SMART NETWORK SOLUTIONS
FOR SERVICE PROVIDERS

GW1000 Series User Manual

Issue:	1.4
Date:	23 October 2015

1 Introduction

This user manual describes the features and how to configure Virtual Access GW1000 Series routers.

Designed for managed network providers, GW1000 Series routers provide secure WAN connectivity for internet and private networking environments over 3G or 4G broadband paths and incorporate optional 802.11n WiFi connectivity.

1.1 Document scope

This document covers the following models in the GW1000 Series.

GW1032:	Dual Ethernet, 3G, Dual SIM, WiFi
GW1042:	Dual Ethernet, 4G/LTE, Dual SIM, WiFi

The above hardware models use the **CPX** branch of firmware. This document was released with firmware version **CPX-19.00.01**. The screenshots and commands may vary slightly if you are using a different firmware version.

1.2 Using this documentation

You can configure your router using either the router's web interface or via the command line using UCI commands. Each chapter explains first the web interface settings, followed by how to configure the router using UCI. The web interface screens are shown along with a path to the screen for example, 'In the top menu, select **Service -> SNMP**.' followed by a screen grab.

After the screen grab there is an information table that describes each of the screen's fields.

1.2.1 Information tables

We use information tables to show the different ways to configure the router using the router's web and command line. The left-hand column shows three options:

- **Web**: refers the command on the router's web page,
- **UCI**: shows the specific UCI command, and
- **Opt**: shows the package option.

The right-hand column shows a description field that describes the feature's field or command and shows any options for that feature.

Some features have a drop-down menu and the options are described in a table within the description column. The default value is shown in a grey cell.

Values for enabling and disabling a feature are varied throughout the web interface, for example, 1/0; Yes/No; True/False; check/uncheck a radio button. In the table descriptions, we use **0** to denote Disable and **1** to denote Enable.

Some configuration sections can be defined more than once. An example of this is the routing table where multiple routes can exist and all are named 'route'. For these sections, the UCI command will have a code value [**0**] or [**x**] (where x is the section number) to identify the section.

Web Field / UCI / Package Option	Description
Web: Metric UCI: network.@route[0].metric Opt: metric	Specifies the route metric to use.

Note: these sections can be given a label for identification when using UCI or package options.

```
network.@route[0]=route
network.@route[0].metric=0
```

can be witten as:

```
network.routename=route
network.routename.metric=0
```

However the documentation usually assumes that a section label is not configured.

The table below shows fields from a variety of chapters to illustrate the explanations above.

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Enable
UCI: cesop.main.enable
Opt: enable</td><td>Enables CESoPSN services.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
<tr><td>Web: Interface
UCI: dropbear.@dropbear[0].Interface
Opt: interface</td><td>Listens only on the selected interface. If unspecified is checked, listens on all interfaces. All configured interfaces will be displayed via the web GUI.<table><tr><td>(unspecified)</td><td>listens on all interfaces.</td></tr><tr><td>Range</td><td>Configured interface names.</td></tr></table></td></tr>
<tr><td>Web: Agent Address
UCI: snmpd.agent[0].agentaddress
Opt: agentaddress</td><td>Specifies the address(es) and port(s) on which the agent should listen.
[(udp|tcp):]port[@address][,...]</td></tr>
</table>

Table 1: Example of an information table

1.2.2 Definitions

Throughout the document, we use the host name 'VA_router' to cover all router models.

UCI commands and package option examples are shown in the following format:

```
root@VA_router:~# vacmd show current config
```

1.2.3 Diagnostics

Diagnostics are explained at the end of each feature's chapter.

1.2.4 UCI commands

For detailed information on using UCI commands, read chapters 'Router File Structure' and 'Using Command Line Interface'.

2 GW1000 series hardware

2.1 Hardware specification

2.1.1 GW1000 series router model variants

GW1032: Dual Ethernet, 3G, Dual SIM, WiFi

GW1042: Dual Ethernet, 4G/LTE, Dual SIM, WiFi

2.2 Hardware features

- Dual SIM sockets
- Dual antenna SMA connectors for 3G/4G main and aux
- GPS antenna with 3.3V active power feed
- Two 10/100 Mbps Ethernet ports
- WiFi with internal antennas
- Concurrent Access Point and Station mode

2.3 GSM technology

- LTE
- HSPA+
- EDGE/GPRS
- GPS
- 2100/1900/1800/900/850 MHz Bands

2.4 WiFi technology

- 802.11 b/g/n
- Single band 2.4GHz
- Up to 20dBm output power
- Internal antenna

2.5 Power supply

The GW1000W Series router has three power supply options:

- Standard 12V DC 0.5 A
- 12V DC 0.5 A with extended temp (-20˚C to -70˚C)
- Power lead with 3 connectors for 12V permanent, 12V switched (ignition sense) and ground

2.6 Dimensions

Unit size:	114W 114D 29Hmm
Unit size with carrier	120W 120D 32Hmm
Unit weight:	209g

2.7 Compliance

The GW1000 Series router is compliant and tested to the following standards:

Safety	EN60950-1: 2006
EMC	EN55022:1998 Class B and EN55024:1998 ETSI 301489-17
Environmental	ETSI 300 019-1-3 Sinusoidal Vibration and Shock ETSI 300 019-2-3 Random Vibration.
WiFi 2.4GHz	ETSI EN 300 328 V1.9 (2015-02)

2.8 Operating temperature range

The operating temperature range depends on the RF Band.

RF Band	2G Bands	3G Bands	4G LTE Bands	Operating Temp
RFA	850/900/1800/1900	900/2100	-	-20°C to 70°C
RFB	850/900/1800/1900	850/900/1900/2100	-	-20°C to 70°C
RFC	850/900/1800/1900	850/900/1900/2100	B1/B2/B3/B5/B7/B8/B20	-20°C to 70°C
RFD	-	-	B3/B7/B20/B31	-20°C to 60°C
RFE	900/1800	900/2100	B1/B3/B7/B8/B20/B38/B40	-20°C to 70°C

RFF	-	CDMA TX 452.500~457.475 RX 462.000~467.475	-	-20°C to 60°C

2.9 Antenna

The GW1000 Series router has two SMA connectors for connection of two antennas for antenna diversity. Antenna diversity helps improve the quality of a wireless link by mitigating problems associated with multipath interference.

2.10 Components

To enable and configure connections on your router, it must be correctly installed.

The GW1000 Series router contains an internal web server that you use for configurations. Before you can access the internal web server and start the configuration, ensure the components are correctly connected and that your PC has the correct networking setup.

The GW1000 Series router comes with the following components as standard:

1 x GW1000 Series router	
1 x lockable SIM cover	

Table 2: GW1000 Series router standard components

Optional components include:

Ethernet cable. RJ45 connector at both ends.	
Power supply unit.	
Right angle antenna for 3G/4G network.	 Virtual Access supplies a wide range of antennas. Please visit our website: www.virtualaccess.com or contact Virtual Access for more information.

Table 3: GW1000 Series router optional components

2.11 Inserting a SIM card

1. Ensure the unit is powered off.
2. Hold the SIM 1 card with the chip side facing down and the cut corner front left.
3. Gently push the SIM card into SIM slot 1 until it clicks in.
4. If using SIM 2 then hold the SIM with the cut corner front right
5. Gently push the SIM card into SIM slot 2 until it clicks in.

2.12 Connecting the SIM lock

Connect the SIM lock using the Allen key provided.

2.13 Connecting cables

Connect one end of the Ethernet cable into port A and the other end to your PC or switch.

2.14 Connecting the antenna

If you are connecting only one antenna, screw the antenna into the MAIN SMA connector.

If you are using two antennas, screw the main antenna into the MAIN SMA connector and the secondary antenna into the AUX SMA connector.

2.15 Powering up

The GW1000 takes approximately 2 minutes to boot up. During this time, the PWR/CONFIG LED flashes in a double flash pattern – 2 quick fashes followed by a pause.

Other LEDs display different diagnostic patterns during boot up.

Booting is complete when the PWR/CONFIG LED stops double flashing and stays solid or flashing steady, indicating the particular running configuration is loaded. Read the chapter ‘GW1000 LED behaviour’, for PWR/CONFIG LED states.

2.16 Reset button

The reset button is used to request a system reset.

When you press the reset button the PWR/CONFIG LED will display different patterns depending on how long you press the button. The flashing patterns will be different for the 2 flashing phases indicated below. The length of time you hold the reset button will determine the router behaviour.

Press Duration	PWR / CONFIG LED behaviour	Router Behaviour on depress
Less than 3 seconds	On	Normal reset.
Between 3 and 15 seconds	Flashing	The router resets to factory configuration.
Between 15 and 20 seconds	On	No action.
Between 20 seconds and 30 seconds	Flashing	The router resets to recovery mode.
Over 30 seconds	On	Normal reset.

Table 4: GW1000 series router reset behaviour

3 Installing a GW1000 into a vehicle

Install the GW1000 using the vehicle installation power cable provided.

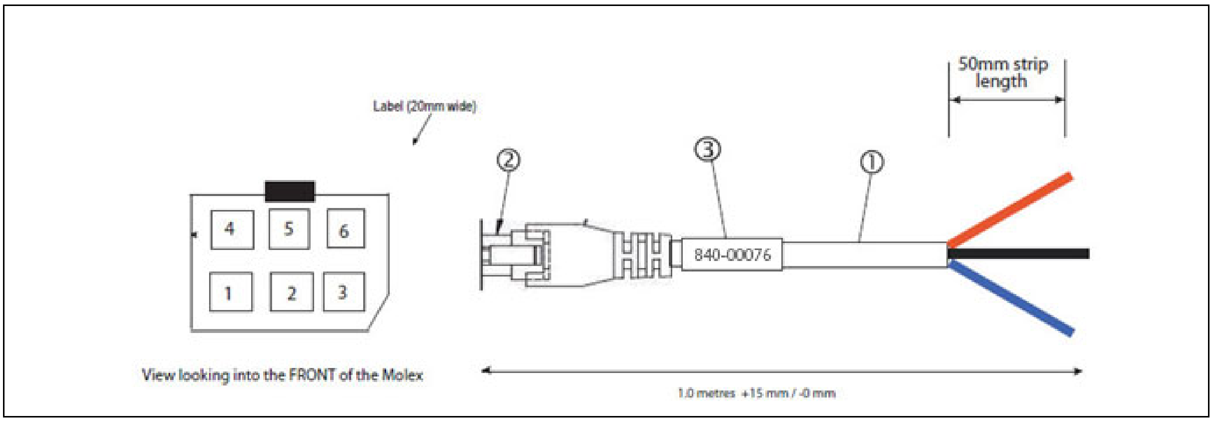


Figure 1: GW1000 3 core power cable

(1)	Each wire is 1.0mm square, with overall PVC sheath
(2)	Connector: Molex Microfit 6circuit standard
(3)	Label 20mm wide
Note:	Requires 5 amp fuse in series with red and blue wires

Table 5: Power cable descriptions

1. Connect the **BLACK** wire to a ground wire.
2. Connect the **BLUE** wire to a 12V switched vehicle ignition wire.
3. Connect the **RED** wire to a 12V permanent wire.
4. Plug the 6 pin connector into the GW1000.

4 GW1000 series LED behaviour

4.1 Main LED behaviour

There are five LEDs on the GW1000 series router

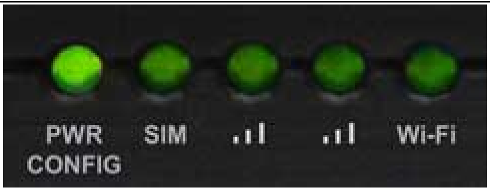


Figure 1: LEDs on the GW1000 series router

The possible LED states are:

- Off
- Flashing slowing (2 flashes per second)
- Flashing quickly (5 flashes per second)
- Double flash (2 quick flashes then a pause)
- On

The following table describes the possible LED behaviours and meanings.

Booting		The GW1000 takes approximately 2 minutes to boot up. During this time, the power LED flashes. Other LEDs display different diagnostic patterns during boot up. Booting is complete when the power LED stops flashing and stays on steady.
PWR/CONFIG LED	Off	No power/boot loader does not exist.
	Double flash	Unit is booting from power on.
	Flashing slowly	Unit is in recovery mode.
	Flashing quickly	Unit is in factory configuration.
	On	Unit has completed booting up process and is in either config 1 or config2.
SIM LEDs	Off	Not selected or SIM not inserted.
	Flashing	SIM selected and data connection is being established.
	On	SIM selected and registered on the network.

Signal LEDs	Both LED’s off	Not connected or signal strength <= -113dBm.
	Left LED on Right LED off	Connected and signal strength <= -89dBm.
	Left LED off Right LED on	Connected and signal strength between -89dBm and -69dBm.
	Both LED’s on	Connected and signal strength >-69dBm.
Wi-Fi LEDs	Off	Wi-Fi not enabled.
	Flashing	Data activity on WiFi interface.
	On	WiFi is enabled.

Table 6: LED behaviour and descriptions

Note: when a data connection does not exist, none of the signal LEDs will light regardless of signal strength.

4.2 Ethernet port LED behaviour

The Ethernet port has two physical LEDs, one is green and one is amber. When looking at the port, the amber LED is on the right and is the only active LED.

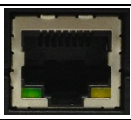


Figure 2: Ethernet LED

Ethernet LED (amber)	On	Physical Ethernet link detected
	Flashing	Data is being transmitted/ received over the link.

Table 7: Ethernet LED activity description

5 Factory configuration extraction from SIM card

Virtual Access routers have a feature to update the factory configuration from a SIM card. This allows you to change the factory configuration of a router when installing the SIM.

1. Make sure the SIM card you are inserting has the required configuration written on it.
2. Ensure the router is powered off.
3. Hold the SIM 1 card with the chip side facing down and the cut corner front left.
4. Gently push the SIM card into SIM slot 1 until it clicks in.
5. Power up the router.
6. Depending on the model, the power LED and/or the configuration LED flash as usual.

The SIM LED starts flashing. This indicates the application responsible for 3G and configuration extraction management is running. It also means the update of the configuration is happening.

When the update is finished, depending on the model, the power LED and/or the configuration LED blink alternatively and very fast for 20 seconds.

6 Accessing the router

Access the router through the web interface or by using SSH. By default, Telnet is disabled.

6.1 Configuration packages used

Package	Sections
dropbear	dropbear
system	main
uhttpd	main cert

6.2 Accessing the router over Ethernet using the web interface

DHCP is disabled by default, so if you do not receive an IP address via DHCP, assign a static IP to the PC which will be connected to the router.

PC IP address	192.168.100.100
Network mask	255.255.255.0
Default gateway	192.168.100.1

Assuming that the PC is connected to Port A on the router, in your internet browser, type in the default local IP address **192.168.100.1**, and press **Enter**. The Authorization page appears.

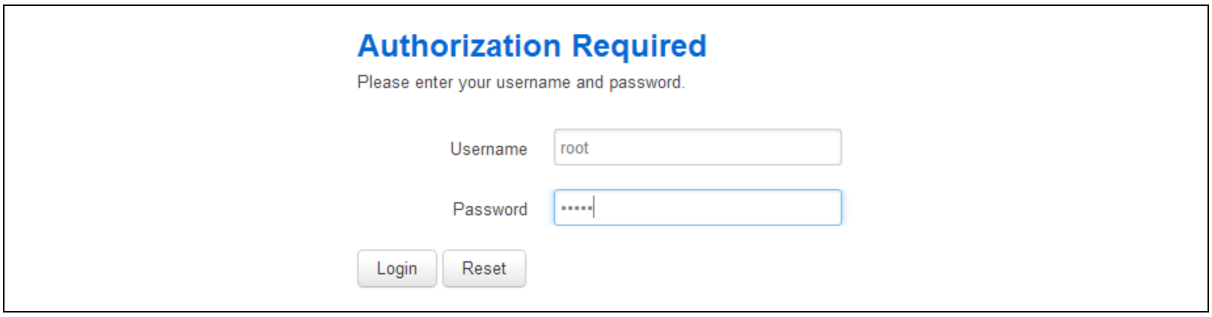


Figure 3: The login page

The password may vary depending on the factory configuration the router has been shipped with. The default settings are shown below. The username and password are case sensitive.

In the username field, type **root**.

In the Password field, type **admin**.

Click **Login**. The Status page appears.

6.3 Accessing the router over Ethernet using an SSH client

You can also access the router over Ethernet, using Secure Shell (SSH) and optionally over Telnet.

To access CLI over Ethernet start an SSH client and connect to the router's management IP address, on port 22: **192.168.100.1/24**.

On first connection, you may be asked to confirm that you trust the host.

Figure 4: Confirming trust of the routers public key over SSH

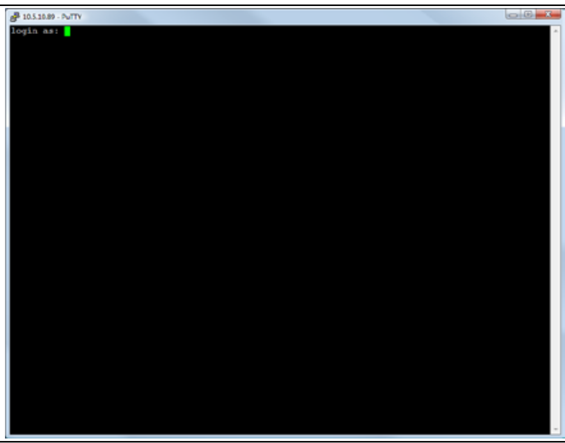


Figure 5: SSH CLI logon screen

In the SSH CLI logon screen, enter the default username and password.

Username: **root**

Password: **admin**

6.4 Configuring the password

6.5 Configuration packages used

Package	Sections
system	main

6.5.1 Configuring the password using the web interface

To change your password, in the top menu click **System -> Administration**. The Administration page appears.

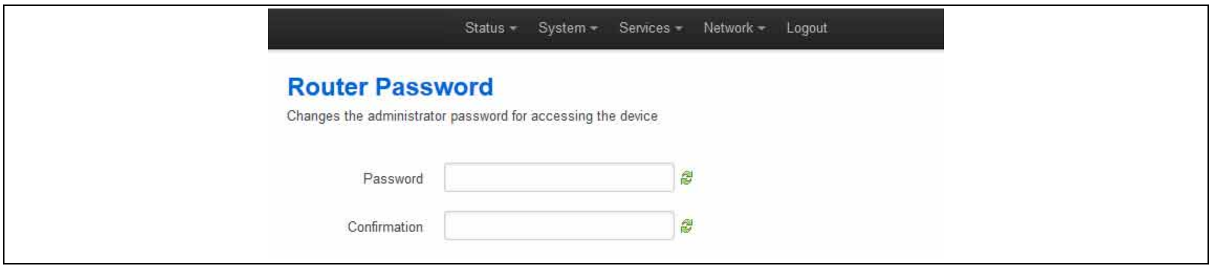


Figure 6: The router password section

In the Router Password section, type your new password in the password field and then retype the password in the confirmation field.

Scroll down the page and click **Save & Apply**.

Note: the username 'root' cannot be changed.

Web Field / UCI / Package Option	Description
Web: Password UCI: system.main.password Opt: password	Defines the root password. The password is displayed encrypted via the CLI using the 'hashpassword' option. UCI: system.main.hashpassword Opt: hashpassword

6.5.2 Configuring the password using UCI

The root password is displayed encrypted via the CLI using the hashpassword option.

```
root@VA_router:~# uci show system
system.main=system
system.main.hostname=VA_router
system.main.hashpassword=$1$jRX/x8A/$U5kLCMpi9dcahRhOl7eZV1
```

If changing the password via the UCI, enter the new password in plain text using the password option.

```
root@VA_router:~# uci system.main.password=newpassword
root@VA_router:~# uci commit
```

The new password will take effect after reboot and will now be displayed in encrypted format via the hashpassword option.

6.5.3 Configuring the password using package options

The root password is displayed encrypted via the CLI using the hashpassword option.

```
root@VA_router:~# uci export system
package system

config system 'main'
      option hostname 'VA_router'
      option hashpassword '$1$wRYYiJOz$EeHN.GQcxXhRgNPVbqxVw
```

If changing the password via the UCI, enter the new password in plain text using the password option.

```
package system

config system 'main'
      option hostname 'VA_router'
      option hashpassword '$1$wRYYiJOz$EeHN.GQcxXhRgNPVbqxVw
      option password ‘newpassword’
```

The new password will take effect after reboot and will now be displayed in encrypted format via the hashpassword option.

6.6 SSH access

SSH allows you to access remote machines over text based shell sessions. SSH uses public key cryptography to create a secure connection. These connections allow you to issue commands remotely via a command line.

The router uses a package called "Dropbear" to configure the SSH server on the box. You can configure Dropbear via the web interface or through an SSH connection by editing the file stored in: **/etc/config_name/dropbear**.

6.6.1 Configuration packages used

Package	Sections
dropbear	dropbear

6.6.2 SSH access using the web interface

In the top menu, click **System -> Administration**. The Administration page appears. Scroll down to the SSH Access section.

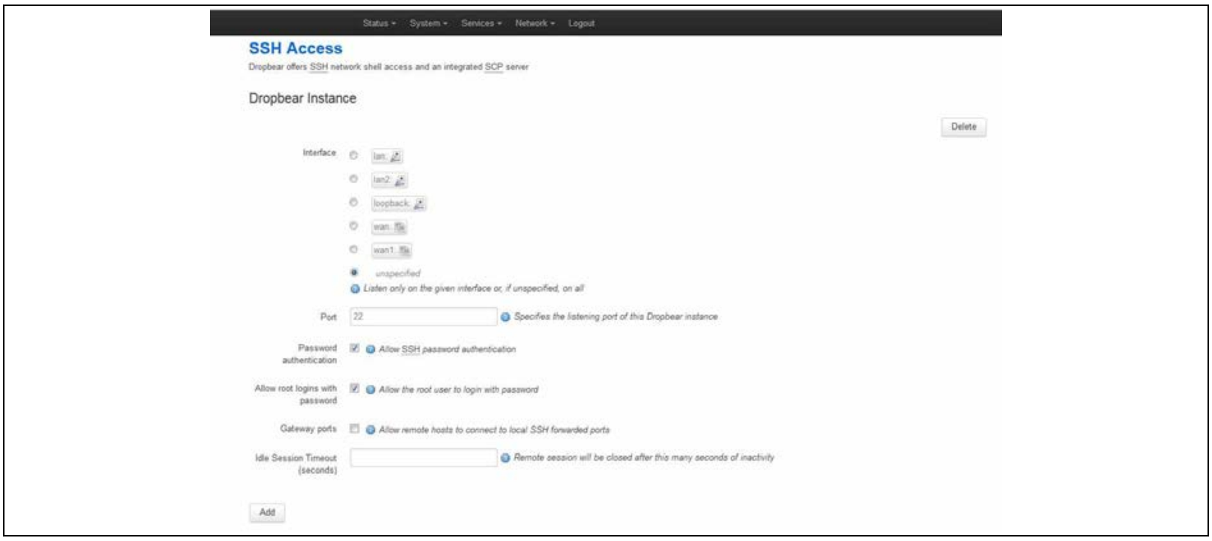


Figure 7: The SSH access section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td colspan="2">Basic settings</td></tr>
<tr><td>Web: Interface
UCI: dropbear.@dropbear[0].Interface
Opt: interface</td><td>Listens only on the selected interface. If unspecified is checked, listens on all interfaces. All configured interfaces will be displayed via the web GUI.<table><tr><td>(unspecified)</td><td>listens on all interfaces.</td></tr><tr><td>Range</td><td>Configured interface names.</td></tr></table></td></tr>
<tr><td>Web: Port
UCI: dropbear.@dropbear[0].Port
Opt: port</td><td>Specifies the listening port of the Dropbear instance.<table><tr><td>22</td><td></td></tr><tr><td>Range</td><td>0-65535</td></tr></table></td></tr>
<tr><td>Web: Password authentication
UCI: dropbear.@dropbear[0].PasswordAuth
Opt: PasswordAuth</td><td>If enabled, allows SSH password authentication.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
<tr><td>Web: Allow root logins with password
UCI: dropbear.@dropbear[0].RootPasswordAuth
Opt: RootPasswordAuth</td><td>Allows the root user to login with password.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
<tr><td>Web: Gateway ports
UCI: dropbear.@dropbear[0].GatewayPorts
Opt: GatewayPorts</td><td>Allows remote hosts to connect to local SSH forwarded ports.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
</table>

Web: Idle Session Timeout UCI: dropbear.@dropbear[0].IdleTimeout Opt: IdleTimeout	Defines the idle period where remote session will be closed after the allocated number of seconds of inactivity. 30 — 30 seconds. Range
Web: n/a UCI: dropbear.@dropbear[0]. BannerFile Opt: BannerFile	Defines a banner file to be displayed during login. /etc/banner Range

Table 8: Information table for SSH access settings

6.6.3 Package dropbear using UCI

```
root@VA_router:~# uci show dropbear
dropbear.@dropbear[0]=dropbear
dropbear.@dropbear[0].PasswordAuth=on
dropbear.@dropbear[0].RootPasswordAuth=on
dropbear.@dropbear[0].GatewayPorts=0
dropbear.@dropbear[0].IdleTimeout=30
dropbear.@dropbear[0].Port=22
```

6.6.4 Package dropbear using package options

```
root@VA_router:~# uci export dropbear
package dropbear
config dropbear'
      option PasswordAuth 'on'
      option RootPasswordAuth 'on'
      option Port '22'
      option GatewayPorts ‘0’
      option IdleTimeout ‘30’
```

6.7 Certs and private keys

Certificates are used to prove ownership of a public key. They contain information about the key, its owner’s ID, and the digital signature of an individual that has verified the content of the certificate.

In asymmetric cryptography, public keys are announced to the public, and a different private key is kept by the receiver. The public key is used to encrypt the message, and the private key is used to decrypt it.

To access certs and private keys, in the top menu, click **System -> Administration**. The Administration page appears. Scroll down to the Certs & Private Keys section.

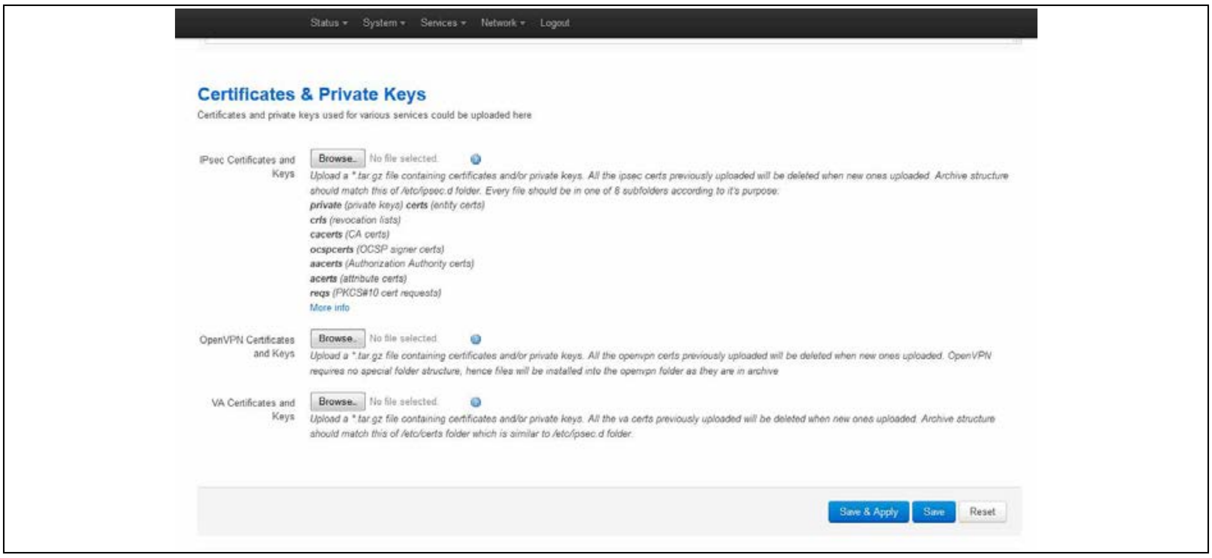


Figure 8: The certificates & private keys section

This section allows you to upload any certificates and keys that you may have stored. There is support for IPSec, OpenVPN and VA certificates and keys.

If you have generated your own SSH public keys, you can input them in the SSH Keys section, for SSH public key authentication.

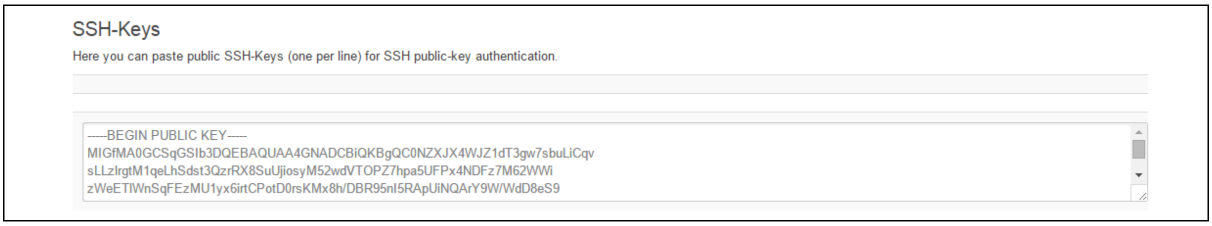


Figure 9: The SSH-Keys box

6.8 Configuring a router's web server

The router's web server is configured in package uhttpd. This file defines the behaviour of the server and default values for certificates generated for SSL operation. uhttpd supports multiple instances, that is, multiple listen ports, each with its own document root and other features, as well as cgi and lua. There are two sections defined:

- **Main:** this uHTTPd section contains general server settings.
- **Cert:** this section defines the default values for SSL certificates.

6.8.1 Configuration packages used

Package	Sections
uhttpd	main
	cert

To configure the router's HTTP server parameters, in the top menu, select **Services -> HTTP Server**. The HTTP Server page has two sections.

Main Settings	Server configurations.
Certificate Settings	SSL certificates.

6.8.2 Main settings

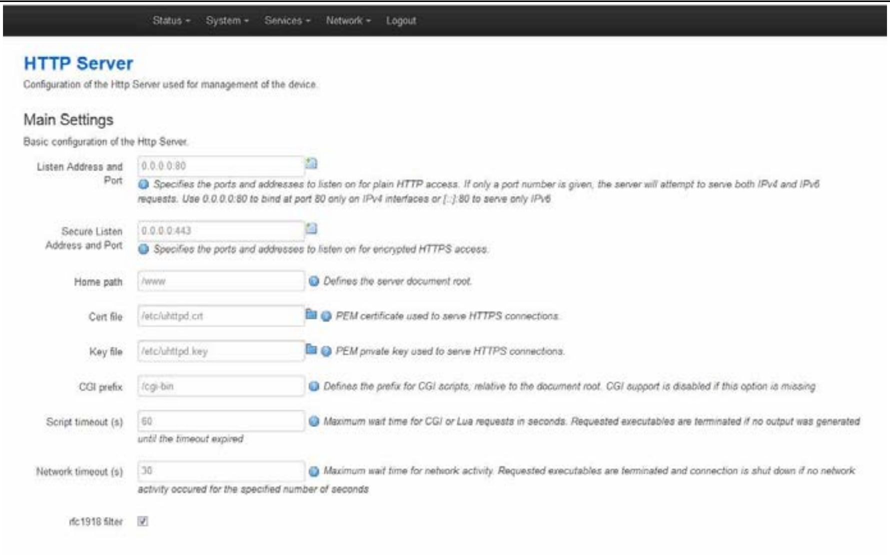


Figure 10: HTTP server settings

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Listen Address and Port
UCI: uhttpd.main.listen_http
Opt: list listen_http</td><td>Specifies the ports and addresses to listen on for plain HTTP access. If only a port number is given, the server will attempt to serve both IPv4 and IPv6 requests.
<table>
<tr><td>0.0.0.0:80</td><td>Bind at port 80 only on IPv4 interfaces.</td></tr>
<tr><td>[::]:80</td><td>Bind at port 80 only on IPv6 interfaces</td></tr>
<tr><td>Range</td><td>IP address and/or port</td></tr>
</table></td></tr>
</table>

Web: Secure Listen Address and Port UCI: uhttpd.main.listen_https Opt: list listen_https	Specifies the ports and address to listen on for encrypted HTTPS access. The format is the same as listen_http. 0.0.0.0:443 — Bind at port 443 only [::]:443 Range — IP address and/or port
Web: Home path UCI: uhttpd.main.home Opt: home	Defines the server document root. /www Range
Web: Cert file UCI: uhttpd.main.cert Opt: cert	ASN.1/DER certificate used to serve HTTPS connections. If no listen_https options are given the key options are ignored. /etc/uhttpd.crt Range
Web: Key file UCI: uhttpd.main.key Opt: key	ASN.1/DER private key used to serve HTTPS connections. If no listen_https options are given the key options are ignored. /etc/uhttpd.key Range
Web: CGI profile UCI: uhttpd.main.cgi_prefix Opt: cgi_prefix	Defines the prefix for CGI scripts, relative to the document root. CGI support is disabled if this option is missing. /cgi-bin Range
Web: N/A UCI: uhttpd.main.lua_prefix Opt: lua_prefix	Defines the prefix for dispatching requests to the embedded lua interpreter, relative to the document root. Lua support is disabled if this option is missing. /luci Range
Web: N/A UCI: uhttpd.main.lua_handler Opt: lua_handler	Specifies the lua handler script used to initialise the lua runtime on server start. /usr/lib/lua/luci/sgi/uhttpd.lua Range
Web: Script timeout UCI: uhttpd.main.script_timeout Opt: script_timeout	Sets the maximum wait time for CGI or lua requests in seconds. Requested executables are terminated if no output was generated. 60 Range
Web: Network timeout UCI: uhttpd.main.network_timeout Opt: network_timeout	Maximum wait time for network activity. Requested executables are terminated and connection is shut down if no network activity occured for the specified number of seconds. 30 Range

Web: N/A UCI: uhttpd.main.realm Opt: realm	Defines basic authentication realm when prompting the client for credentials (HTTP 400). OpenWrt | Range |
Web: N/A UCI: uhttpd.main.config Opt: config	Config file in Busybox httpd format for additional settings. Currently only used to specify basic auth areas. /etc/http.conf | Range |
Web: N/A UCI: uhttpd.main.index_page Opt: index_page	Index file to use for directories, for example, add index.php when using php. | Range |
Web: N/A UCI: httpd.main.error_page Opt: error_page	Virtual URL of file of CGI script to handle 404 requests. Must begin with '/' (forward slash). | Range |
Web: N/A UCI: uhttpd.main.no_symlinks Opt: no_symlinks	Does not follow symbolic links if enabled. 0 | Disabled. 1 | Enabled.
Web: N/A UCI: uhttpd.main.no_dirlists Opt: no_symlinks	Does not generate directory listings if enabled. 0 | Disabled. 1 | Enabled.
Web: rfc 1918 filter UCI: uhttpd.main.rfc1918_filter=1 Opt: rfc1918_filter	Enables option to reject requests from RFC1918 IPs to public server IPs (DNS rebinding counter measure). 0 | Disabled. 1 | Enabled.

Table 9: Information table for http server basic settings

6.8.3 HTTP server using UCI

Multiple sections of the type uhttpd may exist. The init script will launch one webserver instance per section.

A standard uhttpd configuration is shown below.

```
root@VA_router:~# uci show uhttpd
uhttpd.main=uhttpd
uhttpd.main.listen_http=0.0.0.0:80
uhttpd.main.listen_https=0.0.0.0:443
uhttpd.main.home=/www
uhttpd.main.rfc1918_filter=1
uhttpd.main.cert=/etc/uhttpd.crt
```

```
uhttpd.main.key=/etc/uhttpd.key
uhttpd.main.cgi_prefix=/cgi-bin
uhttpd.main.script_timeout=60
uhttpd.main.network_timeout=30
uhttpd.main.config=/etc/http.conf
```

6.8.4 HTTP server using package options

```
root@VA_router:~# uci export dropbear
config uhttpd 'main'
        list listen_http '0.0.0.0:80'
        list listen_https '0.0.0.0:443'
        option home '/www'
        option rfc1918_filter '1'
        option cert '/etc/uhttpd.crt'
        option key '/etc/uhttpd.key'
        option cgi_prefix '/cgi-bin'
        option script_timeout '60'
        option network_timeout '30'
        option config '/etc/http.conf'
```

6.8.5 HTTPs server certificate settings

To configure HTTPs server certificate settings, in the top menu, select **Services -> HTTP Server**. Scroll down to the Certificate Settings section.

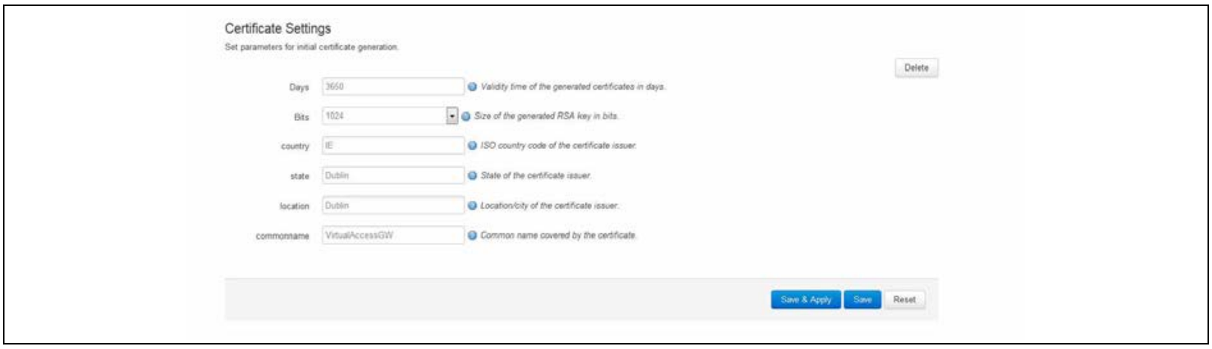


Figure 11: HTTP server certificate settings

Web Field / UCI / Package Option	Description
Web: Days UCI: uhttpd.px5g.days Opt: days	Validity time of the generated certificates in days. 730 Range
Web: Bits UCI: uhttpd.px5g.bits Opt: bits	Size of the generated RSA key in bits. 1024 Range
Web: Country UCI: uhttpd.px5g.country Opt: country	ISO code of the certificate issuer. Range
Web: State UCI: uhttpd.px5g.state Opt: state	State of the certificate issuer. Range
Web: Location UCI: uhttpd.px5g.location Opt: location	Location or city of the certificate user. Range
Web: Commonname UCI: uhttpd.commonname Opt: commonname	Common name covered by the certificate. For the purposes of secure Activation, this **must** be set to the serial number (Eth0 MAC address) of the device.

Table 10: Information table for HTTP server certificate settings

6.8.6 HTTPs server using UCI

```
root@VA_router:~# uci show uhttpd.px5g
uhttpd.px5g=cert
uhttpd.px5g.days=3650
uhttpd.px5g.bits=1024
uhttpd.px5g.country=IE
uhttpd.px5g.state=Dublin
uhttpd.px5g.location=Dublin
uhttpd.px5g.commonname=00E0C8000000
```

6.8.7 HTTPs server using package options

```
root@VA_router:~# uci export uhttpd
package uhttpdconfig 'cert' 'px5g'
      option 'days' '3650'
      option 'bits' '1024'
      option 'state' 'Dublin'
```

```
     option 'location' 'Dublin'
     option 'commonname' '00E0C8000000'
```

6.9 Basic authentication (httpd conf)

For backward compatibility reasons, uhttpd uses the file **/etc/httpd.conf** to define authentication areas and the associated usernames and passwords. This configuration file is not in UCI format.

Authentication realms are defined in the format prefix:username:password with one entry and a line break.

Prefix is the URL part covered by the realm, for example, cgi-bin to request basic auth for any CGI program.

Username specifies the username a client has to login with.

Password defines the secret password required to authenticate.

The password can be either in plain text format, MD5 encoded or in the form $p$user where the user refers to an account in /etc/shadow or /etc/passwd.

If you use p... format, uhttpd will compare the client provided password against the one stored in the shadow or passwd database.

6.10 Securing uhttpd

By default, uhttpd binds to 0.0.0.0 which also includes the WAN port of your router. To bind uhttpd to the LAN port only you have to change the listen_http and listen_https options to your LAN IP address.

To get your current LAN IP address, enter:

```
uci get network.lan.ipaddr
```

Then modify the configuration appropriately:

```
uci set uhttpd.main.listen_http='192.168.1.1:80'
uci set uhttpd.main.listen_https='192.168.1.1:443'

config 'uhttpd' 'main'
        list listen_http        192.168.1.1:80
        list listen_https       192.168.1.1:443
```

7 System settings

The system section contains settings that apply to the most basic operation of the system, such as the host name, time zone, logging details, NTP server, language and style.

The host name appears in the top left hand corner of the interface menu. It also appears when you open a Telnet or SSH session.

Note: this document shows no host name in screen grabs. Throughout the document we use the host name 'VA_router'.

The system configuration contains a logging section for the configuration of a Syslog client.

7.1 Configuration package used

Package	Sections
system	main
	timeserver

7.2 Configuring system properties

To set your system properties, in the top menu, click **System**. There are four sections in the System page.

Section	Description
General settings	Configure host name, local time and time zone.
Logging	Configure a router to log to a server. You can configure a Syslog client in this section.
Language and Style	Configure the router's web language and style.
Time synchronization	Configure the NTP server in this section.

7.2.1 General settings

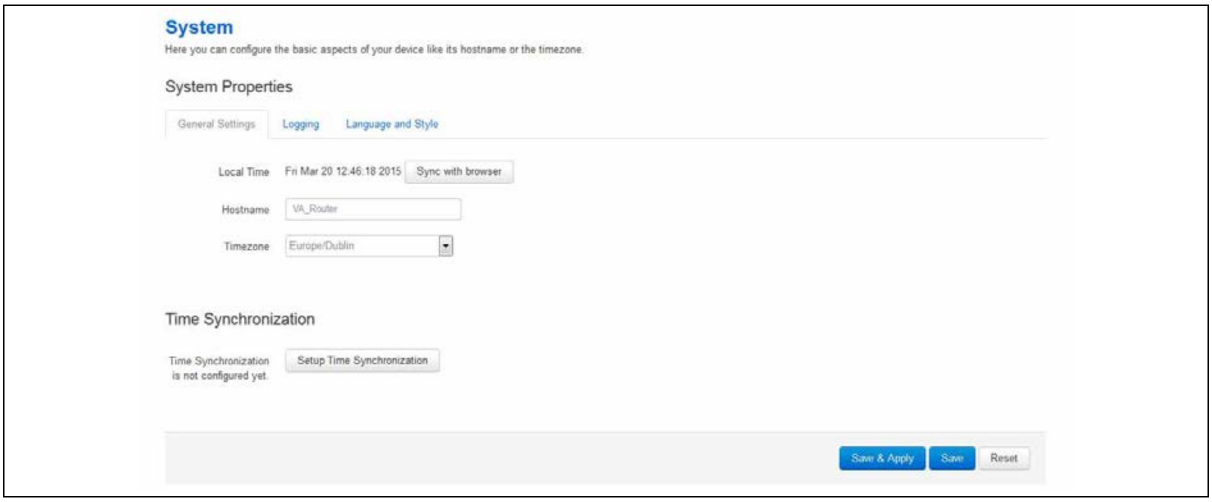


Figure 12: General settings in system properties

Web Field / UCI / Package Option	Description
Web: Local Time	Sets the local time and syncs with browser. You can manually configure on CLI, using: date –s YYYY.MM.DD-hh:mm:ss
Web: hostname UCI: system.main.hostname Opt: hostname	Specifies the hostname for this system.
Web: Timezone UCI: system.main.timezone Opt: timezone	Specifies the time zone that the date and time should be rendered in by default.
Web: n/a UCI: system.main.timezone Opt: time_save_interval_min	Defines the interval in minutes to store the local time for use on next reboot. Range 10m

Table 11: Information table for general settings section

7.2.2 Logging

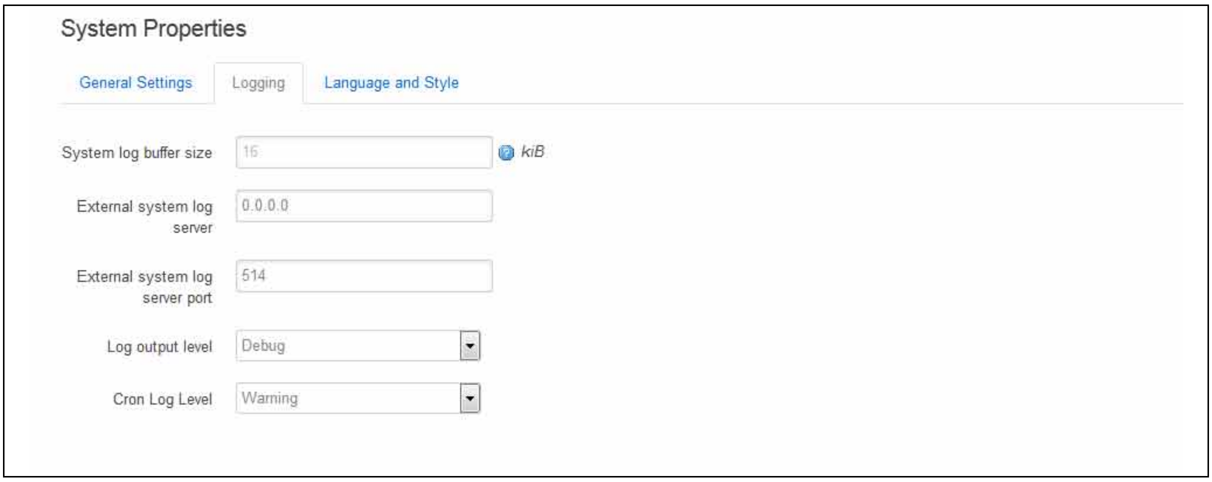


Figure 13: The logging section in system properties

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: System log buffer size
UCI: system.main.log_size
Opt: log_size</td><td>Log buffer size in KB.<table><tr><td>Range</td><td></td></tr><tr><td>16</td><td>16 KB</td></tr></table></td></tr>
<tr><td>Web: External system log server
UCI: system.main.log_ip
Opt: log_ip</td><td>External syslog server IP address.<table><tr><td>Range</td><td></td></tr><tr><td>0.0.0.0</td><td></td></tr></table></td></tr>
<tr><td>Web: External system log server port
UCI: system.main.log_port
Opt: log_port</td><td>External syslog server port number.<table><tr><td>Range</td><td></td></tr><tr><td>514</td><td></td></tr></table></td></tr>
</table>

<table>
<tr><td>Web: Log output level
UCI: system.main.conloglevel
Opt: conloglevel</td><td>Sets the maximum log output level severity for system events. System events are written to the system log. Messages with a lower level or level equal to the configured level are displayed in the console using the logread command, or alternatively written to flash, if configured to do so.
<table>
<tr><th>Web value</th><th>Description</th><th>UCI</th></tr>
<tr><td>Debug</td><td>Information useful to developers for debugging the application.</td><td>8</td></tr>
<tr><td>Info</td><td>Normal operational messages that require no action.</td><td>7</td></tr>
<tr><td>Notice</td><td>Events that are unusual, but not error conditions.</td><td>6</td></tr>
<tr><td>Warning</td><td>May indicate that an error will occur if action is not taken.</td><td>5</td></tr>
<tr><td>Error</td><td>Error conditions</td><td>4</td></tr>
<tr><td>Critical</td><td>Critical conditions</td><td>3</td></tr>
<tr><td>Alert</td><td>Should be addressed immediately</td><td>2</td></tr>
<tr><td>Emergency</td><td>System is unusable</td><td>1</td></tr>
</table></td></tr>
<tr><td>Web: Cron Log Level
UCI: system.main.cronloglevel
Opt: cronloglevel</td><td>Sets the maximum log level for kernel messages to be logged to the console. Only messages with a level lower, or level equal to the configured level will be printed to the console.
<table>
<tr><th>Web value</th><th>Description</th><th>UCI</th></tr>
<tr><td>Normal</td><td>Normal operation messages</td><td>8</td></tr>
<tr><td>Warning</td><td>Error messages</td><td>9</td></tr>
<tr><td>Debug</td><td>Debug messages</td><td>5</td></tr>
</table></td></tr>
<tr><td>Web: n/a
UCI: system.main.log_file
Opt: log_file</td><td>Since logread is only small in size it can be beneficial to write system events to flash. This option defines the file path to write the events. Set to ‘root/syslog.messages’</td></tr>
<tr><td>Web: n/a
UCI: system.main.log_type
Opt: log_type</td><td>Defines whether to write the system events to a file rather than logread. Set to ‘file’ to write to the file configured under log_file option.</td></tr>
</table>

Table 12: Information table for the logging section

7.2.3 Language and style

Figure 14: The language and style section in system properties

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Language</td><td>Sets the language to ‘auto’ or ‘English’.<table><tr><td>Auto</td><td></td></tr><tr><td>English</td><td></td></tr></table></td></tr>
<tr><td>Design</td><td>Sets the router’s style.</td></tr>
</table>

Table 13: Information table for the language and style page

7.2.4 Time synchronization

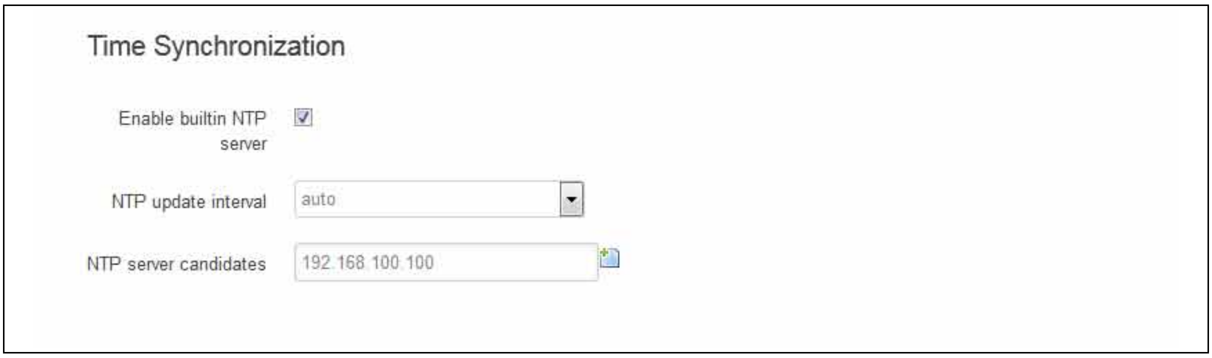


Figure 15: The time synchronization section in system properties

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Enable built-in NTP Server
UCI: system.ntp
Opt: config timeserver</td><td>Enables NTP server.</td></tr>
<tr><td>Web: NTP update interval
UCI: system.ntp.interval_hours
Opt: interval_hours</td><td>Specifies interval of NTP requests in hours. Default value set to auto.<table><tr><td>auto</td><td></td></tr><tr><td>Range</td><td>auto; 1-23</td></tr></table></td></tr>
</table>

Web: NTP server candidates UCI: system.ntp.server Opt: list server	Defines the list of NTP servers to poll the time from. If the list is empty, the built in NTP daemon is not started. Multiple servers can be configured and are separated by a space if using UCI. By default all fields are set to **0.0.0.0**.

Table 14: Information table for time synchronization section

7.3 System reboot

The router can be configured to reboot immediately, or scheduled to reboot a configured time in the future.

In the top menu, select **System -> Reboot**. The System page appears.

Ensure you have saved all your configuration changes before you reboot.

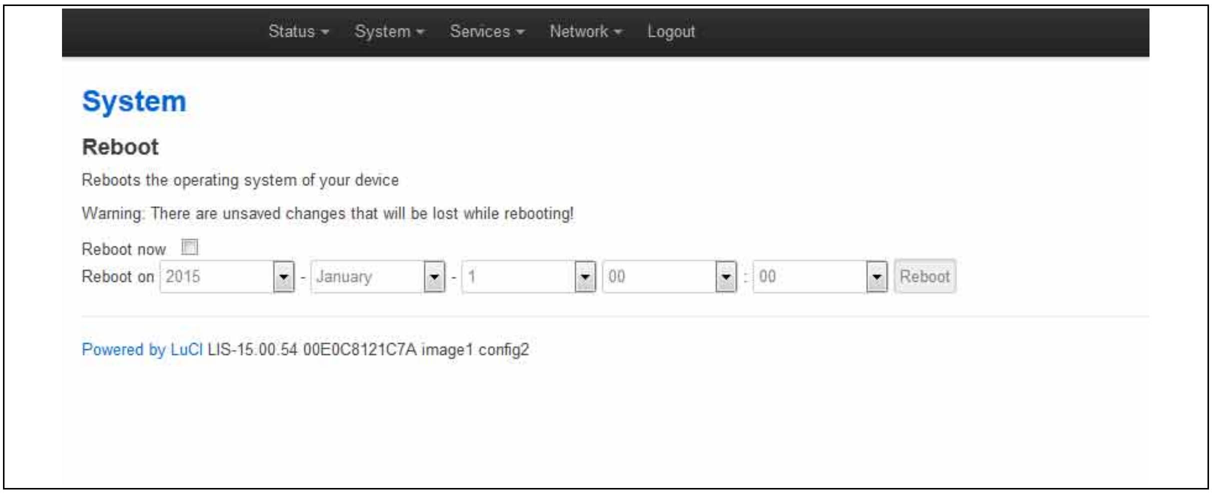


Figure 16: The reboot page

Check the **Reboot now** check box and then click **Reboot**.

7.4 System settings using UCI

```
root@VA_router:~# uci show system
system.main=system
system.main.hostname=VA_router
system.main.timezone=UTC
system.main.log_ip=1.1.1.1
system.main.log_port=514
system.main.conloglevel=8
system.main.cronloglevel=8
system.ntp.interval_hours=auto
system.ntp.server=0.VA_router.pool.ntp.org 10.10.10.10
```

7.5 System settings using package options

```
root@VA_router:~# uci export system
package 'system'

config 'system' 'main'
      option 'hostname' "VA_router"
      option 'timezone' "UTC"
      option 'log_ip' "1.1.1.1"
      option 'log_port' "514"
      option time_save_interval_min "10"
      option conloglevel '8'
      option cronloglevel '8'

config 'timeserver' 'ntp'
      option interval_hours 'auto'
      list server "0.VA_router.pool.ntp.org"
      list server ’10.10.10.10’
```

7.6 System diagnostics

7.6.1 System events

Events in the system have a class, sub class and severity. All events are written to the system log.

7.6.1.1 Logread

To view the system log, use:

```
root@VA_router:~# logread
```

Shows the log.

```
root@VA_router:~# logread |tail
```

Shows end of the log.

```
root@VA_router:~# logread | more
```

Shows the log page by page.

```
root@VA_router:~# logread –f
```

Shows the log on an ongoing basis. To stop this option, press **ctrl-c**.

```
root@VA_router:~# logread –f &
```

Shows the log on an ongoing basis while in the background. This allows you to run other commands while still tracing the event logs. To stop this option, type **fg** to view the current jobs, then press **ctrl-c** to kill those jobs.

7.6.1.2 System events in flash

Since logread is only small in size it can be beneficial to write system events to flash. To do this you need to modify the system config under the system package. Set the options ‘log_file’, ‘log_size’ and ‘log_type’ as below:

```
root@VA_router:~# uci export system
package system
config system 'main'
        option hostname 'VA_router'
        option zonename 'UTC'
        option timezone 'GMT0'
```

```
        option conloglevel '8'
        option cronloglevel '8'
        option time_save_interval_hour '10'
        option log_hostname '%serial'
        option log_ip '1.1.1.1'
        option log_port '514'
        option log_file '/root/syslog.messages'
        option log_size '400'
        option log_type 'file'
```

The above commands will take effect after a reboot.

```
root@VA_router:~# cat /root/syslog.messages
```

Shows all the system events stored in flash.

```
root@VA_router:~# tail /root/syslog.messages
```

Shows end of the events stored flash.

```
root@VA_router:~# tail –f /root/syslog.messages &
```

Shows the log on an ongoing basis. To stop this option, press **ctrl-c**.

8 Upgrading router firmware

Upgrading firmware using the web interface

Copy the new firmware issued by Virtual Access to a PC connected to the router.

In the top menu, select **System tab > Backup/Flash Firmware**. The Flash operations page appears.

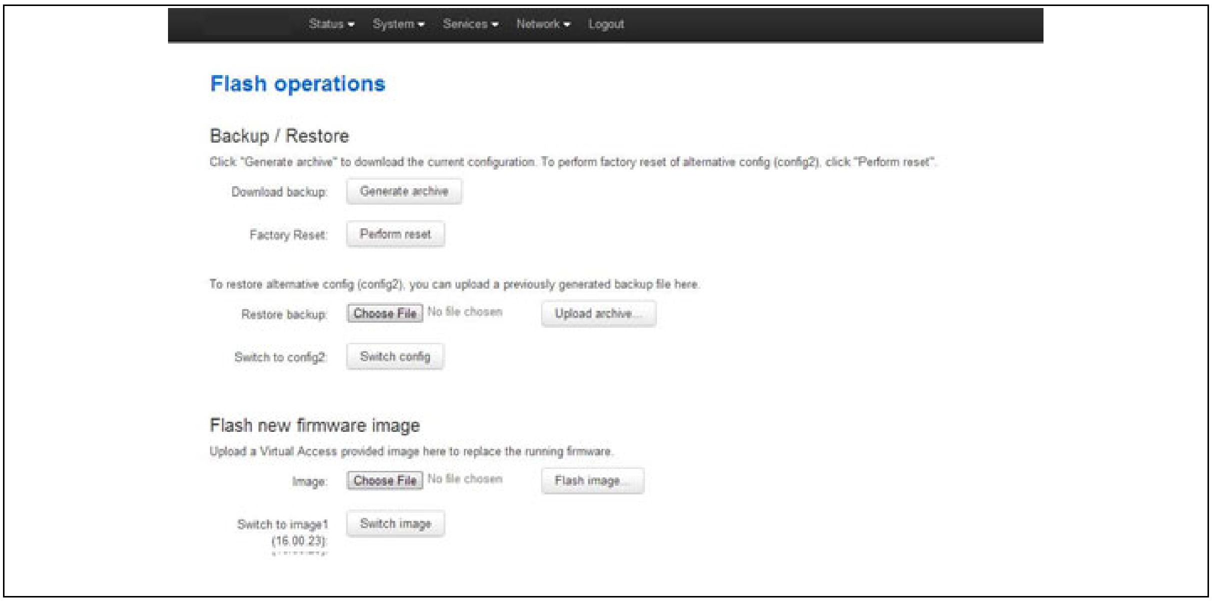


Figure 17: The flash operations page

Under Flash new firmware image, click **Choose File or Browse**.

Note: the button will vary depending on the browser you are using.

Select the appropriate image and then click **Flash Image**. The Flash Firmware – Verify page appears.

Figure 18: The flash firmware - verify page

Click **Proceed**. The System – Flashing… page appears.

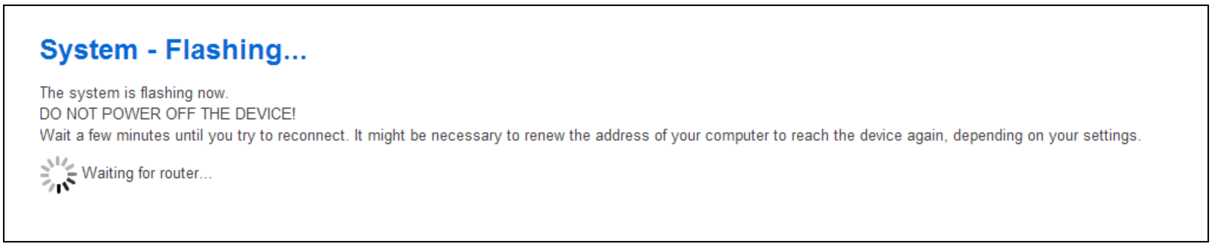


Figure 19: The system – flashing…page

When the ‘waiting for router’ icon disappears, the upgrade is complete, and the login homepage appears.

To verify that the router has been upgraded successfully, click **Status** in the top menu. The Firmware Version shows in the system list.

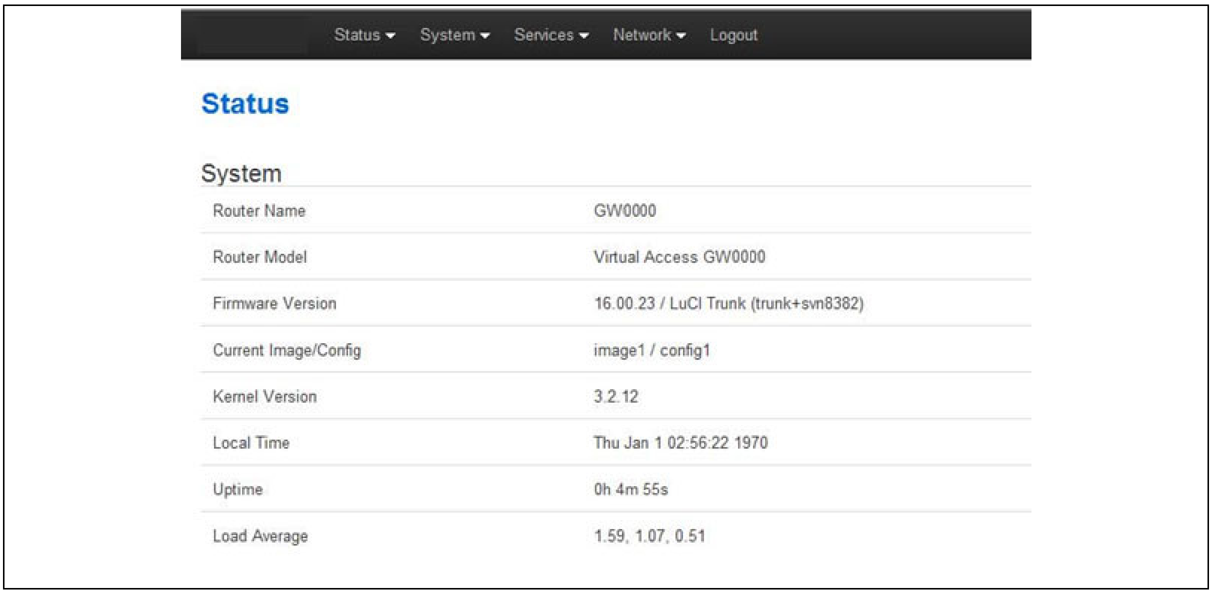


Figure 20: The status page

8.1 Upgrading firmware using CLI

To upgrade firmware using CLI, you will need a TFTP server on a connected PC.

Open up an SSH or Telnet session to the router.

Enter in the relevant username and password.

To change into the temp folder, enter:

cd /tmp

To connect to your TFTP server, enter:

```
atftp x.x.x.x
```

(where x.x.x.x is the IP of your PC).

Press **Enter**.

While in the TFTP application, to get the image enter:

```
get GIG-15.00.38.image
```

Note: this is an example, substitute the correct file name.

When the image has downloaded, to leave TFPT and get back into the command line, enter:

```
quit
```

To write the image into the alternative image, enter:

```
mtd write GIG-15.00.38.image altimage
```

Note: this is an example, substitute the correct file name.

To set the next image to boot to the alternative image, enter:

```
vacmd set next image altimage
```

For your configuration changes to apply, you must reboot your router. Enter:

```
reboot
```

9 Router file structure

This section describes the file structure and location of essential directories and files on Virtual Access routers.

Throughout this document, we use information tables to show the different ways to configure the router using the router's web and command line (CLI).

When showing examples of the command line interface we use the host name 'VA_router' to indicate the system prompt. For example, the table below displays what the user should see when entering the command to show the current configuration in use on the router:

```
root@VA_router:~# va_config.sh
```

9.1 System information

General information about software and configuration used by the router is displayed on the Status page. To view the running configuration file status on the web interface, in the top menu, select **Status -> Overview**. This page also appears immediately after you have logged in.

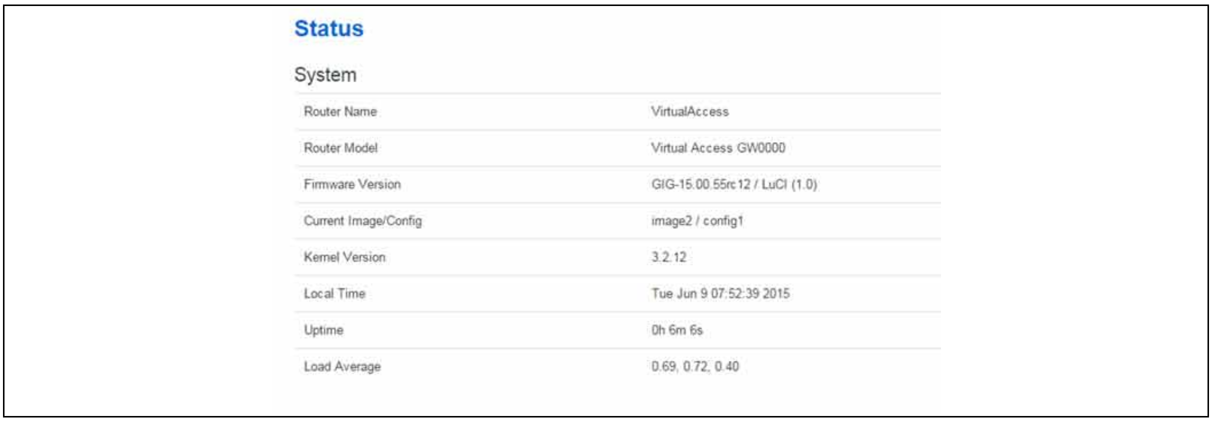


Figure 21: The status page

System information is also available from the CLI if you enter the following command:

```
root@VA_router:~# va_vars.sh
```

The example below shows the output from the above command.

```
VA_SERIAL:              00E0C8121215
VA_MODEL:               GW6610-ALL
VA_ACTIVEIMAGE:         image2
VA_ACTIVECONFIG:        config1
VA_IMAGE1VER:           VIE-16.00.44
VA_IMAGE2VER:           VIE-16.00.44
```

9.2 Image files

The system allows for two firmware image files:

- image1, and
- image2

Two firmware images are supported to enable the system to rollback to a previous firmware version if the upgrade of one image fails.

The image names (image1, image2) themselves are symbols that point to different partitions in the overall file system. A special image name "altimage" exists which always points to the image that is not running.

The firmware upgrade system always downloads firmware to "altimage".

9.3 Directory locations for UCI configuration files

Router configurations files are stored in folders at:

/etc/factconf,

/etc/config1

and

/etc/config2

Multiple configuration files exist in each folder. Each configuration file contains configuration parameters for different areas of functionality in the system.

A symbolic link exists at:

/etc/config, which always points to one of factconf, config1 or config2 is the active configuration file.

Files that appear to be in **/etc/config** are actually in **/etc/factconf|config1|config2** depending on which configuration is active.

If **/etc/config** is missing on start-up, for example on first boot, the links and directories are created with configuration files copied from **/rom/etc/config/**.

At any given time, only one of the configurations is the active configuration. The UCI system tool (Unified Configuration Interface) only acts upon the currently active configuration.

9.4 Viewing and changing current configuration

To show the configuration currently running, enter:

```
root@VA_router:~# va_config.sh
```

To show the configuration to run after the next reboot, enter:

```
root@VA_router:~# va_config.sh next
```

To set the configuration to run after the next reboot, enter:

```
root@VA_router:~# va_config.sh -s [factconf|config1|config2|altconfig]
```

9.5 Configuration file syntax

The configuration files consist of sections – or packages - that contain one or more config statements. These optional statements define actual values.

Below is an example of a simple configuration file.

```
package 'example'
config 'example' 'test'
        option   'string'      'some value'
        option   'boolean'     '1'
        list     'collection'  'first item'
        list     'collection'  'second item'
```

The config `'example' 'test'` statement defines the start of a section with the type example and the name test.

Command	Target	Description
export	[<config>]	Exports the configuration in a machine readable format. It is used internally to evaluate configuration files as shell scripts.
import	[<config>]	Imports configuration files in UCI syntax.
add	<config> <section-type>	Adds an anonymous section of type-section type to the given configuration.

add_list	<config>.<section>.<option>=<string>	Adds the given string to an existing list option.
show	[<config>[.<section>[.<option>]]]	Shows the given option, section or configuration in compressed notation.
get	<config>.<section>[.<option>]	Gets the value of the given option or the type of the given section.
Set	<config>.<section>[.<option>]=<value>	Sets the value of the given option, or adds a new section with the type set to the given value.
delete	<config>[.<section[.<option>]]	Deletes the given section or option.

Table 1: Common commands, target and their descriptions

9.6 Managing configurations

9.6.1 Managing sets of configuration files using directory manipulation

Configurations can also be managed using directory manipulation.

To remove the contents of the current folder, enter:

```
root@VA_router:/etc/config1# rm –f *
```

Warning: the above command makes irreversible changes.

To remove the contents of a specific folder regardless of the current folder (config2), enter:

```
root@VA_router:/ # rm –f /etc/config1/*
```

Warning: the above command makes irreversible changes.

To copy the contents of one folder into another (config2 into config1), enter:

```
root@VA_router:/etc/config1# cp /etc/config2/* /etc/config1
```

10 Using the Command Line Interface

This chapter explains how to view Virtual Access routers' log files and edit configuration files using a Command Line Interface (CLI) and the Unified Configuration Interface (UCI) system.

10.1 Overview of some common commands

Virtual Access router's system has an SSH server typically running on port 22.

The factconf default password for the root user is admin.

To change the factconf default password, enter:

```
root@VA_router:/# uci set system.main.password="******"

root@VA_router:/# uci commit system
```

To reboot the system, enter:

```
root@VA_router:/# reboot
```

The system provides a Unix-like command line. Common Unix commands are available such as `ls`, `cd`, `cat`, `top`, `grep`, `tail`, `head`, `more` and `less`.

Typical pipe and redirect operators are also available, such as: `>`, `>>`, `<`, `|`

The system log can be viewed using any of the following commands:

```
root@VA_router:/# logread


root@VA_router:/# logread | tail


root@VA_router:/# logread –f
```

These commands will show the full log, end of the log (tail) and continuously (-f). Enter **Ctrl-C** to stop the continuous output from logread -f.

To view and edit configuration files, the system uses the "Unified Configuration Interface" (UCI) which is described further on in this chapter. This is the preferred method of editing configuration files. However, these files can also be viewed and edited using some of the standard Unix tools.

For example, to view a text or configuration file in the system, enter:

```
root@VA_router:/# cat /etc/passwd
```

The command output information shows the following, or similar output.

```
root:x:0:0:root:/root:/bin/ash
daemon:*:1:1:daemon:/var:/bin/false
ftp:*:55:55:ftp:/home/ftp:/bin/false
sftp:*:56:56:sftp:/var:/usr/lib/sftp-server
network:*:101:101:network:/var:/bin/false
nobody:*:65534:65534:nobody:/var:/bin/false
```

To view files in the current folder, enter:

```
root@VA_router:/# ls

 bin      etc      lib      opt      sbin     usr
 bkrepos  home     linuxrc  proc     sys      var
 dev      init     mnt      root     tmp      www
```

For more details add the -l argument:

```
root@VA_router:/# ls -l

drwxrwxr-x    2 root     root   642 Jul 16  2012 bin
drwxr-xr-x    5 root     root  1020 Jul  4 01:27 dev
drwxrwxr-x    1 root     root     0 Jul  3 18:41 etc
drwxr-xr-x    1 root     root     0 Jul  9  2012 lib
drwxr-xr-x    2 root     root     3 Jul 16  2012 mnt
drwxr-xr-x    7 root     root     0 Jan  1  1970 overlay
dr-xr-xr-x   58 root     root     0 Jan  1  1970 proc
drwxr-xr-x   16 root     root   223 Jul 16  2012 rom
drwxr-xr-x    1 root     root     0 Jul  3 22:53 root
drwxrwxr-x    2 root     root   612 Jul 16  2012 sbin
drwxr-xr-x   11 root     root     0 Jan  1  1970 sys
drwxrwxrwt   10 root     root   300 Jul  4 01:27 tmp
drwxr-xr-x    1 root     root     0 Jul  3 11:37 usr
lrwxrwxrwx    1 root     root     4 Jul 16  2012 var -> /tmp
drwxr-xr-x    4 root     root    67 Jul 16  2012 www
```

To change the current folder, enter cd followed by the desired path:

```
root@VA_router:/# cd /etc/config1
root@VA_router:/etc/config1#
```

Note: if the specified directory is actually a link to a directory, the real directory will be shown in the prompt.

To view scheduled jobs, enter:

```
root@VA_router:/# crontab –l

 0 * * * * slaupload 00FF5FF92752 TFTP 1 172.16.250.100 69
```

To view currently running processes, enter:

```
root@VA_router:/# ps

 PID  Uid     VmSize Stat Command
    1 root        356 S   init
    2 root            DW  [keventd]
    3 root            RWN [ksoftirqd_CPU0]
    4 root            SW  [kswapd]
    5 root            SW  [bdflush]
    6 root            SW  [kupdated]
    8 root            SW  [mtdblockd]
   89 root        344 S   logger -s -p 6 -t
   92 root        356 S   init
   93 root        348 S   syslogd -C 16
   94 root        300 S   klogd
  424 root        320 S   wifi up
  549 root        364 S   httpd -p 80 -h /www -r VA_router
  563 root        336 S   crond -c /etc/crontabs
 6712 root        392 S   /usr/sbin/dropbear
 6824 root        588 S   /usr/sbin/dropbear
 7296 root        444 S   -ash
  374 root        344 R   ps ax
  375 root        400 S   /bin/sh /sbin/hotplug button
  384 root        396 R   /bin/sh /sbin/hotplug button
  385 root            RW  [keventd]
```

To search for a process, enter `pgrep -fl '<process name or part of name>'`:

```
root@VA_router:/# pgrep -fl ‘wifi’
```

```
424 root        320 S   wifi up
```

To kill a process, enter the PID:

```
root@VA_router:~# kill 424
```

10.2 Using Unified Configuration Interface (UCI)

The system uses Unified Configuration Interface (UCI) for central configuration management. Most common and useful configuration settings can be accessed and configured using the UCI system.

UCI consists of a command line utility (CLI), the files containing the actual configuration data, and scripts that take the configuration data and apply it to the proper parts of the system, such as the networking interfaces. Entering the command 'uci' on its own will display the list of valid arguments for the command and their format.

```
root@VA_router:/lib/config# uci

Usage: uci [<options>] <command> [<arguments>]

Commands:
export     [<config>]
import     [<config>]
changes    [<config>]
commit     [<config>]
add        <config> <section-type>
add_list   <config>.<section>.<option>=<string>
show       [<config>[.<section>[.<option>]]]
get        <config>.<section>[.<option>]
set        <config>.<section>[.<option>]=<value>
delete     <config>[.<section[.<option>]]
rename     <config>.<section>[.<option>]=<name>
```

```
revert     <config>[.<section>[.<option>]]
Options:
-c <path>  set the search path for config files (default: /etc/config)
-d <str>   set the delimiter for list values in uci show
-f <file>  use <file> as input instead of stdin
-m         when importing, merge data into an existing package
-n         name unnamed sections on export (default)
-N         don't name unnamed sections
-p <path>  add a search path for config change files
-P <path>  add a search path for config change files and use as default
-q         quiet mode (don't print error messages)
-s         force strict mode (stop on parser errors, default)

-S         disable strict mode
-X         do not use extended syntax on 'show'
```

The table below describes commands for the UCI command line and some further examples of how to use this utility.

Command	**Target**	**Description**
commit	[<config>]	Writes changes of the given configuration file, or if none is given, all configuration files, to the filesystem. All "uci set", "uci add", "uci rename" and "uci delete" commands are staged into a temporary location and written to flash at once with "uci commit". This is not needed after editing configuration files with a text editor, but for scripts, GUIs and other programs working directly with UCI files.
export	[<config>]	Exports the configuration in a UCI syntax and does validation.
import	[<config>]	Imports configuration files in UCI syntax.
changes	[<config>]	Lists staged changes to the given configuration file or if none given, all configuration files.
add	<config> <section-type>	Adds an anonymous section of type section-type to the given configuration.
add_list	<config>.<section>.<option>=<string>	Adds the given string to an existing list option.

show	[<config>[.<section>[.<option>]]]	Shows the given option, section or configuration in compressed notation.
get	<config>.<section>[.<option>]	Gets the value of the given option or the type of the given section.
set	<config>.<section>[.<option>]=<value>	Sets the value of the given option, or add a new section with the type set to the given value.
delete	<config>[.<section[.<option>]]	Deletes the given section or option.
rename	<config>.<section>[.<option>]=<name>	Renames the given option or section to the given name.
revert	<config>[.<section>[.<option>]]	Deletes staged changes to the given option, section or configuration file.

Table 15: Common commands, target and their descriptions

Note: all operations do not act directly on the configuration files. A commit command is required after you have finished your configuration.

```
root@VA_router:~# uci commit
```

10.2.1 Using uci commit to avoid router reboot

After changing the port, uhttpd listens on from 80 to 8080 in the file /etc/config/uhttpd; save it, then enter:

```
root@VA_router:~# uci commit uhttpd
```

Then enter:

```
root@VA_router:~# /etc/init.d/uhttpd restart
```

For this example, the router does not need to reboot as the changes take effect when the specified process is restarted.

10.2.2 Export a configuration

Using the uci export command it is possible to view the entire configuration of the router or a specific package. Using this method to view configurations does not show comments that are present in the configuration file:

```
root@VA_router:~# uci export httpd


package 'httpd'
config 'httpd'
option 'port' '80'
```

```
option 'home' '/www'
```

10.2.3 Show a configuration tree

The configuration tree format displays the full path to each option. This path can then be used to edit a specific option using the `uci set` command.

To show the configuration 'tree' for a given config, enter `uci show <package>`.

```
root@VA_router:/# uci show network


network.loopback=interface
network.loopback.ifname=lo
network.loopback.proto=static
network.loopback.ipaddr=127.0.0.1
network.loopback.netmask=255.0.0.0
network.lan=interface
network.lan.ifname=eth0
network.lan.proto=dhcp
network.wan=interface
network.wan.username=foo
network.wan.password=bar
network.wan.proto=3g
network.wan.device=/dev/ttyACM0
network.wan.service=umts
network.wan.auto=0
network.wan.apn=arkessa.com
network.@va_switch[0]=va_switch
network.@va_switch[0].eth0=A B C
network.@va_switch[0].eth1=D
```

It is also possible to display a limited subset of a configuration:

```
root@VA_router:/# uci show network.wan
network.wan=interface
network.wan.username=foo
network.wan.password=bar
```

```
network.wan.proto=3g
network.wan.device=/dev/ttyACM0
network.wan.service=umts
network.wan.auto=0
network.wan.apn=hs.vodafone.ie
```

10.2.4 Display just the value of an option

To display a specific value of an individual option within a package, enter `uci get`

```
root@VA_router:~# uci get httpd.@httpd[0].port
80
root@VA_router:~#
High level image commands
The image running at present can be shown using the command:
root@VA_router:~# vacmd show current image
The image to run on next reboot can be set using the command:
root@VA_router:~# vacmd set next image [image1|image2|altimage]
root@VA_router:~# reboot
```

10.2.5 Format of multiple rules

When there are multiple rules next to each other, UCI uses array-like references for them. For example, if there are 8 NTP servers, UCI will let you reference their sections as `timeserver.@timeserver[0]` for the first section;

or

`timeserver.@timeserver[7]` for the last section.

You can also use negative indexes, such as `timeserver.@timeserver[-1]`

'`-1`' means the last one, and '`-2`' means the second-to-last one. This is useful when appending new rules to the end of a list.

```
root@VA_router:/# uci show va_eventd
va_eventd.main=va_eventd
va_eventd.main.enabled=yes
va_eventd.main.event_queue_file=/tmp/event_buffer
```

```
va_eventd.main.event_queue_size=128K
va_eventd.@conn_tester[0]=conn_tester
va_eventd.@conn_tester[0].name=Pinger
va_eventd.@conn_tester[0].enabled=yes
va_eventd.@conn_tester[0].type=ping
va_eventd.@conn_tester[0].ping_dest_addr=192.168.250.100
va_eventd.@conn_tester[0].ping_success_duration_sec=5
va_eventd.@target[0]=target
va_eventd.@target[0].name=MonitorSyslog
va_eventd.@target[0].enabled=yes
va_eventd.@target[0].type=syslog
va_eventd.@target[0].target_addr=192.168.250.100
va_eventd.@target[0].conn_tester=Pinger
va_eventd.@target[0].suppress_duplicate_forwardings=no
va_eventd.@forwarding[0]=forwarding
va_eventd.@forwarding[0].enabled=yes
va_eventd.@forwarding[0].className=ethernet
va_eventd.@forwarding[0].target=MonitorSyslog
va_eventd.@forwarding[1]=forwarding
va_eventd.@forwarding[1].enabled=yes
va_eventd.@forwarding[1].className=auth
va_eventd.@forwarding[1].target=MonitorSyslog
va_eventd.@forwarding[2]=forwarding
va_eventd.@forwarding[2].enabled=yes
va_eventd.@forwarding[2].className=adsl
va_eventd.@forwarding[2].target=MonitorSyslog
va_eventd.@forwarding[3]=forwarding
va_eventd.@forwarding[3].enabled=yes
va_eventd.@forwarding[3].className=ppp
va_eventd.@forwarding[3].target=MonitorSyslog
```

10.3 Configuration files

The table below lists common package configuration files that can be edited using uci commands. Other configuration files may also be present depending on the specific options available on the Virtual Access router.

File	Description
Management	
/etc/config/autoload	Boot up Activation behaviour (typically used in factconf)
/etc/config/httpclient	Activator addresses and urls
/etc/config/monitor	Monitor details
Basic	
/etc/config/dropbear	SSH server options
/etc/config/dhcp	Dnsmasq configuration and DHCP settings
/etc/config/firewall	NAT, packet filter, port forwarding, etc.
/etc/config/network	Switch, interface, L2TP and route configuration
/etc/config/system	Misc. system settings including syslog
Other	
/etc/config/snmpd	SNMPd settings
/etc/config/uhttpd	Web server options (uHTTPd)
/etc/config/strongswan	IPSec settings

10.4 Configuration file syntax

The configuration files usually consist of one or more config statements, so-called sections with one or more option statements defining the actual values.

Below is an example of a simple configuration file.

```
package 'example'
config 'example' 'test'
        option   'string'      'some value'
        option   'boolean'     '1'
        list     'collection'  'first item'
        list     'collection'  'second item'
```

The config `'example' 'test'` statement defines the start of a section with the type example and the name test. There can also be so-called anonymous sections with only a type, but no name identifier. The type is important for the processing programs to decide how to treat the enclosed options.

The option `'string' 'some value'` and option `'boolean' '1'` lines define simple values within the section.

Note: there are no syntactical differences between text and boolean options. Per convention, boolean options may have one of the values `'0', 'no', 'off'` or `'false'` to specify a false value or `'1' , 'yes', 'on'` or `'true'` to specify a true value.

In the lines starting with a list keyword, an option with multiple values is defined. All list statements that share the same name, collection in our example, will be combined into a single list of values with the same order as in the configuration file.

The indentation of the option and list statements is a convention to improve the readability of the configuration file but it is not syntactically required.

Usually you do not need to enclose identifiers or values in quotes. Quotes are only required if the enclosed value contains spaces or tabs. Also it is legal to use double-quotes instead of single-quotes when typing configuration options.

All of the examples below are valid syntax.

```
option example value
option 'example' value
option example "value"
option "example"    'value'
option   'example' "value"
```

In contrast, the following examples are not valid syntax.

```
option 'example" "value'
```

(quotes are unbalanced)

```
option example some value with space
```

(note the missing quotes around the value).

It is important to note that identifiers and config file names may only contain the characters `a-z, A-Z, 0-9` and `_`. However, option values may contain any character, as long they are properly quoted.

11 Management configuration settings

This chapter contains the configuration sections and parameters required to manage and monitor your device using Activator and Monitor.

11.1 Activator

Activator is a Virtual Access proprietary provisioning system, where specific router configurations and firmware can be stored to allow central management and provisioning. Activator has two distinct roles in provisioning firmware and configuration files to a router.

- Zero touch activation of firmware and configuration files on router boot up
  - In this scenario the router will initiate the requesting of firmware and configuration files on boot and is generally used for router installation. The router will be installed with a factory config that will allow it to contact Activator. The autoload feature controls the behaviour of the router in requesting firmware and configuration files; this includes when to start the Activation process and the specific files requested. The HTTP Client (uhttpd) contains information about the Activator server and the protocol used for activation.
- Deployment of firmware to routers after installation
  - In this scenario, Activator will initiate the process. This process, known as Active Update, allows for central automatic deployment of firmware and configuration files. It is used when configuration or firmware changes need to be pushed to live routers.

11.2 Monitor

Monitor is a Virtual Access proprietary tool, based on SNMP protocol, to monitor wide networks of deployed routers. The router will be configured to send information to Monitor, which is then stored and viewed centrally via the Monitor application. This includes features such as traffic light availability status, syslog and SLA monitoring.

11.3 Configuration packages used

Package	Sections
autoload	main
httpclient	default
management_users	user

11.4 Autoload: boot up activation

Autoload configurations specify how the device should behave with respect to activation when it boots up. Autoload entries contain information about the specific files to be downloaded and the destination for the downloaded file. Standard autoload entry configurations to download are:

- A firmware file ($$.img)
- A configuration file ($$.ini)
- A .vas file ($$.vas). This file signals the end of the autolaod sequence to Activator

Activator identifies the device using the serial number of the router. $$ syntax is used to denote the serial number of the router when requesting a file. The requested files are written to the alternate image or config segment.

You can change the settings either directly in the configuration file or via appropriate UCI set commands. It is normal procedure for autoload to be enabled in the router’s factory settings and disabled in running configurations (config 1 and 2).

Autoload may already have been set at factory config level. If you wish to enable autoload services, proceed through the following steps.

11.4.1 Autoload packages

Package	Sections
autoload	main

11.4.2 Create a configuration file

In the top menu, select **Services ->Autoload.** The Autoload page has two sections: Basic Settings and Entries. Click **Add** to access configuration settings for each section.

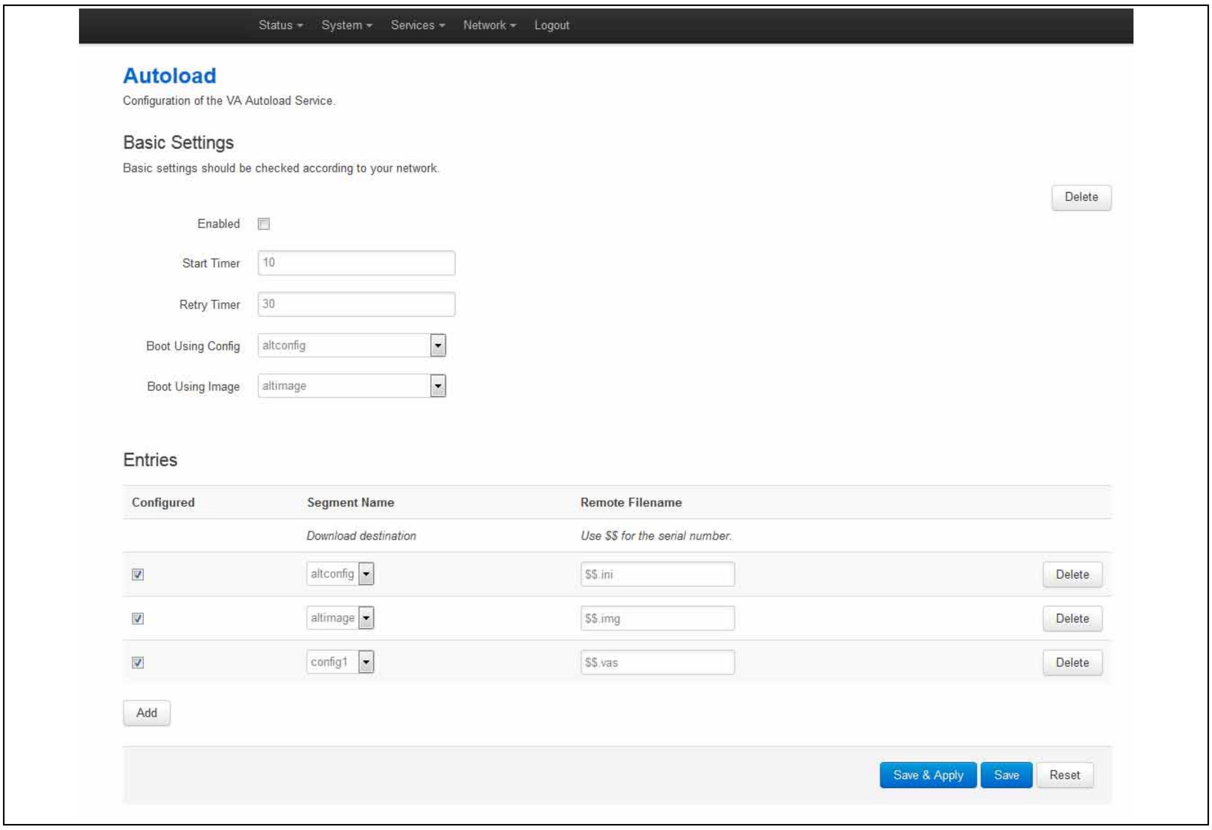


Figure 22: The autoload settings page

Web Field / UCI / Package Option	Description
Basic settings	
Web: Enabled UCI: autoload.main.enabled Opt: Enabled	Enables activation at system boot. 1	Enabled. 0	Disabled.
Web: Start Timer UCI: autoload.main.StartTimer Opt: StartTimer	Defines how long to wait after the boot up completes before starting activation. 10 Range	0-300 secs
Web: Retry Timer UCI: autoload.main.RetryTimer Opt: RetryTimer	Defines how many seconds to wait between retries if a download of a particular autoload entry fails. 30 Range	0-300 secs
Web: N/A UCI: autoload.main.NumberOfRetries Opt: Numberofretries	Defines how many retries to attempt before failing the overall activation sequence, backing off and trying the whole activation sequence again. 5 Range
Web: N/A UCI: autoload.main.BackoffTimer Opt: Backofftimer	Defines how many minutes to back off for if a download and all retires fail. After the backoff period, the entire autoload sequence will start again. 15 Range

Web: Boot Using Config UCI: autoload.main.BootUsingConfig Opt: BootUsingConfig	Specifies which configuration to boot up with after the activation sequence. <table><tr><td>Altconfig</td><td>Alternative configuration</td></tr><tr><td>Config1</td><td>Configuration 1</td></tr><tr><td>Config2</td><td>Configuration 2</td></tr><tr><td>Factconf</td><td>Factory configuration</td></tr></table>
Web: Boot Using Image UCI: autoload.main.BootUsingImage Opt: BootUsingImage	Specifies which image to boot up with after the activation sequence completes successfully. <table><tr><td>Altimage</td><td>Alternative image</td></tr><tr><td>Image 1</td><td>image 1</td></tr><tr><td>Image 2</td><td>image 2</td></tr></table>
Entries	
Web: Configured UCI: autoload.@entry[x].Configured Opt: Configured	Enables the autoload sequence to process this entry. <table><tr><td>1</td><td>Enabled.</td></tr><tr><td>0</td><td>Disabled.</td></tr></table>
Web: Segment Name UCI: autoload.@entry[x].SegmentName Opt: SegmentName	Defines where the downloaded file should be stored: (config1 | config2 | altconfig | image1 | image2 | altimage). Typically only altconfig and altimage are used.
Web: RemoteFilename UCI: autoload.@entry[x].RemoteFilename Opt: RemoteFilename	Defines the name of the file to be downloaded from Activator. <table><tr><td>$$.vas</td><td>Notifies activator sequence is complete.</td></tr><tr><td>$$ ini</td><td>Request configuration</td></tr><tr><td>$$ img</td><td>Request firmware</td></tr></table>**Note**: $$.vas should always be requested last.

Table 16: Information table for autoload

11.4.3 Autoload using UCI

```
root@VA_router:/# uci show autoload
autoload.main=core
autoload.main.Enabled=yes
autoload.main.StartTimer=10
autoload.main.RetryTimer=30
autoload.main.NumberOfRetries=5
autoload.main.BackoffTimer=15
autoload.main.BootUsingConfig=altconfig
autoload.main.BootUsingImage=altimage
autoload.@entry[0]=entry
```

```
autoload.@entry[0].Configured=yes
autoload.@entry[0].SegmentName=altconfig
autoload.@entry[0].RemoteFilename=$$.ini
autoload.@entry[1]=entry
autoload.@entry[1].Configured=yes
autoload.@entry[1].SegmentName=altimage
autoload.@entry[1].RemoteFilename=$$.img
autoload.@entry[2]=entry
autoload.@entry[2].Configured=yes
autoload.@entry[2].SegmentName=config1
autoload.@entry[2].RemoteFilename=$$.vas
```

11.4.4 Autoload using package options

```
root@VA_router:/# uci export autoload
package 'autoload'

config 'core' 'main'
      option 'Enabled' "yes"
      option 'StartTimer' "10"
      option 'RetryTimer' "30"
      option 'NumberOfRetries' "5"
      option 'BackoffTimer' "15"
      option 'BootUsingConfig' "altconfig"
      option 'BootUsingImage' "altimage"

config 'entry'
      option 'Configured' "yes"
      option 'SegmentName' "altconfig"
      option 'RemoteFilename' "\$\$.ini"

config 'entry'
      option 'Configured' "yes"
      option 'SegmentName' "altimage"
      option 'RemoteFilename' "\$\$.img"
```

```
config 'entry'
      option 'Configured' "yes"
      option 'SegmentName' "config1"
      option 'RemoteFilename' "\$\$.vas"
```

11.5 Http Client: configuring activation using the web interface

This section contains the settings for the HTTP Client used during activation and active updates of the device.

The httpclient core section configures the basic functionality of the module used for retrieving files from Activator during the activation process.

11.5.1 HTTP Client configuraton packages

Package	**Sections**
Httpclient	default

11.5.2 Web configuration

To configure HTTP Client for Activator, in the top menu, click **Services -> HTTP Client**. The HTTP Client page has two sections: Basic Settings and Advanced Settings.

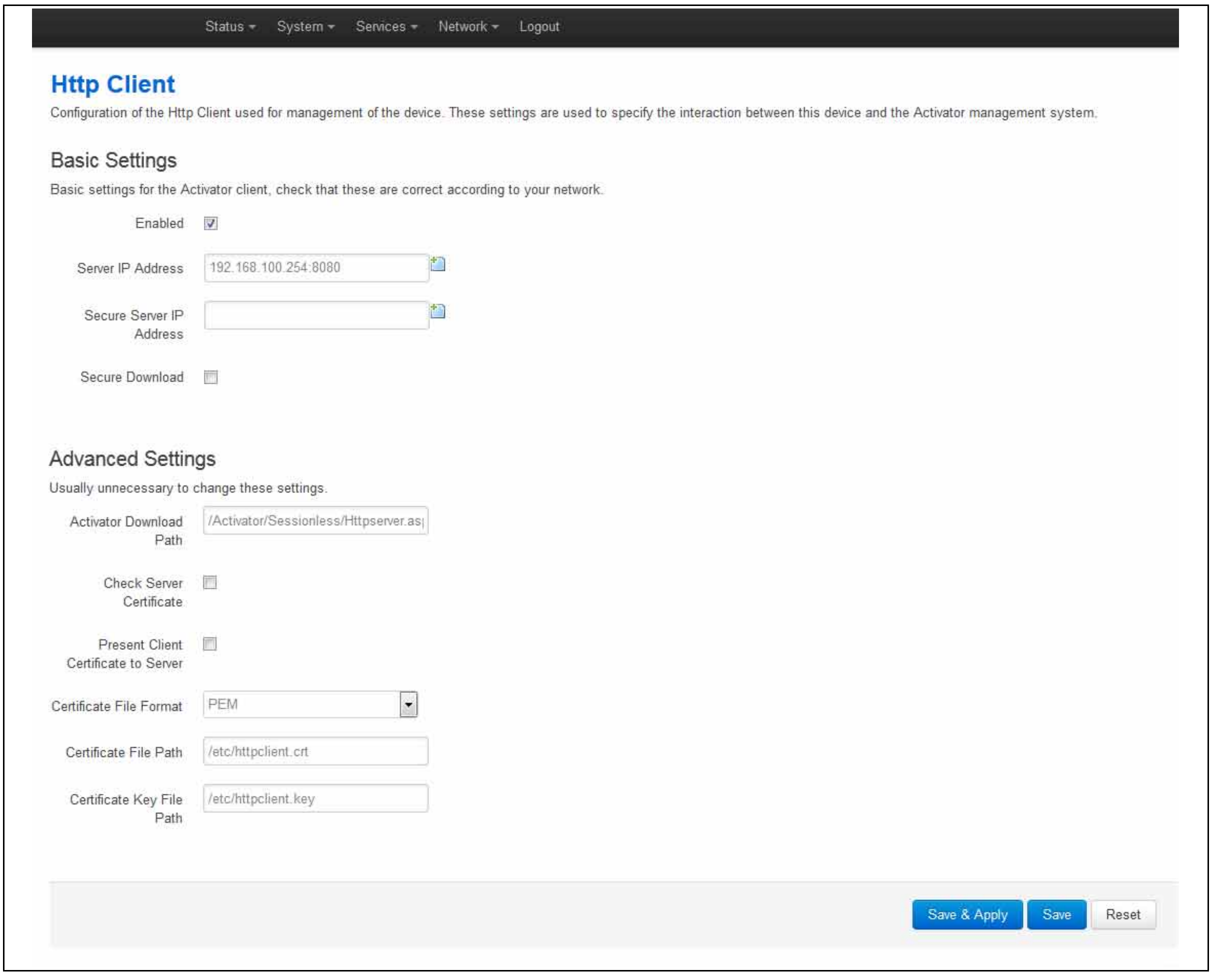


Figure 23: The HTTP client page

Web Field / UCI / Package Option	Description
Basic settings	
Web: Enabled UCI: httpclient.default.enabled Opt: Enabled	Enables the HTTP client. 1 — Enabled. 0 — Disabled.
Web: Server IP Address UCI: httpclient.default.Fileserver Opt: list Fileserver	Specifies the address of Activator that uses http port 80. This can be an IP address or FQDN. The syntax should be x.x.x.x:80 or FQDN:80. Multiple servers should be separated by a space using UCI.
Web: Secure Server IP Address UCI: httpclient.default.SecureFileServer Opt: ListSecureFileServer	Specifies the address of Secure Activator that uses port 443. This can be an IP address or FQDN. The syntax should be x.x.x.x:443 or FQDN:443. Multiple servers should be separated by a space using UCI.
Web: Secure Download UCI: httpclient.default.SecureDownload Opt: SecureDownload	Enables Secure Download (port 443). 1 — Enabled. 0 — Disabled.

Advanced settings	
Web: ActivatorDownloadPath UCI: httpclient.default.ActivatorDownloadPath Opt: ActivatorDownloadPath	Specifies the URL on Activator to which the client should send requests. /Activator/Sessionless/Httpserver.asp Range
Web: Check Server Certificate UCI: httpclient.default.ValidateServerCertificateEnabled Opt: ValidateServerCertificateEnabled	Checks for the certificates presence and validity. 1 — Enabled. 0 — Disabled.
Web: Present Client Certificate to Server UCI: httpclient.default. PresentCertificateEnabled Opt: PresentCertificateEnabled	Specifies if the client presents its certificate to the server to identify itself. 1 — Enabled. 0 — Disabled.
Web: CertificateFile Format UCI: httpclient.default.CertificateFormat Opt: CertificateFormat	Specifies the value the client expects to see in the specified field in the server certificate. PEM DER
Web: Certificate File Path UCI: httpclient.default.CertificateFile Opt: CertificateFile	Defines the directory/location of the certificate. /etc/httpclient.crt Range
Web: Certificate Key File Path UCI: httpclient.default.CertificateKey Opt: CertificateKey	Specifies the directory/location of the certificate key. /etc/httpclient.key Range
Web: N/A UCI: ValidateServerCertificateFieldEnabled Opt: ValidateServerCertificate	Defines the field in the server certificate that the client should check. 1 — Enabled. 0 — Disabled.

Table 17: Information table for HTTP client

11.5.3 Httpclient: Activator configuration using UCI

```
root@VA_router:~# uci show httpclient
httpclient.default=core
httpclient.default.Enabled=yes
httpclient.default.FileServer=10.1.83.36:80 10.1.83.37:80
httpclient.default.SecureFileServer=10.1.83.36:443 10.1.83.37:443
httpclient.default.ActivatorDownloadPath=/Activator/Sessionless/Httpserver.asp
httpclient.default.SecureDownload=no
```

```
httpclient.default.PresentCertificateEnabled=no
httpclient.default.ValidateServerCertificateEnabled=no
httpclient.default.CertificateFile=/etc/httpclient.crt
httpclient.default.CertificateFormat=PEM
httpclient.default.CertificateKey=/etc/httpclient.key
```

11.5.4 Httpclient: Activator configuration package options example

```
root@VA_router:~# uci export httpclient
package httpclient

config core 'default'
	option Enabled 'yes'
	listFileServer '1.1.1.1:80'
	listFileServer '1.1.1.2:80'
	listSecureFileServer '1.1.1.1:443'
	listSecureFileServer '1.1.1.2:443'
	optionActivatorDownloadPath '/Activator/Sessionless/Httpserver.asp'
	optionSecureDownload 'no'
	optionPresentCertificateEnabled 'no'
	optionValidateServerCertificateEnabled 'no'
	optionCertificateFile '/etc/httpclient.crt'
	optionCertificateFormat 'PEM'
	optionCertificateKey '/etc/httpclient.key'
```

11.6 User management using UCI

User management is not currently available using the web interface. You can configure the feature using UCI or Activator.

11.6.1 User management packages

Package	Sections
management_users	users

11.6.2 Configuring user management

You can create different users on the system by defining them in the user management configuration file. This gives users access to different services.

Web Field / UCI / Package Option	Description
General settings	
Web: n/a UCI: management_users.@user[x].enabled Opt: enable	Enables/creates the user. 0 — Disabled. 1 — Enabled.
Web: n/a UCI: management_users.@user[x].username Opt: username	Specifies the user's username.
Web: n/a UCI: management_users.@user[x].password Opt: password	Specifies the user's password. When entering the user password enter in plain text using the password option. After reboot the password is displayed encrypted via the CLI using the hashpassword option. UCI: management_users.@user[x].hashpassword Opt: hashpassword. **Note:** a SRP user password will be displayed using the srphash option
Web: n/a UCI: management_users.@user[x].webuser Opt: webuser	Specifies web access permissions for the user. **Note:** webuser will only work if linuxuser is set to Enabled. 0 — Disabled. 1 — Enabled.
Web: n/a UCI: management_users.@user[x].chapuser Opt: chapuser	Specifies CHAP access permissions for the PPP connection. **Note:** chapuser will only work if linux user is set to Enabled. 0 — Disabled. 1 — Enabled.
Web: n/a UCI: management_users.@user[x].papuser Opt: papuser	Specifies PAP access permissions for the PPP connection. 0 — Disabled. 1 — Enabled.
Web: n/a UCI: management_users.@user[x].srpuser Opt: srpuser	Specifies SRP access permissions for the PPP connection. 0 — Disabled. 1 — Enabled.
Web: n/a UCI: management_users.@user[x].smsuser Opt: smsuser	Specifies SMS access permissions for the user. 0 — Disabled. 1 — Enabled.
Web: n/a UCI: linuxuser Opt: linuxuser	Specifies linuxuser access permissions for the user. 0 — Disabled. 1 — Enabled.
Web: n/a UCI: List allowed_pages Opt: list allowed_pages	Specifies which pages the user can view. Multiple pages should be entered using a space to separate if using UCI.

Table 18: Information table for config user commands

Note:

- webuser will only work if linuxuser is set to '**yes**'
- chapuser will only work if linuxuser is set to '**no**'
- when a new user is created on the system and given web access, you will no longer be able to login to the router web interface with the default root user details. The user must use their new user login details.

11.6.3 Configuring the management user password using UCI

The user password is displayed encrypted via the CLI using the hashpassword option.

```
root@VA_router:~# uci show management_users
management_users.@user[0].username=test
management_users.@user[0].hashpassword=$1$XVzDHHPQ$SKK4geFonctihuffMjS4U0
```

If changing the password via the UCI, enter the new password in plain text using the password option.

```
root@VA_router:~# uci set management_users.@user[0].username=newpassword
root@VA_router:~# uci commit
```

The new password will take effect after reboot and will now be displayed in encrypted format through the hashpassword option.

11.6.4 Configuring the management user password using package options

The root password is displayed encrypted via the CLI using the hashpassword option.

```
root@VA_router:~# uci export management_users
package management_users

config user
      option hashpassword '$1$wRYYiJOz$EeHN.GQcxXhRgNPVbqxVw
```

If changing the password using the UCI, enter the new password in plain text using the password option.

```
package management_users

config user
      option hashpassword '$1$wRYYiJOz$EeHN.GQcxXhRgNPVbqxVw
      option password ‘newpassword’
```

The new password will take effect after reboot and will now be displayed in encrypted format via the hashpassword option.

User management using UCI

```
root@VA_router:~# uci show management_users
management_users.@user[0]=user
management_users.@user[0].enabled=1
management_users.@user[0].username=test
management_users.@user[0].hashpassword=$1$XVzDHHPQ$SKK4geFonctihuffMjS4U0
management_users.@user[0].webuser=1
management_users.@user[0].linuxuser=1
management_users.@user[0].papuser=0
management_users.@user[0].chapuser=0
management_users.@user[0].srpuser=0
management_users.@user[0].smsuser=0
```

11.6.5 User management using package options

```
root@VA_router:~# uci export management_users

package management_users
config user
      option enabled ‘1’
      option username ‘test’
      option hashpassword ‘$1$XVzDHHPQ$SKK4geFonctihuffMjS4U0’
      option webuser ‘1’
      option linuxuser ‘1’
      option papuser ‘0’
      option chapuser ‘0’
```

```
option srpuser ‘0’
options smsuser ‘0’
```

11.6.6 Configuring user access to specific web pages

To specify particular pages a user can view, add the list allowed_pages. Examples are:

```
listallowed_pages '/admin/status'
```

The user can view admin status page only.

```
listallowed_pages 'admin/system/flashops'
```

The user can view flash operation page only.

To specify monitor widgets only, enter:

```
listallowed_pages 'monitor/<widgetname>'
```

Example widget names are: dhcp, arp, 3gstats, interfaces, memory, multiwan, network, openvpn, routes, system, ipsec, dmvpn, tservd.

12 Configuring an Ethernet interface on a GW1000

The GW1000 Series router has two physical Ethernet ports which can be configured in two ways:

- both ports bridged together using the same subnet
- each port operating as a separate network on its own subnet

The default configuration has both ports bridged together in the same subnet.

This section describes how to configure an Ethernet interface on a GW1000 router, including configuring the interface as a DHCP server, adding the interface to a firewall zone and mapping the physical switch ports.

12.1 Configuration packages used

Package	Sections
network	interface
	route
	alias
firewall	zone
dhcp	dhcp

12.2 Configuring an Ethernet interface using the web

To create and edit interfaces via the web interface, in the top menu, click **Network -> Interfaces**. The Interfaces overview page appears.

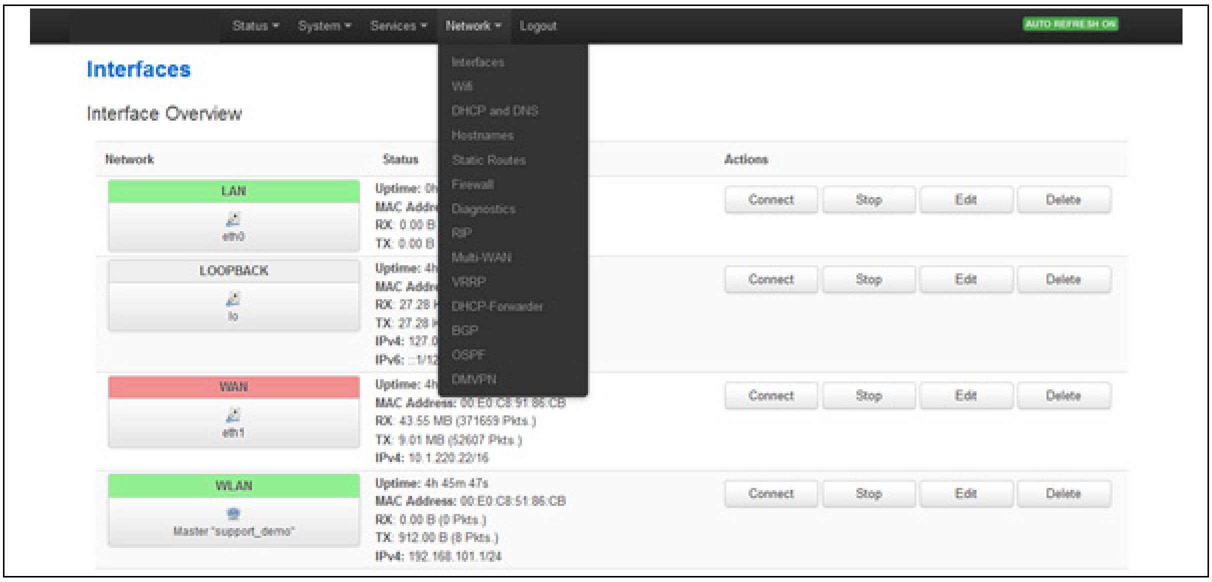


Figure 24: The interfaces overview page

There are two sections in the Interfaces page.

Section	Description
Interface Overview	Shows existing interfaces and their status. You can create new, and edit existing interfaces here.
ATM Bridges	ATM bridges expose encapsulated Ethernet in AAL5 connections as virtual Linux network interfaces, which can be used in conjunction with DHCP or PPP to dial into the provider network.

12.2.1 Interface overview: editing an existing interface

To edit an existing interface, from the interface tabs at the top of the page, select the interface you wish to configure. Alternatively, click **Edit** in the interface's row.

12.2.2 Interface overview: creating a new interface

To create a new interface, in the Interface Overview section, click **Add new interface**. The Create Interface page appears.

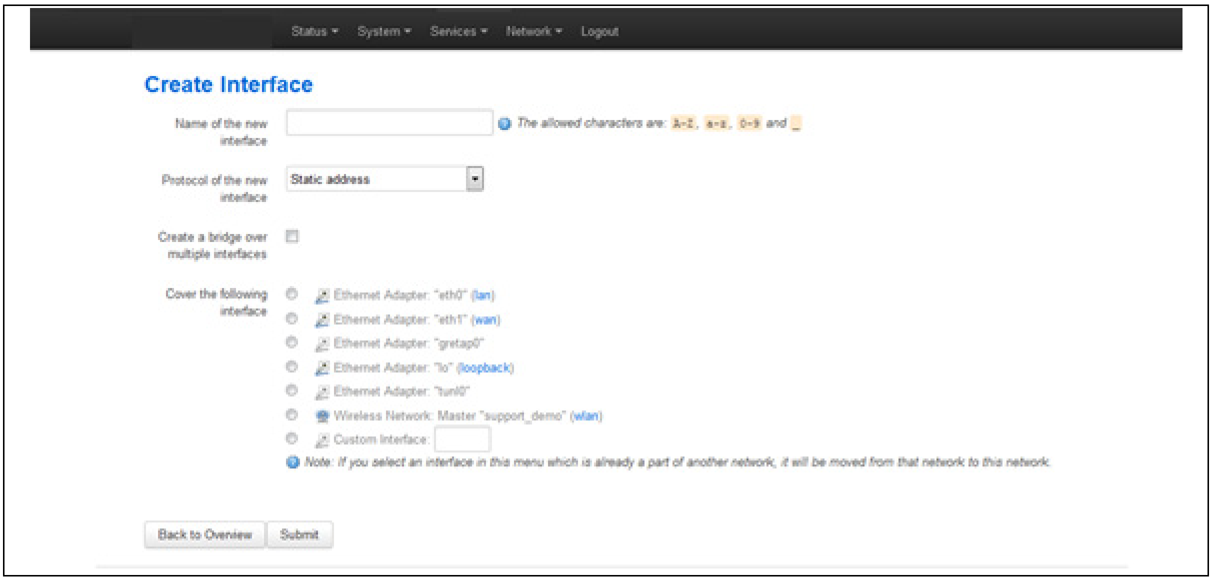


Figure 25: The create interface page

Web Field / UCI / Package Option	Description
Web: Name of the new interface UCI: network.<if name> Opt: config interface	Assigns a logical name to the interface. The network interface section will assign this name (<if name>). Type the name of the new interface. Allowed characters are A-Z, a-z, 0-9 and _

<table>
<tr><td>Web: Protocol of the new interface
UCI: network.<if name>.proto
Opt: proto</td><td>Specifies what protocol the interface will operate on. Select Static.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>IPv6-in-IPv4 (RFC4213)</td><td>Used with tunnel brokers.</td></tr>
<tr><td>IPv6-over-IPv4</td><td>Stateless IPv6 over IPv4 transport.</td></tr>
<tr><td>GRE</td><td>Generic Routing Encapsulation protocol</td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol</td></tr>
<tr><td>PPP</td><td>Point to Point Protocol</td></tr>
<tr><td>PPPoE</td><td>PPP over Ethernet</td></tr>
<tr><td>PPPoATM</td><td>PPP over ATM</td></tr>
<tr><td>LTE/UMTS/GPRS/EV-DO</td><td>CDMA, UMTS or GPRS connection using an AT-style 3G modem.</td></tr>
</table></td></tr>
<tr><td>Web: Create a bridge over multiple interfaces
UCI: network.<if name>.type
Opt: type</td><td>If you select this option, then the new logical interface created will act as a bridging interface between the chosen existing physical interfaces.
<table>
<tr><td>Empty</td><td></td></tr>
<tr><td>Bridge</td><td>Configures a bridge over multiple interfaces.</td></tr>
</table></td></tr>
<tr><td>Web: Cover the following interface
UCI: network.<if name>.ifname
Opt: ifname</td><td>Physical interface name to assign to this logical interface. If creating a bridge over multiple interfaces select two interfaces to bridge. When using uci the interface names should be separated by a space e.g. option ifname ‘eth2 eth3’</td></tr>
</table>

Table 19: Information table for the create new interface page

Click **Submit**. The Interface configuration page appears. There are three sections:

Section	Description
Common Configuration	Configure the interface settings such as protocol, IP address, gateway, netmask, custom DNS servers, MTU and firewall configuration.
IP-Aliases	Assigning multiple IP addresses to the interface
DHCP Server	Configuring DHCP server settings for this interface

12.2.3 Interface overview: common configuration

The common configuration section has four sub sections:

Section	Description
General Setup	Configure the basic interface settings such as protocol, IP address, gateway, netmask, custom DNS servers.

Advanced Settings	'Bring up on boot', 'Monitor interface state', Override MAC address, Override MTU and 'Use gateway metric'
Physical Settings	Bridge interfaces, VLAN PCP to SKB priority mapping,
Firewall settings	Assign a firewall zone to the interface

12.2.3.1 Common configuration – general setup

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Status</td><td>Shows the current status of the interface.</td></tr>
<tr><td>Web: Protocol
UCI: network.<if name>.proto
Opt: proto</td><td>Protocol type. The interface protocol may be one of the options shown below. The protocol selected in the previous step will be displayed as default but can be changed if required.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>IPv6-in-IPv4 (RFC4213)</td><td>Used with tunnel brokers.</td></tr>
<tr><td>IPv6-over-IPv4</td><td>Stateless IPv6 over IPv4 transport.</td></tr>
<tr><td>GRE</td><td>Generic Routing Encapsulation protocol</td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol.</td></tr>
<tr><td>PPP</td><td>Point-to-Point protocol</td></tr>
<tr><td>PPPoE</td><td>PPP over Ethernet</td></tr>
<tr><td>PPPoATM</td><td>PPP over ATM</td></tr>
<tr><td>LTE/UMTS/GPRS/EV-DO</td><td>CDMA, UMTS or GPRS connection using an AT-style 3G modem.</td></tr>
</table></td></tr>
<tr><td>Web: IPv4 address
UCI: network.<if name>.ipaddr
Opt: ipaddr</td><td>The IPv4 address of the interface. This is optional if an IPv6 address is provided.</td></tr>
<tr><td>Web: IPv4 netmask
UCI: network.<if name>.netmask
Opt: netmask</td><td>Subnet mask to be applied to the IP address of this interface.</td></tr>
<tr><td>Web: IPv4 gateway
UCI: network.<if name>.gateway
Opt: gateway</td><td>IPv4 default gateway to assign to this interface (optional).</td></tr>
<tr><td>Web: IPv4 broadcast
UCI: network.<if name>.broadcast
Opt: broadcast</td><td>Broadcast address. This is automatically generated if no broadcast address is specified.</td></tr>
<tr><td>Web: Use custom DNS servers
UCI: network.<if name>.dns
Opt: dns</td><td>List of DNS server IP addresses (optional). Multiple DNS Servers are separated by a space when using UCI or CLI.</td></tr>
</table>

Web: Accept router advertisements UCI: network.<if name>.accept_ra Opt: accept_ra	Specifies whether to accept IPv6 Router Advertisements on this interface (optional). **Note:** default is 1 if protocol is set to DHCP, otherwise defaults to 0. 0 – Disabled. 1 – Enabled.
Web: Send router solicitations UCI: network.<if name>.send_rs Opt: send_rs	Specifies whether to send Router Solicitations on this interface (optional). **Note:** defaults to 1 for Static protocol, otherwise defaults to 0. 0 – Disabled. 1 – Enabled.
Web: IPv6 address UCI: network.<if name>.ip6addr Opt: ip6addr	The IPv6 IP address of the interface. Optional if an IPv4 address is provided. CIDR notation for the IPv6 address is required.
Web: IPv6 gateway UCI: network.<if name>.ip6gw Opt: ip6gw	Assign given IPv6 default gateway to this interface (optional).

Table 20: Information table for LAN interface common configuration settings

12.2.4 Common configuration: advanced settings

Figure 26: The Ethernet connection advanced settings page

Web Field / UCI / Package Option	Description
Web: Bring up on boot UCI: network.<if name>.auto Opt: auto	Enables the interface to connect automatically on boot up. 0 – Disabled. 1 – Enabled.
Web: Monitor interface state UCI: network.<if name>.monitored Opt: monitored	Enabled if status of interface is presented on Monitoring platform. 0 – Disabled. 1 – Enabled.
Web: Override MAC address UCI: network.<if name>.macaddr Opt: macaddr	Override the MAC address assigned to this interface. Must be in the form: hh:hh:hh:hh:hh:hh, where h is a hexadecimal number.

Web: Override MTU UCI: network.<if name>.mtu Opt: mtu	Defines the value to override the default MTU on this interface. 1500 — 1500 bytes Range
Web: Use gateway metric UCI: network.<if name>.metric Opt: metric	Specifies the default route metric to use for this interface (optional). 0 Range

Table 21: Information table for common configuration advanced settings

12.2.4.1 Common configuration: physical settings

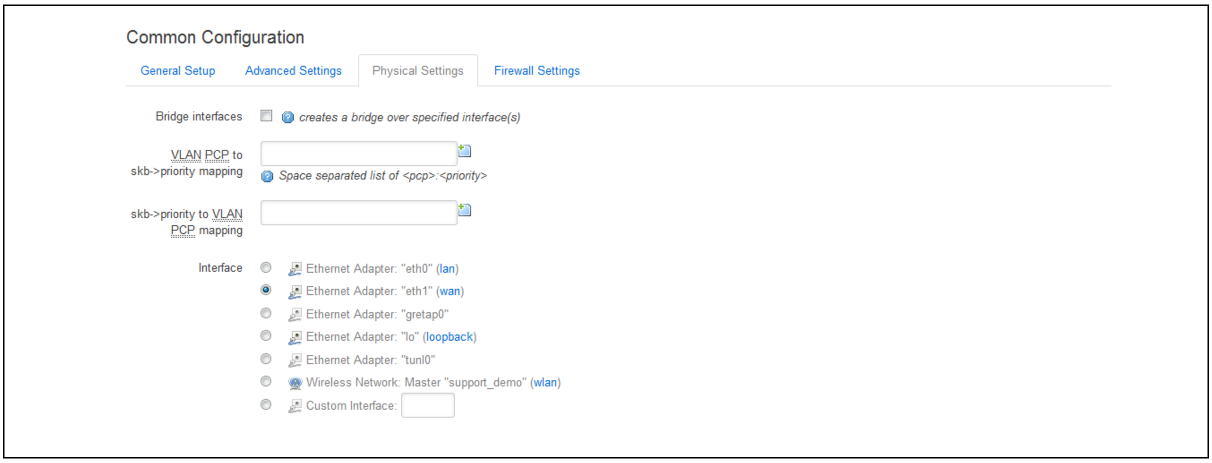


Figure 27: The common configuration physical settings page

Web Field / UCI / Package Option	Description
Web: Bridge interfaces UCI: network.<if name>.type Opt: type	Sets the interface to bridge over a specified interface(s). The physical interfaces can be selected from the list and are defined in network.<if name>.ifname. Blank Bridge — Configures a bridge over multiple interfaces.
Web: Enable STP UCI: network.<if name>.stp Opt: stp	Enable Spanning Tree Protocol. This option is only available when the Bridge Interfaces option is selected. 0 — Disabled. 1 — Enabled.
Web: VLAN PCP to skb>priority mapping UCI: network.<if name>.vlan_qos_map_ingress Opt: list vlan_qos_map_ingress	VLAN priority code point to socket buffer mapping. Multiple priority mappings are entered with a space between them when using UCI. Example: network.<if name>. vlan_qos_map_ingress =1:2 2:1
Web: skb priority to >VLAN PCP mapping UCI: network.<if name>.vlan_qos_map_egress Opt: list vlan_qos_map_egress	Socket buffer to VLAN priority code point mapping. Multiple priority mappings are entered with a space between them when using UCI. Example: network.<if name>. vlan_qos_map_egress =1:2 2:1

Web: Interface UCI: network.<if name>.ifname Opt: ifname	Physical interface to assign the logical interface to. If mapping multiple interfaces for bridging the interface names are separated by a space when using UCI and package options. Example: option ifname ‘eth2 eth3’ or network.<if name>.ifname=eth2 eth 3

Table 22: Information table for physical settings page

12.2.4.2 Common configuration: firewall settings

Use this section to select the firewall zone you want to assign to this interface.

Select **unspecified** to remove the interface from the associated zone or fill out the create field to define a new zone and attach the interface to it.

Figure 28: GRE firewall settings

12.2.5 Interface overview: IP-aliases

IP aliasing is associating more than one IP address to a network interface. You can assign multiple aliases.

12.2.5.1 IP-alias packages

Package	Sections
Network	alias

12.2.5.2 IP-alias using the web

To use IP-Aliases, enter a name for the alias and click **Add**. This name will be assigned to the alias section for this IP-alias. In this example the name ethalias1 is used.

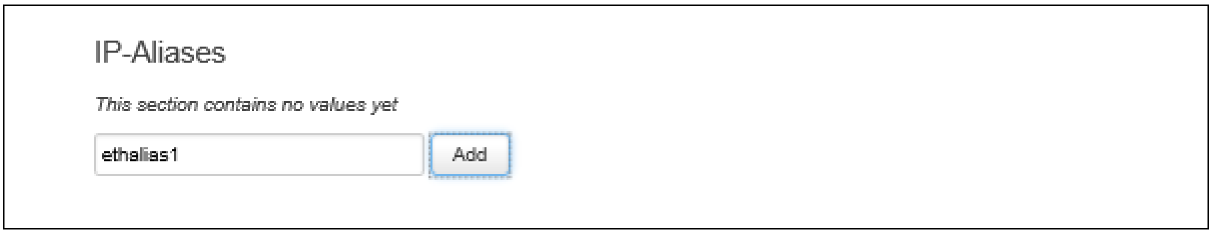


Figure 29: The IP-Aliases section

Web Field / UCI / Package Option	Description
UCI: network.<alias name>=alias Opt: config alias 'aliasname'	Assigns the alias name.
UCI: network.<alias name>.interface Opt: interface	This maps the IP-Alias to the interface.
UCI: network.<alias name>.proto Opt: proto	This maps the interface protocol to the alias.

Table 23: Information table for IP-Aliases name assignment

The IP Aliases configuration options page appears. The IP-Alias is divided into two sub sections – general setup and advanced.

12.2.5.3 IP-aliases: general setup

Figure 30: The IP-aliases general setup section

Web Field / UCI / Package Option	Description
Web: IPv4-Address UCI: network.<alias name>.ipaddr Opt: ipaddr	Defines the IP address for the IP alias.
Web: IPv4-Netmask UCI: network.<alias name>.netmask Opt: netmask	Defines the netmask for the IP alias.
Web: IPv4-Gateway UCI: network.<alias name>.gateway Opt: gateway	Defines the gateway for the IP alias.

Table 24: Information table for IP-Alias general setup page

12.2.5.4 IP-aliases: advanced settings

Figure 31: The IP-Aliases advanced settings section

Web Field / UCI / Package Option	Description
Web: IPv4-Broadcast UCI: network.<alias name>.bcast Opt: bcast	Defines the IP broadcast address for the IP alias.
Web: DNS-Server UCI: network.<alias name>.dns Opt: dns	Defines the DNS server for the IP alias.

Table 25: Information table for IP-Alias advanced settings page

12.2.6 Interface overview: DHCP server

12.2.6.1 DHCP server: packages

Package	Sections
dhcp	dhcp

To assign a DHCP Server to the interface, uncheck the Ignore Interface box.

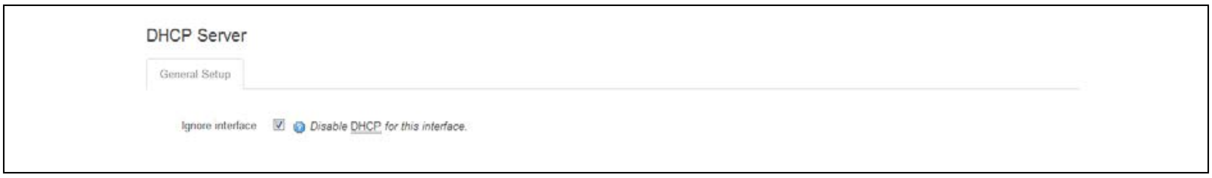


Figure 32: The DHCP Server settings section

The DHCP Server configuration options will appear. The DHCP Server is divided into two sub sections – general setup and advanced.

12.2.6.2 DHCP server: general setup

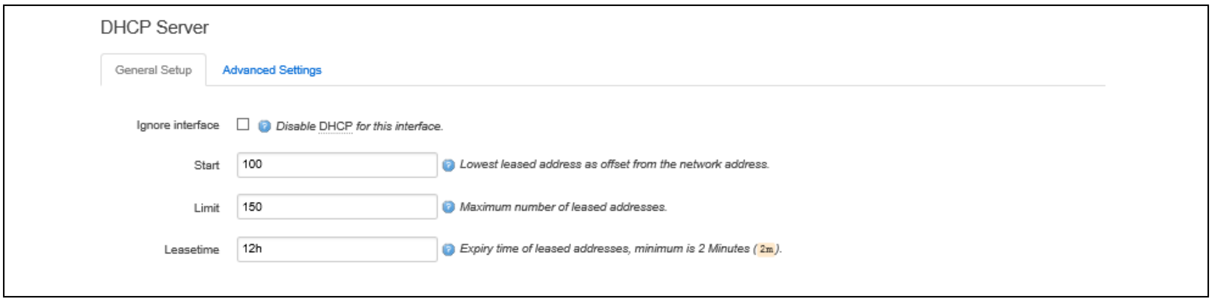


Figure 33: The DHCP server general setup section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Ignore interface
UCI: dhcp.@dhcp[x].ignore
Opt: ignore</td><td>Defines whether the DHCP pool should be enabled for this interface. If not specified for the DHCP pool then default is disabled i.e. dhcp pool enabled.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
<tr><td>Web: n/a
UCI: dhcp.@dhcp[x].start
Opt: start</td><td>Defines the offset from the network address for the start of the DHCP pool. It may be greater than 255 to span subnets.<table><tr><td>100</td><td></td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
<tr><td>Web: n/a
UCI: dhcp.@dhcp[x].limit
Opt: limit</td><td>Defines the offset from the network address for the end of the DHCP pool.<table><tr><td>150</td><td></td></tr><tr><td>Range</td><td>0 – 255</td></tr></table></td></tr>
<tr><td>Web: n/a
UCI: dhcp.@dhcp[x].leasetime
Opt: leasetime</td><td>Defines the lease time of addresses handed out to clients, for example 12h or 30m.<table><tr><td>12h</td><td>12 hours</td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
</table>

Table 26: Information table for DHCP server general setup page

12.2.6.3 DHCP Server: advanced settings

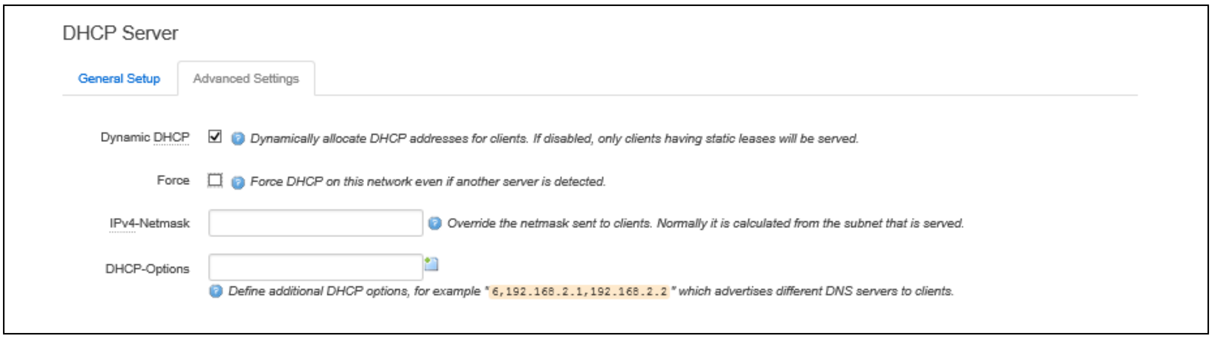


Figure 34: The DHCP server advanced settings section

Web Field/UCI/Package Option	Description
Web: Dynamic DHCP UCI: dhcp.@dhcp[x].dynamicdhcp Opt: dynamicdhcp	Defines whether to allocate DHCP leases. 1: Dynamically allocate leases. 0: Use /etc/ethers file for serving DHCP leases.
Web: Force UCI: dhcp.@dhcp[x].force Opt: force	Forces DHCP serving on the specified interface even if another DHCP server is detected on the same network segment. 0: Disabled. 1: Enabled.
Web: DHCP-Options UCI: dhcp.@dhcp[x].dhcp_option Opt: list dhcp_option	Defines additional options to be added for this dhcp pool. For example with 'list dhcp_option 26,1470' or 'list dhcp_option mtu, 1470' you can assign a specific MTU per DHCP pool. Your client must accept the MTU option for this to work. Options that contain multiple vales should be separated by a space. Example: list dhcp_option 6,192.168.2.1 192.168.2.2 (blank): No options defined. Syntax: Option_number, option_value
Web: n/a UCI: dhcp.@dhcp[x].networkid Opt: networked	Assigns a network-id to all clients that obtain an IP address from this pool.

Table 27: Information table for DHCP advanced settings page

For more advanced configuration on the DHCP server, read 'DHCP server and DNS configuration section.

12.2.7 Interface configuration using UCI

The configuration files are stored at **/etc/config/network**, **/etc/config/firewall** and **/etc/config/dhcp**

```
root@VA_router:~# uci show network
      …..
network.newinterface=interface
network.newinterface.proto=static
network.newinterface.ifname=eth0
network.newinterface.monitored=0
network.newinterface.ipaddr=2.2.2.2
network.newinterface.netmask=255.255.255.0
network.newinterface.gateway=2.2.2.10
network.newinterface.broadcast=2.2.2.255
network.newinterface.vlan_qos_map_ingress=1:2 2:1
```

```
network.ethalias1=alias
network.ethalias1.proto=static
network.ethalias1.interface=newinterface
network.ethalias1.ipaddr=10.10.10.1
network.ethalias1.netmask=255.255.255.0
network.ethalias1.gateway=10.10.10.10
network.ethalias1.bcast=10.10.10.255
network.ethalias1.dns=8.8.8.8
      …..
firewall.@zone[0]=zone
firewall.@zone[0].name=lan
firewall.@zone[0].input=ACCEPT
firewall.@zone[0].output=ACCEPT
firewall.@zone[0].forward=ACCEPT
firewall.@zone[0].network=lan newinterface

root@VA_router:~# uci show dhcp
      …
dhcp.@dhcp[0]=dhcp
dhcp.@dhcp[0].start=100
root@VA_router:~# uci show firewall
dhcp.@dhcp[0].leasetime=12h
dhcp.@dhcp[0].limit=150
dhcp.@dhcp[0].interface=newinterface
```

To change any of the above values use uci set command.

12.2.7.1 Interface common configuration using package options

The configuration files are stored at **/etc/config/network, /etc/config/firewall** and **/etc/config/dhcp**

```
root@VA_router:~# uci export network
package network
      config interface 'newinterface'
              option proto 'static'
              option ifname 'eth0'
```

```
                option monitored '0'
                option ipaddr '2.2.2.2'
                option netmask '255.255.255.0'
                option gateway '2.2.2.10'
                option broadcast '2.2.2.255'
                list vlan_qos_map_ingress '1:2'
                list vlan_qos_map_ingress '2:1'
        config alias 'ethalias1'
                option proto 'static'
                option interface 'newinterface'
                option ipaddr '10.10.10.1'
                option netmask '255.255.255.0'
                option gateway '10.10.10.10'
                option bcast '10.10.10.255'
                option dns '8.8.8.8'
root@VA_router:~# uci export firewall
package firewall
config zone
        option name 'lan'
        option input 'ACCEPT'
        option output 'ACCEPT'
        option network 'lan newinterface'

root@VA_router:~# uci export dhcp
package dhcp
        ……
config dhcp
        option start '100'
        option leasetime '12h'
        option limit '150'
        option interface 'newinterface'
```

To change any of the above values use uci set command.

12.2.8 ATM bridges

The ATM bridges section is not used when configuring an Ethernet interface.

12.3 Interface diagnostics

12.3.1 Interfaces status

To show the current running interfaces, enter:

```
root@VA_router:~# ifconfig
3g-CDMA   Link encap:Point-to-Point Protocol
          inet addr:10.33.152.100  P-t-P:178.72.0.237  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1400  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:23 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:428 (428.0 B)  TX bytes:2986 (2.9 KiB)
eth0      Link encap:Ethernet  HWaddr 00:E0:C8:12:12:15
          inet addr:192.168.100.1  Bcast:192.168.100.255
Mask:255.255.255.0
          inet6 addr: fe80::2e0:c8ff:fe12:1215/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6645 errors:0 dropped:0 overruns:0 frame:0
          TX packets:523 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:569453 (556.1 KiB)  TX bytes:77306 (75.4 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:385585 errors:0 dropped:0 overruns:0 frame:0
          TX packets:385585 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:43205140 (41.2 MiB)  TX bytes:43205140 (41.2 MiB)
```

To display a specific interface enter: ifconfig <if name>:

```
root@VA_router:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:E0:C8:12:12:15
          inet addr:192.168.100.1  Bcast:192.168.100.255
Mask:255.255.255.0
          inet6 addr: fe80::2e0:c8ff:fe12:1215/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:7710 errors:0 dropped:0 overruns:0 frame:0
          TX packets:535 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:647933 (632.7 KiB)  TX bytes:80978 (79.0 KiB)
```

12.3.2 Route status

To show the current routing status, enter:

```
root@VA_router:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
192.168.100.0   *               255.255.255.0   U     0      0        0
eth0
```

Note: a route will only be displayed in the routing table when the interface is up.

13 DHCP server and DNS configuration (Dnsmasq)

Dynamic Host Configuration Protocol (DHCP) server is responsible for assigning IP addresses to hosts. IP addresses can be given out on different interfaces and different subnets. You can manually configure lease time as well as setting static IP to host mappings.

Domain Name Server (DNS) is responsible for resolution of IP addresses to domain names on the internet.

Dnsmasq is the application which controls DHCP and DNS services. Dnsmasq has two sections; one to specify general DHCP and DNS settings and one or more DHCP pools to define DHCP operation on the desired network interface.

13.1 Configuration package used

Package	Sections
dhcp	dnsmasq
	dhcp
	host

13.2 Configuring DHCP and DNS using the web interface

In the top menu, select **Network -> DHCP and DNS**. The DHCP and DNS page appears. There are three sections: Server Settings, Active Leases, and Static Leases.

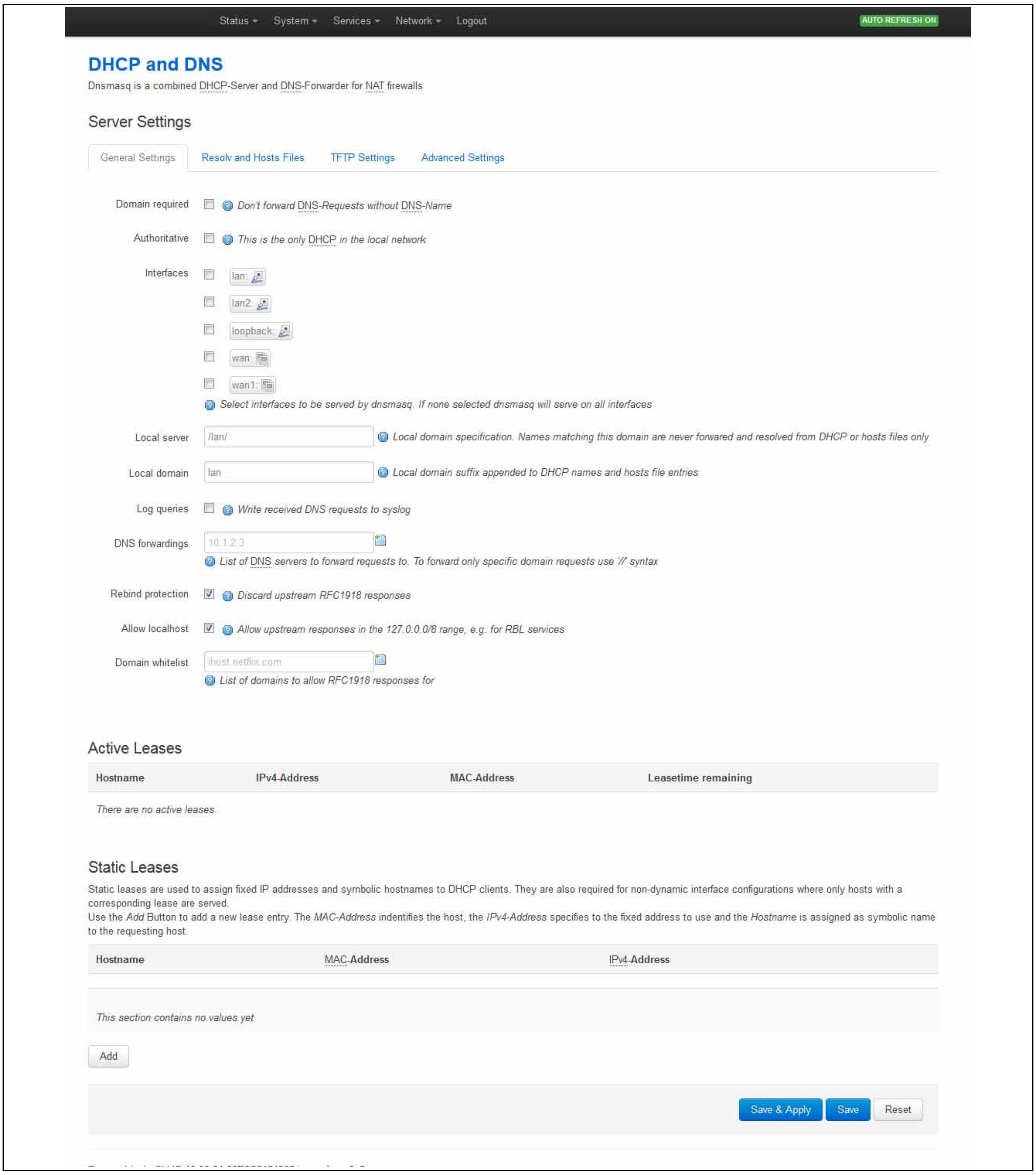


Figure 35: The DHCP and DNS page

13.2.1 Dnsmasq: general settings

Web Field / UCI / Package Option	Description	
Web: Domain required UCI: dhcp.@dnsmasq[0].domainneeded Opt: domainneeded	Defines whether to forward DNS requests without a DNS name. Dnsmasq will never forward queries for plain names, without dots or domain parts, to upstream nameservers. If the name is not known from /etc/hosts or DHCP then a "not found" answer is returned.	
	1	Enabled.
	0	Disabled.
Web: Authoritative UCI: dhcp.@dnsmasq[0]. authoritative Opt: authoritative	Forces authoritative mode, this speeds up DHCP leasing. Used if this is the only server in the network.	
	1	Enabled.
	0	Disabled.
Web: Interfaces UCI: dhcp.@dnsmasq[0].interface Opt: list interface	Defines the list of interfaces to be served by dnsmasq. If you do not select a specific interface, dnsmasq will serve on all interfaces. Configured interfaces are shown via the web GUI.	
	Lan	Serve only on LAN interface
	Range	
Web: Local Server UCI: dhcp.@dnsmasq[0].local Opt: local	Specifies the local domain. Names matching this domain are never forwarded and are resolved from DHCP or host files only.	
	/lan/	
	Range	
Web: Local Domain UCI: dhcp.@dnsmasq[0].domain Opt: domain	Specifies local domain suffix appended to DHCP names and hosts file entries.	
	lan	
	Range	
Web: Log Queries UCI: dhcp.@dnsmasq[0].logqueries Opt: logqueries	Writes received DNS requests to syslog.	
	0	Disabled.
	1	Enabled.
Web: DNS Forwardings UCI: dhcp.@dnsmasq[0].server Opt: list server	List of DNS server to forward requests to. To forward specific domain requests only, use **/ /** syntax. When using UCI, enter multiple servers with a space between them.	
		No DNS server configured.
	Range	
Web: Rebind Protection UCI: dhcp.@dnsmasq[0].rebind_protection Opt: rebind_protection	Enables DNS rebind attack protection by discarding upstream RFC1918 responses.	
	0	Disabled.
	1	Enabled.
Web: Allow Localhost UCI: dhcp.@dnsmasq[0].rebind_localhost Opt: rebind_localhost	Defines whether to allow upstream responses in the 127.0.0.0/8 range. This is required for DNS based blacklist services. Only takes effect if rebind protection is enabled.	
	0	Disabled.
	1	Enabled.

Web: Domain Whitelist UCI: dhcp.@dnsmasq[0].rebind_domain Opt: list rebind_domain	Defines the list of domains to allow RFC1918 responses to. Only takes effect if rebind protection is enabled. When using UCI multiple servers should be entered with a space between them. (blank): No list configured. Range: (blank)

Table 28: Information table for general server settings

13.2.2 Dnsmasq: resolv and host files

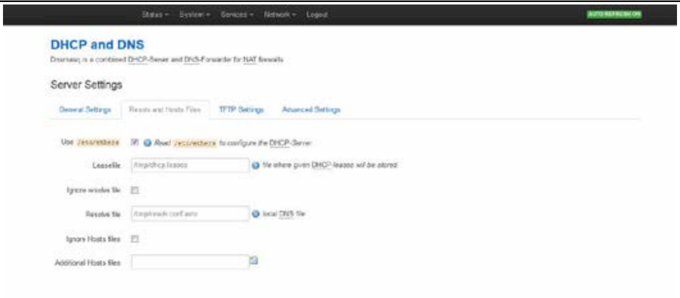


Figure 36: The resolv and host files section

Web Field / UCI / Package Option	Description
Web: Use /etc/ethers UCI: dhcp.@dnsmasq[0].readethers Opt: readethers	Defines whether static lease entries are read from /etc/ethers. 1: Enabled. 0: Disabled.
Web: Leasefile UCI: dhcp.@dnsmasq[0].leasefile Opt: leasefile	Defines the file where given DHCP leases will be stored. The DHCP lease file allows leases to be picked up again if dnsmasq is restarted. /tmp/dhcp.leases: Store DHCP leases in this file. Range: (blank)
Web: Ignore resolve file UCI: dhcp.@dnsmasq[0].noresolv Opt: noresolv	Defines whether to use the local DNS file for resolving DNS. 0: Use local DNS file. 1: Ignore local DNS file.
Web: Resolve file UCI: dhcp.@dnsmasq[0].resolvfile Opt: resolvfile	Defines the local DNS file. Default is /tmp/resolv.conf.auto
Web: Ignore Hosts files UCI: dhcp.@dnsmasq[0].nohosts Opt: nohosts	Defines whether to use local host's files for resolving DNS. 0: Use local hosts file. 1: Ignore local hosts file.
Web: Additional Hosts files UCI: dhcp.@dnsmasq[0].addnhosts Opt: list addnhosts	Defines local host's files. When using UCI multiple servers should be entered with a space between them.

Table 29: Information table for resolv and host files section

13.2.3 Dnsmasq: TFTP settings

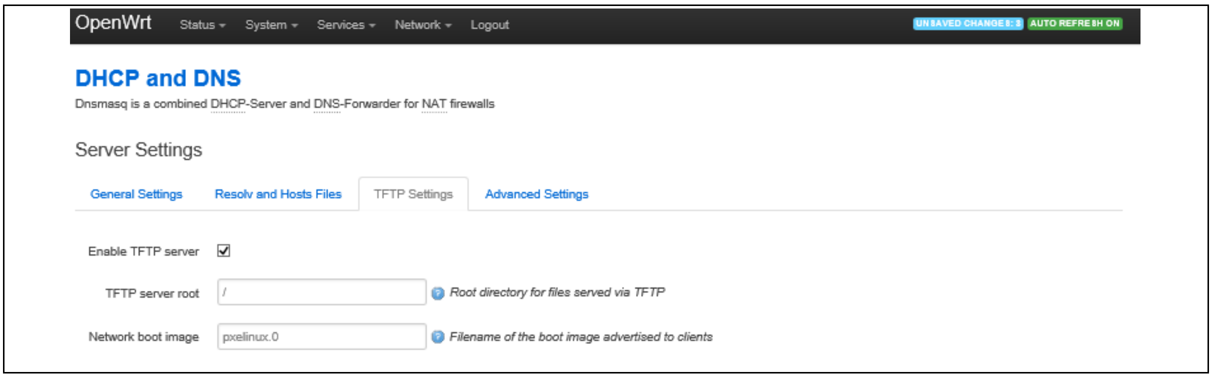


Figure 37: The TFTP settings section

Web Field / UCI / Package Option	Description
Web: Enable TFTP Server UCI: dhcp.@dnsmasq[0].enable_tftp Opt: enable_tftp	Enables the TFTP server. 0 Disabled. 1 Enabled.
Web: Enable TFTP Server UCI: dhcp.@dnsmasq[0].tftp_root Opt: tftp_root	Defines root directory for file served by TFTP.
Web: Enable TFTP Server UCI: dhcp.@dnsmasq[0].dhcp_boot Opt: dhcp_boot	Defines the filename of the boot image advertised to clients. This specifies BOOTP options, in most cases just the file name.

Table 30: Information table for TFTP settings

13.2.4 Dnsmasq: advanced settings

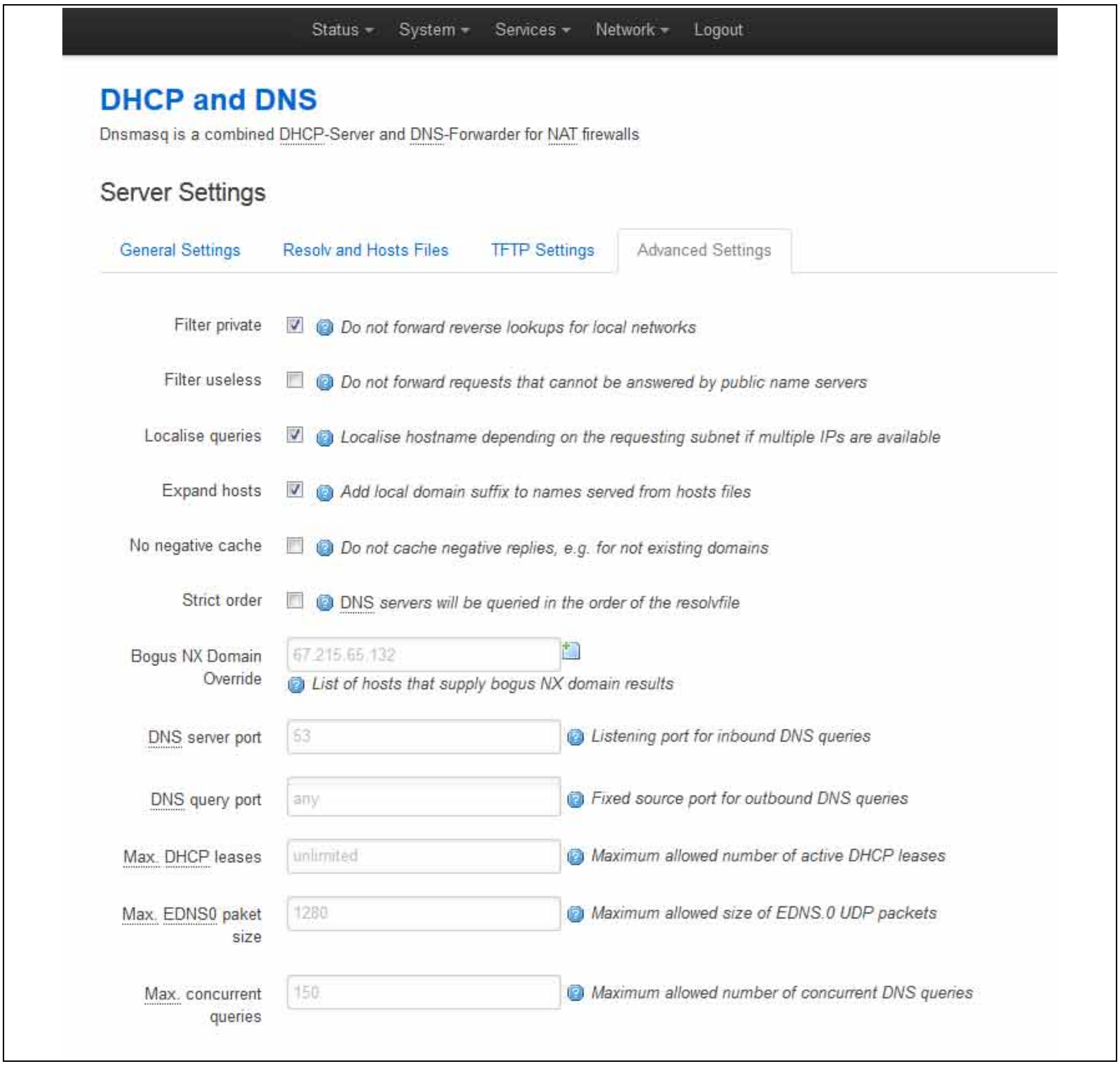


Figure 38: The advanced settings page

Web Field / UCI / Package Option	Description
Web: Filter private UCI: dhcp.@dnsmasq[0]. Opt: boguspriv	Enables disallow option for forwarding reverse lookups for local networks. This rejects reverse lookups to private IP ranges where no corresponding entry exists in /etc/hosts. 1 — Enabled. 0 — Disabled.
Web: Filter useless UCI: dhcp.@dnsmasq[0].filterwin2k Opt: filterwin2k	Enables disallow option for forwarding requests that cannot be answered by public name servers. Normally enabled for dial on demand interfaces. 1 — Enabled. 0 — Disabled.

Web: Localise queries UCI: dhcp.@dnsmasq[0].localise_queries Opt: localise_queries	Defines whether to uses IP address to match the incoming interface if multiple addresses are assigned to a host name in /etc/hosts. 1: Enabled. 0: Disabled.
Web: Expand hosts UCI: dhcp.@dnsmasq[0].expandhosts Opt: expandhosts	Adds a local domain suffix to names served from host files. 1: Enabled. 0: Disabled.
Web: No negative cache UCI: dhcp.@dnsmasq[0].nonegcache Opt: nonegcache	Enable this to stop caching of negative replies. For example, non-existing domains. 1: Enabled. 0: Disabled.
Web: Strict order UCI: dhcp.@dnsmasq[0].strictorder Opt: strictorder	Enable this to query DNS servers in the order of the resolve file. 1: Enabled. 0: Disabled.
Web: Bogus NX Domain override UCI: dhcp.@dnsmasq[0].bogusnxdomain Opt: list bogusnxdomain	A list of hosts that supply bogus NX domain results. When using UCI multiple servers should be entered with a space between them. Empty list Range
Web: DNS server port UCI: dhcp.@dnsmasq[0].port Opt: port	Listening port for inbound DNS queries. 53: Set to 0 to disable DNS functionality. Range: 0 - 65535
Web: DNS query port UCI: dhcp.@dnsmasq[0].queryport Opt: queryport	Defines fixed source port for outbound DNS queries. any Range: any; 0 - 65535
Web: Max DHCP leases UCI: dhcp.@dnsmasq[0].dhcpleasemax Opt:dhcpleasemax	Defines the maximum allowed number of active DHCP leases. unlimited Range
Web: Max EDNS0 packet size UCI: dhcp.@dnsmasq[0].ednspacket_max Opt: ednspacket_max	Defines the maximum allowed size of EDNS.0 UDP packets in bytes. 1280: 1280 bytes Range
Web: Max concurrent queries UCI: dhcp.@dnsmasq[0].dnsforwardmax Opt: dnsforwardmax	Maximum allowed number of concurrent DNS queries. 150: 1280 bytes Range

Table 31: Information table for advanced settings

13.2.5 Active leases

This section displays all currently active leases.

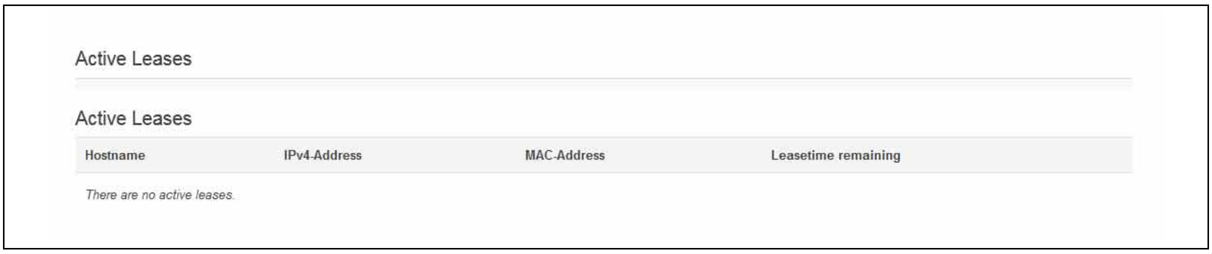


Figure 39: The active leases section

Web Field / UCI / Package Option	Description
Web: Hostname UCI: dhcp.@host[0].name Opt: name	Displays the hostname of the client.
Web: IPv4 Address UCI: dhcp.@host[0].ip Opt: ip	Displays the IP address of the client.
Web: MAC Address UCI: dhcp.@host[0].mac Opt: mac	Displays the MAC address of the client.
Web: Lease time remaining UCI: n/a Opt: n/a	Displays the remaining lease time.

Table 32: Information table for active leases section

13.2.6 Static leases

Use static leases to assign fixed IP addresses and symbolic hostnames to DHCP clients. Static leases are also required for non-dynamic interface configurations where only hosts with a corresponding lease are served. Click **Add** to add a new lease entry.

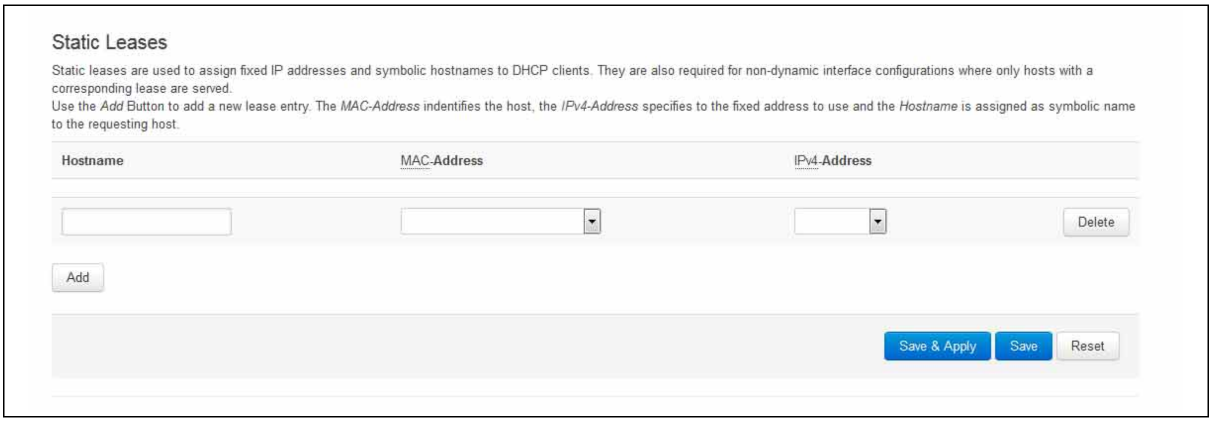


Figure 40: The static leases section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Hostname
UCI: dhcp.@host[0].name
Opt: name</td><td>Defines the optional symbolic name to assign to this static DHCP entry.<table><tr><td>1</td><td>Enabled.</td></tr><tr><td>0</td><td>Disabled.</td></tr></table></td></tr>
<tr><td>Web: MAC Address
UCI: dhcp.@host[0].mac
Opt: mac</td><td>Defines the hardware address that identifies the host.</td></tr>
<tr><td>Web: IPv4 Address
UCI: dhcp.@host[0].ip
Opt: ip</td><td>The IPv4 address specifies the fixed address to use for this host..</td></tr>
</table>

Table 33: Information table for static leases

13.3 Configuring DHCP and DNS using UCI

13.3.1 Common options section

Possible section types of the DHCP configuration file are defined below. Not all types may appear in the file and most of them are only needed for special configurations. Common configurations are Common Options, DHCP Pools and Static Leases.

The configuration section type dnsmasq determines values and options relevant to the overall operation of dnsmasq and the DHCP options on all interfaces served. The following table lists all available options, their default value, as well as the corresponding dnsmasq command line option.

These are the default settings for the common options:

```
root@VA_router:~# uci show dhcp
dhcp.@dnsmasq[0]=dnsmasq
dhcp.@dnsmasq[0].domainneeded=1
dhcp.@dnsmasq[0].boguspriv=1
dhcp.@dnsmasq[0].filterwin2k=0
dhcp.@dnsmasq[0].localise_queries=1
dhcp.@dnsmasq[0].logqueries=1
dhcp.@dnsmasq[0].rebind_protection=1
dhcp.@dnsmasq[0].rebind_localhost=1
dhcp.@dnsmasq[0].local=/lan/
dhcp.@dnsmasq[0].domain=lan
dhcp.@dnsmasq[0].expandhosts=1
```

```
dhcp.@dnsmasq[0].nonegcache=0
dhcp.@dnsmasq[0].authoritative=1
dhcp.@dnsmasq[0].readethers=1
dhcp.@dnsmasq[0].leasefile=/tmp/dhcp.leases
dhcp.@dnsmasq[0].noresolve=0
dhcp.@dnsmasq[0].resolvfile=/tmp/resolv.conf.auto
dhcp.@dnsmasq[0].nohosts=0
dhcp.@dnsmasq[0].addnhosts=hostfile1 hostfile2
dhcp.@dnsmasq[0].interface=lan
dhcp.@dnsmasq[0].server=1.1.1.1 2.2.2.2
dhcp.@dnsmasq[0].rebind domain=tes.domain
dhcp.@dnsmasq[0].enable_tftp=0
dhcp.@dnsmasq[0].tftp_root=/tmp/tftp
dhcp.@dnsmasq[0].dhcp_boot=boot.image
dhcp.@dnsmasq[0].nonegcache=0
dhcp.@dnsmasq[0].strictorder=0
dhcp.@dnsmasq[0].bogusnxdomain=1.1.1.1  2.2.2.2
dhcp.@dnsmasq[0].port=53
dhcp.@dnsmasq[0].dhcpleasemax=150
dhcp.@dnsmasq[0].ednspacket_max=1280
dhcp.@dnsmasq[0].dnsforwardmax=150
root@VA_router:~# uci show dhcp
config 'dnsmasq'
      option domainneeded '1'
        option rebind_protection '1'
        option rebind_localhost '1'
        option local '/lan/'
        option domain 'lan'
        option authoritative '1'
        option readethers '1'
        option leasefile '/tmp/dhcp.leases'
        list interface 'lan'
        list server '1.2.3.4'
        list server '4.5.6.7'
        list rebind_domain 'test1.domain'
```

```
        list rebind_domain 'tes2.domain'
        option logqueries '1'
        option resolvfile '/tmp/resolv1.conf.auto'
        list addnhosts 'hosts1'
        list addnhosts 'hosts2'
        option enable_tftp '1'
        option tftp_root '/tmp/tftp'
        option dhcp_boot 'boot.image'
        option filterwin2k '1'
        option nonegcache '1'
        option strictorder '1'
        list bogusnxdomain '1.1.1.1 '
        list bogusnxdomain '2.2.2.2'
        option port '53'
        option dhcpleasemax '150'
        option ednspacket_max '1280'
        option dnsforwardmax '150'
```

Options `local` and domain enable dnsmasq to serve entries in `/etc/hosts` as well as the DHCP client's names as if they were entered into the LAN DNS domain.

For options domainneeded, `boguspriv`, `localise_queries`, and `expandhosts` make sure that requests for these local host names (and the reverse lookup) never get forwarded to the upstream DNS servers.

13.4 Configuring DHCP pools using UCI

Sections of the type dhcp specify per interface lease pools and settings. Typically there is at least one section of this type present in the `/etc/config/dhcp` file to cover the LAN interface.

You can disable a lease pool for a specific interface by specifying the ignore option in the corresponding section.

A minimal example of a dhcp section is shown below.

```
root@VA_router:~# uci show dhcp.lan
dhcp.lan=dhcp
dhcp.lan.interface=lan
dhcp.lan.start=100
dhcp.lan.limit=150
dhcp.lan.leasetime=12h
dhcp.lan.ignore=0
root@VA_router:~# uci export dhcp
config 'dhcp' 'lan'
      option 'interface'   'lan'
      option 'start'       '100'
      option 'limit'        '150'
      option 'leasetime'   '12h'
      option ignore   0
```

<table>
<tr><th>UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: n/a
UCI: dhcp.<pool_name>.interface
Opt: interface</td><td>Defines the interface that is served by this DHCP pool. This must be one of the configured interfaces.<table><tr><td>lan</td><td>Enabled.</td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
<tr><td>Web: n/a
UCI: dhcp.<pool_name>.start
Opt: start</td><td>Defines the offset from the network address for the start of the DHCP pool. It may be greater than 255 to span subnets<table><tr><td>100</td><td></td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
<tr><td>Web: n/a
UCI: dhcp.<pool_name>.limit
Opt: limit</td><td>Defines the offset from the network address for the end of the DHCP pool<table><tr><td>150</td><td></td></tr><tr><td>Range</td><td>0 - 255</td></tr></table></td></tr>
<tr><td>Web: n/a
UCI: dhcp.<pool_name>.leasetime
Opt: leasetime</td><td>Defines the lease time of addresses handed out to clients, for example 12h or 30m.<table><tr><td>12h</td><td>12 hours</td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
<tr><td>Web: n/a
UCI: dhcp.<pool_name>.ignore
Opt: ignore</td><td>Defines whether this DHCP pool is enabled.<table><tr><td>0</td><td>DHCP pool enabled.</td></tr><tr><td>1</td><td>DHCP pool disabled.</td></tr></table></td></tr>
<tr><td>Web: n/a
UCI: dhcp.<pool_name>.force
Opt: force</td><td>Forces DHCP serving on the specified interface even if another DHCP server is detected on the same network segment.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
</table>

Web: n/a UCI: dhcp.<pool_name>.dhcp_option Opt: list dhcp_option	Defines additional options to be added for this dhcp pool. For example with 'list dhcp_option 26,1470' or 'list dhcp_option mtu, 1470' you can assign a specific MTU per DHCP pool. Your client must accept the MTU option for this to work. No options defined Syntax: Option_number, option_value.
Web: n/a UCI: dhcp.<pool_name>.dynamicdhcp Opt: dynamicdhcp	Defines whether to allocate DHCP leases. 1: Dynamically allocate leases. 0: Use /etc/ethers file for serving DHCP leases.
Web: n/a UCI: dhcp.<pool_name>.dynamicdhcp Opt: networkid	Assigns a network-id to all clients that obtain an IP address from this pool.

Table 34: Information table for DHCP pool UCI and package options

13.5 Configuring static leases using UCI

You can assign fixed IP addresses to hosts on your network, based on their MAC (hardware) address.

```
root@VA_router:~# uci show dhcp.mypc
dhcp.mypc=host
root@VA_router:~# uci show dhcp.mypc
dhcp.mypc.ip=192.168.1.2
dhcp.mypc.mac=00:11:22:33:44:55
dhcp.mypc.name=mypc
root@VA_router:~# uci export dhcp
config host 'mypc'
        option ip       '192.168.1.2'
        option mac      '00:11:22:33:44:55'
        option name     'mypc'
```

This adds the fixed IP address 192.168.1.2 and the name "mypc" for a machine with the (Ethernet) hardware address 00:11:22:33:44:55.

14 Configuring VLAN

14.1 Configuration package used

Package	Sections
Network	

14.2 Configuring VLAN using the web interface

14.2.1 Create a VLAN interface

To configure VLAN using the web interface, in the top menu, select **Network ->Interfaces**.

Click **Add new interface**. The Create Interface page appears.

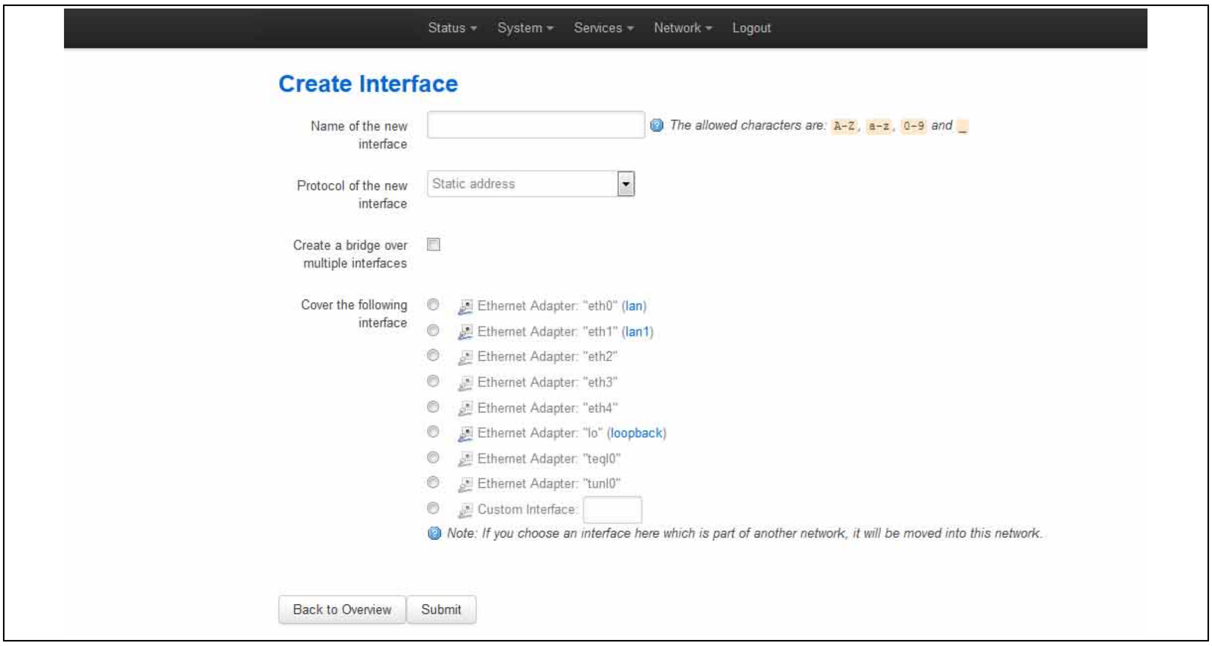


Figure 41: The create interface page

Web Field/UCI/Package Option	Description
Web: Name of the new interface UCI: network.vlan1=interface Opt: interface	Type the name of the new interface. For example, VLAN1.

<table>
<tr><td>Web: Protocol of the new interface
UCI: network.vlan_test.proto
Opt: proto</td><td>Protocol type. Select Static.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>IPv6-in-IPv4 (RFC4213)</td><td>Used with tunnel brokers.</td></tr>
<tr><td>IPv6-over-IPv4</td><td>Stateless IPv6 over IPv4 transport.</td></tr>
<tr><td>GRE</td><td>Generic Routing Encapsulation protocol</td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol</td></tr>
<tr><td>PPP</td><td>Point to Point Protocol</td></tr>
<tr><td>PPPoE</td><td>PPP over Ethernet</td></tr>
<tr><td>PPPoATM</td><td>PPP over ATM</td></tr>
<tr><td>LTE/UMTS/GPRS/EV-DO</td><td>CDMA, UMTS or GPRS connection using an AT-style 3G modem.</td></tr>
</table>
</td></tr>
<tr><td>Web: Create a bridge over multiple interfaces
UCI: network.vlan1.type
Opt: type</td><td>Create a bridge over multiple interfaces.</td></tr>
<tr><td>Web: Cover the following interface
UCI: network.vlan1.ifname
Opt: ifname</td><td>Check the Custom Interface radio button.
Enter a name, for example eth0.100. This will assign VLAN 100 to the eth0 interface.</td></tr>
</table>

Table 35: Information table for the create interface page

Click **Submit**. The Interfaces page for VLAN1 appears.

14.2.2 General setup: VLAN

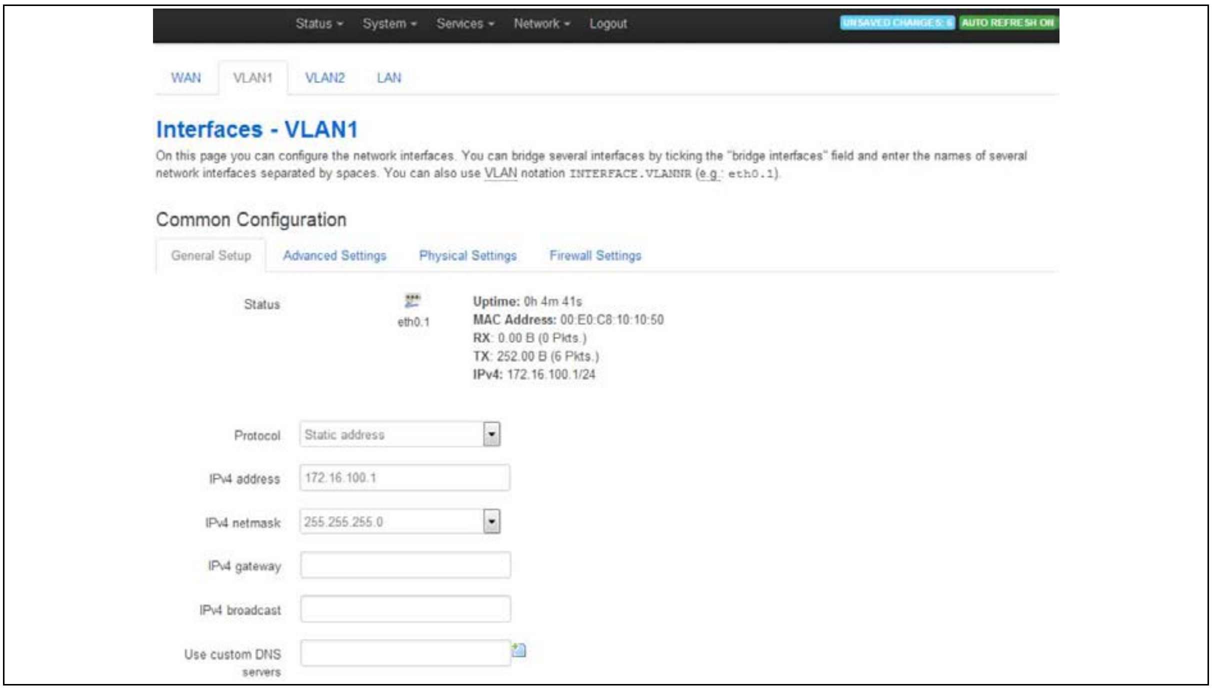


Figure 42: The VLAN 1 interface page

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Protocol
UCI: network.VLAN1.proto
Opt: proto</td><td>Protocol type.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>IPv6-in-IPv4 (RFC4213)</td><td>Used with tunnel brokers.</td></tr>
<tr><td>IPv6-over-IPv4</td><td>Stateless IPv6 over IPv4 transport.</td></tr>
<tr><td>GRE</td><td>Generic Routing Encapsulation protocol</td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol</td></tr>
<tr><td>PPP</td><td>Point to Point Protocol</td></tr>
<tr><td>PPPoE</td><td>PPP over Ethernet</td></tr>
<tr><td>PPPoATM</td><td>PPP over ATM</td></tr>
<tr><td>LTE/UMTS/GPRS/EV-DO</td><td>CDMA, UMTS or GPRS connection using an AT-style 3G modem.</td></tr>
</table></td></tr>
</table>

Web: IPv4 address UCI: network.VLAN1.ipaddr Opt: ipaddr	The IPv4 address of the interface. This is optional if an IPv6 address is provided.
Web: IPv4 netmask UCI: network.VLAN1.netmask Opt: netmask	Subnet mask to be applied to the IP address of this interface.
Web: IPv4 gateway UCI: network.VLAN1.gateway Opt: gateway	IPv4 default gateway to assign to this interface (optional).
Web: Use custom DNS servers UCI: network.VLAN1.dns Opt: dns	List of DNS server IP addresses (optional).

Table 36: Information table for VLAN general settings

Enter the relevant information and click **Save**.

14.2.3 Firewall settings: VLAN

Use this section to select the firewall zone you want to assign to the VLAN interface.

Select unspecified to remove the interface from the associated zone or fill out the create field to define a new zone and attach the interface to it.

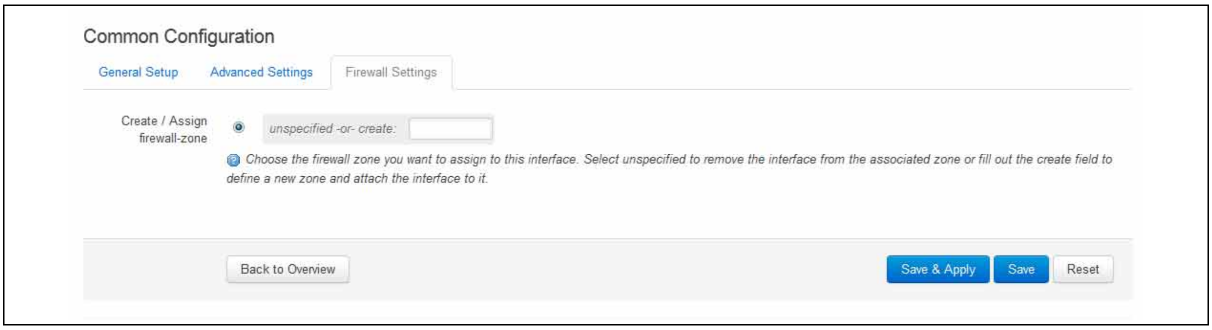


Figure 43: Firewall settings page

When you have added all the VLAN interfaces you require, click **Save & Apply**.

14.3 Viewing VLAN interface settings

To view the new VLAN interface settings, in the top menu, select **Network -> Interfaces**. The Interfaces Overview page appears.

The example below shows two VLAN interfaces configured.

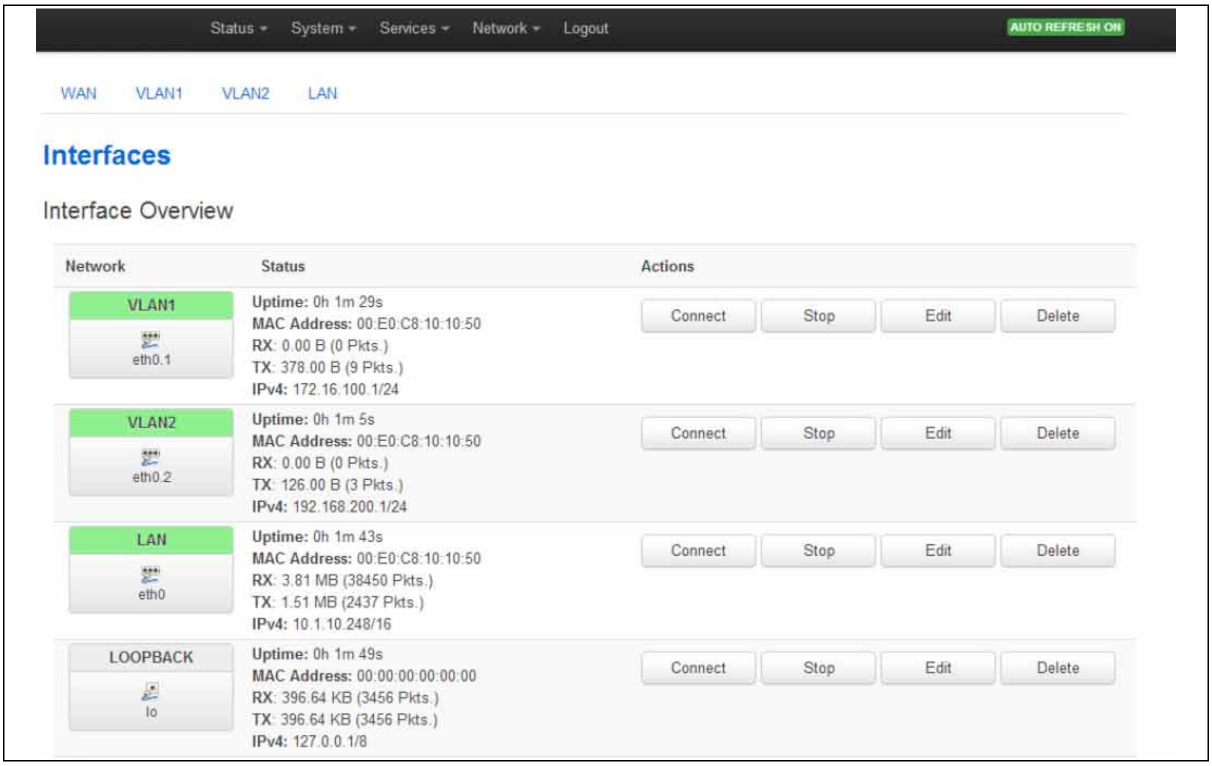


Figure 44: The interface overview page showing two VLAN interfaces

14.4 Configuring VLAN using the UCI interface

You can configure VLANs through CLI.

The VLAN configuration file is stored at:

/etc/config/network

```
# uci export network
package network

config interface 'vlan100'
        option proto 'static'
        option ifname 'eth0.100'
        option monitored '0'
        option ipaddr '192.168.100.1'
        option netmask '255.255.255.0'
        option gateway '192.168.100.10'
        option broadcast '192.168.100.255'
        option dns '8.8.8.8'
```

Modify these settings by running uci set <parameter> command.

When specifying the ifname ensure that it is written in dotted mode, that is, eth1.100 where eth1 is the physical interface assigned to VLAN tag 100.

Note: VLAN1 is, by default the native VLAN and will not be tagged.

15 Configuring static routes

It is possible to define arbitrary IPv4 routes on specific interfaces using route sections. As for aliases, multiple sections can be attached to an interface. These types of routes are most commonly known as static routes.

You can add static routes to the routing table to forward traffic to specific subnets when dynamic routing protocols are not used or they are not configured for such subnets. They can be created based on outgoing interface or next hop IP address.

15.1 Configuration package used

Package	Sections
network	route

15.2 Configuring static routes using the web interface

In the top menu, select **Network -> Static Routes**. The Routes page appears.

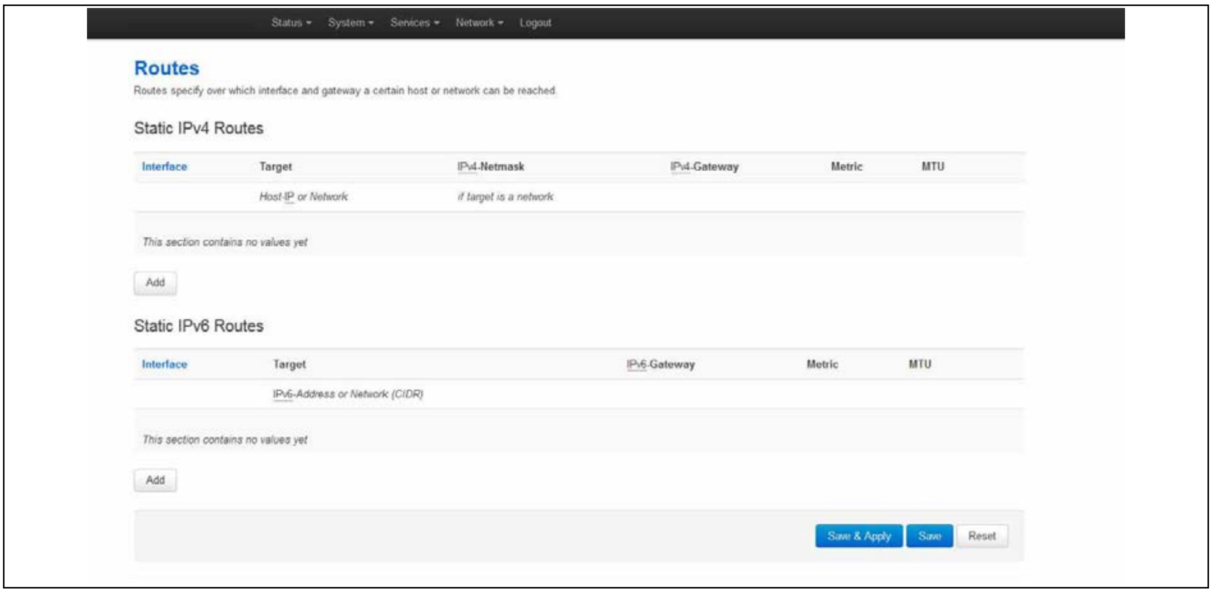


Figure 45: The routes page

In the IPv4 Routes section, click **Add**.

Web Field/UCI/Package Option	Description
Web: Interface UCI: network.@route[0].interface Opt: Interface	Specifies the logical interface name of the parent or master interface this route belongs to. It must refer to one of the defined interface sections.

Web: target UCI: network.@route[0].target Opt: target	Specifies the route network IP address.
Web: netmask UCI: network.@route[0].netmask Opt: netmask	Defines the route netmask. If omitted, 255.255.255.255 is assumed, which makes the target a host address.
Web: Gateway UCI: network.@route[0].gateway Opt: Gateway	Network gateway. If omitted, the gateway from the parent interface is taken. If set to 0.0.0.0 no gateway will be specified for the route.
Web: Metric UCI: network.@route[0].metric Opt: metric	Specifies the route metric to use. 0 Range
Web: MTU UCI: network.@route[0].mtu Opt:mtu	Defines a specific MTU for this route. If omitted, the MTU from the parent interface will be taken. Empty Range

Table 37: Information table for IPv4 static routes section

15.3 Configuring IPv6 routes using the web interface

You can also specify IPv6 routes by defining one or more IPv6 routes. In the IPv6 routes section, click **Add**.

Web Field / UCI / Package Option	Description
Web: Interface UCI: network.@route[1].interface Opt: interface	Specifies the logical interface name of the parent or master interface this route belongs to. It must refer to one of the defined interface sections.
Web: target UCI: network.@route[1].target Opt: target	Specifies the route network IP address, or subnet in CIDR notation: Eample: 2001:0DB8:100:F00:BA3::1/64
Web: Gateway UCI: network.@route[1].gateway Opt: Gateway	Network gateway. If omitted, the gateway from the parent interface is taken. If set to 0.0.0.0 no gateway will be specified for the route.
Web: Metric UCI: network.@route[1].metric Opt: metric	Specifies the route metric to use. 0 Range
Web: MTU UCI: network.@route[1].mtu Opt:mtu	Defines a specific MTU for this route. If omitted the MTU from the parent interface will be taken. Empty Range

Table 38: Information table for IPv6 routes

When you have made your changes, click **Save & Apply**.

15.4 Configuring routes using command line

By default all routes are named 'route', it is identified by @route then the route's position in the package as a number. For example, for the first route in the package using UCI:

```
network.@route[0]=route
network.@route[0].interface=lan
```

Or using package options:

```
config route
        option 'interface' 'lan'
```

However a route can be given a name if desired, for example, a route named 'myroute' will be network.myroute.

To define a named route using UCI, enter:

```
network.name_your_route=route
network.name_your_route.interface=lan
```

To define a named route using package options, enter:

```
config route 'name_your_route'
        option 'interface' 'lan'
```

15.4.1 IPv4 routes using UCI

The command line example routes in the subsections below do not have a configured name.

```
root@VA_router:~# uci show network
network.@route[0]=route
network.@route[0].interface=lan
network.@route[0].target=3.3.3.10
network.@route[0].netmask=255.255.255.255
network.@route[0].gateway=10.1.1.2
network.@route[0].metric=3
network.@route[0].mtu=1400
```

15.4.2 IPv4 routes using pcakage options

```
root@VA_router:~# uci export network
package network
      ….
config route
        option interface 'lan'
        option target '2.2.2.2'
        option netmask '255.255.255.255'
        option gateway '192.168.100.1'
        option metric '1'
        option mtu '1500'
```

15.4.3 IPv6 routes using UCI

```
root@VA_router:~# uci show network
network.@route[1]=route
network.@route[1].interface=lan
network.@route[1].target=2001:0DB8:100:F00:BA3::1/64
network.@route[1].gateway=2001:0DB8:99::1
network.@route[1].metric=1
network.@route[1].mtu=1500
```

15.4.4 IPv6 routes using packages options

```
root@VA_router:~# uci export network
package network
      ….
config route
        option interface 'lan'
        option target '2001:0DB8:100:F00:BA3::1/64'
        option gateway '2001:0DB8:99::1'
      option metric ‘1’
        option mtu '1500'
```

15.5 Static routes diagnostics

15.5.1 Route status

To show the current routing status, enter

```
root@VA_router:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
192.168.100.0   *               255.255.255.0   U     0      0        0
eth0
```

Note: a route will only be displayed in the routing table when the interface is up.

16 Configuring BGP (Border Gateway Protocol)

BGP is a protocol for exchanging routing information between gateway hosts, each with its own router, in a network of autonomous systems. BGP is often the protocol used between gateway hosts on the internet. The routing table contains a list of known routers, the addresses they can reach, and a cost metric associated with the path to each router so that the best available route is chosen.

16.1 Configuration package used

Package	Sections
bgpd	routing
	peer
	routemap

16.2 Configuring BGP using the web interface

In the top menu, select **Network -> BGP**. BGP configuration page appears. The page has three sections: Global Settings, BGP Neighbours and BGP Route Map.

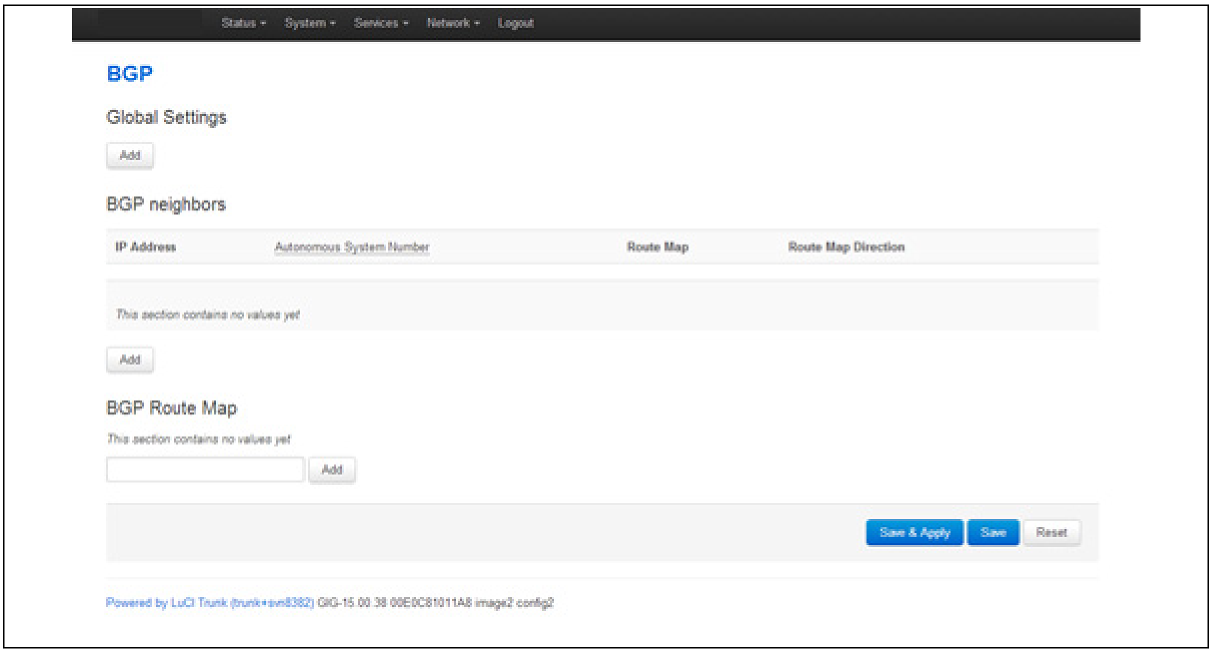


Figure 46: BGP page

16.2.1 BGP global settings

To configure global BGP settings, click **Add**. The Global Settings page appears.

Figure 47: BGP global settings page

Web Field / UCI / Package Option	Description
Web: BGP Enabled UCI: bgpd.bgpd.enabled Opt: enabled	Enables or disables BGP protocol. 1 — Enabled. 0 — Disabled.
Web: Router ID UCI: bgpd.bgpd.router_id Opt: router_id	Sets a Unique Router ID in 4 byte format 0.0.0.0.
Web: Autonomous System Number UCI: bgpd.bgpd.asn Opt: asn	Defines the ASN for the local router. Type in the **ASN**. Blank Range — 1-4294967295
Web: Network UCI: bgpd.bgpd.network Opt: list network	Sets the list of networks that will be advertised to neighbours in prefix format 0.0.0.0/0. Separate multiple networks by a space using UCI. Ensure the network prefix matches the one shown in the routing table. See ‘Routes’ section below.

Table 39: Information table for BGP global settings

16.3 Optionally configure a BGP route map

Route maps provide a means to both filter and/or apply actions to a route. This allows a policy to be applied to routes. Route maps are an ordered list of route map entries each with a set of criteria that must be matched before specific attributes of the route are modified.

Scroll down to the BGP Route Map section.

Type in a name for the BGP route map name and then click **Add**. The ROUTEMAP configuration section appears. Multiple route maps can be configured.

Figure 48: The routemap section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Order
UCI: bgpd.ROUTEMAP.order
Opt: order</td><td>Defines the Route Map order number.<table><tr><td>Blank</td><td></td></tr><tr><td>Range</td><td>1-65535</td></tr></table></td></tr>
<tr><td>Web: Policy Type
UCI: bgpd.ROUTEMAP.permit
Opt: permit</td><td>Defines the actions taken if the entry is matched.<table><tr><td>Deny</td><td>Denies the route.</td></tr><tr><td>Permit</td><td>Permits the route so process the set actions for this entry.</td></tr></table></td></tr>
<tr><td>Web: Match Type
UCI: bgpd.ROUTEMAP.match_type
Opt: match_type</td><td>Defines match type. Available options are as follows:<table><tr><td>IP address</td><td>Matches IP address.</td></tr><tr><td>IP Next Hop</td><td>Matches next hop IP address.</td></tr><tr><td>AS-Path</td><td>Matches AS-path.</td></tr><tr><td>Route Metric</td><td>Matches route metric.</td></tr><tr><td>BGP Community</td><td>Matches BGP community.</td></tr></table></td></tr>
<tr><td>Web: Match value
UCI: bgpd.ROUTEMAP.match
Opt: match</td><td>Defines the value of the match type. Format depends on the Match Type selected. In the case of IP address and BGP Community values, the match value is parsed as a list of items to match.</td></tr>
<tr><td>Web: Set Option
UCI: bgpd.ROUTEMAP.set_type
Opt: set_type</td><td>Defines the set option to be processed on a match. Available options are shown below.<table><tr><td>None</td><td></td></tr><tr><td>IP Next Hop</td><td>Setting option for IP next hop.</td></tr><tr><td>Local Preference</td><td>Setting option for Local Preference.</td></tr><tr><td>Route Weight</td><td>Setting option for Route Weight.</td></tr><tr><td>BGP MED</td><td>Setting option for BGP multi-exit discriminator (BGP metric).</td></tr><tr><td>AS Path to Prepend</td><td>Setting option to prepend AS to AS path.</td></tr><tr><td>BGP Community</td><td>Setting option for BGP community.</td></tr><tr><td>IPv6 Next Hop Global</td><td>Setting option for IPv6 Next Hop Global.</td></tr><tr><td>IPv6 Next Hop Local</td><td>Setting option for IPv6 Next Hop Local.</td></tr></table></td></tr>
</table>

Web: Value UCI: bgpd.ROUTEMAP.set Opt: set	Defines the set value when a match occurs. Value format depends on the set option you have selected.

Table 40: Information table for routemap

16.4 BGP neighbours

To configure BGP neighbours, in the BGP neighbours section, click **Add**. The BGP Neighbours page appears. Multiple BGP neighbours can be configured.

Figure 49: The BGP neighbours section

Web Field/UCI/Package Option	Description
Web: IP Address UCI: bgpd.@peer[0].ipaddr Opt: ipaddr	Sets the IP address of the neighbour.
Web: Autonomous System Number UCI: bgpd.@peer[0].asn Opt: asn	Sets the ASN of the remote peer. Blank | Range | 1-4294967295
Web: Route Map UCI: bgpd.@peer[0].route_map Opt: route_map	Sets route map name to use with this neighbour.
Web: Route Map Direction UCI: bgpd.@peer[0].route_map_in Opt: route_map_in	Defines the direction the route map should be applied. 1 | In 0 | Out

Table 41: Information table for BGP neighbours

16.5 Configuring BGP using UCI

You can also configure BGP using UCI. The configuration file is stored at:
/etc/config/bgpd

```
root@VA_router:~# uci show bgpd
bgpd.bgpd=routing
bgpd.bgpd.enabled=yes
bgpd.bgpd.router_id=3.3.3.3
```

```
bgpd.bgpd.asn=1
bgpd.bgpd.network=11.11.11.0/29 192.168.103.1/32
bgpd.@peer[0]=peer
bgpd.@peer[0].route_map_in=yes
bgpd.@peer[0].ipaddr=11.11.11.1
bgpd.@peer[0].asn=1
bgpd.@peer[0].route_map=ROUTEMAP
bgpd.ROUTEMAP=routemap
bgpd.ROUTEMAP.order=10
bgpd.ROUTEMAP.permit=yes
bgpd.ROUTEMAP.match_type=ip address
bgpd.ROUTEMAP.match=192.168.101.1/32
bgpd.ROUTEMAP.set_type=ip next-hop
bgpd.ROUTEMAP.set='192.168.101.2/32'
```

To change any of the above values use UCI set command.

16.6 Configuring BGP using packages options

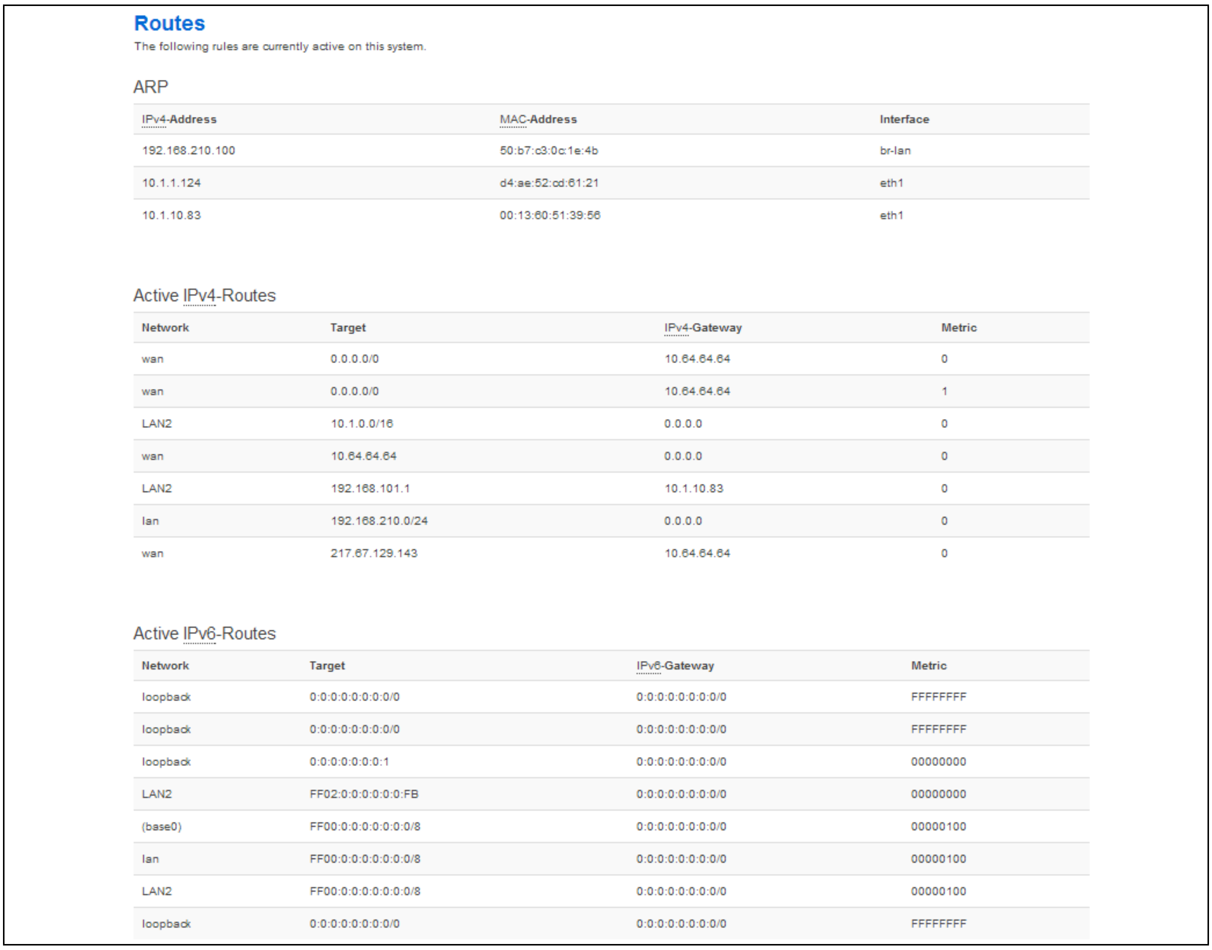
```
root@VA_router:~# uci export bgpd
package bgpd
config routing 'bgpd'
        option enabled 'yes'
        option router_id '3.3.3.3'
        option asn '1'
        list network '11.11.11.0/29'
        list network '192.168.103.1/32'
config peer
        option route_map_in 'yes'
        option ipaddr '11.11.11.1'
        option asn '1'
        option route_map 'ROUTEMAP'

config routemap 'ROUTEMAP'
        option order '10'
```

```
        option permit 'yes'
        option match_type 'ip address'
        option match '192.168.101.1/32'
        option set_type 'ip next-hop'
        option set '192.168.101.2/32'
```

16.7 View routes statistics

To view routes statistics, in the top menu click **Status -> Routes.** The routing table appears.



Routes

The following rules are currently active on this system.

ARP

IPv4-Address	MAC-Address	Interface
192.168.210.100	50:b7:c3:0c:1e:4b	br-lan
10.1.1.124	d4:ae:52:cd:61:21	eth1
10.1.10.83	00:13:60:51:39:56	eth1

Active IPv4-Routes

Network	Target	IPv4-Gateway	Metric
wan	0.0.0.0/0	10.64.64.64	0
wan	0.0.0.0/0	10.64.64.64	1
LAN2	10.1.0.0/16	0.0.0.0	0
wan	10.64.64.64	0.0.0.0	0
LAN2	192.168.101.1	10.1.10.83	0
lan	192.168.210.0/24	0.0.0.0	0
wan	217.67.129.143	10.64.64.64	0

Active IPv6-Routes

Network	Target	IPv6-Gateway	Metric
loopback	0:0:0:0:0:0:0:0/0	0:0:0:0:0:0:0:0/0	FFFFFFFF
loopback	0:0:0:0:0:0:0:0/0	0:0:0:0:0:0:0:0/0	FFFFFFFF
loopback	0:0:0:0:0:0:0:1	0:0:0:0:0:0:0:0/0	00000000
LAN2	FF02:0:0:0:0:0:0:FB	0:0:0:0:0:0:0:0/0	00000000
(base0)	FF00:0:0:0:0:0:0:0/8	0:0:0:0:0:0:0:0/0	00000100
lan	FF00:0:0:0:0:0:0:0/8	0:0:0:0:0:0:0:0/0	00000100
LAN2	FF00:0:0:0:0:0:0:0/8	0:0:0:0:0:0:0:0/0	00000100
loopback	0:0:0:0:0:0:0:0/0	0:0:0:0:0:0:0:0/0	FFFFFFFF

Figure 50: The routing table

To view routes via the command line, enter **route -n.** The routing table appears.

```
root@support:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
10.1.0.0        0.0.0.0         255.255.0.0     U     0      0        0 br-
lan2
```

Figure 51: The routing table using command line

17 Configuring a mobile connection

17.1 Configuration package used

Package	Sections
network	

17.2 Configuring a mobile connection using the web interface

In the top menu, select **Network -> Interfaces**.The Interfaces Overview page appears.

17.2.1 Creating a new mobile interface

To create a new mobile interface, in the Interface Overview section, click **Add new**

interface. The Create Interface page appears.

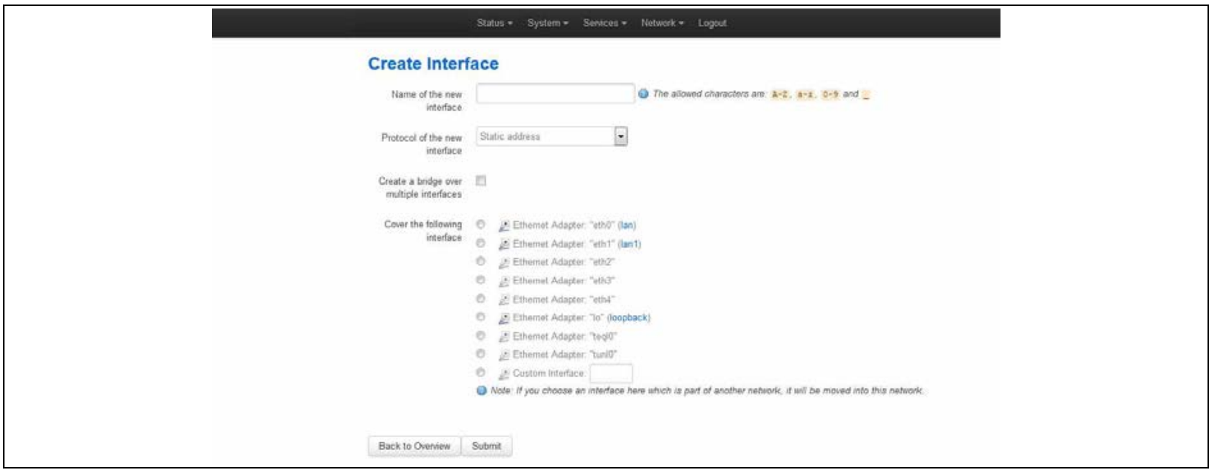


Figure 52: The create interface page

Web Field / UCI / Package Option	Description
Web:Name of the new interface UCI: network.3G=interface Opt: interface	Allowed characters are A-Z, a-z, 0-9 and _

<table>
<tr><td>Web: Protocol of the new interface
UCI: network.3G.proto
Opt: proto</td><td>Protocol type. Select LTE/UMTS/GPRS/EV-DO.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>IPv6-in-IPv4</td><td></td></tr>
<tr><td>IPv6-over-IPv4</td><td></td></tr>
<tr><td>GRE</td><td></td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol.</td></tr>
<tr><td>PPP</td><td></td></tr>
<tr><td>PPPoE</td><td></td></tr>
<tr><td>PPPoATM</td><td></td></tr>
<tr><td>LTE/UMTS/GPRS/EV-DO</td><td>CDMA, UMTS or GPRS connection using an AT-style 3G modem.</td></tr>
</table></td></tr>
<tr><td>Web: Create a bridge over multiple interfaces
UCI: network.3G.type
Opt: type</td><td>Enables bridge between two interfaces.
<table>
<tr><td>0</td><td>Disabled</td></tr>
<tr><td>1</td><td>Enabled</td></tr>
</table></td></tr>
<tr><td>Web: Cover the following interface
UCI: network.3G.ifname
Opt: ifname</td><td>Select interfaces for bridge connection.</td></tr>
</table>

Table 42: Information table for the create interface page

Click **Submit**. The Common Configuration page appears. There are three sections in the mobile interface common configurations:

Section	Description
General Setup	Configure the basic interface settings such as protocol, service type, APN information, user name and password.
Advanced Settings	Setup more indept features such as initionalization timeout, LCP echo failure thresholds and inactivity timeouts.
Firewall settings	Assign a firewall zone to the connection.

17.2.2 Mobile interface: general setup

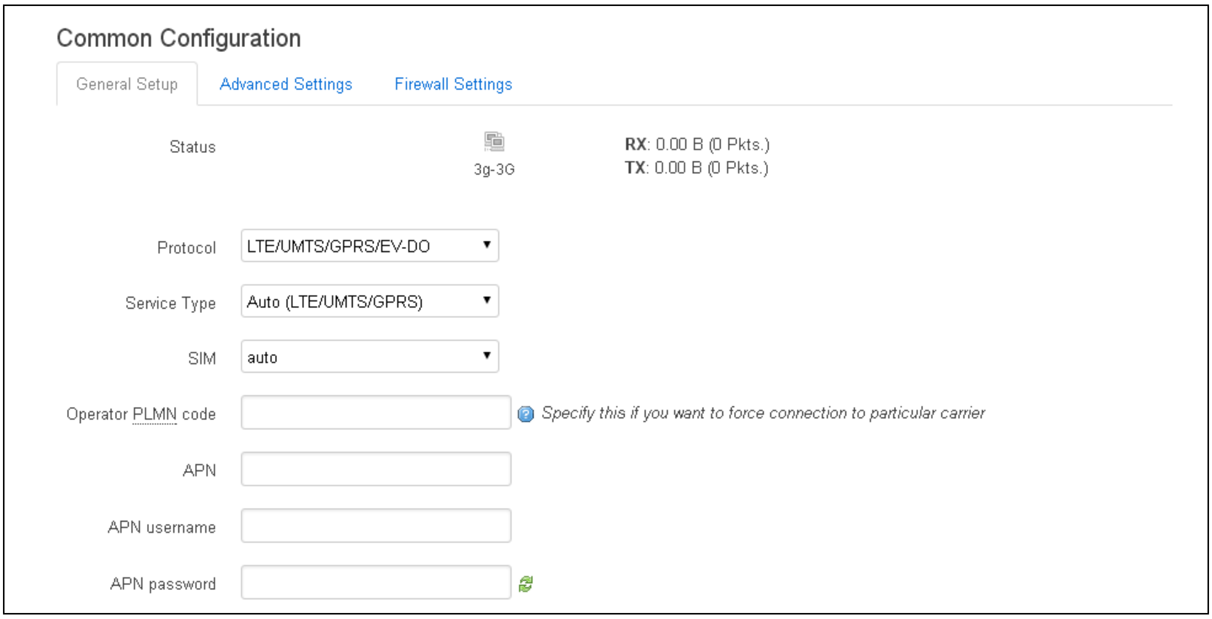


Figure 53: The common configuration page

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web:Status
UCI: n/a
Opt: n/a</td><td>Shows the current status of the interface.</td></tr>
<tr><td>Web: Protocol
UCI: network.3G.proto
Opt: proto</td><td>Protocol type. Select LTE / UMTS / GPRS / EV-DO.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>GRE</td><td></td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol.</td></tr>
<tr><td>PPP</td><td></td></tr>
<tr><td>PPPoE</td><td></td></tr>
<tr><td>PPPoATM</td><td></td></tr>
<tr><td>LTE/UMTS/ GPRS/EV-DO</td><td>CDMA, UMTS or GPRS connection using an AT-style 3G modem.</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>Web: Service Type
UCI: network.3G.service
Opt: service</td><td>Service type that will be used to connect to the network.<table><tr><td>gprs_only</td><td>Allows GSM module to only connect to gprs network</td></tr><tr><td>lte_only</td><td>Allows GSM module to only connect to lte network</td></tr><tr><td>cdma</td><td>Allows GSM module to only connect to cdma network</td></tr><tr><td>auto</td><td>GSM module will automatically detect the best available technology code.</td></tr></table></td></tr>
<tr><td>Web: Operator PLMN code
UCI: network.3G.operator
Opt: operator</td><td>Specify an operator code to force the connection to a particular carrier.</td></tr>
<tr><td>Web: SIM
UCI: network.3G.sim
Opt: sim</td><td>Defines which SIM (any, 1 or2) is used on this interface.</td></tr>
<tr><td>Web: APN
UCI: network.3G.apn
Opt: apn</td><td>APN name of Mobile Network Operator.</td></tr>
<tr><td>Web: APN username
UCI: network.3G.username
Opt:username</td><td>Username used to connect to APN.</td></tr>
<tr><td>Web: APN password
UCI: network.3G.password
Opt:password</td><td>Password used to connect to APN.</td></tr>
</table>

Table 43: Information table for comman configuration settings

The Modem Configuration link at the bottom of the page is used for SIM pincode and SMS configuration. Read the chapter ‘Mobile Manager’.

17.2.3 Mobile interface: advanced settings

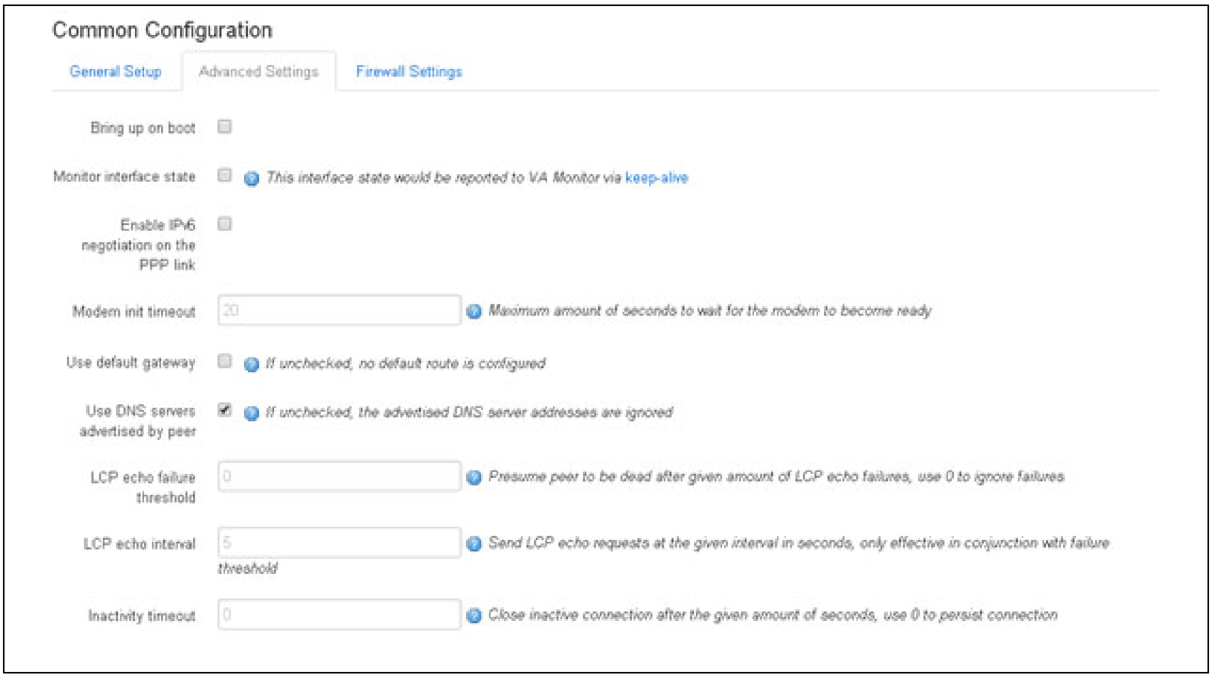


Figure 54: The advanced settings tab

Web Field / UCI / Package Option	Description
Web:Bring up on boot UCI: network.3G.auto Opt: auto	Enables the interface to connect automatically on boot up
Web: Monitor interface state UCI: network.3G.monitored Opt: monitored	Enabled if status of interface is presented on Monitoring platform
Web: Enable IPv6 negotiation on the PPP link UCI: network.3G.ipv6 Opt: ipv6	Enables IPv6 routing on the interface.
Web: Modem int timeout UCI: network.3G.maxwait Opt: maxwait	Maximum amount of seconds to wait for the modem to become ready.
Web: Use default gateway UCI: network.3G.defaultroute Opt: defaultroute	If unchecked, no default route is configured.
Web: Use gateway metric UCI: network.3G.metric Opt: metric	Uses the specified metric.
Web: Use DNS servers advertised by peer UCI: network.3G.peerdns Opt: peerdns	If unchecked, the advertised DNS server addresses are ignored

Web: Use custom DNS servers UCI: network.3G.dns Opt: dns	Specify DNS server.
Web: LCP echo failure threshold UCI: network.3G.keepalive Opt: keepalive	Presume peer to be dead after given amount of LCP echo failures, use 0 to ignore failures This command is used in conjunction with the LCP echo interval. The syntax is as follows uci network.3G.keepalive=<echo failure threshold> <echo interval> Example: Uci set network.3G.keepalive=15 10
Web: LCP echo internal UCI: network.3G.keepalive Opt: keepalive	Send LCP echo requests at the given interval in seconds, only effective in conjunction with failure This command is used in conjunction with the LCP echo failure threshold. The syntax is as follows uci network.3G.keepalive=<echo failure threshold> <echo interval> Example: Uci set network.3G.keepalive=15 10
Web: Inactivity timeout UCI: network.3G.demand Opt: demand	Close inactive connection after the given amount of seconds, use **0** to persist connection.

Table 44: Information table for general set up page

17.2.4 Mobile interface: firewall settings

Use this section to select the firewall zone you want to assign to the interface.

Select **unspecified** to remove the interface from the associated zone or fill out the create field to define a new zone and attach the interface to it.

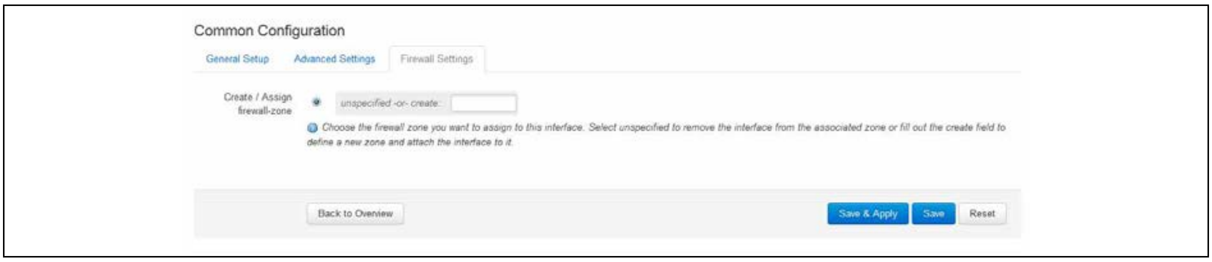


Figure 55: Firewall settings page

17.3 Configuring a mobile connection using UCI

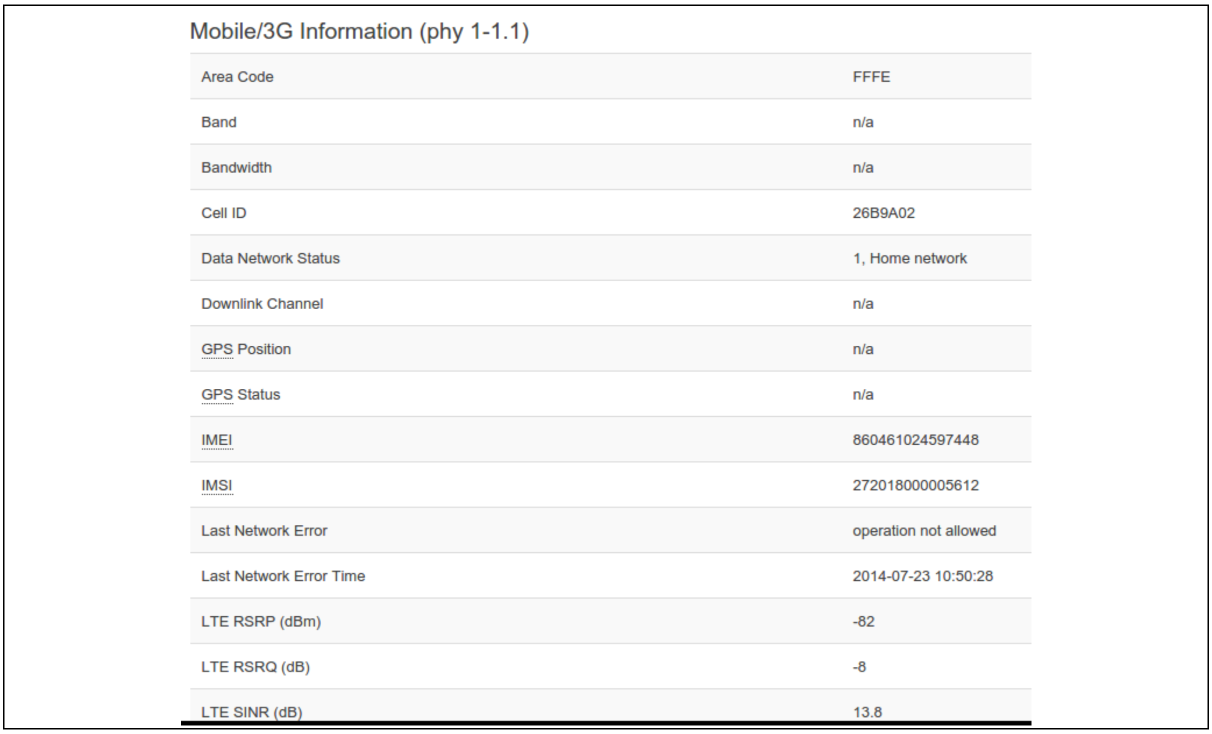
A basic mobile connection can be established by using the following UCI commands:

```
uci set network.3G=interface
uci set network.3G.proto=3g
```

```
uci set network.3G.device=/dev/ttyACM0
uci set network.3G.auto=no
uci set network.3G.defaultroute=1
uci set network.3G.service=auto
```

17.4 Mobile interface diagnostics

To view mobile connectivity information, in the top menu, select **Status -> Mobile Stats**.


Mobile/3G Information (phy 1-1.1)

Area Code	FFFE
Band	n/a
Bandwidth	n/a
Cell ID	26B9A02
Data Network Status	1, Home network
Downlink Channel	n/a
GPS Position	n/a
GPS Status	n/a
IMEI	860461024597448
IMSI	272018000005612
Last Network Error	operation not allowed
Last Network Error Time	2014-07-23 10:50:28
LTE RSRP (dBm)	-82
LTE RSRQ (dB)	-8
LTE SINR (dB)	13.8

Figure 56: The mobile stats page

17.4.1 Mobile status using UCI

To display information and status of mobile interfaces such as 3G, 4G or CDMA, enter:

```
root@VA_router:~# cat /var/state/mobile
mobile.3g_1_1_1=status
mobile.3g_1_1_1.auto_info=/etc/3g_1-1.1.auto
mobile.3g_1_1_2=status
mobile.3g_1_1_2.auto_info=/etc/3g_1-1.2.auto
mobile.3g_1_1_1.sim_slot=1
```

```
mobile.3g_1_1_1.sim_in=yes
mobile.3g_1_1_1.imsi=240016005892879
mobile.3g_1_1_1.registered=1, Home network
mobile.3g_1_1_1.reg_code=1
mobile.3g_1_1_1.registered_pkt=1, Home network
mobile.3g_1_1_1.reg_code_pkt=1
mobile.3g_1_1_1.area=FFFE
mobile.3g_1_1_1.cell=189150A
mobile.3g_1_1_1.tech=7
mobile.3g_1_1_1.technology=E-UTRAN
mobile.3g_1_1_1.operator=0,0,"Vodafone",7
mobile.3g_1_1_1.sim1_iccid=89460127120912066226
mobile.3g_1_1_2.sim_slot=1
mobile.3g_1_1_2.sim_in=yes
mobile.3g_1_1_2.operator="Vodafone"
mobile.3g_1_1_2.cdma_roaming=Not Roaming
mobile.3g_1_1_2.cdma_roaming_code=0
mobile.3g_1_1_2.cdma_srvmode=EVDO Rev B
mobile.3g_1_1_2.cdma_srvmode_code=5
mobile.3g_1_1_2.cdma_total_drc=0.0 kbps
mobile.3g_1_1_2.cdma_carr_cnt=2
mobile.3g_1_1_2.cdma_rx0=78
mobile.3g_1_1_2.sig_dbm=nan
mobile.3g_1_1_2.cdma_rx1=105
```

18 Configuring mobile manager

The Mobile Manager feature allows you to configure SIM settings.

Basic settings	Enable SMS, configure SIM pincode, select roaming SIM and collect ICCCIDs.
Callers	Configure callers that can use SMS.
Roaming Interface Template	Configure common values for interface created by Automatic Operator Selection.

18.1 Configuration package used

Package	Sections
mobile	Main
	Calllers
	Roaming template

18.2 Configuring mobile manager using the web interface

Select **Services -> Mobile Manager**. The Mobile Manager page appears.

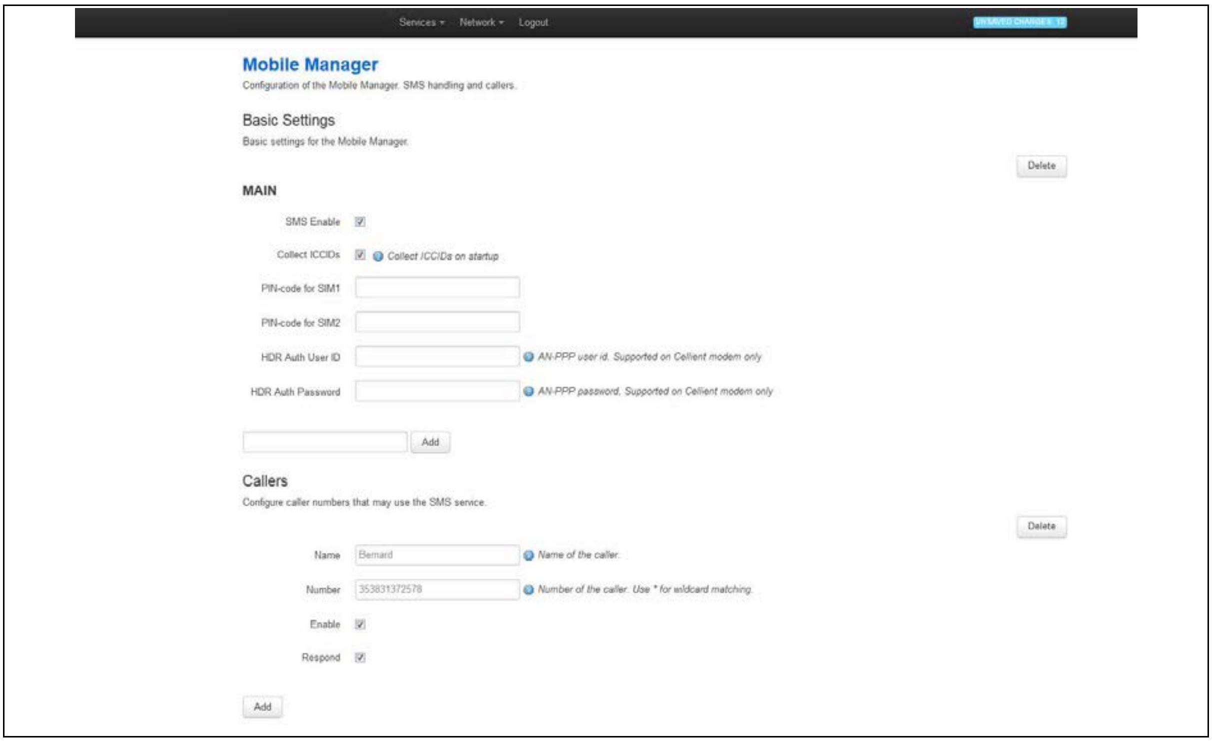


Figure 57: The mobile manager page

Web Field / UCI / Package Option	Description
Web: SMS Enable UCI: mobile.main.sms Opt: sms	Enables or disables SMS functionality. 0 Disabled. 1 Enabled.

Web: Collect ICCIDs UCI: mobile.main.init_get_iccids Opt: init_get_iccids	Enables or disables integrated circuit card identifier ICCID's collection functionality. If enabled then both SIM 1 and SIM 2 ICCIDs will be collected otherwise it will default to SIM 1. This will be displayed under mobile stats. 0 – Disabled. 1 – Enabled.
Web: PIN code for SIM1 UCI: mobile.main.sim1pin Opt:sim1pin	Depending on the SIM card specifiy the pin code for SIM 1. Blank Range – Depends on the SIM provider.
Web: PIN code for SIM2 UCI: mobile.main.sim2pin Opt:sim2pin	Depending on the SIM card specifiy the pin code for SIM 2. Blank Range – Depends on the SIM provider.
Web: HDR Auto User ID UCI: mobile.main.hdr_userid Opt:hdr_userid	AN-PPP user ID. Supported on Cellient (CDMA) modem only. Blank Range – Depends on the CDMA provider.
Web: HDR Auto User Password UCI: mobile.main.hdr_password Opt:hdr_password	AN-PPP password. Supported on Cellient (CDMA) modem only. Blank Range – Depends on the CDMA provider.
Web: Name UCI: mobile.@caller[0].name Opt:name	Name assigned to the caller. Blank Range – No limit.
Web: Number UCI: mobile.@caller[0].number Opt:number	Number of the caller allowed to SMS the router. Add in specific caller numbers, or use the wildcard symbol. Blank Range – No limit Characters – Global value (*) is accepted; International value (+) is accepted
Web: Enable UCI: mobile.@caller[0].enabled Opt:enabled	Enables or disables incoming callerID. 0 – Disabled. 1 – Enabled.
Web: Respond UCI: mobile.@caller[0].respond Opt:respond	If checked, the router will return an SMS. Select **Respond** if you want the router to reply. 0 – Disabled. 1 – Enabled.

Table 45: Information table for mobile manager

When you have made your changes, click **Save & Apply** and then reboot.

18.3 Configuring mobile manager using UCI

The following example shows how to enable the SMS functionality to receive and respond from certain caller ID numbers.

```
uci set mobile.main=mobile
uci set mobile.main.sim1pin=0000
uci set mobile.main.sim2pin=0000
uci set mobile.main.roaming_sim=none
uci set mobile.main.sms=yes
uci set mobile.main.hdr_password=5678
uci set mobile.main.hdr_userid=1234
uci set mobile.main.init_get_iccids=yes
uci set mobile.@caller[0]=caller
uci set mobile.@caller[0].name=user1
uci set mobile.@caller[0].number=3538712345678
uci set mobile.@caller[0].enabled=yes
uci set mobile.@caller[0].respond=yes
uci set mobile.@caller[1]=caller
uci set mobile.@caller[1].name=user2
uci set mobile.@caller[1].number=3538723456789
uci set mobile.@caller[1].enabled=yes
uci set mobile.@caller[1].respond=yes
package mobile
config mobile 'main'
        option sim1pin '0000'
        option sim2pin '0000'
        option roaming_sim 'none'
        option sms 'yes'
        option hdr_password '5678'
        option hdr_userid '1234'
        option init_get_iccids 'yes'
config caller
        option name 'vasupport'
        option number '353871234567'
        option enabled 'yes'
        option respond 'yes'

config caller
        option name 'vasupport1'
```

```
        option number '353872345678'
        option enabled 'yes'
        option respond 'yes'
```

18.4 Configuring a roaming interface template via the web interface

For more information on Roaming Interface Template configuration, read the chapter, ‘Automatic Operator Selection’.

18.5 Monitoring SMS

You can monitor inbound SMS messages using the router’s web browser or via an SSH session.

To monitor via the web browser, login and select **Status >system log**.

Scroll to the bottom of the log to view the SMS message.

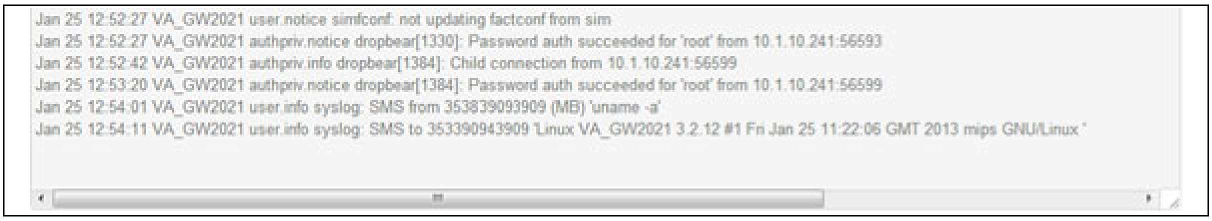


Figure 58: Example of output from system log

To monitor via SSH, login and enter

```
logread –f &.
```

An outgoing SMS message appears.

```
sendsms 353879876543 ‘hello’
root@VirtualAccess:~# Jul 23 14:29:11 user.notice VirtualAccess
mobile[1737]: Queue sms to 353879876543 "hello"
```

18.6 Sending SMS from the router

You can send an outgoing message via the command line using the following syntax:

```
sendsms 353879876543 ‘hello’
root@VirtualAccess:~# Jul 23 14:29:11 user.notice VirtualAccess
mobile[1737]: Queue sms to 353879876543 "hello"
```

18.7 Sending SMS to the router

The router can accept UCI show and set commands via SMS if the caller is enabled.

Note: commands are case sensitive.

An example would be to SMS the SIM card number by typing the following command on the phone and checking the SMS received from the router.

```
uci show mobile.@caller[0].number
```

19 Configuring a WiFi connection

This section explains how to configure WiFi on a Virtual Access router using the web interface or via UCI.

WiFi can act as an Access Point (AP) to another device in the network or it can act as a client to an existing AP.

You can configure WiFi in two different ways:

- on a new interface, or
- on an existing interface

19.1 Configuration packages used

Package	Sections
network	wlan_ap wlan_client
wireless	wifi-device wifi-iface

19.2 Configuring a WiFi interface

To create a new WiFi interface via the web interface, in the top menu, click **Network -> Wifi**. The Wireless overview page appears.

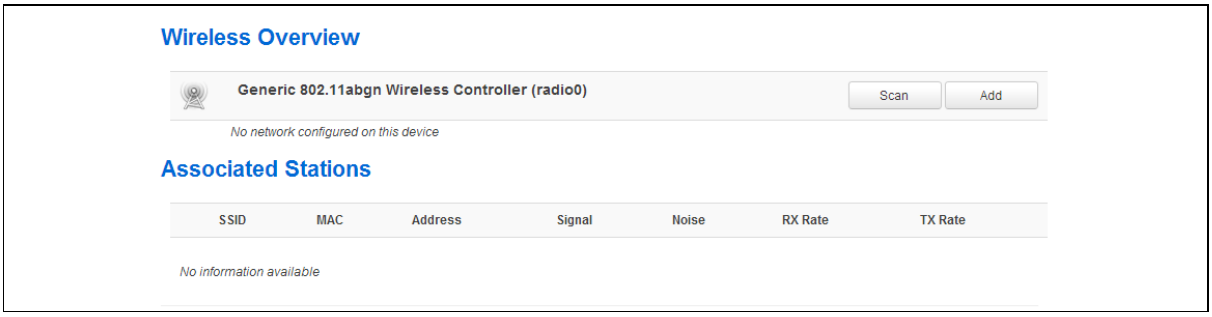


Figure 59: The wireless overview page

Click **Add** to create a new WiFi interface. The Wireless Network configuration page appears. The Wireless Network configuration page consists of two sections:

Section	Description
Device Configuration	Configuration of physical wireless radio settings such as channel and transmit power settings, HT mode, country code, distance optimization, fragmentation threshold and RTS/CTS threshold. The settings are shared among all defined wireless networks.
Interface Configuration	Configuration of the network interface - interface name, mode, network settings, security and filtering

19.2.1 Wireless network: device configuration

The Device Configuration section covers physical settings of the radio hardware such as channel, transmit power or antenna selection, which is shared among all defined wireless networks (if the radio hardware is multi-SSID capable). There are two sections within the Device Configuration section.

Section	Description
General Setup	Channel and transmit power settings.
Advanced Settings	HT mode, country code, distance optimization, fragmentation threshold and RTS/CTS threshold.

19.2.1.1 Device configuration: general setup

Figure 60: The device configuration general setup section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Wireless network
UCI: wireless.radio0.disabled
Opt: disanabled</td><td>Enable or disables a wireless<table><tr><td>1</td><td>Disable Wifi interface</td></tr><tr><td>0</td><td>Enable Wifi interface</td></tr></table></td></tr>
<tr><td>Web: Channel
UCI: wireless.radio0.channel
Opt: channel</td><td>Select the channel you require.<table><tr><td>Range</td><td>1-11</td></tr><tr><td>11 (2.462GHz)</td><td></td></tr></table></td></tr>
<tr><td>Web: Transmit power
UCI: wireless.radio0.txpower
Opt: txpower</td><td>Select the transmit power range range you require.<table><tr><td>Range</td><td>0dBm(1mW)-17dBm(50mW)</td></tr><tr><td>17dBM(50mW)</td><td></td></tr></table></td></tr>
</table>

Table 46: Information table for the device configuration section

19.2.1.2 Device configuration: advanced settings

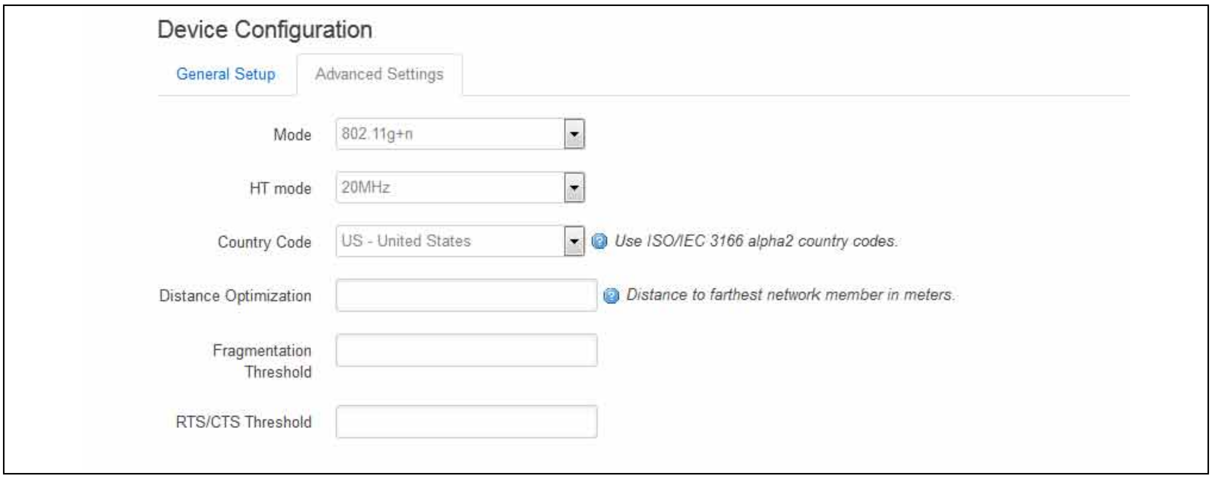


Figure 61: The device configuration advanced settings section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Mode
UCI: wireless. radio0.hwmode
Opt: hwmode</td><td>Mode options.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Auto</td><td>Wireless protocl negotiate with supplicat device.</td></tr>
<tr><td>802.11b</td><td>Select the wireless protocol to use</td></tr>
<tr><td>802.11g</td><td>Select the wireless protocol to use</td></tr>
<tr><td>802.11a</td><td>Select the wireless protocol to use</td></tr>
<tr><td>802.11g+n</td><td>Select the wireless protocol to use</td></tr>
<tr><td>802.11a+n</td><td>Select the wireless protocol to use</td></tr>
</table></td></tr>
<tr><td>Web: HT mode
UCI: wireless.radio0.htmode
Opt: country</td><td>HT mode options.
<table>
<tr><td>20MHz</td><td>specifies the channel width in 802.11</td></tr>
<tr><td>40MHz 2nd channel below</td><td>specifies the channel width in 802.11</td></tr>
<tr><td>40MHz 2nd channel above</td><td>specifies the channel width in 802.11</td></tr>
</table></td></tr>
<tr><td>Web: Country Code
UCI: wireless.radio0.country
Opt: country</td><td>Sets the country code. Use ISO/1EC 3166 alpha2 country codes.</td></tr>
<tr><td>Web: Distance Optimization
UCI: wireless.radio0.distance
Opt: distance</td><td>Defines the distance between the AP and the furthest client in meters
<table>
<tr><td>15</td><td>15 meters</td></tr>
<tr><td>Range</td><td></td></tr>
</table></td></tr>
<tr><td>Web: Fragmentation Threshold
UCI: wireless.radio0.frag
Opt: frag</td><td>Defines the fragmentation threshold
<table>
<tr><td>None</td><td>Routers defults applied</td></tr>
<tr><td>Range</td><td></td></tr>
</table></td></tr>
</table>

Web: RTS/CTS Threshold UCI: wireless.radio0.rts Opt: rts	Defines the RTS/CTS threshold None: Router defaults applied Range:

Table 47: Information table for device configuration advanced settings

19.2.2 Wireless network: interface configuration

The interface configuration section is used to configure the network and security settings. It has three sub sections.

Section	Description
General Setup	Identification, network and mode settings.
Wireless Security	Encryption, cipher and key security settings
MAC Filter	MAC address filter settings.

19.2.2.1 Interface configuration: general setup

Use this section to configure the interface name, mode and network settings. Differing web options may be presented depending on the Mode selected.

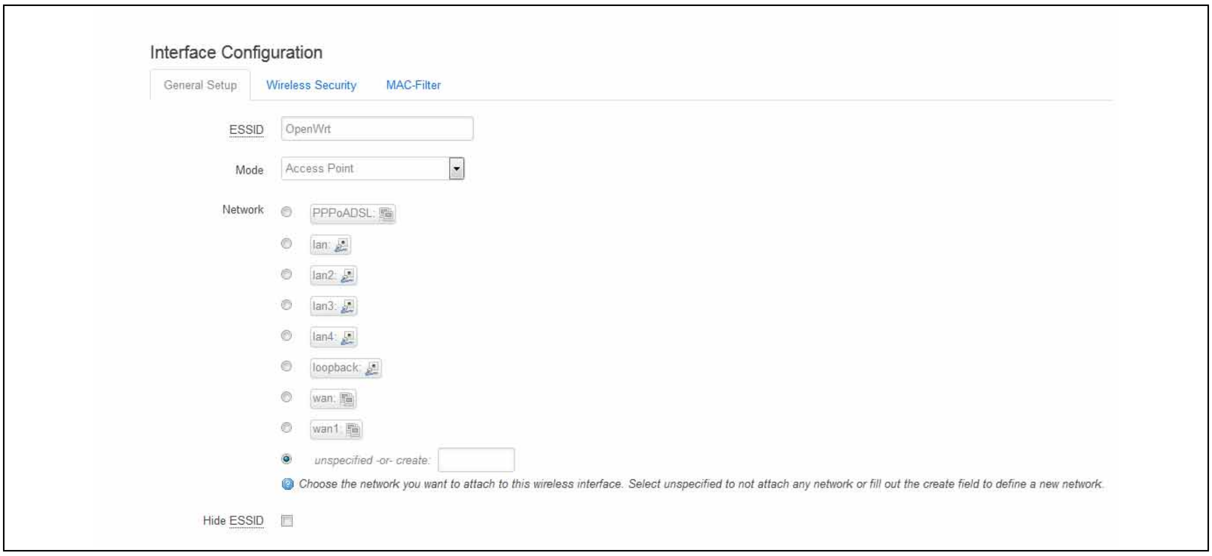


Figure 62: The interface configuration general setup section

Web Field / UCI / Package Option	Description
Web: ESSID UCI: wireless. @wifi-iface[0]..ssid Opt: ssid	Extended Service Set Identification. Type the name of the wireless local area network.

<table>
<tr><td>Web: Mode
UCI: wireless.@wifi-iface[0].mode
Opt: mode</td><td>Mode type. For AP mode, select Access Point.<table><tr><th>Web value</th><th>UCI</th></tr><tr><td>Access Point</td><td>ap</td></tr><tr><td>Client</td><td>sta</td></tr><tr><td>Ad-Hoc</td><td>adhoc</td></tr><tr><td>802.11s</td><td>mesh</td></tr><tr><td>Pseudo Ad-Hoc (ah demo)</td><td>ahdemo</td></tr><tr><td>Monitor</td><td>monitor</td></tr><tr><td>Access Point (WDS)</td><td>ap-wds</td></tr><tr><td>Client (WDS)</td><td>sta-wds</td></tr></table></td></tr>
<tr><td>Web: Mode
UCI: wireless.@wifi-iface[0].bssid
Opt: bssid</td><td>Defines the BSSID value. Only displayed if using client, ad-hoc or client (wds) modes.</td></tr>
<tr><td>Web: Network
UCI:wireless.@wifi-iface[0].network
Opt: network</td><td>The network the wireless interface is attached to. If using an existing interface select the appropriate network. Select unspecified to not attach to any network or fill out the create field to define a new network.</td></tr>
<tr><td>Web: Hide ESSID
UCI: wireless.@wifi-iface[0].hidden
Opt: hidden</td><td>Hides the SSID when enabled. Only displayed if using access point or access point (wds) modes<table><tr><td>1</td><td>Enabled.</td></tr><tr><td>0</td><td>Disabled.</td></tr></table></td></tr>
</table>

Table 48: Information table for the interface configuration general setup section

19.2.2.2 Interface configuration: wireless security

Use this section to configure encryption, ciper and create a security key. Differing options wil be defined depending on the encryption selected.

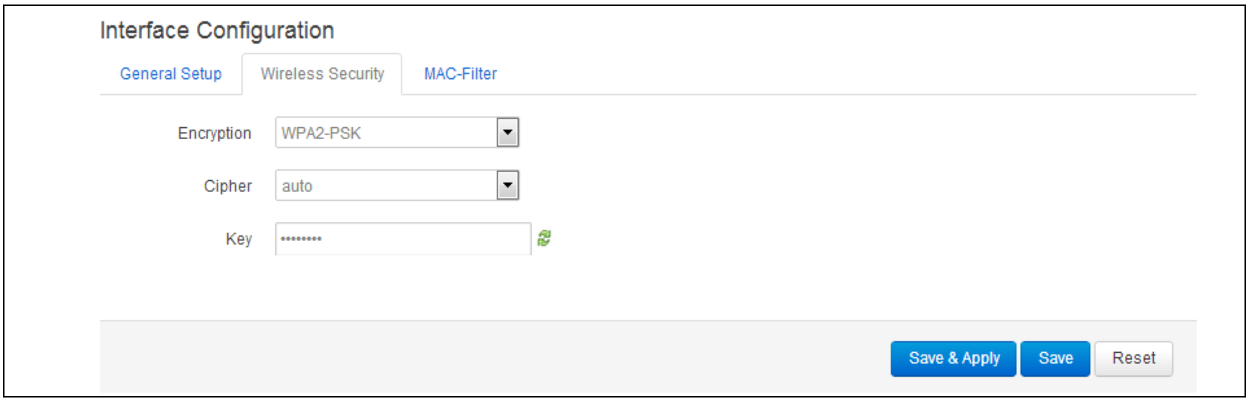


Figure 63: The wireless security section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Encryption
UCI: wireless.@wifi-iface[0].encryption
Opt: encryption</td><td>Method of Encryption.
<table><tr><th>Web value</th><th>UCI value</th></tr><tr><td>No encryption</td><td>none</td></tr><tr><td>WEP Open System</td><td>wep-open</td></tr><tr><td>WEP Shared Key</td><td>wep-shared</td></tr><tr><td>WPA–PSK</td><td>psk</td></tr><tr><td>WPA2–PSK</td><td>psk2</td></tr><tr><td>WPA-PSK/WPA2-PSK Mixed Mode</td><td>psk-mixed</td></tr><tr><td>WPA-EAP</td><td>wpa</td></tr><tr><td>WPA2-WAP</td><td>wpa2</td></tr></table></td></tr>
<tr><td>Web: Cipher
UCI: wireless.@wifi-iface[0].cipher=
Opt: cipher</td><td>Cipher type. Only displayed if WPA encryption modes are selected.
<table><tr><th>Web value</th><th>UCI</th></tr><tr><td>Auto</td><td>auto</td></tr><tr><td>Force CCMP (AES)</td><td>ccmp</td></tr><tr><td>Force TKIP</td><td>tkip</td></tr><tr><td>Force TKIP and CCMP</td><td>tkip+ccmp</td></tr></table></td></tr>
<tr><td>Web: Key
UCI:wireless.@wifi-iface[0].key
Opt: key</td><td>Specifies the wireless key authentication phrase.</td></tr>
<tr><td>Web: Key #1
UCI:wireless.@wifi-iface[0].key1
Opt: key1</td><td>Specifies the first wireless key authentication phrase.</td></tr>
<tr><td>Web: Key #2
UCI:wireless.@wifi-iface[0].key2
Opt: key2</td><td>Specifies the second wireless key authentication phrase.</td></tr>
<tr><td>Web: Key #3
UCI:wireless.@wifi-iface[0].key3
Opt: key3</td><td>Specifies the third wireless key authentication phrase.</td></tr>
<tr><td>Web: Key #4
UCI:wireless.@wifi-iface[0].key4
Opt: key4</td><td>Specifies the fourth wireless key authentication phrase.</td></tr>
<tr><td>Web: Radius Authentication-Server
UCI:wireless.@wifi-iface[0].auth_server
Opt: auth server</td><td>Defines the Radius server for EAP authentication.</td></tr>
<tr><td>Web: Radius Authentication-Port
UCI:wireless.@wifi-iface[0].auth_port
Opt: auth_port</td><td>Defines the Radius server port for EAP authentication.</td></tr>
<tr><td>Web: Radius Authentication-Secret
UCI:wireless.@wifi-iface[0].auth_secret
Opt: auth_secret</td><td>Defines the Radius server secret for EAP authentication.</td></tr>
</table>

Web: Radius Accounting-Server UCI:wireless.@wifi-iface[0].acct_server Opt: acct_server	Defines the Radius server for EAP accounting.
Web: Radius Accounting -Port UCI:wireless.@wifi-iface[0].acct_port Opt: acc_port	Defines the Radius port for EAP accounting.
Web: Radius Accounting -Secret UCI:wireless.@wifi-iface[0].acct_secret Opt: acct_secret	Defines the Radius secret for EAP accounting.
Web: NAS ID UCI:wireless.@wifi-iface[0].nasid Opt: nasid	Defines the nas id for the wireless interface.

Table 49: Information table for the interface configuration wireless security section

19.2.2.3 Interface configuration: MAC filter

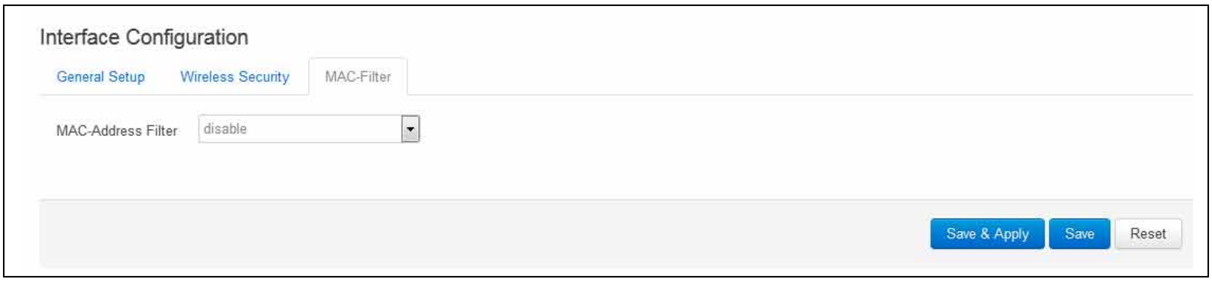


Figure 64: The MAC filter section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: MAC-Address Filter
UCI: wireless.@wifi-iface[0].macfilter
Opt: macfilter</td><td>MAC Address filtering process.
<table>
<tr><th>Option</th><th>Description</th><th>UCI</th></tr>
<tr><td>Disable</td><td>Disables MAC Address filter.</td><td>disable</td></tr>
<tr><td>Allow listed only</td><td>Allows only the MAC address listed in the text field.</td><td>allow</td></tr>
<tr><td>Allow all except listed</td><td>Allows everything but the MAC address listed in the text field.</td><td>deny</td></tr>
</table></td></tr>
<tr><td>Web: MAC -List
UCI: wireless.@wifi-iface[0].maclist
Opt: list maclist</td><td>Defines the MAC addresses to use. Multiple MAC address should be separated by a space if using UCI. MAC must be in the format hh:hh:hh:hh:hh:hh</td></tr>
</table>

Table 50: Information table for interface configuration MAC filter section

19.3 Configuring WiFi in AP mode

AP mode is when the routers WiFi is used as an access point to one of the routers other interfaces. For example, if a router is connected to the internet via 3G, the WiFi on the router can be used as an access point for other devices to connect to the router and use its 3G internet connection.

19.3.1 AP Mode on a new interface

Configure the Wifi network in AP mode as described in the above section 'Configuring a WiFi interface', selecting a new interface for the Wireless Network in the Interface Configuration section.

Example:

```
wireless.@wifi-iface[0].network=newwifiAP
wireless.@wifi-iface[0].mode=ap
```

Next in the top menu, select **Network -> Interfaces**. The Interface Overview page appears.

In the Interface Overview page, click **Edit** on the newly created WiFi interface.

19.3.2 AP mode on an existing Ethernet Interface

Configure the WWiFi network in AP mode as described in the above section 'Configuring a WiFi interface'.

Next, in the top menu, select **Network -> Interfaces**. The Interface Overview page appears.

In the Interface Overview page, click **Edit** on the Ethernet interface that will be bridged into the router's WiFi AP. The Common Configuration page appears. It has four sections.

This configuration only uses the **Physical Settings** section.

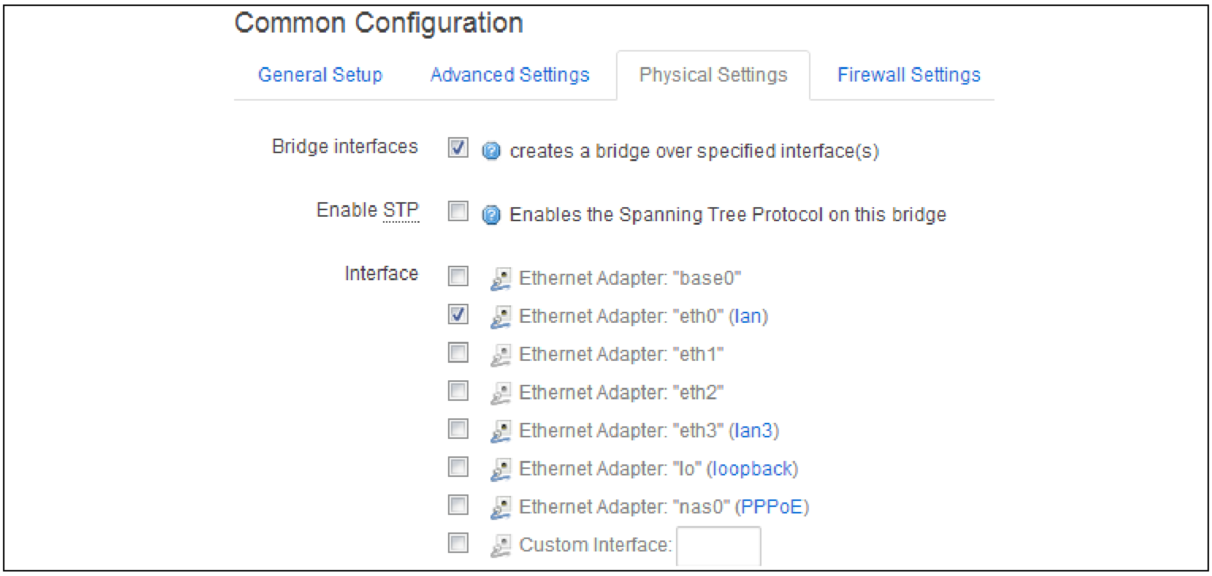


Figure 65: The physical settings section in the common configuration page

Web Field/UCI/Package Option	Description
Web: Bridge Interfaces UCI: network.lan.type Opt: Type	Creates a bridge over the specified interface. Empty: Bridge: Configures a bridge over multiple interfaces.
Web: Enable STP UCI: network.lan.stp Opt: stp	Enables the Spanning Tree Protocol on this bridge. 0: Disabled. 1: Enabled.
Web: Interface UCI: network.lan.ifname Opt:ifname	Select the physical interfaces to bridge. If mapping multiple interfaces for bridging the interface names are separated by a space when using UCI and package options. Example: option ifname 'eth2 eth3' or network.<if name>.ifname=eth2 eth 3

Table 51: Information table for the physical section on the common configuration page

19.4 Configuring WiFi using CLI

The configuration files are stored at:

- Network file **/etc/config/network**
- Wireless file **/etc/config/wireless**

19.4.1 AP modem on a new Ethernet interface using package options

```
root@VA_router:~# uci export network
package network
config interface 'newwifilan'
        option proto 'static'
        option ipaddr '192.168.111.1'
        option netmask '255.255.255.0'
root@VA_router:~# uci export wireless
package wireless
config wifi-device 'radio0'
        option type 'mac80211'
        option channel '11'
        option phy 'phy0'
        option hwmode '11ng'
        option htmode 'HT20'
```

```
list ht_capab 'SHORT-GI-40'
        list ht_capab 'TX-STBC'
        list ht_capab 'RX-STBC1'
        list ht_capab 'DSSS_CCK-40'
        option txpower '17'
        option country 'US'
config wifi-iface
        option device 'radio0'
        option mode 'ap'
        option disabled '1'
        option ssid 'Test_AP'
        option network 'newwifilan'
        option encryption 'psk'
        option key 'secretkey'
```

19.4.2 AP modem on a new Ethernet interface using UCI

```
root@VA_router:~# uci show network
network.newlan=interface
network.newlan.proto=static
network.newlan.ipaddr=192.168.111.1
network.newlan.netmask=255.255.255.0
root@VA_router:~# uci show wireless
wireless.radio0=wifi-device
wireless.radio0.type=mac80211
wireless.radio0.channel=11
wireless.radio0.phy=phy0
wireless.radio0.hwmode=11ng
wireless.radio0.htmode=HT20
wireless.radio0.ht_capab=SHORT-GI-40 TX-STBC RX-STBC1 DSSS_CCK-40
wireless.radio0.txpower=17
wireless.radio0.country=US
wireless.@wifi-iface[0]=wifi-iface
wireless.@wifi-iface[0].device=radio0
wireless.@wifi-iface[0].mode=ap
wireless.@wifi-iface[0].disabled=1
```

```
wireless.@wifi-iface[0].ssid=Test_AP
wireless.@wifi-iface[0].network=newlan
wireless.@wifi-iface[0].encryption=psk
wireless.@wifi-iface[0].key=secretkey
```

19.4.3 AP mode on an existing Ethernet interface using packages options

```
root@VA_router:~# uci export network
package network
config interface 'lan'
        option ifname 'eth0'
        option proto 'static'
        option ipaddr '192.168.100.1'
        option netmask '255.255.255.0'
        option type 'bridge'
root@VA_router:~# uci export wireless
package wireless
config wifi-device 'radio0'
        option type 'mac80211'
        option channel '11'
        option phy 'phy0'
        option hwmode '11ng'
        option htmode 'HT20'
        list ht_capab 'SHORT-GI-40'
        list ht_capab 'TX-STBC'
        list ht_capab 'RX-STBC1'
        list ht_capab 'DSSS_CCK-40'
        option txpower '17'
        option country 'US'

config wifi-iface
        option device 'radio0'
        option mode 'ap'
        option disabled '1'
        option ssid 'Test_AP'
        option network 'lan'
```

```
        option encryption 'psk'
        option key 'secretkey'
```

19.4.4 AP mode on an existing Ethernet interface using UCI

```
root@VA_router:~# uci show network
network.lan=interface
network.lan.ifname=eth0
network.lan.proto=static
network.lan.ipaddr=192.168.6.1
network.lan.netmask=255.255.255.0
network.lan.type=bridge
root@VA_router:~# uci show wireless
wireless.radio0=wifi-device
wireless.radio0.type=mac80211
wireless.radio0.channel=11
wireless.radio0.phy=phy0
wireless.radio0.hwmode=11ng
wireless.radio0.htmode=HT20
wireless.radio0.ht_capab=SHORT-GI-40 TX-STBC RX-STBC1 DSSS_CCK-40
wireless.radio0.txpower=17
wireless.radio0.country=US
wireless.@wifi-iface[0]=wifi-iface
wireless.@wifi-iface[0].device=radio0
wireless.@wifi-iface[0].mode=ap
wireless.@wifi-iface[0].disabled=1
wireless.@wifi-iface[0].ssid=Test_AP
wireless.@wifi-iface[0].network=lan
wireless.@wifi-iface[0].encryption=psk
wireless.@wifi-iface[0].key=secretkey
```

19.5 Creating a WiFi in client mode using the web interface

A WiFi network in client mode receives a wireless network from another WiFi AP.

Configure the Wifi network in Client mode as described in the above section ‘Configuring a WiFi interface’, selecting a new interface for the Wireless Network in the Interface Configuration section. For the examples below the new WiFi interface will be called ‘newwifiClient’

Example:

```
wireless.@wifi-iface[0].network=newwifiClient
wireless.@wifi-iface[0].mode=sta
```

In the top menu, select **Network -> Interfaces**. The Interfaces Overview page appears. Click **Edit** in the newly created WiFi Client interface. The Common Configuration page appears.

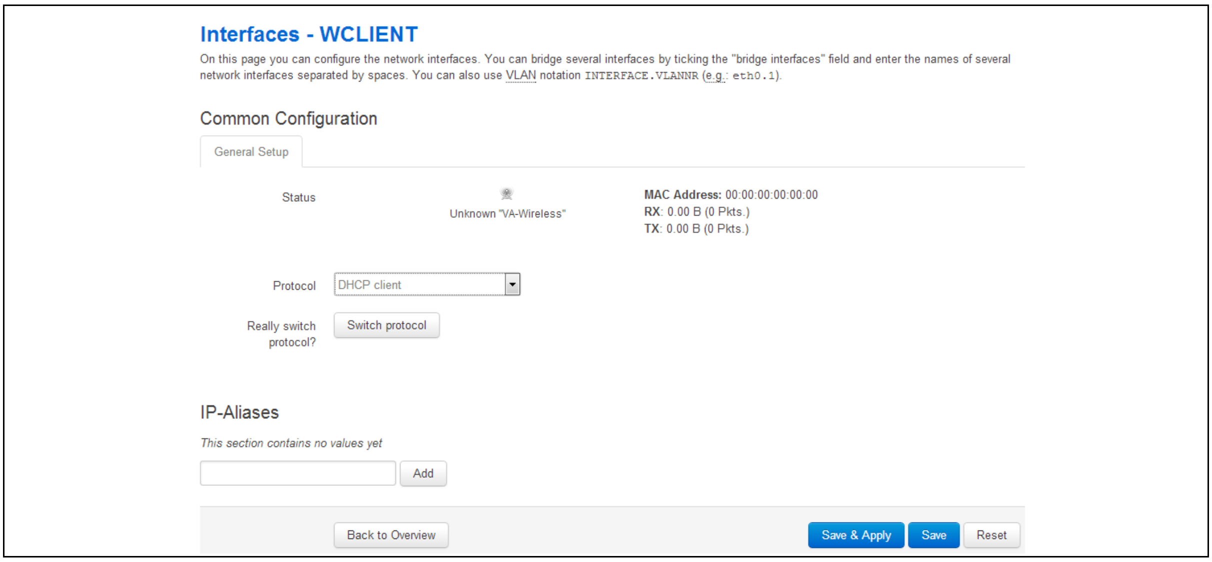


Figure 66: The client interface page

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Protocol
UCI: network. newwifiClient.proto
Opt: proto</td><td>Specifies what protocol the interface will operate on. Select DHCP Client.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>IPv6-in-IPv4 (RFC4213)</td><td>Used with tunnel brokers.</td></tr>
<tr><td>IPv6-over-IPv4</td><td>Stateless IPv6 over IPv4 transport.</td></tr>
<tr><td>GRE</td><td>Generic Routing Encapsulation protocol</td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol</td></tr>
<tr><td>PPP</td><td>Point to Point Protocol</td></tr>
<tr><td>PPPoE</td><td>PPP over Ethernet</td></tr>
</table></td></tr>
</table>

	PPPoATM	PPP over ATM
	LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.

Table 52: Information table for interfaces WClient page

When you have clicked **Save and Apply**, the router will restart the network package. It may take up to one minute for connectivity to the router to be restored.

19.6 Configuring WiFi in client mode using command line

The configuration files are stored at:

- Network file **/etc/config/network**
- Wireless file **/etc/config/wireless**

19.6.1 Client modem using package options

```
root@VA_router:~# uci export network
package network
config interface ' newwifiClient '
        option proto 'dhcp'
root@VA_router:~# uci export wireless
package wireless
config wifi-device 'radio0'
        option type 'mac80211'
        option channel '11'
        option phy 'phy0'
        option hwmode '11ng'
        option htmode 'HT20'
        list ht_capab 'SHORT-GI-40'
        list ht_capab 'TX-STBC'
        list ht_capab 'RX-STBC1'
        list ht_capab 'DSSS_CCK-40'
        option txpower '17'
        option country 'US'
```

```
config wifi-iface
        option device 'radio0'
        option ssid 'Remote-AP'
        option mode 'sta'
        option network ' newwifiClient '
        option encryption 'psk2'
        option key 'testtest'
```

19.6.2 Client modem using UCI

```
root@VA_router:~# uci show network
network.new=interface
network.WCLIENT.proto=dhcp
```

uci show wireless

```
root@VA_router:~# uci show wireless
wireless.radio0=wifi-device
wireless.radio0.type=mac80211
wireless.radio0.channel=11
wireless.radio0.phy=phy0
wireless.radio0.hwmode=11ng
wireless.radio0.htmode=HT20
wireless.radio0.ht_capab=SHORT-GI-40 TX-STBC RX-STBC1 DSSS_CCK-40
wireless.radio0.txpower=17
wireless.radio0.country=US
wireless.@wifi-iface[0]=wifi-iface
wireless.@wifi-iface[0].device=radio0
wireless.@wifi-iface[0].ssid=Remote-AP
wireless.@wifi-iface[0].mode=sta
wireless.@wifi-iface[0].network= newwifiClient
wireless.@wifi-iface[0].encryption=psk2
wireless.@wifi-iface[0].key=testtest
```

20 Configuring Multi-WAN

Multi-WAN is used for managing WAN interfaces on the router, for example, 3G interfaces to ensure high-availability. You can customise Multi-WAN for various needs, but its main use is to ensure WAN connectivity and provide a failover system in the event of failure or poor coverage.

20.1 Configuration package used

Package	Sections
multiwan	config wan

20.2 Configuring Multi-WAN using the web interface

In the top menu, select **Network -> Multi-Wan**. The Multi-WAN page appears.

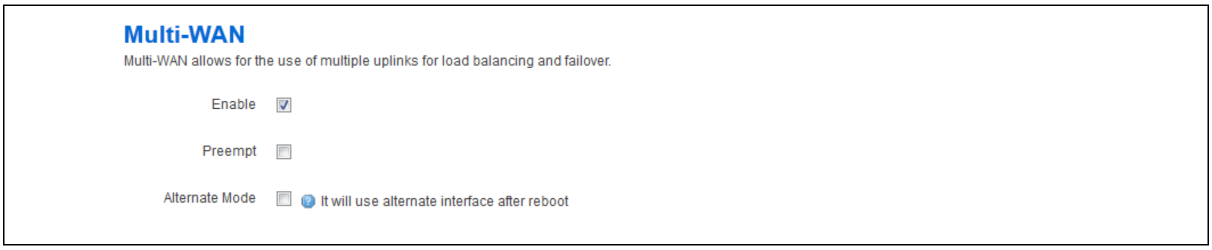


Figure 67: The multi-WAN page

Web Field / UCI / Package Option	Description
Web: Enable UCI: multiwan.config.enabled Opt: enabled	Enables or disables Multi-WAN. 0 Disabled. 1 Enabled.
Web: Preempt UCI: multiwan.config.preempt Opt: preempt	Enables or disables pre-emption for Multi-WAN. If enabled the router will keep trying to connect to a higher priority interface depending on timer set. 0 Disabled. 1 Enabled.
Web: Alternate Mode UCI: multiwan.config.alt_mode Opt: alt_mode	Enables or disables alternate mode for Multi-WAN. If enabled the router will use an alternate interface after reboot. 0 Disabled. 1 Enabled.

Table 53: Information table for multi-WAN page

When you have enabled Multi-WAN, you can add the interfaces that will be managed by Multi-WAN, for example 3G interfaces.

Note: the name used for Multi-WAN must be identical, including upper and lowercases, to the actual 3G interface name defined in your network configuration. To check the names and settings are correct, select **Network - > Interfaces** and view the Interfaces Overview page.

In the WAN interfaces section, enter the name of the WAN interface to configure, and then click **Add**. The new section for configuring specific parameters will appear.

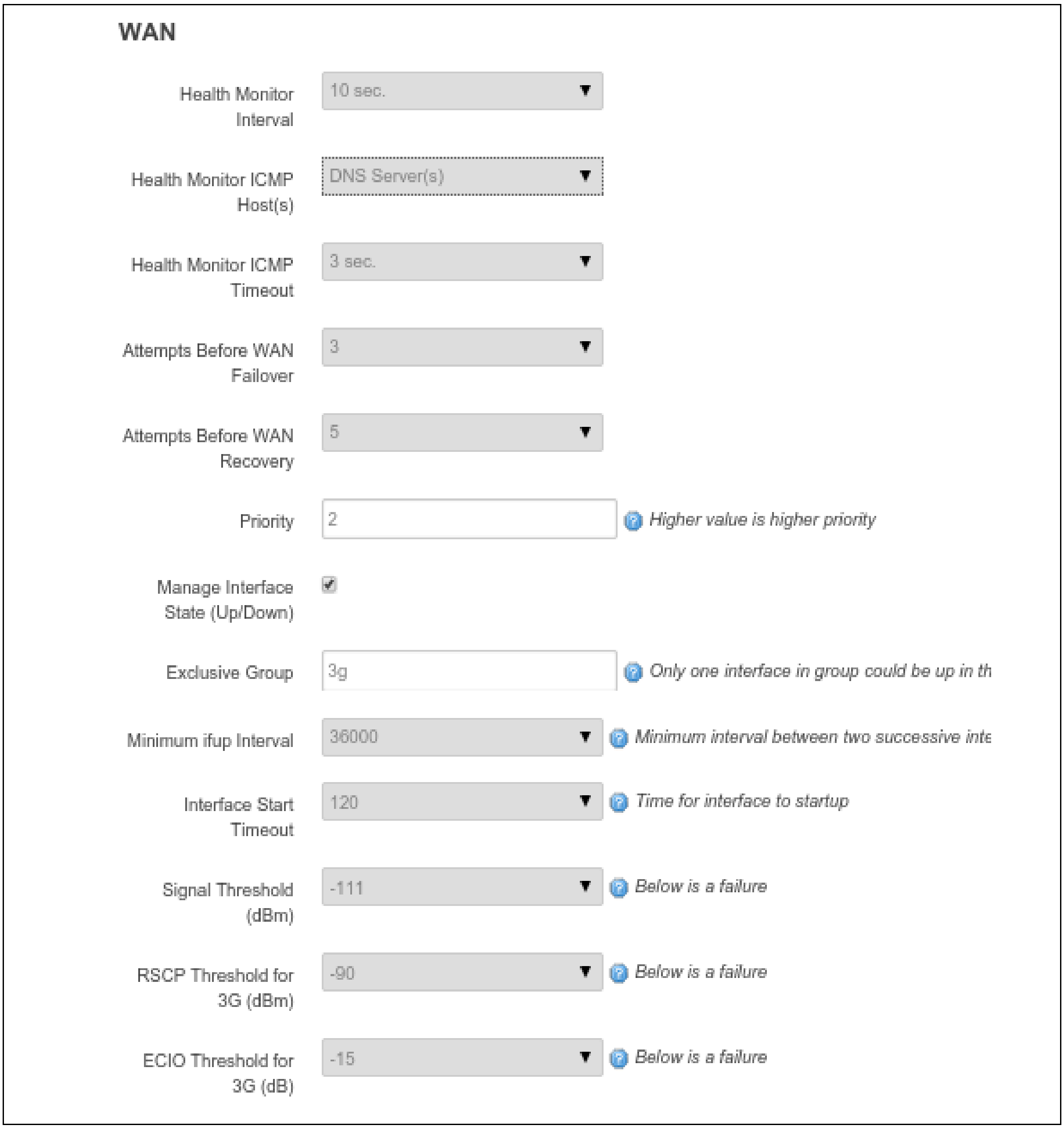


Figure 68: Example interface showing failover traffic destination as the added multi-WAN interface

Web Field/UCI/Package Option	Description
Web: Health Monitor Interval UCI: multiwan.wan.health_interval Opt: health _interval	Sets the period to check the health status of the interface.Choose the interval in seconds that will be used to monitor signal strength.
Web: Health Monitor ICMP Host(s) UCI: multiwan.wan.hosts Opt: icmp_hosts	Sends health ICMPs to configured value DNS servers by default. Configure to any address. Disable DNS servers WAN Gateway Custom
Web: Health Monitor ICMP Timeout UCI: multiwan.wan.timeout Opt: timeout	Sets Ping timeout in seconds. Choose the time in seconds that the health monitor ICMP will timeout at.
Web: Attempts Before WAN Failover UCI: multiwan.wan.health_fail_retries Opt: health_fail_retries	Sets the amount of retries before interface is considered a failure.
Web: Attempts Before WAN Recovery UCI: multiwan.wan.health_recovery_retries Opt: health_recovery_retries	Sets the number of healthy pings before the interface is considered healthy.
Web: Priority UCI: multiwan.wan.priority Opt: priority	Specifies the priority of the interface. The higher the value, the higher the priority.
Web: Manage Interface State (Up/Down) UCI: multiwan.wan.manage_state Opt: manage_state	Sets the interface start/stop by Multi-WAN.
Web: Exclusive Group UCI: multiwan.wan.exclusive_group Opt: exclusive_group	Defines the interface within the group, only one interface can be active: SIM 1 or SIM 2.
Web: Minimum ifup Interval UCI: multiwan.wan.ifup_retry_sec Opt: ifup_retry_sec	Specifieds the minimum interval between two successive interface start attempts.
Web: Interface Start Timeout UCI: multiwan.wan.start_timeout Opt: start_timeout	Specifies the time for interface to start up. If it is not up after this period, it will be considered a fail.
Web: Signal Threshold (dBm) UCI: multiwan.wan.signal_threshold Opt: signal_threshold	If signal is lower than this value, then it is marked as fail.
Web: RSCP Threshold (dBm) UCI: multiwan.wan.rscp_threshold Opt: rscp_threshold	Specifies the minimum RSCP signal strength before considering if the interface fails signal health check.

Web: ECIO Threshold (dBm) UCI: multiwan.wan.ecio_threshold Opt: ecip_threshold	Specifies the minimum ECIO signal strength before considering if the interface fails signal health check.

Table 54: Information table for multi-WAN interface page

20.2.1 Multi-WAN traffic rules

You can also set up traffic rules, to forward specific traffic out of the right WAN interface, based on source, destination address, protocol or port. This is useful to force traffic on specific interfaces when using multiple WAN interfaces simultaneously.

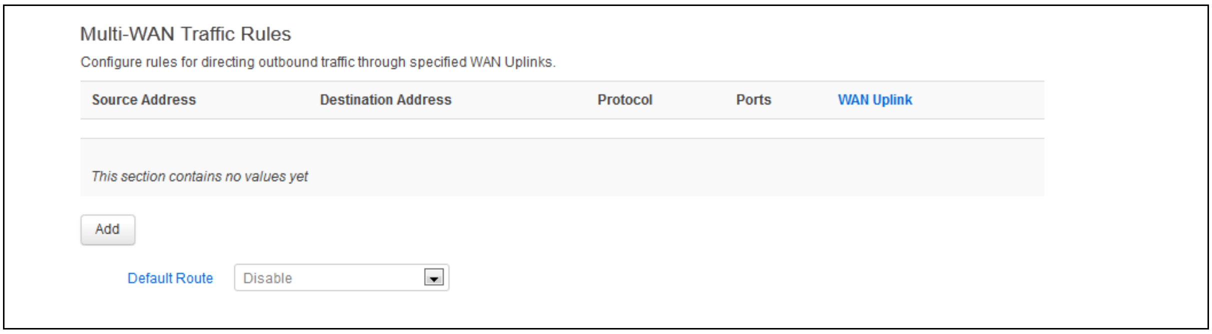


Figure 69: The multi-WAN traffic rules page

20.3 Configuring Multi-WAN using the UCI interface

Multi-WAN UCI configuration settings are stored in the following file:

/etc/config/multiwan

Run UCI export or show commands to see Multi-WAN UCI configuration settings. A sample is shown below.

```
~# uci export multiwan

package multiwan

config multiwan 'config'
        option preempt 'yes'
        option alt_mode 'no'
        option enabled 'yes'
config interface 'wan'
        option disabled '0'
```

```
        option health_interval '10'
        option timeout '3'
        option health_fail_retries '3'
        option health_recovery_retries '5'
        option priority '2'
        option manage_state 'yes'
        option exclusive_group '3g'
        option ifup_retry_sec '36000'
        option icmp_hosts 'disable'
        option signal_threshold '-111'
        option rscp_threshold '-90'
        option ecio_threshold '-15'
        option ifup_timeout_sec '120'

~# uci show multiwan
multiwan.config=multiwan
multiwan.config.preempt=yes
multiwan.config.alt_mode=no
multiwan.config.enabled=yes
multiwan.wan=interface
multiwan.wan.disabled=0
multiwan.wan.health_interval=10
multiwan.wan.timeout=3
multiwan.wan.health_fail_retries=3
multiwan.wan.health_recovery_retries=5
multiwan.wan.priority=2
multiwan.wan.manage_state=yes
multiwan.wan.exclusive_group=3g
multiwan.wan.ifup_retry_sec=36000
multiwan.wan.icmp_hosts=disable
multiwan.wan.signal_threshold=-111
multiwan.wan.rscp_threshold=-90
multiwan.wan.ecio_threshold=-15
```

20.4 Multi-WAN diagnostics

The multi-WAN package is an agent script that makes multi-WAN configuration simple, easy to use and manageable. It comes complete with load balancing, failover and an easy to manage traffic ruleset. The uci configuration file/etc/config/multiwan is provided as part of the multi-WAN package.

The multi-WAN package is linked to the network interfaces within /etc/config/network.

Note: multi-WAN will not work if the WAN connections are on the same subnet and share the same default gateway.

To view the multi-WAN package, enter:

```
root@VA_router:~# uci export /etc/config/multiwan
package multiwan

config multiwan 'config'
        option enabled 'yes'
        option preempt 'yes'
        option alt_mode 'no'

config interface 'ADSL'
        option health_interval '10'
        option icmp_hosts 'dns'
        option timeout '3'
        option health_fail_retries '3'
        option health_recovery_retries '5'
        option priority '1'
        option manage_state 'yes'
        option exclusive_group '0'
        option ifup_retry_sec '300'
        option ifup_timeout_sec '40'

config interface 'Ethernet'
        option health_interval '10'
        option icmp_hosts 'dns'
        option timeout '3'
```

```
        option health_fail_retries '3'
        option health_recovery_retries '5'

        option priority '2'
        option manage_state 'yes'
        option exclusive_group '0'
        option ifup_retry_sec '300'
        option ifup_timeout_sec '40'
```

The following output shows the multi-WAN standard stop/start commands for troubleshooting.

```
root@VA_router:~# /etc/init.d/multiwan
Syntax: /etc/init.d/multiwan [command]
```

Available commands:

```
        start   Start the service
        stop    Stop the service
        restart Restart the service
        reload  Reload configuration files (or restart if that fails)
        enable  Enable service autostart
        disable Disable service autostart
```

When troubleshooting, make sure that the routing table is correct using `route –n`.

Ensure all parameters in the multi-WAN package are correct. The name used for multi-WAN interfaces must be identical, including upper and lowercases, to the interface name defined in the network configuration.

To check the names and settings are correct, browse to **Network - > interfaces** (or alternatively, run: **cat/etc/config/network** through CLI).

Enter the name of the WAN interface to configure, and then click **Add**. The new section for configuring specific parameters will appear.

21 Automatic operator selection

This section describes how to configure and operate the Automatic Operator Selection feature of a Virtual Access router.

When the roaming SIM is connected, the radio module has the ability to scan available networks. The router, using mobile and multi-WAN packages, finds available networks to create and sort interfaces according to their signal strength. These interfaces are used for failover purposes.

21.1 Configuration package used

Package	Sections
Multiwan	General, interfaces
Mobile	Main, Template interface
Network	2G/3G/4G interface

21.2 Configuring automatic operator selection via the web interface

While the router boots up it checks for mobile networks. Based on available networks, the router creates interfaces and the multi-WAN package is used to run failover between interfaces. Details for these interfaces are provided in the mobile package. When you have created the interfaces, multi-WAN manages the operation of primary (predefined) and failover (auto created) interfaces.

There are three PMP (Primary Mobile Provider) scenarios:

1. PMP + roaming: pre-empt enabled
2. PMP + roaming: pre-empt disabled
3. No PMP + roaming

21.3 Scenario 1: PMP + roaming: pre-empt enabled

In this scenario, the primary interface is used whenever possible. If there is no PMP defined, go straight to section 1.6 ‘No PMP + roaming’.

Software operations

1. Connect the PMP interface.
2. Wait until the signal level on the PMP interface goes under sig_dbm option value.
3. Disconnect the PMP interface.
4. Connect the first auto-generated interface.
5. Wait until the signal level on the first auto-generated interface goes under the sig_dbm option in the mobile package, or until the primary interface is available to connect after it was disconnected in step 3. ifup_retry_sec option value of primary interface in multi-WAN package identifies retry timer.

6. Disconnect auto-generated interface. If the interface was disconnected due to low signal level then connect the next auto-generated interface and repeat step 5. If the interface was disconnected because ifup_retry_sec of Primary interface timed out then go back to step 1 and repeat the process.

The primary predefined interface is defined in the network package. Ensure the interface name matches the interface name defined in the multi-WAN package.

21.3.1 Create a primary predefined interface

In the web interface top menu, go to **Network -> Interfaces**. The Interfaces page appears.

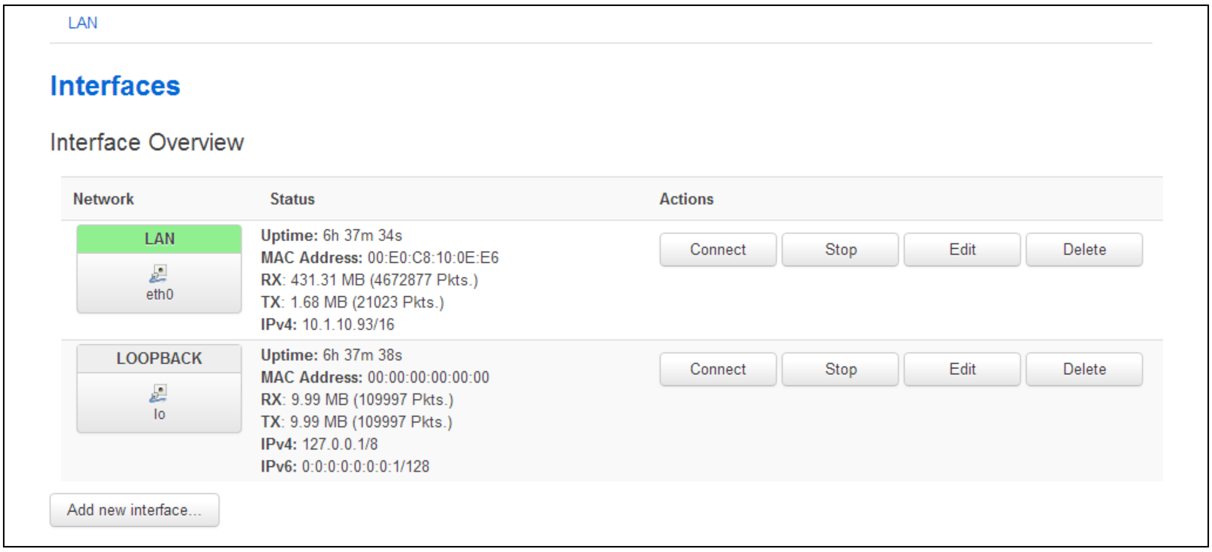


Figure 70: The interface overview page

Click **Add new interface...** The Create Interface page appears.

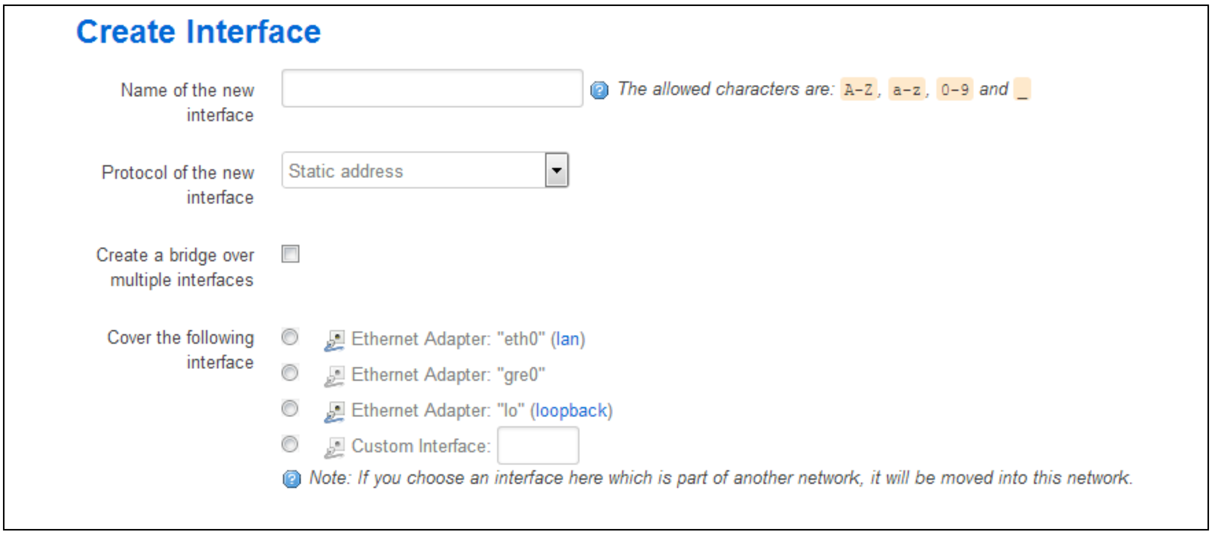


Figure 71: The create interface page

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Name of the new interface
UCI: network.3g_s<sim-number>_<short-operator-name>.
Opt: 3g_s<sim-number>_<short-operator-name>.</td><td>Type the name of the new interface.
Type the interface name in following format:
3g_s<sim-number>_<short-operator-name>. Where <sim-number> is number of roaming SIM (1 or 2) and <short-operator-name> is first four alphanumeric characters of operator name (as reported by 'AT+COPS=?' command).
Type the short operator name in lower case, for example:
<table>
<tr><th>Operator name</th><th>First four alphanumeric numbers</th></tr>
<tr><td>Vodafone UK</td><td>voda</td></tr>
<tr><td>O2 – UK</td><td>o2uk</td></tr>
<tr><td>Orange</td><td>oran</td></tr>
</table></td></tr>
<tr><td>Web: Protocol of the new interface
UCI: network.[..x..].proto
Opt: proto</td><td>Protocol type. Select LTE / UMTS / GPRS / EV-DO.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>IPv6-in-IPv4 (RFC4213)</td><td>IPv4 tunnels that carry IPv6.</td></tr>
<tr><td>IPv6 over IPv4</td><td>IPv6 over IPv4 tunnel.</td></tr>
<tr><td>GRE</td><td>Generic Routing Encapsulation.</td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol.</td></tr>
<tr><td>PPP</td><td>Point to Point Protocol.</td></tr>
<tr><td>PPPoE</td><td>Point to Point Protocol over Ethernet.</td></tr>
<tr><td>PPPoATM</td><td>Point to Point Protocol over ATM.</td></tr>
<tr><td>LTE/UMTS/GPRS/EV-DO</td><td>CDMA, UMTS or GPRS connection using an AT-style 3G modem.</td></tr>
</table></td></tr>
<tr><td>Web: Create a bridge over multiple interfaces
UCI: network.[..x..].typeOpt: type</td><td>Enables bridge between two interfaces.
<table>
<tr><td>0</td><td>Disabled.</td></tr>
<tr><td>1</td><td>Enabled</td></tr>
</table></td></tr>
<tr><td>Web: Cover the following interface
UCI: network.[..x..].ifname
Opt: ifname</td><td>Select interfaces for bridge connection.</td></tr>
</table>

Table 55: Information table for the create interface page

Click **Submit**. The Common Configuration page appears.

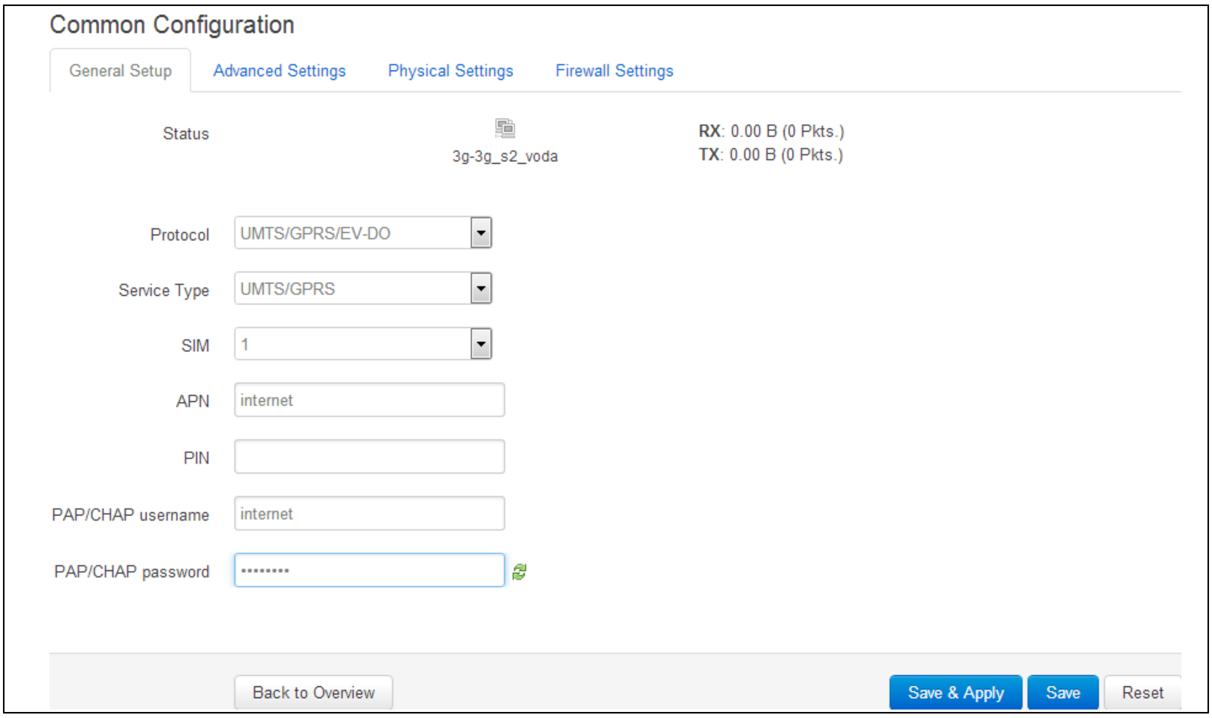


Figure 72: The common configuration page

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Protocol
UCI: network.[..x..].proto
Opt: proto</td><td>Protocol type. Select LTE / UMTS / GPRS / EV-DO.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>IPv6-in-IPv4 (RFC4213)</td><td>IPv4 tunnels that carry IPv6.</td></tr>
<tr><td>IPv6 over IPv4</td><td>IPv6 over IPv4 tunnel.</td></tr>
<tr><td>GRE</td><td>Generic Routing Encapsulation.</td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol.</td></tr>
<tr><td>PPP</td><td>Point to Point Protocol.</td></tr>
<tr><td>PPPoE</td><td>Point to Point Protocol over Ethernet.</td></tr>
<tr><td>PPPoATM</td><td>Point to Point Protocol over ATM.</td></tr>
<tr><td>LTE/UMTS/GPRS/EV-DO</td><td>CDMA, UMTS or GPRS connection using an AT-style 3G modem.</td></tr>
</table></td></tr>
</table>

Web: Service Type UCI: network.[..x..].service Opt: service	Service type that will be used to connect to the network. gprs_only: Allows GSM module to only connect to GPRS network. lte_only: Allows GSM module to only connect to LTE network. cdma: Allows GSM module to only connect to CDMA network. auto: GSM module will automatically detect the best available technology code.
Web: SIM UCI: network.[..x..].sim Opt: sim	Select SIM 1 or SIM 2. auto: Automatically detects which SIM slot is used. SIM 1: Selects Sim from slot 1. SIM 2: Selects Sim from slot 2.
Web: APN UCI: [X] Opt: [X]	APN name of Mobile Network Operator.
Web: APN username UCI: [X] Opt: [X]	Username used to connect to APN.
Web: APN password UCI: [X] Opt: [X]	Password used to connect to APN.
Web: Modem Configuration UCI: N/A Opt: N/A	Click the link if you need to configure additional options from Mobile Manager.

Table 56: Information table for the general set up section

Click **Save & Apply**.

21.3.2 Set multi-WAN options for primary predefined interface

On the web interface go to **Network ->Multi-Wan**. The Multi-WAN page appears.

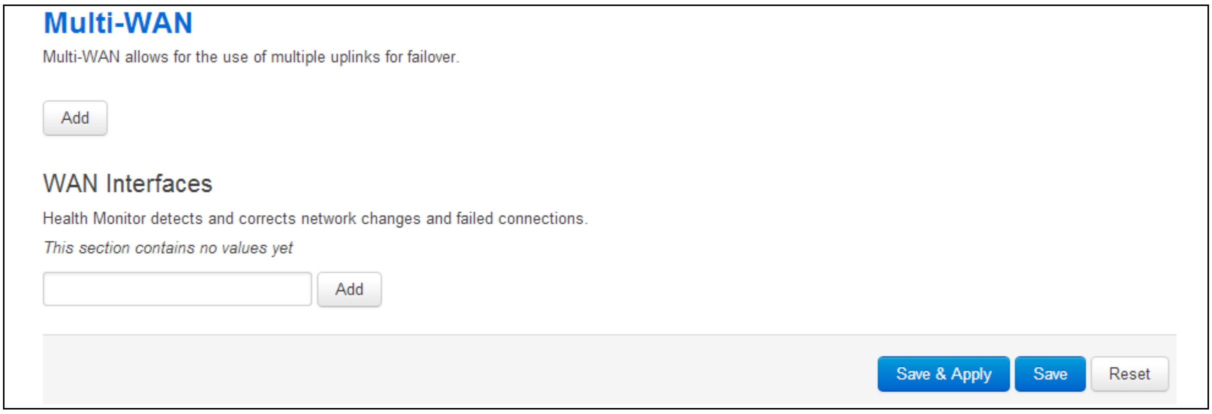


Figure 73: The multi-WAN page

In the WAN Interfaces section, type in the name of the Multi-WAN interface. Click **Add**. The Multi-WAN page appears.

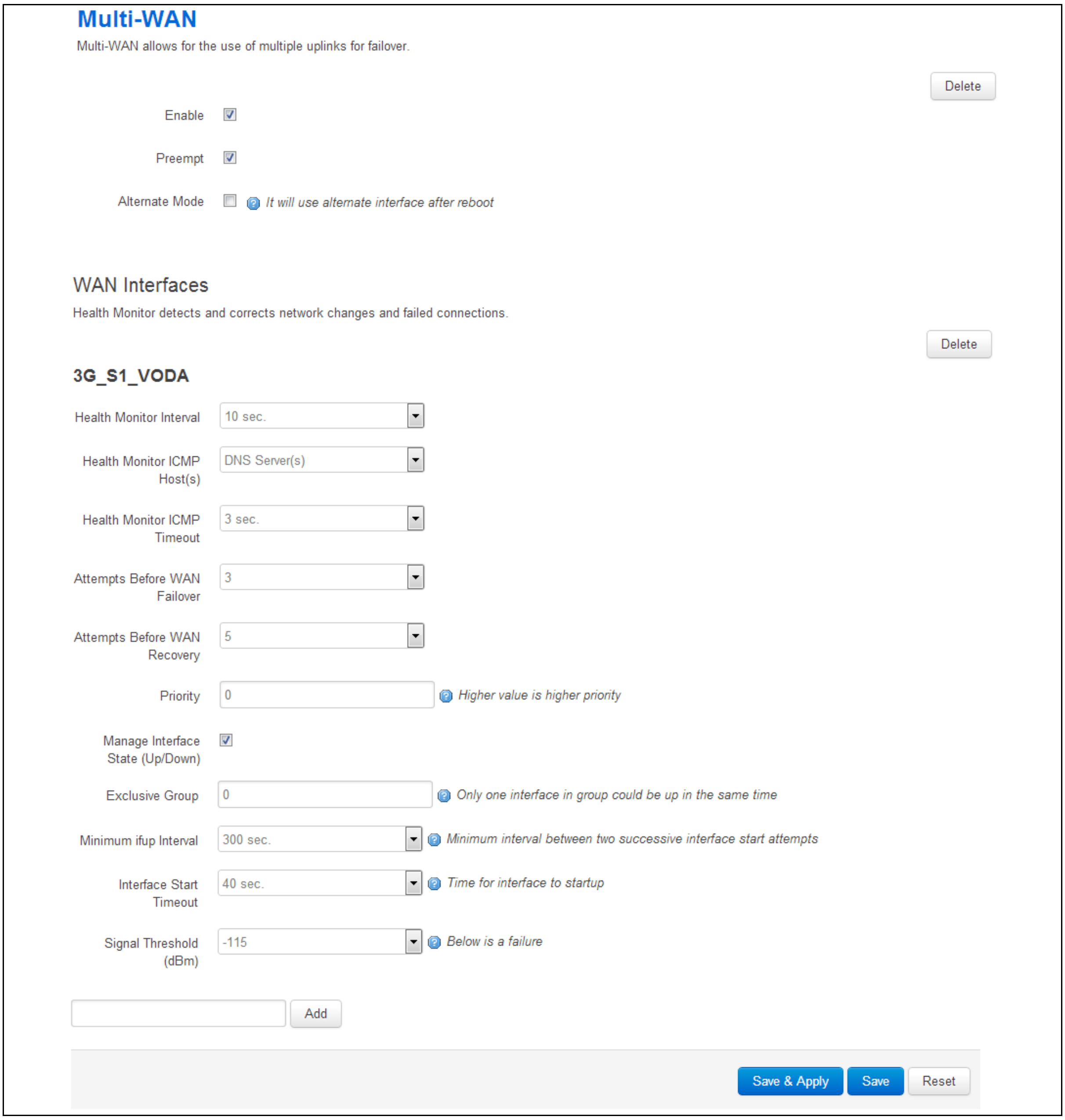


Figure 74: The multi-WAN page

Web Field / UCI / Package Option	Description
Web: Enable UCI: multiwan.config.enabled Opt: enabled	Enables Multi-WAN. 0 — Disabled. 1 — Enabled.
Web: Preempt UCI: multiwan.config.preempt Opt: preempt	Enables Preempt mode. Select this option. 0 — Disabled. 1 — Enabled.

<table>
<tr><td>Web: Alternate Mode
UCI: multiwan.config.alt
Opt: alt</td><td>Alternate interface will be used after reboot.</td></tr>
<tr><td>Web: WAN Interfaces
UCI: multiwan.3g_s<sim-number>_<short-operator-name>
Opt: 3g_s<sim-number>_<short-operator-name></td><td>Provide the same interface name as chosen in Multi-WAN section below and click Add.</td></tr>
<tr><td>Web: Health Monitor Interval
UCI: multiwan.[..x..].health_interval
Opt: health_interval</td><td>Interval used to monitor signal strength. Choose the interval in seconds that will be used to monitor signal strength.</td></tr>
<tr><td>Web: Health Monitor ICMP Host(s)
UCI: multiwan.[..x..].icmp_hosts
Opt: icmp_hosts</td><td>Specifies target IP address for ICMP packets.<table><tr><td>Disable</td><td>Disables the option.</td></tr><tr><td>DNS servers</td><td>DNS IP addresses will be used.</td></tr><tr><td>WAN Gateway</td><td>Gateway IP address will be used.</td></tr><tr><td>custom</td><td>Ability to provide IP address.</td></tr></table></td></tr>
<tr><td>Web: Health Monitor ICMP Timeout
UCI: multiwan.[..x..].timeout
Opt: timeout</td><td>Choose the time in seconds that the health monitor ICMP will timeout at.</td></tr>
<tr><td>Web: Attempts Before WAN Failover
UCI: multiwan.health_fail_retries
Opt: health_fail_retries</td><td>Number of fail attempts of health monitor before interface is disconnected.
Select the number of fail attempts of health monitor checks that will cause the interface to be disconnected.</td></tr>
<tr><td>Web: Attempts Before WAN Recovery
UCI: multiwan. health_recovery_retries
Opt: health_recovery_retries</td><td>Select the number of fail attempts, in seconds, of health monitor checks that will cause the interface to be disconnected.</td></tr>
<tr><td>Web: Priority
UCI: multiwan.[..x..].priority
Opt: priority</td><td>Type the priority number. The higher the value, the higher the priority.
This multi-WAN interface priority must be higher than the one specified in the priority field in the ‘Roaming Interface Template’ page described in the following section.<table><tr><td>0</td><td></td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
<tr><td>Web: Exclusive Group
UCI: multiwan.[..x..].exclusive_group
Opt: exclusive_group</td><td>Only one interface in group could be up in the same time.
For this scenario type 3G.<table><tr><td>0</td><td></td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
<tr><td>Web: Manage Interface State (Up/Down)
UCI: [x]
Opt: [x]</td><td>Select Enabled.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
<tr><td>Web: Minimum ifup Interval
UCI: multiwan.[..x..]ifup_retry_sec
Opt: ifup_retry_sec</td><td>Minimum interval between two successive interface start attempts.</td></tr>
<tr><td>Web: Interface Start Timeout
UCI: multiwan.[..x..]ifup_timeout
Opt: ifup_timeout</td><td>Time allowed for interface to start up.
Choose timer greater than 120 seconds.</td></tr>
</table>

Web: Signal Threshold (dBm) UCI: multiwan.[..x..].signal_threshold Opt: signal_threshold	If signal is lower than this value, then it is marked as fail. Range: -46 to -120 dBm -115dBm

Table 57: Information table for Multi-WAN page

Click **Save**.

21.3.3 Set options for automatically created interfaces (failover)

From the top menu on the web interface page, select **Services ->Mobile Manager**. The Mobile Manager page appears.

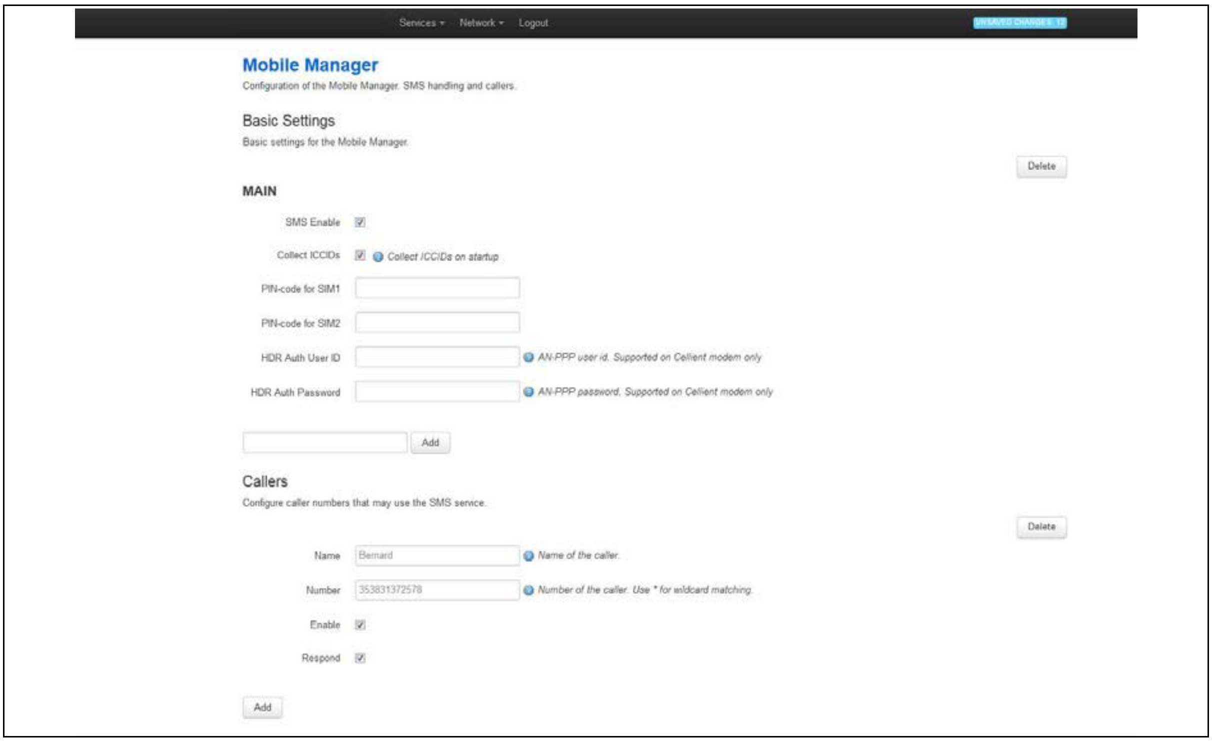


Figure 75: The mobile manager page

There are three sections in Mobile Manager.

Basic settings	Configure SMS, select roaming SIM and collect ICCIDs.
Callers	Configure callers that can use SMS.
Roaming Interface Template	Configure common values for interface created by Automatic Operator Selection.

21.3.3.1 Basic settings

Web Field / UCI / Package Option	Description
Web: SMS Enable UCI: mobile.main.sms Opt: sms	Enables SMS. no: Disabled. yes: Enabled.

Web: Collect ICCIDs UCI: mobile.main.init_get_iccids Opt: init_get_iccids	Enables or disables integrated circuit card identifier ICCID's collection functionality. If enabled then both SIM 1 and SIM 2 ICCIDs will be collected, otherwise it will default to SIM 1. This will be display under mobile stats no: Disabled. yes: Enabled.
Web: PIN code for SIM1 UCI: mobile.main.sim2pin Opt: sim2pin	Depending on the SIM card, specify the PIN code for SIM 1.
Web: PIN code for SIM2 UCI: mobile.main.sim2pin Opt: sim2pin	Depending on the SIM card, specify the PIN code for SIM 2.
Web: HDR Auto User ID UCI: mobile.main.hdr_userid Opt: hdr_userid	AN-PPP user ID. Supported on Cellient (CDMA) modem only.

Table 58: Information table for mobile manager basic settings

21.3.3.2 Caller settings

Web: Name UCI: mobile.@caller[0].name Opt: name	Name assigned to the caller.
Web: Number UCI: mobile.@caller[0].number Opt: number	Number of the caller allowed to SMS the router. Add in specific caller numbers, or use the wildcard symbol.
Web: Enable UCI: mobile.@caller[0].enabled Opt: enabled	Enables or disables incoming caller ID. 0: Disabled. 1: Enabled.
Web: Respond UCI: mobile.@caller[0].respond Opt: respond	If checked, the router will return an SMS. Select **Respond** if you want the router to reply. 0: Disabled. 1: Enabled.

Table 59: Information table for caller settings

21.3.3.3 Roaming interface template

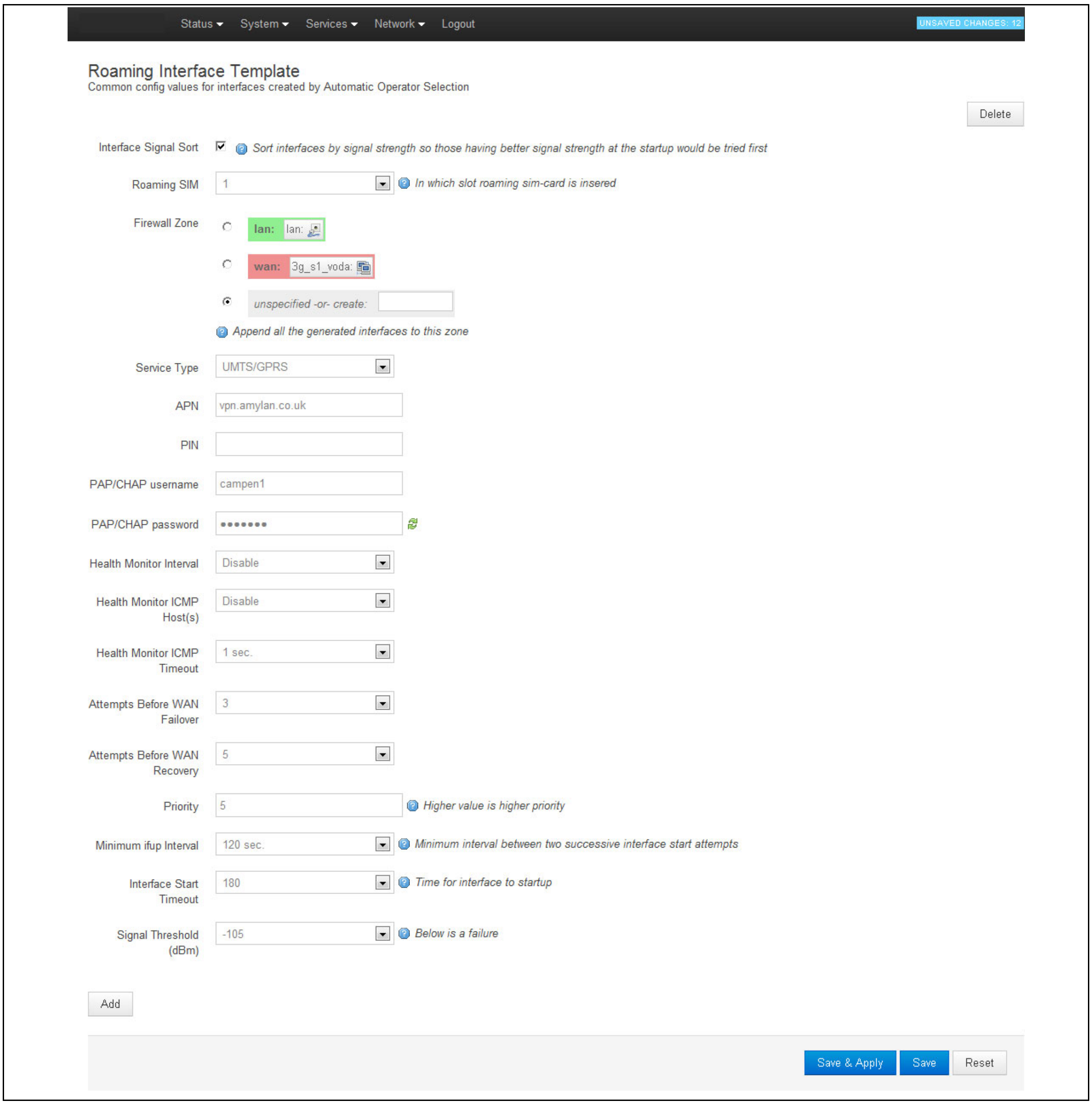


Figure 76: The roaming interface template page

Web Field / UCI / Package Option	Description
Web: Interface Signal Sort UCI: mobile.@roaming_template[0].sort_sig_strength Opt: sort_sig_strength	Sorts interfaces by signal strength priority so those that have a better signal strength will be tried first. 0 – Disabled. 1 – Enabled.
Web: Roaming SIM UCI: mobile.main.roaming_sim Opt: roaming_sim	Sets in which slot to insert roaming SIM card. 1 – SIM slot 1. 2 – SIM slot 2.

<table>
<tr><td>Web: Firewall Zone
UCI: mobile.@roaming_template[0].firewall_zone
Opt: firewall_zone</td><td>Adds all generated interfaces to this zone. Select existing zone or click unspecified or create to create new zone.</td></tr>
<tr><td>Web: Service Type
UCI: mobile.@roaming_template[0].service
Opt: service</td><td>Specifies the service type that will be used to connect to the network.<table><tr><td>UMTS/GPRS</td><td>GSM module will automatically detect the best available technology code.</td></tr><tr><td>Umts_only</td><td>Allows GSM module to only connect to 3G network.</td></tr><tr><td>GPRS_only</td><td>Allows GSM module to only connect to GPRS network.</td></tr><tr><td>cdma</td><td>Allows GSM module to only connect to cdma network.</td></tr></table></td></tr>
<tr><td>Web: APN
UCI: mobile.@roaming_template[0].apn
Opt: apn</td><td>APN name of Mobile Network Operator.</td></tr>
<tr><td>Web: PIN
UCI: mobile.@roaming_template[0].pincode
Opt: pincode</td><td>SIM Card's PIN number.</td></tr>
<tr><td>Web: PAP/CHAP username
UCI: mobile.@roaming_template[0].username
Opt: username</td><td>Username used to connect to APN.</td></tr>
<tr><td>Web: PAP/CHAP password
UCI: mobile.@roaming_template[0].password
Opt: password</td><td>Password used to connect to APN.</td></tr>
<tr><td>Web: Health Monitor Interval
UCI: mobile.@roaming_template[0].health_interval
Opt: health_interval</td><td>Sets the interval used to monitor signal strength in seconds.</td></tr>
<tr><td>Web: Health Monitor ICMP Host(s)
UCI: mobile.@roaming_template[0].icmp_hosts
Opt: icmp_hosts</td><td>Specifies target IP address for ICMP packets.<table><tr><td>DNS servers</td><td>IP address of DNS servers.</td></tr><tr><td>WAN gateway</td><td>IP address of Gateway.</td></tr><tr><td>custom</td><td>Custom Interface IP address.</td></tr></table></td></tr>
<tr><td>Web: Health Monitor ICMP Timeout
UCI: mobile.@roaming_template[0].timeout
Opt: timeout</td><td>Specifies the time in seconds that Health Monitor ICMP will timeout at.</td></tr>
</table>

Web: Attempts Before WAN Failover UCI: mobile.@roaming_template[1].health_fail_retries Opt: health_fail_retries	Number of fail attempts of health monitor before interface is disconnected. Select the number of fail attempts of health monitor checks that will cause the interface to be disconnected. 3 Range
Web: Attempts Before WAN Recovery UCI: mobile.@roaming_template[0].health_recovery_retries Opt: health_recovery_retries	Select the number of fail attempts of health monitor checks that will cause the interface to be disconnected. 5 Range
Web: Priority UCI: mobile.@roaming_template[0].priority Opt: priority	Type the priority number. The higher the value, the higher the priority. This multi-WAN interface priority must be higher than the one specified in the priority field in the 'Roaming Interface Template' page described in the following section. 0 Range
Web: Minimum ifup interval UCI: Opt:	Minimum interval between two successive interface start attempts.
Web: Interface Start Timeout UCI: mobile.@roaming_template[0].ifup_timeout_sec Opt: ifup_timeout	Time allowed for interface to start up. Set a value greater than 120 seconds. 40 Range
Web: Signal Threshold (dBm) UCI: mobile.@roaming_template[0].signal_threshold Opt: signal_threshold	If signal is lower than this value, then it is marked as fail. Range: -46 to -120 dBm -115dBm

Table 60: Information table for roaming interface template

When you have configured your settings, click **Save & Apply**.

In the top menu, select **System -> Reboot**. The System page appears.

Figure 77: The reboot page

Check the **Reboot now** check box and then click **Reboot**.

21.4 Scenario 2: PMP + roaming: pre-empt disabled

As in the previous section, multi-WAN connects the primary predefined interface and uses auto created interfaces. However, in this scenario, the auto-created interface will not be disconnected as soon as the primary interface is available. The primary interface will be reconnected when auto-created interface is down and when the ifup_retry_sec timeout expires.

Follow the instruction in the section above. The only change in configuration compared to the PMP + roaming: pre-empt enabled, is that you must disable the pre-empt option in the multi-WAN package.

21.4.1 Set multi-WAN options for pre-empt disabled

To disable PMP + roaming pre-empt, in the top menu, select **Network -> Multi-Wan**.

In the Multi-WAN page, ensure Preempt is **not** selected.

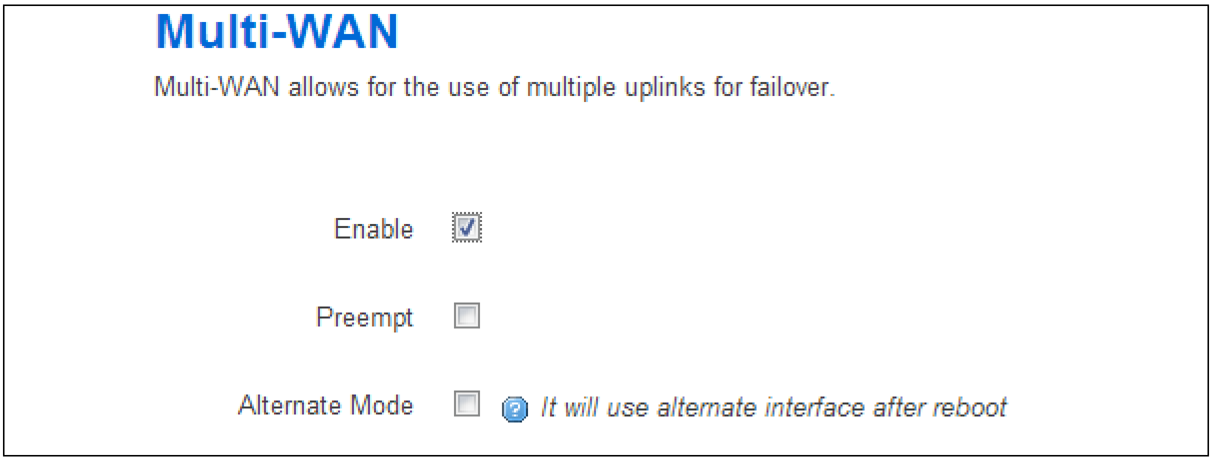


Figure 78: The multi-wan page, pre-empt not selected

Click **Save & Apply**.

In the top menu, select **System -> Reboot**. The System Reboot page appears.

Figure 79: The system reboot page

Check the **Reboot now** check box and then click **Reboot**.

21.5 Configure PMP + roaming: pre-empt enabled & disabled via UCI

Network file /etc/config/network

To view the configuration file, enter:

```
uci export network
package network

config interface 'loopback'
        option ifname 'lo'
        option proto 'static'
        option ipaddr '127.0.0.1'
        option netmask '255.0.0.0'

config interface 'lan'
        option ifname 'eth0'
        option proto 'static'
        option ipaddr '192.168.100.1'
        option netmask '255.255.255.0'

config interface 'main_voda'
        option auto '0'
        option proto '3g'
        option service 'umts'
        option apn 'testIE'
        option username 'test'
        option password 'test'
        option sim '1'
option operator 'vodafone IE'
```

To view uci commands, enter:

```
uci show network
network.loopback=interface
network.loopback.ifname=lo
network.loopback.proto=static
network.loopback.ipaddr=127.0.0.1
network.loopback.netmask=255.0.0.0
network.lan=interface
network.lan.ifname=eth0
network.lan.proto=static
network.lan.ipaddr=192.168.100.1
network.lan.netmask=255.255.255.0
network.main_voda=interface
network.main_voda.auto=0
network.main_voda.proto=3g
network.main_voda.service=umts
network.main_voda.apn=test IE
network.main_voda.username=test
network.main_voda.password=test
network.main_voda.sim=1
network.main_voda.operator=vodafone IE
```

package mobile configuration file is stored at:

/etc/config/mobile

```
config mobile 'main'
        option sms 'yes'
        option roaming_sim '1'
        option init_get_iccids 'no'
config caller
        option name 'Test'
        option number '*'
        option enabled 'yes'
```

```
        option respond 'yes'
config roaming_template
        option roaming_sim '1'
        option firewall_zone 'wan'
        option apn 'test IE'
        option username 'test'
        option password 'test'
        option service 'umts'
        option health_interval '4'
        option icmp_hosts 'disable'
        option timeout 'disable'
        option health_fail_retries '3'
        option signal_threshold '-95'
        option priority '5'
        option ifup_retry_sec '120'
        option ifup_timeout_sec '180'
        option defaultroute 'yes'
        option sort_sig_strength 'yes'
```

To view the uci command of package mobile, enter:

```
mobile.main=mobile
mobile.main.sms=yes
mobile.main.roaming_sim=1
mobile.main.init_get_iccids=no
mobile.@caller[0]=caller
mobile.@caller[0].name=Test
mobile.@caller[0].number=*
mobile.@caller[0].enabled=yes
mobile.@caller[0].respond=yes
mobile.@roaming_template[0]=roaming_template
mobile.@roaming_template[0].roaming_sim=1
mobile.@roaming_template[0].firewall_zone=wan
mobile.@roaming_template[0].apn=test IE
```

```
mobile.@roaming_template[0].username=test
mobile.@roaming_template[0].password=test
mobile.@roaming_template[0].service=umts
mobile.@roaming_template[0].health_interval=4
mobile.@roaming_template[0].icmp_hosts=disable
mobile.@roaming_template[0].timeout=disable
mobile.@roaming_template[0].health_fail_retries=3
mobile.@roaming_template[0].signal_threshold=-95
mobile.@roaming_template[0].priority=5
mobile.@roaming_template[0].ifup_retry_sec=120
mobile.@roaming_template[0].ifup_timeout_sec=180
mobile.@roaming_template[0].defaultroute=yes
mobile.@roaming_template[0].sort_sig_strength=yes
```

The configuration file for package multiwan is stored at

/etc/config/multiwan

To see configuration file of mobile package, enter:

```
config multiwan 'config'
        option enabled '1'
        option preempt '1'

config interface 'main_voda'
        option health_fail_retries '3'
        option health_interval '3'
        option timeout '1'
        option icmp_hosts 'disable'
        option priority '10'
        option exclusive_group '3g'
        option signal_threshold '-95'
        option ifup_retry_sec '350'
        option ifup_timeout_sec '180'
        option manage_state '1'
```

The configuration file for package multiwan is stored at:

/etc/config/multiwan

To see the content of the package, enter:

```
uci export multiwan


multiwan.config=multiwan
multiwan.config.enabled=1
multiwan.config.preempt=1
multiwan.main_voda=interface
multiwan.main_voda.health_fail_retries=3
multiwan.main_voda.health_interval=3
multiwan.main_voda.timeout=1
multiwan.main_voda.icmp_hosts=disable
multiwan.main_voda.priority=10
multiwan.main_voda.exclusive_group=3g
multiwan.main_voda.signal_threshold=-95
multiwan.main_voda.ifup_retry_sec=350
multiwan.main_voda.ifup_timeout_sec=180
multiwan.main_voda.manage_state=1
```

The difference between PMP + roaming: pre-empt enabled and disabled is setting one option parameter. To disable pre-empt, enter:

```
uci set multiwan.config.preempt=0
uci commit
```

Note: available values are:

0	Disabled
1	Enabled

21.6 Scenario 3: No PMP + roaming

In this scenario there is no primary interface that can be used for a connection. The router uses the network that offers the best signal threshold.

The logic is as follows:

1. Connect to the first roaming operator interface.
2. Check for signal strength every 'health_interval'. If the signal goes down below 'signal_threshold'
3. Disconnect from first roaming interface

4. Connect to second roaming operator interface.
5. Check for signal strength every 'health_interval'. Stays there until signal goes below 'signal_threshold'
6. Disconnect from second roaming interface.

21.6.1 Set options for automatically created interfaces (failover)

In the top menu on the web interface page, select **Services ->Mobile Manager**. The Mobile Manager page appears.

There are three sections:

Basic settings	Configure SMS, select roaming SIM and collect ICCCIDs
Callers	Configure callers that can use SMS.
Roaming Interface Template	Configure common values for interface created by Automatic Operator Selection.

21.6.1.1 Basic settings

Web Field / UCI / Package Option	Description
Web: SMS Enable UCI: mobile.main.sms Opt: sms	Enables SMS. no: Disabled. yes: Enabled.
Web: Collect ICCIDs UCI: mobile.main.init_get_iccids Opt: init_get_iccids	Enables or disables integrated circuit card identifier ICCID's collection functionality. If enabled then both SIM 1 and SIM 2 ICCID's will be collected otherwise it will default to SIM 1. This will be display under mobile stats. no: Disabled. yes: Enabled.
Web: PIN code for SIM1 UCI: mobile.main.sim2pin Opt: sim2pin	Depending on the SIM card specify the pin code for SIM 1. blank range
Web: PIN code for SIM2 UCI: mobile.main.sim2pin Opt: sim2pin	Depending on the SIM card specify the pin code for SIM 2. blank range
Web: HDR Auto User ID UCI: mobile.main.hdr_userid Opt: hdr_userid	AN-PPP user ID. Supported on Cellient (CDMA) modem only. blank range

Table 61: Information table for mobile manager basic settings

21.6.1.2 Caller settings

Web: Name UCI: mobile.@caller[0].name Opt: name	Name assigned to the caller. blank range

<table>
<tr><td>Web: Number
UCI: mobile.@caller[0].number
Opt: number</td><td>Number of the caller allowed to SMS the router. Add in specific caller numbers, or use the wildcard symbol.<table><tr><td>blank</td><td></td></tr><tr><td>range</td><td></td></tr></table></td></tr>
<tr><td>Web: Enable
UCI: mobile.@caller[0].enabled
Opt: enabled</td><td>Enables or disables incoming caller ID.<table><tr><td>no</td><td>Disabled.</td></tr><tr><td>yes</td><td>Enabled.</td></tr></table></td></tr>
<tr><td>Web: Respond
UCI: mobile.@caller[0].respond
Opt: respond</td><td>If checked, the router will return an SMS. Select Respond if you want the router to reply.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
</table>

Table 62: Information table for mobile manager caller settings

21.6.1.3 Roaming interface template

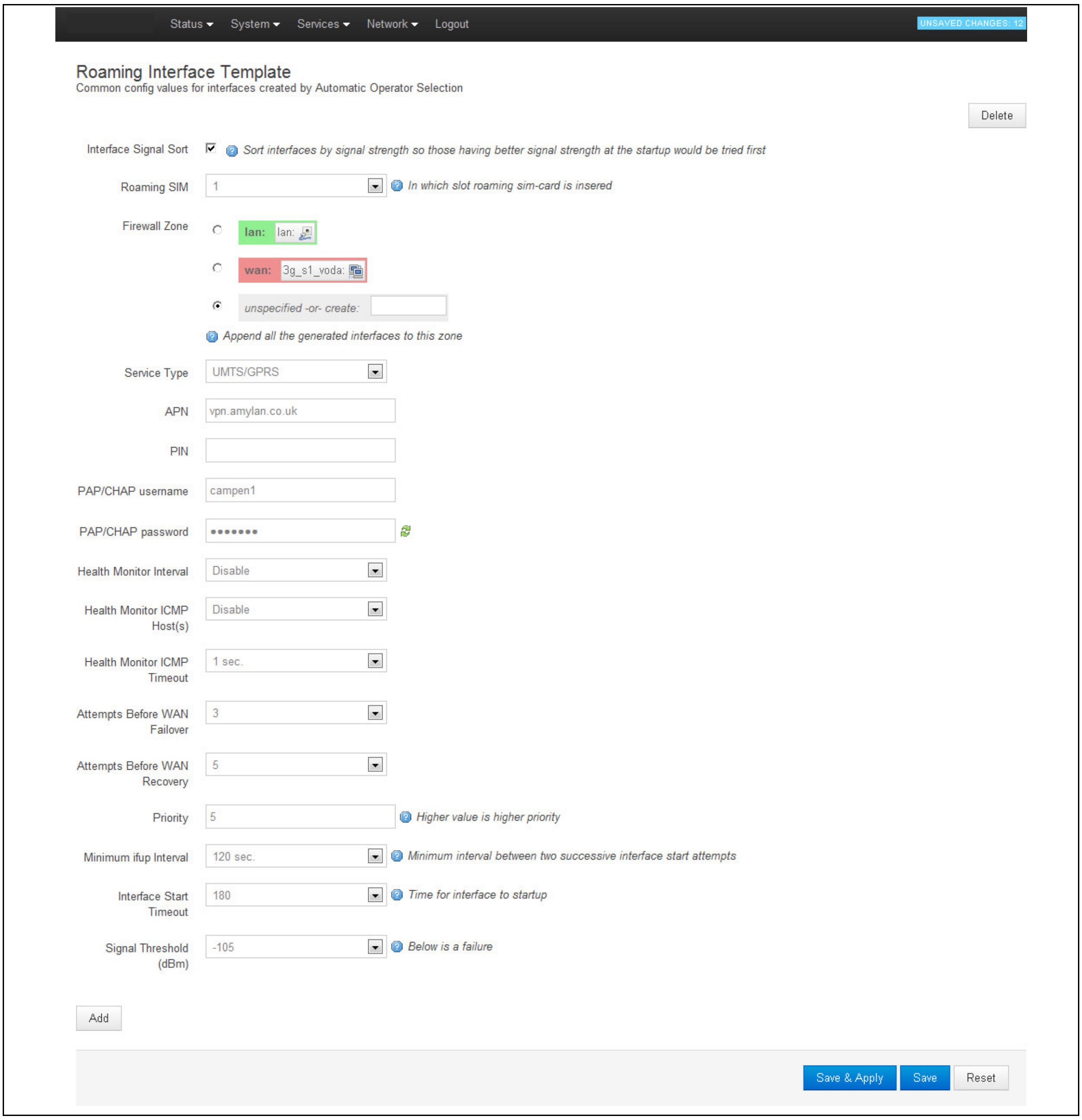


Figure 80: The roaming interface template page

Web Field / UCI / Package Option	Description
Web: Interface Signal Sort UCI: mobile.@roaming_template[0].sort_sig_strength Opt: sort_sig_strength	Sorts interfaces by signal strength priority so those that have a better signal strength will be tried first.
Web: Roaming SIM UCI: mobile.main.roaming_sim Opt: roaming_sim	Sets which slot to insert roaming SIM card. 1 SIM slot 1. 2 SIM slot 2.

<table>
<tr><td>Web: Firewall Zone
UCI: mobile.@roaming_template[0].firewall_zone
Opt: firewall_zone</td><td>Adds all generated interfaces to this zone.
Select existing zone or click unspecified or create to create a new zone.</td></tr>
<tr><td>Web: Service Type
UCI: mobile.@roaming_template[0].service
Opt: service</td><td>Specifies the service type that will be used to connect to the network.<table><tr><td>UMTS/GPRS</td><td>GSM module will automatically detect the best available technology code.</td></tr><tr><td>Umts_only</td><td>Allows GSM module to only connect to 3G network.</td></tr><tr><td>GPRS_only</td><td>Allows GSM module to only connect to GPRS network.</td></tr><tr><td>cdma</td><td>Allows GSM module to only connect to cdma network.</td></tr></table></td></tr>
<tr><td>Web: APN
UCI: mobile.@roaming_template[0].apn
Opt: apn</td><td>APN name of Mobile Network Operator.</td></tr>
<tr><td>Web: PIN
UCI: mobile.@roaming_template[0].pincode
Opt: pincode</td><td>SIM Card’s PIN number.</td></tr>
<tr><td>Web: PAP/CHAP username
UCI: mobile.@roaming_template[0].username
Opt: username</td><td>Username used to connect to APN.</td></tr>
<tr><td>Web: PAP/CHAP password
UCI: mobile.@roaming_template[0].password
Opt: password</td><td>Password used to connect to APN.</td></tr>
<tr><td>Web: Health Monitor Interval
UCI: mobile.@roaming_template[0].health_interval
Opt: health_interval</td><td>Sets the interval used to monitor signal strength in seconds.</td></tr>
<tr><td>Web: Health Monitor ICMP Host(s)
UCI: mobile.@roaming_template[0].icmp_hosts
Opt: icmp_hosts</td><td>Specifies target IP address for ICMP packets.<table><tr><td>DNS servers</td><td>IP address of DNS servers.</td></tr><tr><td>WAN gateway</td><td>IP address of Gateway.</td></tr><tr><td>custom</td><td>Custom Interface IP address.</td></tr></table></td></tr>
<tr><td>Web: Health Monitor ICMP Timeout
UCI: mobile.@roaming_template[0].timeout
Opt: timeout</td><td>Specifies the time in seconds that Health Monitor ICMP will timeout at.</td></tr>
</table>

Web: Attempts Before WAN Failover UCI: mobile.@roaming_template[1].health_fail_retries Opt: health_fail_retries	Number of fail attempts of health monitor before interface is disconnected. Select the number of fail attempts of health monitor checks that will cause the interface to be disconnected. 3 | range |
Web: Attempts Before WAN Recovery UCI: mobile.@roaming_template[0].health_recovery_retries Opt: health_recovery_retries	Select the number of fail attempts of health monitor checks that will cause the interface to be disconnected. 5 | range |
Web: Priority UCI: mobile.@roaming_template[0].priority Opt: priority	Type the priority number. The higher the value, the higher the priority. This multi-WAN interface priority must be higher than the one specified in the priority field in the 'Roaming Interface Template' page described in the following section. 0 | range |
Web: Minimum ifup interval UCI: Opt:	Minimum interval between two successive interface start attempts.
Web: Interface Start Timeout UCI: mobile.@roaming_template[0].ifup_timeout_sec Opt: ifup_timeout	Time allowed for interface to start up. Put value greater than 120 seconds. 40 | range |
Web: Signal Threshold (dBm) UCI: mobile.@roaming_template[0].signal_threshold Opt: signal_threshold	If signal is lower than this value, then it is marked as fail. -115 dBm | range |

Table 63: Information table for roaming interface template

When you have configured your settings, click **Save & Apply**.

21.6.2 Set multi-WAN operation

From the top menu, select **Network -> Multi-Wan**. The Multi-WAN page appears.

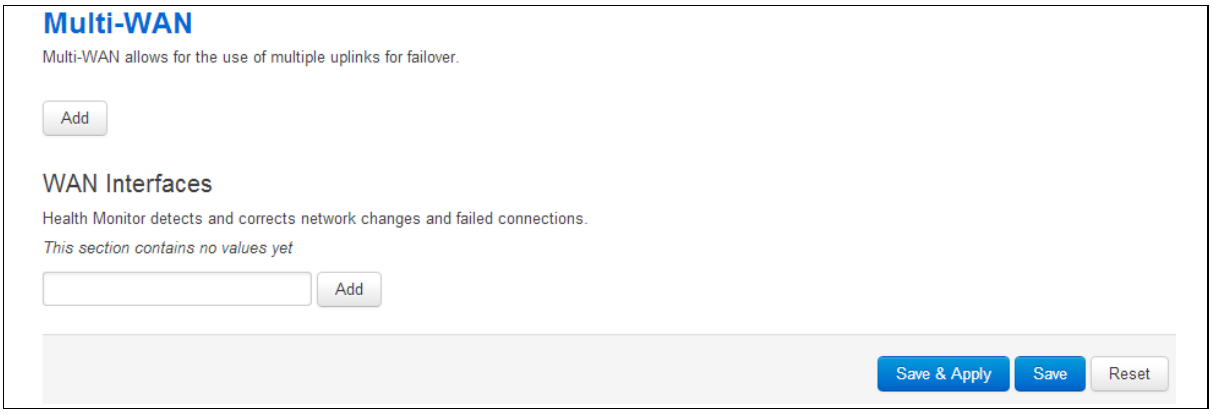


Figure 81: The multi-WAN page

Under Multi-WAN section click **Add**.

Web Field/UCI/Package Option	Description
Web: Enable UCI: multiwan.config.enabled Opt: enabled	Enables Multi-WAN. Select this option. 0 Disabled. 1 Enabled.
Web: Preempt UCI: multiwan.config.preempt Opt: preempt	Enables Preempt mode. Leave this option unselected. 0 Disabled. 1 Enabled.
Web: Alternate Mode UCI: multiwan.config.alt Opt: alt	It will use alternate interface after reboot.

Table 64: Information table for multi-WAN operation

Click **Save & Apply**.

21.7 Configuring No PMP + roaming using UCI

The configuration file is stored at:

Mobile package file /etc/config/mobile

```
uci export mobile


package mobile
config mobile 'main'
        option sms 'yes'
        option roaming_sim '1'
        option debug '1'


config caller
```

```
        option name 'Eval'
        option number '*'
        option enabled 'yes'
        option respond 'yes'
config roaming_template
        option roaming_sim '1'
        option firewall_zone 'wan'
        option apn 'test IE'
        option username 'test'
        option password 'test'
        option service 'umts'
        option health_fail_retries '2'
        option signal_threshold '-100'
        option priority '5'
        option ifup_timeout_sec '180'
        option defaultroute 'yes'
        option sort_sig_strength 'yes'
        option ifup_retry_sec '200'
        option health_interval '120'
        option icmp_hosts '172.31.4.129'
        option timeout '3'
        option health_recovery_retries '3'
```

To view uci commands, enter:

```
uci export mobile

mobile.main=mobile
mobile.main.sms=yes
mobile.main.roaming_sim=1
mobile.main.debug=1
mobile.@caller[0]=caller
mobile.@caller[0].name=Eval
mobile.@caller[0].number=*
mobile.@caller[0].enabled=yes
```

```
mobile.@caller[0].respond=yes
mobile.@roaming_template[0]=roaming_template
mobile.@roaming_template[0].roaming_sim=1
mobile.@roaming_template[0].firewall_zone=wan
mobile.@roaming_template[0].apn=stream.co.uk
mobile.@roaming_template[0].username=default
mobile.@roaming_template[0].password=void
mobile.@roaming_template[0].service=umts
mobile.@roaming_template[0].health_fail_retries=2
mobile.@roaming_template[0].signal_threshold=-100
mobile.@roaming_template[0].priority=5
mobile.@roaming_template[0].ifup_timeout_sec=180
mobile.@roaming_template[0].defaultroute=yes
mobile.@roaming_template[0].sort_sig_strength=yes
mobile.@roaming_template[0].ifup_retry_sec=200
mobile.@roaming_template[0].health_interval=120
mobile.@roaming_template[0].icmp_hosts=172.31.4.129
mobile.@roaming_template[0].timeout=3
mobile.@roaming_template[0].health_recovery_retries=3
```

The package multiwan file is stored at

/etc/config/multiwan

To view multiwan file, enter:

```
uci export multiwan
package multiwan

config multiwan 'config'
        option enabled 'yes'
        option preempt 'no'
        option alt_mode 'no'
```

To see package multiwan uci commands, enter:

```
uci show multiwan

multiwan.config=multiwan
multiwan.config.enabled=yes
multiwan.config.preempt=no
multiwan.config.alt_mode=no
```

21.8 Automatic operator selection diagnostics via the web interface

21.8.1 Checking the status of the Multi-WAN package

When interfaces are auto created they are presented in the network and in the Multi-WAN package.

To check interfaces created in the Multi-WAN package, from the top menu, select **Network -> Multi-WAN**.

To check interfaces that have been created in the network package, from the top menu, select **Network -> Interfaces**.

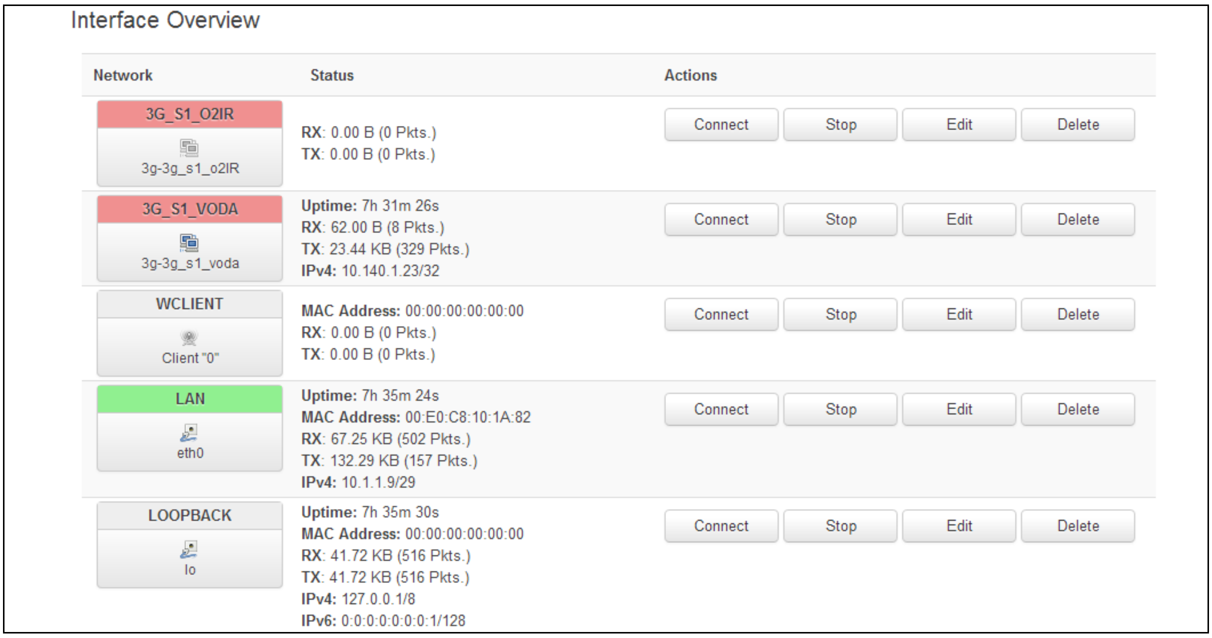


Figure 82: The interface overview page

To check the status of the interface you are currently using, in the top menu, click **Status**. The Interface Status page appears.

Scroll down to the bottom of the page to view Multi-WAN Stats.

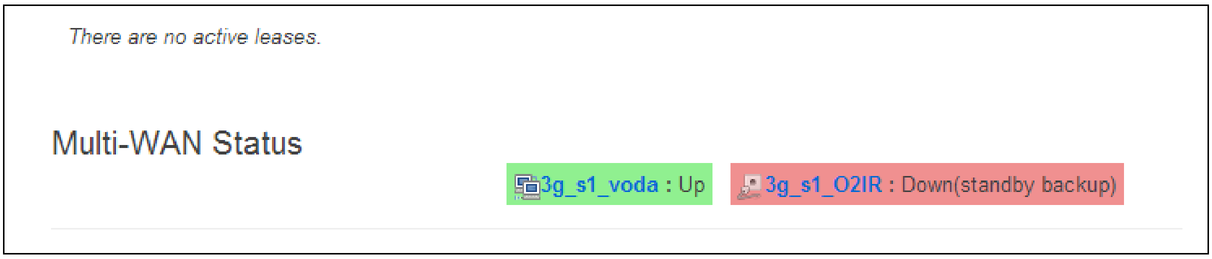
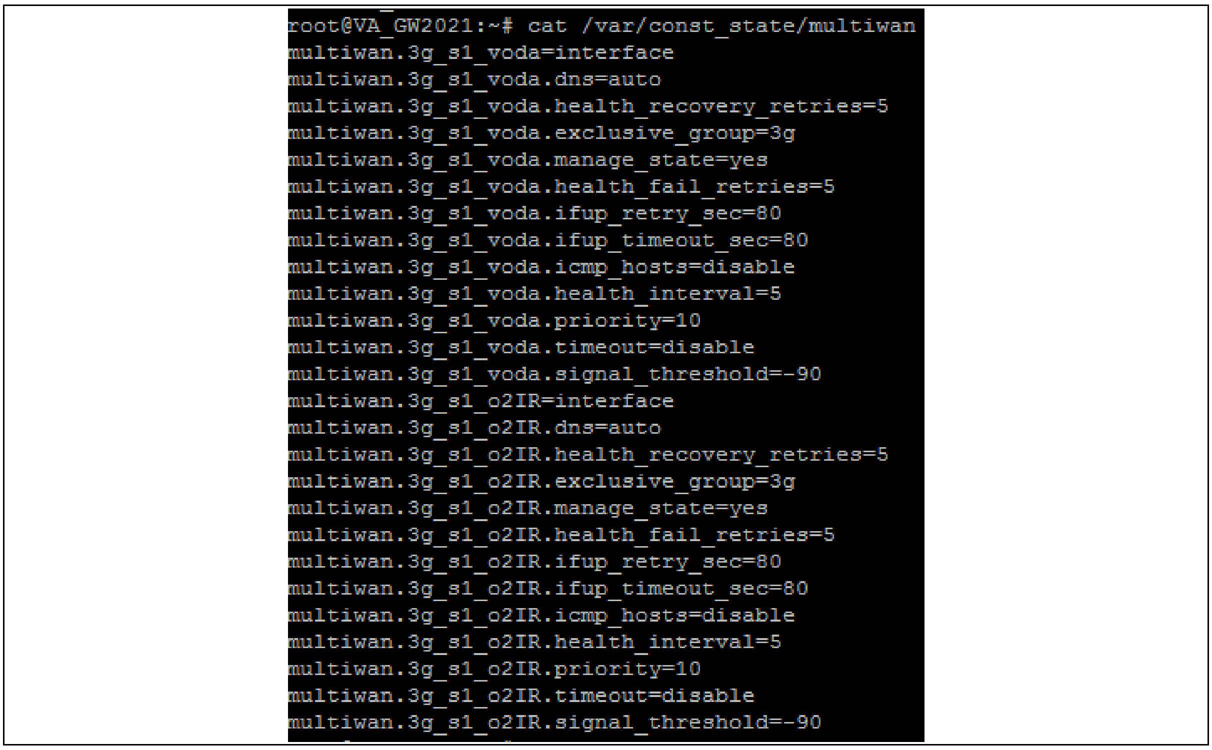


Figure 83: The status page: multi-WAN status section page

21.9 Automatic operator selection diagnostics via UCI

To check interfaces created in the multi-WAN package, enter:

```
cat /var/const_state/multiwan
```

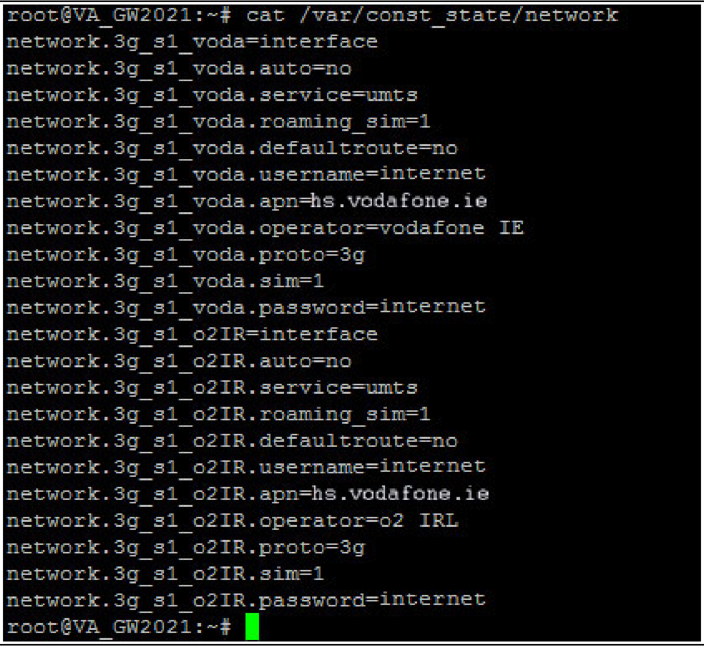


```
root@VA_GW2021:~# cat /var/const_state/multiwan
multiwan.3g_s1_voda=interface
multiwan.3g_s1_voda.dns=auto
multiwan.3g_s1_voda.health_recovery_retries=5
multiwan.3g_s1_voda.exclusive_group=3g
multiwan.3g_s1_voda.manage_state=yes
multiwan.3g_s1_voda.health_fail_retries=5
multiwan.3g_s1_voda.ifup_retry_sec=80
multiwan.3g_s1_voda.ifup_timeout_sec=80
multiwan.3g_s1_voda.icmp_hosts=disable
multiwan.3g_s1_voda.health_interval=5
multiwan.3g_s1_voda.priority=10
multiwan.3g_s1_voda.timeout=disable
multiwan.3g_s1_voda.signal_threshold=-90
multiwan.3g_s1_o2IR=interface
multiwan.3g_s1_o2IR.dns=auto
multiwan.3g_s1_o2IR.health_recovery_retries=5
multiwan.3g_s1_o2IR.exclusive_group=3g
multiwan.3g_s1_o2IR.manage_state=yes
multiwan.3g_s1_o2IR.health_fail_retries=5
multiwan.3g_s1_o2IR.ifup_retry_sec=80
multiwan.3g_s1_o2IR.ifup_timeout_sec=80
multiwan.3g_s1_o2IR.icmp_hosts=disable
multiwan.3g_s1_o2IR.health_interval=5
multiwan.3g_s1_o2IR.priority=10
multiwan.3g_s1_o2IR.timeout=disable
multiwan.3g_s1_o2IR.signal_threshold=-90
```

Figure 84: Output from the command: cat /var/const_stat/multiwan

To check interfaces created in the network package, enter:

```
cat /var/const_state/network
```

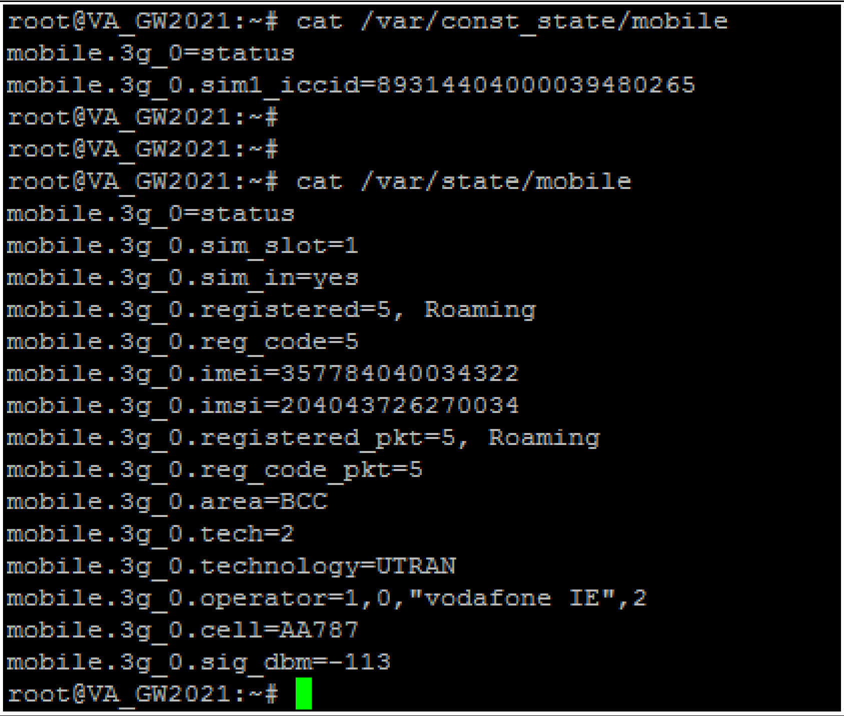


```
root@VA_GW2021:~# cat /var/const_state/network
network.3g_s1_voda=interface
network.3g_s1_voda.auto=no
network.3g_s1_voda.service=umts
network.3g_s1_voda.roaming_sim=1
network.3g_s1_voda.defaultroute=no
network.3g_s1_voda.username=internet
network.3g_s1_voda.apn=hs.vodafone.ie
network.3g_s1_voda.operator=vodafone IE
network.3g_s1_voda.proto=3g
network.3g_s1_voda.sim=1
network.3g_s1_voda.password=internet
network.3g_s1_o2IR=interface
network.3g_s1_o2IR.auto=no
network.3g_s1_o2IR.service=umts
network.3g_s1_o2IR.roaming_sim=1
network.3g_s1_o2IR.defaultroute=no
network.3g_s1_o2IR.username=internet
network.3g_s1_o2IR.apn=hs.vodafone.ie
network.3g_s1_o2IR.operator=o2 IRL
network.3g_s1_o2IR.proto=3g
network.3g_s1_o2IR.sim=1
network.3g_s1_o2IR.password=internet
root@VA_GW2021:~#
```

Figure 85: Output from the command cat /var/const_state/network

To check the status of the interface you are currently using, enter:

```
cat /var/const_state_/mobile
```



```
root@VA_GW2021:~# cat /var/const_state/mobile
mobile.3g_0=status
mobile.3g_0.sim1_iccid=89314404000039480265
root@VA_GW2021:~#
root@VA_GW2021:~#
root@VA_GW2021:~# cat /var/state/mobile
mobile.3g_0=status
mobile.3g_0.sim_slot=1
mobile.3g_0.sim_in=yes
mobile.3g_0.registered=5, Roaming
mobile.3g_0.reg_code=5
mobile.3g_0.imei=357784040034322
mobile.3g_0.imsi=204043726270034
mobile.3g_0.registered_pkt=5, Roaming
mobile.3g_0.reg_code_pkt=5
mobile.3g_0.area=BCC
mobile.3g_0.tech=2
mobile.3g_0.technology=UTRAN
mobile.3g_0.operator=1,0,"vodafone IE",2
mobile.3g_0.cell=AA787
mobile.3g_0.sig_dbm=-113
root@VA_GW2021:~#
```

Figure 86: Output from the command cat /vat/const_state_/mobile

22 Configuring IPSec

Internet Protocol Security (IPSec) is a protocol suite used to secure communications at IP level. Use IPSec to secure communications between two hosts or between two networks. Virtual Access routers implement IPSec using strongSwan software.

If you need to create an IPSec template for DMVPN, read the chapter 'Dynamic Multipoint Virtual Private Network (DMVPN)'.

22.1 Configuration package used

Package	Sections
strongswan	general connection secret

22.2 Configuring IPSec using the web interface

To configure IPSec using the web interface, in the top menu, select **Services -> IPSec**. The strongSwan IPSec VPN page appears. There are three sections:

Common Settings	Control the overall behaviour of strongSwan. This behaviour is common across all tunnels.
Connection Settings	Together, these sections define the required parameters for a two-way IKEv1 tunnel.
Secret Settings	

22.2.1 Configure common settings

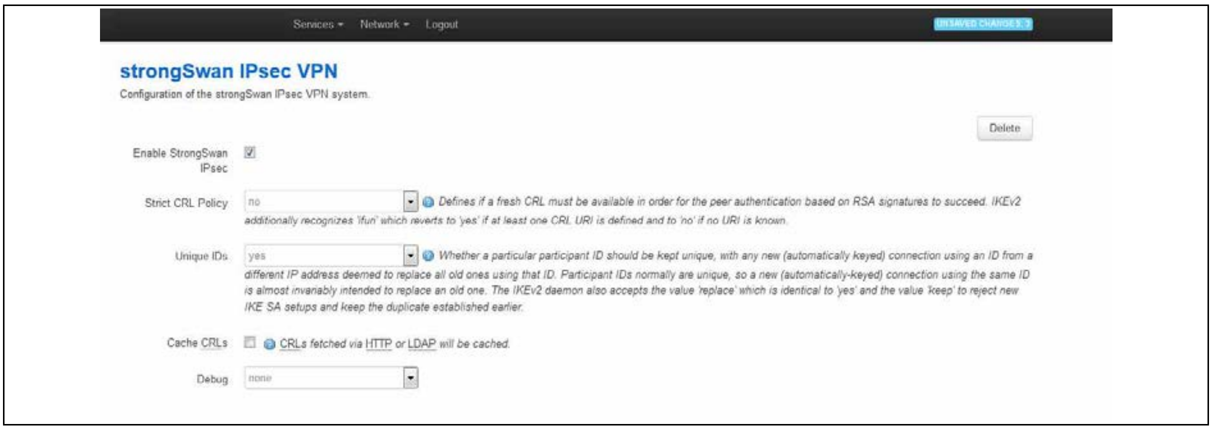


Figure 87: The common settings section

Web Field/UCI/Package Option	Description
Web: Enable strongswan UCI: strongswan.general.enable Opt: enabled	Enables or disables IPSec. 0: Disabled. 1: Enabled.
Web: Strict CRL Policy UCI: strongswan.general.strictcrlpolicy Opt: strictcrlpolicy	Defines if a fresh CRL must be available for the peer authentication based on RSA signatures to succeed. 0: Disabled. 1: Enabled. ifuri: The IKEv2 application additionally recognizes the “ifuri” option which reverts to ‘yes’ if at least one CRL URI is defined and to ‘no’ if no URI is known.
Web: Unique IDs UCI: strongswan.general.uniqueids Opt: uniqueids	Defines whether a particular participant ID should be kept unique, with any new (automatically keyed) connection using an ID from a different IP address deemed to replace all old ones using that ID. Participant IDs normally are unique, so a new (automatically-keyed) connection using the same ID is almost invariably intended to replace an old one. 0: Disabled. 1: Enabled. replace: Identical to Yes keep: Rejects new IKE SA and keep the duplicate established earlier
Web: Cache CRLs UCI: strongswan.general.cachecrls Opt: cachecrls	Certificate Revocation Lists (CRLs) fetched via HTTP or LDAP will be cached in /etc/ipsec.d/crls/ under a unique file name derived from the certification authority's public key. 0: Disabled. 1: Enabled.
Web: Debug UCI: strongswan.general.debug Opt: debug	Enable debugging. This option is used for trouble shooting issues. It is not suitable for a production environment. None: Debug disabled. Control: Debug enabled. Shows generic control flow with errors and very basic auditing logs. All: Debug enabled. Most verbose logging also includes sensitive information such as keys.

Table 65: Information table for IPSec common settings

22.2.2 Configure connection settings

Scroll down to view connection settings.

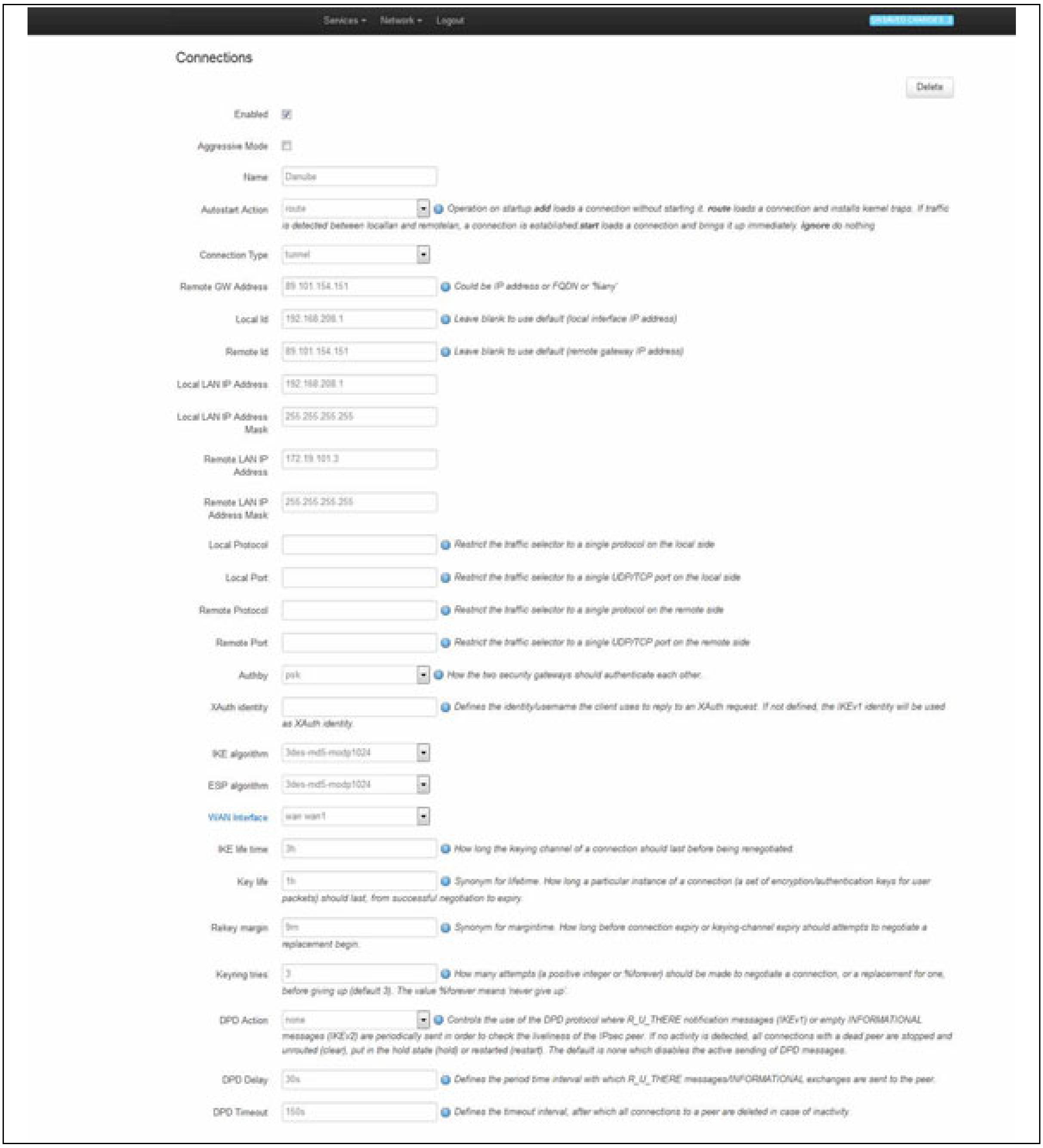


Figure 88: The connections settings section

Web Field / UCI / Package Option	**Description**
Web: Enabled UCI: strongswan.@connection[X].enabled Opt: enable	Enables or disables IPSec connection. 0 — Disabled. 1 — Enabled.
Web: Aggressive UCI: strongswan.@connection[X].aggressive Opt: aggressive	Enables or disables IKE aggressive mode. Note: using aggressive mode along with PSK authentication is less secure method than main mode and should be avoided. 0 — Disabled. 1 — Enabled.

Web: Name UCI: strongswan.@connection[X].name Opt: name	Specifies a name for the tunnel.
Web: Autostart Action UCI: strongswan.@connection[X].auto Opt: auto	Specifies when the tunnel is initiated. start: On start up. route: When traffic routes this way. add: Loads a connection without starting it. ignore: Ignores the connection. always: Actively retries to establish the tunnel if it went down.
Web: Connection Type UCI: strongswan.@connection[X].type Opt: type	Defines the type of IPSec connection. tunnel: Connection uses tunnel mode. transport: Connection uses transport mode. pass: Connection does not perform any IPSec processing. drop: Connection drops all the packets.
Web: Remote GW Address UCI: strongswan.@connection[X]. remoteaddress Opt: remoteaddress	Sets the public IP address of the remote peer.
Web: Local ID UCI: strongswan.@connection[X].localid Opt: localid	Defines the local peer identifier.
Web: Remote ID UCI: strongswan.@connection[X].remoteid Opt:remoteid	Defines the remote peer identifier.
Web: Local LAN IP Address UCI: strongswan.@connection[X]. locallan Opt: locallan	Defines the local IP of LAN.
Web: Local LAN IP Address Mask UCI: strongswan.@connection[X]. locallanmask Opt: locallanmask	Defines the subnet of local LAN.
Web: Remote LAN IP Address UCI: strongswan.@connection[X]. remotelan Opt:remotelan	Defines the IP address of LAN serviced by remote peer.
Web: Remote LAN IP Address Mask UCI: strongswan.@connection[X]. remotelanmask Opt:remotelanmask	Defines the Subnet of remote LAN.
Web: Local Protocol UCI: strongswan.@connection[X].localproto Opt: localproto	Restricts the connection to a single protocol on the local side.

<table>
<tr><td>Web: Local Port
UCI: strongswan.@connection[X].localport
Opt: localport</td><td>Restricts the connection to a single port on the local side.</td></tr>
<tr><td>Web: Remote Protocol
UCI: strongswan.@connection[X].remoteproto
Opt:remoteproto</td><td>Restricts the connection to a single protocol on the remote side.</td></tr>
<tr><td>Web: Remote Port
UCI: strongswan.@connection[X].remoteport
Opt: remoteport</td><td>Restricts the connection to a single port on the remote side.</td></tr>
<tr><td>Web: Authby
UCI: strongswan.@connection[X].authby
Opt: authby</td><td>Defines how the two secure gateways should authenticate.
Note: using aggressive mode along with PSK authentication is unsecure and should be avoided.
<table>
<tr><td>Pubkey</td><td>For public key signatures.</td></tr>
<tr><td>Rsasig</td><td>For RSA digital signatures.</td></tr>
<tr><td>ecdsasig</td><td>For Elliptic Curve DSA signatures.</td></tr>
<tr><td>Psk</td><td>Using a preshared key.</td></tr>
<tr><td>xauthrsasig</td><td>Enables eXtended Authentication (XAuth) with addition to RSA signatures.</td></tr>
<tr><td>xauthpsk</td><td>Using extended authentication and preshared key.</td></tr>
<tr><td>never</td><td>Can be used if negotiation is never to be attempted or accepted (shunt connections).</td></tr>
</table></td></tr>
<tr><td>Web: XAuth Identity
UCI: strongswan.@connection[X].xauth_identity
Opt: xauth_identity</td><td>Defines Xauth ID.</td></tr>
</table>

Web: IKE Algorithm UCI: strongswan.@connection[X].ike Opt: ike	Specifies the IKE algorithm to use. The format is: encAlgo | authAlgo | DHGroup • encAlgo: o 3des o aes o serpent o twofish o blowfish • authAlgo: o md5 o sha o sha2 • DHGroup: o modp1024 o modp1536 o modp2048 o modp3072 o modp4096 o modp6144 o modp8192 For example, a valid IKE algorithm is aes128-sha-modp1536.
Web: ESP algorithm UCI: strongswan.@connection[X].esp Opt: esp	Specifies the esp algorithm to use. The format is: encAlgo | authAlgo | DHGroup • encAlgo: o 3des o aes o serpent o twofish o blowfish • authAlgo: o md5 o sha o sha2 • DHGroup: o modp1024 o modp1536 o modp2048 o modp3072 o modp4096 o modp6144 o modp8192 For example, a valid encryption algorithm is: aes128-sha-modp1536. If no DH group is defined then PFS is disabled.

Web: WAN Interface UCI: strongswan.@connection[X].waniface Opt: waniface	This is a space separated list of the WAN interfaces the router will use to establish a tunnel with the secure gateway. On the web, a list of the interface names is automatically generated. If you want to specify more than one interface use the “custom” value. **Example**: If you have a 3G WAN interface called ‘wan and a WAN ADSL interface called ‘dsl’ and wanted to use one of these interfaces for this IPSec connection, you would use: ‘wan adsl’.
Web: IKE Life Time UCI: strongswan.@connection[X].ikelifetime Opt:ikelifetime	Specifies how long the keyring channel of a connection (ISAKMP or IKE SA) should last before being renegotiated. 3h Timespec: 1d, 3h, 25m, 10s.
Web: Key Life UCI: strongswan.@connection[X].keylife Opt: keylife	Specifies how long a particular instance of a connection (a set of encryption/authentication keys for user packets) should last, from successful negotiation to expiry. Normally, the connection is renegotiated (via the keying channel) before it expires (see rekeymargin). 1h Timespec: 1d, 1h, 25m, 10s.
Web: Rekey Margin UCI: strongswan.@connection[X].rekeymargin Opt: rekeymargin	Specifies how long before connection expiry or keying-channel expiry should attempt to negotiate a replacement begin. Relevant only locally, other end need not agree on it. 9m Timespec: 1d, 2h, 9m, 10s.
Web: Keyring Tries UCI: strongswan.@connection[X].keyringtries Opt: keyringtries	Specifies how many attempts (a positive integer or %forever) should be made to negotiate a connection, or a replacement for one, before giving up. The value %forever means 'never give up'. Relevant only locally, other end need not agree on it.
Web: DPD Action UCI: strongswan.@connection[X].dpdaction Opt: dpdaction	Defines DPD (Dead Peer Detection) action. None: Disables DPD. Clear: Clear down the tunnel if peer does not respond. Reconnect when traffic brings the tunnel up. Hold: Clear down the tunnel and bring up as soon as the peer is available. Restart: Restarts DPD when no activity is detected.
Web: DPD Delay UCI: strongswan.@connection[X].dpddelay Opt: dpddelay	Defines the period time interval with which R_U_THERE messages and INFORMATIONAL exchanges are sent to the peer. These are only sent if no other traffic is received. 30s Timespec: 1d, 2h, 25m, 10s.

Web: DPD Timeout UCI: strongswan.@connection[X].dpdtimeout Opt: dpdtimeout	Defines the timeout interval, after which all connections to a peer are deleted in case of inactivity. 150s | Timespec | 1d, 2h, 25m, 10s.

Table 66: Information table for IPSec connections settings

22.2.3 Configure secrect settings

Each tunnel requires settings to configure how the local end point of the tunnel proves its identity to the remote end point.

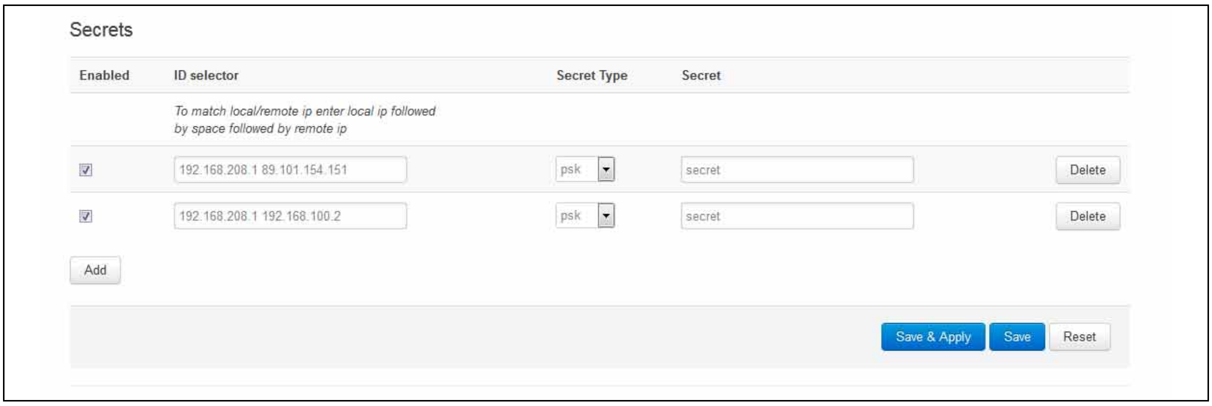


Figure 89: IPSec secrets settings

Web Field / UCI / Package Option	**Description**
Web: Enabled UCI: strongswan.@secret[X].enabled Opt: enabled	Defines whether this set of credentials is to be used or not. 0 | Disabled. 1 | Enabled.
Web: ID selector UCI: strongswan.@secret[X].idtype Opt: idtype	Defines whether IP address or userfqdn is used.
Web: ID selector UCI: strongswan.@secret[X].localaddress Opt: localaddress	Defines the local address this secret applies to.
Web: ID selector UCI: strongswan.@secret[X].remoteaddress Opt: remoteaddress	Defines the remote address this secret applies to.
Web: N/A UCI: strongswan.@secret[X].userfqnd Opt: userfqnd	FQDN or Xauth name used of Extended Authentication. This must match xauth_identity from the configuration connection section.

<table>
<tr><td>Web: Secret Type
UCI: strongswan.@secret[X].secrettype
Opt: secrettype</td><td>Specifies the authentication mechanism to be used by the two peers.<table><tr><td>Psk</td><td>Preshared secret</td></tr><tr><td>Pubkey</td><td>Public key signatures</td></tr><tr><td>Rsasig</td><td>RSA digital signatures</td></tr><tr><td>Ecdsasig</td><td>Elliptic Curve DSA signatures</td></tr><tr><td>Xauth</td><td>Extended authentication</td></tr></table></td></tr>
<tr><td>Web: Secret
UCI: strongswan.@secret[X].secret
Opt: secret</td><td>Defines the secret.</td></tr>
</table>

Table 67: Information table for IPSec secret settings

22.3 Configuring IPSec using UCI

22.3.1 Common settings

An example of a typical set of common settings for strongSwan is shown below.

```
# Commands
touch /etc/config/strongswan
uci set strongswan.general=general
uci set strongswan.general.enabled=yes
uci set strongswan.general.strictcrlpolicy=no
uci set strongswan.general.uniqueids=yes
uci set strongswan.general.cachecrls=no
uci set strongswan.general.debug=none
uci commit

# This will create the following
config general 'general'
        option enabled 'yes'
        option strictcrlpolicy 'no'
        option uniqueids 'yes'
        option cachecrls 'no'
        option debug 'none'
```

22.3.2 Connection settings

A typical tunnel configuration is shown below.

```
# Commands to configure a typical tunnel using uci
touch /etc/config/strongswan
uci add strongswan connection
uci set strongswan.@connection[0].ikelifetime=3h
uci set strongswan.@connection[0].keylife=1h
uci set strongswan.@connection[0].rekeymargin=9m
uci set strongswan.@connection[0].keyingtries=3
uci set strongswan.@connection[0].dpddelay=30s
uci set strongswan.@connection[0].dpdtimeout=150s
uci set strongswan.@connection[0].enabled=yes
uci set strongswan.@connection[0].name=3G_Backup
uci set strongswan.@connection[0].auto=start
uci set strongswan.@connection[0].type=tunnel
uci set strongswan.@connection[0].remoteaddress=100.100.100.100
uci set strongswan.@connection[0].localid=192.168.209.1
uci set strongswan.@connection[0].remoteid=100.100.100.100
uci set strongswan.@connection[0].locallan=192.168.209.1
uci set strongswan.@connection[0].locallanmask=255.255.255.255
uci set strongswan.@connection[0].remotelan=172.19.101.3
uci set strongswan.@connection[0].remotelanmask=255.255.255.255
uci set strongswan.@connection[0].authby=xauthpsk
uci set strongswan.@connection[0].xauth_identity=testxauth
uci set strongswan.@connection[0].ike=3des-md5-modp1024
uci set strongswan.@connection[0].esp=3des-md5
uci set strongswan.@connection[0].waniface=wan
uci set strongswan.@connection[0].dpdaction=hold
uci commit
```

This will create the following output:

```
config connection
        option ikelifetime '3h'
        option keylife '1h'
        option rekeymargin '9m'
        option keyingtries '3'
```

```
        option dpddelay '30s'
        option dpdtimeout '150s'
        option enabled 'yes'
        option name '3G_Backup'
        option auto 'start'
        option type 'tunnel'
        option remoteaddress '100.100.100.100 '
        option localid '192.168.209.1'
        option remoteid '100.100.100.100 '
        option locallan '192.168.209.1'
        option locallanmask '255.255.255.255'
        option remotelan '172.19.101.3'
        option remotelanmask '255.255.255.255'
        option authby 'xauthpsk'
        option xauth_identity 'testxauth'
        option ike '3des-md5-modp1024'
        option esp '3des-md5'
        option waniface 'wan'
        option dpdaction 'hold'
```

22.3.3 Shunt connection

If the remote LAN network is 0.0.0.0/0 then all traffic generated on the local LAN will be sent via the IPSec tunnel. This includes the traffic destined to the router's IP address. To avoid this situation you must include an additional config connection section.

```
# Commands
touch /etc/config/strongswan
uci add strongswan connection
uci set strongswan.@connection[1].name=local
uci set strongswan.@connection[1].enabled=yes
uci set strongswan.@connection[1].locallan=10.1.1.1
uci set strongswan.@connection[1].locallanmask=255.255.255.255
uci set strongswan.@connection[1].remotelan=10.1.1.0
uci set strongswan.@connection[1].remotelanmask=255.255.255.0
```

```
uci set strongswan.@connection[1].type=pass
uci set strongswan.@connection[1].auto=route
uci commit
```

This will create the following output:

```
config connection
        option name 'local'
        option enabled 'yes'
        option locallan '10.1.1.1'
        option locallanmask '255.255.255.255'
        option remotelan '10.1.1.0'
        option remotelanmask '255.255.255.0'
        option type 'pass'
        option auto 'route'
```

Traffic originated on remotelan and destined to locallan address is excluded from VPN IPSec policy.

22.3.4 Secret settings

Each tunnel also requires settings for how the local end point of the tunnel proves its identity to the remote end point.

A sample secret section which could be used with the connection section in ‘Connection Settings’ is shown below.

```
# Commands to add a secret for psk auth
touch /etc/config/strongswan
uci add strongswan secret
uci set strongswan.@secret[0].enabled=yes
uci set strongswan.@secret[0].localaddress=192.168.209.1
uci set strongswan.@secret[0].remoteaddress= 100.100.100.100
uci set strongswan.@secret[0].secrettype=psk
uci set strongswan.@secret[0].secret=secret
uci commit
```

This will create the following output:

```
config secret
        option enabled 'yes'
        option localaddress '192.168.209.1'
        option remoteaddress '100.100.100.100 '
        option secrettype 'psk'
        option secret 'secret'
```

If xauth is defined as the authentication method then you must include an additional config secret section, as shown in the example below.

```
# Commands to add a secret for xauth auth
touch /etc/config/strongswan
uci add strongswan secret
uci set strongswan.@secret[1].enabled=yes
uci set strongswan.@secret[1].idtype=userfqdn
uci set strongswan.@secret[1].userfqdn=testxauth
uci set strongswan.@secret[1].remoteaddress=100.100.100.100
uci set strongswan.@secret[1].secret=xauth
uci set strongswan.@secret[1].secrettype=XAUTH
uci commit

# This will create the following:
config secret
        option enabled 'yes'
        option idtype 'userfqdn'
        option userfqdn 'testxauth'
        option remoteaddress '100.100.100.100'
        option secret 'xauth'
        option secrettype 'XAUTH'
```

22.4 Configuring an IPSec template for DMVPN via the web interface

To configure IPSec using the web interface, in the top menu, select **Services -> IPSec**. The strongSwan IPSec VPN page appears. There are three sections:

Common Settings	Control the overall behaviour of strongSwan. This behaviour is common across all tunnels.
Connection Settings	Together, these sections define the required parameters for a two-way IKEv1 tunnel.
Secret Settings	

22.4.1 Configure common settings

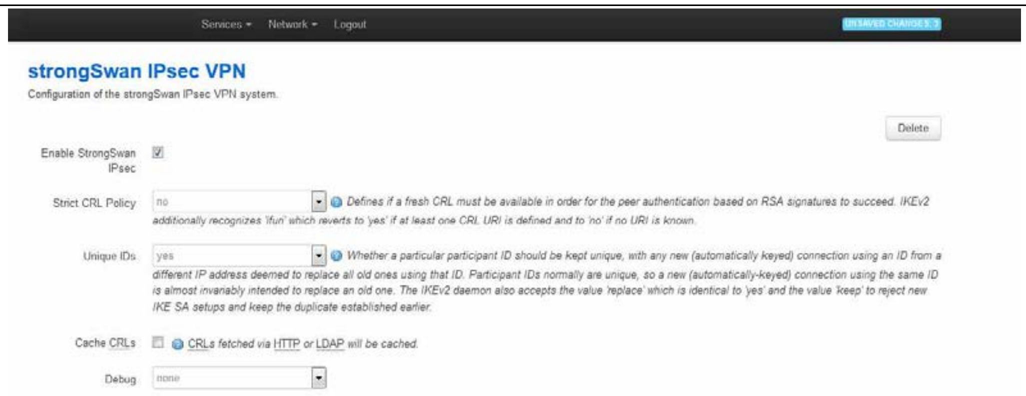


Figure 90: The common settings section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Enable strongswan
UCI: strongswan.general.enable
Opt: enabled</td><td>Enables or disables IPSec.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
<tr><td>Web: Strict CRL Policy
UCI: strongswan.general.strictcrlpolicy
Opt: strictcrlpolicy</td><td>Defines if a fresh CRL must be available for the peer authentication based on RSA signatures to succeed.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr><tr><td>ifuri</td><td>The IKEv2 application additionnaly recognizes the “ifuri” option which reverts to ‘yes’ if at least one CRL URI is defined and to ‘no’ if no URI is known.</td></tr></table></td></tr>
<tr><td>Web: Unique IDs
UCI: strongswan.general.uniqueids
Opt: uniqueids</td><td>Defines whether a particular participant ID should be kept unique, with any new (automatically keyed) connection using an ID from a different IP address deemed to replace all old ones using that ID.
Participant IDs normally are unique, so a new (automatically-keyed) connection using the same ID is almost invariably intended to replace an old one.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr><tr><td>replace</td><td>Identical to Yes</td></tr><tr><td>keep</td><td>Rejects new IKE SA and keep the duplicate established earlier</td></tr></table></td></tr>
<tr><td>Web: Cache CRLs
UCI: strongswan.general.cachecrls
Opt: cachecrls</td><td>Certificate Revocation Lists (CRLs) fetched via HTTP or LDAP will be cached in /etc/ipsec.d/crls/ under a unique file name derived from the certification authority's public key.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table></td></tr>
</table>

<table>
<tr><td rowspan="4">Web: Debug
UCI: strongswan.general.debug
Opt: debug</td><td colspan="2">Enable debugging. This option is used for trouble shooting issues. It is not suitable for a production environment.</td></tr>
<tr><td>None</td><td>Debug disabled.</td></tr>
<tr><td>Control</td><td>Debug enabled. Shows generic control flow with errors and very basic auditing logs.</td></tr>
<tr><td>All</td><td>Debug enabled. Most verbose logging also includes sensitive information such as keys.</td></tr>
</table>

Table 68: Information table for IPSec common settings

22.4.2 Configure connection settings

Note: If you want to create a DMVPN, you do not need to configure all settings as the DMVPN will automatically create them using the template. Leave the following sections blank:

- Remote GW Address
- Local ID
- Remote Id
- Local LAN IP Address
- Local LAN IP Address Mask
- Remote LAN IP Address
- Remote LAN IP Address Mask

Scroll down from common settings section to view connection settings.

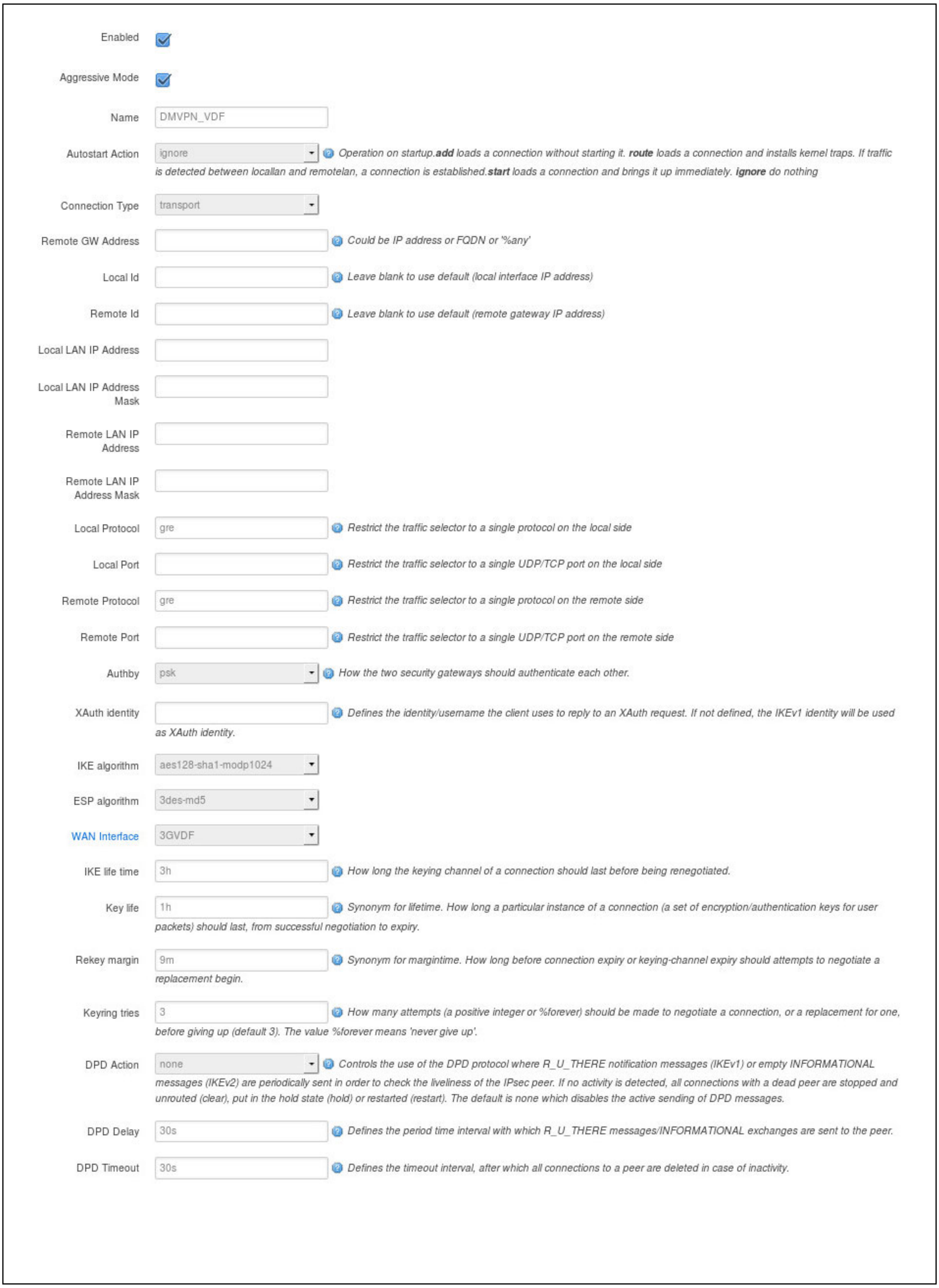


Figure 91: The connections settings section

Web Field / UCI / Package Option	Description
Web: Enabled UCI: strongswan.@connection[X].enabled Opt: enable	Enables or disables IPSec connection. 0: Disabled. 1: Enabled.
Web: Aggressive UCI: strongswan.@connection[X].aggressive Opt: aggressive	Enables or disables IKE aggressive mode. Note: using aggressive mode along with PSK authentication is less secure method than main mode and should be avoided. 0: Disabled. 1: Enabled.
Web: Name UCI: strongswan.@connection[X].name Opt: name	Specifies a name for the tunnel.
Web: Autostart Action UCI: strongswan.@connection[X].auto Opt: auto	Specifies when the tunnel is initiated. start: On start up. route: When traffic routes this way. add: Loads a connection without starting it. ignore: Ignores the connection. always: Actively retries to establish the tunnel if it went down.
Web: Connection Type UCI: strongswan.@connection[X].type Opt: type	Defines the type of IPSec connection. tunnel: Connection uses tunnel mode. transport: Connection uses transport mode. pass: Connection does not perform any IPSec processing. drop: Connection drops all the packets.
Web: Remote GW Address UCI: strongswan.@connection[X]. remoteaddress Opt: remoteaddress	Sets the public IP address of the remote peer. (leave it blank for DMVPN)
Web: Local ID UCI: strongswan.@connection[X].localid Opt: localid	Defines the local peer identifier. (leave it blank for DMVPN)
Web: Remote ID UCI: strongswan.@connection[X].remoteid Opt:remoteid	Defines the remote peer identifier. (leave it blank for DMVPN)
Web: Local LAN IP Address UCI: strongswan.@connection[X]. locallan Opt: locallan	Defines the local IP of LAN. (leave it blank for DMVPN)
Web: Local LAN IP Address Mask UCI: strongswan.@connection[X]. locallanmask Opt: locallanmask	Defines the subnet of local LAN. (leave it blank for DMVPN)

<table>
<tr><td>Web: Remote LAN IP Address
UCI: strongswan.@connection[X].
remotelan
Opt:remotelan</td><td>Defines the IP address of LAN serviced by remote peer.
(leave it blank for DMVPN)</td></tr>
<tr><td>Web: Remote LAN IP Address Mask
UCI: strongswan.@connection[X].
remotelanmask
Opt:remotelanmask</td><td>Defines the Subnet of remote LAN.
(leave it blank for DMVPN)</td></tr>
<tr><td>Web: Local Protocol
UCI:
strongswan.@connection[X].localproto
Opt: localproto</td><td>Restricts the connection to a single protocol on the local side.</td></tr>
<tr><td>Web: Local Port
UCI:
strongswan.@connection[X].localport
Opt: localport</td><td>Restricts the connection to a single port on the local side.</td></tr>
<tr><td>Web: Remote Protocol
UCI:
strongswan.@connection[X].remotepro
to
Opt:remoteproto</td><td>Restricts the connection to a single protocol on the remote side.</td></tr>
<tr><td>Web: Remote Port
UCI:
strongswan.@connection[X].remotepor
t
Opt: remoteport</td><td>Restricts the connection to a single port on the remote side.</td></tr>
<tr><td>Web: Authby
UCI:
strongswan.@connection[X].authby
Opt: authby</td><td>Defines how the two secure gateways should authenticate.
Note: using aggressive mode along with PSK authentication is unsecure and should be avoided.
<table>
<tr><td>Pubkey</td><td>For public key signatures.</td></tr>
<tr><td>Rsasig</td><td>For RSA digital signatures.</td></tr>
<tr><td>ecdsasig</td><td>For Elliptic Curve DSA signatures.</td></tr>
<tr><td>Psk</td><td>Using a preshared key.</td></tr>
<tr><td>xauthrsasi
g</td><td>Enables eXtended Authentication (XAuth) with addition to RSA signatures.</td></tr>
<tr><td>xauthpsk</td><td>Using extended authentication and preshared key.</td></tr>
<tr><td>never</td><td>Can be used if negotiation is never to be attempted or accepted (shunt connections).</td></tr>
</table></td></tr>
<tr><td>Web: XAuth Identity
UCI:
strongswan.@connection[X].xauth_ide
ntity
Opt: xauth_identity</td><td>Defines Xauth ID.</td></tr>
</table>

Web: IKE Algorithm UCI: strongswan.@connection[X].ike Opt: ike	Specifies the IKE algorithm to use. The format is: encAlgo | authAlgo | DHGroup: • encAlgo: o 3des o aes o serpent o twofish o blowfish • authAlgo: o md5 o sha o sha2 • DHGroup: o modp1024 o modp1536 o modp2048 o modp3072 o modp4096 o modp6144 o modp8192 For example, a valid IKE algorithm is: aes128-sha-modp1536.
Web: ESP algorithm UCI: strongswan.@connection[X].esp Opt: esp	Specifies the esp algorithm to use. The format is: encAlgo | authAlgo | DHGroup • encAlgo: o 3des o aes o serpent o twofish o blowfish • authAlgo: o md5 o sha o sha2 • DHGroup: o modp1024 o modp1536 o modp2048 o modp3072 o modp4096 o modp6144 o modp8192 For example, a valid encryption algorithm is: aes128-sha-modp1536. If no DH group is defined then PFS is disabled.

<table>
<tr><td>Web: WAN Interface
UCI: strongswan.@connection[X].waniface
Opt: waniface</td><td>This is a space separated list of the WAN interfaces the router will use to establish a tunnel with the secure gateway.
On the web, a list of the interface names is automatically generated. If you want to specify more than one interface use the “custom” value.
Example: If you have a 3G WAN interface called ‘wan and a WAN ADSL interface called ‘dsl’ and wanted to use one of these interfaces for this IPSec connection, you would use: ‘wan adsl’.</td></tr>
<tr><td>Web: IKE Life Time
UCI: strongswan.@connection[X].ikelifetime
Opt:ikelifetime</td><td>Specifies how long the keyring channel of a connection (ISAKMP or IKE SA) should last before being renegotiated.<table><tr><td>3h</td><td></td></tr><tr><td>Timespec</td><td>1d, 3h, 25m, 10s.</td></tr></table></td></tr>
<tr><td>Web: Key Life
UCI: strongswan.@connection[X].keylife
Opt: keylife</td><td>Specifies how long a particular instance of a connection (a set of encryption/authentication keys for user packets) should last, from successful negotiation to expiry.
Normally, the connection is renegotiated (via the keying channel) before it expires (see rekeymargin).<table><tr><td>1h</td><td></td></tr><tr><td>Timespec</td><td>1d, 1h, 25m, 10s.</td></tr></table></td></tr>
<tr><td>Web: Rekey Margin
UCI: strongswan.@connection[X].rekeymargin
Opt: rekeymargin</td><td>Specifies how long before connection expiry or keying-channel expiry should attempt to negotiate a replacement begin.
Relevant only locally, other end need not agree on it.<table><tr><td>9m</td><td></td></tr><tr><td>Timespec</td><td>1d, 2h, 9m, 10s.</td></tr></table></td></tr>
<tr><td>Web: Keyring Tries
UCI: strongswan.@connection[X].keyringtries
Opt: keyringtries</td><td>Specifies how many attempts (a positive integer or %forever) should be made to negotiate a connection, or a replacement for one, before giving up. The value %forever means 'never give up'. Relevant only locally, other end need not agree on it.</td></tr>
<tr><td>Web: DPD Action
UCI: strongswan.@connection[X].dpdaction
Opt: dpdaction</td><td>Defines DPD (Dead Peer Detection) action.<table><tr><td>None</td><td>Disables DPD.</td></tr><tr><td>Clear</td><td>Clear down the tunnel if peer does not respond. Reconnect when traffic brings the tunnel up.</td></tr><tr><td>Hold</td><td>Clear down the tunnel and bring up as soon as the peer is available.</td></tr><tr><td>Restart</td><td>Restarts DPD when no activity is detected.</td></tr></table></td></tr>
<tr><td>Web: DPD Delay
UCI: strongswan.@connection[X].dpddelay
Opt: dpddelay</td><td>Defines the period time interval with which R_U_THERE messages and INFORMATIONAL exchanges are sent to the peer.
These are only sent if no other traffic is received.<table><tr><td>30s</td><td></td></tr><tr><td>Timespec</td><td>1d, 2h, 25m, 10s.</td></tr></table></td></tr>
</table>

Web: DPD Timeout UCI: strongswan.@connection[X].dpdtimeout Opt: dpdtimeout	Defines the timeout interval, after which all connections to a peer are deleted in case of inactivity. 150s Timespec: 1d, 2h, 25m, 10s.

Table 69: Information table for IPSec connections settings

22.4.3 Configure secrect settings

Each tunnel requires settings to configure how the local end point of the tunnel proves its identity to the remote end point.

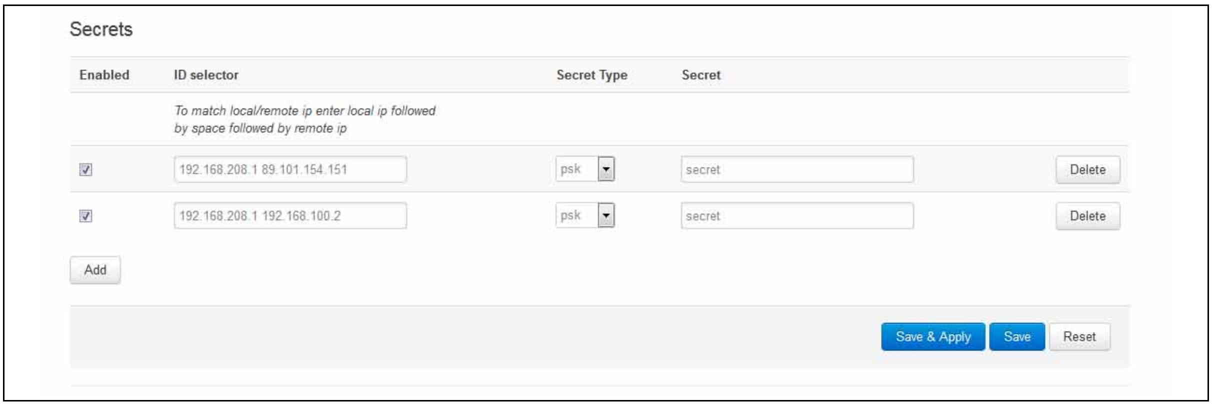


Figure 92: IPSec secrets settings

Web Field / UCI / Package Option	Description
Web: Enabled UCI: strongswan.@secret[X].enabled Opt: enabled	Defines whether this set of credentials is to be used or not. 0: Disabled. 1: Enabled.
Web: ID selector UCI: strongswan.@secret[X].idtype Opt: idtype	Defines whether IP address or userfqdn is used.
Web: ID selector UCI: strongswan.@secret[X].localaddress Opt: localaddress	Defines the local address this secret applies to.
Web: ID selector UCI: strongswan.@secret[X]. remoteaddress Opt: remoteaddress	Defines the remote address this secret applies to.
Web: N/A UCI: strongswan.@secret[X].userfqnd Opt: userfqnd	FQDN or Xauth name used of Extended Authentication. This must match xauth_identity from the configuration connection section.

<table>
<tr><td>Web: Secret Type
UCI: strongswan.@secret[X].secrettype
Opt: secrettype</td><td>Specifies the authentication mechanism to be used by the two peers.<table><tr><td>Psk</td><td>Preshared secret</td></tr><tr><td>Pubkey</td><td>Public key signatures</td></tr><tr><td>Rsasig</td><td>RSA digital signatures</td></tr><tr><td>Ecdsasig</td><td>Elliptic Curve DSA signatures</td></tr><tr><td>Xauth</td><td>Extended authentication</td></tr></table></td></tr>
<tr><td>Web: Secret
UCI: strongswan.@secret[X].secret
Opt: secret</td><td>Defines the secret.</td></tr>
</table>

Table 70: Information table for IPSec secret settings

22.5 Configuring an IPSec template to use with DMVPN

The following example shows how to configure an IPSec connection template to use with DMVPN.

```
# Commands
touch /etc/config/strongswan
uci set strongswan.general=general
uci set strongswan.general.enabled=yes
uci set strongswan.general.strictcrlpolicy=no
uci set strongswan.general.uniqueids=yes
uci set strongswan.general.cachecrls=yes
uci set strongswan.general.nattraversal=yes
uci add strongswan connection
uci set strongswan.@connection[0].enabled=yes
uci set strongswan.@connection[0].name=dmvpn
uci set strongswan.@connection[0].type=transport
uci set strongswan.@connection[0].localproto=gre
uci set strongswan.@connection[0].remoteproto=gre
uci set strongswan.@connection[0].ike=aes-sha1-modp1024
uci set strongswan.@connection[0].esp=aes128-sha1
uci set strongswan.@connection[0].waniface=lan4
uci set strongswan.@connection[0].auto=ignore
uci set strongswan.@connection[0].ikelifetime=28800s
uci set strongswan.@connection[0].keylife=300s
uci set strongswan.@connection[0].rekeymargin=30s
uci set strongswan.@connection[0].keyingtries=%forever
```

```
uci set strongswan.@connection[0].dpdaction=hold
uci set strongswan.@connection[0].dpddelay=30s
uci set strongswan.@connection[0].dpdtimeout=150s

uci add strongswan secret
uci set strongswan.@secret[0].enabled=yes
uci set strongswan.@secret[0].secrettype=psk
uci set strongswan.@secret[0].secret=secret
```

This will create package strongswan.

```
 config general 'general'
option enabled 'yes'
option strictcrlpolicy 'no'
option uniqueids 'yes'
option cachecrls 'yes'
option nattraversal 'yes'

config connection
option enabled 'yes'
option name 'dmvpn'
option type 'transport'
option localproto 'gre'
option remoteproto 'gre'
option ike 'aes-sha1-modp1024'
option esp 'aes128-sha1'
option waniface 'lan4'
option auto 'ignore'
option ikelifetime '28800s'
option keylife '300s'
option rekeymargin '30s'
option keyingtries '%forever'
option dpdaction 'hold'
option dpddelay '30s'
option dpdtimeout '150s'
```

```
config secret
option enabled 'yes'
option secrettype 'psk'
option secret 'secret'
```

22.6 IPSec diagnostics using the web interface

22.6.1 IPSec status

In the top menu, click **Status -> IPSec**. The IPSec Connections page appears.

IPsec Connections

Name	IKE					SA			
	Status	Remote	Established	Encryption	Integrity	Status	Policy	Data In/Out	Rekey in
dmvpn_213_233_148_2	ESTABLISHED	213.233.148.2	2 hours ago	3DES_CBC	HMAC_MD5_96	INSTALLED			
dmvpn_89_101_154_151	ESTABLISHED	89.101.154.151	2 hours ago	3DES_CBC	HMAC_MD5_96	INSTALLED			

Figure 93: The IPSec connections page

In the Name column, the syntax contains the IPSec Name defined in package dmvpn and the remote IP address of the hub, or the spoke separated by an underscore; for example, dmvpn_213.233.148.2.

22.7 IPSec diagnostics using UCI

22.7.1 IPSec configuration

To view IPSec configuration via UCI, enter:

```
root@VA_router:~# uci export strongswan
```

To restart strongSwan, enter:

```
root@VA_router:~# etc/init.d/strongswan restart
```

22.7.2 IPSec status

To view IPSec status, enter:

```
root@VA_router:~# ipsec statusall
Security Associations (1 up, 0 connecting):
```

```
dmvpn_89_101_154_151[1]: ESTABLISHED 2 hours ago,
10.68.234.133[10.68.234.133]...89.101.154.151[89.101.154.151]
dmvpn_89_101_154_151{1}:  REKEYING, TRANSPORT, expires in 55 seconds
dmvpn_89_101_154_151{1}:   10.68.234.133/32[gre] === 192.168./32[gre]
dmvpn_89_101_154_151{1}:  INSTALLED, TRANSPORT, ESP in UDP SPIs: cca7b970_i
d874dc90_o
dmvpn_89_101_154_151{1}:   10.68.234.133/32[gre] === 89.101.154.151/32[gre]
```

To view a list of IPSec commands, enter:

```
root@VA_router:~# ipsec –help
```

23 Configuring a GRE interface

General Routing Encapsulation (GRE) is a tunnelling protocol used for encapsulation of other communication protocols inside point to point links over IP.

23.1 Configuration packages used

Package	Sections
network	interface

23.2 Creating a GRE connection using the web interface

To create GRE interfaces through the web interface, in the top menu, select **Network -> Interfaces**.

There are three sections in the Interfaces page.

Section	Description
Interface Overview	Shows existing interfaces and their status. You can create new, and edit existing interfaces here.
Port Map	In this section you can map device ports to Ethernet interfaces. Ports are marked with capital letters starting with 'A'. Type in space separated port numbers in the port map fields.
ATM Bridges	ATM bridges expose encapsulated Ethernet in AAL5 connections as virtual Linux network interfaces, which can be used in conjunction with DHCP or PPP to dial into the provider network.

In the Interface Overview section, click **Add new interface**. The Create Interface page appears.

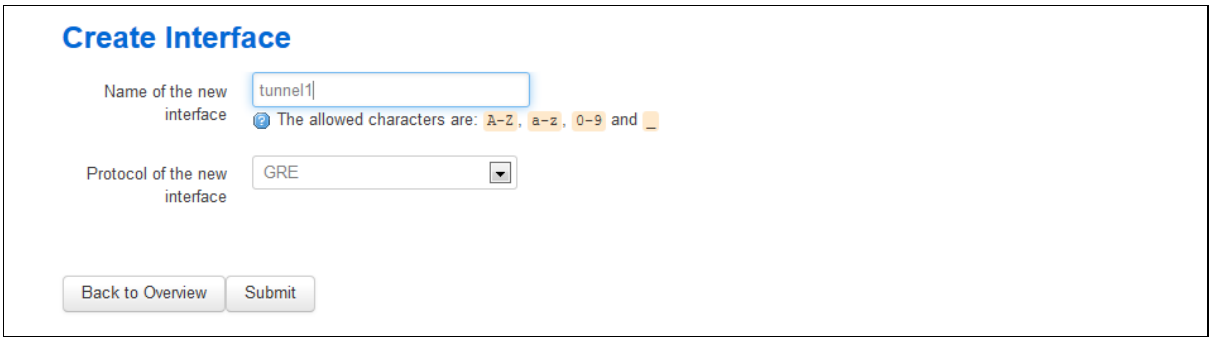


Figure 94: The create interface page

Web Field/UCI/Package Option	Description
Web: Name of the new interface UCI: network. .<if name> Opt: config interface	Assigns a logical name to the GRE tunnel, The network interface section will be assigned this name <if name>. Type the name of the new interface. Allowed characters are A-Z, a-z, 0-9 and _. Must be less than 11 characters.

<table>
<tr><td>Web: Protocol of the new interface
UCI: network.<if name>.proto
Opt: proto</td><td>Specifies what protocol the interface will operate on. Select GRE.
<table>
<tr><th>Option</th><th>Description</th></tr>
<tr><td>Static</td><td>Static configuration with fixed address and netmask.</td></tr>
<tr><td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td></tr>
<tr><td>Unmanaged</td><td>Unspecified</td></tr>
<tr><td>IPv6-in-IPv4 (RFC4213)</td><td>Used with tunnel brokers.</td></tr>
<tr><td>IPv6-over-IPv4</td><td>Stateless IPv6 over IPv4 transport.</td></tr>
<tr><td>GRE</td><td>Generic Routing Encapsulation protocol</td></tr>
<tr><td>IOT</td><td></td></tr>
<tr><td>L2TP</td><td>Layer 2 Tunnelling Protocol</td></tr>
<tr><td>PPP</td><td>Point-to-Point protocol</td></tr>
<tr><td>PPPoE</td><td>PPP over Ethernet</td></tr>
<tr><td>PPPoATM</td><td>PPP over ATM</td></tr>
<tr><td>LTE/UMTS/GPRS/EV-DO</td><td>CDMA, UMTS or GPRS connection using an AT-style 3G modem.</td></tr>
</table></td></tr>
<tr><td>Web: Create a bridge over multiple interfaces
UCI: network.<if name>
Opt: n/a</td><td>Not applicable for GRE.</td></tr>
<tr><td>Web: Cover the following interface
UCI: network.<if name>
Opt:n/a</td><td>Not applicable for GRE.</td></tr>
</table>

Table 71: Information table for the create new interface page

Click **Submit**. The Common Configuration page appears. There are three sections in the Common Configurations page.

Section	Description
General Setup	Configure the basic interface settings such as protocol, IP address, mask length, local interface, remote IP address, TTL, tunnel key and MTU.
Advanced Settings	'Bring up on boot' and 'monitor interface state' settings.
Firewall settings	Assign a firewall zone to the connection.

23.2.1 GRE connection: common configuration - general setup

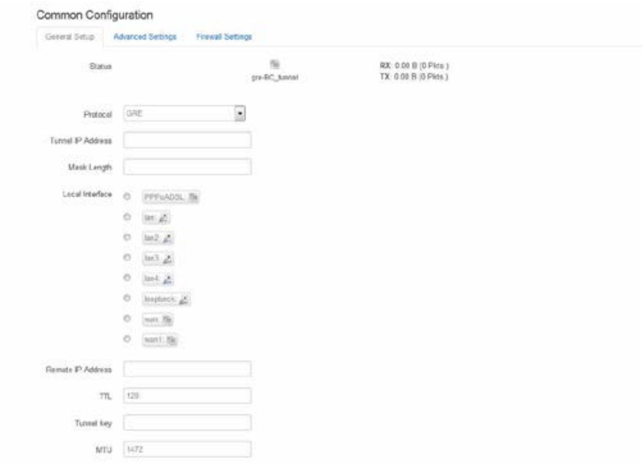


Figure 95: The GRE common configuration page

<table>
<tr><th>Web Field/UCI/Package Option</th><th>Description</th></tr>
<tr><td>Web: Protocol of the new interface
UCI: network.<if name>.proto
Opt: proto</td><td>Shows the protocol the interface will operate on. GRE should be currently selected.</td></tr>
<tr><td>Web: Tunnel IP Address
UCI: network.<if name>.ipaddr
Opt: ipaddr</td><td>Configures local IP address of the GRE interface.</td></tr>
<tr><td>Web: Mask Length
UCI: network.<if name>.mask_length
Opt: mask_length</td><td>Subnet mask, in CIDR notation, to be applied to the tunnel. Typically '30' for point-to-point tunnels.<table><tr><td>24</td><td></td></tr><tr><td>Range</td><td>0 - 30</td></tr></table></td></tr>
<tr><td>Web: Local Interface
UCI: network.<if name>.local_interface
Opt: local_interface</td><td>Specifies which interface is going to be linked with the GRE tunnel interface (optional).</td></tr>
<tr><td>Web: Remote IP address
UCI: network.<if name>.remote_ip
Opt: remote_ip</td><td>For point to point tunnels specifies Remote IP address.</td></tr>
<tr><td>Web: TTL
UCI: network.<if name>.ttl
Opt: ttl</td><td>Sets Time-To-Live value on the interface.<table><tr><td>128</td><td></td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
<tr><td>Web: Tunnel key
UCI: network.<if name>.key
Opt: key</td><td>Sets GRE tunnel ID key (optional).
Usually an integer.</td></tr>
<tr><td>Web: MTU
UCI: network.<if name>.mtu
Opt: mtu</td><td>Configures MTU (maximum transmission unit) size of PDUs using this interface.<table><tr><td>1472</td><td></td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
</table>

Table 72: Information table for GRE

23.2.2 GRE connection: common configuration-advanced settings

Figure 96: GRE advanced settings page

Web Field / UCI / Package Option	Description
Web: Bring up on boot UCI: network.<if name>.auto Opt: auto	Enables the interface to connect automatically on boot up. 0 — Disabled. 1 — Enabled.
Web: Monitor interface state UCI: network.<if name>.monitored Opt: monitored	Enabled if status of interface is presented on Monitoring platform. 0 — Disabled. 1 — Enabled.

Table 73: Information table for GRE advanced settings

23.2.3 GRE connection: firewall settings

Use this section to select the firewall zone you want to assign to this interface.

Select **unspecified** to remove the interface from the associated zone or fill out the create field to define a new zone and attach the interface to it.

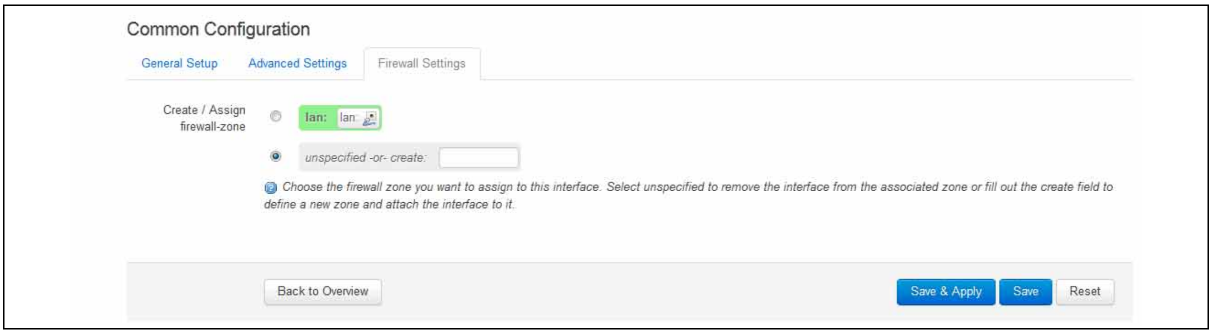


Figure 97: GRE firewall settings

Click **Save and Apply**. This will save the current settings and return you to the Interface Overview page. To configure further settings on the GRE interface select **EDIT** for the relevant GRE interface.

23.2.4 GRE connection: adding a static route

After the GRE interface has been configured, a static route needs to be configured to route the desired traffic over the GRE tunnel. To do this, go to

Network->Static Routes. For more information, read the chapter ‘Configuring Static Routes’.

23.3 GRE configuration using command line

The configuration file is stored at:

/etc/config/network

For the examples below tunnel1 is used as the interface logical name.

23.4 GRE configuration using UCI

```
root@VA_router:~# uci show network
network.tunnel1=interface
network.tunnel1.proto=gre
network.tunnel1.monitored=0
network.tunnel1.ipaddr=172.255.255.2
network.tunnel1.mask_length=24
network.tunnel1.local_interface=wan
network.tunnel1.remote_ip=172.255.255.100
network.tunnel1.ttl=128
network.tunnel1.key=1234
network.tunnel1.mtu=1472
network.tunnel1.auto=1
```

23.4.1 GRE configuration using package options

```
root@VA_router:~# uci export network
config interface 'tunnel1'
      option proto 'gre'
      option monitored ‘0’
      option ipaddr '172.255.255.2'
      option mask_length '24'
      option local_interface 'wan'
      option remote_ip ‘172.255.255.100’
      option ttl '128'
      option key '1234'
```

```
      option mtu '1472'
      option auto ‘1’
```

To change any of the above values use `uci set` command.

23.5 GRE diagnostics

23.5.1 GRE interface status

To show the current running interfaces, enter:

```
root@VA_router:~# ifconfig
base0       Link encap:Ethernet  HWaddr 00:00:00:00:01:01
            inet6 addr: fe80::200:ff:fe00:101/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST  MTU:1504  Metric:1
            RX packets:39810 errors:0 dropped:0 overruns:0 frame:0
            TX packets:365 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:10889090 (10.3 MiB)  TX bytes:68820 (67.2 KiB)
eth4        Link encap:Ethernet  HWaddr 00:1E:10:1F:00:00
            inet addr:10.68.66.54  Bcast:10.68.66.55  Mask:255.255.255.252
            inet6 addr: fe80::21e:10ff:fe1f:0/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
            RX packets:81 errors:0 dropped:0 overruns:0 frame:0
            TX packets:127 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:8308 (8.1 KiB)  TX bytes:12693 (12.3 KiB)
gre-Tunnel1 Link encap:UNSPEC  HWaddr 0A-44-42-36-DB-B0-00-48-00-00-00-00-
00-00-00-00
            inet addr:13.13.13.2  Mask:255.255.255.248
            inet6 addr: fe80::5efe:a44:4236/64 Scope:Link
            UP RUNNING MULTICAST  MTU:1472  Metric:1
            RX packets:7 errors:0 dropped:0 overruns:0 frame:0
            TX packets:7 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:912 (912.0 B)  TX bytes:884 (884.0 B)
```

```
lo          Link encap:Local Loopback
            inet addr:127.0.0.1  Mask:255.0.0.0
            inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING  MTU:16436  Metric:1
            RX packets:1465 errors:0 dropped:0 overruns:0 frame:0
            TX packets:1465 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:166202 (162.3 KiB)  TX bytes:166202 (162.3 KiB)
```

To display a specific GRE interface enter: ifconfig gre-<if name>:

```
root@VA_router:~# ifconfig gre-Tunnel1
gre-Tunnel1   Link encap:UNSPEC  HWaddr 0A-44-42-36-00-00-7F-E2-00-00-00-
00-00-00-00-00
              inet addr:13.13.13.2  Mask:255.255.255.248
              inet6 addr: fe80::5efe:a44:4236/64 Scope:Link
              UP RUNNING MULTICAST  MTU:1472  Metric:1
              RX packets:7 errors:0 dropped:0 overruns:0 frame:0
              TX packets:7 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:0
              RX bytes:912 (912.0 B)  TX bytes:8GRE route status
```

To show the current GRE route status, enter:

```
root@VA_router:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
0.0.0.0         10.68.66.53     0.0.0.0         UG    0      0        0
eth4
0.0.0.0         13.13.13.1      0.0.0.0         UG    1      0        0
gre-Tunnel1
10.68.66.52     0.0.0.0         255.255.255.252 U     0      0        0
eth4
13.13.13.0      0.0.0.0         255.255.255.248 U     0      0        0
gre-Tunnel1
```

```
172.19.101.3    13.13.13.1      255.255.255.255 UGH   0      0        0
gre-Tunnel1
```

Note: a GRE route will only be displayed in the routing table when the interface is up.

24 Dynamic Multipoint Virtual Private Network (DMVPN)

Dynamic Multipoint Virtual Private Network (DMVPN) is a scalable method of creating VPN IPSec Networks. DMVPN is a suite of three protocols: NHRP, GRE and IPSec, used to dynamically create VPN tunnels between different endpoints in the network without having to pre-configure each device with VPN details of the rest of endpoints in the network.

24.1 Prerequisites for configuring DMVPN

Before configuring DMVPN, you must first configure:

- A GRE interface. Read the previous chapter,' Configuring GRE interfaces'.
- An IPSec connection to use as a template. Read the previous chapter, 'Configuring IPSec'.

24.2 Advantages of using DMVPN

- Using DMVPN eliminates the need of IPSec configuration to the physical interface. This reduces the number of lines of configuration required for a VPN development. For example, for a 1000-site deployment, DMVPN reduces the configuration effort at the hub from 3900 lines to 13.
- Adding new peers (spokes) to the VPN requires no changes at the hub.
- Better scalability of the network.
- Dynamic IP addresses can be used at the peers' site.
- Spokes can be connected in private or public network.
- NHRP NAT extension allows spoke-to-spoke tunnels to be built, even if one or more spokes is behind a Network Address Translation (NAT) device.
- New hubs can be added to the network to improve the performances and reliability.
- Ability to carry multicast and main routing protocols traffic (RIP, OSPF, BGP).
- DMVPN can be deployed using Activator, the Virtual Access automated provisioning system.
- Simplifies branch communications by enabling direct branch to branch connectivity.
- Simplifies configuration on the spoke routers. The same IPSec template configuration is used to create spoke-to-hub and spoke-to-spoke VPN IPSec tunnel.

- Improves business resiliency by preventing disruption of business-critical applications and services by incorporating routing with standards-based IPsec technology.

24.3 DMVPN scenarios

Scenario 1: Spoke1, spoke2 and a hub are in the same public or private network.

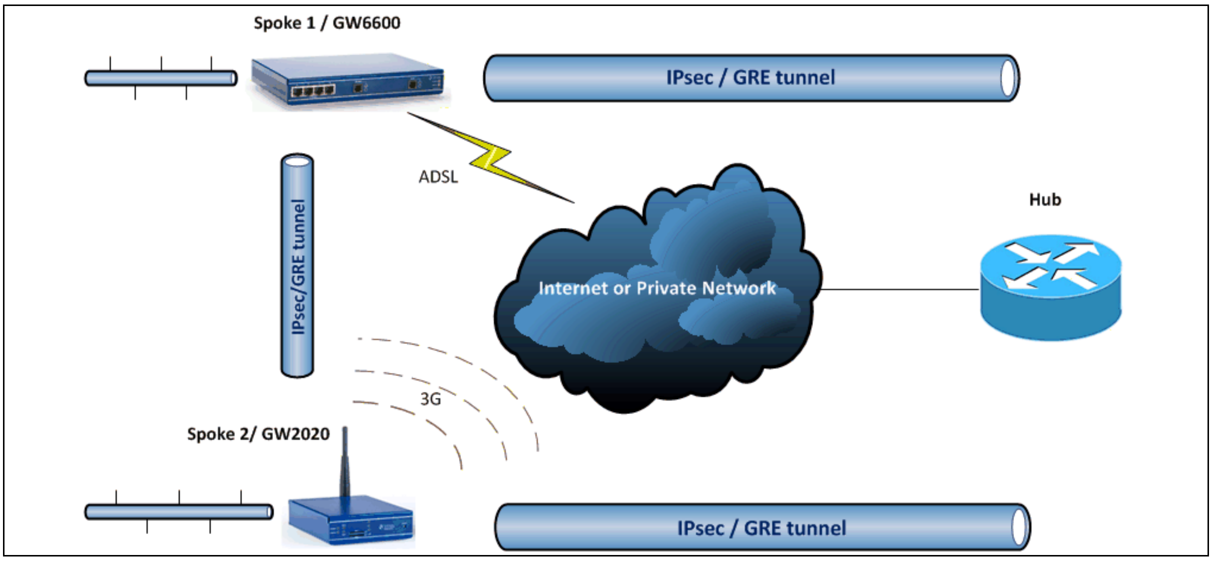


Figure 98: Network diagram for DMVPN spoke to spoke

- Spoke1 and spoke2 connect on their WAN interface: ADSL, 3G and initiate main mode IPSec in transport mode to the hub.
- After an IPSec tunnel is established, spokes register their NHRP membership with the hub.
- GRE tunnels come up.
- Hub caches the GRE tunnel and real IP addresses of each spoke.
- When spoke1 wants to talk to spoke2, it sends an NHRP resolution request to the hub.
- The hub checks its cache table and forwards that request to spoke2.
- Spoke2 caches spoke1's GRE and real IP address and sends an NHRP resolution reply via the hub.
- Spoke1 receives an NHRP resolution reply and updates its NHRP table with spoke2 information. Then it initiates VPN IPSec connection to spoke2.
- When an IPSec tunnel is established, spoke1 and spoke2 can send traffic directly to each other.

Scenario 2: Spoke1 is in a private (NAT-ed) network, spoke2 and hub are in public network.

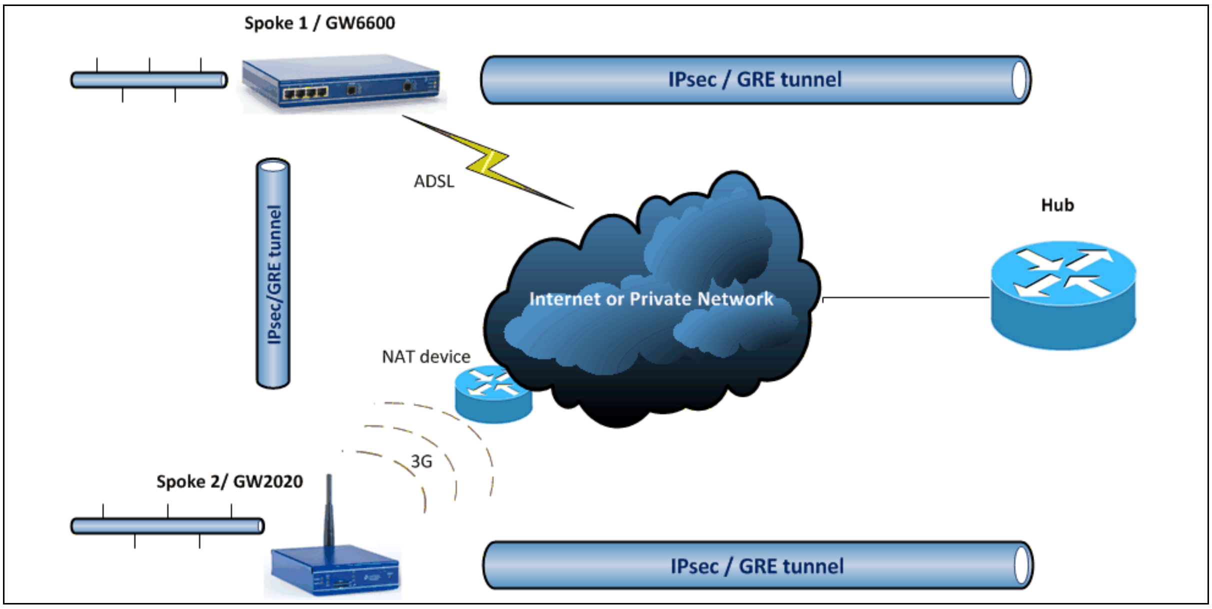


Figure 99: Network diagram for DMVPN spoke behind NAT

- Spoke1 sends an NHRP registration request to the hub.
- Hub receives this request and compares the source tunnel address of the spoke with the source of the packet.
- Hub sends an NHRP registration reply with a NAT extension to spoke1.
- The NAT extension informs spoke1 that it is behind the NAT-ed device.
- Spoke1 registers its pre- and post-NAT address.
- When spoke1 wants to talk to spoke2, it sends an NHRP resolution request to the hub.
- Hub checks its cache table and forwards that request to spoke2.
- Spoke2 caches spoke1's GRE pre- and post-NAT IP address and sends an NHRP resolution reply via the hub.
- Spoke1 receives the NHRP resolution reply and updates its NHRP table with spoke2 information. It initiates a VPN IPSec connection to spoke2.
- When the IPSec tunnel is established, spoke1 and spoke2 can send traffic directly to each other.

Note: If an IPSec tunnel fails to be established between the spokes then packets between the spokes are sent via the hub.

24.4 Configuration packages used

Package	Sections
network	For configuring the GRE tunnels.
strongswan	For enabling and configuring the IPSec connection template
dmvpn	

24.5 Configuring DMVPN using the web interface

The DMVPN section contains fields required to configure the parameters relative to the DMVPN Hub. These are used for DMVPN tunnels, such as GRE tunnels, GRE tunnel remote IP, DMVPN Hub IP and password.

24.5.1 DMVPN general settings

In the top menu, select **Network -> DMVPN**. The DMVPN page appears. There are two sections: General and DMVPN Hub Settings.

Figure 100: The DMVPN general section

Web Field / UCI / Package Option	Description
Web: Enable DMVPN UCI: dmvpn.common.enabled Opt: enable	Enables DMVPN. 0 — Disabled. 1 — Enabled.
Web: IPSec template connection UCI: dmvpn.common.ipsec_template_name Opt: ipsec_template_name	Selects the IPSec connection, defined in strongSwan, to be used as a template.

Table 74: Information table for DMVPN general settings

24.5.2 DMVPN hub settings

Figure 101: The DMVPN hub settings

Web Field / UCI / Package Option	Description
Web: GRE Interface UCI: dmvpn.@interface[X].gre_interface Opt: gre_interface	Specifies which GRE interface will be used with this DMVPN configuration.
Web: GRE Remote Endpoint IP Address UCI: dmvpn.@interface[X].gre_endpoint_ip Opt: gre_endpoint_ip	Configures the GRE IP address of the hub.
Web: GRE Remote Endpoint Mask Length UCI: dmvpn.@interface[X].gre_endpoint_mask_length Opt: gre_endpoint_mask_length	Configures the length of the mask of the GRE interface on the hub. For example if the mask is 255.255.0.0 the length will be 16.
Web: DMVPN Hub IP Address UCI: dmvpn.@interface[X].nhs_ip Opt: nhs_ip	Configures the physical IP address for the DMVPN hub.
Web: NHRP Authentication UCI: dmvpn.@interface[X].cisco_auth Opt: cisco_auth	Enables authentication on NHRP. The password will be applied in plaintext to the outgoing NHRP packets. Maximum length is 8 characters.
Web: NHRP Holding Time UCI: dmvpn.@interface[X].holding_time Opt: holding_time	Timeout for cached NHRP requests.
Web: Use As Default Route UCI : dmvpn.@interface[X].defaultroute Opt: defaultroute	Adds a default route into tunnel interface. 0 – Disabled. 1 – Enabled.
Web: Default Route Metric UCI: dmvpn.@interface[X].defaultroutemetric Opt: defaultroutemetric	Metric to use for the default route.

Web: LED state indication UCI: dmvpn.@interface[X].led Opt: led	LED to use for indicating if the VPN is up.

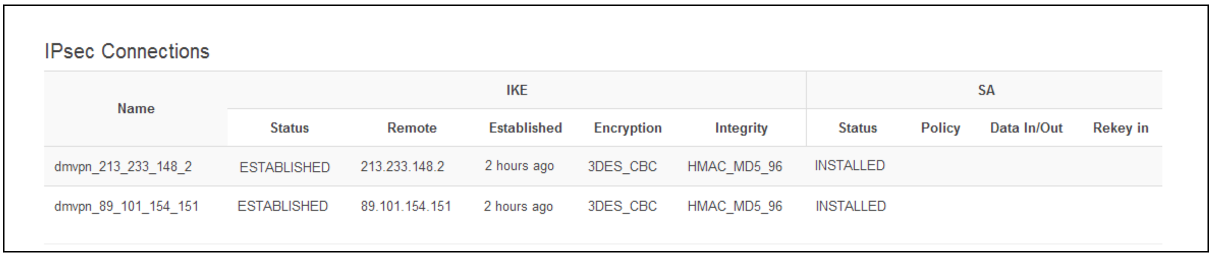
Table 75: Information table for DMVPN hub settings

24.5.3 Configuring an IPSec template for DMVPN using the web interface

Configuring an IPSec template is covered in the chapter 'Configuring IPSec'.

24.6 DMVPN diagnostics

In the top menu, click **Status -> IPSec**. The IPSec Connections page appears.



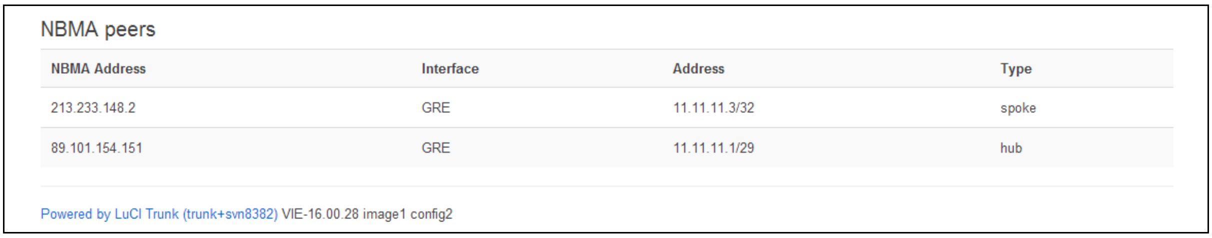
IPsec Connections

Name	IKE					SA			
	Status	Remote	Established	Encryption	Integrity	Status	Policy	Data In/Out	Rekey in
dmvpn_213_233_148_2	ESTABLISHED	213.233.148.2	2 hours ago	3DES_CBC	HMAC_MD5_96	INSTALLED			
dmvpn_89_101_154_151	ESTABLISHED	89.101.154.151	2 hours ago	3DES_CBC	HMAC_MD5_96	INSTALLED			

Figure 102: The IPSec connections page

In the Name column, the syntax contains the IPSec name defined in package dmvpn and the remote IP address of the hub, or the spoke separated by an underscore; for example, dmvpn_213.233.148.2.

To check the status of DMVPN, in the top menu, click **Status -> DMVPN**.



NBMA peers

NBMA Address	Interface	Address	Type
213.233.148.2	GRE	11.11.11.3/32	spoke
89.101.154.151	GRE	11.11.11.1/29	hub

Powered by LuCI Trunk (trunk+svn8382) VIE-16.00.28 image1 config2

Figure 103: The NBMA peers page

To check DMVPN status, enter:

```
:~# opennhrpctl show
Status: ok
Interface: gre-GRE
Type: local
Protocol-Address: 11.11.11.7/32
Alias-Address: 11.11.11.3
Flags: up
```

```
Interface: gre-GRE
Type: local
Protocol-Address: 11.11.11.3/32
Flags: up
Interface: gre-GRE
Type: cached
Protocol-Address: 11.11.11.2/32
NBMA-Address: 178.237.115.129
NBMA-NAT-OA-Address: 172.20.38.129
Flags: used up
Expires-In: 0:18

Interface: gre-GRE
Type: static
Protocol-Address: 11.11.11.1/29
NBMA-Address: 89.101.154.151
Flags: up
```

<table>
<tr><th>Interface</th><th colspan="2">Description</th></tr>
<tr><td rowspan="10">Type</td><td>incomplete</td><td>Resolution request sent.</td></tr>
<tr><td>negative</td><td>Negative cached.</td></tr>
<tr><td>cached</td><td>Received/relayed resolution reply.</td></tr>
<tr><td>shortcut_route</td><td>Received/relayed resolution for route.</td></tr>
<tr><td>dynamic</td><td>NHC resolution.</td></tr>
<tr><td>dynamic_nhs</td><td>Dynamic NHS from dns-map.</td></tr>
<tr><td>static</td><td>Static mapping from config file.</td></tr>
<tr><td>dynamic_map</td><td>Static dns-map from config file.</td></tr>
<tr><td>local_route</td><td>Non-local destination, with local route.</td></tr>
<tr><td>local_addr</td><td>Local destination (IP or off-NBMA subnet).</td></tr>
<tr><td>Protocol Address</td><td colspan="2">Tunnel IP address</td></tr>
<tr><td>NBMA-Address</td><td colspan="2">Pre-NAT IP address if NBMA-NAT-OA-Address is present or real address if NAT is not present.</td></tr>
<tr><td>NBMA-NAT-OA-Address</td><td colspan="2">Post NAT IP address. This field is present when Address is translated in the network.</td></tr>
<tr><td rowspan="4">Flags</td><td>up</td><td>Can send all packets (registration ok).</td></tr>
<tr><td>unique</td><td>Peer is unique.</td></tr>
<tr><td>used</td><td>Peer is kernel ARP table.</td></tr>
<tr><td>lower-up</td><td>openhrp script executed successfully.</td></tr>
<tr><td>Expires-In</td><td colspan="2">Expiration time.</td></tr>
</table>

Table 76: Information table for DMVPN status

You can check IPSec status using UCI commands.

```
root@VA-router:~# ipsec status
Security Associations (1 up, 0 connecting):
dmvpn_89_101_154_151[1]: ESTABLISHED 2 hours ago,
10.68.234.133[10.68.234.133]...89.101.154.151[89.101.154.151]
dmvpn_89_101_154_151{1}:  REKEYING, TRANSPORT, expires in 55 seconds
dmvpn_89_101_154_151{1}:   10.68.234.133/32[gre] === 192.168./32[gre]
dmvpn_89_101_154_151{1}:  INSTALLED, TRANSPORT, ESP in UDP SPIs: cca7b970_i
d874dc90_o
dmvpn_89_101_154_151{1}:   10.68.234.133/32[gre] === 89.101.154.151/32[gre]
```

You can check DMVPN status using uci commands.

```
:~# opennhrpctl show
Status: ok

Interface: gre-GRE
Type: local
Protocol-Address: 11.11.11.7/32
Alias-Address: 11.11.11.3
Flags: up

Interface: gre-GRE
Type: local
Protocol-Address: 11.11.11.3/32
Flags: up
Interface: gre-GRE
Type: cached
Protocol-Address: 11.11.11.2/32
NBMA-Address: 178.237.115.129
NBMA-NAT-OA-Address: 172.20.38.129
Flags: used up
Expires-In: 0:18
Interface: gre-GRE
Type: static
Protocol-Address: 11.11.11.1/29
```

```
NBMA-Address: 89.101.154.151
Flags: up
```

25 Configuring firewall

The firewall itself is not required. It is a set of scripts which configure Netfilter. If preferred, you can use Netfilter directly to achieve the desired firewall behaviour.

Note: the UCI firewall exists to simplify the configuration of Netfilter for many scenarios, without requiring the knowledge to deal with the complexity of Netfilter.

The firewall configuration consists of several zones covering one or more interfaces. Permitted traffic flow between the zones is controlled by forwardings. Each zone can include multiple rules and redirects (port forwarding rules).

The Netfilter system is a chained processing filter where packets pass through various rules. The first rule that matches is executed often leading to another rule-chain until a packet hits either ACCEPT or DROP/REJECT.

Accepted packets pass through the firewall. Dropped packets are prohibited from passing. Rejected packets are also prohibited but an ICMP message is returned to the source host.

A minimal firewall configuration for a router usually consists of one 'defaults' section, at least two 'zones' (LAN and WAN) and one forwarding to allow traffic from LAN to WAN. Other sections that exist are 'redirects', 'rules' and 'includes'.

25.1 Configuration package used

Package	**Sections**
firewall	

25.2 Configuring firewall using the web interface

In the top menu, select **Network -> Firewall**. The Firewall page appears. It is divided into four sections: General Zone Settings, Port Forwards, Traffic Rules, and Custom Rules.

25.2.1 Firewall general settings

The General Zone, or defaults, section declares global firewall settings that do not belong to any specific zones. These default rules take effect last and more specific rules take effect first.

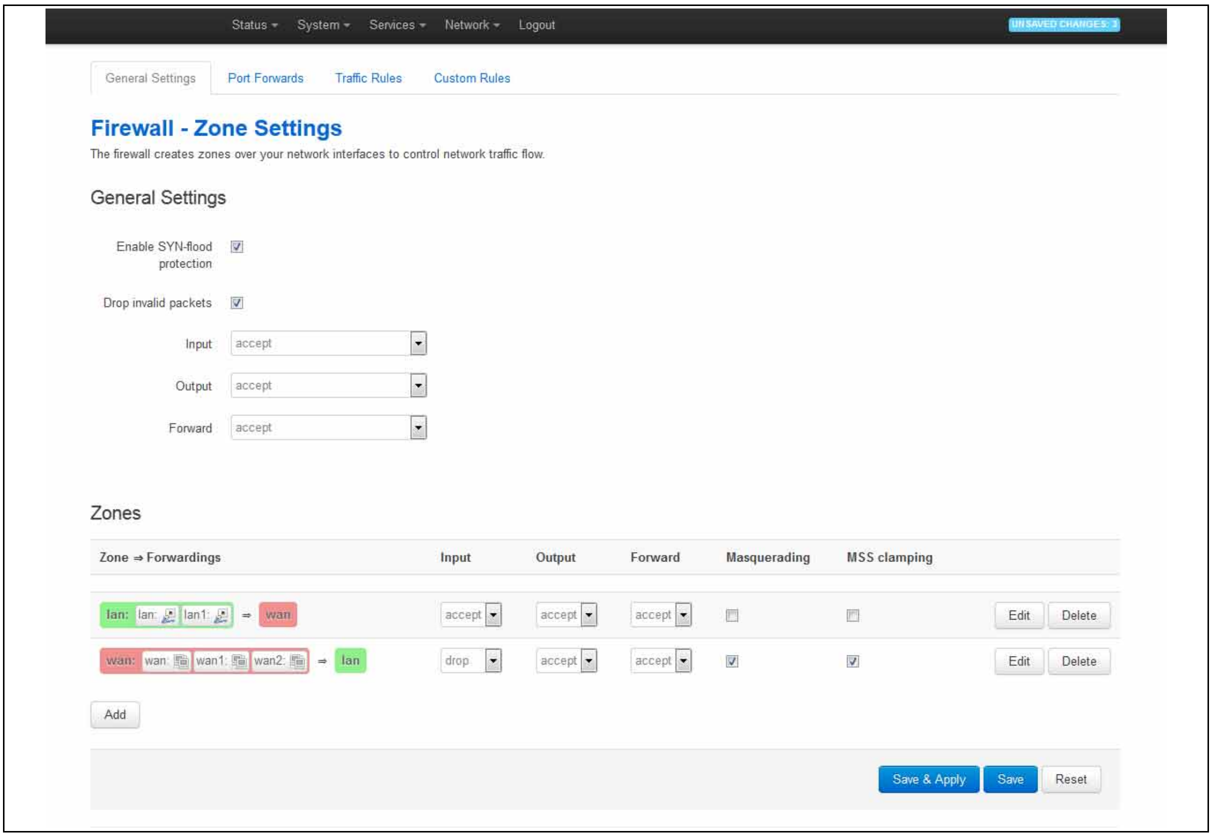


Figure 104: The firewall zone settings page

Web Field / UCI / Package Option	Description
Web: Enable SYN-flood protection UCI: firewall.defaults.syn_flood Opt: syn_flood	Enables SYN flood protection. 0 — Disabled. 1 — Enabled.
Web: Drop invalid packets UCI: firewall.defaults.drop_invalid Opt: drop_invalid	Drops packets not matching any active connection. 0 — Disabled. 1 — Enabled.
Web: Input UCI: firewall.defaults.input Opt: input	Default policy for the INPUT chain. Accept — Accepted packets pass through the firewall. Reject — Rejected packets are blocked by the firewall and ICMP message is returned to the source host. Drop — Dropped packets are blocked by the firewall.

Web: Output UCI: firewall.defaults.output Opt: output	Default policy for the Output chain. <table><tr><td>Accept</td><td>Accepted packets pass through the firewall.</td></tr><tr><td>Reject</td><td>Rejected packets are blocked by the firewall and ICMP message is returned to the source host.</td></tr><tr><td>Drop</td><td>Dropped packets are blocked by the firewall.</td></tr></table>
Web: Forward UCI: firewall.defaults.forward Opt: forward	Default policy for the Forward chain. <table><tr><td>Accept</td><td>Accepted packets pass through the firewall.</td></tr><tr><td>Reject</td><td>Rejected packets are blocked by the firewall and ICMP message is returned to the source host.</td></tr><tr><td>Drop</td><td>Dropped packets are blocked by the firewall.</td></tr></table>

Table 77: Information table for general settings page

25.3 Firewall zone settings

The zone section groups one or more interfaces and serves as a source or destination for forwardings, rules and redirects. Masquerading (NAT) of outgoing traffic is controlled on a per-zone basis. Click Edit to view a zone's settings.

25.3.1.1 Firewall zone: general settings

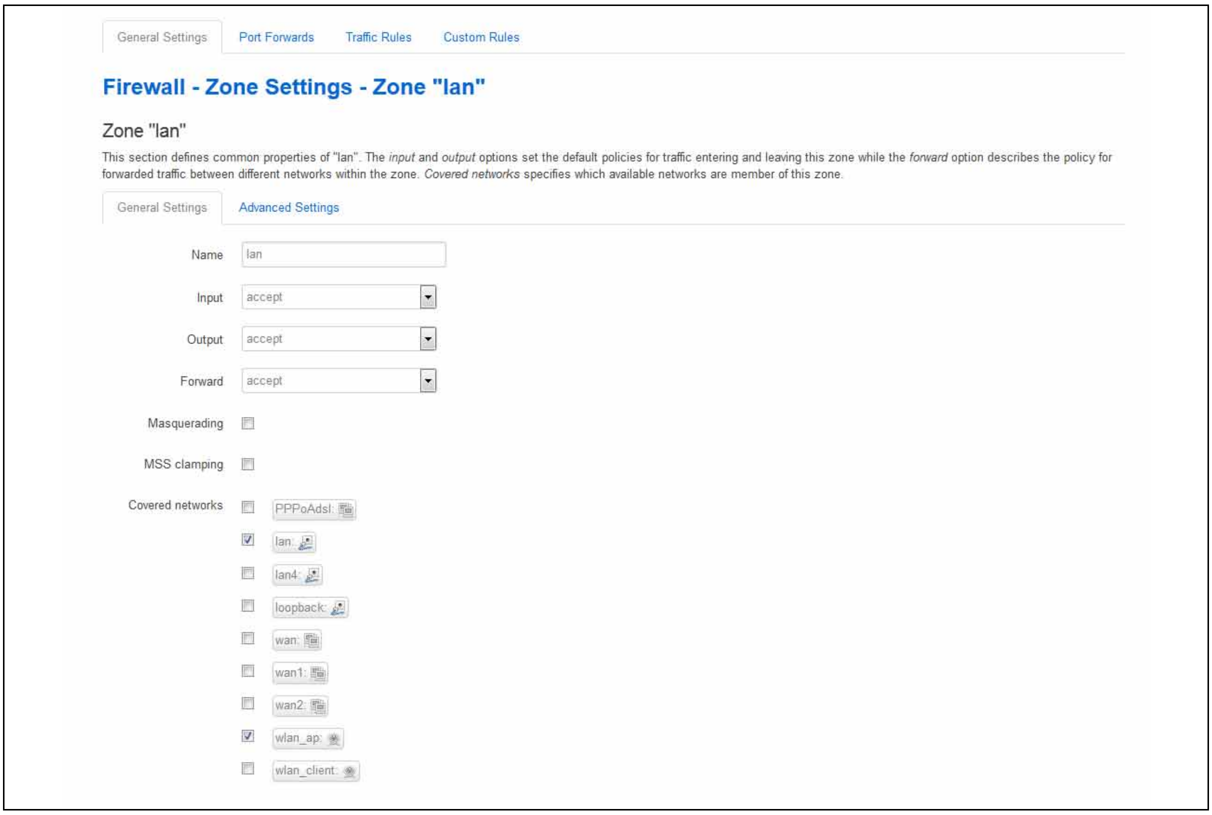


Figure 105: The firewall zone general settings

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: name
UCI: firewall.<zone label>.name
Opt: name</td><td>Sets the unique zone name. Maximum of 11 characters allowed. Note: the zone label is obtained by using the 'uci show firewall' command and is of the format '@zone[x]' where x is an integer starting at 0.</td></tr>
<tr><td>Web: Input
UCI: firewall.<zone label>.input
Opt: input</td><td>Default policy for incoming zone traffic. Incoming traffic is traffic entering the router through an interface selected in the 'Covered Networks' option for this zone.
<table>
<tr><td>Accept</td><td>Accepted packets pass through the firewall.</td></tr>
<tr><td>Reject</td><td>Rejected packets are blocked by the firewall and ICMP message is returned to the source host.</td></tr>
<tr><td>Drop</td><td>Dropped packets are blocked by the firewall.</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>Web: Output
UCI: firewall.<zone label>.output
Opt: output</td><td>Default policy for outgoing zone traffic. Outgoing traffic is traffic leaving the router through an interface selected in the 'Covered Networks' option for this zone.
<table>
<tr><td>Accept</td><td>Accepted packets pass through the firewall.</td></tr>
<tr><td>Reject</td><td>Rejected packets are blocked by the firewall and ICMP message is returned to the source host.</td></tr>
<tr><td>Drop</td><td>Dropped packets are blocked by the firewall.</td></tr>
</table></td></tr>
<tr><td>Web: Forward
UCI: firewall.<zone label>.forward
Opt: forward</td><td>Default policy for forwarded zone traffic. Forward rules for a zone describe what happens to traffic passing between different interfaces within that zone.
<table>
<tr><td>Accept</td><td>Accepted packets pass through the firewall.</td></tr>
<tr><td>Reject</td><td>Rejected packets are blocked by the firewall and ICMP message is returned to the source host.</td></tr>
<tr><td>Drop</td><td>Dropped packets are blocked by the firewall.</td></tr>
</table></td></tr>
<tr><td>Web: Masquerading
UCI: firewall.<zone label>.masq
Opt: masq</td><td>Specifies whether outgoing zone traffic should be masqueraded (NATTED). This is typically enabled on the wan zone.</td></tr>
<tr><td>Web: MSS Clamping
UCI: firewall.<zone label>.mtu_fix
Opt: mtu_fix</td><td>Enables MSS clamping for outgoing zone traffic. Subnets are allowed.</td></tr>
<tr><td>Web: Covered networks
UCI: firewall.<zone label>.network
Opt: network</td><td>Defines a list of interfaces attached to this zone, if omitted, the value of name is used by default.
Note: use the uci list syntax to edit this setting through UCI.</td></tr>
<tr><td>Web: Restrict to address family
UCI: firewall.<zone label>.family
Opt: family</td><td>Defines protocol family (ipv4, ipv6 or any) to generate iptables rules for.</td></tr>
</table>

Table 78: Information table for firewall zone settings

25.3.1.2 Firewall zone: advanced settings

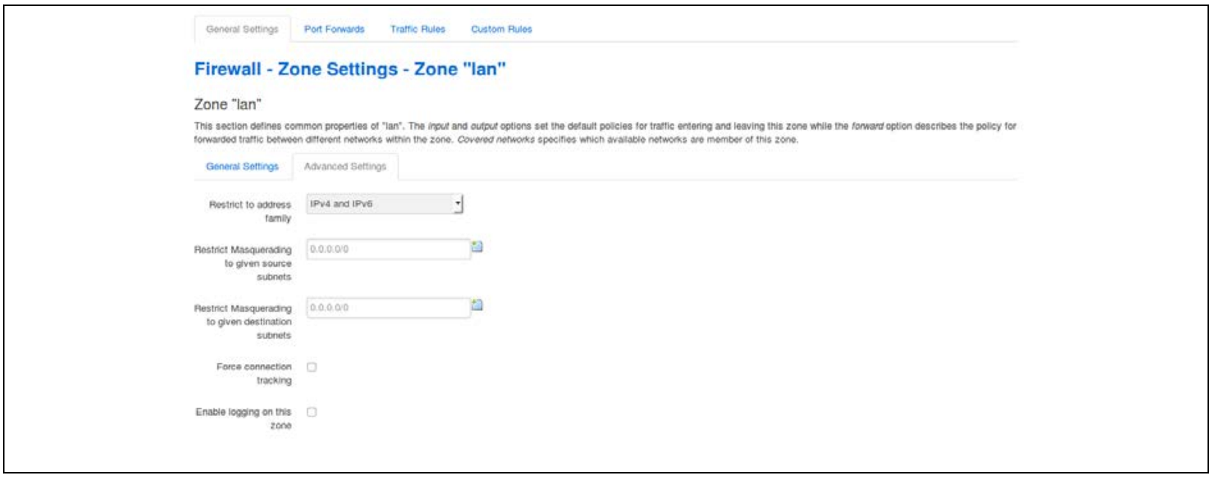


Figure 106: Firewall zone advanced settings

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Restrict Masquerading to given source subnets.
UCI: firewall.<zone label>.masq_src
Opt: masq_src</td><td>Limits masquerading to the given source subnets. Negation is possible by prefixing the subnet with ‘!’. Multiple subnets are allowed.</td></tr>
<tr><td>Web: Restrict Masquerading to given destination subnets.
UCI: firewall.<zone label>.masq_dest
Opt: masq_dest</td><td>Limits masquerading to the given destination subnets. Negation is possible by prefixing the subnet with ‘!’. Multiple subnets are allowed.</td></tr>
<tr><td>Web: Force connection tracking
UCI: firewall.<zone label>.conntrack
Opt: conntrack</td><td>Forces connection tracking for this zone.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>If masquerading is used. Otherwise, default is 0.</td></tr></table></td></tr>
<tr><td>Web: Enable logging on this zone
UCI: firewall.<zone label>.log
Opt: log</td><td>Creates log rules for rejected and dropped traffic in this zone.</td></tr>
<tr><td>Web: Limit log messages
UCI: firewall.<zone label>.log_limit
Opt: log_limit</td><td>Limits the amount of log messages per interval.</td></tr>
</table>

Table 79: Information table for zone settings

25.3.1.3 Inter-zone forwarding

This section controls the traffic flow between zones. Selecting a source or destination zone generates a Forwarding rule. Only one direction is covered by any forwarding rule. Hence for bidirectional traffic flow between two zones then two rules are required, with source and destination alternated.

Figure 107: The inter-zone forwarding section

Web Field / UCI / Package Option	Description
Web: Allow forward to destination zones UCI: firewall.<forwarding label>.dest Opt: dest	Allows forward to other zones. Enter the current zone as the source. Enabling this option puts two entries into the firewall file: destination and source.
UCI firewall.<forwarding label>.src Opt: src	
Web: Allow forward from source zones UCI: firewall.<forwarding label>.dest Opt: dest	Allows forward from other zones. Enter the current zone as the destination. Enabling this option puts two entries into the firewall file: destination and source.
UCI: firewall.<forwarding label>.src Opt: src	

Table 80: Information table for inter-zone forwarding settings

Note: the rules generated for forwarding traffic between zones relay connection tracking to be enabled on at least one of the source or destination zones. This can be enabled through the conntrack option or through masq.

25.3.2 Firewall port forwards

Port Forwards are also known as Redirects. This section creates the redirects using DNAT (Destination Network Address Translation) with Netfilter. The redirects are from the firewall zone labelled as wan to the firewall zone labelled as lan. These zones can refer to multiple external and internal interfaces as defined in the Firewall Zone settings.

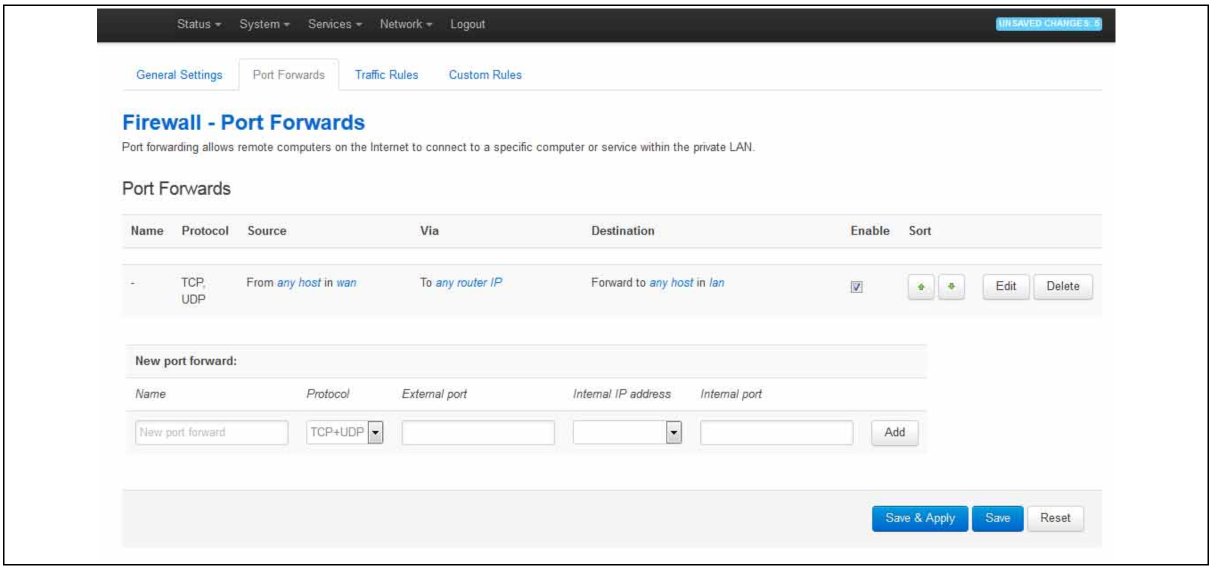


Figure 108: The firewall port forward page

Web Field / UCI / Package Option	**Description**
Web: name UCI: firewall.<redirect label>.name Opt: name	Sets the port forwarding name. For Web UI generated redirects the <redirect label> takes the form of @redirect[x], where x is an integer starting from 0.
Web: Protocol UCI: firewall.<redirect label>.proto Opt: proto	Defines layer 4 protocol to match incoming traffic. tcp+udp: Match either TCP or UDP packets. tcp: Match TCP packets only. udp: Match UDP packets only.
Web: Source UCI: firewall.<redirect label>.src Opt: src	Specifies the traffic source zone. It must refer to one of the defined zone names. When using the web interface, this is set to WAN initially. You can change this option through the web interface by editing the redirect after it is created.
Web: External port UCI: firewall.<redirect label>.src_dport Opt: src_dport	Specifies the incoming TCP/UDP port or port range to match. This is the incoming destination port specified by the external host. Port ranges specified as start:stop, for example, 2001:2020.
Web: Destination UCI: firewall.<redirect label>.dest Opt: dest	Specifies the traffic destination zone, must refer to one of the defined zone names. You can change this option through the web interface by editing the redirect after it is created.
Web: Internal IP address UCI: firewall.<redirect label>.dest_ip Opt: dest_ip	Specifies the internal (LAN) IP address for the traffic to be redirected to.
Web: Internal port UCI: firewall.<redirect label>.dest_port Opt: dest_port	Specifies the destination tcp/udp port for the redirect traffic.

Web: Enable UCI: firewall.<redirect label>.enabled Opt: enabled	Specifies if this redirect should be enabled or disabled. 0: Disabled. 1: Enabled.

Table 81: Information table for firewall port forward settings

The defined redirects can be sorted into a specific order to be applied. More specific rules should be placed first.

After the redirect is created and saved, to make changes, click Edit. This will provide further options to change the source/destination zones; specify source mac addresses and enable NAT loopback (reflection).

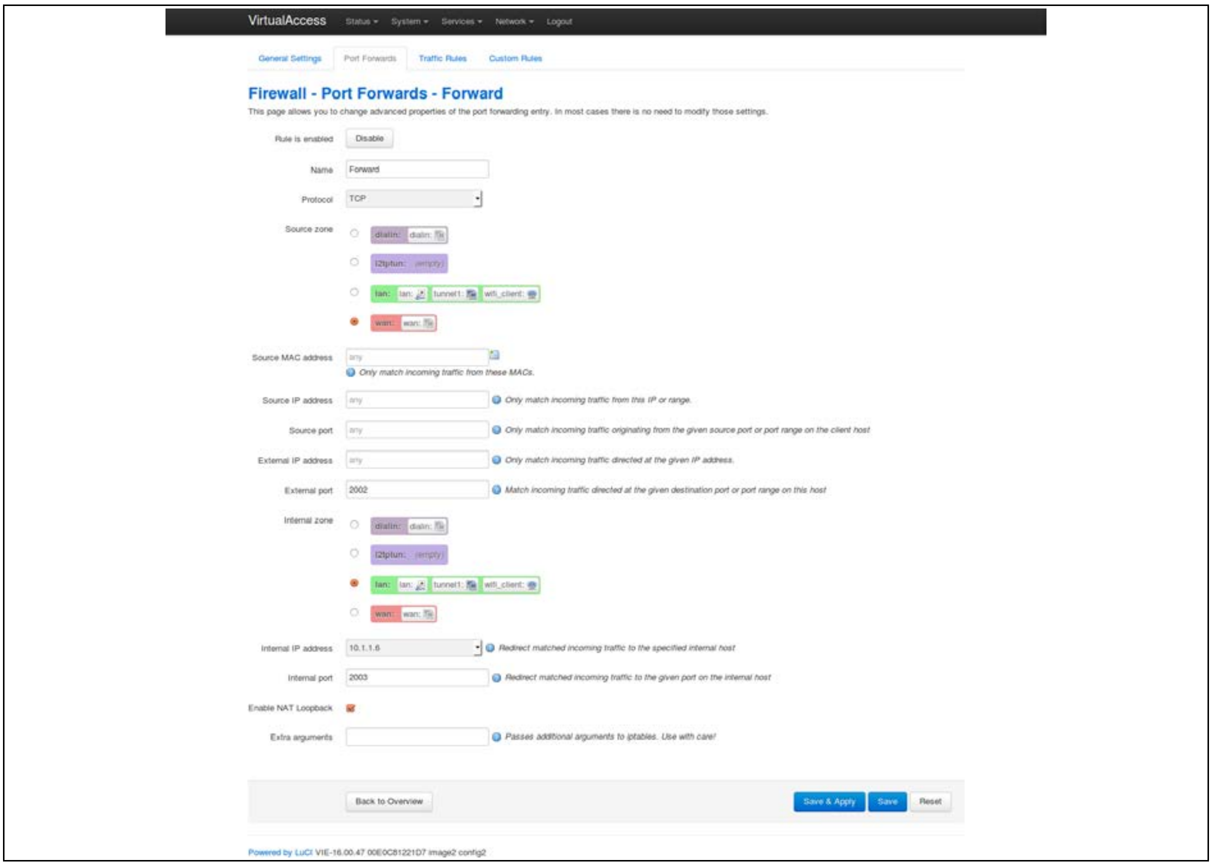


Figure 109: The firewall – port forwards – forward edits page

Web Field/UCI/Package Option	Description
Web: Enable NAT Loopback UCI: firewall.<redirect label>.reflection Opt: reflection	Enable or disable NAT reflection for this redirect. 0: reflection disabled 1: reflection enabled
Web: Extra arguments UCI: firewall.<redirect label>.extra Opt: extra	Passes extra arguments to IP tables. This is useful to specify additional match options, like -m policy --dir in for IPSec. The arguments are entered as text strings.

Table 82: Information table for port forward edits fields

25.3.3 Firewall traffic rules

Rules can be defined to allow or restrict access to specific ports, hosts or protocols.

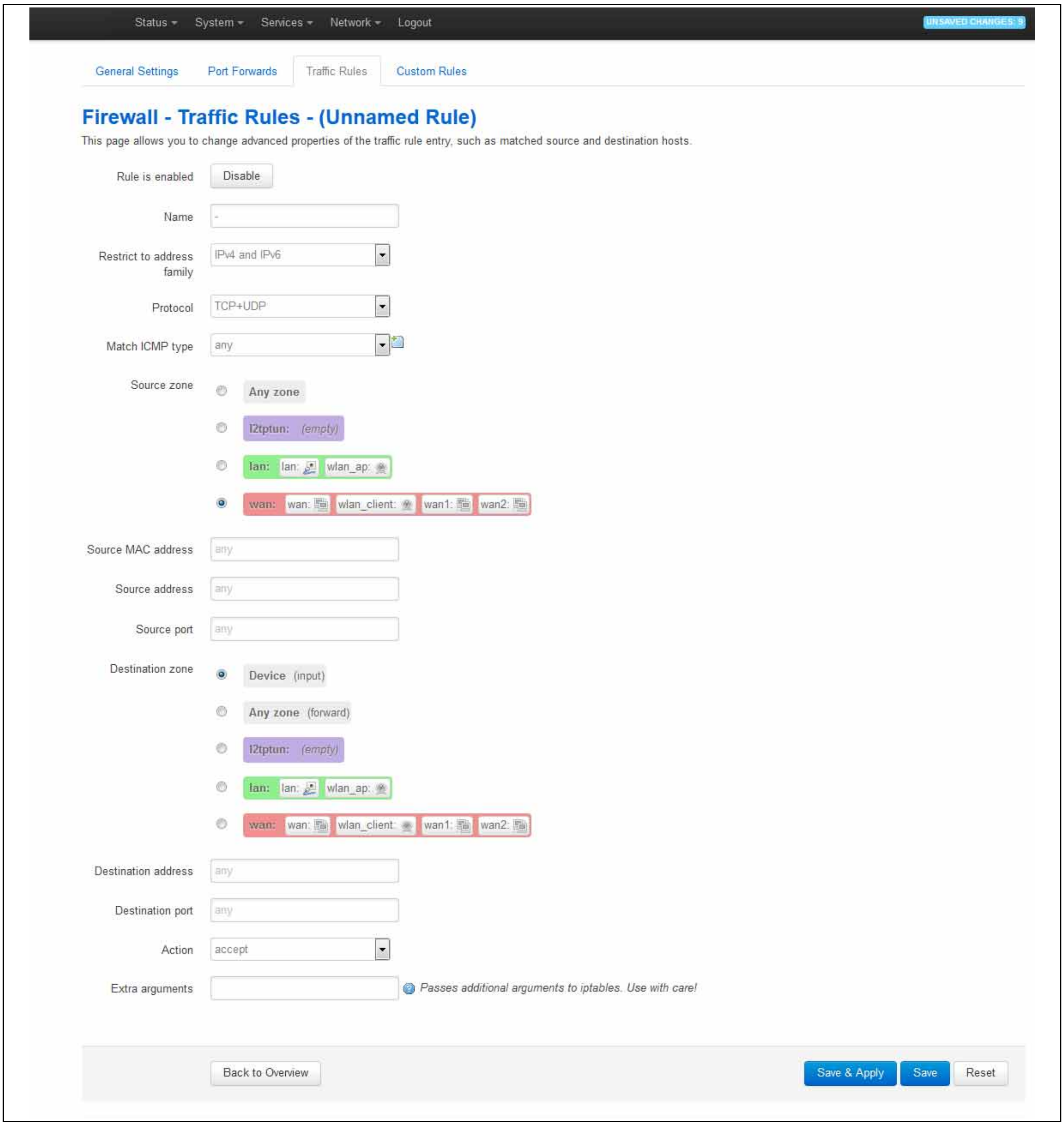


Figure 110: The firewall traffic rules page

<table>
<tr><th>Web Field / UCI / Package Option</th><th colspan="2">Description</th></tr>
<tr><td rowspan="3">Web: Rule is enabled
UCI: firewall.<rule label>.enabled
Opt: enabled</td><td colspan="2">Enables or disables traffic rule.</td></tr>
<tr><td>0</td><td>Rule is disabled.</td></tr>
<tr><td>1</td><td>Rule is enabled.</td></tr>
</table>

<table>
<tr><td>Web: Name
UCI: firewall.<rule label>.name
Opt: name</td><td>Select a descriptive name limited to less than 11 characters.</td></tr>
<tr><td>Web: Restrict to address family
UCI: firewall.<rule label>.family
Opt: family</td><td>Restrict to protocol family.<table><tr><td>IPv4 and IPv6</td><td>'any'. This applies the rule to both IPv4 and IPv6</td></tr><tr><td>IPv4 only</td><td>This applies the rule to IPv4 only</td></tr><tr><td>IPv6 only</td><td>This applies the rule to IPv6 only</td></tr></table></td></tr>
<tr><td>Web: Protocol
UCI: firewall.<rule label>.proto
Opt: proto</td><td>Matches incoming traffic using the given protocol.<table><tr><td>Any</td><td>Applies the rule to all protocols</td></tr><tr><td>TCP+UDP</td><td>Applies rule to TCP and UDP only</td></tr><tr><td>TCP</td><td>Applies rule to TCP only</td></tr><tr><td>UDP</td><td>Applies rule to UDP only</td></tr><tr><td>ICMP</td><td>Applies rule to ICMP only</td></tr><tr><td>custom</td><td>Specify protocol from /etc/protocols</td></tr></table></td></tr>
<tr><td>Web: Match ICMP type
UCI: firewall.<rule label>.icmp_type
Opt: icmp_type</td><td>Match specific icmp types.
This option is only valid when ICMP is selected as the protocol. ICMP types can be listed as either type names or type numbers.
Note: for a full list of valid ICMP type names, see the table below.</td></tr>
<tr><td>Web: Source zone
UCI: firewall.<rule label>.src
Opt: src</td><td>Specifies the traffic source zone, must refer to one of the defined zone names. For typical port forwards, this is usually WAN.</td></tr>
<tr><td>Web: Source MAC address
UCI: firewall.<rule label>.
Opt: src_mac</td><td>Matches incoming traffic from the specified mac address.</td></tr>
<tr><td>Web: Source address
UCI: firewall.<rule label>.src_ip
Opt: src_ip</td><td>Matches incoming traffic from the specified source IP address.</td></tr>
<tr><td>Web: Source port
UCI: firewall.<rule label>.src_port
Opt: src_port</td><td>Matches incoming traffic originating from the given source port or port range on the client host.</td></tr>
<tr><td>Web: Destination zone
UCI: firewall.<rule label>.dest
Opt: dest</td><td>Specifies the traffic destination zone. Must refer to one of the defined zone names.</td></tr>
<tr><td>Web: Destination address
UCI: firewall.<rule label>.dest_ip
Opt: dest_ip</td><td>For DNAT, redirects matched incoming traffic to the specified internal host.
For SNAT, matches traffic directed at the given address.</td></tr>
<tr><td>Web: Destination port
UCI: firewall.<rule label>.dest_port
Opt: dest_port</td><td>For DNAT, redirects matched incoming traffic to the given port on the internal host.
For SNAT, matches traffic directed at the given ports.</td></tr>
</table>

<table>
<tr><td>Web: Action
UCI: firewall.<rule label>.target
Opt: target</td><td>Action to take when rule is matched.<table><tr><td>drop</td><td></td></tr><tr><td>accept</td><td></td></tr><tr><td>reject</td><td></td></tr><tr><td>don't track</td><td></td></tr></table></td></tr>
<tr><td>Web: Extra arguments
UCI: firewall.<rule label>.extra
Opt: extra</td><td>Passes extra arguments to IP tables. This is useful to specify additional match options, like -m policy --dir in for IPSec.</td></tr>
<tr><td>Web: n/a
UCI: firewall.<rule label>.reflection
Opt: reflection</td><td>Disables NAT reflection for this redirect if set to 0. Applicable to DNAT targets.</td></tr>
<tr><td>Web: n/a
UCI: firewall.<rule label>.limit
Opt: limit</td><td>Sets maximum average matching rate; specified as a number, with an optional /second, /minute, /hour or /day suffix. Example 3/hour.</td></tr>
<tr><td>Web: n/a
UCI: firewall.<rule label>.limit_burst
Opt: limit_burst</td><td>Sets maximum initial number of packets to match. This number gets recharged by one every time the limit specified above is not reached, up to this number.</td></tr>
</table>

Table 83: Information table for firewall traffic rules

The following table shows Match ICMP type options.

ICMP Options	ICMP Options	ICMP Options	ICMP Options
address-mask-reply	host-redirect	pong	time-exceeded
address-mask-request	host-unknown	port-unreachable	timestamp-reply
any	host-unreachable	precedence-cutoff	timestamp-request
communication-prohibited	ip-header-bad	protocol-unreachable	TOS-host-redirect
destination-unreachable	network-prohibited	redirect	TOS-host-unreachable
echo-reply	network-redirect	required-option-missing	TOS-network-redirect
echo-request	network-unknown	router-advertisement	TOS-network-unreachable
fragmentation-needed	network-unreachable	router-solicitation	ttl-exceeded
host-precedence-violation	parameter-problem	source-quench	ttl-zero-during-reassembly
host-prohibited	ping	source-route-failed	ttl-zero-during-transit

Table 84: Information table for match ICMP type drop-down menu

25.3.4 Custom rules

Iptables rules can be defined here. Custom rules are applied after all other rules are applied. Consult official iptables documentation for exact syntax and details.

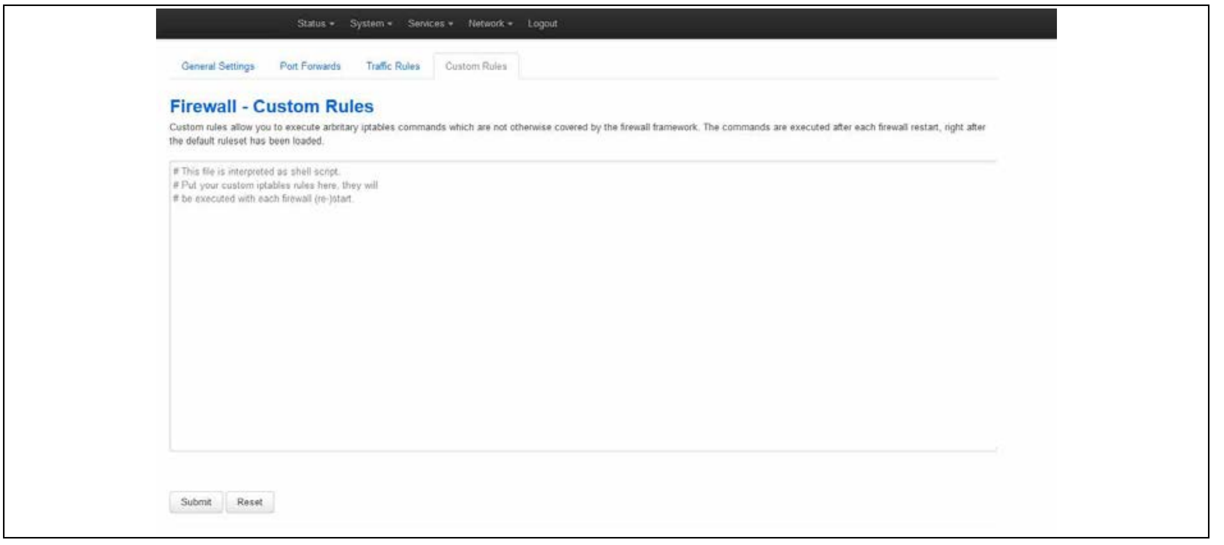


Figure 111: The custom rules page

Command	Description
src	Specifies the traffic source zone, must refer to one of the defined zone names.
src_ip	Match incoming traffic from the specified source IP address.
src_mac	Match incoming traffic from the specified mac address.
src_port	Match incoming traffic originating from the given source port or port range on the client host if tcp or udp is specified as protocol.
proto	Match incoming traffic using the given protocol. Can be one of tcp, udp, tcpudp, udplite, icmp, esp, ah, sctp, or all or it can be a numeric value, representing one of these protocols or a different one. A protocol name from /etc/protocols is also allowed. The number 0 is equivalent to all.
Dest	Specifies the traffic destination zone, must refer to one of the defined zone names. If specified, the rule applies to forwarded traffic else it is treated as input rule.
dest_ip	Match incoming traffic directed to the specified destination IP address.
dest_port	Match incoming traffic directed at the given destination port or port range on this host if tcp or udp is specified as protocol.
target	Firewall action (ACCEPT, REJECT, DROP) for matched traffic.
family	Protocol family (ipv4, ipv6 or any) to generate iptables rules for.
limit	Maximum average matching rate; specified as a number, with an optional /second, /minute, /hour or /day suffix. Example3/hour.
limit_burst	Maximum initial number of packets to match; this number gets recharged by one every time the limit specified above is not reached, up to this number.
extra	Extra arguments to pass to iptables, this is mainly useful to specify additional match options, like -m policy --dir in for IPSec.

Table 85: Information table for custom rules commands

25.4 Configuring firewall using UCI

25.4.1 Firewall general settings

To set general (default) settings, enter:

```
uci add firewall defaults
uci set firewall.@defaults[0].syn_flood=1
uci set firewall.@defaults[0].drop_invalid=1
uci set firewall.@defaults[0].input=ACCEPT
uci set firewall.@defaults[0].output=ACCEPT
uci set firewall.@defaults[0].forward=ACCEPT
```

Note: this command is only required if there is no defaults section.

25.4.2 Firewall zone settings

To set up a firewall zone, enter:

```
uci add firewall zone
uci set firewall.@zone[1].name=lan
uci set firewall.@zone[1].input=ACCEPT
uci set firewall.@zone[1].output=ACCEPT
uci set firewall.@zone[1].forward=ACCEPT
uci set firewall.@zone[1].network=lan1 wifi_client
uci set firewall.@zone[1].family=any
uci set firewall.@zone[1].masq_src=10.0.0.0/24
uci set firewall.@zone[1].masq_dest=20.0.0.0/24
uci set firewall.@zone[1].conntrack=1
uci set firewall.@zone[1].masq=1
uci set firewall.@zone[1].mtu_fix=1
uci set firewall.@zone[1].log=1
uci set firewall.@zone[1].log_limit=5
```

25.4.3 Inter-zone forwarding

To enable forwarding of traffic from WAN to LAN, enter:

```
uci add firewall forwarding
uci set firewall.@forwarding[1].dest=wan
uci set firewall.@forwarding[1].src=lan
```

25.4.4 Firewall port forwards

To set port forwarding rules, enter:

```
uci add firewall redirect
```

```
uci set firewall.@redirect[1].name=Forward
uci set firewall.@redirect[1].proto=tcp
uci set firewall.@redirect[1].src=wan   #  <- zone names
uci set firewall.@redirect[1].dest=lan   # <- zone names
uci set firewall.@redirect[1].src_dport=2001
uci set firewall.@redirect[1].dest_ip=192.168.0.100
uci set firewall.@redirect[1].dest_port=2005
uci set firewall.@redirect[1].enabled=1
```

25.4.5 Firewall traffic rules

To set traffic rules, enter:

```
uci add firewall rule
uci set firewall.@rule[1].enabled=1
uci set firewall.@rule[1].name=Allow_ICMP
uci set firewall.@rule[1].family=any
uci set firewall.@rule[1].proto=ICMP
uci set firewall.@rule[1].icmp_type=any
uci set firewall.@rule[1].src=wan
uci set firewall.@rule[1].src_mac=ff:ff:ff:ff:ff:ff
uci set firewall.@rule[1].src_port=
uci set firewall.@rule[1].dest=lan
uci set firewall.@rule[1].dest_port=
uci set firewall.@rule[1].dest_ip=192.168.100.1
uci set firewall.@rule[1].target=ACCEPT
uci set firewall.@rule[1].extra=
uci set firewall.@rule[1].src_ip=8.8.8.8
uci set firewall.@rule[1].src_dip=9.9.9.9
uci set firewall.@rule[1].src_dport=68
uci set firewall.@rule[1].reflection=1
uci set firewall.@rule[1].limit=3/second
uci set firewall.@rule[1].limit_burst=30
```

25.5 Custom firewall scripts: includes

It is possible to include custom firewall scripts by specifying one or more include sections in the firewall configuration.

There is only one possible parameter for includes:

Parameter	Description
path	Specifies a shell script to execute on boot or firewall restarts.

Custom scripts are executed as shell scripts and are expected to contain iptables commands.

25.6 IPv6 notes

As described above, the option family is used for distinguishing between IPv4, IPv6 and both protocols. However, the family is inferred automatically if a specific IP address family is used. For example; if IPv6 addresses are used then the rule is automatically treated as IPv6 only rule.

```
config rule
        option src wan
        option src_ip fdca:f00:ba3::/64
        option target ACCEPT
```

Similarly, the following rule is automatically treated as IPv4 only.

```
config rule
        option src wan
        option dest_ip 88.77.66.55
        option target REJECT
```

Rules without IP addresses are automatically added to iptables and ip6tables, unless overridden by the family option. Redirect rules (port forwards) are always IPv4 since there is no IPv6 DNAT support at present.

25.7 Implications of DROP vs. REJECT

The decision whether to drop or to reject traffic should be done on a case-by-case basis. Many people see dropping traffic as a security advantage over rejecting it because it exposes less information to a hypothetical attacker. While dropping slightly increases security, it can also complicate the debugging of network issues or cause unwanted side-effects on client programs.

If traffic is rejected, the router will respond with an icmp error message ("destination port unreachable") causing the connection attempt to fail immediately. This also means that for each connection attempt a certain amount of response traffic is generated. This can actually harm if the firewall is attacked with many simultaneous connection attempts, the resulting backfire of icmp responses can clog up all available upload and make the connection unusable (DoS).

When connection attempts are dropped the client is not aware of the blocking and will continue to re-transmit its packets until the connection eventually times out. Depending on the way the client software is implemented, this could result in frozen or hanging programs that need to wait until a timeout occurs before they're able to continue.

DROP

- less information is exposed
- less attack surface
- client software may not cope well with it (hangs until connection times out)
- may complicate network debugging (where was traffic dropped and why)

REJECT

- may expose information (like the IP at which traffic was actually blocked)
- client software can recover faster from rejected connection attempts
- network debugging easier (routing and firewall issues clearly distinguishable)

25.8 Connection tracking

By default, the firewall will disable connection tracking for a zone if no masquerading is enabled. This is achieved by generating NOTRACK firewall rules matching all traffic passing via interfaces referenced by the firewall zone. The purpose of NOTRACK is to speed up routing and save memory by circumventing resource intensive connection tracking in cases where it is not needed. You can check if connection tracking is disabled by issuing iptables -t raw -S, it will list all rules, check for NOTRACK target.

NOTRACK will render certain iptables extensions unusable, for example the MASQUERADE target or the state match will not work.

If connection tracking is required, for example by custom rules in /etc/firewall.user, the conntrack option must be enabled in the corresponding zone to disable NOTRACK. It should appear as option 'conntrack' '1' in the right zone in /etc/config/firewall.

25.9 Firewall examples

25.9.1 Opening ports

The default configuration accepts all LAN traffic, but blocks all incoming WAN traffic on ports not currently used for connections or NAT. To open a port for a service, add a rule section:

```
config rule
        option src              wan
        option dest_port        22
        option target           ACCEPT
        option proto            tcp
```

This example enables machines on the internet to use SSH to access your router.

25.9.2 Forwarding ports (destination NAT/DNAT)

This example forwards http, but not HTTPS, traffic to the web server running on 192.168.1.10:

```
config redirect
        option src       wan
        option src_dport 80
        option proto     tcp
        option dest_ip   192.168.1.10
```

The next example forwards one arbitrary port that you define to a box running SSH behind the firewall in a more secure manner because it is not using default port 22.

```
config 'redirect'
        option 'name' 'ssh'
        option 'src' 'wan'
        option 'proto' 'tcpudp'
        option 'src_dport' '5555'
        option 'dest_ip' '192.168.1.100'
        option 'dest_port' '22'
        option 'target' 'DNAT'
        option 'dest' 'lan'
```

25.9.3 Source NAT (SNAT)

Source NAT changes an outgoing packet destined for the system so that is looks as though the system is the source of the packet.

Define source NAT for UDP and TCP traffic directed to port 123 originating from the host with the IP address 10.55.34.85. The source address is rewritten to 63.240.161.99.

```
config redirect
        option src              lan
        option dest             wan
        option src_ip           10.55.34.85
        option src_dip          63.240.161.99
        option dest_port        123
        option target           SNAT
```

When used alone, Source NAT is used to restrict a computer's access to the internet, but allows it to access a few services by manually forwarding what appear to be a few local services; for example, NTP to the Internet. While DNAT hides the local network from the Internet, SNAT hides the Internet from the local network.

Source NAT and destination NAT are combined and used dynamically in IP masquerading to make computers with private (192.168.x.x, etc.) IP addresses appear on the internet with the system's public WAN IP address.

25.9.4 True destination port forwarding

This usage is similar to SNAT, but as the destination IP address is not changed, machines on the destination network need to be aware that they'll receive and answer requests from a public IP address that is not necessarily theirs. Port forwarding in this fashion is typically used for load balancing.

```
config redirect
        option src              wan
        option src_dport        80
        option dest             lan
        option dest_port        80
        option proto            tcp
```

25.9.5 Block access to a specific host

The following rule blocks all connection attempts to the specified host address.

```
config rule
        option src              lan
        option dest             wan
        option dest_ip          123.45.67.89
        option target           REJECT
```

25.9.6 Block access to the internet using MAC

The following rule blocks all connection attempts from the client to the internet.

```
config rule
        option src              lan
        option dest             wan
        option src_mac          00:00:00:00:00:00
        option target           REJECT
```

25.9.7 Block access to the internet for specific IP on certain times

The following rule blocks all connection attempts to the internet from 192.168.1.27 on weekdays between 21:00pm and 09:00am.

```
config rule
        option src              lan
        option dest             wan
        option src_ip           192.168.1.27
        option extra            '-m time --weekdays Mon,Tue,Wed,Thu,Fri --
timestart 21:00 --timestop 09:00'
        option target           REJECT
```

25.9.8 Restricted forwarding rule

The example below creates a forward rule rejecting traffic from LAN to WAN on the ports 1000-1100.

```
config rule
        option src              lan
        option dest             wan
        option dest_port        1000-1100
        option proto            tcpudp
        option target           REJECT
```

25.9.9 Transparent proxy rule (same host)

The rule below redirects all outgoing HTTP traffic from LAN through a proxy server listening at port 3128 on the router itself.

```
config redirect
      option src              lan
      option proto            tcp
      option src_dport        80
      option dest_port        3128
```

25.9.10 Transparent proxy rule (external)

The following rule redirects all outgoing HTTP traffic from LAN through an external proxy at 192.168.1.100 listening on port 3128. It assumes the router LAN address to be 192.168.1.1 - this is needed to masquerade redirected traffic towards the proxy.

```
config redirect
        option src              lan
        option proto            tcp
        option src_ip           !192.168.1.100
        option src_dport        80
        option dest_ip          192.168.1.100
        option dest_port        3128
        option target           DNAT

config redirect
        option dest             lan
        option proto            tcp
        option src_dip          192.168.1.1
        option dest_ip          192.168.1.100
        option dest_port        3128
        option target           SNAT
```

25.9.11 Simple DMZ rule

The following rule redirects all WAN ports for all protocols to the internal host 192.168.1.2.

```
config redirect
      option src              wan
      option proto            all
      option dest_ip          192.168.1.2
```

25.9.12 IPSec passthrough

This example enables proper forwarding of IPSec traffic through the WAN.

```
# AH protocol
config rule
        option src              wan
        option dest             lan
        option proto            ah
        option target           ACCEPT

# ESP protocol
config rule
        option src              wan
        option dest             lan
        option proto            esp
        option target           ACCEPT
```

For some configurations you also have to open port 500/UDP.

```
# ISAKMP protocol
config rule
        option src              wan
        option dest             lan
        option proto            udp
        option src_port         500
        option dest_port        500
        option target           ACCEPT
```

25.9.13 Manual iptables rules

You can specify traditional iptables rules, in the standard iptables unix command form, in an external file and included in the firewall config file. It is possible to use this process to include multiple files.

```
config include
       option path /etc/firewall.user

config include
       option path /etc/firewall.vpn
```

The syntax for the includes is Linux standard and therefore different from UCIs.

25.9.14 Firewall management

After a configuration change, to rebuild firewall rules, enter:

```
root@VA_router:/# /etc/init.d/firewall restart
```

Executing the following command will flush all rules and set the policies to ACCEPT on all standard chains:

```
root@VA_router:/# /etc/init.d/firewall stop
```

To manually start the firewall, enter:

```
root@VA_router:/# /etc/init.d/firewall start
```

To permanently disable the firewall, enter:

```
root@VA_router:/# /etc/init.d/firewall disable
```

Note: disable does not flush the rules, so you might be required to issue a stop before.

To enable the firewall again, enter:

```
root@VA_router:/# /etc/init.d/firewall enable
```

25.9.15 Debug generated rule set

It is possible to observe the iptables commands generated by the firewall programme. This is useful to track down iptables errors during firewall restarts or to verify the outcome of certain UCI rules.

To see the rules as they are executed, run the fw command with the FW_TRACE environment variable set to **1** (one):

```
root@VA_router:/# FW_TRACE=1 fw reload
```

To direct the output to a file for later inspection, enter:

```
root@VA_router:/# FW_TRACE=1 fw reload 2>/tmp/iptables.lo
```

26 Configuring SNMP

SNMP (Simple Network Management Protocol) is an internet-standard protocol for managing devices on IP networks.

26.1 Configuration package used

Package	Sections				
snmpd	access agent com2sec constant	exec group heartbeat informreceiver	inventory inventory_iftable monitor_disk monitor_ioerror	monitor_load monitor_memory monitor_process pass	system trapreceiver usm_user view

The SNMP application has several configuration sections:

System and Agent	Configures the SNMP agent.
Com2Sec	Maps SNMP community names into an arbitrary security name.
Group	Assigns community names and SNMP protocols to groups.
View and Access	Creates views and sub views of the whole available SNMP tree and grants specific access to those views on a group by group basis.
Trap receiver	Address of a notification receiver that should be sent SNMPv1 TRAPs and SNMPv2c TRAP2s.
Inform receiver	Address of a notification receiver that should be sent SNMPv2 INFORM notifications respectively

26.2 Configuring SMNP using the web interface

In the top menu, select **Services -> SNMP**. The SNMP Service page appears.

26.2.1 System and agent settings

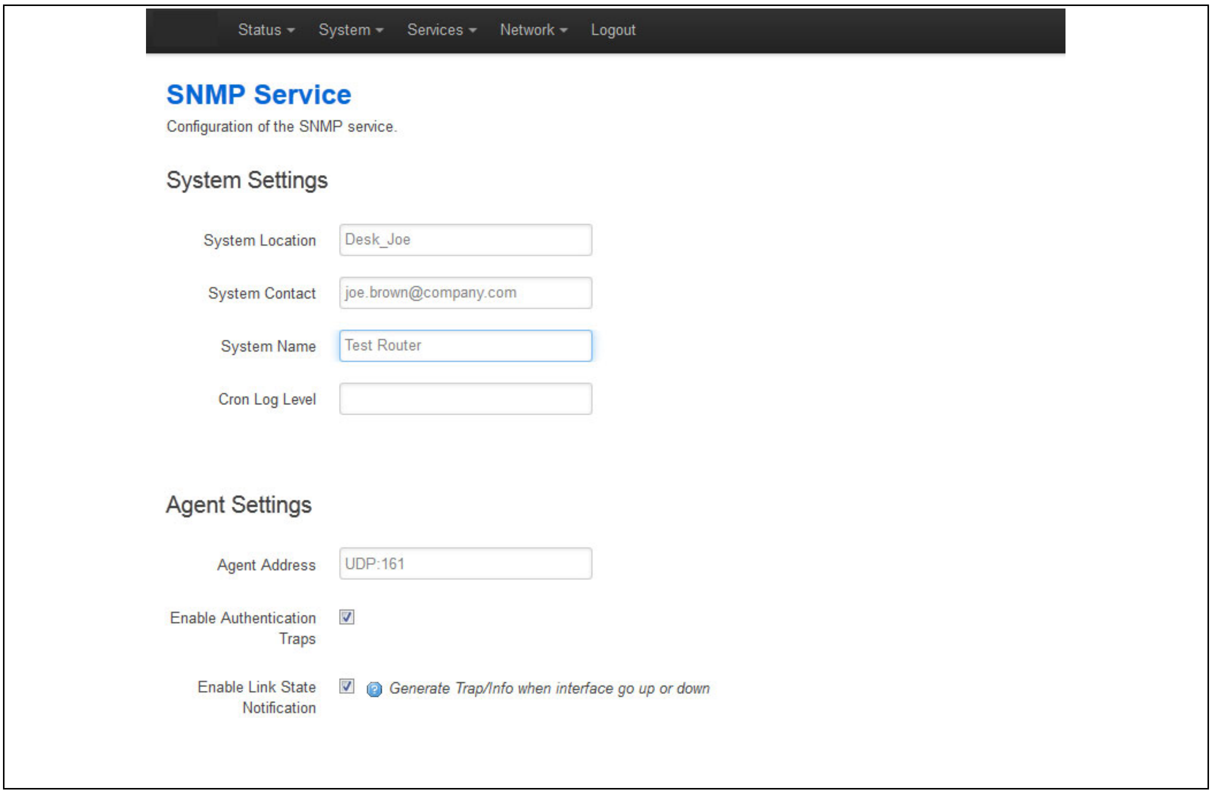


Figure 112: The SNMP service page

Web Field / UCI / Package Option	Description
System settings	
Web: System Location UCI: snmpd.system[0].sysLocation Opt: sysLocation	Sets the system location, system contact or system name for the agent. This information is reported in the 'system' group in the mibII tree.
Web: System Contact UCI: snmpd.system[0].sysContact Opt: sysContact	
Web: System Name UCI: snmpd.system[0].sysName Opt: sysName	
Agent Settings	
Web: Agent Address UCI: snmpd.agent[0].agentaddress Opt: agentaddress	Specifies the address(es) and port(s) on which the agent should listen. [(udp|tcp):]port[@address][,...]
Web: Enable Authentication Traps UCI: snmpd.agent[0].authtrapenabled Opt: authtrapenabled	Enables or disables SNMP authentication trap. 0 — Disabled. 1 — Enabled. **Note**: this is the SNMP poll authentication trap to be set when there is a community mismatch.

Web: Enable Link State Notification UCI: snmpd.agent[0].link_updown_notify Opt: link_updown_notify	Generates trap/info when interface goes up or down. When enabled, the router sends a trap notification link up or down. 0 — Disabled. 1 — Enabled.

Table 86: Information table for system and agent settings

26.2.2 Com2Sec settings

To access Com2Sec settings, scroll down the SNMP Services page.

Use the COM2Sec section to map SNMP community names into an arbitrary security name. Map community names into security names based on the community name and the source subnet. Use the first source/community combination that matches the incoming packet.

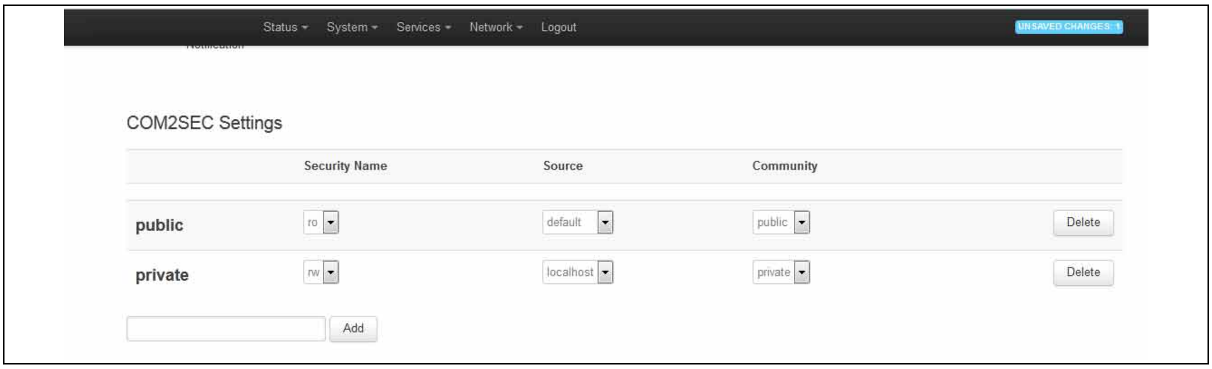


Figure 113: The COM2Sec settings

Web Field/UCI/Package Option	Description
Web: Security Name UCI: snmpd.com2sec[x].secname Opt: secname	Specifies an arbitrary security name for the user.
Web: Source UCI: snmpd.com2sec[x].source Opt: source	A hostname, localhost or a subnet specified as a.b.c.d/mask or a.b.c.d/bits or 'default' for no restrictions.
Web: Community UCI: snmpd.com2sec[x].community Opt: community	Specifies the community string being presented in the request.

Table 87: Information table for Com2Sec settings

26.2.3 Group settings

Group settings assign community names and SNMP protocols to groups.

Figure 114: The group settings section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Group
UCI: snmpd.group[x].group
Opt: group</td><td>Specifies an arbitrary group name.</td></tr>
<tr><td>Web: Version
UCI: snmpd.group[x].version
Opt: version</td><td>Specifies the SNMP version number being used in the request: v1, v2c and usm are supported.
<table>
<tr><td>v1</td><td>SNMP v1</td></tr>
<tr><td>v2v</td><td>SNMP v2</td></tr>
<tr><td>usm</td><td>SNMP v3</td></tr>
<tr><td>any</td><td>Any SNMP version</td></tr>
</table></td></tr>
<tr><td>Web: Security Name
UCI: snmpd.group[x].secname
Opt: secname</td><td>An already defined security name that is being included in this group.</td></tr>
</table>

Table 88: Information table for group settings

26.2.4 View settings

View settings define a named "view", which is a subset of the overall OID tree. This is most commonly a single subtree, but several view directives can be given with the same view name, to build up a more complex collection of OIDs.

Figure 115: The view settings section

Web Field / UCI / Package Option	Description
Web: Name UCI: snmpd.view[x].viewname Opt: viewname	Specifies an arbitrary view name. Typically it describes what the view shows.

<table>
<tr><td>Web: Type
UCI: snmpd.view[x].type
Opt: type</td><td>Specifies whether the view lists oids that are included in the view or lists oids to be excluded from the view (in which case all other oids are visible apart from those ones listed).<table><tr><td>included</td><td></td></tr><tr><td>excluded</td><td></td></tr></table></td></tr>
<tr><td>Web: OID
UCI: snmpd.view[x].oid
Opt: oid</td><td>OID to be included in or excluded from the view. Only numerical representation is supported.
Example<table><tr><td>1</td><td>Everything</td></tr><tr><td>1.3.6.1.2.1.2</td><td>Interfaces table</td></tr></table></td></tr>
</table>

Table 89: Information table for view settings

26.2.5 Access settings

Access settings map from a group of users/communities, in a specific context and with a particular SNMP version and minimum security level, to one of three views, depending on the request being processed.

Figure 116: The access settings section

<table>
<tr><th>Web Field / UCI / Package Option</th><th>Description</th></tr>
<tr><td>Web: Group
UCI: snmpd.access[x].group
Opt: group</td><td>Specifies the group to which access is being granted.</td></tr>
<tr><td>Web: Context
UCI: snmpd.access[x].context
Opt: context</td><td>SNMPv3 request context is matched against the value according to the prefix below. For SNMP v1 and SNMP v2c, the context must be none.<table><tr><td>none</td><td></td></tr><tr><td>all</td><td></td></tr></table></td></tr>
<tr><td>Web: Version
UCI: snmpd.access[x].version
Opt: version</td><td>Specifies the SNMP version number being used in the request: any, v1, v2c and usm are supported.<table><tr><td>v1</td><td>SNMP v1</td></tr><tr><td>v2v</td><td>SNMP v2</td></tr><tr><td>usm</td><td>SNMP v3</td></tr><tr><td>any</td><td>Any SNMP version</td></tr></table></td></tr>
</table>

Web: Level UCI: snmpd.access[x].level Opt: level	Specifies the security level. For SNMP v1 and SNMP v2c level must be noauth. noauth auth priv
Web: Prefix UCI: snmpd.access[x].prefix Opt: prefix	Prefix specifies how context (above) should be matched against the context of the incoming pdu. exact any all
Web: Read UCI: snmpd.access[x].read Opt: read	Specifies the view to be used for read access.
Web: Write UCI: snmpd.access[x].write Opt: write	Specifies the view to be used for write access.
Web: Notify UCI: snmpd.access[x].notify Opt: notify	Specifies the view to be used for notify access.

Table 90: Information table for access settings

26.2.6 Trap receiver

Trap receiver settings define a notification receiver that should be sent SNMPv1 TRAPs and SNMPv2c TRAP2.

Figure 117: The trap receiver settings page

Web Field / UCI / Package Option	Description
Web: Host UCI: snmpd.trapreceiver[x].host Opt: host	Host address. Can be either an IP address or a FQDN.
Web: Port UCI: snmpd.trapreceiver[x].port Opt: port	UDP port to be used for sending traps. Range 162
Web: Version UCI: snmpd.trapreceiver[x].version Opt: version	SNMP version. v1 V2

Web: Community UCI: snmpd.trapreceiver[x].community Opt: community	Community to use in trap messages for this host.

Table 91: Information table for trap receiver settings

26.2.7 Inform receiver

Inform receiver settings define a notification receiver that should be sent SNMPv2c INFORM notifications.

Figure 118: The inform receiver settings page

Web Field/UCI/Package Option	Description
Web: Host UCI: snmpd.informreceiver[x].host Opt: host	Host address. Can be either an IP address or a FQDN.
Web: Port UCI: snmpd.informreceiver[x].port Opt: port	UDP port to be used for sending traps. Range 162
Web: Community UCI: snmpd.informreceiver[x].community Opt: community	Community to use in inform messages for this host.

Table 92: Information table for trap receiver settings

26.3 Configuring SNMP using command line

The configuration files are stored at **/etc/config/snmpd**

26.3.1 System settings using UCI

```
root@VA_router:~# uci show snmpd
snmpd.system=system
snmpd.system.sysLocation=Office 123
snmpd.system.sysContact=Mr White
snmpd.system.sysName=Backup Access 4
```

```
snmpd.agent=agent
snmpd.agent.agentaddress=UDP:161
snmpd.agent.authtrapenabled=yes
snmpd.agent.link_updown_notify=yes
```

26.3.2 System settings using package options

```
root@VA_router:~# uci export snmpd
package snmpd
config 'system'
      option sysLocation 'Office 123'
      option sysContact 'Mr White'
      option sysName 'Backup Access 4'

config 'agent'
      option agentaddress 'UDP:161'
      option authtrapenabled '1'
       option link_updown_notify '1'
```

Another sample agent configuration shown below causes the agent to listen on UDP port 161, TCP port 161 and UDP port 9161 on only the interface associated with the localhost address.

```
config 'agent'
      option agentaddress 'UDP:161,tcp:161,9161@localhost'
```

26.3.3 com2sec settings

The following sample specifies that a request from any source using “public” as the community string will be dealt with using the security name “ro”. However, any request from the localhost itself using “private” as the community string will be dealt with using the security name “rw”.

Note: the security names of “ro” and “rw” here are simply names – the fact of a security name having read only or read-write permissions is handled in the access section and dealt with at a group granularity.

26.3.3.1 Com2sec using UCI

```
snmpd.c2s_1=com2sec
snmpd.c2s_1.source=default
snmpd.c2s_1.community=public
snmpd.c2s_1.secname=rw
snmpd.c2s_2=com2sec
snmpd.c2s_2.source=localhost
snmpd.c2s_2.community=private
snmpd.c2s_2.secname=ro
```

26.3.3.2 Com2sec using package options

```
config 'com2sec' 'public'
      option secname 'ro'
      option source 'default'
      option community 'public'

config 'com2sec' 'private'
      option secname 'rw'
      option source 'localhost'
      option community 'private'
```

26.3.4 Group settings

The following example specifies that a request from the security name “ro” using snmp v1, v2c or USM (User Based Security Model for SNM P v3) are all mapped to the “public” group. Similarly, requests from the security name “rw” in all protocols are mapped to the “private” group.

26.3.4.1 Group settings using UCI

```
snmpd.grp_1_v1=group
snmpd.grp_1_v1.version=v1
snmpd.grp_1_v1.group=public
snmpd.grp_1_v1.secname=ro
snmpd.grp_1_v2c=group
snmpd.grp_1_v2c.version=v2c
snmpd.grp_1_v2c.group=public
snmpd.grp_1_v2c.secname=ro
snmpd.grp_1_usm=group
```

```
snmpd.grp_1_usm.version=usm
snmpd.grp_1_usm.group=public
snmpd.grp_1_usm.secname=ro
snmpd.grp_1_access=access
snmpd.grp_1_access.context=none
snmpd.grp_1_access.version=any
snmpd.grp_1_access.level=noauth
snmpd.grp_1_access.prefix=exact
snmpd.grp_1_access.read=all
snmpd.grp_1_access.write=none
snmpd.grp_1_access.notify=none
snmpd.grp_1_access.group=public
snmpd.grp_2_v1=group
snmpd.grp_2_v1.version=v1
snmpd.grp_2_v1.group=public
snmpd.grp_2_v1.secname=ro
snmpd.grp_2_v2c=group
snmpd.grp_2_v2c.version=v2c
snmpd.grp_2_v2c.group=public
snmpd.grp_2_v2c.secname=ro
snmpd.grp_2_usm=group
snmpd.grp_2_usm.version=usm
snmpd.grp_2_usm.group=public
snmpd.grp_2_usm.secname=ro
snmpd.grp_2_access=access
snmpd.grp_2_access.context=none
snmpd.grp_2_access.version=any
snmpd.grp_2_access.level=noauth
snmpd.grp_2_access.prefix=exact
snmpd.grp_2_access.read=all
snmpd.grp_2_access.write=all
snmpd.grp_2_access.notify=all
snmpd.grp_2_access.group=public
```

26.3.4.2 Group settings using package options

```
config 'group' 'public_v1'
      option group 'public'
```

```
      option version 'v1'
      option secname 'ro'

config 'group' 'public_v2c'
      option group 'public'
      option version 'v2c'
      option secname 'ro'

config 'group' 'public_usm'
      option group 'public'
      option version 'usm'
      option secname 'ro'

config 'group' 'private_v1'
      option group 'private'
      option version 'v1'
      option secname 'rw'
config 'group' 'private_v2c'
      option group 'private'

      option version 'v2c'
      option secname 'rw'

config 'group' 'private_usm'
      option group 'private'
      option version 'usm'
      option secname 'rw'
```

26.3.5 View settings

The following example defines two views, one for the entire system and another for only mib2.

26.3.5.1 View settings using UCI

```
snmpd.all=view
snmpd.all.viewname=all
```

```
snmpd.all.oid=.1
snmpd.mib2=view
snmpd.mib2.viewname=mib2
snmpd.mib2.type=included
snmpd.mib2.oid=.iso.org.dod.Internet.mgmt.mib-2
```

26.3.5.2 View settings using package options

```
config 'view' 'all'
      option viewname 'all'
      option type 'included'
      option oid '.1'

config 'view' 'mib2'
      option viewname 'mib2'
      option type 'included'
      option oid '.iso.org.dod.Internet.mgmt.mib-2'
```

26.3.6 Access settings

The following example shows the “public” group being granted read access on the “all” view and the “private” group being granted read and write access on the “all” view.

26.3.6.1 Access using package options

```
config 'access' 'public_access'
      option group 'public'
      option context 'none'
      option version 'any'
      option level 'noauth'
      option prefix 'exact'
      option read 'all'
      option write 'none'
      option notify 'none'
```

```
config 'access' 'private_access'
      option group 'private'
      option context 'none'
      option version 'any'
      option level 'noauth'
      option prefix 'exact'
      option read 'all'
      option write 'all'
      option notify 'all'
```

26.3.7 SNMP traps settings

26.3.7.1 SNMP trap using UCI

```
snmpd.@trapreceiver[0]=trapreceiver
snmpd.@trapreceiver[0].host=1.1.1.1:161
snmpd.@trapreceiver[0].version=v1
snmpd.@trapreceiver[0].community=public
```

26.3.7.2 SNMP trap using package options

```
# for SNMPv1 or v2c trap receivers
config trapreceiver
   option host 'IPADDR[:PORT]'
   option version 'v1|v2c'
   option community 'COMMUNITY STRING'
# for SNMPv2c inform request receiver

config informreceiver
   option host 'IPADDR[:PORT]'
   option community 'COMMUNITY STRING'
An additional option was added to the 'agent' subsection:
   option authtrapenabled '0|1
```

27 Configuring VRRP

27.1 Overview

Virtual Router Redundancy Protocol (VRRP) is a networking protocol designed to eliminate the single point of failure inherent in the static default routed environment.

VRRP specifies an election protocol that dynamically assigns responsibility for a virtual router to one of the VRRP routers on a LAN. The VRRP router controlling the IP address(es) associated with a virtual router is called the Master, and forwards packets sent to these IP addresses. The election process provides dynamic failover in the forwarding responsibility from the Master to a backup router should the Master become unavailable. This process allows the virtual router IP address(es) on the LAN to be used as the default first hop router by end hosts. The advantage gained from using VRRP is a higher availability default path without requiring configuration of dynamic routing or router discovery protocols on every end host.

Two or more routers forming the redundancy cluster are configured with the same Router ID and Virtual IP address. A VRRP router group operates within the scope of the single LAN. Additionally, the VRRP routers are configured with its initial role (Master or Backup) and the router priority, which is a factor in the master router election process. A password authentication may also be configured to protect VRRP protocol messages against spoofing.

The VRRP protocol is implemented according to Internet standard RFC2338.

27.2 Configuration package used

Package	Sections
vrrp	

27.3 Configuring VRRP using the web interface

To configure VRRP through the web interface, in the top menu, select **Network -> VRRP**. The VRRP page appears. To access configuration settings, click **ADD**.

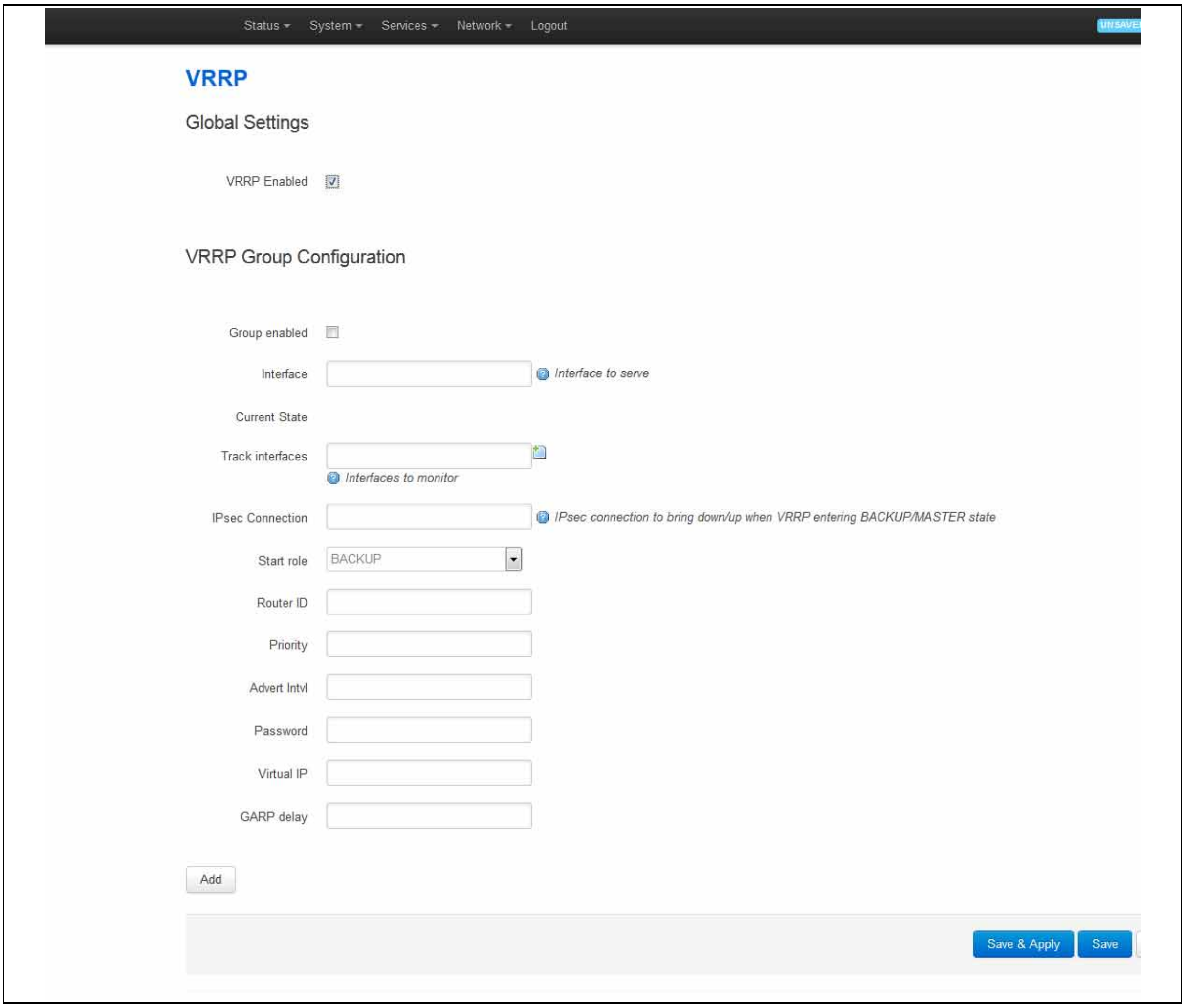


Figure 119: The VRRP group configuration page

Web Field / UCI / Package Option	Description
Global settings	
Web: VRRP Enabled UCI: vrrp.main.enabled Opt: Enabled	Globally enables VRRP on the router. 0 Disabled. 1 Enabled
VRRP Group Configuration	
Web: Group Enabled UCI: vrrp.g1.enabled Opt: Enabled	Enables a VRRP group on the router. 0 Disabled. 1 Enabled
Web: Interface UCI: vrrp.g1.interface Opt: interface	Sets the local LAN interface name in which the VRRP cluster is to operate. For example, 'lan'. The interface name is taken from the package network.
Web: Track Interfaces UCI: vrrp.g1.track_iface Opt: track_iface	Sets one or more WAN interfaces that VRRP should monitor. If a monitored interface goes down on the Master VRRP router, it goes into 'Fault' state and the Backup VRRP router becomes the Master.
Web: IPSec connection UCI: vrrp.g1.ipsec_connection Opt: ipsec_connection	Sets which IPSec connection to bring up or down when VRRP enters 'Backup/Master' state.

<table>
<tr><td>Web: Start role
UCI: vrrp.g1.init_state
Opt: init_state</td><td>Sets the initial role in which a VRRP router starts up. In a cluster of VRRP routes, set one as a Master and the others as Backup.<table><tr><td>BACKUP</td><td></td></tr><tr><td>MASTER</td><td></td></tr></table></td></tr>
<tr><td>Web: Router ID
UCI: vrrp.g1.router_id
Opt: router_id</td><td>Sets the VRRP router ID (1 to 255). All co-operating VRRP routers serving the same LAN must be configured with the same router ID.<table><tr><td>0</td><td></td></tr><tr><td>Range</td><td>1-255</td></tr></table></td></tr>
<tr><td>Web: Priority
UCI: vrrp.g1.priority
Opt: priority</td><td>Sets the VRRP router’s priority. Higher values equal higher priority. The VRRP routers must use priority values between 1-254. The Master router uses a higher priority.<table><tr><td>0</td><td></td></tr><tr><td>Range</td><td>0-255</td></tr></table></td></tr>
<tr><td>Web: Advert intvl
UCI: vrrp.g1.advert_int_sec
Opt: advert_int_sec</td><td>Sets the VRRP hello value in seconds. This value must match the value set on a peer.<table><tr><td>0</td><td></td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
<tr><td>Web: Password
UCI: vrrp.g1.password
Opt: password</td><td>Sets the password to use in the VRRP authentication (simple password authentication method). This field may be left blank if no authentication is required.</td></tr>
<tr><td>Web: Virtual IP
UCI: vrrp.g1.virtual_ipaddr
Opt: virtual_ipaddr</td><td>Sets the virtual IP address and mask in prefix format. For example, ’11.1.1.99/24’. All co-operating VRRP routers serving the same LAN must be configured with the same virtual IP address.</td></tr>
<tr><td>Web: GARP
UCI: vrrp.g1.garp_delay_sec
Opt: garp_delay_sec</td><td>Sets the Gratuitous ARP message sending delay in seconds.<table><tr><td>5</td><td></td></tr><tr><td>Range</td><td></td></tr></table></td></tr>
</table>

Table 93: Information table for VRRP settings

27.4 Configuring VRRP using UCI

You can configure VRRP through CLI using UCI commands.

The configuration file is stored at:

/etc/config/vrrp

To view the configuration in UCI format, use the command:

uci export vrrp

```
~# uci export vrrp
config vrrp 'main'
        option enabled 'yes'
config vrrp_group 'g1'
        option enabled 'yes'
```

```
        option interface 'lan1'
        list track_iface 'lan'
        option init_state 'BACKUP'
        option router_id '1'
        option priority '115'
        option advert_int_sec '2'
        option password 'secret'
        option virtual_ipaddr '10.1.10.150/16'
        option garp_delay_sec '5'
       option ipsec_connection 'Test'
```

or use the command: **uci show vrrp**

```
~# uci show vrrp
vrrp.main=vrrp

vrrp.main.enabled=yes
vrrp.g1=vrrp_group
vrrp.g1.enabled=yes
vrrp.g1.interface=lan1
vrrp.g1.track_iface=lan
vrrp.g1.init_state=BACKUP
vrrp.g1.router_id=1
vrrp.g1.priority=115
vrrp.g1.advert_int_sec=2
vrrp.g1.password=secret
vrrp.g1.virtual_ipaddr=10.1.10.150/16
vrrp.g1.garp_delay_sec=5
vrrp.g1.ipsec_connection=Test
```

To change any of the above values use uci set command.

28 Configuring Multicasting using PIM and IGMP interfaces

28.1 Overview

IP multicast is a bandwidth-conserving technology that reduces traffic by simultaneously delivering a single stream of information to potentially thousands of corporate recipients. Applications that take advantage of multicast include video conferencing and corporate communications.

IP multicast delivers application source traffic to multiple receivers without burdening the source or the receivers while using a minimum of network bandwidth.

PIM (Protocol Independent Multicast) and IGMP (Internet Group Management Protocol) are protocols used to create multicasting networks within a regular IP network.

A multicast group is an arbitrary group of receivers that expresses an interest in receiving a particular data stream. The receivers (the designated multicast group) are interested in receiving a data stream from the source. They indicate this by sending an Internet Group Management Protocol (IGMP) host report to their closest router in the network. The routers are then responsible for delivering the data from the source to the receivers. The routers use Protocol Independent Multicast (PIM) between themselves to dynamically create a multicast distribution tree. The data stream will then be delivered only to the network segments that are in the path between the source and the receivers.

To summarize: PIM is used between routers while IGMP is used between a receiver and its router only. As a result, PIM must be enabled on all the interfaces on the route from the multicast source to the multicast client while IGMP must be enabled on the interface to the multicast client only.

28.2 Configuration package used

Package	Sections
pimd	pimd interface

28.3 Configuring PIM and IGMP using the web interface

To configure PIM through the web interface, in the top menu, select **Network -> PIM**. The PIM page appears. To access the Global settings, click **Add**.

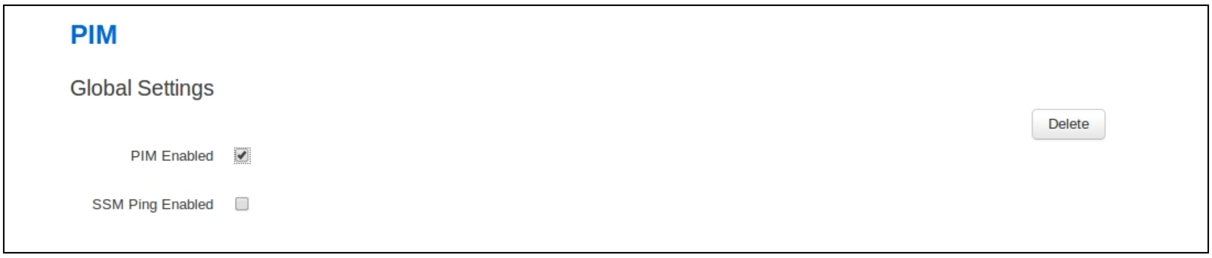


Figure 120: The global settings interface

28.3.1 Global settings

Web Field / UCI / Package Option	Description
Global settings	
Web: PIM Enabled UCI: pimd.pimd.enabled Opt: enabled	Globally enables PIM on the router. 0 — Disabled. 1 — Enabled
Web: SSM Ping Enabled UCI: pimd.pimd.ssmpingd Opt: ssmpingd	Enables answers to SSM pings. 0 — Disabled. 1 — Enabled

Table 94: Information table for PIM global settings

28.3.2 Interfaces configuration

Figure 121: The interfaces configuration section

Web Field / UCI / Package Option	Description
Interface settings	
Web: Enabled UCI: pimd.interface[x].enabled Opt: enabled	Enables multicast management of the given interface by the PIM application. 0 — Disabled. 1 — Enabled.
Web: Interface UCI: pimd.interface[x].interface Opt: interface	Selects the interface to apply PIM settings to.

<table>
<tr><td>Web: Enable IGMP
UCI: pimd.interface[x].igmp
Opt: igmp</td><td>Enable IGMP on given interface.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled</td></tr></table>Note: you must enable PIM SSM and/or IGMP depending on your requirements.
ICMP must be enabled on the interface to the multicast client only.</td></tr>
<tr><td>Web: Enable SSM
UCI: pimd.interface[x].ssm
Opt: ssm</td><td>Enable SSM on given interface.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled</td></tr></table></td></tr>
</table>

Table 95: Information table for interface settings

To save your configuration updates, click **Save & Apply**.

28.4 Configuring PIM and IGMP using UCI

You can configure PIM and IGMP through CLI using UCI.

The configuration file is stored at:

/etc/config/pimd

To view the configuration file, enter:

```
uci export pimd
root@VA_router:/etc/config1# uci export pimd
package pimd
config routing 'pimd'
        option enabled 'yes'

config interface
        option enabled 'yes'
        option interface 'lan'
        option ssm 'yes'
        option igmp 'yes'

config interface
        option enabled 'yes'
        option interface 'wan'
        option ssm 'yes'
        option igmp 'no'
```

Alternatively, enter:

```
uci show pimd
root@VA_router:/etc/config1# uci show pimd
pimd.pimd=routing
pimd.pimd.enabled=yes
pimd.@interface[0]=interface
pimd.@interface[0].enabled=yes
pimd.@interface[0].interface=lan
pimd.@interface[0].ssm=yes
pimd.@interface[0].igmp=yes
pimd.@interface[1]=interface

pimd.@interface[1].enabled=yes
pimd.@interface[1].interface=wan
pimd.@interface[1].ssm=yes
pimd.@interface[1].igmp=no
```

To change any of the above values use uci set command.

29 Event system

Virtual Access routers feature an event system. It allows you to forward router events to predefined targets for efficient control and management of devices.

This chapter explains how the event system works and how to configure it using UCI commands.

29.1 Configuration package used

Package	Section
va_eventd	main
	forwarding
	target
	conn_tester

29.2 Implementation of the event system

The event system is implemented by the va_eventd application.

The va_eventd application defines three types of object:

Forwardings	Rules that define what kind of events should be generated. For example, you might want an event to be created when an IPSec tunnel comes up or down.
Targets	Define the targets to send the event to. The event may be sent to a target via a syslog message, a snmp trap or email.
Connection testers	Define methods to test the target is reachable. IP connectivity to a server and link state may be checked prior to sending events.

For example, if you want to configure an SNMP trap to be sent when an IPSec tunnel comes up, you will need to:

- Define a forwarding rule for IPSec tunnel up events.
- Set an SNMP manager as the target.
- Optionally use a connection tester to ensure the SNMP manager is reachable.

29.3 Supported events

Events have a class, ID, name and a severity. These properties are used to fine tune which events to report.

Note: only VA events can be forwarded using the event system. A comprehensive table of events is available from the CLI by entering **'vae_cli -d'**.

29.4 Supported targets

The table below describes the targets currently supported.

Target	Description
Syslog	Event sent to syslog server.
Email	Event sent via email.
SNMP	Event sent via SNMP trap.
Exec	Command executed when event occurs.

The attributes of a target vary significantly depending on its type.

29.5 Supported connection testers

The table below describes the methods to test a connection that are currently supported:

Type	Description
link	Checks if the interface used to reach the target is up.
ping	Pings the target. And then assumes there is connectivity during a configurable amount of time.

Table 96: Event system - supported connection tester methods

29.6 Configuring the event system using the web interface

Configuring the event system using the web interface is not currently supported.

29.7 Configuring the event system using UCI

The event system configuration files are stored at:

/etc/config/va_eventd

The configuration is composed of a main section and as many forwardings, targets and connection testers as required.

29.7.1 Va_eventd: main section

29.7.1.1 Main using UCI

```
root@VA_router:~# uci show va_eventd
va_eventd.main=va_eventd
va_eventd.main.enabled=yes
va_eventd.main.event_queue_file=/tmp/event_buffer
va_eventd.main.event_queue_size=128K
```

29.7.1.2 Main using package options

```
root@VA_router:~# uci export va_eventd
package va_eventd

config va_eventd main
	option enabled ‘1’
	option event_queue_file '/tmp/event_buffer'
	option event_queue_size ‘128K’
```

29.7.1.3 Main table options

UCI / Package Option	Description	
UCI: va_eventd.main.enabled Opt: enabled	Enables or disables the event system.	
	0	Disabled.
	1	Enabled
UCI: va_eventd.main.event_queue_file Opt: event_queue_file	File where the events will be stored before being processed. Default file is /tmp/event_buffer.	
	/tmp/event_buffer	
	Range	
UCI: va_eventd.main.event_queue_size Opt: event_queue_size	Maximum size of the event queue in bytes. Default value is 128k.	
	128K	128 kilobytes
	Range	

Table 97: Information table for event settings main section

29.7.2 Va_eventd: forwarding

Forwardings are section rules that define what kind of events should be generated. Multiple forwardings can be defined and each forwarding section can be given a forwarding label for identification. For example:

To define a forwarding label of Monitor using package options:

```
config forwarding ‘Monitor’
```

To define a forwarding label of Monitor using UCI:

```
va_eventd.Monitor=forwarding
```

In the examples below no forwarding label has been defined.

29.7.2.1 Forwarding using UCI

```
root@VA_router:~# uci show va_eventd
va_eventd.@forwarding[0]=forwarding
va_eventd.@forwarding[0].enabled=1
va_eventd.@forwarding[0].className=ethernet
va_eventd.@forwarding[0].eventName=LinkUp
va_eventd.@forwarding[0].severity=warning-critical
va_eventd.@forwarding[0].target=syslog1
```

29.7.2.2 Forwarding using package options

```
root@VA_router:~# uci export va_eventd
config forwarding
        option enabled ‘1’
        option className ‘ethernet’
        option eventName ‘LinkUp’
        option severity ‘warning-critical’
        option target ‘syslog1’
```

29.7.2.3 Forwarding table options

<table>
<tr><th>UCI / Package Option</th><th>Description</th></tr>
<tr><td>UCI: va_eventd.<forwarding label>.enabled
Opt: enabled</td><td>Enables or disables event generation.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled</td></tr></table></td></tr>
<tr><td>UCI: va_eventd.<forwarding label>.className
Opt: className</td><td>Only generate events with the given className. Available class names can be viewed using 'vae_cli -d' command.<table><tr><th>ClassName</th></tr><tr><td>internal</td></tr><tr><td>mobile</td></tr><tr><td>ethernet</td></tr><tr><td>isdn</td></tr><tr><td>power</td></tr><tr><td>usage</td></tr><tr><td>pvc</td></tr><tr><td>l2tp</td></tr><tr><td>auth</td></tr><tr><td>ipsec</td></tr><tr><td>wifi</td></tr><tr><td>ppp</td></tr><tr><td>adsl</td></tr><tr><td>system</td></tr><tr><td>ntp</td></tr></table></td></tr>
</table>

UCI: va_eventd.<forwarding label>.eventName Opt: eventName	Only generate events with the given className and the given eventName. The eventName is optional and can be omitted.
UCI: va_eventd.<forwarding label>.severity Opt: severity	Only generate events with a severity in the severity range. This is optional. Severity must be a range in the form severity1-severity2. Example: va_eventd.@forwarding[0].severity=emergency-warning **Severity levels** debug info notice warning error critical alert emergency
UCI: va_eventd.<forwarding label>.target Opt: target	Target to send the event to. This parameter refers to the target name as defined in a target config section.

Table 98: Information table for event system forwarding rules

29.7.3 Va_eventd: connection testers

There are two types of connection testers:

- ping connection tester, and
- link connection tester.

Multiple connection testers can be defined and each forwarding section can be given a label for identification. For example:

To define a connection tester label of Tester1 using package options:

```
config conn_tester ‘Tester1’
```

To define a forwarding label of Tester1 using UCI:

```
va_eventd.Tester1=conn_tester
```

In the examples below no connection tester label has been defined.

29.7.3.1 Ping connection tester

A ping connection tester tests that a connection can be established by sending pings.

If successful, the event system assumed the connection is valid for a configurable amount of time.

29.7.3.2 Ping connection tester using UCI

```
va_eventd.@conn_tester[0]=conn_tester
va_eventd.@conn_tester[0].name=pinger
va_eventd.@conn_tester[0].enabled=1
va_eventd.@conn_tester[0].type=ping
va_eventd.@conn_tester[0].ping_dest_addr=192.168.0.1
va_eventd.@conn_tester[0].ping_source=eth0
va_eventd.@conn_tester[0].ping_success_duration_sec=60
```

29.7.3.3 Ping connection tester using package options

```
config conn_tester
        option name ‘pinger’
        option enabled ‘1’
        option type ‘ping’
        option ping_dest_addr ‘192.168.0.1’
        option ping_source ‘eth0’
        option ping_success_duration_sec ‘60’
```

29.7.3.4 Ping connection tester table options

UCI / Package Option	Description
UCI: va_eventd.<conn_tester label>.name Opt: name	Name of this connection tester. This name is referred to by the target section.
UCI: va_eventd.<conn_tester label>.enabled Opt: enabled	Enable this connection tester. 0 Disabled. 1 Enabled
UCI: va_eventd.<conn_tester label>.type Opt: type	Set to ping for a ping connection tester. ping Ping connection tester link Link connection tester
UCI: va_eventd.<conn_tester label>.ping_dest_addr Opt: ping_dest_addr	IP Address to ping.
UCI: va_eventd.<conn_tester label>.ping_source Opt: ping_source	Source IP Address of the pings. This is optional. It can also be an interface name.
UCI: va_eventd.<conn_tester label>.ping_success_duration_sec Opt: ping_success_duration_sec	Defines the time in seconds the target is considered up for after a successful ping.

Table 99: Information table for ping connection tester settings

29.7.3.5 Link connection tester

A link connection tester tests a connection by checking the status of the interface being used.

29.7.3.6 Link connection tester using UCI

```
va_eventd.@conn_tester[0]=conn_tester
va_eventd.@conn_tester[0].name=linktest
va_eventd.@conn_tester[0].enabled=1
va_eventd.@conn_tester[0].type=link
va_eventd.@conn_tester[0].link_iface=eth0
```

29.7.3.7 Link connection tester using package options

```
config conn_tester
       option name ‘linktest’
       option enabled ‘1’
       option type ‘link’
       option link_iface ‘eth0’
```

29.7.3.8 Link connection tester table options

UCI / Package Option	Description
UCI: va_eventd.<conn_tester label>.name Opt: name	Name of this connection tester. This name is referred to by the target section.
UCI: va_eventd.<conn_tester label>.enabled Opt: enabled	Enable this connection tester. 0 Disabled. 1 Enabled.
UCI: va_eventd.<conn_tester label>.type Opt: type	Set to 'link' for a link connection tester. ping Ping connection tester. link Link connection tester.
UCI: va_eventd.<conn_tester label>.link_iface Opt: link_iface	Interface name to check.

Table 100: Information table for link connection tester settings

29.7.4 Supported targets

There are four possible targets.

- Syslog target
- Email target
- SNMP target
- Exec target

Multiple targets can be defined and each target can be given a label for identification. For example:

To define a connection tester label of Target1 using package options:

```
config target ‘Target1’
```

To define a target label of Target1 using UCI:

```
va_eventd.Target1=target
```

In the examples below no target label has been defined.

29.7.4.1 Syslog target

When a syslog target receives an event, it sends it to the configured syslog server.

29.7.4.2 Syslog target using UCI

```
va_eventd.@target[0]=target
va_eventd.@target[0].name=syslog1
va_eventd.@target[0].enabled=1
va_eventd.@target[0].type=syslog
va_eventd.@target[0].addr=192.168.0.1:514
va_eventd.@target[0].conn_tester=pinger
```

29.7.4.3 Syslog target using package options

```
config target
	option name syslog1
	option enabled ‘1’
	option type ‘syslog’
	option target_addr ‘192.168.0.1:514’
	option conn_tester ‘pinger’
```

29.7.4.4 Syslog target table options

<table>
<tr><th>UCI / Package Option</th><th>Description</th></tr>
<tr><td>UCI: va_eventd.<target label>.name
Opt: name</td><td>Name of the target.This is to be used in the forwarding section</td></tr>
<tr><td>UCI: va_eventd.<target label>.enabled
Opt: enabled</td><td>Enable this target.<table><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled</td></tr></table></td></tr>
<tr><td>UCI: va_eventd.<target label>.type
Opt: type</td><td>Must be 'syslog' for a syslog target.<table><tr><td>syslog</td><td>Syslog target</td></tr><tr><td>email</td><td>Email target</td></tr><tr><td>snmptrap</td><td>SNMP target</td></tr><tr><td>exec</td><td>Exec target</td></tr></table></td></tr>
<tr><td>UCI: va_eventd.<target label>.target_addr
Opt: target_addr</td><td>IP Address or FQDN and Port number to send the syslog message to. If no port is given, 514 is assumed. Format: x.x.x.x:port or FQDN:port</td></tr>
</table>

UCI: va_eventd.<target label>.conn_tester Opt: conn_tester	Name of the connection tester to use for this target.

Table 101: Information table for syslog target settings

29.7.4.5 Email target

When an email target receives an event, it sends it to the configured email address.

29.7.4.6 Email target using UCI

```
va_eventd.@target[0]=target
va_eventd.@target[0].name=email1
va_eventd.@target[0].enabled=1
va_eventd.@target[0].type=email
va_eventd.@target[0].smtp_addr=smtp.site.com:587
va_eventd.@target[0].smtp_user=john_smith@site.com
va_eventd.@target[0].smtp_password=secret word
va_eventd.@target[0].use_tls=0
va_eventd.@target[0].tls_starttls=0
va_eventd.@target[0].tls_forcessl3=0
va_eventd.@target[0].timeout_sec=10
va_eventd.@target[0].from=x@example.com
va_eventd.@target[0].to=y@example.com
va_eventd.@target[0].subject_template=%{severityName} %{eventName}!!!
va_eventd.@target[0].body_template=%{eventName} (%{class}.%{subclass})
happened!
va_eventd.@target[0].conn_tester=pinger
```

29.7.4.7 Email target using package options

```
config target
       option name email1
       option enabled 1
       option type email
       option smtp_addr "smtp.site.com:587"
       option smtp_user 'john_smith@site.com'
       option smtp_password 'secret word'
       option use_tls '0'
       option tls_starttls '0'
```

```
        option tls_forcessl3 '0'
        option timeout_sec "10"
        option from x@example.com
        option to y@example.com
        option subject_template "%{severityName} %{eventName}!!!"
        option body_template "%{eventName} (%{class}.%{subclass}) happened!"
```

29.7.4.8 Option conn_tester 'pinger'email target table options

UCI / Package Option	Description
UCI: va_eventd.<target label>.name Opt: name	Name of the target to be used in the forwarding section.
UCI: va_eventd.<target label>.enabled Opt: enabled	Enable this target. 0: Disabled. 1: Enabled
UCI: va_eventd.<target label>.type Opt: type	Must be 'email' for a syslog target. syslog: Syslog target. email: Email target. snmptrap: SNMP target. exec: Exec target.
UCI: va_eventd.<target label>.smpt_addr Opt: smtp_addr	IP address or FQDNand port of the SMTP server to use. Format: x.x.x.x:port or fqdn:port
UCI: va_eventd.<target label>.smtp_user Opt: smtp_user	Username for smtp authentication.
UCI: va_eventd.<target label>.smtp_password Opt: smtp_password	Password for smtp authentication.
UCI: va_eventd.<target label>.use_tls Opt: use_tis	Enable TLS (Transport Layer Security) support. 0: Disabled. 1: Enabled.
UCI: va_eventd.<target label>.tls_starttls Opt: tis_starttis	Enable StartTLS support. 0: Disabled. 1: Enabled.
UCI: va_eventd.<target label>.tls_forcessl3 Opt: tis_forcessl3	Force SSLv3 for TLS. 0: Disabled. 1: Enabled.
UCI: va_eventd.<target label>.timeout_sec Opt: timeout_sec	Email send timeout in seconds. 10: 10 seconds Range:
UCI: va_eventd.<target label>.from Opt: from	Source email address.
UCI: va_eventd.<target label>.to Opt: to	Destination email address.

UCI: va_eventd.<target label>.subject_template Opt: subject_template	Template to use for the email subject.
UCI: va_eventd.<target label>.body_template Opt: body_template	Template to use for the email body.
UCI: va_eventd.<target label>.conn_tester Opt: conn_tester	Name of the connection tester to use for this target.

Table 102: Information table for email target settings

29.7.5 SNMP target

When a SNMP target receives an event, it sends it in a trap to the configured SNMP manager.

29.7.5.1 SNMP target using UCI

```
va_eventd.@target[0]=target
va_eventd.@target[0].name=snmp1
va_eventd.@target[0].enabled=1
va_eventd.@target[0].type=snmptrap
va_eventd.@target[0].target_addr=192.168.0.1
va_eventd.@target[0].agent_addr=192.168.0.4
va_eventd.@target[0].conn_tester=pinger
```

29.7.5.2 SNMP target using package options

```
config target
       option name ‘snmp1’
       option enabled ‘1’
       option type ‘snmptrap’
       option community ‘public’
       option target_addr ‘192.168.0.1’
       option agent_addr ‘192.168.0.4’
       option conn_tester ‘pinger’
```

29.7.5.3 SNMP target table options

UCI / Package Option	Description
UCI: va_eventd.<target label>.name Opt: name	Name of the target to be used in the forwarding section.
UCI: va_eventd.<target label>.enabled Opt: enabled	Enable this target. 0: Disabled. 1: Enabled

UCI: va_eventd.<target label>.type Opt: type	Must be snmptrap for a snmp target. syslog: Syslog target email: Email target snmptrap: SNMP target exec: Exec target
UCI: va_eventd.<target label>.community Opt: community	Community name to use to send the trap.
UCI: va_eventd.<target label>.target_addr Opt: target_addr	IP address of the SNMP Manager.
UCI: va_eventd.<target label>.agent_addr Opt: agent_addr	Optional IP address to use as the trap source IP address.
UCI: va_eventd.<target label>.conn_tester Opt: conn_tester	Name of the connection tester to use for this target.

Table 103: Information table for snmp target settings

29.7.5.4 Exec target

When an exec target receives an event, it executes a shell command.

29.7.5.5 Exec target using UCI

```
va_eventd.@target[0]=target
va_eventd.@target[0].name=logit
va_eventd.@target[0].enabled=1
va_eventd.@target[0].type=exec
va_eventd.@target[0].cmd_template=logger -t eventer %{eventName}
```

29.7.5.6 Exec target using package options

```
config target
	option name ‘logit’
	option enabled ‘1’
	option type ‘exec’
	option cmd_template "logger -t eventer %{eventName}"
```

29.7.5.7 Exec target table options

UCI / Package Option	Description
UCI: va_eventd.<target label>.name Opt: name	Name of the target to be used in the forwarding section.
UCI: va_eventd.<target label>.enabled Opt: enabled	Enable this target. 0: Disabled. 1: Enabled

<table>
<tr><td>UCI: va_eventd.<target label>.type
Opt: type</td><td>Must be exec for an exec target.<table><tr><td>syslog</td><td>Syslog target</td></tr><tr><td>email</td><td>Email target</td></tr><tr><td>snmptrap</td><td>SNMP target</td></tr><tr><td>exec</td><td>Exec target</td></tr></table></td></tr>
<tr><td>UCI: va_eventd.<target label>.cmd_target
Opt: cmd_target</td><td>Template of the command to execute.</td></tr>
</table>

Table 104: Information table for exec target settings

29.8 Event system diagnostics

29.8.1 Displaying VA events

To view a list of all available class names, events and severity levels enter the command below:

```
vae_cli -d
```

The following is an example of the output from this command:

```
 | Class     | ID  | Name                     | Severity | Specific
Template
| internal  |   1 | EventdConfigErr          | error
| %{p1} %{p2}: %{p3} has bad value..
| internal  |   2 | EventdConfigWarn         | warning
| %{p1} %{p2}: %{p3} has bad value..
| internal  |   3 | EventdConfigUnknown      | informat | %{p1} %{p2}:
field '%{p3}' is no..

| internal  |   4 | EventdSystemErr          | error
| %{p1} %{p2}: %{p3} %{p4} %{p5} %..
| internal  |   5 | EventdSystemWarn         | error
| %{p1} %{p2}: %{p3} %{p4} %{p5} %..
| internal  |   6 | EventdUpAndRunning       | informat |
| internal  |   7 | EventdStopped            | warning  | %{p1}
| mobile    |   1 | SIMin                    | notice   | SIM card #%{p1}
inserted
```

```
| mobile    |   2 | SIMout                  | notice   | SIM card #%{p1}
removed
| mobile    |   3 | LinkUp                  | notice   | 3g link %{p1} up
using sim #%{p2..
| mobile    |   4 | LinkDown                | notice   | 3g link %{p1}
down
| mobile    |   5 | SMSByPassword           | notice   | Received SMS
from %{p1} (by pass..
| mobile    |   6 | SMSByCaller             | notice   | Received SMS
from %{p1} (%{p2}):..
| mobile    |   7 | SMSFromUnknown          | warning  | Received SMS from
unknown sender..
| mobile    |   8 | SMSSendSuccess          | informat | SMS send
success: %{p1}
| mobile    |   9 | SMSSendError            | warning  | SMS send
error: %{p1}
| mobile    |  10 | SMSSent                 | notice   | Sent SMS
to %{p1}: %{p2}
| ethernet  |   1 | LinkUp                  | notice   | Ethernet %{p1} up
| ethernet  |   2 | LinkDown                | notice   | Ethernet %{p1}
down
| auth      |   2 | BadPasswordSSH          | warning  | SSH login attempt
from %{p2}: ba..
| auth      |   3 | BadUserConsole          | warning  | Console login
attempt on %{p1}: ..
| auth      |   4 | BadPasswordConsole      | warning  | Console login
attempt on %{p2}: ..
| auth      |   5 | BadUserTelnet           | warning  | Telnet login
attempt: bad username
| auth      |   6 | BadPasswordTelnet       | warning  | Telnet login
attempt: bad passwo..
| auth      |   7 | BadUserLuCI             | warning  | LuCI login
attempt: bad username..
| auth      |   8 | BadPasswordLuCI         | warning  | LuCI login
attempt: bad password..
| auth      |   9 | LoginSSH                | notice   | SSH login:
user %{p2} from %{p3}
| auth      |  10 | LogoffSSH               | notice   | SSH logoff:
user %{p1} due to "%..
| auth      |  11 | LoginConsole            | notice   | Console login:
user %{p1} on %{p2}
| auth      |  12 | LogoffConsole           | notice   | Console logoff
```

```
on %{p1}
| auth      |  13 | LoginTelnet               | notice   | Telnet login:
user %{p1}
| auth      |  14 | LoginLuCI                 | notice   | LuCI login:
user %{p1}
| auth      |  15 | ConsoleCommand            | informat | %{p1}@%{p2} %{p3}
| auth      |  16 | LuCIAction                | informat
| %{p1}@%{p2} %{p3} %{p4} %{p5}
| ipsec     |   6 | IPSecInitIKE              | informat | IPSec IKE %{p1}
established
| ipsec     |   7 | IPSecInitSA               | informat | IPSec SA %{p1}
established
| ipsec     |   8 | IPSecCloseIKE             | informat | IPSec IKE %{p1}
deleted
| ipsec     |   9 | IPSecCloseSA              | informat | IPSec SA %{p1}
closed
| ipsec     |  10 | IPSecDPDTimeOut           | informat | IPSec IKE %{p1}
DPD timed out
| wifi      |   1 | WiFiConnectedToAP         | notice   | WiFi %{p1}
connected to AP %{p2}
| wifi      |   1 | WiFiConnectedToAP         | notice   | WiFi %{p1}
connected to AP %{p2}
| wifi      |   2 | WiFiDisconnectedFromAP    | notice   | WiFi %{p1}
disconnected from AP
| wifi      |   2 | WiFiDisconnectedFromAP    | notice   | WiFi %{p1}
disconnected from AP
| wifi      |   3 | WiFiStationAttached       | notice   | WiFi
station %{p2} connected to ..
| wifi      |   3 | WiFiStationAttached       | notice   | WiFi
station %{p2} connected to ..
| wifi      |   4 | WiFiStationDetached       | notice   | WiFi
station %{p2} disconnected ..
| wifi      |   4 | WiFiStationDetached       | notice   | WiFi
station %{p2} disconnected ..
| wifi      |   5 | WiFiStationAttachFailed   | notice   | WiFi
station %{p2} failed to con..
| wifi      |   5 | WiFiStationAttachFailed   | notice   | WiFi
station %{p2} failed to con..
| ppp       |   1 | LinkUp                    | informat | PPP for
interface %{p2} (protoco..
| ppp       |   2 | LinkDown                  | informat | PPP for
interface %{p2} (protoco..
| ppp       |   3 | ConnEstablished           | informat | PPP connection
```

```
for interface %{p..
| adsl      |   1 | LinkUp                  | notice   | ADSL trained.
Starting interface..
| adsl      |   2 | LinkDown                | notice   | ADSL down.
Stopping interface %{..
| adsl      |   3 | Silent                  | debug    | ADSL silent
| adsl      |   4 | Training                | debug    | ADSL training
| adsl      |   5 | TrainingSuccess         | notice   | ADSL training
successfull: data ..
| system    |   1 | BootSuccess             | informat | Success booting
into %{p1}
| system    |   2 | DigitalInputChange      | notice   | Digital
Input %{p1} changed valu..
| ntp       |   1 | InitialSync             | notice   | Initial NTP sync:
time: %{p1}; o..
| ntp       |   2 | Adjust                  | informat | NTP adjust
by %{p1}
| ntp       |   3 | QueryTimeout            | warning  | NTP query
to %{p1} timed out. Ne..
| ntp       |   4 | QueryFailed             | warning  | NTP query
failed: %{p1}
```

29.8.2 Viewing the event system config

To view the event system configuration via UCI

```
root@VA_router:~# uci show va_eventd
```

To view the event system config via package options

```
root@VA_router:~# uci export va_eventd
```

29.9 Example of event system configuration

As an example, the event system can be configured to:

- Forward the “l2tp” event “CannotFindTunnel” with a severity between debug and critical to a syslog server
- Forward all “mobile” events with a severity between notice and critical to a SNMP trap manager
- Execute “logger -t eventer %{eventName}” when an “Ethernet” event occurs

- Forward all “auth” events via email
- Connection to the SNMP and syslog server is checked by sending pings
- Connection to the smtp server is verified by checking the state of “eth0”

Example of output event package configuration:

```
package va_eventd

config va_eventd 'main'
        option enabled 'yes'
        option event_queue_file '/tmp/event_buffer'
        option event_queue_size '128K'

config forwarding
        option enabled 'yes'
        option className 'l2tp'
        option eventName 'CannotFindTunnel'
        option severity 'debug-critical'
        option target 'syslog'
config forwarding
        option enabled 'yes'
        option className 'mobile'
        option severity 'notice-critical'
        option target 'snmp'

config forwarding
        option enabled 'yes'
        option className 'ethernet'
        option target 'logit'

config forwarding
        option enabled 'yes'
        option className 'auth'
        option target 'email'
```

```
config conn_tester
        option name 'mon_server'
        option enabled '1'
        option type 'ping'
        option ping_dest_addr '192.168.100.254'
        option ping_source 'eth0'
        option ping_success_duration_sec '10'

config conn_tester
        option name 'smtp_server'
        option enabled '1'
        option type 'link'
        option link_iface 'eth0'

config target
        option name 'syslog'
        option enabled 'yes'
        option type 'syslog'
        option target_addr '192.168.100.254:514'
        option conn_tester 'mon_server'

config target
        option name 'email'
        option enabled 'yes'
        option type 'email'
        option smtp_addr '89.101.154.148:465'
        option smtp_user 'x@example.com'
        option smtp_password '******'
        option use_tls 'yes'
        option tls_starttls 'no'
        option tls_forcessl3 'no'
        option timeout_sec '10'
        option from 'y@example.com'
        option to 'z@example.com'
        option subject_template '%{severityName} %{eventName}!!!'
```

```
        option body_template '%{eventName} (%{class}.%{subclass})
happened!'
        option conn_tester 'smtp_server'

config target
        option name 'snmp'
        option enabled 'yes'
        option type 'snmptrap'
        option community 'public'
        option target_addr '192.168.100.254'
        option agent_addr '192.168.100.1'
        option conn_tester 'mon_server'

config target
        option name 'logit'
        option enabled 'yes'
        option type 'exec'
        option cmd_template 'logger -t eventer %{eventName}'
```

30 Configuring SLA reporting on Monitor

30.1 Introduction

This section describes how to configure and view SLA reporting on Monitor, the Virtual Access monitoring system.

The Virtual Access Monitor system provides:

- centralised access to router connectivity status,
- access to advanced router diagnostic tools, and
- access to SLA Report Management.

When enabled, SLA will present daily graphs for each router for the following:

- Latency – average and max
- Packet loss – average and max
- Signal strength – average and max
- Availability

The SLA Report Manager can build reports from a list of selected routers presenting a range of statistics over extended periods of time.

Note: as well as configuring Monitor for SLA, you must configure each router. To configure the router for Monitor, read the chapter ‘Configuring SLA for a router’.

30.2 Configuring SLA reporting

On the monitoring platform, select a particular router for SLA.

Click **SLA Reporting** tab.

Click **ON**.

When enabled, Monitor will instruct the routers to periodically send up their data for SLA reporting

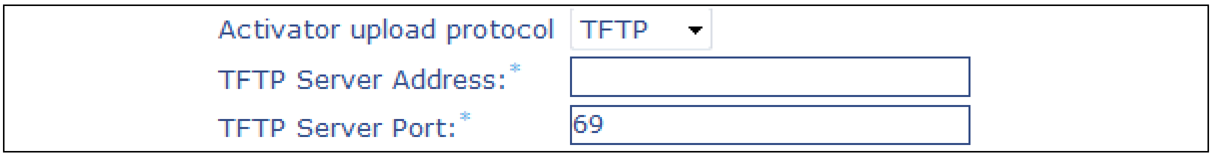
To enable all devices under a particular reseller for SLA, under the SLA tab, click **ON**.

30.3 Configuring router upload protocol

The protocol the router uses to upload the files is set for each device on Monitor. Monitor will send a command to the router to use this protocol to upload the SLA files.

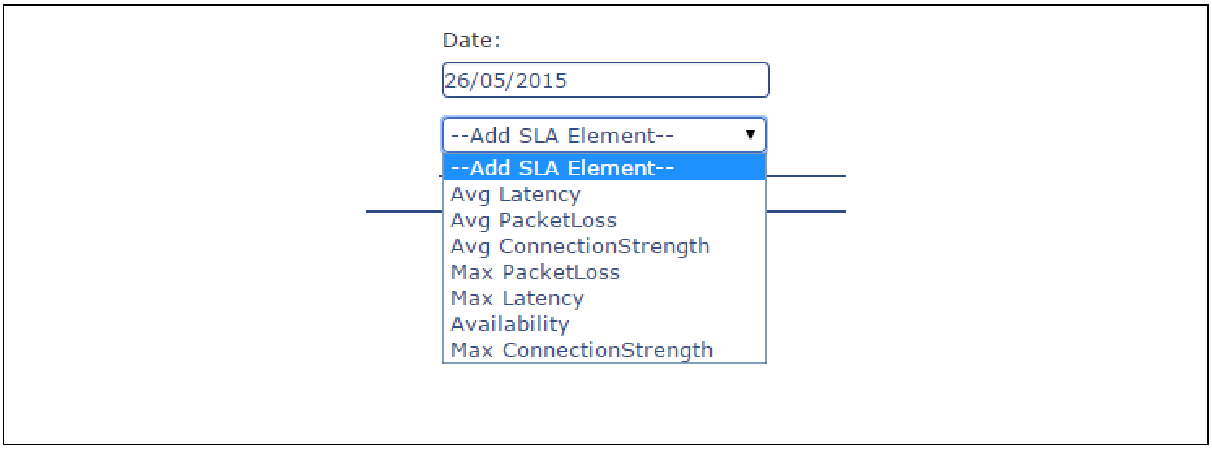
To edit a device, on the device settings page in the Activator Upload Protocol drop-down menu, select the desired protocol and enter in the relevant TFTP Server Address.

Enter the TFTP Server Port number to match.



30.4 Viewing graphs

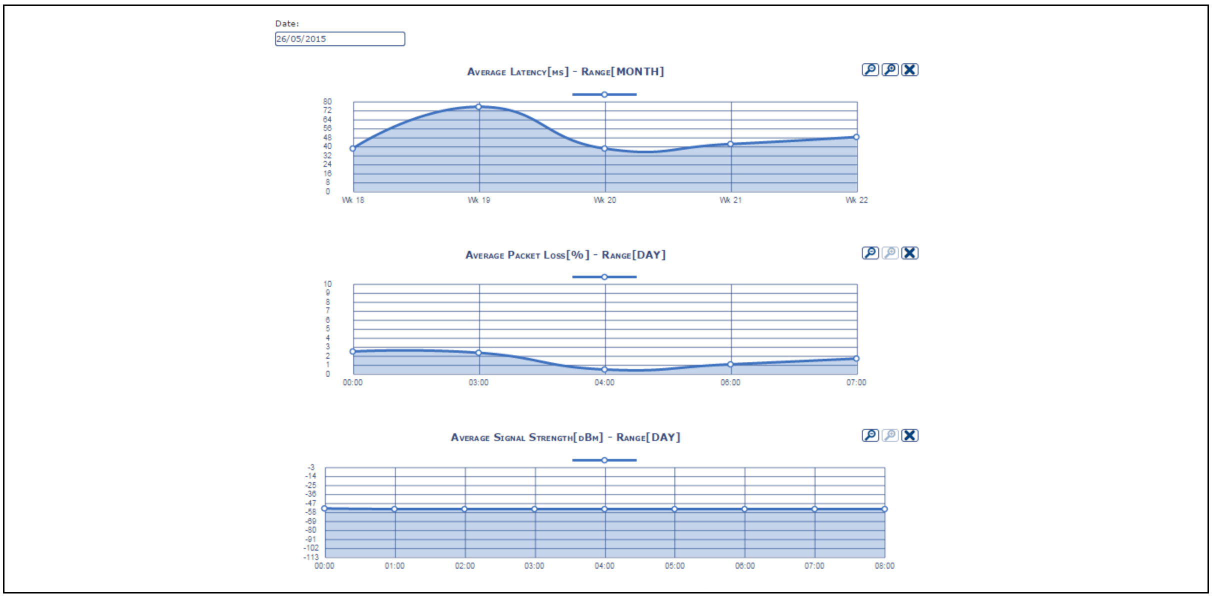
When the router has started to send SLA statistics to the Monitoring platform, default graphs are displayed on the SLA Reporting screen. To view the graphs, simply add the relevant one from the drop-down list.



The following graphs can be displayed.

- Latency (ms) – average and max
- Packet loss (%) – average and max
- Signal strength (dBm) – average and max
- Availability (%)

The graphs appear in daily format. To expand or reduce the time access, use the zoom buttons. To remove a graph, click **X**.



To view raw data, click each graph to produce the following information.


Date:
26/05/2015

Average Latency[ms] - Range[MONTH]

Date	Average Latency [ms]
WEEK 18, May 2015	38.00 ms
WEEK 19, May 2015	75.00 ms
WEEK 20, May 2015	38.00 ms
WEEK 21, May 2015	42.00 ms
WEEK 22, May 2015	48.00 ms

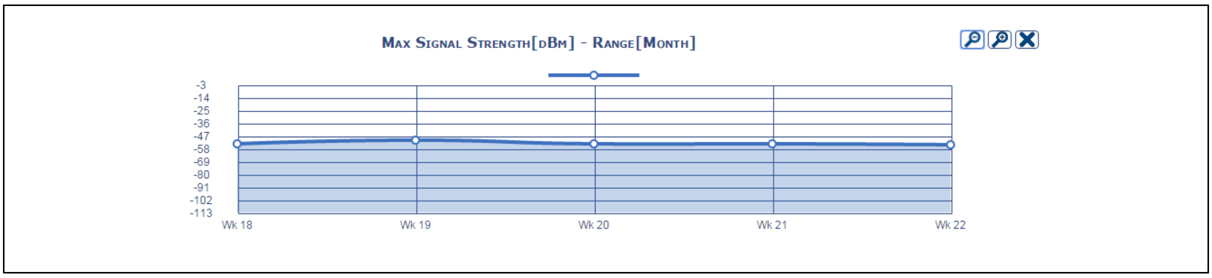
Average Packet Loss[%] - Range[DAY]

Date	Average Packet Loss [%]
00:00:00 26th May 2015	02.44 %
03:00:00 26th May 2015	02.29 %
04:00:00 26th May 2015	00.44 %
06:00:00 26th May 2015	01.02 %
07:00:00 26th May 2015	01.63 %

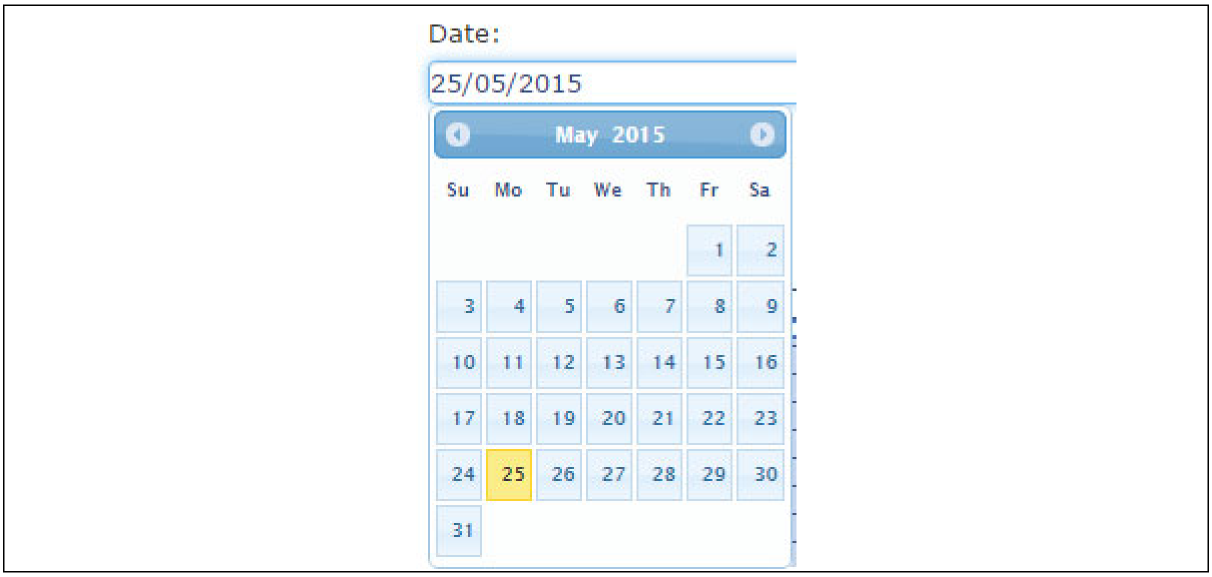
Average Signal Strength[dBm] - Range[DAY]

Date	Average Signal Strength [dBm]
00:00:00 26th May 2015	-54.00 dBm
01:00:00 26th May 2015	-55.00 dBm
02:00:00 26th May 2015	-55.00 dBm
03:00:00 26th May 2015	-55.00 dBm
04:00:00 26th May 2015	-55.00 dBm
05:00:00 26th May 2015	-55.00 dBm
06:00:00 26th May 2015	-55.00 dBm
07:00:00 26th May 2015	-55.00 dBm
08:00:00 26th May 2015	-55.00 dBm

To change the range of the graph, click **zoom**.

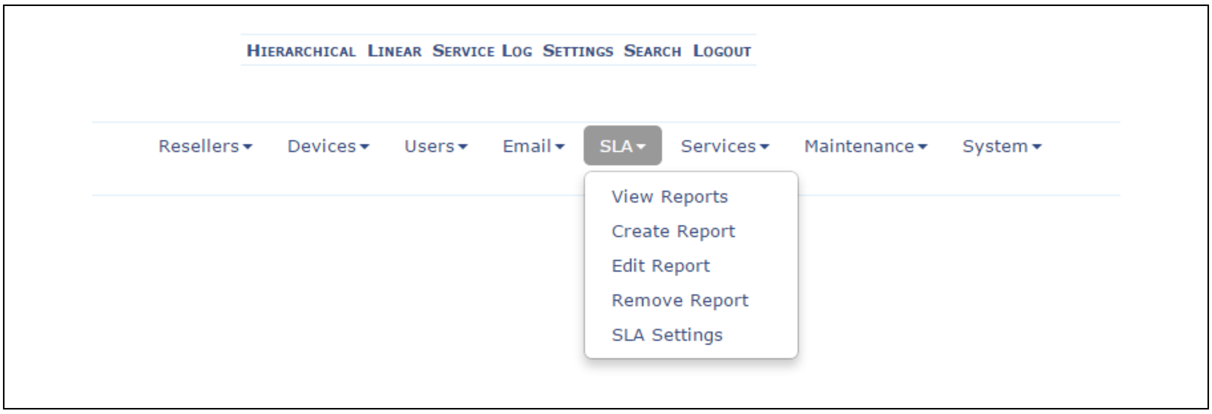


To select a particular day, click **Date**.



30.5 Generating a report

In the top menu, select **Settings**.



Click the **SLA** drop-down menu. The menu has the following options.

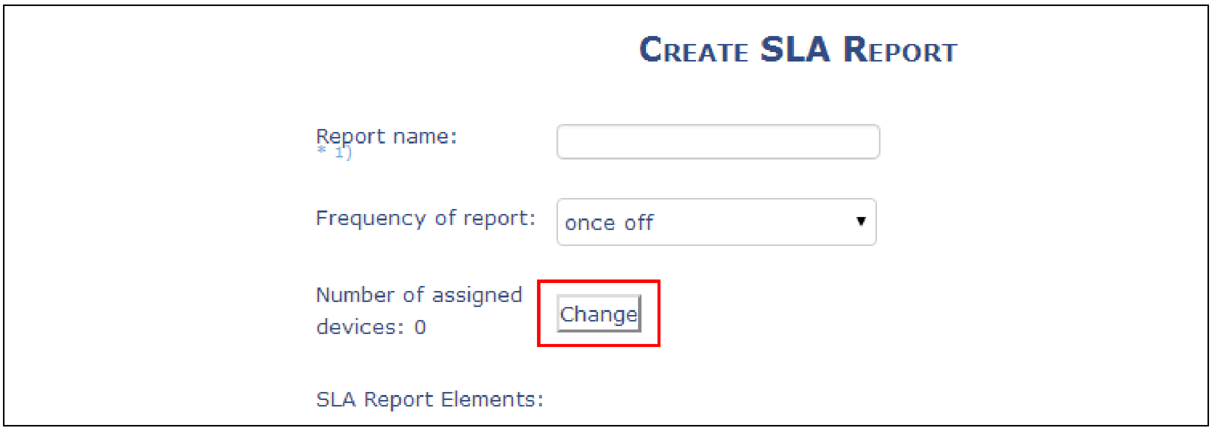
- View Reports
- Create Reports
- Edit Reports
- Remove Reports
- SLA Settings

30.5.1 Create a report

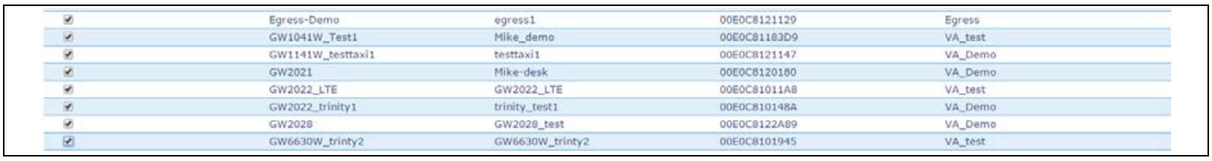
Select **Create Report**. Enter the relevant parameters.

- Report name
- Frequency of report
- Once off
- Hourly
- Daily
- Weekly
- Assigned devices
- SLA Report Elements

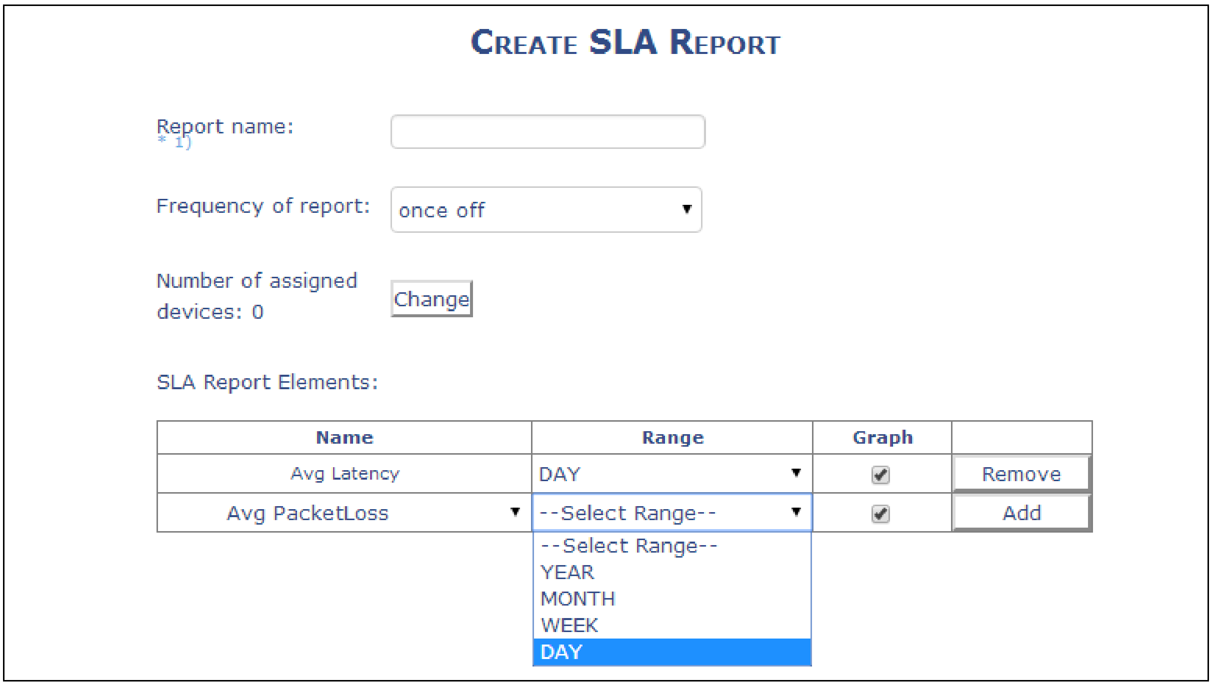
To assign devices to the report, click **Change**.



After clicking **Change**, the select devices page appears, this allows you to select which devices are to be members of the report.



Click **Continue** and then proceed to add SLA Report Elements.



The graph options are:

- Avg Latency
- AvgPacketloss
- AvgConnectionStrength
- Max Latency
- Max Packetloss
- Max ConnectionStrength
- Availability

Select a graph name and then select a relevant range:.

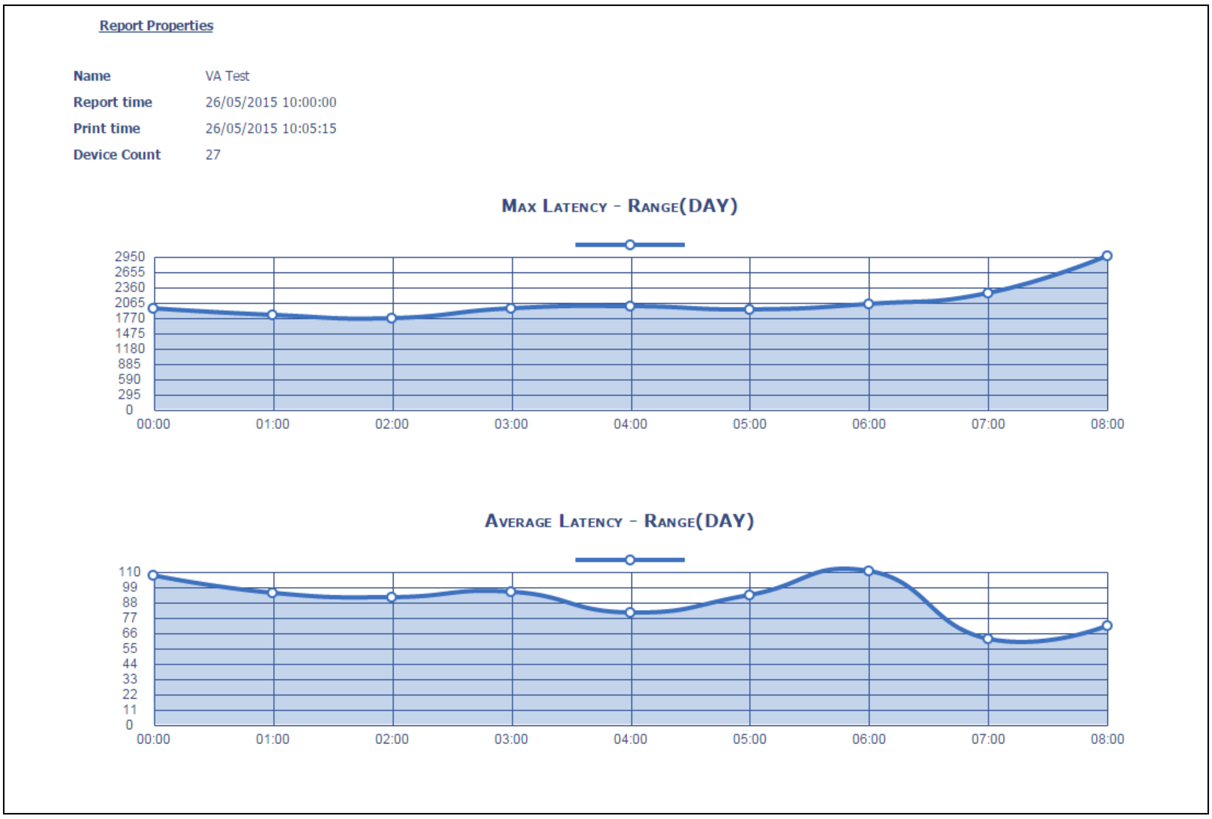
- Year
- Month
- Week
- Day

Click **Add** and when you have selected all graphs, click **Save**.

30.5.2 View reports

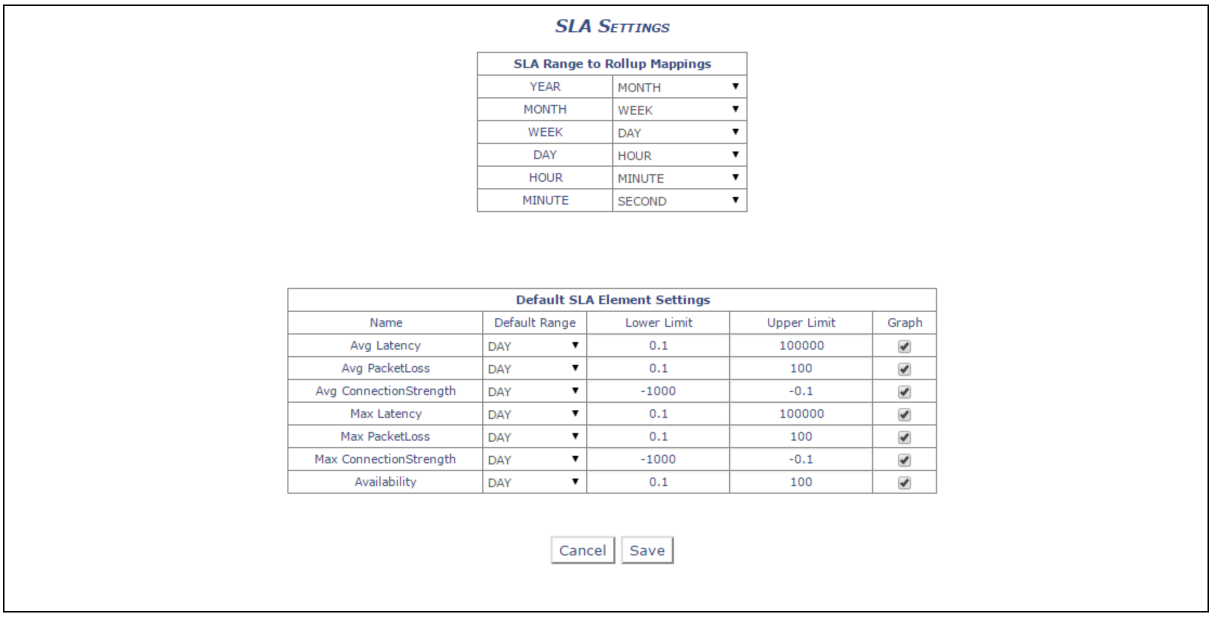
To view a report, select **Settings > SLA > View Reports**.

From the drop down box, select the relevant report and click **Generate**. The report appears as shown below.



30.5.3 SLA settings

By selecting SLA Settings, you can change the SLA parameters.



SLA Settings

SLA Range to Rollup Mappings	
YEAR	MONTH
MONTH	WEEK
WEEK	DAY
DAY	HOUR
HOUR	MINUTE
MINUTE	SECOND

Default SLA Element Settings

Name	Default Range	Lower Limit	Upper Limit	Graph
Avg Latency	DAY	0.1	100000	☑
Avg PacketLoss	DAY	0.1	100	☑
Avg ConnectionStrength	DAY	-1000	-0.1	☑
Max Latency	DAY	0.1	100000	☑
Max PacketLoss	DAY	0.1	100	☑
Max ConnectionStrength	DAY	-1000	-0.1	☑
Availability	DAY	0.1	100	☑

Cancel Save

30.5.3.1 SLA range to rollup mappings

SLA Range to Rollup Mappings allow you to configure what intervals are used for the various date ranges used to display the graphs. For example, the screenshot shows that data will be shown for every minute. If you select **Day**, data will be shown for every day; if you select **Week** range, data will be shown for every week, and so on.

30.5.3.2 Default SLA element settings

The Default SLA Element settings control range and graphs.

Range	Sets what the default range will be when a new user is created.
Graph	Selects whteher each report element is displayed as a graph or in tabular data form.

The view of SLA data is customisable per user. These default values set how graphs appear when you use SLA for the first time. You can then configure your view of SLA by altering the SLA page using the various controls. These changes are remembered by Monitor so that your view of SLA remains the same when you next return to it.Upper and lower limits control what data is to be ignored when generating SLA graphs.

30.6 Reporting device status to Monitor using UCI

The following sample contains the settings to enable the device to report its status to Monitor. To allow Monitor to track the IP address and ongoing presence of the device, a heartbeat SNMP trap is sent by default every minute.

Web Field / UCI / Package Option	Description
UCI: monitor.main.enable Opt: Enable	Enables Monitor to send heartbeats to the router. 0: Disabled. 1: Enabled
UCI: monitor.main.interval_min Opt: interval_min	Specifies the interval at which traps are sent. 1 Range
UCI: monitor.main.dev_reference Opt: dev_reference	Sets a unique identification for this device known to monitor.
UCI: monitor.main.monitor_ip Opt: monitor_ip	Defines the IP address of Monitor. It is possible to specify multiple addresses to which SNMP heartbeat traps will be sent.

A sample Monitor configuration is shown below.

```
root@VA_router:~# uci show monitor
monitor.main=keepalive
monitor.main.enable=yes
monitor.main.interval_min=1
monitor.main.dev_reference=mikesamazondev
monitor.main.monitor_ip=10.1.83.36
root@VA_router:~# uci export monitor
```

```
package 'monitor'

config 'keepalive' 'main'
      option 'enable' "yes"
      optioninterval_min "1"
      option 'dev_reference' "mikesamazondev"
      list 'monitor_ip' "10.1.83.36"
```

31 Configuring SLA for a router

SLA reporting works in two parts:

The Virtual Access Monitor system server connects via SSH into the router and schedules the task of uploading statistics to Monitor.

The Virtual Access router monitors UDP keepalive packets. It creates and stores statistics in bins. These statistics are uploaded every hour to the Monitor server.

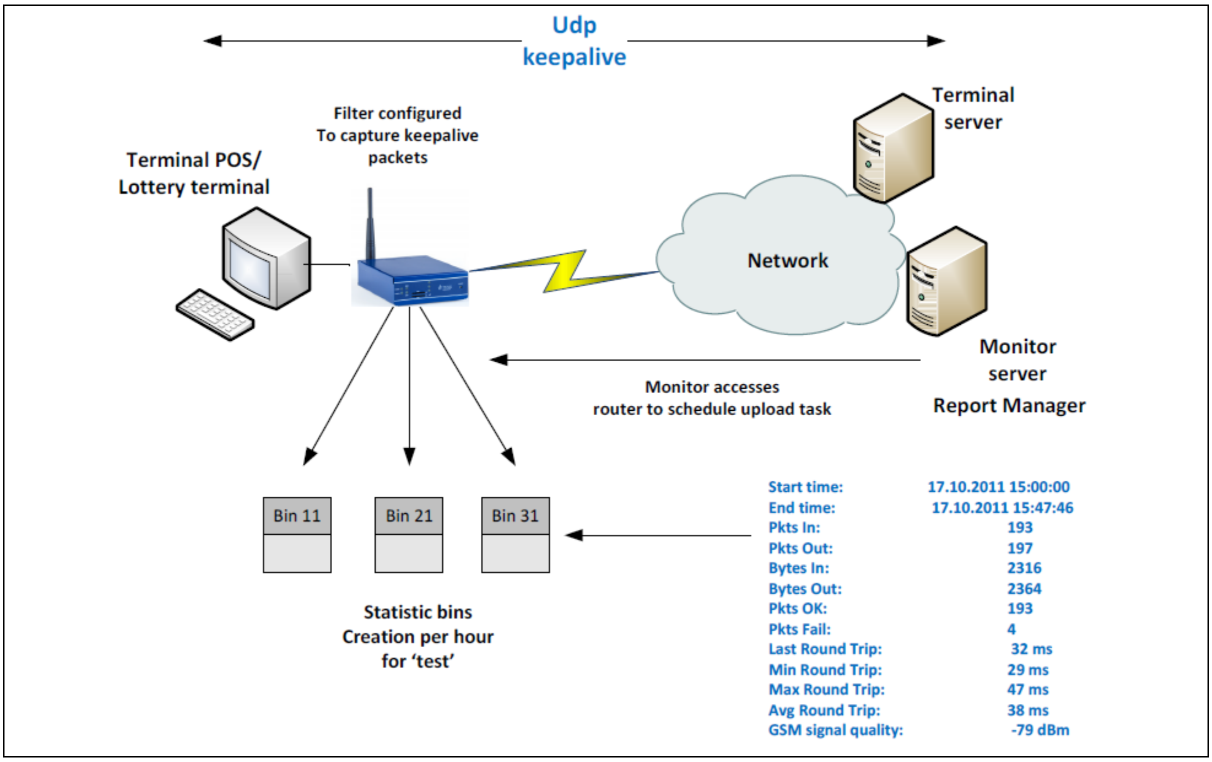


Figure 122: The SLA function

This section describes how to configure SLA on a router. For information on how to configure Monitor for SLA reporting read the previous section 'Configuring SLA on Monitor'.

31.1 Configuration package used

Package	Section
slad	

31.2 Configuring SLA for a router using the web interface

In the top menu, select **Services -> SLA Daemon**. The SLA Daemon page appears.

In the Basic Settings section, click **Add**. The basic settings section for SLA Daemon appears.

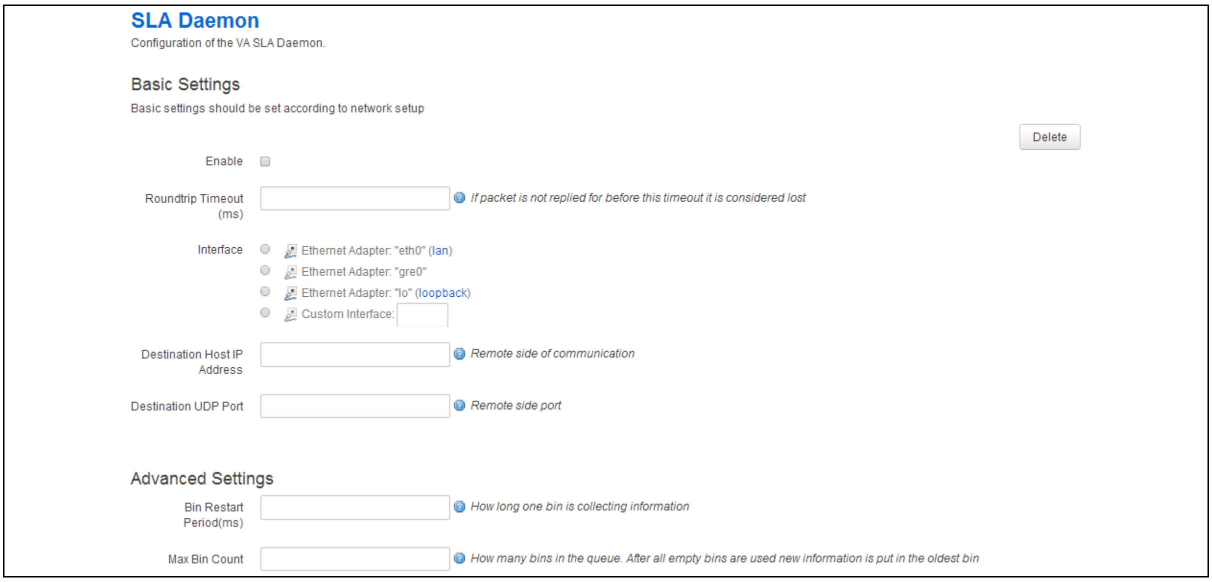


Figure 123: The SLA daemon page

Web Field / UCI / Package Option	**Description**
Web: Enable UCI: slad.main.enable Opt: Enable	Enables or disables SLAD application. 0 Disabled. 1 Enabled.
Web: Roundtrip Timeout (ms) UCI: slad.main.roundtrip_timeout_msec Opt: roundtrip_timeout_msec	Specifies the time in milliseconds that a packet is not replied before this timeout expires and is considered as lost. 0 Disabled. 1 Enabled.
Web: Interface UCI: slad.main.interface Opt: interface	Specifies the interface on which traffic should be monitored.
Web: Destination Host IP Address UCI: slad.main_destination_host_ip_address Opt: destination_host_ip_address	Specifies the destination IP address for the keepalive packets that are originated on the LAN.
Web: Destination UDP port UCI: slad.main. destination_udp_ip_address Opt: destination_udp_ip_address	Specifies the destination UDP port for the keepalive packets that are originated on the LAN.
Web: Bin Restart Period (ms) UCI: slad.main.bin_restart_period_msec Opt: bin_restart_period_msec	Specifies how long one bin is collecting information.
Web: Max Bin Count UCI: slad.main.max_bin_count Opt: max_bin_count	Specifies how many bins are in the queue. After all empty bins are used new information is put in the oldest bin.

Table 105: Information table for SLA settings

When you have made all your configuration changes, click **Save & Apply**.

31.3 Configuring SLA for a router using the UCI interface

You can also configure SLA UCI using UCI command suite.

The configuration file is stored at:

/etc/config/slad

To view the configuration file, enter:

uci export slad

or

uci show slad

```
uci export slad
package slad
config slad 'main'
        option enable 'yes'
        option roundtrip_timeout_msec '5000'
        option interface 'lan'
        option destination_host_ip_address '10.1.1.2'
        option destination_udp_port '53'
        option bin_restart_period_msec '3600000'
        option max_bin_count '73'
uci show slad
slad.main=slad

slad.main.enable=yes
slad.main.roundtrip_timeout_msec=5000
slad.main.interface=lan
slad.main.destination_host_ip_address=10.1.1.2
slad.main.destination_udp_port=53
slad.main.bin_restart_period_msec=3600000
slad.main.max_bin_count=73
```

31.4 Viewing SLA statistics using UCI

To show all available statistic options, enter:

```
root@VA_router:~# sla
sla [current] | [all] | [oldest] | [newest] | [newest N] | [range:
YYYMMDDHH-YYYYMMDDHH]
```

Option	Description
current	Shows current sla bin
all	Shows all bin stored on the router
oldest	Shows the oldest sla bin stored
newest	Shows two newest valid bins
newest N	Shows the newest valid bin
range YYYYMMDDHH-YYYYMMDDHH	Shows all bins that match specified time range

Type the command `sla current` To show current statistics, enter:

```
root@VA_router: ~# sla current
----------------------------------------
Bin valid:              no
Start time              01.01.1970 03:34:00
End time                        n/a
Pkts In:                        1
Pkts Out:                       1
Bytes In:                       15
Bytes Out:              15
Pkts OK:                        1
Pkts Fail:              0
Last Round Trip:                1 ms
Min Last Trip:          1 ms
Max Round Trip:         1 ms
Avg Round Trip:         1 ms
Min GSM signal quality:         n/a
Max GSM signal quality:         n/a
Avg GSM signal quality n/a
Availability:           100.00%
```

To show the newest statistics, enter:

```
root@VA_router: ~# sla newest
----------------------------------------
Bin valid:              yes
Start time              01.01.1970 03:32:00
End time                        01.01.1970 03:33:00
Pkts In:                        6
Pkts Out:                       6
Bytes In:                       90
Bytes Out:              90
Pkts OK:                        6
Pkts Fail:              0
Last Round Trip:                0 ms
Min Last Trip:          1 ms
Max Round Trip:         1 ms
Avg Round Trip:         1 ms
Min GSM signal quality:         -63 dBm
Max GSM signal quality:         -63 dBm
Avg GSM signal quality -63 dBm
Availability:           100.00%
```