SATELLAR and VPN

VPN i.e. Virtual Private Network is a method to create an authenticated and in case wanted secured method for communication. When enabled and configured, it is possible to have the Satellar radio network communication to go over VPN. VPN can be set to either server or client state. This makes the view of VPN category bit different depending on the mode and this can be seen in pictures below.

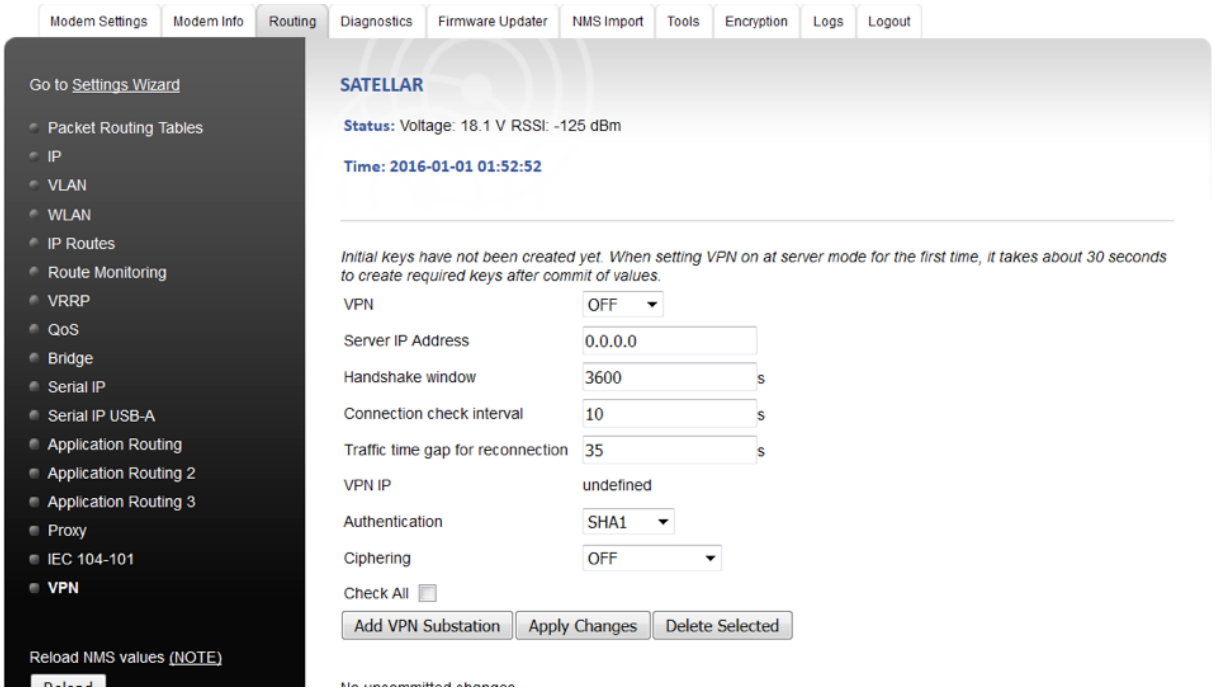

Figure 1.1 Configuration view

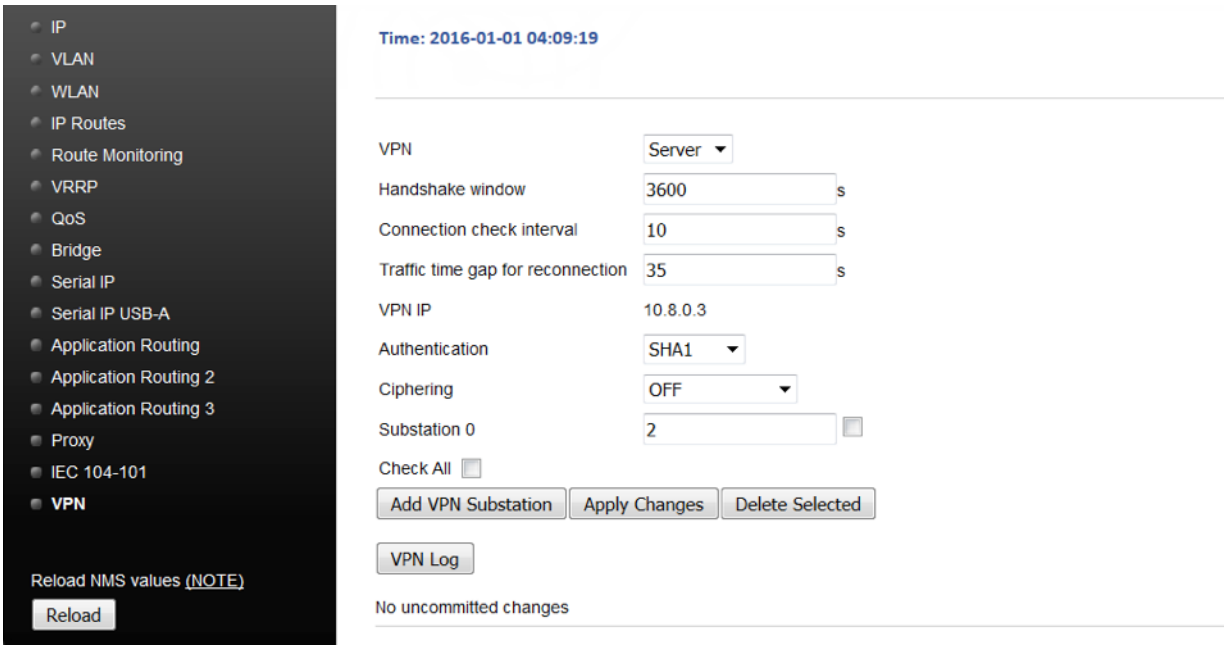

Figure 1.2 VPN view at server when server setup has been done

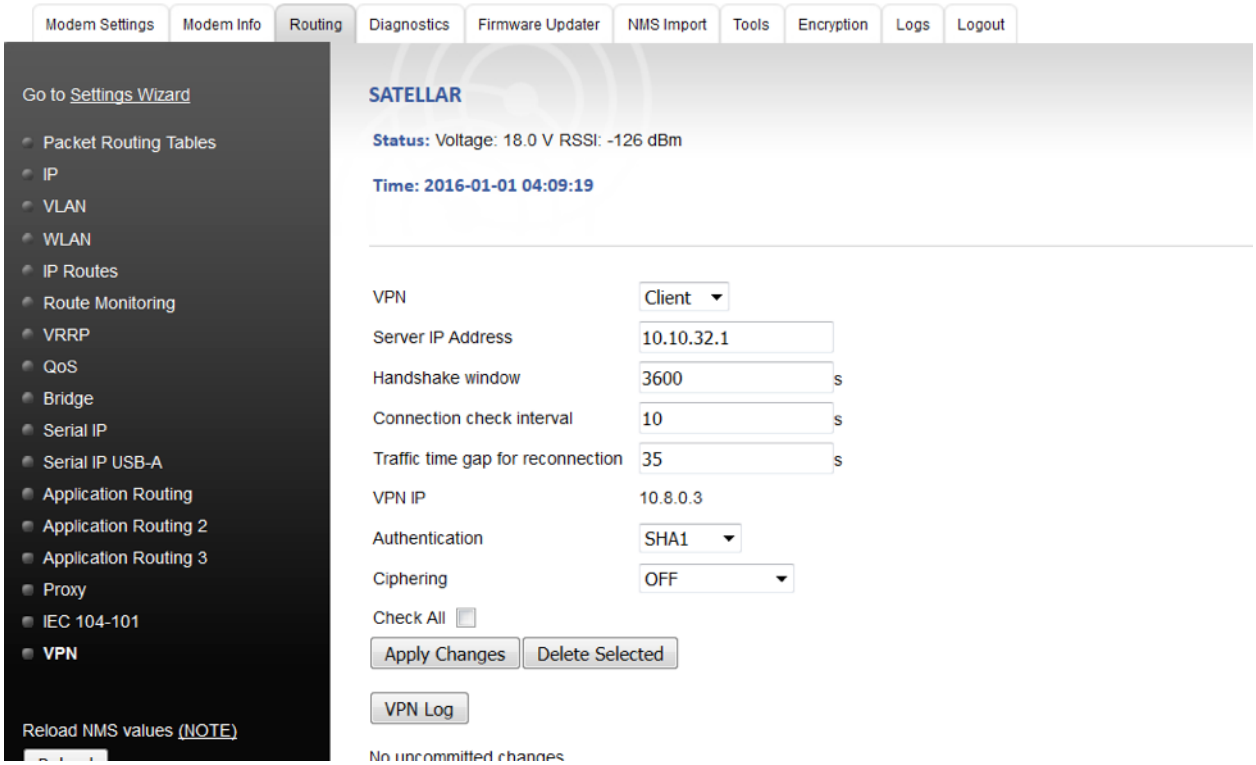

Figure 1.3 VPN view at client mode when connection has been established

At initial view, all available parameters are visible. At server mode there is server IP and at client mode there are no substations visible nor “Add Substation” button available. At both cases, after the VPN has been set on at first time a button “VPN Log” appears. This can be used to observe start of VPN procedure and connection activity.

Following parameters are used to configure VPN.

Name	Description	Available values (default italic)	Subunit	NMS ID
VPN	Status of IEC service, off or on	Off, On	1	1.3664
Server IP Address*	IP address to connect the VPN to	IP Address (0.0.0.0)	1	1.3665
Handshake Window	Key exchange must finalize within seconds defined here of handshake initiation by any peer. If the handshake fails we will attempt to reset our connection with our peer and try again.	1 ... 65535 (3600)	1	1.3666

Name	Description	Available values (default italic)	Subunit	NMS ID
Connection Check Interval	This defines the time in seconds how often there is the need for some traffic to verify that connection works. If no traffic has been received from other side during this time, a connection check message is sent. See Traffic time gap for reconnection definition and more details below.	0 ... 65535 (10)	1	1.3667
Traffic time gap for reconnection	This value is in connection to check interval value. If no data – not even connection check messages – has been received during this time, the connection to other side will be reinitialized.	0 ... 65535 (35)	1	1.3668
VPN IP	This value shows the VPN IP address of this device. This is read-only parameter that is generated automatically. It is based on radio address.	undefined	1	1.3669
Authentication	The mode of authentication used in authentication of packets with HMAC using message digest algorithm defined here.	SHA1, MD5, SHA224, SHA256, SHA348, SHA512	1	1.3673
Ciphering	Ciphering type used in VPN communication. Optional, default is off.	OFF, Blowfish, CAST5, AES-128 CBC, AES-128 CFB, AES-128 OFB, AES-192 CBC, AES-192 CFB, AES-192 OFB, AES-256 CBC, AES-256 CFB, AES-256 OFB, AES-256 CFB1, AES-256 CFB8	1	1.3674

Name	Description	Available values (default italic)	Subunit	NMS ID
Substation***	Substation address that is added at server side. Notice that substation number is analog to radio address i.e. RMAC value. Thus if Satellar station that has radio address 4 is wanted to be set as VPN substation, then substation with address 4 must be added.		1	1.3675

* Usable and available only at client mode
** Noticeable that radio has the internal encryption mode available
*** Usable and available only at server mode
VPN feature is offered as beta version and is not yet including all of the functionalities of final version.

Certificate based VPN

The basic VPN connection in Satellar network is based on VPN certificate method. This means that connection is based on certificates created at server and distributed to clients.

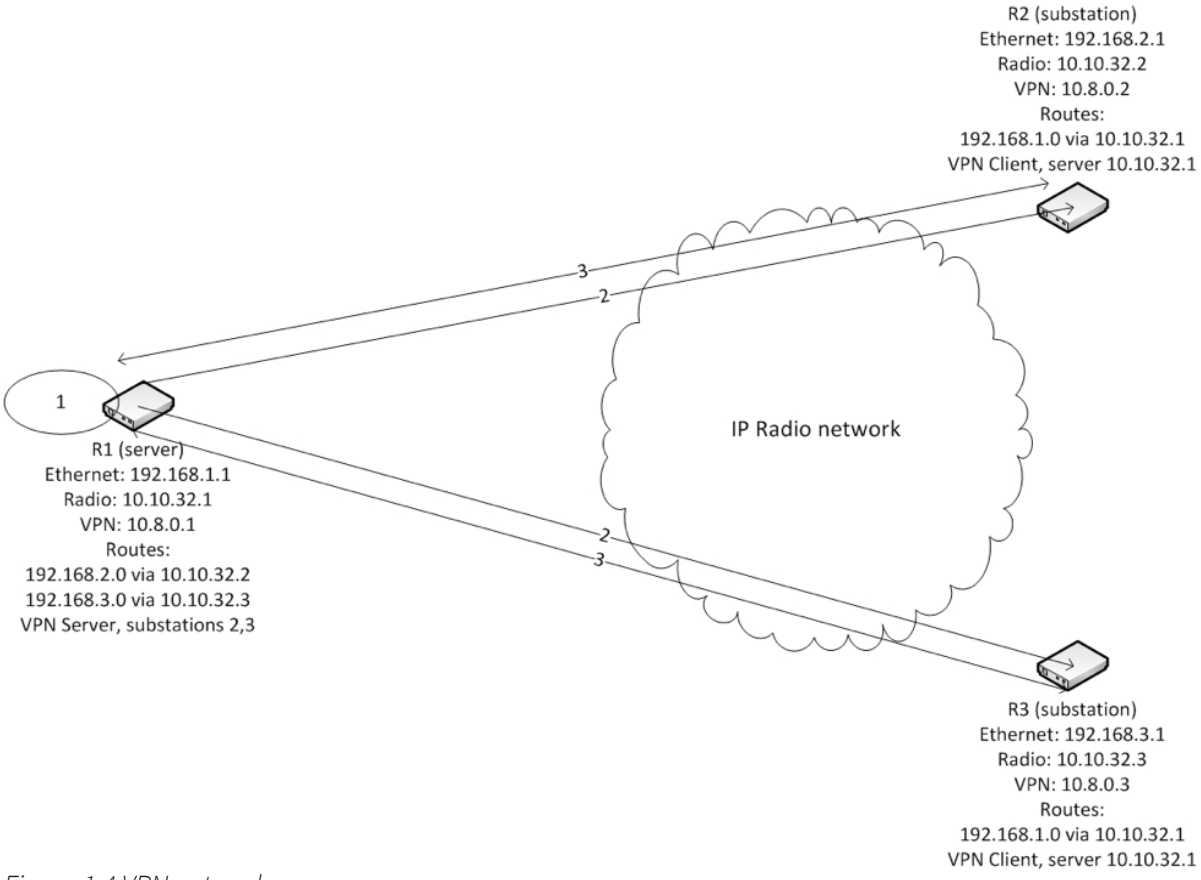

Figure 1.4 VPN network

At start point, there are no keys that are required for connection at any device. This is informed with text "*Initial keys have not been created yet. When setting VPN on at server mode for the first time, it takes about 30 seconds to create required keys after commit of values.*". This means basically that there are no such keys nor certificates that are required at server side for connection. Here are the steps how to create all keys and certificates and to setup VPN network (with same numbers as in picture).

1. User sets some device to server mode. When committed, device creates the keys and certificates required for server activity. This can be seen at WWW UI.

2. User creates substations at server device. This procedure creates the keys and certificates for server-client connectivity for each substation and delivers them to substations. The proceeding of this can be observed from WWW UI.

3. Once the UI is not reporting from any key creation, substation devices can be set to client mode. This starts the client side process which connects to server and negotiates the connection with it. VPN connection is ready to be used after this.

The proceeding of procedure can be seen from WWW UI. First when initial server key creation is ongoing, there appears a text "*VPN initial server key creation ongoing, takes approximately 30-40 seconds.*" The text with "*keys have not been created yet.*." may still appear for a moment at this point.

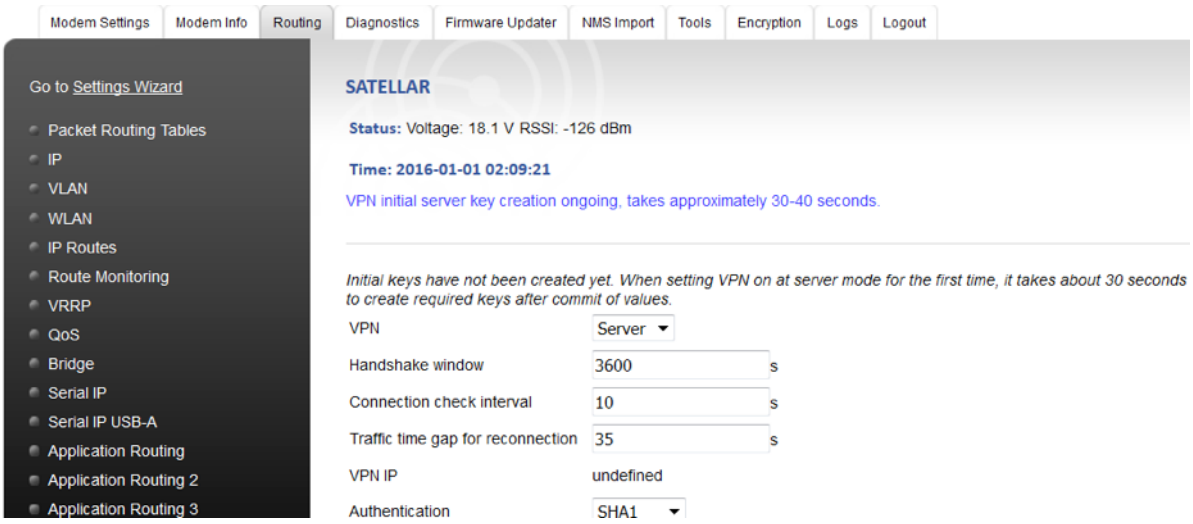

Figure 1.5 Initial server key creation.

Next, the keys and certificates are created for substation by adding one and delivered. When this procedure is ongoing, a text "*VPN substation key creation ongoing for station X - takes approximately 1 minute per substation*" appears.

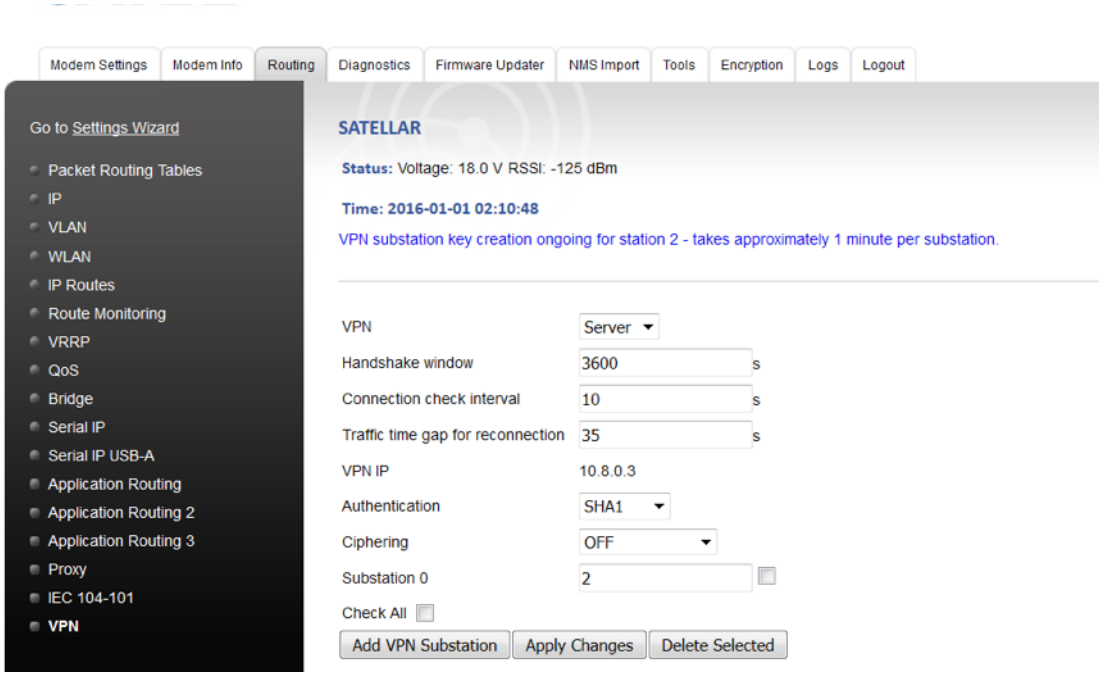

Figure 1.6 VPN key distribution ongoing

Once all items are created and delivered, connection can be checked by e.g. pinging. Traffic can be routed to other devices by whatever means is needed same way as with SATELLAR regular 10.10.32.x radio IPs.

It must be noted that even the procedure of creating the keys and certificates and delivering them to correct place – especially with substations – takes some time, this is a one-time process.

Connectivity verification

The behavior of connection check interval and traffic time gap for reconnection parameters is besides dependent of each other, also definitions in other devices. First of all, connection check mechanism requires the corresponding definitions to both server and client side. Basically both sides should have equivalent connection check interval values. It could be done so that e.g. clients would require less frequent check interval than server, but this should be considered case-by-case.

Next, connection check interval must be corresponding to value of traffic time gap for reconnection. If check interval is e.g. 10, it means that connection is checked at least every ten seconds. In case check is failing, next check is done after 10 seconds. Default value of 35 seconds in traffic time gap for reconnection means that if no traffic is received at 35 seconds the connection is reinitialized. Server side default traffic time gap for reconnection is recommended as 70. It must be noted that the connection verification is not based on the sent check messages but instead the ones that are being received. Thus, the values of connection check interval and traffic time gap for reconnection must be considered through whole network.

Furthermore, it is advisable to keep the traffic time gap for reconnection time at server slightly higher than at client devices. This is because the way of behavior: if the server e.g. restarts, it does not do any renegotiation or equivalent with client, but if the client restarts, it always renegotiates with server. In case of e.g. heavy congestion (perhaps with e.g. UDP traffic), if server is waiting for longer than clients, then only those clients who do not have been having any traffic are being reinitialized instead of server reinitializing more connections than needed. This is rather basic principle.

Default values provide at least some kind of start point. Server side traffic time gap for reconnection is set to twice the one at client devices, with default values to 70. Thus in case of clients, 3 check interval messages from server must be lost before reinitializing is done. On the other hand server device is acting less sensitive or slower to traffic gaps. Of course the check interval could be modified as well, e.g. so that clients send only every 20 seconds which would mean that server with time gap value 70 seconds would wait for at least 3 connection check messages from client. This can be considered case by case. It must be noticed also that

- Each connection is handled individually
- In case any other traffic is received, it resets the counters

The reason for using these check mechanisms is that some incidents may cause discontinuity point into connection. Such might be e.g. reset of server or client device or software, although client reset means always renegotiation with server which then solves the case. Values can be set to 0 as well and then connection checking is not done. Though it does take some traffic off, it is not recommended because the connection can get broken for some incident. In any case, these values must be corresponding with each other at one device and in network. In practice, if check interval is 0, time gap must be also set to 0 and this disables both mechanisms at one device. If e.g. server has check interval of zero and time gap value 70 and clients have check interval of 10 and time gap 35, it would mean that server never sends check messages and with no other traffic clients would reinitialize the connections in every 35 seconds. Server would not mind as it would get check messages all the time.