Template:Networking rut2xx manual vpn stunnel

Stunnel

Stunnel is an open-source a proxy service that adds TLS encryption to clients and servers already existing on a VPN network. TLS encryption provided by Stunnel can be used as an additional layer of encryption for data sent by VPN. This procedure increases the security of the established connection and provides higher chances of passing a Deep packet inspection (DPI) check.

For a more in-depth Stunnel configuration example visit this page: [[OpenVPN_over_Stunnel_{{{name}}}|OpenVPN over Stunnel]].

Stunnel Globals

The **Stunnel Globals** section is used to manage the Stunnel service as a whole. Refer to the figure and table below for information on the fields contained in the Stunnel Globals section.

[[File:{{{file_stunnel_globals}}}]]

Field	Value	Description
Use alternative config	yes | no; default: **no**	Turns the possibility to upload an external Stunnel configuration file on or off.if you turn this on, other Stunnel configurations present in the router will become inactive.
Upload alternative config	file; default: **none**	Uploads an Stunnel configuration file.

Stunnel client/server

To create a new Stunnel instance, go to the *Services → VPN → Stunnel* section, enter a custom name and click the 'Add' button. An Stunnel instance with the given name will appear in the "Stunnel Configuration" list.

To begin configuration, click the 'Edit' button located next to the instance. Refer to the figure and table below for information on the Stunnel instance's configuration fields:

[[File:{{{file_stunnel_client_server_config}}}]]

Field	Value	Description
Enable	yes | no; default: **no**	Turns the Stunnel instance on or off.

Operating Mode	Server | Client; default: **Server**	Selects the Stunnel instance's role. • **Server** - listens for connecting Stunnel clients. • **Client** - listens for connecting OpenVPN clients and connects to an Stunnel server.
Listen IP	ip; default: **none**	Makes the instance "listen" for incoming connections on the specified IP address. When left empty, the value of this field defaults to *localhost* (*127.0.0.1*).
Listen Port	integer [0..65535]; default: **none**	Makes the instance "listen" for incoming connections on the specified TCP port. Make sure you chose a port that is not being used by another service. You will also have to allow traffic on the specified port. You can do this via the **Network → Firewall → Traffic Rulles → [[{{{name}}}_Firewall#Open_Ports_On_Router|Open Ports On Router]]** section.
Connect IP's	ip:port; default: **none**	IP:Port to listen for VPN connections. When left empty the value of this field is interpreted as *localhost*. Must contain at least one item. If multiple options are specified, remote address is chosen using a round-robin algorithm.
TLS Cipher	None | Secure | Custom; default: **None**	Packet encryption algorithm cipher.
Allowed TLS Ciphers	string; default: **none**	A list of TLS ciphers accepted for this connection.
Application Protocol	Connect | SMTP | Not specified; default: **Not specified**	This option enables initial, protocol-specific negotiation of the TLS encryption. The protocol option should not be used with TLS encryption on a separate port.
Protocol Authentication	**Connect**: Basic | NTLM; default: **Basic** **SMTP**: Plain | Login; default: **Plain**	Authentication type for the protocol negotiations.
Protocol Domain	string; default: **none**	Domain for the protocol negotiations.
Protocol Host	host:port; default: **none**	Specifies the final TLS server to be connected to by the proxy, and not the proxy server directly connected by Stunnel. The proxy server should be specified along with the *connect* option.
Protocol Username	string; Default: **none**	Username for authentication to the protocol negotiations.
Protocol Password	string; default: **none**	Password for authentication to the protocol negotiations.

Certificate File	.crt file; default: **none**	TLS client or server certificate file.
Private Key	.key file; default: **none**	TLS client or server key file.