

Secure Transfers

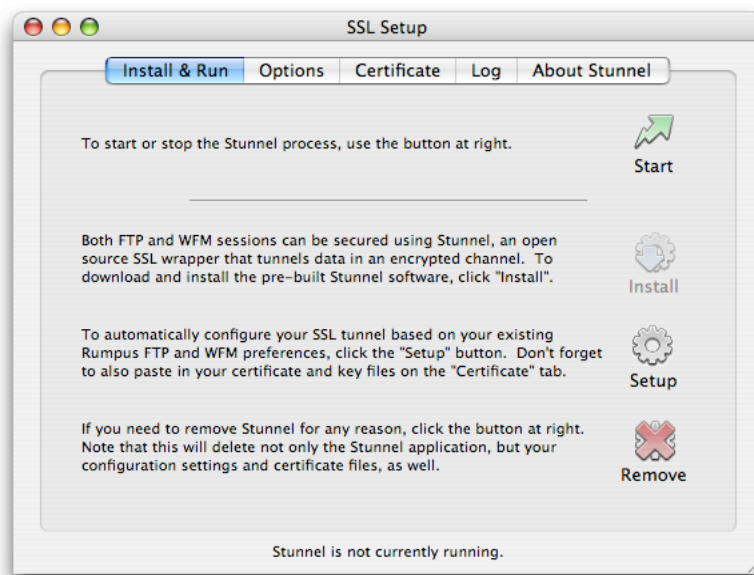
Securing file transfers using encrypted connections across the Internet.

Rumpus supports SSL (Secure Sockets Layer) encryption for both HTTP (Web) and FTP connections using Stunnel, an Open Source SSL tunneling program. While Rumpus will download, install and configure Stunnel for you, SSL encryption is not trivial, especially when applied to FTP connections. You will need to create or purchase an SSL Certificate, and clients will need to use compatible client software in order to transfer files securely.

Thanks to the need for secure transactions in e-commerce, and the fact that HTTP (the Web) is a simpler protocol from a networking perspective, secure file transfers via the Rumpus WFM are consistently and almost universally supported by modern Web browsers. Unfortunately, there is no consistent and widely implemented encryption standard used by FTP clients. Options for encrypted FTP exist, but they can be complicated to implement and will restrict the choice of FTP client software that can be used. The most commonly used encryption methods used by FTP clients are SFTP and FTPS, with Rumpus supporting FTPS (SSL-encrypted FTP).

Installing The SSL Tunnel

All configuration of the SSL tunnel is done using the “SSL Setup” window, accessible from the “Setup” tab of the main Rumpus configuration window. Basic installation and setup of the tunnel is very simple, and is performed using the “Install & Run” tab, shown here.

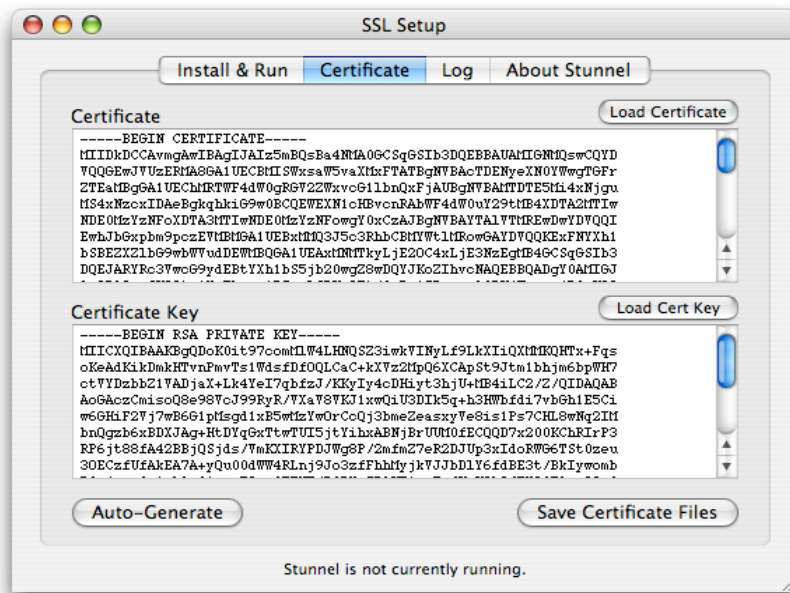


To begin, click the Install button. Rumpus will download the Stunnel binary from the Maxum software support server and install it onto your local system, so this process may take a minute or two, depending on the speed of your Internet connection.

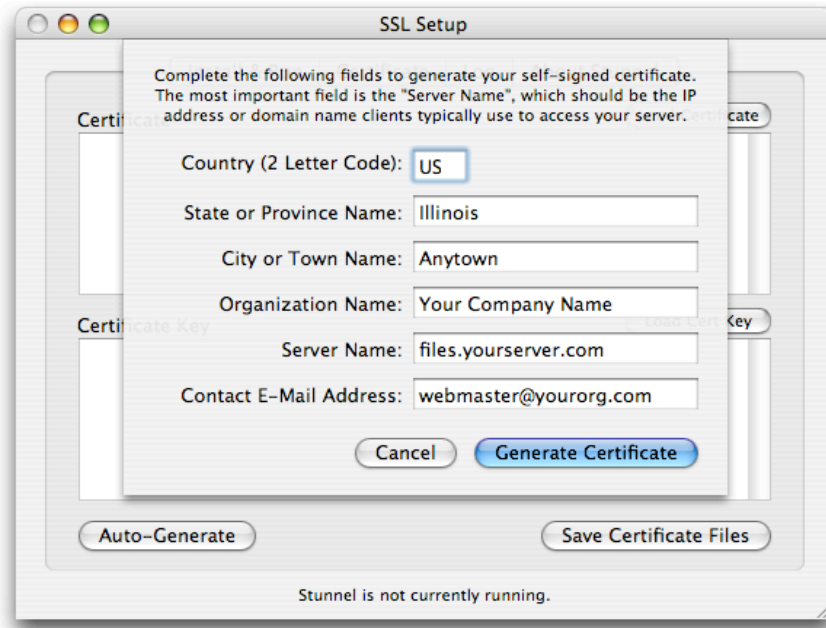
Next, click “Setup”. Your Rumpus configuration settings, including the services that have been enabled, server ports used, etc. will be used to automatically create the needed Stunnel configuration file. If you ever change the Web server, FTP server, or passive mode port range, the Stunnel configuration file can be regenerated by clicking the Setup button again.

Apply A Certificate

Before you can start the Stunnel service, you need to generate and apply an SSL certificate and certificate key file. This is done using the “Certificate” tab on the SSL Setup window.



The easiest way to generate a certificate in Rumpus is to use the Auto-Generate function. Clicking the “Auto-Generate” button starts the process by bringing down the SSL information sheet, as shown below.



The information you supply will be encoded into your certificate, and will be visible to end users who choose to display the details of your company when they connect via the secure connection. For this reason, you should supply the best information available, but there is no right or wrong answer for any field except the “Server Name”. For this field, be sure to supply the domain name or IP address people will use when connecting to your server. Web browsers will compare the “Server Address” of the certificate with the address used to connect in order to confirm that the certificate is valid.

Why Do I Need A Certificate?

SSL provides not only encryption of file transfers, but server authenticity as well. In other words, not only will your clients be assured that their data is transferred using an encrypted connection, but that the file is being sent to the intended server, and that the server is owned by a reputable organization. This requires that you obtain a “certificate”, which is a digital file that describes your organization and server and is “signed” by a trusted authority. You can pay for an authority to provide you with a trusted certificate, or you can “self-sign” the certificate. Certificates signed by a known authority will usually be automatically trusted by common Web browsers, while self-signed certificates will cause browsers to display a warning message to users declaring that they are connecting to a non-trusted server.

Starting Service

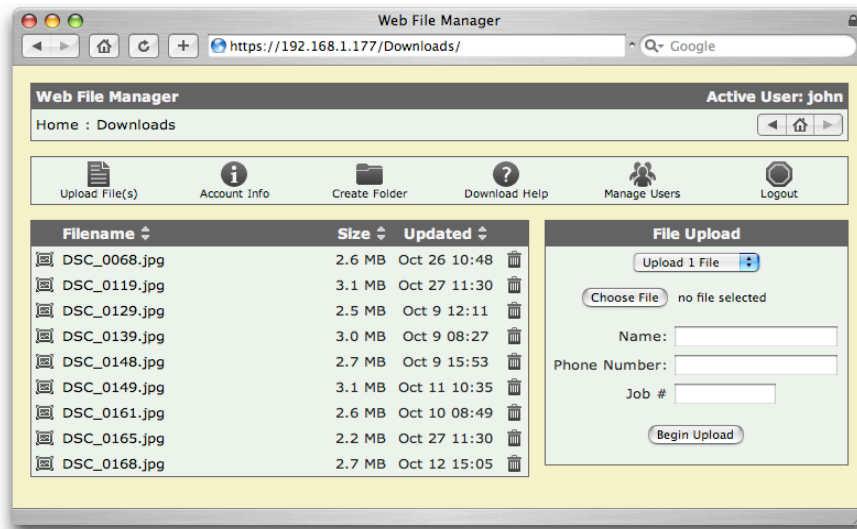
The SSL tunnel is now ready. Flip back to the “Install & Run” tab and click “Start” to start the Stunnel process. If you want Stunnel started whenever the Mac is booted, click the “Automatically Launch Stunnel At System Start” checkbox (found on the “Options” window), and Rumpus will create the needed Stunnel startup script.

After starting Stunnel for the first time, be sure to check the error log for problems. Flip to the “Log” tab and click the “Reload” button to review the most recent entries in the Stunnel log. The log file will include details about Stunnel’s ability to start and confirm your certificate, and if a problem occurs it will be reported.

Connecting Via Web Browser

Even if your users primarily access your Rumpus server via FTP, we recommend that you first test your SSL encrypted server using a Web browser. Browsers almost universally support HTTPS (HTTP encrypted using SSL), and do so in a standard way that is simpler than FTP. Once you have confirmed that the SSL tunnel is functioning correctly, you can then move on to connecting with a compatible FTP client.

On another computer on your network, open a Web browser and enter the connection URL as “https://your.server.address/”. Note the “s” in “https://”, which tells the browser to make the connection via the SSL encrypted channel. Most Web browsers will display a lock icon somewhere on the display to indicate that the connection is secure. The WFM session will continue normally, the lock icon being the primary visible evidence that data is now being transferred securely.



Connecting Via FTP Client

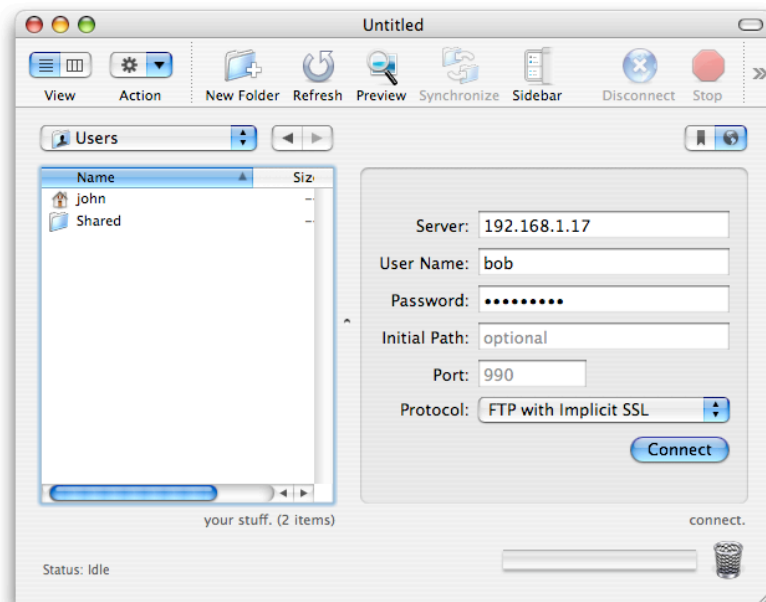
Unfortunately, there is no single encryption standard for FTP connections. Tunnelling FTP inside an SSL connection is known as FTPS, and is one of the leading alternatives. It is also the best documented standard available, and is supported by a number of popular FTP clients for both Mac and Windows PCs.

Assuming your first test will be from a Mac, we suggest using Transmit, available from Panic Software at:

<http://www.panic.com/>

If you don't already have a copy of Transmit on a local Mac, download a trial copy from Transmit's Web site.

When Transmit starts, the transfer window allows you to specify the connection information to be used in establishing the session. Enter the server address, username and password as normal, then set the "Protocol" to "FTP with Implicit SSL". Transmit will set the port to the correct FTPS default of 990 automatically. Finally, click "Connect" to connect via the secure channel.



Both the control and passive mode data connections will now be encrypted, though as with Web sessions, very little difference will be apparent to the end user. To confirm that the session is indeed encrypted, check the Fetch transcript (from the "View" menu, choose "Show Transcript"). In the transcript, just after user login, you will notice a command sequence that looks like this:

```
Cmd: PBSZ 0
Cmd: PROT P
200: PBSZ=0
      PRIVATE data channel protection level set
```

The “PBSZ” and “PROT” commands are unique to an FTPS session, and indicate that the client and server have negotiated a secure channel for data connections (though again, the control connection is also encrypted).

Making secure connections from PC clients is, of course, also possible. The popular FTP clients CuteFTP and WS_FTP, for example, both support FTPS in both their basic and professional versions. When connecting using CuteFTP, choose the connection method of “FTP with TLS/SSL (Port 990-Implicit)”. For clients using WS_FTP, the connection option is called “FTP/Implicit SSL”.

Ports And Firewalls

Each standard Web and FTP port used by Rumpus for providing service needs to map to a corresponding SSL encrypted port. In other words, standard ports for HTTP, FTP control and FTP data ports all need to be duplicated for secure access.

By default, Rumpus will use the standard HTTPS and FTPS ports, 443 and 990 respectively, for accepting secure Web and secure FTP connections. Rumpus also requires that the port range for passive mode secure data connections be exactly the port range for standard passive mode connections, plus 1000.

For example, a Rumpus server configured for FTP access on port 21, Web access on port 80 and a passive mode port range of 3000-3008 would also need to be configured to allow incoming connections on ports 443, 990 and 4000-4008.

On private LANs bridged to the Internet, your router will need to be configured to forward these additional ports to the Rumpus server, just as it is for the standard ports. And of course, if a firewall is enabled on the Rumpus server, it will also need to be configured to allow connections on the extra ports.

Obtaining A Trusted SSL Certificate

While self-signed certificates are usually the best way to get started and test SSL encrypted services, most professional organizations will eventually need to purchase a certificate from a verified signing authority. In this case, you will generate a “Certificate Signing Request” (CSR), which will be sent to the signing authority. The authority will then confirm your company information and return to you a verified certificate.

For details on generating the CSR, see the instructions provided by the authority from which you will be purchasing your trusted certificate. Note that the authority will return to you a certificate, not a key file. The private key is generated when you create the CSR, so be sure to put a copy in a safe place for installation once you get your certificate.

Use the "Load Certificate" and "Load Cert Key" buttons on the “Certificate” tab of the SSL Setup window to load the verified certificate and the key file that was generated along with your CSR. In each case, simply click the button, then navigate to and select the ".cert" and ".key" file as needed. The certificate and certificate key will then be loaded into the text area for you.

If you purchased your certificate from a signing authority, you may also have been provided with an authority certificate. This can be pasted into the “Certificate” text area immediately following your own. Be sure to include the full text of the signing authority certificate, including the “begin” and “end” lines, positioned after the “end” line of your own certificate.

When the full text of the certificate and key have been pasted into the text areas, click the “Save Certificate Files” button. Rumpus will then store both the “Certificate” and “Certificate Key” text in files called “stunnel.cert” and “stunnel.cert.key” in the directory “/usr/local/Stunnel/conf”.