recompiling them.
- Runs on UNIX, Windows and Mac(?)

## What is it not?

- A tool to encrypt non-TCP connections (UDP, IPX, etc, etc)
- Truly easy to manage for large user bases while maintaining a high level of security.
- The answer to all of your problems.

## Getting Started

Get a certificate for your service

- Get one from a CA
- Create your own with OpenSSL

- `openssl req -new -x509 -days 365 -nodes  
-config stunnel.cnf -out stunnel.pem  
-keyout stunnel.pem`

(stunnel.cnf is found in the stunnel source distribution)

## Getting Started

The certificate and UNENCRYPTED private key for the service needs to be installed where stunnel expects it to be. Or use the '-p' flag to override the default location.

You can also add client's certificates to a directory for authentication with the '-a' and '-v' flags.

# Running STunnel

## Server Side

```
stunnel -d 3333 -l /bin/cat -- cat /etc/motd
```

## Client Side

```
stunnel -c -r 3333
```

This causes 'cat /etc/motd' to be run on the server and the results displayed on the client.

# Running STunnel

## Server Side

```
stunnel -d 3333 -r localhost:pop3
```

## Client Side

```
stunnel -c -d pop3 -r SERVER:3333
```

You can now connect your POP3 client to the computer running the stunnel client. The connection will be forwarded over port 3333 to the stunnel server's POP3 server.

# Running STunnel

## Server Side

```
stunnel -d 5555 -L /usr/sbin/pppd -- pppd local
```

## Client Side

```
stunnel -c -r SERVER:5555 -L /usr/sbin/pppd -- pppd local
```

This creates an encrypted VPN. Adding client certificates and '-v 3' to both sides gives full authentication on both sides of the connection.