Encrypt the talk between clients and servers who don’t

“The stunnel program is designed to work as SSL encryption wrapper between remote clients and local (inetd-startable) or remote servers. The concept is that having non-SSL aware daemons running on your system you can easily set them up to communicate with clients over secure SSL channels.

stunnel man page
Ordinary ssl-unaware applications

- Client application
- Socket API
- Transport
- Network
- Data link
- Physical

SSL-aware applications

- Client application
- Socket API
- Transport
- Network
- Data link
- Physical

SSL

- Server application
- Socket API
- Transport
- Network
- Data link
- Physical

SSL

- Encrypt

© David Morgan 2006-10
stunnel – 3 TCP conversations

app viewpoint: stunnel-oblivious
Ports: non-stunnel scenario

- client application
  - talk to remote:60000
- server application
  - listen to 60000

stunnel – 3 TCP conversations

- ssl-unaware client
  - talk to local:2000
- stunnel
  - listen to local:2000
  - talk to remote:30000
- stunnel
  - listen to 30000
  - talk to 60000
- ssl-unaware server
  - listen to 60000
Vanilla config files

```plaintext
# stunnel client
client=yes

[stunnel service name]
  accept = 127.0.0.1:2000
  connect = 192.168.3.12:30000

# stunnel server at 192.168.3.12
cert = /etc/stunnel/stunnel.pem

[example service name]
  accept = 30000
  connect = 60000
```

stunnel – genl case topology
stunnel server needs certificate

- create it with
  
  ```
  cd /etc/stunnel
  openssl req -new -x509 -days 3650 -nodes -out stunnel.pem -keyout stunnel.pem
  ```

- reference it in stunnel server’s config file
- [http://www.stunnel.org/faq/certs.html#ToC5](http://www.stunnel.org/faq/certs.html#ToC5)

---

With and without

Both work

via stunnel on local port

to server on its port (no encryption)

see concurrent capture files
With and without

Both work

stunnel’s not really a tunnel

- stunnel is a conversation endpoint
- and a (different) conversation startpoint
- arriving packets are stripped of header at endpoint
- their content repackaged, new header, at startpoint
- headers do not nest/accumulate as in tunnels
True tunneling

- Headers accumulate

Payload forward/relay/proxy

- Headers replace each other
info

- http://www.stunnel.org/
- http://freshmeat.net/articles/view/1781