



SATEL-GW100/120 User manual

SATEL-GW100

SATEL-GW120



Issue: 1.97

Date: 17 October 2017

1	Introduction	9
1.1	Document scope	9
1.2	Using this documentation	9
2	GW100 Series hardware	12
2.1	Hardware model variants	12
2.2	GW100 Series hardware features	12
2.3	GSM technology.....	13
2.4	WiFi technology for GW120	13
2.5	Power supply.....	13
2.6	GW100/GW120 Series router dimensions.....	13
2.7	Compliance	14
2.8	Operating temperature range	14
2.9	Antenna.....	14
2.10	Components.....	15
2.11	Inserting a SIM card.....	16
2.12	Connecting the SIM lock	16
2.13	Connecting cables	16
2.14	Connecting the antenna	16
2.15	Powering up	16
2.16	Reset button	16
3	GW100 and GW120 Series LED behaviour.....	18
3.1	Main LED behaviour.....	18
3.2	GW100 and GW120 Series Ethernet port LED behaviour	19
4	Factory configuration extraction from SIM card	20
5	Accessing the router.....	21
5.1	Configuration packages used	21
5.2	Accessing the router over Ethernet using the web interface.....	21
5.3	Accessing the router over Ethernet using an SSH client	22
5.4	Accessing the router over Ethernet using a Telnet client	23
5.5	Configuring the password.....	23
5.6	Configuring the password using the web interface	23
5.7	Configuring the password using UCI	24
5.8	Configuring the password using package options.....	24
5.9	Accessing the device using RADIUS authentication	25
5.10	Accessing the device using TACACS+ authentication	26
5.11	SSH	29
5.12	Package dropbear using UCI.....	31
5.13	Certs and private keys.....	32
5.14	Configuring a router's web server	33

5.15 Basic authentication (httpd conf)	38
5.16 Securing uhttpd	39
5.17 Displaying custom information via login screen	39
6 Router file structure	41
6.1 System information	41
6.2 Identify your software version	42
6.3 Image files	42
6.4 Directory locations for UCI configuration files	43
6.5 Viewing and changing current configuration	43
6.6 Configuration file syntax	44
6.7 Managing configurations	44
6.8 Exporting a configuration file	45
6.9 Importing a configuration file	46
7 Using the Command Line Interface.....	50
7.1 Overview of some common commands	50
7.2 Using Unified Configuration Interface (UCI)	53
7.3 Configuration files	58
7.4 Configuration file syntax	58
8 Upgrading router firmware.....	60
8.1 Software versions	60
8.2 Upgrading firmware using CLI	66
9 System settings	69
9.1 Configuration package used	69
9.2 Configuring system properties	69
9.3 System settings using UCI	73
9.4 System diagnostics	74
10 Configuring an Ethernet interface on a GW100 router	76
10.1 Configuration packages used	76
10.2 Configuring an Ethernet interface using the web interface	76
10.3 Interface overview: editing an existing interface	77
10.4 Configuring an Ethernet interface using UCI	86
10.5 Interface diagnostics	89
11 Configuring VLAN	91
11.1 Maximum number of VLANs supported	91
11.2 Configuration package used	91
11.3 Configuring VLAN using the web interface	91
11.4 Viewing VLAN interface settings	94
11.5 Configuring VLAN using the UCI interface	95
12 Configuring ignition sense	96
12.1 Configuration packages used	96

12.2	Configuring vapowermond using the web interface	96
12.3	Configuring vapowermond using the command line	98
12.4	Ignition sense diagnositcs	99
13	Configuring a WiFi connection	100
13.1	Configuration packages used	100
13.2	Configuring a WiFi interface using the web interface	100
13.3	Configuring WiFi in AP mode.....	106
13.4	Configuring WiFi using UCI.....	108
13.5	Creating a WiFi in Client mode using the web interface	111
13.6	Configuring WiFi in Client mode using command line	112
14	Configuring OSPF (Open Shortest Path First)	114
14.1	Introduction	114
14.2	Configuration package used	119
14.3	Configuring OSPF using the web interface	119
14.4	Configuring OSPF using the command line	123
14.5	OSPF using UCI	124
14.6	OSPF using package options.....	125
14.7	OSPF diagnostics	126
14.8	Quagga/Zebra console.....	127
15	Configuring a mobile connection	134
15.1	Configuration package used	134
15.2	Configuring a mobile connection using the web interface.....	134
15.3	Configuring a mobile connection using CLI	140
15.4	Diagnositcs	141
16	Configuring mobile manager.....	145
16.1	Configuration package used	145
16.2	Configuring mobile manager using the web interface	145
16.3	Configuring mobile manager using command line.....	152
16.4	Monitoring SMS	153
16.5	Sending SMS from the router	154
16.6	Sending SMS to the router	154
17	Configuring a GRE interface	155
17.1	Configuration packages used	155
17.2	Creating a GRE connection using the web interface	155
17.3	GRE configuration using command line	160
17.4	GRE configuration using UCI.....	160
17.5	GRE configuration using package options	160
17.6	GRE diagnostics	161
18	Configuring static routes	163
18.1	Configuration package used	163

18.2	Configuring static routes using the web interface	163
18.3	Configuring IPv6 routes using the web interface	164
18.4	Configuring routes using command line	164
18.5	IPv4 routes using UCI.....	165
18.6	IPv4 routes using package options	166
18.7	IPv6 routes using UCI.....	166
18.8	IPv6 routes using packages options.....	166
18.9	Static routes diagnostics	167
19	Configuring BGP (Border Gateway Protocol)	168
19.1	Configuration package used	168
19.2	Configuring BGP using the web interface.....	168
19.3	Configuring BGP using UCI	171
19.4	Configuring BGP using packages options	172
19.5	View routes statistics.....	173
20	Configuring VRRP	174
20.1	Overview	174
20.2	Configuration package used	174
20.3	Configuring VRRP using the web interface	174
20.4	Configuring VRRP using command line	176
21	Configuring Routing Information Protocol (RIP)	179
21.1	Introduction	179
21.2	Configuration package used	180
21.3	Configuring RIP using the web interface.....	180
21.4	Configuring RIP using command line	185
21.5	RIP diagnostics	188
22	Configuring Multi-WAN	193
22.1	Configuration package used	193
22.2	Configuring Multi-WAN using the web interface.....	193
22.3	Multi-WAN traffic rules.....	198
22.4	Configuring Multi-WAN using UCI	198
22.5	Multi-WAN diagnostics	199
23	Automatic operator selection.....	202
23.1	Configuration package used	202
23.2	Configuring automatic operator selection via the web interface	202
23.3	Configuring via UCI	221
23.4	Configuring no PMP + roaming using UCI.....	225
23.5	Automatic operator selection diagnostics via the web interface	227
23.6	Automatic operator selection diagnostics via UCI	229
24	Configuring Connection Watch (cwatch)	231
24.1	Configuration package used	231

24.2	Configuring Connection Watch using the web interface	231
24.3	Configuring cwatch using command line.....	233
24.4	cwatch diagnostics	234
25	Configuring DHCP server and DNS (Dnsmasq)	235
25.1	Configuration package used	235
25.2	Configuring DHCP and DNS using the web interface	235
25.3	Configuring DHCP and DNS using UCI.....	243
25.4	Configuring DHCP pools using UCI.....	245
25.5	Configuring static leases using UCI.....	246
26	Configuring DHCP client.....	247
26.1	Configuration packages used	247
26.2	Configuring DHCP client using the web interface	247
26.3	Configuring DHCP client using command line	253
26.4	DHCP client diagnostics	254
27	Configuring DHCP forwarding	257
27.1	Configuration packages used	257
27.2	Configuring DHCP forwarding using the web interface	257
27.3	Configuring DHCP forwarding using command line	258
27.4	DHCP forwarding over IPSec.....	259
27.5	DHCP forwarding diagnostics	262
28	Configuring Dynamic DNS.....	264
28.1	Overview	264
28.2	Configuration packages used	264
28.3	Configuring Dynamic DNS using the web interface	264
28.4	Dynamic DNS using UCI.....	266
29	Configuring hostnames.....	268
29.1	Overview	268
29.2	Local host file records.....	268
29.3	PTR records.....	270
29.4	Static leases.....	272
30	Configuring firewall	275
30.1	Configuration package used	275
30.2	Configuring firewall using the web interface	275
30.3	Configuring firewall using UCI.....	286
30.4	IPv6 notes	290
30.5	Implications of DROP vs. REJECT	290
30.6	Connection tracking	291
30.7	Firewall examples	291
31	Configuring IPsec.....	299
31.1	Configuration package used	299

31.2	Configuring IPSec using the web interface.....	299
31.3	Configuring IPSec using UCI.....	307
31.4	Configuring an IPSec template for DMVPN via the web interface	311
31.5	Configuring an IPSec template to use with DMVPN	319
31.6	IPSec diagnostics using the web interface	321
31.7	IPSec diagnostics using UCI	321
32	Dynamic Multipoint Virtual Private Network (DMVPN)	322
32.1	Prerequisites for configuring DMVPN.....	322
32.2	Advantages of using DMVPN.....	322
32.3	DMVPN scenarios	323
32.4	Configuration packages used	325
32.5	Configuring DMVPN using the web interface	325
32.6	DMVPN diagnostics.....	327
33	Open VPN	330
33.1	Client configuration	330
34	Configuring multicasting using PIM and IGMP interfaces	333
34.1	Overview	333
34.2	Configuration package used	333
34.3	Configuring PIM and IGMP using the web interface	333
34.4	Configuring PIM and IGMP using UCI	334
35	QoS: VLAN 802.1Q PCP tagging	336
35.1	Configuring VLAN PCP tagging	336
36	QoS: type of service.....	339
36.1	QoS configuration overview	339
36.2	Configuration packages used	339
36.3	Configuring QoS using the web interface.....	339
36.4	Configuring QoS using UCI	341
36.5	Example QoS configurations	344
37	Management configuration settings	345
37.1	Activator.....	345
37.2	Monitor.....	345
37.3	Configuration packages used	345
37.4	Autoload: boot up activation.....	345
37.5	Autoload packages	346
37.6	Autoload using UCI	349
37.7	HTTP Client: configuring activation using the web interface	350
37.8	Httpclient: Activator configuration using UCI	352
37.9	Httpclient: Activator configuration using package options.....	353
37.10	User management using UCI	354
37.11	Configuring the management user password using UCI.....	355

37.12	Configuring management user password using package options.....	355
37.13	User management using UCI	356
37.14	User management using package options	356
37.15	Configuring user access to specific web pages	357
38	Configuring Monitor.....	358
38.1	Introduction	358
38.2	Reporting device status to Monitor	358
38.3	Reporting GPS location to Monitor	364
38.4	Reporting syslog to Monitor.....	366
38.5	Configuring ISAD	368
39	Configuring SNMP	371
39.1	Configuration package used	371
39.2	Configuring SMNP using the web interface.....	371
39.3	Configuring SNMP using command line	377
39.4	Configuring SNMP interface alias with static SNMP index	383
39.5	SNMP diagnostics.....	385
40	Event system	388
40.1	Configuration package used	388
40.2	Implementation of the event system	388
40.3	Supported events.....	388
40.4	Supported targets	389
40.5	Supported connection testers	389
40.6	Configuring the event system using the web interface	389
40.7	Configuring the event system using UCI	389
40.8	Event system diagnostics	401
41	Configuring Terminal Server.....	407
41.1	Overview	407
41.2	Configuration packages used	407
41.3	Configuring Terminal Server using the web interface	407
41.4	Terminal Server using UCI	417
41.5	Terminal Server using package options.....	418
41.6	Terminal Server diagnostics	418

1 Introduction

This user manual describes the features and how to configure SATEL-GW100 and GW120 series routers.

The SATEL-GW100 and GW120 Series routers enable 3G/LTE connectivity for Utility customers, where secure, reliable all-IP networking is required. It is also a perfect telecommunications router in vehicles such as buses, taxis and fleet vehicles for applications such as passenger WiFi internet access, telemetry and employee WiFi access to corporate network services.

Designed for managed network providers, GW100 and GW120 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 models in the GW100 Series and the GW120 Series. For general references, we refer to the GW100 Series throughout. Feature variations between GW100 Series and GW120 Series are described in separate sections.

1.1.1 GW100 Series routers

The SATEL XPRS SATEL-GW100 Series router is a compact 3G/4G LTE router designed with a rugged metal casing with optional carrier for use in vehicles and a wide range of site-based applications.

GW100: Dual Ethernet, 4G/LTE, Dual SIM

1.1.2 GW120 Series routers

The SATEL GW120 Series router is a compact 3G/4G LTE router with WiFi, designed with a rugged metal housing for use in vehicles and a wide range of site-based applications.

GW120: Dual Ethernet, 4G/LTE, Dual SIM, Dual WiFi SMA female connectors

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	Specifies the route metric to use.
UCI: network.@route[0].metric	
Opt: metric	

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 written 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.

Web Field/UCI/Package Option	Description																
Web: Enable UCI: cesop.main.enable Opt: enable	Enables CESoPSN services. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.												
0	Disabled.																
1	Enabled.																
Web: Syslog Severity UCI: cesop.main.severity Opt: log_severity	Selects the severity used for logging events CESoPSN in syslog. The following levels are available. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Emergency</td> </tr> <tr> <td>1</td> <td>Alert</td> </tr> <tr> <td>2</td> <td>Critical</td> </tr> <tr> <td>3</td> <td>Error</td> </tr> <tr> <td>4</td> <td>Warning</td> </tr> <tr> <td>5</td> <td>Notice</td> </tr> <tr> <td>6</td> <td>Informational</td> </tr> <tr> <td>7</td> <td>Debug</td> </tr> </table>	0	Emergency	1	Alert	2	Critical	3	Error	4	Warning	5	Notice	6	Informational	7	Debug
0	Emergency																
1	Alert																
2	Critical																
3	Error																
4	Warning																
5	Notice																
6	Informational																
7	Debug																
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][,...]																

Table 1: Example of an information table

1.2.2 Definitions

Throughout the document, we use the host name ‘GW_router’ to cover all router models.

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

```
root@GW_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 GW100 Series hardware

2.1 Hardware model variants

2.1.1 GW100/120 Series router



Figure 1: GW120 series router front (GW100 without WiFi antenna connector)



Figure 2: GW120 series router back

GW100	Dual Ethernet, 4G/LTE, dual SIM, metal casing, optional carrier
GW120	Dual Ethernet, 4G/LTE, dual SIM, dual WiFi, dual WiFi SMA connectors, metal casing, optional carrier

2.2 GW100 Series hardware features

2.2.1 GW100 Series router

- 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

2.2.2 GW120 Series router

- 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
- Dual WiFi internal antennas
- Dual WiFi SMA female connectors
- Concurrent Access Point and Station mode

2.3 GSM technology

- LTE
- HSPA+
- EDGE/GPRS
- GPS

2.4 WiFi technology for GW120

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

2.5 Power supply

The GW100 and GW120 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 GW100/GW120 Series router dimensions

Unit size:	114W 114D 38Hmm
Unit size with carrier:	120W 120D 42Hmm
Unit weight:	450g

2.7 Compliance

The GW100 and GW120 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. Ask for availability of separate bands.

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
RFG	850/900/1800/1900	850/900/2100	B1/B3/B5/B7/B20	40°C to 70°C
RFH	-	850/1900	B2/B4/B5/B17	30°C to 70°C
RFJ	450			40°C to 70°C

2.9 Antenna

The GW100 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. The GW120 has two additional SMA female WiFi antenna sockets.

2.9.1 GW100 Series router

- 2 x 4G/LTE SMA female antenna connectors
- MIMO support in LTE versions
- 1 x GPS SMA female antenna connector with 3v3 active power feed

2.9.2 GW120 Series router

- 2 x 4G/LTE SMA female antenna connectors
- MIMO support in LTE versions
- 1 x GPS SMA female antenna connector with 3v3 active power feed

- 2 x SMA female WiFi antenna sockets

2.10 Components

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

The routers contain 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.

2.10.1 Standard components

1 x GW100/GW120 Series router	
1 x plastic carrier	

Table 2: GW100 Series router standard components

2.10.2 Optional components

Ethernet cable. RJ45 connector at both ends.	
Power supply unit.	
Right angle antenna for 3G/4G network.	 SATEL supplies a wide range of antennas. Please visit our website: www.SATEL.com or contact SATEL for more information.
1 x fused automotive cable	
1 x non-fused automotive cable	

Table 3: GW100 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. For information on connecting cables for a vehicle installation, read chapter 4, 'Installing a router into a vehicle'.

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 router takes approximately 2 minutes to boot up. During this time, the PWR/CONFIG LED flashes in a double flash pattern – 2 quick flashes 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 'GW100 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
0-3 seconds	On	Normal reset to running config. No special LED activity.
Between 3 and 15 seconds	Flashing slowly	Releasing between 3-15 seconds switches the router back to factory configuration.
Between 15 and 20 seconds	On	Releasing between 15-20 seconds performs a normal reset to running config.
Between 20 seconds and 30 seconds	Flashing faster	Releasing between 20-30 seconds reboots the router in recovery mode.
Over 30 seconds	On	Releasing after 30 seconds performs a normal reset.

Table 4: GW100 series router reset behaviour

2.16.1 Recovery mode

Recovery mode is a fail-safe mode where the router can load a default configuration from the routers firmware. If your router goes into recovery mode, all config files are kept intact. After the next reboot, the router will revert to the previous config file.

You can use recovery mode to manipulate the config files, but should only be used if all other config files are corrupt. If your router has entered recovery mode, contact your local reseller for access information.

3 GW100 and GW120 Series LED behaviour

3.1 Main LED behaviour

There are five LEDs on the GW100 and GW120 Series router



Figure 3: LEDs on the GW120 Series router (GW100 without WIFI indication LED)

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 on the GW100 and GW120 Series router.

Booting		The router 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 LEDs 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 LEDs on	Connected and signal strength >-69dBm.
WiFi LEDs	Off	WiFi not enabled.
	Flashing	Data activity on WiFi interface.
	On	WiFi is enabled.

Table 5: LED behaviour and descriptions

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

3.2 GW100 and GW120 Series 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 4: Ethernet LED

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

Table 6: Ethernet LED activity description

4 Factory configuration extraction from SIM card

SATEL 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.

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.

Note: factory configuration extraction is only supported on mobile modules that support phone book operations.

5 Accessing the router

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

5.1 Configuration packages used

Package	Sections
dropbear	dropbear
system	main
uhttpd	main cert

5.2 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 that 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.

The screenshot shows a login form titled "Authorization Required". The text "Please enter your username and password." is displayed above the input fields. There are two input fields: "Username" containing "root" and "Password" containing "*****". Below the password field is a small blue asterisk icon. At the bottom are two buttons: "Login" and "Reset".

Figure 5: 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.

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



Figure 6: Confirming trust of the routers public key over SSH

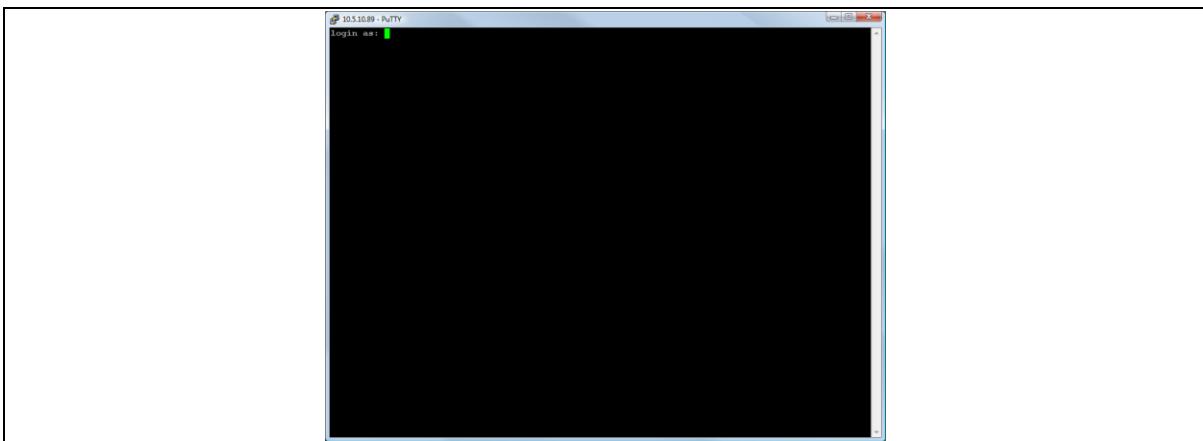


Figure 7: SSH CLI logon screen

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

Username: **root**

Password: **admin**

5.3.1 SCP (Secure Copy Protocol)

As part of accessing the router over SSH, you can also use SCP protocol. Use the same user authentication credentials as for SSH access. You can use SCP protocol to securely, manually transfer files from and to the router's SCP server.

No dedicated SPC client is supported; select the SCP client software of your own choice.

5.4 Accessing the router over Ethernet using a Telnet client

Telnet is disabled by default, when you enable Telnet, SSH is disabled.

To enable Telnet, enter:

```
root@GW_router: ~# /etc/init.d/dropbear disable
root@GW_router: ~# reboot -f
```

To re-enable SSH, enter:

```
root@GW_router: ~# /etc/init.d/dropbear enable
root@GW_router: ~# reboot -f
```

Note: As SSH is enabled by default, initial connection to the router to enable Telnet must be established over SSH.

5.5 Configuring the password

5.5.1 Configuration packages used

Package	Sections
system	main

5.6 Configuring the password using the web interface

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

Figure 8: 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	Defines the root password. The password is displayed encrypted via the CLI using the 'hashpassword' option.
UCI: system.main.password	UCI: system.main.hashpassword
Opt: password	Opt: hashpassword

5.7 Configuring the password using UCI

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

```
root@GW_router:~# uci show system
system.main=system
system.main.hostname=GW_router
system.main.hashpassword=$1$jRX/x8A/$U5kLCMp19dcahRh017eZV1
```

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

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

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

5.8 Configuring the password using package options

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

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

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

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

```
package system

config system 'main'
    option hostname 'GW_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.

5.9 Accessing the device using RADIUS authentication

You can configure RADIUS authentication to access the router over SSH, web or local console interface.

```
package system

config system 'main'
    option hostname 'SATEL'
    option timezone 'UTC'

config pam_auth
    option enabled 'yes'
    option pamservice 'login'
    option pammodule 'auth'
    option pamcontrol 'sufficient'
    option type 'radius'
    option servers '192.168.0.1:3333|test|20 192.168.2.5|secret|10'

config pam_auth
    option enabled 'yes'
    option pamservice 'sshd'
    option pammodule 'auth'
    option pamcontrol 'sufficient'           it checks package
management_users
    option type 'radius'
    option servers '192.168.0.1:3333|test|20 192.168.2.5|secret|10'

config 'pam_auth'
    option enabled 'yes'
    option pamservice 'luci"
    option pammodule 'auth'
    option pamcontrol 'sufficient'
    option type 'radius'
    servers '192.168.0.1:3333|test|20 192.168.2.5|secret|10'
```

UCI/Package Option	Description	
UCI: system.@pam_auth[0].enabled=yes Opt: enabled	Enables and disables RADIUS configuration sections.	
	yes	Enables following RADIUS configuration section.
	no	Disables following RADIUS configuration section.
UCI: system.@pam_auth[0].pamservice Opt: pamservice	Selects the method which users should be authenticated by.	
	login	User connecting over console cable.
	sshd	User connecting over SSH.
	luci	User connecting over web.
UCI: system.@pam_auth[0].pamcontrol Opt: pamcontrol	Specifies authentication behaviour after authentication fails or connection to RADIUS server is broken.	
	Sufficient	First authenticates against remote RADIUS if password authentication fails then it tries local database (user defined in package management_users)
	Required	If either authentication fails or RADIUS server is not reachable then user is not allowed to access the router.
	[success=done new_authtok_reqd=done authinfo_unavail=ignore default=die]	Local database is only checked if RADIUS server is not reachable.
UCI: system.@pam_auth[0].pammodule.auth Opt: pammodule	Enables user authentication.	
UCI: system.@pam_auth[0].type.radius Opt: type	Specifies the authentication method.	
UCI: system.@pam_auth[0].servers Opt: servers	Specifies the RADIUS server or multiple servers along with port number and password. The example below explains the syntax. 192.168.0.1:3333 test 20 192.168.2.5 secret 10	

Table 7: Information table for RADIUS authentication

5.10 Accessing the device using TACACS+ authentication

TACACS+ authentication can be configured for accessing the router over SSH, web or local console interface.

```
package system

config system 'main'
    option hostname 'SATEL'
    option timezone 'UTC'

config pam_auth
    option enabled 'yes'
    option pamservice 'sshd'
```

```
option pammodule 'auth'
option pamcontrol 'sufficient'
option type 'tacplus'
option servers '192.168.0.1:49|secret'

config pam_auth
    option enabled 'yes'
    option pamservice 'sshd'
    option pammodule 'account'
    option pamcontrol 'sufficient'
    option type 'tacplus'
    option servers '192.168.0.1:49|secret'
    option args 'service=ppp'

config pam_auth
    option enabled 'yes'
    option pamservice 'sshd'
    option pammodule 'session'
    option pamcontrol 'sufficient'
    option type 'tacplus'
    option servers '192.168.0.1:49|secret'
    option args 'service=ppp'

config pam_auth
    option enabled 'yes'
    option pamservice 'luci'
    option pammodule 'auth'
    option pamcontrol 'sufficient'
    option type 'tacplus'
    option servers '192.168.0.1:49|secret'

config pam_auth
    option enabled 'yes'
    option pamservice 'luci'
    option pammodule 'account'
    option pamcontrol 'sufficient'
    option type 'tacplus'
```

```
option servers '192.168.0.1:49|secret'
option args 'service=ppp'

config pam_auth
    option enabled 'yes'
    option pamservice 'luci'
    option pammodule 'session'
    option pamcontrol 'sufficient'
    option type 'tacplus'
    option servers '192.168.0.1:49|secret'
    option args 'service=ppp'

config pam_auth
    option enabled 'yes'
    option pamservice 'login'
    option pammodule 'auth'
    option pamcontrol 'sufficient'
    option type 'tacplus'
    option servers '192.168.0.1:49|secret'

config pam_auth
    option enabled 'yes'
    option pamservice 'login'
    option pammodule 'account'
    option pamcontrol 'sufficient'
    option type 'tacplus'
    option servers '192.168.0.1:49|secret'
    option args 'service=ppp'

config pam_auth
    option enabled 'yes'
    option pamservice 'login'
    option pammodule 'session'
    option pamcontrol 'sufficient'
    option type 'tacplus'
    option servers '192.168.0.1:49|secret'
    option args 'service=ppp'
```

UCI/Package Option	Description	
UCI: system.@pam_auth[0].enabled=yes Opt: enabled	Enables and disables TACACS configuration sections.	
	yes	Enables following TACACS configuration section.
	no	Disables following TACACS configuration section.
UCI: system.@pam_auth[0].pamservice Opt: pamservice	Selects the method which users should be authenticated by.	
	login	User connecting over console cable.
	sshd	User connecting over SSH.
	luci	User connecting over web.
UCI: system.@pam_auth[0].pamcontrol Opt: pamcontrol	Specifies authentication behaviour after authentication fails or connection to TACACS server is broken.	
	Sufficient	First authenticates against remote TACACS if password authentication fails then it tries local database (user defined in package management_users)
	Required	If either authentication fails or TACACS server is not reachable then user is not allowed to access the router.
	[success=done new_authtok_reqd=done authinfo_unavail=ignore default=die]	Local database is only checked if TACACS server is not reachable.
UCI: system.@pam_auth[0].pammodule.auth Opt: pammodule	Selects which TACACS module this part of configuration relates to.	
	auth	auth module provides the actual authentication and sets credentials
	account	account module checks to make sure that access is allowed for the user
	session	session module performs additional tasks which are needed to allow access
system.@pam_auth[0].type=tacplus Opt: type	Specifies the authentication method.	
UCI: system.@pam_auth[0].servers Opt: servers	Specifies the TACACS servers along with port number and password. The example below explains the syntax. 192.168.0.1:49 secret '	
UCI: system.@pam_auth[1].args=service=ppp Opt: args	Additional arguments to pass to TACACS server.	

Table7: Information table for TACACS authentication

5.11 SSH

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 on: /etc/config_name/dropbear.

5.11.1 Configuration packages used

Package	Sections
dropbear	dropbear

5.11.2 SSH access using the web interface

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

SSH Access
Dropbear offers SSH network shell access and an integrated SCP server

Dropbear Instance

Interface: 3G:
 ADSL:
 lan:
 lan1:
 loopback:
 unspecified
(?) Listen only on the given interface or, if unspecified, on all

Port: (?) Specifies the listening port of this Dropbear instance

Password authentication: (?) Allow SSH password authentication

Allow root logins with password: (?) Allow the root user to login with password

Gateway ports: (?) Allow remote hosts to connect to local SSH forwarded ports

Idle Session Timeout (seconds): (?) Remote session will be closed after this many seconds of inactivity

Figure 9: The SSH access section

Web Field/UCI/Package Option	Description	
Basic settings		
Web: Interface UCI: dropbear.@dropbear[0].Interface Opt: interface	Listens only on the selected interface. If unspecified is checked, listens on all interfaces. All configured interfaces will be displayed via the web GUI.	
	(unspecified)	listens on all interfaces.
Web: Port UCI: dropbear.@dropbear[0].Port Opt: port	Range	Configured interface names.
	Specifies the listening port of the Dropbear instance.	
Web: Password authentication UCI: dropbear.@dropbear[0].PasswordAuth Opt: PasswordAuth	22	
	Range	0-65535
Web: Allow root logins with password UCI: dropbear.@dropbear[0].RootPasswordAuth Opt: RootPasswordAuth	If enabled, allows SSH password authentication.	
	0	Disabled.
Web: Gateway ports UCI: dropbear.@dropbear[0].GatewayPorts Opt: GatewayPorts	1	Enabled.
	Allows remote hosts to connect to local SSH forwarded ports.	
Web: Idle Session Timeout UCI: dropbear.@dropbear[0].IdleTimeout Opt: IdleTimeout	0	Disabled.
	1	Enabled.
Web: n/a UCI: dropbear.@dropbear[0].BannerFile Opt: BannerFile	Defines the idle period where remote session will be closed after the allocated number of seconds of inactivity.	
	30	30 seconds.
Web: n/a UCI: dropbear.@dropbear[0].MaxLoginAttempts Opt: MaxLoginAttempts	Range	
	Defines a banner file to be displayed during login.	
Web: n/a UCI: dropbear.@dropbear[0].MaxLoginAttempts Opt: MaxLoginAttempts	/etc/banner	
	Range	
Web: n/a UCI: dropbear.@dropbear[0].MaxLoginAttempts Opt: MaxLoginAttempts	Specifies maximum login failures before session terminates.	
	10	
	0-infinite	

Table 8: Information table for SSH access settings

5.12 Package dropbear using UCI

```
root@GW_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
dropbear.@dropbear[0].MaxLoginAttempts=3
Package dropbear using package options
```

```
root@GW_router:~# uci export dropbear
package dropbear
config dropbear'
    option PasswordAuth 'on'
    option RootPasswordAuth 'on'
    option Port '22'
    option GatewayPorts '0'
    option IdleTimeout '30'
    option MaxLoginAttempts '3'
```

5.13 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.

Certificates & Private Keys

Certificates and private keys used for various services could be uploaded here

IPsec Certificates and Keys

Choose file No file chosen

Upload a *.tar.gz file containing certificates and/or private keys. All the ipsec certs previously uploaded will be deleted when new ones uploaded. Archive structure should match this of /etc/ipsec.d folder. Every file should be in one of 8 subfolders according to its purpose:

- private (private keys)
- certs (entity certs)
- crls (revocation lists)
- cacerts (CA certs)
- ocspcerts (OCSP signer certs)
- aacerts (Authorization Authority certs)
- acerts (attribute certs)
- reqs (PKCS#10 cert requests)

[More info](#)

OpenVPN Certificates and Keys

Choose file No file chosen

Upload a *.tar.gz file containing certificates and/or private keys. All the openvpn certs previously uploaded will be deleted when new ones uploaded. OpenVPN requires no special folder structure, hence files will be installed into the openvpn folder as they are in archive

VA Certificates and Keys

Choose file No file chosen

Upload a *.tar.gz file containing certificates and/or private keys. All the va certs previously uploaded will be deleted when new ones uploaded. Archive structure should match this of /etc/certs folder which is similar to /etc/ipsec.d folder.

Save & Apply **Save** **Reset**

Figure 10: 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 custom 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.

SSH-Keys
Here you can paste public SSH-Keys (one per line) for SSH public-key authentication.

Figure 11: The SSH-keys box

5.14 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.

5.14.1 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.

5.14.2 Main settings

The screenshot shows the 'HTTP Server' configuration page. At the top, there are navigation links: Status, System, Services, Network, and Logout. Below the header, the title 'HTTP Server' is displayed, followed by a subtitle 'Configuration of the Http Server used for management of the device.' Under the heading 'Main Settings', there are several configuration fields:

- Listen Address and Port:** 0.0.0.0:80. A note explains: '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. Use 0.0.0.0:80 to bind at port 80 only on IPv4 interfaces or [::]:80 to serve only IPv6'.
- Secure Listen Address and Port:** 0.0.0.0:443. A note explains: 'Specifies the ports and addresses to listen on for encrypted HTTPS access'.
- Home path:** /www. A note explains: 'Defines the server document root'.
- Cert file:** /etc/uhttpd.crt. A note explains: 'PEM certificate used to serve HTTPS connections'.
- Key file:** /etc/uhttpd.key. A note explains: 'PEM private key used to serve HTTPS connections'.
- CGI prefix:** /cgi-bin. A note explains: 'Defines the prefix for CGI scripts; relative to the document root. CGI support is disabled if this option is missing'.
- Script timeout (s):** 60. A note explains: 'Maximum wait time for CGI or Lua requests in seconds. Requested executables are terminated if no output was generated until the timeout expired'.
- Network timeout (s):** 30. A note explains: 'Maximum wait time for network activity. Requested executables are terminated and connection is shut down if no network activity occurred for the specified number of seconds'.
- rfc1918 filter:**

Figure 12: HTTP server settings

Web Field/UCI/Package Option	Description						
Web: Listen Address and Port UCI: uhttpd.main.listen_http Opt: list listen_http	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 border="1"> <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>	0.0.0.0:80	Bind at port 80 only on IPv4 interfaces.	[::]:80	Bind at port 80 only on IPv6 interfaces	Range	IP address and/or port
0.0.0.0:80	Bind at port 80 only on IPv4 interfaces.						
[::]:80	Bind at port 80 only on IPv6 interfaces						
Range	IP address and/or port						
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. <table border="1"> <tr> <td>0.0.0.0:443</td><td>Bind at port 443 only</td></tr> <tr> <td>[::]:443</td><td></td></tr> <tr> <td>Range</td><td>IP address and/or port</td></tr> </table>	0.0.0.0:443	Bind at port 443 only	[::]:443		Range	IP address and/or port
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. <table border="1"> <tr> <td>/www</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	/www		Range			
/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. <table border="1"> <tr> <td>/etc/uhttpd.crt</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	/etc/uhttpd.crt		Range			
/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. <table border="1"> <tr> <td>/etc/uhttpd.key</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	/etc/uhttpd.key		Range			
/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. <table border="1"> <tr><td>/cgi-bin</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	/cgi-bin		Range	
/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. <table border="1"> <tr><td>/luci</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	/luci		Range	
/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. <table border="1"> <tr><td>/usr/lib/lua/luci/cgi/uhttpd.lua</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	/usr/lib/lua/luci/cgi/uhttpd.lua		Range	
/usr/lib/lua/luci/cgi/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. <table border="1"> <tr><td>60</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	60		Range	
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 occurred for the specified number of seconds. <table border="1"> <tr><td>30</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	30		Range	
30					
Range					
Web: N/A UCI: uhttpd.main.realm Opt: realm	Defines basic authentication realm when prompting the client for credentials (HTTP 400). <table border="1"> <tr><td>OpenWrt</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	OpenWrt		Range	
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. <table border="1"> <tr><td>/etc/http.conf</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	/etc/http.conf		Range	
/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. <table border="1"> <tr><td></td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>			Range	
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). <table border="1"> <tr><td></td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>			Range	
Range					
Web: N/A UCI: uhttpd.main.no_symlinks Opt: no_symlinks	Does not follow symbolic links if enabled. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: N/A UCI: uhttpd.main.no_dirlists Opt: no_symlinks	Does not generate directory listings if enabled. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	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).				
	<table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

Table 9: Information table for http server basic settings

5.14.3 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@GW_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
HTTP server using package options
root@GW_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'
```

5.14.4 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.

The screenshot shows a 'Certificate Settings' configuration page. It includes fields for Days (3650), Bits (1024), country (IE), state (Dublin), location (Dublin), and commonname (GW). At the bottom are 'Save & Apply', 'Save', and 'Reset' buttons.

Figure 13: 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.
Web: State UCI: uhttpd.px5g.state Opt: state	State of the certificate issuer.
Web: Location UCI: uhttpd.px5g.location Opt: location	Location or city of the certificate user.
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

5.14.5 HTTPS server using UCI

```
root@GW_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
HTTPS server using package options
root@GW_router:~# uci export uhttpd
package uhttpdconfig 'cert' 'px5g'
    option 'days' '3650'
    option 'bits' '1024'
    option 'state' 'Dublin'

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

5.15 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.

5.16 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
```

5.17 Displaying custom information via login screen

The login screen, by default, shows the hostname of the router in addition to the username and password prompt. However, the router can be configured to show some other basic information if required using a UDS script.

Note: this can only be configured via the command line.

5.17.1 Configuration packages used

Package	Sections
luci	main
uds	script

5.17.2 Configuring login screen custom information

The luci package option `login_page_info_template` is configured with the path to a UDS script that would render the required information on the right side of the login page.

The following example shows how to display serial number and mobile signal strength.

Note: this can only be configured via the command line.



Figure 14: Example login screen displaying serial and signal strength

5.17.2.1 Login screen custom information using UCI

```
root@GW_router:~# uci show luci
luci.main=core
luci.main.login_page_info_template=/tmp/uds/sysauth_template

root@GW_router:~# uci show uds
uds.sysauth_template=script
uds.sysauth_template.enabled=1
uds.sysauth_template.exec_type=none
uds.sysauth_template.fname=sysauth_template.htm
uds.sysauth_template.type=none
uds.sysauth_template.text=Serial: <%=pcdata(luci.version.serial)%><br/> <%
local sig = luci.dispatcher.uci.cursor_state():get("mobile", "3g_1_1",
"sig_dbm") or -113 sig = tonumber(sig) local hue = (sig + 113) * 2 local
hue = math.min(math.max(hue, 0), 120) %> Signal strength: <h3
style="color:hsl(<%=hue%>, 90%, 50%); display:inline;"><%=sig%></h3> dBm
```

5.17.2.2 Login screen custom information using package options

```
root@GW_router:~# uci export luci
package luci
config core 'main'
    option login_page_info_template '/tmp/uds/sysauth_template'
root@GW_router:~# uci export uds
package uds
config script 'sysauth_template'
    option enabled '1'
    option exec_type 'none'
    option fname 'sysauth_template.htm'
    option type 'none'
    list text 'Serial: <%=pcdata(luci.version.serial)%><br/>'
    list text '<% local sig =
luci.dispatcher.uci.cursor_state():get("mobile", "3g_1_1", "sig_dbm") or -
113'
    list text 'sig = tonumber(sig)'
    list text 'local hue = (sig + 113) * 2'
    list text 'local hue = math.min(math.max(hue, 0), 120) %> '
    list text 'Signal strength: <h3 style="color:hsl(<%=hue%>, 90%, 50%); display:inline;"><%=sig%></h3> dBm'
```

6 Router file structure

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

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

When showing examples of the command line interface we use the host name 'GW_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@GW_router:~# va_config.sh
```

6.1 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.

Status	
System	
Router Name	GW0000
Router Model	GW0031W-AA0179E
Firmware Version	VIE-16.00.55
Current Image/Config	image2 / config2
Kernel Version	3.2.12
Local Time	Fri Aug 5 11:43:52 2016
Uptime	0h 10m 8s
Load Average	0.27, 0.35, 0.31

Figure 15: The status page

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

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

The example below shows the output from the above command.

VA_SERIAL:	00E0C8121215
VA_MODEL:	GW0000
VA_ACTIVEIMAGE:	image2
VA_ACTIVECONFIG:	config1

VA_IMAGE1VER:	VIE-16.00.44
VA_IMAGE2VER:	VIE-16.00.44

6.2 Identify your software version

To check which software version your router is running, in the top menu, browse to **Status -> Overview**.

Status	
System	
Router Name	GW0000
Router Model	GW0031W-AA0179E
Firmware Version	VIE-16.00.55
Current Image/Config	image2 / config2
Kernel Version	3.2.12
Local Time	Fri Aug 5 11:43:52 2016
Uptime	0h 10m 8s
Load Average	0.27, 0.35, 0.31

Figure 16: The status page showing a software version prior to 72.002

Status	
System	
Router Name	dmvpn
Router Model	GW
Firmware Version	LIS-15.00.72.002rc4
Current Image/Config	image1 / config1
Kernel Version	3.2.12
Local Time	Thu Jan 26 14:46:03 2017
Uptime	0h 39m 37s
Load Average	1.02, 0.53, 0.48

Figure 17: The status page showing software version 72.002

In the Firmware Version row, the first two digits of the firmware version identify the hardware platform, for example LIS-15; while the remaining digits: .00.72.002, show the software version.

6.3 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".

6.4 Directory locations for UCI configuration files

Router configurations files are stored in folders on:

- /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.

6.5 Viewing and changing current configuration

To show the configuration currently running, enter:

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

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

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

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

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

6.6 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

6.7 Managing configurations

6.7.1 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@GW_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@GW_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@GW_router:/etc/config1# cp /etc/config2/* /etc/config1
```

6.8 Exporting a configuration file

If you have software versions prior to 72.002, to export a configuration file using the web interface, go to section 7.8.1.

If you have software version 72.002 or above, export a configuration file using the web interface go to section 7.8.2.

To export a configuration file using CLI, for any software version, go to section 7.8.3.

6.8.1 Exporting a configuration file using the web interface for software versions pre- 72.002

The current running configuration file may be exported using the web interface.

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

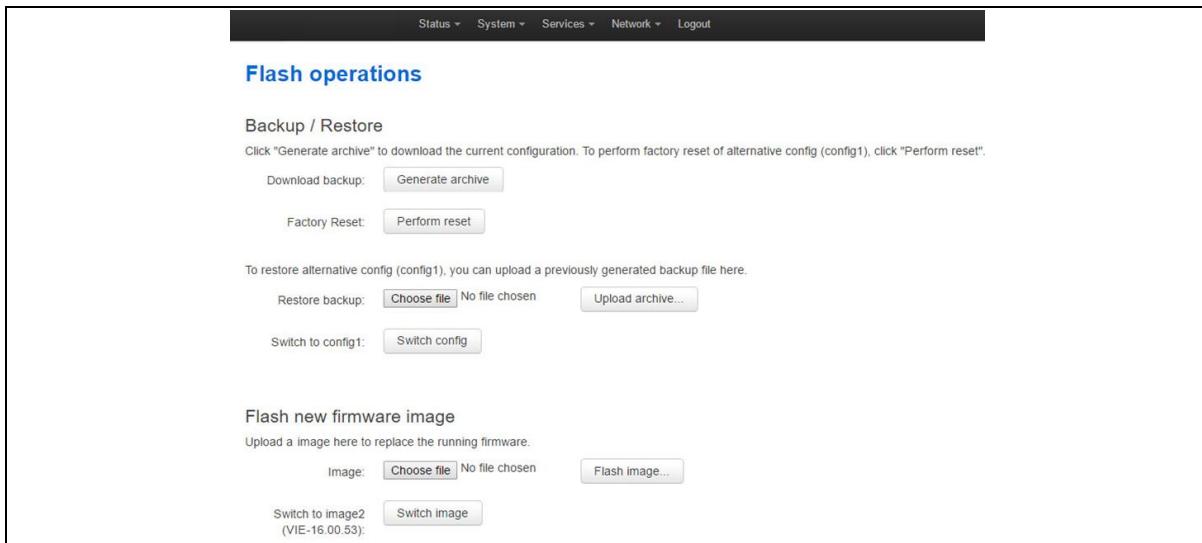


Figure 18: The flash operations page

In the Backup/Restore section, select **Generate Archive**.

6.8.2 Exporting a configuration file using the web interface for software version 72.002 and above

The current running configuration file may be exported using the web interface.

In the top menu, select **System > Flash Operations**. The Flash operations page appears.

Flash Operations			
Contents	Current Operational Status	After Reboot	Operations
Image 1 LIS-15.00.72.002rc1	active	will be active	<input type="button" value="Make active (after reboot)"/> <input type="button" value="Flash image"/>
Image 2 LIS-15.00.72.002rc1		<input type="button" value="Make active (after reboot)"/>	<input type="button" value="Flash image"/>
Config 1 Configuration (19013 bytes)	active	will be active	<input type="button" value="Make active (after reboot)"/>
Config 2 Configuration (19037 bytes)		<input type="button" value="Make active (after reboot)"/>	<input type="button" value="Upload new..."/>
Factory Config Configuration (12203 bytes)		<input type="button" value="Make active (after reboot)"/>	

Reboot using Active Configuration
Reboot the device. The image and config that will be used are shown in green above.

Factory Reset
Here you can reset the router to factory configuration.
On reboot, the factory defaults will be running and you will be able to make changes to the configuration.
A choice of config 1 or config 2 is given in case you have a preference for which config to use (and which to preserve). If you have no preference then either can be used.

Figure 19: The flash operations page

In the **Flash Operation** section, click the configuration file in the Contents column to download it.

6.8.3 Exporting a configuration file using UCI

You can view any configuration file segment using UCI.

To export the running configuration file, enter:

```
root@GW_router:~# uci export
```

To export the factory configuration file, enter:

```
root@GW_router:~# uci -c /etc/factconf/ export
```

To export config1 or config2 configuration file, enter:

```
root@GW_router:~# uci -c /etc/config1/ export
root@GW_router:~# uci -c /etc/config2/ export
```

6.9 Importing a configuration file

If you have software versions prior to 72.002, to export a configuration file using the web interface, go to section 7.9.1

If you have software version 72.002 or above, export a configuration file using the web interface go to section 7.9.2

To export a configuration file using CLI, for any software version, go to section 7.9.3

6.9.1 Importing a configuration file using the web interface for software versions pre- 72.002

You can import a configuration file to the alternate configuration segment using the web interface. This will automatically reboot the router into this configuration file.

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

The screenshot shows the 'Flash operations' page with a dark header bar containing 'Status', 'System', 'Services', 'Network', and 'Logout' links. Below the header, the main title 'Flash operations' is displayed in blue. The 'Backup / Restore' section contains two sets of buttons: 'Download backup:' and 'Generate archive' (disabled), and 'Factory Reset:' and 'Perform reset' (disabled). A note below says 'To restore alternative config (config1), you can upload a previously generated backup file here.' It includes fields for 'Restore backup:' (with 'Choose file' and 'No file chosen' buttons) and 'Upload archive...' (disabled). Another set of buttons for 'Switch to config1:' and 'Switch config' (disabled) is also present. Below this, the 'Flash new firmware image' section has a note 'Upload a image here to replace the running firmware.', a 'Image:' field with 'Choose file' and 'No file chosen' buttons, and 'Flash image...' (disabled). A final set of buttons for 'Switch to image2 (VIE-16.00.53):' and 'Switch image' (disabled) is shown.

Figure 20: The flash operations page

Under Backup/Restore, choose **Restore Backup: Choose file**. Select the appropriate file and then click **Upload archive**.

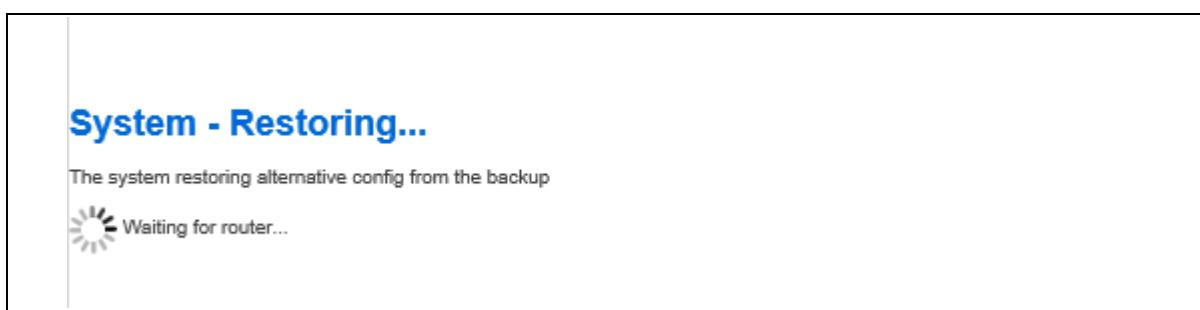


Figure 21: The system – restoring...page

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

6.9.2 Importing a configuration file using the web interface for software version 72.002 and above

You can import a configuration file to the alternate configuration segment using the web interface.

In the top menu, select **System > Flash Operations**. The Flash operations page appears.

The screenshot shows the 'Flash Operations' page with the following details:

Contents	Current Operational Status	After Reboot	Operations
Image 1 LIS-15.00.72.002rc1	active	will be active	<input type="button" value="Flash image..."/>
Image 2 LIS-15.00.72.002rc1		<input type="button" value="Make active (after reboot)"/>	<input type="button" value="Upload new..."/>
Config 1 Configuration (19013 bytes)	active	will be active	
Config 2 Configuration (19037 bytes)		<input type="button" value="Make active (after reboot)"/>	<input type="button" value="Upload new..."/>
Factory Config Configuration (12203 bytes)		<input type="button" value="Make active (after reboot)"/>	

Below the table, there is a section titled 'Reboot using Active Configuration' with a note: 'Reboot the device. The image and config that will be used are shown in green above.' It contains a 'Reboot Now' button.

At the bottom, there is a 'Factory Reset' section with notes about resetting to factory configuration and two buttons: 'Factory Reset Config 1' and 'Factory Reset Config 2'.

Figure 22: The flash operations page

In the Operations column, click **Upload new**. Select the appropriate file.

The screenshot shows the 'Flash Operations' page with a green header bar indicating 'Imported uploaded file to config2'. The table and other sections are identical to Figure 22, except for the green header bar.

Figure 23: The flash operations succeed upload configuration page

If you select 'Flash image and do not reboot', the router will only run this configuration if you click **OK** to return to the Flash Operations page. There you can manually select **Made Active (after reboot)**. Then click **Reboot Now** in the 'Reboot using Active Configuration' section.

6.9.3 Importing a configuration file using UCI

You can import a configuration file to any file segment using UCI.

To import to config1, enter:

```
root@GW_router:~# uci -c /etc/config1/ import  
<paste in config file>  
<CTRL-D>
```

Note: it is very important that the config file is in the correct format otherwise it will not import correctly.

7 Using the Command Line Interface

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

7.1 Overview of some common commands

SATEL routers' 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@GW_router:/# uci set system.main.password="*****"
root@GW_router:/# uci commit system
```

To reboot the system, enter:

```
root@GW_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@GW_router:/# logread

root@GW_router:/# logread | tail

root@GW_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, you can also view and edit these files using some of the standard Unix tools.

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

```
root@GW_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@GW_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@GW_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@GW_router:/# cd /etc/config1
root@GW_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@GW_router:/# crontab -l

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

To view currently running processes, enter:

```
root@GW_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 GW_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@GW_router:/# pgrep -fl 'wifi'
424 root      320 S    wifi up
```

To kill a process, enter the PID:

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

7.2 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@GW_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 11: 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@GW_router:~# uci commit
```

7.2.1 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@GW_router:~# uci commit uhttpd
```

Then enter:

```
root@GW_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.

7.2.2 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@GW_router:~# uci export httpd

package 'httpd'
config 'httpd'
option 'port' '80'
option 'home' '/www'
```

7.2.3 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:

```
root@GW_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@GW_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
```

7.2.4 Display just the value of an option

To display a specific value of an individual option within a package, enter:

```
root@GW_router:~# uci get httpd.@httpd[0].port
80
root@GW_router:~#
```

7.2.5 High level image commands

To show the image running currently, enter:

```
root@GW_router:~# vacmd show current image
```

To set the image to run on next reboot, enter:

```
root@GW_router:~# vacmd set next image [image1|image2|altimage]
```

```
root@GW_router:~# reboot
```

7.2.6 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@GW_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=etherne
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

```

7.3 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 SATEL 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

7.4 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
```

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.

8 Upgrading router firmware

This chapter describes how to upgrade router firmware. The upgrade process is as follows:

- Firmware is transferred to the device.
- Firmware is checked to ensure there are no corruptions.
- Firmware is saved to persistent storage.
- Data in persistent storage is validated.

To avoid any unrecoverable errors during the process, you must follow several safety steps described in this chapter.

On successful completion of the process, you can restart the device running the new firmware.

8.1 Software versions

If you have software versions prior to 72.002, to upgrade firmware using the web interface, go to section 9.1.2.

If you have software version 72.002 or above, to upgrade firmware using the web interface go to section 9.1.3.

To upgrade firmware using CLI, for any software version, go to section 9.1.4.

8.1.1 Identify your software version

To check which software version your router is running, in the top menu, browse to **Status -> Overview**.

Status	
System	
Router Name	GW0000
Router Model	GW0031W-AA0179E
Firmware Version	VIE-16.00.55
Current Image/Config	image2 / config2
Kernel Version	3.2.12
Local Time	Fri Aug 5 11:43:52 2016
Uptime	0h 10m 8s
Load Average	0.27, 0.35, 0.31

Figure 24: The status page showing a software version prior to 72.002

Status	
System	
Router Name	dmvpn
Router Model	GW
Firmware Version	LIS-15.00.72.002rc4
Current Image/Config	image1 / config1
Kernel Version	3.2.12
Local Time	Thu Jan 26 14:46:03 2017
Uptime	0h 39m 37s
Load Average	1.02, 0.53, 0.48

Figure 25: The status page showing software version 72.002

In the Firmware Version row, the first two digits of the firmware version identify the hardware platform, for example LIS-15; while the remaining digits: .00.72.002, show the software version.

8.1.2 Upgrading router firmware for software versions pre- 72.002

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

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

Figure 26: 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.

Flash Firmware - Verify

The flash image was uploaded. Below is the checksum and file size listed, compare them with the original file to ensure data integrity.
Click "Proceed" below to start the flash procedure.

- Checksum: 4f5aa18ebb3ec575ce16dcc9e18273af
- Size: 7.63 MB (14.00 MB available)

Proceed

Figure 27: The flash firmware - verify page

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

System - Flashing...

The system is flashing now.
DO NOT POWER OFF THE DEVICE!
Wait a few minutes until you try to reconnect. It might be necessary to renew the address of your computer to reach the device again, depending on your settings.

Waiting for router...

Figure 28: 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.

Status

System

Router Name	GW0000
Router Model	GW0031W-AA0179E
Firmware Version	VIE-16.00.55
Current Image/Config	image2 / config2
Kernel Version	3.2.12
Local Time	Fri Aug 5 11:43:52 2016
Uptime	0h 10m 8s
Load Average	0.27, 0.35, 0.31

Figure 29: The system status list

8.1.3 Upgrading router firmware for software version 72.002 and above

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

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

The screenshot shows the 'Flash Operations' page with the following details:

Contents	Current Operational Status	After Reboot	Operations
Image 1 LIS-15.00.72.002rc1	active	will be active	
Image 2 LIS-15.00.72.002rc1		<input type="button" value="Make active (after reboot)"/>	<input type="button" value="Flash image..."/>
Config 1 Configuration (19013 bytes)	active	will be active	
Config 2 Configuration (19037 bytes)		<input type="button" value="Make active (after reboot)"/>	<input type="button" value="Upload new..."/>
Factory Config Configuration (12203 bytes)		<input type="button" value="Make active (after reboot)"/>	

Below the table:

- Reboot using Active Configuration**: Reboot the device. The image and config that will be used are shown in green above.
- Factory Reset**: Here you can reset the router to factory configuration.
On reboot, the factory defaults will be running and you will be able to make changes to the configuration.
A choice of config 1 or config 2 is given in case you have a preference for which config to use (and which to preserve). If you have no preference then either can be used.

Figure 30: The flash operations page

Under Flash Operations, click **Flash Image**. Only the inactive image is available to flash.

Select the appropriate image and then wait until image has loaded.

Note: this process may take a while depending on the available connection speed.

When the image has loaded, the Update Firmware page appears.

The screenshot shows the 'Update Firmware' page with the following details:

The flash image was uploaded.
Click one of the "Flash Image" buttons below to start the flash procedure.

- MD5 Checksum: 47b323412e2e26403dc1f832fc9bb011
- Size: 6.68 MB (14.00 MB available)

LIS-15.00.72.002rc1

Figure 31: The flash firmware - verify page

Click either: **Flash image and do not reboot**, or **Flash image and reboot using new image immediately**. The 'Firmware update is being applied' message appears.

When the firmware update is complete, the Update Firmware page appears. There are various messages, depending on which option you selected, or if any corruptions have occurred.

8.1.4 Flash image and do not reboot option

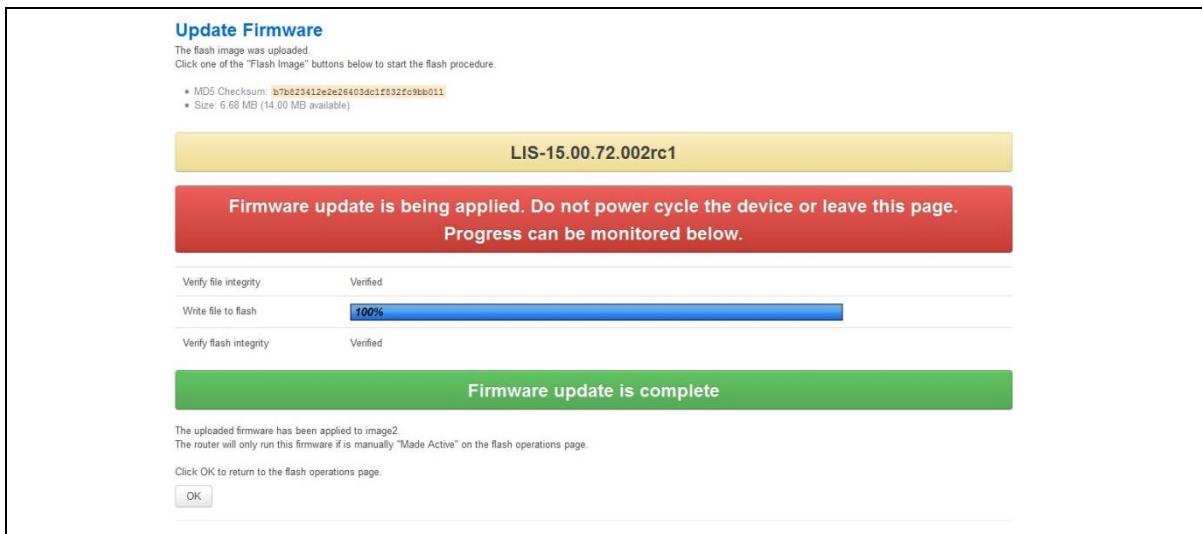


Figure 32: The firmware update page after ‘...do not reboot’ option selected

If you select ‘Flash image and do not reboot’, the router will only run the firmware if you click **OK** to return to the Flash Operations page. There you can manually select **Made Active (after reboot)**. Then click **Reboot Now** in the ‘Reboot using Active Configuration’ section.

8.1.5 Update flash image and reboot using new image immediately option

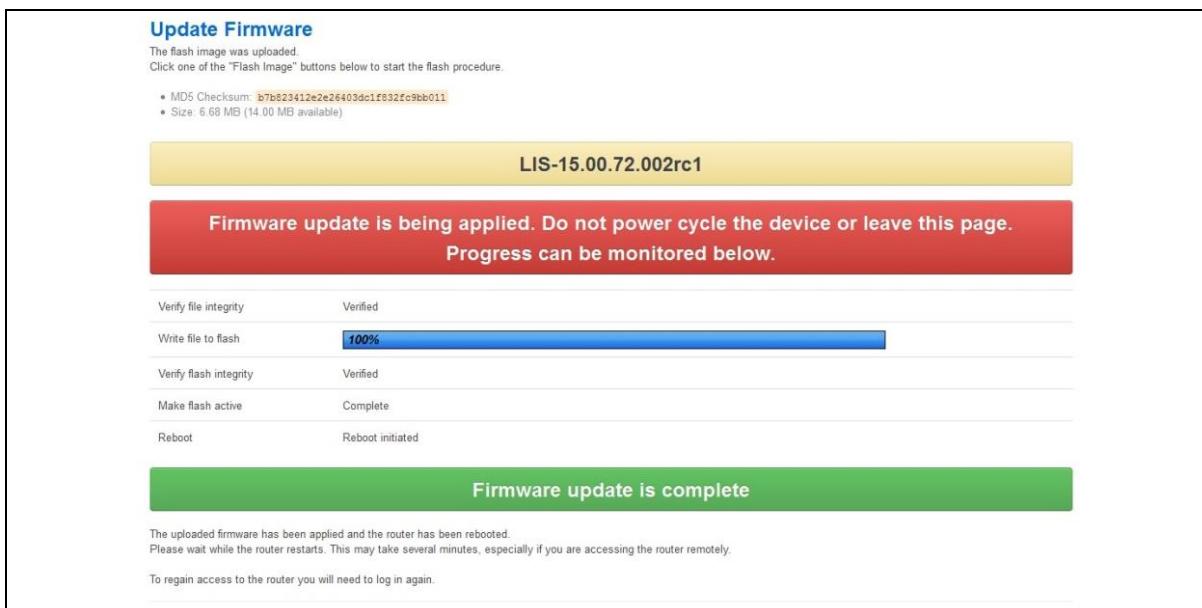


Figure 33: The firmware update page after ‘update flash image and reboot...’ option selected

If you select ‘Update flash image and reboot using new image immediately’ and the overall validation and flashing process has succeeded, the router will reboot immediately. To regain access to the router you must login again. If any part of the processes encounters an error the reboot does **not** occur and a report is given.

8.1.6 Possible file corruption

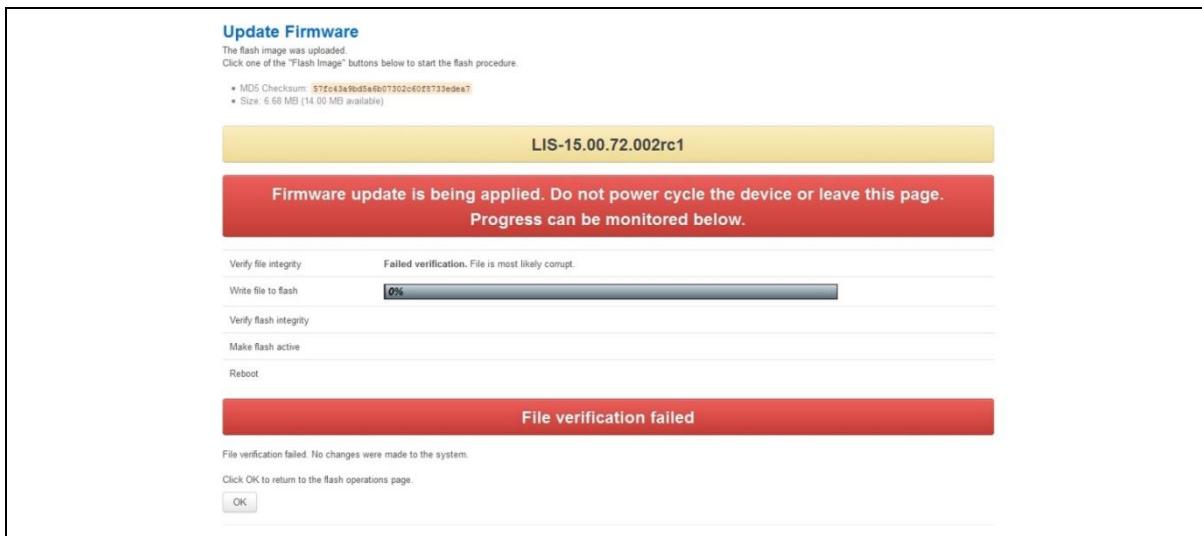


Figure 34: The firmware update failure page

In the unfortunate event that the firmware upgrade fails, the 'Failed verification File is most likely corrupt' or similar message will appear in the Verify file integrity row. No changes will be made to the system and the general message **File verification failed** appears.

8.1.7 Verify the firmware has been upgraded successfully

To check the firmware version, in the top menu, browse to **System -> Flash Operations**, or after router reboots, in the top menu, click **Status**. The Firmware Version shows in the system list and also in the right top corner of the menu bar.

Status	
System	
Router Name	GW0000
Router Model	GW0031W-AA0179E
Firmware Version	VIE-16.00.55
Current Image/Config	image2 / config2
Kernel Version	3.2.12
Local Time	Fri Aug 5 11:43:52 2016
Uptime	0h 10m 8s
Load Average	0.27, 0.35, 0.31

Figure 35: The system status list showing current firmware version

8.2 Upgrading firmware using CLI

8.2.1 Transfer file to router

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

Open up an SSH or Telnet session to the router.

Enter in the relevant username and password.

To access the temp folder, enter **cd /tmp**

Depending on the router's software version the following TFTP clients are available:

- atftp
- curl

To determine which is available on your router, enter:

```
which curl || which atftp
```

The output shows the available application:

```
/usr/bin/curl
```

ATFTP

Inline command usage:

```
atftp -g -r LIS-15.00.72.002.image -l /tmp/LIS-15.00.72.002.image x.x.x.x
```

where x.x.x.x is the IP address of your PC, **-g** is get operation and **-l** / **-r** are local and remote file name to store.

CURL

Inline command usage:

```
curl tftp://x.x.x.x/LIS-15.00.72.002.image -o /tmp/LIS-15.00.72.002.image
```

where x.x.x.x is the IP of your PC, **-o** is local file name to store.

SCP

Secure Copy (SCP) is a part of Secure Shell (SSH) and enables file transfers to the router using authentication and encryption. It is different to TFTP, which uses UDP, while SCP uses a TCP connection. On Unix machines, SCP is a standard part of the system; on Windows it requires an additional application.

The usage example below is for a Unix machine and therefore assumes the image file is in the current folder.

```
scp LIS-15.00.72.002.image root@x.x.x.x:/tmp/LIS-15.00.72.002.image
```

Where the first argument ‘LIS-15.00.72.002.image’ in SCP is the source and the second argument ‘tmp/LIS-15.00.72.002.image’ is the destination path, “root” is the username used to connect to x.x.x.x IP address.

After you execute the above command you will be asked to provide a root password.

At this stage the output shows the process of copying the software file into destination directory.

```
root@192.168.100.1's password:  
LIS-15.00.72.000.image          100%   6812KB    2.2MB/s    00:03
```

Image verification before flashing

To verify the integrity of the image, firmware version xx.yy.72.002 and later uses an image-check application.

Note: it is the user’s responsibility to verify the image before starting to write image to flash process.

To use the image-check on downloaded image, enter:

```
image-check /tmp/LIS-15.00.72.002.image
```

In the case of any image corruption, appropriate error message will be displayed:

```
Error: no SquashFS filesystem after CRC'd section - data length 3  
Error: read failed, expected at least 3 more bytes  
or other.
```

Note: image is valid only if none of error message appears. This process is done automatically during Web UI firmware update.

Flashing

When downloaded firmware verification succeeds, the new image can be written to flash.

To write the image into the alternative image, enter:

```
mtd write LIS-15.00.72.002.image altimage
```

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

Flash verification after flashing

After the write process has finished, you must complete post verification of the firmware.

To verify the checksum of downloaded firmware, enter:

```
va_image_csum.sh /tmp/LIS-15.00.72.002.image
```

The checksum of the downloaded binary is shown:

```
08761cd03e33c569873bcc24cf2b7389 7006920 LIS-15.00.72.002 This MD5
```

To verify the checksum of written firmware, enter:

```
va_image_csum.sh alt
```

After a while the checksum will be calculated:

```
Calculating checksum.....
```

```
08761cd03e33c569873bcc24cf2b7389 7006920 LIS-15.00.72.002 This MD5
```

Verify and compare the checksum with the MD5 sum of the downloaded image.

If the checksum of the written firmware in altimage matches the one from the downloaded image in /tmp, then the new firmware has been programmed successfully.

Setup an alternative image

Provided the programming has succeeded, you can set it as the next image to use after reboot, enter:

```
vacmd set next image altimage
```

To reboot using the new firmware, enter:

```
reboot
```

9 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 'GW_router'.

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

9.1 Configuration package used

Package	Sections
system	main
	timeserver

9.2 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.

9.2.1 General settings

The screenshot shows the 'System Properties' page with the 'General Settings' tab active. It includes fields for Local Time, Hostname, and Timezone, along with a note about Time Synchronization. At the bottom are Save & Apply, Save, and Reset buttons.

Figure 36: General settings in system properties

Web Field/UCI/Package Option	Description
Web: Local Time UCI: system.main.hostname Opt: hostname	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.timezone Opt: timezone	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. <input type="text" value="10m"/>

Table 12: Information table for general settings section

9.2.2 Logging

The screenshot shows the 'System Properties' dialog box with the 'Logging' tab selected. The tabs at the top are 'General Settings', 'Logging' (which is active and highlighted in blue), and 'Language and Style'. The 'Logging' tab contains the following configuration options:

- System log buffer size: A text input field containing '16' followed by a unit indicator 'kiB'.
- External system log server: A text input field containing '0.0.0.0'.
- External system log server port: A text input field containing '514'.
- Log output level: A dropdown menu set to 'Debug'.
- Cron Log Level: A dropdown menu set to 'Warning'.

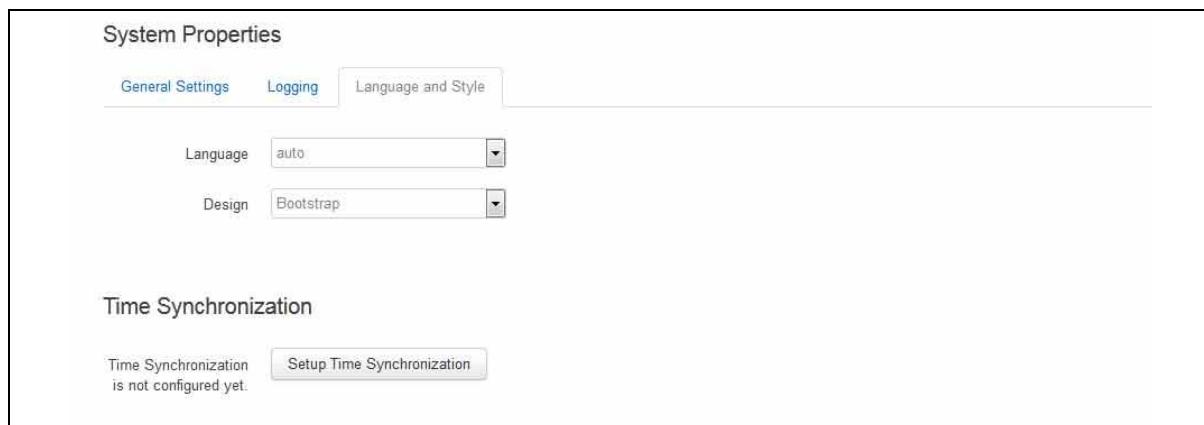
Figure 37: The logging section in system properties

Web Field/UCI/Package Option	Description				
Web: System log buffer size UCI: system.main.log_size Opt: log_size	Log buffer size in KB. <table border="1"> <tr> <td>Range</td> <td></td> </tr> <tr> <td>16</td> <td>16 KB</td> </tr> </table>	Range		16	16 KB
Range					
16	16 KB				
Web: External system log server UCI: system.main.log_ip Opt: log_ip	External syslog server IP address. <table border="1"> <tr> <td>Range</td> <td></td> </tr> <tr> <td>0.0.0.0</td> <td></td> </tr> </table>	Range		0.0.0.0	
Range					
0.0.0.0					
Web: External system log server port UCI: system.main.log_port Opt: log_port	External syslog server port number. <table border="1"> <tr> <td>Range</td> <td></td> </tr> <tr> <td>514</td> <td></td> </tr> </table>	Range		514	
Range					
514					

Web: Log output level UCI: system.main.conloglevel Opt: conloglevel	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 border="1"> <thead> <tr> <th>Web value</th> <th>Description</th> <th>UCI</th> </tr> </thead> <tbody> <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> </tbody> </table>	Web value	Description	UCI	Debug	Information useful to developers for debugging the application.	8	Info	Normal operational messages that require no action.	7	Notice	Events that are unusual, but not error conditions.	6	Warning	May indicate that an error will occur if action is not taken.	5	Error	Error conditions	4	Critical	Critical conditions	3	Alert	Should be addressed immediately	2	Emergency	System is unusable	1
Web value	Description	UCI																										
Debug	Information useful to developers for debugging the application.	8																										
Info	Normal operational messages that require no action.	7																										
Notice	Events that are unusual, but not error conditions.	6																										
Warning	May indicate that an error will occur if action is not taken.	5																										
Error	Error conditions	4																										
Critical	Critical conditions	3																										
Alert	Should be addressed immediately	2																										
Emergency	System is unusable	1																										
Web: Cron Log Level UCI: system.main.cronloglevel Opt: cronloglevel	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 border="1"> <thead> <tr> <th>Web value</th> <th>Description</th> <th>UCI</th> </tr> </thead> <tbody> <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> </tbody> </table>	Web value	Description	UCI	Normal	Normal operation messages	8	Warning	Error messages	9	Debug	Debug messages	5															
Web value	Description	UCI																										
Normal	Normal operation messages	8																										
Warning	Error messages	9																										
Debug	Debug messages	5																										
Web: n/a UCI: system.main.log_file Opt: log_file	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'																											
Web: n/a UCI: system.main.log_type Opt: log_type	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.																											
Web: n/a UCI: system.main.log_file_count Opt: log_file_count	Defines the number of archive syslog files to store in flash. When configured above to write to /root/syslog.messages files will be stored at /root/syslog.messages,x (where x starts at 0). <table border="1"> <tr> <td>Range</td> <td></td> </tr> <tr> <td>1</td> <td>Stores 1 archive log file in flash</td> </tr> </table>	Range		1	Stores 1 archive log file in flash																							
Range																												
1	Stores 1 archive log file in flash																											

Table 13: Information table for the logging section

9.2.3 Language and style

**Figure 38: The language and style section in system properties**

Web Field/UCI/Package Option	Description					
Language	Sets the language to 'auto' or 'English'.	<table border="1"> <tr> <td>Auto</td> <td></td> </tr> <tr> <td>English</td> <td></td> </tr> </table>	Auto		English	
Auto						
English						
Design	Sets the router's style.					

Table 14: Information table for the language and style page

9.2.4 Time synchronization

The router time must be synchronised using NTP. The router can act as both an NTP client and an NTP server. It is enabled as an NTP client by default and individual interfaces can be configured to respond to NTP requests.

The screenshot shows the 'Time Synchronization' configuration page. It includes the following fields:

- NTP update interval: A dropdown menu set to "auto".
- NTP server candidates: A list box containing three entries: "0.openwrt.pool.ntp.org", "1.openwrt.pool.ntp.org", and "3.openwrt.pool.ntp.org". Each entry has a red 'X' icon to its right.
- NTP Server Interface: A dropdown menu set to "lan".
- NTP Server Stratum: An empty input field.

Figure 39: The time synchronization section in system properties

Web Field/UCI/Package Option	Description					
Web: NTP update interval UCI: system.ntp.interval_hours Opt: interval_hours	Specifies interval of NTP requests in hours. Default value set to auto.	<table border="1"> <tr> <td>auto</td> <td></td> </tr> <tr> <td>Range</td> <td>auto; 1-23</td> </tr> </table>	auto		Range	auto; 1-23
auto						
Range	auto; 1-23					
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.					
Web: NTP Server Interface UCI: system.ntp.listen Opt: listen	Defines a list of interfaces that respond to NTP requests. Interfaces should be delimited using space. Example: option listen 'LAN1 LAN2'	<table border="1"> <tr> <td>blank</td> <td>Do not respond to NTP requests</td> </tr> <tr> <td>Range</td> <td></td> </tr> </table>	blank	Do not respond to NTP requests	Range	
blank	Do not respond to NTP requests					
Range						
Web: NTP Server Stratum UCI: system.ntp.stratum Opt: stratum	Defines how far this NTP Server is from the reference clock. For example, an NTP server getting time directly from the reference clock will have a stratum of 1. In general, this should be left blank, which means that the router NTP Server will derive the stratum from the NTP dialogue.	<table border="1"> <tr> <td>blank</td> <td>NTP server will derive stratum</td> </tr> <tr> <td>Range</td> <td></td> </tr> </table>	blank	NTP server will derive stratum	Range	
blank	NTP server will derive stratum					
Range						

Table 15: Information table for time synchronization section

9.2.5 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.

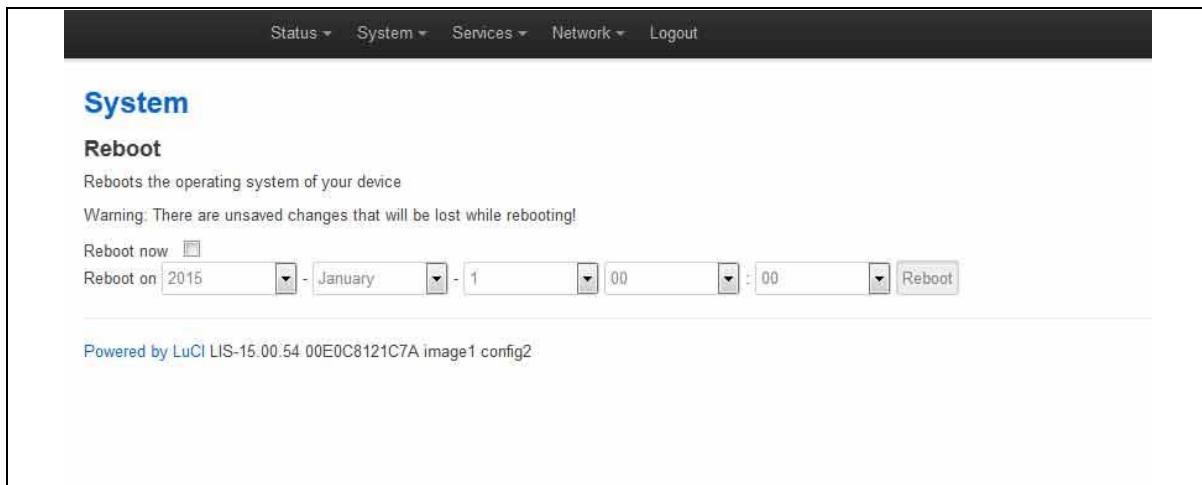


Figure 40: The reboot page

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

9.3 System settings using UCI

```
root@GW_router:~# uci show system
system.main=system
system.main.hostname=GW_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.GW_router.pool.ntp.org 10.10.10.10
System settings using package options
root@GW_router:~# uci export system
package 'system'

config 'system' 'main'
    option 'hostname' "GW_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.GW_router.pool.ntp.org"
    list server '10.10.10.10'
    option listen 'LAN1 LAN2'

```

9.4 System diagnostics

9.4.1 System events

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

9.4.1.1 Logread

To view the system log, enter:

```
root@GW_router:~# logread
```

Shows the log.

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

Shows end of the log.

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

Shows the log page by page.

```
root@GW_router:~# logread -f
```

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

```
root@GW_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.

9.4.2 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@GW_router:~# uci export system
package system
config system 'main'
    option hostname 'GW_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@GW_router:~# cat /root/syslog.messages
```

Shows all the system events stored in flash.

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

Shows end of the events stored flash.

```
root@GW_router:~# tail -f /root/syslog.messages &
```

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

10 Configuring an Ethernet interface on a GW100 router

This section describes how to configure an Ethernet interface on a GW100, including configuring the interface as a DHCP server, adding the interface to a firewall zone and mapping the physical switch ports.

10.1 Configuration packages used

Package	Sections
network	interface
	route
	alias
	zone
firewall	
dhcp	dhcp

10.2 Configuring an Ethernet interface using the web interface

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

Figure 41: 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.

10.3 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.

10.3.1 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.

The allowed characters are: A-Z, a-z, 0-9 and _

Static address

Create a bridge over multiple interfaces

Cover the following interface:

- Ethernet Adapter: "eth0" (lan)
- Ethernet Adapter: "eth1" (lan1)
- Ethernet Adapter: "eth2"
- Ethernet Adapter: "eth3"
- Ethernet Adapter: "eth4"
- Ethernet Adapter: "lo" (loopback)
- Ethernet Adapter: "teg10"
- Ethernet Adapter: "tunl0"
- Custom Interface:

Note: If you choose an interface here which is part of another network, it will be moved into this network.

Back to Overview Submit

Figure 42: 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 _																										
Web: Protocol of the new interface UCI: network.<if name>.proto Opt: proto	Specifies what protocol the interface will operate on. Select Static . <table border="1" data-bbox="695 415 1394 977"> <thead> <tr> <th>Option</th><th>Description</th></tr> </thead> <tbody> <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> </tbody> </table>	Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.	IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.	GRE	Generic Routing Encapsulation protocol	IOT		L2TP	Layer 2 Tunnelling Protocol	PPP	Point to Point Protocol	PPPoE	PPP over Ethernet	PPPoATM	PPP over ATM	LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.
Option	Description																										
Static	Static configuration with fixed address and netmask.																										
DHCP Client	Address and netmask are assigned by DHCP.																										
Unmanaged	Unspecified																										
IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.																										
IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.																										
GRE	Generic Routing Encapsulation protocol																										
IOT																											
L2TP	Layer 2 Tunnelling Protocol																										
PPP	Point to Point Protocol																										
PPPoE	PPP over Ethernet																										
PPPoATM	PPP over ATM																										
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																										
Web: Create a bridge over multiple interfaces UCI: network.<if name>.type Opt: type	If you select this option, then the new logical interface created will act as a bridging interface between the chosen existing physical interfaces. <table border="1" data-bbox="695 1066 1330 1156"> <thead> <tr> <th>Empty</th><th></th></tr> </thead> <tbody> <tr><td>Bridge</td><td>Configures a bridge over multiple interfaces.</td></tr> </tbody> </table>	Empty		Bridge	Configures a bridge over multiple interfaces.																						
Empty																											
Bridge	Configures a bridge over multiple interfaces.																										
Web: Cover the following interface UCI: network.<if name>.ifname Opt: ifname	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'																										

Table 16: 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

10.3.2 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

10.3.2.1 Common configuration – general setup

Common Configuration

- [General Setup](#)
- [Advanced Settings](#)
- [Physical Settings](#)
- [Firewall Settings](#)

Status	 eth3	MAC Address: 00:E0:C8:D3:18:20 RX: 0.00 B (0 Pkts.) TX: 0.00 B (0 Pkts.)
Protocol	Static address	
IPv4 address	<input type="text"/>	
IPv4 netmask	<input type="text"/>	
IPv4 gateway	<input type="text"/>	
IPv4 broadcast	<input type="text"/>	
Use custom DNS servers	<input type="text"/> 	
Accept router advertisements	<input type="checkbox"/>	
Send router solicitations	<input checked="" type="checkbox"/>	
IPv6 address	<input type="text"/>	
IPv6 gateway	<input type="text"/>	

Figure 43: The Ethernet connection common configuration settings page

Web Field/UCI/Package Option	Description
Web: status	Shows the current status of the interface.

Web:Protocol UCI: nework.<if name>.proto Opt:proto	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 border="1"> <thead> <tr> <th>Option</th><th>Description</th></tr> </thead> <tbody> <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 Tunneling 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> </tbody> </table>	Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.	Ipv6-over-IPv4	Stateless IPv6 over IPv4 transport.	GRE	Generic Routing Encapsulation protocol.	IOT		L2TP	Layer 2 Tunneling Protocol.	PPP	Point-to-Point Protocol.	PPPoE	PPP over Ethernet.	PPPoATM	PPP over ATM	LTE/UMTS/GPRS/EV-DO	CDMA, UMTS, or GPRS connection using an AT-style 3G modem.
Option	Description																										
Static	Static configuration with fixed address and netmask.																										
DHCP Client	Address and netmask are assigned by DHCP.																										
Unmanaged	Unspecified																										
IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.																										
Ipv6-over-IPv4	Stateless IPv6 over IPv4 transport.																										
GRE	Generic Routing Encapsulation protocol.																										
IOT																											
L2TP	Layer 2 Tunneling Protocol.																										
PPP	Point-to-Point Protocol.																										
PPPoE	PPP over Ethernet.																										
PPPoATM	PPP over ATM																										
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS, or GPRS connection using an AT-style 3G modem.																										
Web: IPv4 address UCI: network.<if name>.ipaddr Opt: ipaddr	The IPv4 address of the interface. This is optional if an IPv6 address is provided.																										
Web:IPv4 netmask UCI: network.<if name> .netmask Opt: netmask	Subnet mask to be applied to the IP address of this interface.																										
Web:IPv4 gateway UCI: network.<if name> .gateway Opt: gateway	IPv4 default gateway to assign to this interface (optional).																										
Web:IPv4 broadcast UCI: network.<if name> .broadcast Opt: broadcast	Broadcast address. This is automatically generated if no broadcast address is specified.																										
Web:Use custom DNS servers UCI: network.<if name> .dns Opt: dns	List of DNS server IP addresses (optional). Multiple DNS Servers are separated by a space when using UCI or CLI.																										
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 .																										
Web:Send router solicitations UCI: network.<if name> Opt:send_rs	Specifies whether to send Router Soliticitations on this interface (optional). Note: defaults to 1 for static protocol, otherwise defaults to 0 .																										
Web:IPv6 address UCI: network.<if name> .ip6addr Opt: ip6addr	The IPv6 IP address if 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 17: Information table for LAN interface common configuration settings

10.3.2.2 Common configuration: advanced settings

The screenshot shows the 'Common Configuration' page with the 'Advanced Settings' tab selected. Key settings include:

- Bring up on boot:** Checked.
- Monitor interface state:** Disabled (unchecked).
- Override MAC address:** 00:E0:C8:12:27:20.
- Override MTU:** 1500.
- Use gateway metric:** 0.
- Dependant interfaces:** MOBILE, PPPADSL, lan, loopback.

Figure 44: 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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
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. <table border="1"> <tr> <td>1500</td><td>1500 bytes</td></tr> <tr> <td>Range</td><td></td></tr> </table>	1500	1500 bytes	Range							
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). <table border="1"> <tr> <td>0</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	0		Range							
0											
Range											
Web: Dependant Interfaces UCI: network.[..x..].dependants Opt: dependants	Lists interfaces that are dependant on this parent interface. Dependant interfaces will go down when parent interface is down and will start or restart when parent interface starts. Separate multiple interfaces by a space when using UCI. Example: option dependants 'PPPADSL MOBILE' This replaces the following previous options in child interfaces. <table border="1"> <tr> <td>gre</td><td>option local_interface</td></tr> <tr> <td>lt2p</td><td>option src_ipaddr</td></tr> <tr> <td>iot</td><td>option wan1 wan2</td></tr> <tr> <td>6in4</td><td>option ipaddr</td></tr> <tr> <td>6to4</td><td>option ipaddr</td></tr> </table>	gre	option local_interface	lt2p	option src_ipaddr	iot	option wan1 wan2	6in4	option ipaddr	6to4	option ipaddr
gre	option local_interface										
lt2p	option src_ipaddr										
iot	option wan1 wan2										
6in4	option ipaddr										
6to4	option ipaddr										
Web: SNMP Alias ifindex UCI: network.[..x..].snmp_alias_ifindex Opt: snmp_alias_ifindex	Defines a static SNMP interface alias index for this interface, that can be polled via the SNMP interface index (snmp_alias_ifindex+1000). See <i>Configuring SNMP</i> section for more information <table border="1"> <tr> <td>Blank</td><td>No SNMP interface alias index</td></tr> <tr> <td>Range</td><td>0 - 4294966295</td></tr> </table>	Blank	No SNMP interface alias index	Range	0 - 4294966295						
Blank	No SNMP interface alias index										
Range	0 - 4294966295										

Table 18: Information table for common configuration advanced settings**10.3.2.3 Common configuration: physical settings**

The screenshot shows the 'Common Configuration' interface with the 'Physical Settings' tab active. It includes sections for 'Bridge interfaces', 'VLAN PCP to skb->priority mapping', 'skb->priority to VLAN PCP mapping', and a list of available 'Interface' options.

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. <table border="1"> <tr> <td>Blank</td> <td></td> </tr> <tr> <td>Bridge</td> <td>Configures a bridge over multiple interfaces.</td> </tr> </table>	Blank		Bridge	Configures a bridge over multiple interfaces.
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. <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
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				
Web: Auto Negotiation UCI: network.<if name>.autoneg Opt: autoneg	Specifies if Speed and Duplex mode should be autonegotiated <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Full Duplex UCI: network.<if name>.fullduplex Opt: fullduplex	Ability to change duplex mode <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Ethernet Speed UCI: network.<if name>.speed Opt: speed	Sets Ethernet speed. Available options are: Eth0:10,100,1000 Eth1:10,100				

Figure 45: 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. <table border="1"> <tr> <td>Blank</td><td></td></tr> <tr> <td>Bridge</td><td>Configures a bridge over multiple interfaces.</td></tr> </table>	Blank		Bridge	Configures a bridge over multiple interfaces.
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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
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				
Web: Auto Negotiation UCI: network.<if name>.autoneg Opt: autoneg	Specifies if Speed and Duplex mode should be autonegotiated <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Full Duplex UCI: network.<if name>.fullduplex Opt: fullduplex	Ability to change duplex mode <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Ethernet Speed UCI: network.<if name>.speed Opt: speed	Sets Ethernet speed. Available options are: Eth0:10,100,1000 Eth1:10,100				

Table 19: Information table for physical settings page

10.3.2.4 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 46: GRE firewall settings

10.3.3 Interface overview: IP-aliases

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

10.3.3.1 IP-alias packages used

Package	Sections
Network	alias

10.3.3.2 Configuring 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 47: 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 20: 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.

10.3.3.3 IP-aliases: general setup

The screenshot shows a web-based configuration interface for an IP alias named 'ETHALIAS1'. At the top right is a 'Delete' button. Below it, there are tabs for 'General Setup' (which is selected) and 'Advanced Settings'. Under 'General Setup', there are three input fields: 'IPv4-Address' (with a placeholder '192.168.1.100'), 'IPv4-Netmask' (with a dropdown menu containing '255.255.255.0'), and 'IPv4-Gateway' (with a placeholder '192.168.1.1'). At the bottom left is a large empty input field, and at the bottom right is an 'Add' button.

Figure 48: 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 21: Information table for IP-Alias general setup page

10.3.3.4 IP-aliases: advanced settings

The screenshot shows the same web-based configuration interface for 'ETHALIAS1'. The 'Advanced Settings' tab is now selected. It contains two input fields: 'IPv4-Broadcast' (with a placeholder '192.168.1.255') and 'DNS-Server' (with a placeholder '8.8.8.8'). At the bottom left is a large empty input field, and at the bottom right is an 'Add' button.

Figure 49: 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 22: Information table for IP-Alias advanced settings page

10.3.4 Interface overview: DHCP server

10.3.4.1 DHCP server: packages used

Package	Sections
dhcp	dhcp

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

DHCP Server
General Setup
Ignore interface Disable DHCP for this interface.

Figure 50: The DHCP Server settings section

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

10.3.4.2 DHCP server: general setup

DHCP Server
General Setup Advanced Settings
Ignore interface Disable DHCP for this interface.
Start 100 Lowest leased address as offset from the network address.
Limit 150 Maximum number of leased addresses.
Leasetime 12h Expiry time of leased addresses, minimum is 2 Minutes (2m).

Figure 51: The DHCP server general setup section

Web Field/UCI/Package Option	Description					
Web: Ignore interface UCI: dhcp.@dhcp[x].ignore Opt: ignore	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 border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.					
1	Enabled.					
Web: n/a UCI: dhcp.@dhcp[x].start Opt: start	Defines the offset from the network address for the start of the DHCP pool. It may be greater than 255 to span subnets.	<table border="1"> <tr> <td>100</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	100		Range	
100						
Range						
Web: n/a UCI: dhcp.@dhcp[x].limit Opt: limit	Defines the offset from the network address for the end of the DHCP pool.	<table border="1"> <tr> <td>150</td><td></td></tr> <tr> <td>Range</td><td>0 – 255</td></tr> </table>	150		Range	0 – 255
150						
Range	0 – 255					
Web: n/a UCI: dhcp.@dhcp[x].leasetime Opt: leasetime	Defines the lease time of addresses handed out to clients, for example 12h or 30m.	<table border="1"> <tr> <td>12h</td><td>12 hours</td></tr> <tr> <td>Range</td><td></td></tr> </table>	12h	12 hours	Range	
12h	12 hours					
Range						

Table 23: Information table for DHCP server general setup page

10.3.4.3 DHCP Server: advanced settings

DHCP Server

General Setup **Advanced Settings**

Dynamic DHCP Dynamically allocate DHCP addresses for clients. If disabled, only clients having static leases will be served.

Force Force DHCP on this network even if another server is detected.

IPv4-Netmask Override the netmask sent to clients. Normally it is calculated from the subnet that is served.

DHCP-Options Define additional DHCP options, for example "6,192.168.2.1,192.168.2.2" which advertises different DNS servers to clients.

Figure 52: 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. <table border="1"> <tr> <td>1</td><td>Dynamically allocate leases.</td></tr> <tr> <td>0</td><td>Use /etc/ethers file for serving DHCP leases.</td></tr> </table>	1	Dynamically allocate leases.	0	Use /etc/ethers file for serving 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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
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 values should be separated by a space. Example: list dhcp_option 6,192.168.2.1 192.168.2.2 <table border="1"> <tr> <td></td><td>No options defined.</td></tr> <tr> <td>Syntax</td><td>Option_number, option_value</td></tr> </table>		No options defined.	Syntax	Option_number, option_value
	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 24: Information table for DHCP advanced settings page

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

10.4 Configuring an Ethernet interface using UCI

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

```
root@GW_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

root@GW_router:~# uci show firewall
.....
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@GW_router:~# uci show dhcp
...
dhcp.@dhcp[0]=dhcp
dhcp.@dhcp[0].start=100
root@GW_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.

```

10.4.1 Interface common configuration using package options

The configuration files are stored on /etc/config/network, /etc/config/firewall and /etc/config/dhcp.

```

root@GW_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 wlan_qos_map_ingress '1:2'
    list wlan_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@GW_router:~# uci export firewall
package firewall
config zone
    option name 'lan'
    option input 'ACCEPT'
    option output 'ACCEPT'
    option network 'lan newinterface'

root@GW_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.

10.4.2 Configuring ATM bridges

The ATM bridges section is not used when configuring an Ethernet interface on a GW100 router.

10.5 Interface diagnostics

10.5.1 Interfaces status

To show the current running interfaces, enter:

```
root@GW_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:

```
root@GW_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)
```

10.5.2 Route status

To show the current routing status, enter:

```
root@GW_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.

11 Configuring VLAN

11.1 Maximum number of VLANs supported

SATEL GW routers support up to 4095 VLANs.

11.2 Configuration package used

Package	Sections
Network	

11.3 Configuring VLAN using the web interface

11.3.1 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.

The screenshot shows the 'Create Interface' configuration page. At the top, there is a navigation bar with links for Status, System, Services, Network, and Logout. Below the navigation bar, the title 'Create Interface' is displayed. The form contains the following fields:

- Name of the new interface: A text input field with a note: "The allowed characters are: A-Z, a-z, 0-9 and _".
- Protocol of the new interface: A dropdown menu set to "Static address".
- Create a bridge over multiple interfaces: A checkbox that is unchecked.
- Cover the following interface: A list of radio buttons next to interface names. The options are:
 - Ethernet Adapter: "eth0" (lan) (selected)
 - Ethernet Adapter: "eth1" (lan1)
 - Ethernet Adapter: "eth2"
 - Ethernet Adapter: "eth3"
 - Ethernet Adapter: "eth4"
 - Ethernet Adapter: "lo" (loopback)
 - Ethernet Adapter: "teq10"
 - Ethernet Adapter: "tunl0"
 - Custom Interface: (input field)
- Note: If you choose an interface here which is part of another network, it will be moved into this network.

At the bottom of the page are two buttons: "Back to Overview" and "Submit".

Figure 53: 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.																										
Web: Protocol of the new interface UCI: network.vlan_test.proto Opt: proto	<p>Protocol type. Select Static.</p> <table border="1"> <thead> <tr> <th>Option</th><th>Description</th></tr> </thead> <tbody> <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> </tbody> </table>	Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.	IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.	GRE	Generic Routing Encapsulation protocol	IOT		L2TP	Layer 2 Tunnelling Protocol	PPP	Point to Point Protocol	PPPoE	PPP over Ethernet	PPPoATM	PPP over ATM	LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.
Option	Description																										
Static	Static configuration with fixed address and netmask.																										
DHCP Client	Address and netmask are assigned by DHCP.																										
Unmanaged	Unspecified																										
IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.																										
IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.																										
GRE	Generic Routing Encapsulation protocol																										
IOT																											
L2TP	Layer 2 Tunnelling Protocol																										
PPP	Point to Point Protocol																										
PPPoE	PPP over Ethernet																										
PPPoATM	PPP over ATM																										
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																										
Web: Create a bridge over multiple interfaces UCI: network.vlan1.type Opt: type	Create a bridge over multiple interfaces.																										
Web: Cover the following interface UCI: network.vlan1.ifname Opt: ifname	<p>Check the Custom Interface radio button. Enter a name, for example eth0.100. This will assign VLAN 100 to the eth0 interface.</p>																										

Table 25: Information table for the create interface page

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

11.3.2 General setup: VLAN

The screenshot shows the 'Interfaces - VLAN1' configuration page. At the top, there are tabs for WAN, VLAN1 (which is selected), VLAN2, and LAN. Below the tabs, there's a section titled 'Common Configuration' with tabs for General Setup (selected), Advanced Settings, Physical Settings, and Firewall Settings. The main area displays the status of interface eth0.1, including uptime, MAC address, RX/TX statistics, and IP configuration (IPv4: 172.16.100.1/24). Below this, there are fields for setting the protocol (Static address), IPv4 address (172.16.100.1), IPv4 netmask (255.255.255.0), IPv4 gateway, IPv4 broadcast, and custom DNS servers.

Figure 54: The VLAN 1 interface page

Web Field/UCI/Package Option	Description																										
Web: Protocol UCI: network.VLAN1.proto Opt: proto	<p>Protocol type.</p> <table border="1"> <thead> <tr> <th>Option</th><th>Description</th></tr> </thead> <tbody> <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> </tbody> </table>	Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.	IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.	GRE	Generic Routing Encapsulation protocol	IOT		L2TP	Layer 2 Tunnelling Protocol	PPP	Point to Point Protocol	PPPoE	PPP over Ethernet	PPPoATM	PPP over ATM	LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.
Option	Description																										
Static	Static configuration with fixed address and netmask.																										
DHCP Client	Address and netmask are assigned by DHCP.																										
Unmanaged	Unspecified																										
IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.																										
IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.																										
GRE	Generic Routing Encapsulation protocol																										
IOT																											
L2TP	Layer 2 Tunnelling Protocol																										
PPP	Point to Point Protocol																										
PPPoE	PPP over Ethernet																										
PPPoATM	PPP over ATM																										
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																										
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 26: Information table for VLAN general settings

11.3.3 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.

The screenshot shows a 'Common Configuration' screen with three tabs: 'General Setup', 'Advanced Settings', and 'Firewall Settings'. The 'Firewall Settings' tab is active. Below the tabs, there is a section titled 'Create / Assign firewall-zone' with a radio button next to 'unspecified -or- create:'. A tooltip below the radio button says: 'Choose 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.' At the bottom of the screen, there are buttons for 'Back to Overview', 'Save & Apply' (highlighted in blue), 'Save', and 'Reset'.

Figure 55: Firewall settings page

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

11.4 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.

The screenshot shows the 'Interfaces' section of the MikroTik interface overview. At the top, there are tabs for WAN, VLAN1, VLAN2, and LAN. Below the tabs, the title 'Interfaces' is followed by 'Interface Overview'. A table lists four network interfaces:

Network	Status	Actions
VLAN1	Uptime: 0h 1m 29s MAC Address: 00:E0:C8:10:10:50 RX: 0.00 B (0 Pkts.) TX: 376.00 B (9 Pkts.) IPv4: 172.16.100.1/24	Connect Stop Edit Delete
VLAN2	Uptime: 0h 1m 5s MAC Address: 00:E0:C8:10:10:50 RX: 0.00 B (0 Pkts.) TX: 126.00 B (3 Pkts.) IPv4: 192.168.200.1/24	Connect Stop Edit Delete
LAN	Uptime: 0h 1m 43s MAC Address: 00:E0:C8:10:10:50 RX: 3.81 MB (38450 Pkts.) TX: 1.51 MB (2437 Pkts.) IPv4: 10.1.10.248/16	Connect Stop Edit Delete
LOOPBACK	Uptime: 0h 1m 49s MAC Address: 00:00:00:00:00:00 RX: 396.64 KB (3456 Pkts.) TX: 396.64 KB (3456 Pkts.) IPv4: 127.0.0.1/8	Connect Stop Edit Delete

Figure 56: The interface overview page showing two VLAN interfaces

11.5 Configuring VLAN using the UCI interface

You can configure VLANs through CLI. The VLAN configuration file is stored on:
/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.

12 Configuring ignition sense

In automotive applications, the ignition sense input can detect when the vehicle's ignition has been enabled. This allows the router to remain powered on after the vehicle has stopped. The time delay between ignition off and power down is configurable.

Routers for ignition sense applications are supplied with a power lead with 3 connectors for 12V permanent, 12V switched (ignition sense) and ground.

12.1 Configuration packages used

Package	Sections
vapowermond	main

12.2 Configuring vapowermond using the web interface

You can configure the Vapowermond package using the web interface. In the top menu, click **Services ->Power Monitor**. The basic settings page appears.

The screenshot shows the 'Power Monitor' configuration page. It has a 'Basic' tab selected under 'Settings'. The configuration includes:

- Enable:** A checked checkbox labeled 'Enable Power Monitor'.
- Ignition Timeout:** A text input field containing '130' with a tooltip explaining it's the time in minutes from IGNITION POWER OFF to board power down.
- Enable Scripts:** An unchecked checkbox labeled 'Execute scripts upon detection of power loss/restoration'.
- Voltage On Script:** A text input field containing '/usr/bin/powermon_voltage_on.sh' with a tooltip explaining it's the script to execute on detection of power on.
- Voltage Off Script:** A text input field containing '/usr/bin/powermon_voltage_off.sh' with a tooltip explaining it's the script to execute on detection of power off.
- Message Prefix:** A text input field containing 'IgnitionPower' with a tooltip explaining it's the Syslog message prefix for messages IgnitionPowerOn, IgnitionPowerOff.

At the bottom right are buttons for 'Save & Apply', 'Save', and 'Reset'.

Figure 57: Power monitor basic settings page

12.2.1 Power monitor basic settings

Web field/UCI/Package Option	Description	
Web field: Enable UCI: vapowermond.main.enabled Opt: enabled	Enables vapowermond package on a router.	
	0	
	Range	0-1
Web field: Ignition Timeout UCI: vapowermond.main.timeout Opt: timeout	Time in minutes from ignition power off to router power down. Set to 0 to disable the timer.	
	30	
	Range	0-infinite
Web field: Enable Scripts UCI: vapowermond.main.voltage_sense_scripts_enable Opt:voltage_sense_scripts_enable	Execute scripts upon detection of power loss/restoration.	
	0	Disabled
	Range	0-1
Web field: Voltage On Script UCI: vapowermond.main.voltage_on_script Opt: voltage_on_script	Script to execute on detection of power on.	
	<code>/usr/bin/powermon_voltage_on.sh</code>	
Web field: Voltage Off Script UCI: vapowermond.main.voltage_off_script Opt: voltage_off_script	Script to execute on detection of power off.	
	<code>/usr/bin/powermon_voltage_off.sh</code>	
Web field: Message Prefix UCI: vapowermond.main.voltage_msg Opt: voltage_msg	Syslog message prefix for messages IgnitionPowerOn, IgnitionPowerOff.	

Table 27: Information table for power monitor basic settings

12.2.2 Power monitor advanced settings

Click the Advance tab to access advanced settings.

The screenshot shows the 'Power Monitor' configuration interface. At the top, it says 'Power Monitor Configuration'. Below that, there's a 'Settings' section with two tabs: 'Basic' (which is currently selected) and 'Advanced'. Under 'Basic' settings, there is a 'Syslog Severity' dropdown set to 'Informational'. A tooltip for this field states: 'Specifies the lowest severity to be logged by Power Monitor'. At the bottom of the page are three buttons: 'Save & Apply', 'Save', and 'Reset'.

Figure 58: Power monitor advanced settings page

Web field/UCI/Package Option	Description	
Web field: Syslog Severity UCI: vapowermond.main.log_severity Opt: log_severity	Specifies the lowest severity to be logged by Power Monitor.	
0	Emergency	
1	Alert	
2	Critical	
3	Error	
4	Warning	
5	Notice	
6	Informational	
7	Debug	

Table 28: Information table for power monitor advanced settings

12.3 Configuring vapowermond using the command line

12.3.1 UCI

```
root@GW_router:~# uci show vapowermond
vapowermond.main=vapowermond
vapowermond.main.enabled=1
vapowermond.main.timeout=30
vapowermond.main.voltage_sense_scripts_enable=0
vapowermond.main.voltage_on_script=/usr/bin/powermon_voltage_on.sh
vapowermond.main.voltage_off_script=/usr/bin/powermon_voltage_off.sh
vapowermond.main.voltage_msg=powermon
vapowermond.main.log_severity=5
```

12.3.2 Package options

```
root@GW_router:~# uci export vapowermond
package vapowermond

config vapowermond 'main'
    option enabled '1'
    option timeout '30'
    option voltage_sense_scripts_enable '0'
    option voltage_on_script '/usr/bin/powermon_voltage_on.sh'
    option voltage_off_script '/usr/bin/powermon_voltage_off.sh'
    option voltage_msg 'powermon'
    option log_severity '5'
```

12.4 Ignition sense diagnostics

12.4.1 Monitoring Vapowermond status using the command line interface

To view status information about the current ignition sense state enter:

```
root@GW_router:~# cat /sys/class/gpio/gpio29/value  
1
```

1 for ignition **on**;0 for ignition **off**

13 Configuring a WiFi connection

This section explains how to configure WiFi on a SATEL 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

13.1 Configuration packages used

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

13.2 Configuring a WiFi interface using the web interface

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

The screenshot shows the 'Wireless Overview' page. At the top, there is a header with the title 'Wireless Overview'. Below the header, there is a section for a 'Generic 802.11abgn Wireless Controller (radio0)'. This section includes a small icon of a radio tower, the controller's name, and two buttons: 'Scan' and 'Add'. A message below the icon states 'No network configured on this device'. Below this, there is a section titled 'Associated Stations' with a table header containing columns for SSID, MAC, Address, Signal, Noise, RX Rate, and TX Rate. A message below the table header says 'No information available'.

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

13.2.1 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.

13.2.1.1 Device configuration: general setup

Device Configuration

General Setup Advanced Settings

Status: SSID: OpenWrt | Mode: Master
100% Wireless is disabled or not associated

Wireless network is enabled

Channel: 4 (2.427 GHz)

Transmit Power: 20 dBm (100 mW)

Figure 60: The device configuration general setup section

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

Table 29: Information table for the device configuration section

13.2.1.2 Device configuration: advanced settings

The screenshot shows the 'Device Configuration' page with the 'Advanced Settings' tab selected. The 'Mode' dropdown is set to '802.11g+n'. The 'HT mode' dropdown is set to '20MHz'. The 'Country Code' dropdown is set to 'US - United States' with a note: 'Use ISO/IEC 3166 alpha2 country codes.' The 'Distance Optimization' field is empty with a note: 'Distance to farthest network member in meters.' The 'Fragmentation Threshold' and 'RTS/CTS Threshold' fields are also empty.

Figure 61: The device configuration advanced settings section

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

Table 30: Information table for device configuration advanced settings

13.2.2 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.

13.2.2.1 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.

Interface Configuration

General Setup Wireless Security MAC-Filter

ESSID: OpenWrt

Mode: Access Point

Network:

- 3G
- ADSL
- Test_BC
- Tunnel1
- lan
- lan2
- lan3
- lan4
- loopback
- unspecified -or- create:

Choose the network you want to attach to this wireless interface.
Select unspecified to not attach any network or fill out the create field to define a new network.

Hide ESSID

Save & Apply Save Reset

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.																		
Web: Mode UCI: wireless.@wifi-iface[0].mode Opt: mode	Mode type. For AP mode, select Access Point . <table border="1" style="margin-left: 20px;"> <tr><td>Web value</td><td>UCI</td></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>	Web value	UCI	Access Point	ap	Client	sta	Ad-Hoc	adhoc	802.11s	mesh	Pseudo Ad-Hoc (ah demo)	ahdemo	Monitor	monitor	Access Point (WDS)	ap-wds	Client (WDS)	sta-wds
Web value	UCI																		
Access Point	ap																		
Client	sta																		
Ad-Hoc	adhoc																		
802.11s	mesh																		
Pseudo Ad-Hoc (ah demo)	ahdemo																		
Monitor	monitor																		
Access Point (WDS)	ap-wds																		
Client (WDS)	sta-wds																		
Web: Mode UCI: wireless.@wifi-iface[0].bssid Opt: bssid	Defines the BSSID value. Only displayed if using client, ad-hoc or client (wds) modes.																		
Web: Network UCI: wireless.@wifi-iface[0].network Opt: network	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.																		
Web: Hide ESSID UCI: wireless.@wifi-iface[0].hidden Opt: hidden	Hides the SSID when enabled. Only displayed if using access point or access point (wds) modes <table border="1" style="margin-left: 20px;"> <tr><td>1</td><td>Enabled.</td></tr> <tr><td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.														
1	Enabled.																		
0	Disabled.																		

Table 31: Information table for the interface configuration general setup section

13.2.2.2 Interface configuration: wireless security

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

The screenshot shows the 'Interface Configuration' screen with the 'General Setup' tab active. Within the 'General Setup' tab, the 'Wireless Security' sub-tab is selected. The configuration includes:

- Encryption:** WPA2-PSK
- Cipher:** auto
- Key:** A redacted password (represented by dots).

At the bottom of the screen are three buttons: 'Save & Apply' (highlighted in blue), 'Save', and 'Reset'.

Figure 63: The wireless security section

Web Field/UCI/Package Option	Description																		
Web: Encryption UCI: wireless.@wifi-iface[0].encryption Opt: encryption	<p>Method of encryption.</p> <table border="1"> <thead> <tr> <th>Web value</th><th>UCI value</th></tr> </thead> <tbody> <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> </tbody> </table>	Web value	UCI value	No encryption	none	WEP Open System	wep-open	WEP Shared Key	wep-shared	WPA-PSK	psk	WPA2-PSK	psk2	WPA-PSK/WPA2-PSK Mixed Mode	psk-mixed	WPA-EAP	wpa	WPA2-WAP	wpa2
Web value	UCI value																		
No encryption	none																		
WEP Open System	wep-open																		
WEP Shared Key	wep-shared																		
WPA-PSK	psk																		
WPA2-PSK	psk2																		
WPA-PSK/WPA2-PSK Mixed Mode	psk-mixed																		
WPA-EAP	wpa																		
WPA2-WAP	wpa2																		
Web: Cipher UCI: wireless.@wifi-iface[0].cipher= Opt: cipher	<p>Cipher type. Only displayed if WPA encryption modes are selected.</p> <table border="1"> <thead> <tr> <th>Web value</th><th>UCI</th></tr> </thead> <tbody> <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> </tbody> </table>	Web value	UCI	Auto	auto	Force CCMP (AES)	ccmp	Force TKIP	tkip	Force TKIP and CCMP	tkip+ccmp								
Web value	UCI																		
Auto	auto																		
Force CCMP (AES)	ccmp																		
Force TKIP	tkip																		
Force TKIP and CCMP	tkip+ccmp																		
Web: Key UCI: wireless.@wifi-iface[0].key Opt: key	Specifies the wireless key authentication phrase.																		
Web: Key #1 UCI: wireless.@wifi-iface[0].key1 Opt: key1	Specifies the first wireless key authentication phrase.																		
Web: Key #2 UCI: wireless.@wifi-iface[0].key2 Opt: key2	Specifies the second wireless key authentication phrase.																		
Web: Key #3 UCI: wireless.@wifi-iface[0].key3 Opt: key3	Specifies the third wireless key authentication phrase.																		
Web: Key #4 UCI: wireless.@wifi-iface[0].key4 Opt: key4	Specifies the fourth wireless key authentication phrase.																		
Web: Radius Authentication-Server UCI: wireless.@wifi-iface[0].auth_server Opt: auth_server	Defines the Radius server for EAP authentication.																		
Web: Radius Authentication-Port UCI: wireless.@wifi-iface[0].auth_port Opt: auth_port	Defines the Radius server port for EAP authentication.																		
Web: Radius Authentication-Secret UCI: wireless.@wifi-iface[0].auth_secret Opt: auth_secret	Defines the Radius server secret for EAP authentication.																		
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 32: Information table for the interface configuration wireless security section**13.2.2.3 Interface configuration: MAC filter**
Figure 64: The MAC filter section

Web Field/UCI/Package Option	Description		
Web: MAC-Address Filter UCI: wireless.@wifi-iface[0].macfilter Opt: macfilter	MAC address filtering process.		
Option	Description	UCI	
Disable	Disables MAC Address filter.	disable	
Allow listed only	Allows only the MAC address listed in the text field.	allow	
Allow all except listed	Allows everything but the MAC address listed in the text field.	deny	
Web: MAC -List UCI: wireless.@wifi-iface[0].maclist Opt: list maclist	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		

Table 33: Information table for interface configuration MAC filter section**13.3 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.

13.3.1 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.

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. Then configure the interface by following instructions in the chapter 'Configuring an Ethernet interface'. When you have completed those steps, continue with the section below.

13.3.2 AP mode on an existing Ethernet interface

Configure the WiFi 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. <table border="1"> <tr> <td>Empty</td><td></td></tr> <tr> <td>Bridge</td><td>Configures a bridge over multiple interfaces.</td></tr> </table>		Empty		Bridge	Configures a bridge over multiple interfaces.
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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>		0	Disabled.	1	Enabled.
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 34: Information table for the physical section on the common configuration page

13.4 Configuring WiFi using UCI

The configuration files are stored on:

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

13.4.1 AP modem on a new Ethernet interface using package options

```
root@GW_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@GW_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'
```

13.4.2 AP modem on a new Ethernet interface using UCI

```
root@GW_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@GW_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
```

13.4.3 AP mode on an existing Ethernet interface using packages options

```
root@GW_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@GW_router:~# uci export wireless
package wireless
```

```

config wifi-device 'radio0'
    option type 'mac80211'
    option channel '11'
    option phy 'phy0'
    option hemode '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'
```

13.4.4 AP mode on an existing Ethernet interface using UCI

```

root@GW_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@GW_router:~# uci show wireless
wireless.radio0=wifi-device
wireless.radio0.type=mac80211
wireless.radio0.channel=11
wireless.radio0.phy=phy0
wireless.radio0.hemode=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
```

13.5 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.

Interfaces - WCLIENT

On this page you can configure the network interfaces. You can bridge several interfaces by ticking the "bridge interfaces" field and enter the names of several network interfaces separated by spaces. You can also use VLAN notation INTERFACE.VLANNR (e.g.: eth0.1).

Common Configuration

Status: Unknown "VA-Wireless"

Protocol: DHCP client

Really switch protocol?

MAC Address: 00:00:00:00:00:00
RX: 0.00 B (0 Pkts.)
TX: 0.00 B (0 Pkts.)

IP-Aliases

This section contains no values yet

Figure 66: The client interface page

Web Field/UCI/Package Option	Description																											
Web: Protocol UCI: network. newwifiClient.proto Opt: proto	Specifies what protocol the interface will operate on. Select DHCP Client .																											
	<table border="1"> <thead> <tr> <th>Option</th><th>Description</th></tr> </thead> <tbody> <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> </tbody> </table>		Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.	IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.	GRE	Generic Routing Encapsulation protocol	IOT		L2TP	Layer 2 Tunnelling Protocol	PPP	Point to Point Protocol	PPPoE	PPP over Ethernet	PPPoATM	PPP over ATM	LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.
Option	Description																											
Static	Static configuration with fixed address and netmask.																											
DHCP Client	Address and netmask are assigned by DHCP.																											
Unmanaged	Unspecified																											
IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.																											
IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.																											
GRE	Generic Routing Encapsulation protocol																											
IOT																												
L2TP	Layer 2 Tunnelling Protocol																											
PPP	Point to Point Protocol																											
PPPoE	PPP over Ethernet																											
PPPoATM	PPP over ATM																											
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																											

Table 35: 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.

13.6 Configuring WiFi in Client mode using command line

The configuration files are stored on:

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

13.6.1 Client modem using package options

```
root@GW_router:~# uci export network
package network
config interface 'newwifiClient'
    option proto 'dhcp'

root@GW_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'

```

13.6.2 Client modem using UCI

```

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

```

13.6.2.1 uci show wireless

```

root@GW_router:~# uci show wireless
wireless.radio0=wifi-device
wireless.radio0.type=mac80211
wireless.radio0.channel=11
wireless.radio0.phy=phy0
wireless.radio0.hemode=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

```

14Configuring OSPF (Open Shortest Path First)

14.1 Introduction

OSPF is a standardised Link State routing protocol, designed to scale efficiently to support larger networks. Link State protocols track the status and connection type of each link and produce a calculated metric based on these and other factors, including some set by the network administrator. Link State protocols will take a path which has more hops, but that uses a faster medium over a path using a slower medium with fewer hops.

- OSPF adheres to the following Link State characteristics:
- OSPF employs a hierarchical network design using areas.
- OSPF will form neighbour relationships with adjacent routers in the same area.
- Instead of advertising the distance to connected networks, OSPF advertises the status of directly connected links using Link-State Advertisements (LSAs).
- OSPF sends updates (LSAs) when there is a change to one of its links, and will only send the change in the update. LSAs are additionally refreshed every 30 minutes.
- OSPF traffic is multicast either to address 224.0.0.5 (all OSPF routers) or 224.0.0.6 (all designated routers).
- OSPF uses the Dijkstra Shortest Path First algorithm to determine the shortest path.
- OSPF is a classless protocol, and therefore supports variable Length Subnet Masks (VLSMs).

Other characteristics of OSPF include:

- OSPF supports only IP routing.
- OSPF routes have an administrative distance is 110.
- OSPF uses cost as its metric, which is computed based on the bandwidth of the link. OSPF has no hop-count limit.

The OSPF process builds and maintains three separate tables:

- **A neighbour table** containing a list of all neighbouring routers
- **A topology table** containing a list of all possible routes to all known networks within an area
- **A routing table** containing the best route for each known network

14.1.1 OSPF areas

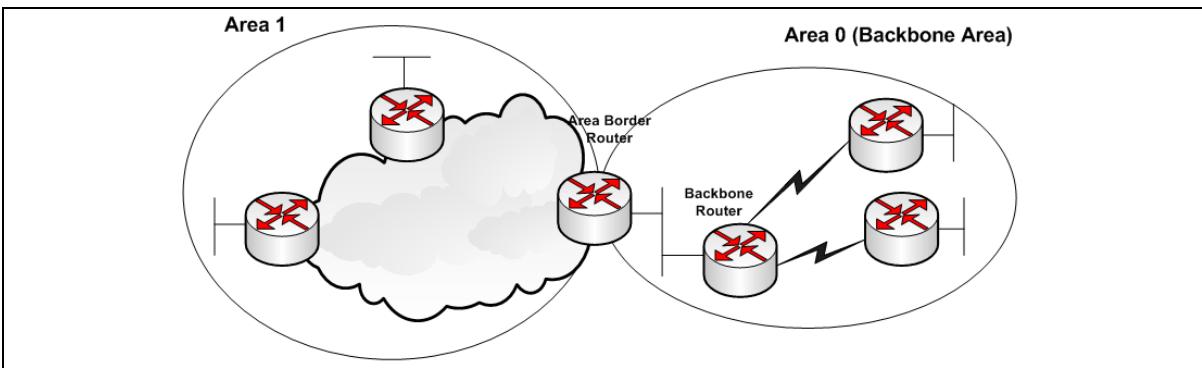


Figure 67: OSPF areas

OSPF has a number of features that allow it to scale well for larger networks. One of these features is OSPF areas. OSPF areas break up the topology so that routers in one area know less topology information about the subnets in the other area, and they do not know anything about the routers in the other area at all. With smaller topology databases, routers consume less memory and take less processing time to run SPF.

The Area Border Router (ABR) is the border between two areas. The ABR does not advertise full topology information about the part of the network in area 0 to routers in area 1. Instead the ABR advertises summary information about the subnets in area 0. Area 1 will just see a number of subnets reachable via area 0.

14.1.2 OSPF neighbours

OSPF forms neighbour relationships, called adjacencies, with other routers in the same Area by exchanging 'Hello' packets to multicast address 224.0.0.5. Only after an adjacency is formed can routers share routing information.

Each OSPF router is identified by a unique router ID. The router ID can be determined in one of three ways:

- The router ID can be manually specified.
- If not manually specified, the highest IP address configured on any Loopback interface on the router will become the router ID.
- If no loopback interface exists, the highest IP address configured on any physical interface will become the router ID.

By default, Hello packets are sent out OSPF-enabled interfaces every 10 seconds for broadcast and point-to-point interfaces, and 30 seconds for non-broadcast and point-to-multipoint interfaces.

OSPF also has a 'Dead Interval', which indicates how long a router will wait without hearing any hellos before announcing a neighbour as 'down'. The default setting for the Dead Interval is 40 seconds for broadcast and point-to-point interfaces, and 120 seconds for non-broadcast and point-to-multipoint interfaces. By default, the Dead Interval timer is four times the Hello interval.

OSPF routers will only become neighbours if the following parameters within a Hello packet are identical on each router:

-
- Area ID
 - Area Type (stub, NSSA, etc.)
 - Prefix
 - Subnet Mask
 - Hello Interval
 - Dead Interval
 - Network Type (broadcast, point-to-point, etc.)
 - Authentication

The Hello packets also serve as keepalives to allow routers to quickly discover if a neighbour is down. Hello packets also contain a neighbour field that lists the router IDs of all neighbours the router is connected to. A neighbour table is constructed from the OSPF Hello packets, which includes the following information:

- The router ID of each neighbouring router
- The current 'state' of each neighbouring router
- The interface directly connecting to each neighbour
- The IP address of the remote interface of each neighbour

14.1.3 OSPF designated routers

In multi-access networks such as Ethernet, there is the possibility of many neighbour relationships on the same physical segment. This leads to a considerable amount of unnecessary Link State Advertisement (LSA) traffic. If a link of a router were to fail, it would flood this information to all neighbours. Each neighbour, in turn, would then flood that same information to all other neighbours. This is a waste of bandwidth and processor load.

To prevent this, OSPF will elect a Designated Router (DR) for each multi-access networks, accessed via multicast address 224.0.0.6. For redundancy purposes, a Backup Designated Router (BDR) is also elected.

OSPF routers will form adjacencies with the DR and BDR. If a change occurs to a link, the update is forwarded only to the DR, which then forwards it to all other routers. This greatly reduces the flooding of LSAs. DR and BDR elections are determined by a router's OSPF priority, which is configured on a per-interface basis (a router can have interfaces in multiple multi-access networks). The router with the highest priority becomes the DR; second highest becomes the BDR. If there is a tie in priority, whichever router has the highest Router ID will become the DR.

14.1.4 OSPF neighbour states

Neighbour adjacencies will progress through several states, described in the table below.

State	Description
Down	Indicates that no Hellos have been heard from the neighbouring router
Init	Indicates a Hello packet has been heard from the neighbour, but two-way communication has not yet been initialized.

2-Way	Indicates that bidirectional communication has been established. Recall that Hello packets contain a neighbour field. Thus, communication is considered 2-Way once a router sees its own Router ID in its neighbour's Hello Packet. Designated and Backup Designated Routers are elected at this stage.						
ExStart	Indicates that the routers are preparing to share link state information. Master/slave relationships are formed between routers to determine who will begin the exchange.						
Exchange	Indicates that the routers are exchanging Database Descriptors (DBDs). DBDs contain a description of the router's Topology Database. A router will examine a neighbour's DBD to determine if it has information to share.						
Loading	Indicates the routers are finally exchanging Link State Advertisements, containing information about all links connected to each router. Essentially, routers are sharing their topology tables with each other.						
Full	Indicates that the routers are fully synchronized. The topology table of all routers in the area should now be identical. Depending on the role of the neighbour, the state may appear as: <table border="1" style="margin-left: 20px;"> <tr> <td>Full/DR</td> <td>Indicating that the neighbour is a Designated Router (DR)</td> </tr> <tr> <td>Full/BDR</td> <td>Indicating that the neighbour is a Backup Designated Router (BDR)</td> </tr> <tr> <td>Full/DROther</td> <td>Indicating that the neighbour is neither the DR nor BDR. On a multi-access network, OSPF routers will only form Full adjacencies with DRs and BDRs. Non-DRs and non-BDRs will still form adjacencies, but will remain in a 2-Way State. This is normal OSPF behaviour.</td> </tr> </table>	Full/DR	Indicating that the neighbour is a Designated Router (DR)	Full/BDR	Indicating that the neighbour is a Backup Designated Router (BDR)	Full/DROther	Indicating that the neighbour is neither the DR nor BDR. On a multi-access network, OSPF routers will only form Full adjacencies with DRs and BDRs. Non-DRs and non-BDRs will still form adjacencies, but will remain in a 2-Way State. This is normal OSPF behaviour.
Full/DR	Indicating that the neighbour is a Designated Router (DR)						
Full/BDR	Indicating that the neighbour is a Backup Designated Router (BDR)						
Full/DROther	Indicating that the neighbour is neither the DR nor BDR. On a multi-access network, OSPF routers will only form Full adjacencies with DRs and BDRs. Non-DRs and non-BDRs will still form adjacencies, but will remain in a 2-Way State. This is normal OSPF behaviour.						

Table 36: Neighbour adjacency states

14.1.5 OSPF network types

OSPF's functionality is different across several different network topology types.

State	Description
Broadcast Multi-Access	Indicates a topology where broadcast occurs. Examples include Ethernet, Token Ring and ATM. OSPF characteristics are: OSPF will elect DRs and BDRs Traffic to DRs and BDRs is multicast to 224.0.0.6. Traffic from DRs and BDRs to other routers is multicast to 224.0.0.5 Neighbours do not need to be manually specified.
Point-to-Point	Indicates a topology where two routers are directly connected. An example would be a point-to-point T1. OSPF characteristics are: OSPF will not elect DRs and BDRs All OSPF traffic is multicast to 224.0.0.5 Neighbours do not need to be manually specified
Point-to-Multipoint	Indicates a topology where one interface can connect to multiple destinations. Each connection between a source and destination is treated as a point-to-point link. An example would be point to Point-to-Multipoint Frame Relay. OSPF characteristics are: OSPF will not elect DRs and BDRs. All OSPF traffic is multicast to 224.0.0.5. Neighbours do not need to be manually specified.
Non-broadcast Multi-access Network (NBMA)	Indicates a topology where one interface can connect to multiple destinations; however, broadcasts cannot be sent across a NBMA network. An example would be Frame Relay. OSPF characteristics are: OSPF will elect DRs and BDRs. OSPF neighbours must be manually defined, thus All OSPF traffic is unicast instead of multicast. Note: on non-broadcast networks, neighbours must be manually specified, as multicast Hello's are not allowed

Table 37: OSPF functionality over different topology types

14.1.6 The OSPF hierarchy

OSPF is a hierarchical system that separates an autonomous system into individual areas. OSPF traffic can either be:

- intra-area (within one area),
- inter-area (between separate areas), or
- external (from another AS).

OSPF routers build a topology database of all links within their area, and all routers within an area will have an identical topology database. Routing updates between these routers will only contain information about links local to their area. Limiting the topology database to include only the local area conserves bandwidth and reduces CPU loads.

Area 0 is required for OSPF to function, and is considered the backbone area. As a rule, all other areas must have a connection into area 0, though this rule can be bypassed using virtual links. Area 0 is often referred to as the transit area to connect all other areas.

OSPF routers can belong to multiple areas, and therefore contain separate topology databases for each area. These routers are known as Area Border Routers (ABRs).

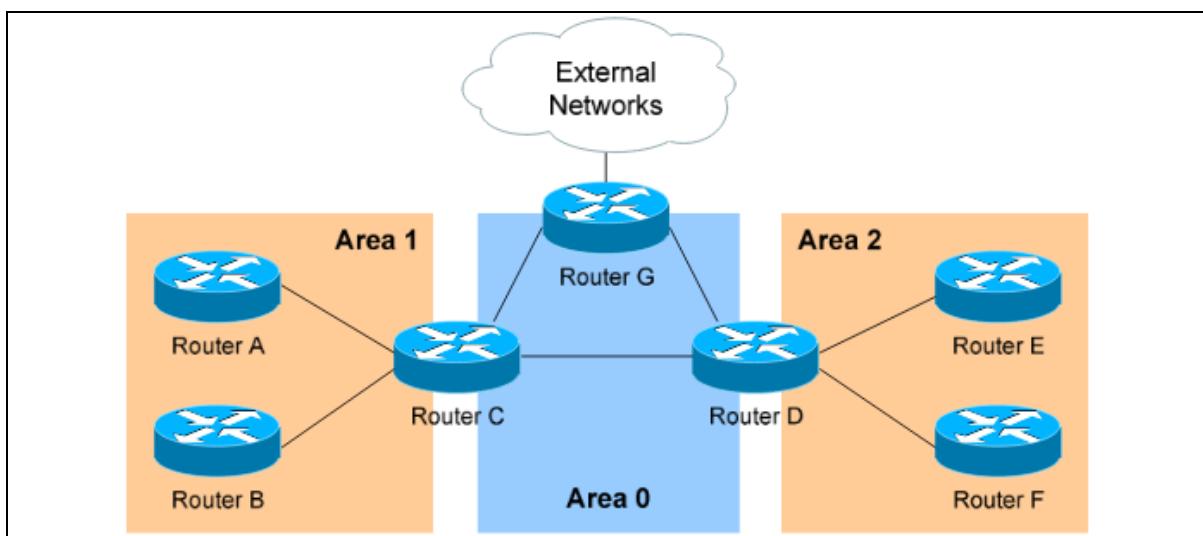


Figure 68: OSPF hierarchy

In the above example three areas exist: Area 0, Area 1, and Area 2.

Area 0 is the backbone area for this autonomous system.

Both Area 1 and Area 2 must directly connect to Area 0. Routers A and B belong fully to Area 1, while Routers E and F belong fully to Area 2. These are known as Internal Routers.

Router C belongs to both Area 0 and Area 1; so it is an ABR. Because it has an interface in Area 0, it can also be considered a Backbone Router (BR). The same can be said for Router D, as it belongs to both Area 0 and Area 2.

Router G also belongs to Area 0 however it also has a connection to the internet, which is outside this autonomous system. This makes Router G an Autonomous System Border Router (ASBR).

A router can become an ASBR in one of two ways:

- By connecting to a separate Autonomous System, such as the internet
- By redistributing another routing protocol into the OSPF process.

ASBRs provide access to external networks. OSPF defines two types of external routes, as shown in the table below.

Type 2 (E2)	Includes only the external cost to the destination network. External cost is the metric being advertised from outside the OSPF domain. This is the default type assigned to external routes.
Type 1 (E1)	Includes both the external cost, and the internal cost to reach the ASBR, to determine the total metric to reach the destination network. Type 1 routes are always preferred over Type 2 routes to the same destination.

Table 38: Types of external routes

14.1.7 OSPF router types

The four separate OSPF router types are shown in the table below.

Route Type	Description
Internal Router	All router interfaces belong to only one area.
Area Border Router (ABR)	Have interfaces in at least two separate areas.
Backbone Router	Have at least one interface in area 0.
Autonomous System Border Router (ABR)	Have a connection to a separate autonomous system.

14.2 Configuration package used

Package	Sections
ospfd	routing network interface

14.3 Configuring OSPF using the web interface

Select **Network -> OSPF**. The OSPF page appears.

There are three sections in the OSPF page:

Section	Description
Global Settings	Enables OSPF and configures the OSPF routing section containing global configuration parameters. The web automatically names the routing section 'ospfd'
Topology Configuration	Configures the network sections.
Interfaces Configuration	Configures the interface sections. Defines interface configuration for OSPF and interface specific parameters

14.3.1 Global settings

The Global Settings section configures the ospfd routing section. The web automatically names the routing section 'ospfd'.

The screenshot shows the 'OSPF' configuration page. At the top left is the title 'OSPF'. Below it is the section 'Global Settings'. There are three main configuration items:

- 'OSPF Enabled' with a checked checkbox.
- 'Router ID' with an empty input field and a note: 'IP address format, must be unique, if blank it generates Router ID automatically'.
- 'Make Default Router' with a checked checkbox.

Figure 69: The OSPF global settings configuration page

Web Field/UCI/Package Option	Description				
Web: OSPF Enabled UCI: ospfd.ospfd.enabled Opt: enabled	Enables OSPF advertisements on router. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Router ID UCI: ospfd.ospfd.router_id Opt: router_id	This sets the Router ID of the OSPF process. The Router ID may be an IP address of the router, but need not be - it can be any arbitrary 32bit number. However it MUST be unique within the entire OSPF domain to the OSPF speaker. If one is not specified, then ospfd will obtain a router-ID automatically from the zebra daemon. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>Empty</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	Empty		Range	
Empty					
Range					
Web: Make Default Router UCI: ospfd.ospfd.default_info_originate Opt: default_info_originate	Defines whether to originate an AS-External (type-5) LSA describing a default route into all external-routing capable areas, of the specified metric and metric type. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: n/a UCI: ospfd.ospfd.vty_enabled Opt: vty_enabled	Enable vty for OSPFd (telnet to localhost:2604)				

Table 39: Information table for OSPF global settings

14.3.2 Topology configuration

The Topology section configures the ospfd network section. This section specifies the OSPF enabled interface(s). The router can provide network information to the other OSPF routers via this interface.

Note: to advertise OSPF on an interface, the network mask prefix length for the topology configuration statement for the desired interface advertisement must be equal or smaller (IE. larger network) than the network mask prefix length for the interface.

For example, the topology configuration statement in the screenshot below does not enable OSPF on an interface with address 12.1.1.1/23, but it would on an interface with address 12.1.1.129/25.

Topology Configuration			
Network	Mask Length	Area	Stub Area
<i>Only for non-backbone areas</i>			
12.1.1.1	24	0	<input checked="" type="checkbox"/>
<input type="button" value="Add"/>			

Figure 70: The OSPF Topology configuration page

Web Field/UCI/Package Option	Description				
Web: Network UCI: ospfd.@network[0].ip_addr Opt: ip_addr	Specify the IP address for OSPF enabled interface. Format: A.B.C.D				
Web: Mask Length UCI: ospfd.@network[0].mask_length Opt: mask_length	Specify the mask length for OSPF enabled interface. The mask length should be entered in CIDR notation.				
Web: Area UCI: ospfd.@network[0].area Opt: area	Specify the area number for OSPF enabled interface.				
Web: Stub Area UCI: ospfd.@network[0].stub_area Opt: stub_area	<p>Only for non-backbone areas.</p> <p>Configure the area to be a stub area. That is, an area where no router originates routes external to OSPF and hence an area where all external routes are via the ABR(s).</p> <p>ABRs for such an area do not need to pass AS-External LSAs (type-5s) or ASBR-Summary LSAs (type-4) into the area. They need only pass Network-Summary (type-3) LSAs into such an area, along with a default-route summary.</p> <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

Table 40: Information table for OSPF topology configuration

14.3.3 Interfaces configuration

The Interfaces section contains settings to configure the ospfd interface. It defines interface configuration for OSPF and interface specific parameters.

OSPFv2 allows packets to be authenticated using either an insecure plain text password, included with the packet, or by a more secure MD5 based HMAC (keyed-Hashing for Message AuthentiCation). Enabling authentication prevents routes being updated by unauthenticated remote routers, but still can allow routes, that is, the entire OSPF routing table, to be queried remotely, potentially by anyone on the internet, via OSPFv1.

This section defines key_chains to be used for MD5 authentication

Interfaces Configuration

Interface <input type="radio"/> PPPoE:  <input type="radio"/> l2tpun:  <input type="radio"/> lan:  <input type="radio"/> lan2:  <input type="radio"/> lan3:  <input type="radio"/> lan4:  <input type="radio"/> lan5:  <input type="radio"/> lan6:  <input type="radio"/> lan7:  <input checked="" type="radio"/> lan8:  <input type="radio"/> loopback:  <input type="radio"/> wlan100:  <input type="radio"/> wan: 
Network Type <input type="text" value="broadcast"/> <small>Leave as default if unknown. Default depends on the type of interface</small>
<input checked="" type="checkbox"/> Passive
Hello Interval * <input type="text" value="10"/> <small>Defaults: broadcast/point-to-point 10 secs, non-broadcast/point-to-multipoint 30 secs</small>
Dead Interval * <input type="text" value="40"/> <small>Defaults: broadcast/point-to-point 40 secs, non-broadcast/point-to-multipoint 120 secs</small>
Authentication <input type="text" value="text"/>
Text Auth. Key <input type="text" value="secret"/>

Figure 71: The OSPF Interfaces configuration section

Web Field/UCI/Package Option	Description										
Web: Interface UCI: ospfd.@interface[0].ospf_interface Opt: ospf_interface	Defines the interface name										
Web: Network Type UCI: ospfd.@interface[0].network_type Opt: network_type	<p>Defines network type for specified interface.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;">Default</td> <td style="padding: 5px;">Autodetect – it will be broadcast. If broadcast is not supported on that interface then point-to-point.</td> </tr> <tr> <td style="padding: 5px;">broadcast</td> <td style="padding: 5px;"></td> </tr> <tr> <td style="padding: 5px;">non-broadcast</td> <td style="padding: 5px;"></td> </tr> <tr> <td style="padding: 5px;">point-to-point</td> <td style="padding: 5px;"></td> </tr> <tr> <td style="padding: 5px;">point-to-multipoint</td> <td style="padding: 5px;"></td> </tr> </table>	Default	Autodetect – it will be broadcast. If broadcast is not supported on that interface then point-to-point.	broadcast		non-broadcast		point-to-point		point-to-multipoint	
Default	Autodetect – it will be broadcast. If broadcast is not supported on that interface then point-to-point.										
broadcast											
non-broadcast											
point-to-point											
point-to-multipoint											
Web: Passive UCI: ospfd.@interface[0].passive Opt: passive	<p>Do not send hello packets on the given interface, but do advertise the interface as a stub link in the router-LSA (Link State Advertisement) for this router.</p> <p>This allows you to advertise addresses on such connected interfaces without having to originate AS-External/Type-5 LSAs (which have global flooding scope) as would occur if connected addresses were redistributed into OSPF. This is the only way to advertise non-OSPF links into stub areas.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25px; padding: 5px;">0</td> <td style="width: 75px; padding: 5px;">Disabled.</td> </tr> <tr> <td style="width: 25px; padding: 5px;">1</td> <td style="width: 75px; padding: 5px;">Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.						
0	Disabled.										
1	Enabled.										

Web: Hello Interval UCI: ospfd.@interface[0].hello_interval Opt: hello_interval	Defines the number of seconds for the Hello Interval timer value. A Hello packet will be sent every timer value seconds on the specified interface. This value must be the same for all routers attached to a common network. The default is every 10 seconds for broadcast and point-to-point interfaces, and 30 seconds for non-broadcast and point-to-multipoint interfaces.						
	<table border="1"> <tr> <td>10</td><td>10 seconds</td></tr> <tr> <td>Range</td><td>.</td></tr> </table>	10	10 seconds	Range	.		
10	10 seconds						
Range	.						
Web: Dead Interval UCI: ospfd.@interface[0].dead_interval Opt: dead_interval	Defines the number of seconds for the Dead Interval timer value used for Wait Timer and Inactivity Timer. This value must be the same for all routers attached to a common network. The default is 40 seconds for broadcast and point-to-point interfaces, and 120 seconds for non-broadcast and point-to-multipoint interfaces. By default, the Dead Interval timer is four times the Hello interval.						
	<table border="1"> <tr> <td>40</td><td>40 seconds</td></tr> <tr> <td>Range</td><td>.</td></tr> </table>	40	40 seconds	Range	.		
40	40 seconds						
Range	.						
Web: Authentication UCI: ospfd.@interface[0].auth_mode Opt: auth_mode	OSPFv2 (only) allows packets to be authenticated via either an insecure plain text password, included with the packet, or via a more secure MD5 based HMAC (keyed-Hashing for Message AuthentiCation). Enabling authentication prevents routes being updated by unauthenticated remote routers, but still can allow routes (i.e. the entire OSPF routing table) to be queried remotely, potentially by anyone on the internet, via OSPFv1.						
	<table border="1"> <tr> <td>no</td><td>Default value. No authentication.</td></tr> <tr> <td>md5</td><td>Set the interface with OSPF MD5 authentication</td></tr> <tr> <td>text</td><td>Set the interface with OSPF simple password authentication.</td></tr> </table>	no	Default value. No authentication.	md5	Set the interface with OSPF MD5 authentication	text	Set the interface with OSPF simple password authentication.
no	Default value. No authentication.						
md5	Set the interface with OSPF MD5 authentication						
text	Set the interface with OSPF simple password authentication.						
Web: Text Auth. Key UCI: ospfd.@interface[0].text_auth_key Opt: text_auth_key	This command sets authentication string for text authentication. text_auth_key option can have length up to 8 characters. Displayed only when Authentication is set to text.						
Web: Key ID UCI: ospfd.@interface[0].key_id Opt: key_id	Specifies key ID. Must be unique and match at both ends. Displayed only when Authentication is set to MD5.						
Web: MD5 Auth. Key UCI: ospfd.@interface[0].md5_auth_key Opt: md5_auth_key	Specify Keyed MD5 chain. Displayed only when Authentication is set to MD5.						

Table 41: Information table for OSPF Interface commands

14.4 Configuring OSPF using the command line

OSPF is configured under the ospfd package /etc/config/ospfd.

There are three config sections: ospfd, interface and network.

You can configure multiple interface and network sections.

By default, all OSPF interface instances are named interface, instances are identified by @interface then the interface position in the package as a number. For example, for the first interface in the package using UCI:

```
ospfd.@interface[0]=interface
ospfd.@interface[0].ospf_interface=lan
```

Or using package options:

```
config interface
    option ospf_interface 'lan'
```

By default, all OSPF network instances are named network, it is identified by @network then the interface position in the package as a number. For example, for the first network in the package using UCI:

```
ospfd.@network[0]=network
ospfd.@network[0].ip_addr=12.1.1.1
```

Or using package options:

```
config network
    option ip_addr '12.1.1.1'
```

14.5 OSPF using UCI

```
root@GW_router:~# uci show ospfd
ospfd.ospfd=routing
ospfd.ospfd.enabled=yes
ospfd.ospfd.default_info_originate=yes
ospfd.ospfd.router_id=1.2.3.4
ospfd.@network[0]=network
ospfd.@network[0].ip_addr=12.1.1.1
ospfd.@network[0].mask_length=24
ospfd.@network[0].area=0
ospfd.@network[0].stub_area=yes
ospfd.@interface[0]=interface
ospfd.@interface[0].ospf_interface=lan8
ospfd.@interface[0].hello_interval=10
ospfd.@interface[0].dead_interval=40
ospfd.@interface[0].network_type=broadcast
ospfd.@interface[0].passive=yes
ospfd.@interface[0].auth_mode=text
ospfd.@interface[0].text_auth_key=secret
ospfd.@interface[1]=interface
ospfd.@interface[1].ospf_interface=lan7
ospfd.@interface[1].network_type=point-to-point
ospfd.@interface[1].passive=no
```

```
ospfd.@interface[1].hello_interval=30
ospfd.@interface[1].dead_interval=120
ospfd.@interface[1].auth_mode=md5
ospfd.@interface[1].key_id=1
ospfd.@interface[1].md5_auth_key=test
```

14.6 OSPF using package options

```
root@GW_router:~# uci export ospfd
package ospfd

config routing 'ospfd'
    option enabled 'yes'
    option default_info_originate 'yes'
    option router_id '1.2.3.4'

config network
    option ip_addr '12.1.1.1'
    option mask_length '24'
    option area '0'
    option stub_area 'yes'

config interface
    option ospf_interface 'lan8'
    option hello_interval '10'
    option dead_interval '40'
    option network_type 'broadcast'
    option passive 'yes'
    option auth_mode 'text'
    option text_auth_key 'secret'

config interface
    option ospf_interface 'lan7'
    option network_type 'point-to-point'
    option passive 'no'
    option hello_interval '30'
    option dead_interval '120'
```

```

option auth_mode 'md5'
option key_id '1'
option md5_auth_key 'test'

```

14.7 OSPF diagnostics

14.7.1 Route status

To show the current routing status, enter:

```

root@GW_router:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref    Use
Iface
0.0.0.0         10.206.4.65   0.0.0.0       UG      1      0      0
usb0
10.1.0.0        0.0.0.0       255.255.0.0   U       0      0      0
eth1
10.206.4.64     0.0.0.0       255.255.255.252 U      0      0      0
usb0
11.11.11.0      0.0.0.0       255.255.255.248 U      0      0      0
gre-GRE
89.101.154.151  10.206.4.65   255.255.255.255 UGH     0      0      0
usb0
192.168.100.0   0.0.0.0       255.255.255.0   U      0      0      0
eth0
192.168.101.1   11.11.11.1    255.255.255.255 UGH     11     0      0
gre-GRE
192.168.104.1   11.11.11.4    255.255.255.255 UGH     20     0      0
gre-GRE

```

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

14.7.2 Tracing OSPF packets

Typically, OSPF uses IP as its transport protocol. The well-known IP protocol type for OSPF traffic is 0x59. To trace OSPF packets on any interface on the router, enter:
`tcpdump -i any -n proto ospf &`

```

root@GW_router:~# tcpdump -i any -n proto ospf &
root@GW_router:~# tcpdump: verbose output suppressed, use -v or -vv for
full protocol decode

```

```
listening on any, link-type LINUX_SLL (Linux cooked), capture size 65535
bytes
```

To stop tracing enter `fg` to bring tracing task to foreground, and then <**CTRL-C**> to stop the trace.

```
root@GW_router:~# fg
tcpdump -i any -n proto ospf
^C
33 packets captured
33 packets received by filter
0 packets dropped by kernel
```

14.8 Quagga/Zebra console

Quagga is the routing protocol suite embedded in the router firmware. Quagga is split into different daemons for implementation of each routing protocol. Zebra is a core daemon for Quagga, providing the communication layer to the underlying Linux kernel, and routing updates to the client daemons.

Quagga has a console interface to Zebra for advanced debugging of the routing protocols.

To access, enter:

```
root@GW_router:~# telnet localhost zebra

Entering character mode
Escape character is '^]'.

Hello, this is Quagga (version 0.99.21).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:
```

To see OSPF routing from Zebra console, enter:

```
root@GW_router:~# sh ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
```

```

O - OSPF, I - IS-IS, B - BGP, P - PIM, H - HSLS, o - OLSR,
b - BATMAN, A - Babel,
> - selected route, * - FIB route

K>* 0.0.0.0/0 via 10.206.4.65, usb0
O 10.1.0.0/16 [110/11] via 11.11.11.1, gre-GRE, 02:35:28
C>* 10.1.0.0/16 is directly connected, eth1
C>* 10.206.4.64/30 is directly connected, usb0
O 11.11.11.0/29 [110/10] is directly connected, gre-GRE, 02:35:29
C>* 11.11.11.0/29 is directly connected, gre-GRE
K>* 89.101.154.151/32 via 10.206.4.65, usb0
C>* 127.0.0.0/8 is directly connected, lo
C>* 192.168.100.0/24 is directly connected, eth0
O>* 192.168.101.1/32 [110/11] via 11.11.11.1, gre-GRE, 02:35:28
O>* 192.168.104.1/32 [110/20] via 11.11.11.4, gre-GRE, 02:30:45
O 192.168.105.1/32 [110/10] is directly connected, lo, 02:47:52
C>* 192.168.105.1/32 is directly connected, lo

```

14.8.1 OSPF debug console

When option `tty_enabled` (see Global settings section above) is enabled in the OSPF configuration, OSPF debug console can be accessed for advanced OSPF debugging.

To access OSPF debug console enter: `telnet localhost ospfd` (password zebra)

```

root@GW_router:~# telnet localhost ospfd

Entering character mode
Escape character is '^]'.

Hello, this is Quagga (version 0.99.21).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:

```

To see OSPF routing from OSPF debug console, enter: `sh ip ospf route`

```

UUT> sh ip ospf route
=====
OSPF network routing table =====
N   10.1.0.0/16          [11] area: 0.0.0.0
                                         via 11.11.11.1, gre-GRE
N   11.11.11.0/29          [10] area: 0.0.0.0
                                         directly attached to gre-GRE
N   192.168.101.1/32        [11] area: 0.0.0.0
                                         via 11.11.11.1, gre-GRE
N   192.168.104.1/32        [20] area: 0.0.0.0
                                         via 11.11.11.4, gre-GRE
N   192.168.105.1/32        [10] area: 0.0.0.0
                                         directly attached to lo

=====
OSPF router routing table =====

=====
OSPF external routing table =====

```

To see OSPF neighbours from OSPF debug console, enter: `sh ip ospf neighbour`

```

root@GW_router:~# sh ip ospf neighbor

      Neighbor ID Pri State             Dead Time Address           Interface
RXmtL RqstL DBsmL
1.1.1.1          255 Full/DR         33.961s 11.11.11.1       gre-
GRE:11.11.11.5      0    0    0

```

To see OSPF interface details from OSPF debug console, enter: `sh ip ospf interface`

```

root@GW_router:~# sh ip ospf interface
base0 is up
  ifindex 8, MTU 1518 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  OSPF not enabled on this interface
eth0 is up
  ifindex 9, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  OSPF not enabled on this interface
eth1 is up
  ifindex 10, MTU 1500 bytes, BW 0 Kbit
<UP,BROADCAST,RUNNING,PROMISC,MULTICAST>

```

```
OSPF not enabled on this interface
eth2 is down
    ifindex 11, MTU 1500 bytes, BW 0 Kbit <BROADCAST,MULTICAST>
    OSPF not enabled on this interface
eth3 is down
    ifindex 12, MTU 1500 bytes, BW 0 Kbit <BROADCAST,MULTICAST>
    OSPF not enabled on this interface
eth4 is down
    ifindex 13, MTU 1500 bytes, BW 0 Kbit <BROADCAST,MULTICAST>
    OSPF not enabled on this interface
eth5 is down
    ifindex 14, MTU 1500 bytes, BW 0 Kbit <BROADCAST,MULTICAST>
    OSPF not enabled on this interface
eth6 is down
    ifindex 15, MTU 1500 bytes, BW 0 Kbit <BROADCAST,MULTICAST>
    OSPF not enabled on this interface
eth7 is down
    ifindex 16, MTU 1500 bytes, BW 0 Kbit <BROADCAST,MULTICAST>
    OSPF not enabled on this interface
gre-GRE is up
    ifindex 19, MTU 1472 bytes, BW 0 Kbit <UP,RUNNING,MULTICAST>
    Internet Address 11.11.11.5/29, Area 0.0.0.0
    MTU mismatch detection:enabled
    Router ID 192.168.105.1, Network Type BROADCAST, Cost: 10
    Transmit Delay is 1 sec, State Backup, Priority 1
    Designated Router (ID) 1.1.1.1, Interface Address 11.11.11.1
    Backup Designated Router (ID) 192.168.105.1, Interface Address 11.11.11.5
    Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
    Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
        Hello due in 3.334s
    Neighbor Count is 1, Adjacent neighbor count is 1
gre0 is down
    ifindex 6, MTU 1476 bytes, BW 0 Kbit <NOARP>
    OSPF not enabled on this interface
ifb0 is down
    ifindex 2, MTU 1500 bytes, BW 0 Kbit <BROADCAST,NOARP>
    OSPF not enabled on this interface
```

```

ifb1 is down
  ifindex 3, MTU 1500 bytes, BW 0 Kbit <BROADCAST,NOARP>
  OSPF not enabled on this interface

lo is up
  ifindex 1, MTU 16436 bytes, BW 0 Kbit <UP,LOOPBACK,RUNNING>
  Internet Address 192.168.105.1/32, Broadcast 192.168.105.1, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 192.168.105.1, Network Type LOOPBACK, Cost: 10
  Transmit Delay is 1 sec, State Loopback, Priority 1
  No designated router on this network
  No backup designated router on this network
  Multicast group memberships: <None>
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in inactive
  Neighbor Count is 0, Adjacent neighbor count is 0

sit0 is down
  ifindex 7, MTU 1480 bytes, BW 0 Kbit <NOARP>
  OSPF not enabled on this interface

teql0 is down
  ifindex 4, MTU 1500 bytes, BW 0 Kbit <NOARP>
  OSPF not enabled on this interface

tunl0 is down
  ifindex 5, MTU 1480 bytes, BW 0 Kbit <NOARP>
  OSPF not enabled on this interface

usb0 is up
  ifindex 17, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  OSPF not enabled on this interface

```

To see OSPF database details from OSPF debug console, enter: `sh ip ospf database`

```

root@GW_router:~# sh ip ospf database

      OSPF Router with ID (192.168.105.1)

      Router Link States (Area 0.0.0.0)

      Link ID          ADV Router        Age   Seq#        CkSum  Link count

```

1.1.1.1	1.1.1.1	873	0x80006236	0xd591	3
192.168.104.1	192.168.104.1	596	0x8000000a	0x3a2d	2
192.168.105.1	192.168.105.1	879	0x8000000b	0x4919	2

Net Link States (Area 0.0.0.0)

Link ID	ADV Router	Age	Seq#	CkSum
11.11.11.1	1.1.1.1	595	0x80000004	0x5712

15 Configuring a mobile connection

15.1 Configuration package used

Package	Sections
network	interface

15.2 Configuring a mobile connection using the web interface

Note: if you are creating multiple mobile interfaces, simply repeat the steps in this chapter for each interface. Multiple interfaces are required for dual SIM or multiple radio module scenarios. Configuring static routes and/or Multi-WAN can be used to manage these interfaces.

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

15.2.1 Create a new mobile interface

To create a new mobile interface, in the Interface Overview section, click **Add new interface**. The Create Interface page appears. In the examples below, 3G has been used for the interface name.

The screenshot shows the 'Create Interface' page with the following fields:

- Name of the new interface: A text input field with a note: "The allowed characters are: A-Z, a-z, 0-9 and _".
- Protocol of the new interface: A dropdown menu set to "Static address".
- Create a bridge over multiple interfaces: A checkbox that is unchecked.
- Cover the following interface: A list of radio buttons next to interface names:
 - Ethernet Adapter: "eth0" (lan)
 - Ethernet Adapter: "eth1" (lan1)
 - Ethernet Adapter: "eth2"
 - Ethernet Adapter: "eth3"
 - Ethernet Adapter: "eth4"
 - Ethernet Adapter: "lo" (loopback)
 - Ethernet Adapter: "teq10"
 - Ethernet Adapter: "tunl0"
 - Custom Interface: An input field.
- A note at the bottom: "Note: If you choose an interface here which is part of another network, it will be moved into this network."

At the bottom are two buttons: "Back to Overview" and "Submit".

Figure 72: 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 _																										
Web: Protocol of the new interface UCI: network.3G.proto Opt: proto	<p>Protocol type. Select LTE/UMTS/GPRS/EV-DO.</p> <table border="1"> <thead> <tr> <th>Option</th> <th>Description</th> </tr> </thead> <tbody> <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> </tbody> </table>	Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	IPv6-in-IPv4		IPv6-over-IPv4		GRE		IOT		L2TP	Layer 2 Tunnelling Protocol.	PPP		PPPoE		PPPoATM		LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.
Option	Description																										
Static	Static configuration with fixed address and netmask.																										
DHCP Client	Address and netmask are assigned by DHCP.																										
Unmanaged	Unspecified																										
IPv6-in-IPv4																											
IPv6-over-IPv4																											
GRE																											
IOT																											
L2TP	Layer 2 Tunnelling Protocol.																										
PPP																											
PPPoE																											
PPPoATM																											
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																										
Web: Create a bridge over multiple interfaces UCI: network.3G.type Opt: type	<p>Enables bridge between two interfaces. Not relevant when configuring a mobile interface.</p> <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.																						
0	Disabled.																										
1	Enabled.																										
Web: Cover the following interface UCI: network.3G.ifname Opt: ifname	Select interfaces for bridge connection. Not relevant when configuring a mobile interface.																										

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	Set up more in-depth features such as initialization timeout, LCP echo failure thresholds and inactivity timeouts.
Firewall settings	Assign a firewall zone to the connection.

15.2.1.1 Mobile interface: general setup

Common Configuration

General Setup	Advanced Settings	Firewall Settings
Status 3g-3G		RX: 0.00 B (0 Pkts.) TX: 0.00 B (0 Pkts.)
Protocol	LTE/UMTS/GPRS/EV-DO	
Service Type	Auto (LTE/UMTS/GPRS)	
SIM	auto	
Operator PLMN code		<small>Specify this if you want to force connection to particular carrier</small>
APN		
APN username		
APN password		

Figure 73: The common configuration page

Web Field/UCI/Package Option	Description																							
Web:Status UCI: n/a Opt: n/a	Shows the current status of the interface.																							
Web: Protocol UCI: network.3G.proto Opt: proto	Protocol type. Select LTE/UMTS/GPRS/EV-DO . <table border="1"> <thead> <tr> <th>Option</th> <th>Description</th> </tr> </thead> <tbody> <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> </tbody> </table>		Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	GRE		IOT		L2TP	Layer 2 Tunnelling Protocol.	PPP		PPPoE		PPPoATM		LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.
Option	Description																							
Static	Static configuration with fixed address and netmask.																							
DHCP Client	Address and netmask are assigned by DHCP.																							
Unmanaged	Unspecified																							
GRE																								
IOT																								
L2TP	Layer 2 Tunnelling Protocol.																							
PPP																								
PPPoE																								
PPPoATM																								
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																							
Web: Service Type UCI: network.3G.service Opt: service	Service type that will be used to connect to the network. <table border="1"> <tbody> <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> </tbody> </table>		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.														
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: Operator PLMN code UCI: network.3G.operator Opt: operator	Specifies an operator PLMN code to force the connection to a particular carrier. The PLMN code is identified as a combination of the MCC and the MNC. Note: the operator option is used in conjunction with the operator format option <code>option opformat</code> which is used to define how the operator string is parsed. If configuring via the web GUI the opformat is automatically set to '2' to indicate it is a PLMN code. See below for alternate options for the operator format option.						
Web: n/a UCI: network.3G.opformat Opt: opformat	Defines the operator format. We recommended you use PLMN code. The operator is case sensitive so if using long or short character format it must match the operator exactly. To see the current operator using SSH enter the command: cat /var/state/mobile or using the web mobile stats page at Status -> Mobile Stats . <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Long character format</td> </tr> <tr> <td>1</td> <td>Short character format</td> </tr> <tr> <td>2</td> <td>PLMN code</td> </tr> </table>	0	Long character format	1	Short character format	2	PLMN code
0	Long character format						
1	Short character format						
2	PLMN code						
Web: SIM UCI: network.3G.sim Opt: sim	Defines which SIM is used on this interface. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>1</td> <td>SIM 1</td> </tr> <tr> <td>2</td> <td>SIM 2</td> </tr> <tr> <td>any</td> <td>Automatically detect</td> </tr> </table>	1	SIM 1	2	SIM 2	any	Automatically detect
1	SIM 1						
2	SIM 2						
any	Automatically detect						
Web: APN UCI: network.3G.apn Opt: apn	APN name of Mobile Network Operator.						
Web: APN username UCI: network.3G.username Opt: username	Username used to connect to APN.						
Web: APN password UCI: network.3G.password Opt: password	Password used to connect to APN.						
Web: n/a UCI: network.3G.retry_interval_sec Opt: retry_interval_sec	Allows you to specify exact integer or range that will be used to calculate random number to delay PPP connection. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>PPP will connect immediately, without any delay.</td> </tr> <tr> <td>1-infinite</td> <td>PPP will attempt to connect again after specified interval.</td> </tr> <tr> <td>Range</td> <td>PPP will attempt to connect within specified range. The exact interval is calculated randomly from specified range. <code>retry_interval_sec 60 180</code></td> </tr> </table>	0	PPP will connect immediately, without any delay.	1-infinite	PPP will attempt to connect again after specified interval.	Range	PPP will attempt to connect within specified range. The exact interval is calculated randomly from specified range. <code>retry_interval_sec 60 180</code>
0	PPP will connect immediately, without any delay.						
1-infinite	PPP will attempt to connect again after specified interval.						
Range	PPP will attempt to connect within specified range. The exact interval is calculated randomly from specified range. <code>retry_interval_sec 60 180</code>						

Table 43: Information table for common configuration settings

The Modem Configuration link at the bottom of the page is used for SIM pin code and SMS configuration. For more information, read the chapter 'Configuring mobile manager'.

15.2.1.2 Mobile interface: advanced settings

Common Configuration

General Setup Advanced Settings **Firewall Settings**

Bring up on boot

Monitor interface state This interface state would be reported to VA Monitor via keep-alive

Enable IPv6 negotiation on the PPP link

Modem init timeout Maximum amount of seconds to wait for the modem to become ready

Use default gateway If unchecked, no default route is configured

Use gateway metric

IPv4 Mode

IPv6 Mode

Use DNS servers advertised by peer If unchecked, the advertised DNS server addresses are ignored

LCP echo failure threshold Presume peer to be dead after given amount of LCP echo failures, use 0 to ignore failures

LCP echo interval Send LCP echo requests at the given interval in seconds, only effective in conjunction with failure threshold

Inactivity timeout Close inactive connection after the given amount of seconds, use 0 to persist connection

Dependant interfaces MOBILE1:

Figure 74: 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 or reconnect automatically when disconnected.									
Web: Monitor interface state UCI: network.3G.monitored Opt: monitored	Enabled if status of interface is presented on Monitoring platform. <table border="1"> <tr> <td>0</td><td>Do not monitor interface.</td></tr> <tr> <td>1</td><td>Monitor interface.</td></tr> </table>	0	Do not monitor interface.	1	Monitor interface.					
0	Do not monitor interface.									
1	Monitor interface.									
Web: Enable IPv6 negotiation on the PPP link UCI: network.3G.ipv6 Opt: ipv6	Enables IPv6 routing on the interface. <table border="1"> <tr> <td>0</td><td>Do not enable IPv6.</td></tr> <tr> <td>1</td><td>Enable IPv6.</td></tr> </table>	0	Do not enable IPv6.	1	Enable IPv6.					
0	Do not enable IPv6.									
1	Enable IPv6.									
Web: Modem int timeout UCI: network.3G.maxwait Opt: maxwait	Maximum amount of seconds to wait for the modem to become ready. <table border="1"> <tr> <td>20</td><td>Seconds</td></tr> <tr> <td>Range</td><td></td></tr> </table>	20	Seconds	Range						
20	Seconds									
Range										
Web: Use default gateway UCI: network.3G.defaultroute Opt: defaultroute	Enables this interface as a default route <table border="1"> <tr> <td>0</td><td>Do not use as a default route.</td></tr> <tr> <td>1</td><td>Use as a default route.</td></tr> </table>	0	Do not use as a default route.	1	Use as a default route.					
0	Do not use as a default route.									
1	Use as a default route.									
Web: Use gateway metric UCI: network.3G.metric Opt: metric	Defines the metric for the default route. Lower number metrics are used first when route is up. <table border="1"> <tr> <td>0</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	0		Range						
0										
Range										
Web: IPv4 Mode UCI: network.3G.ipv4mode Opt: ipv4mode	Defines the IPv4 address assignment approach for mobile interfaces in Ethernet Mode. Note: by default, mobile interfaces are in Ethernet mode. <table border="1"> <thead> <tr> <th>Option</th><th>Description</th><th>UCI</th></tr> </thead> <tbody> <tr> <td>None</td><td>No dynamic assignment</td><td>none</td></tr> <tr> <td>DHCP</td><td>DHCP address assignment</td><td>dhcp</td></tr> </tbody> </table>	Option	Description	UCI	None	No dynamic assignment	none	DHCP	DHCP address assignment	dhcp
Option	Description	UCI								
None	No dynamic assignment	none								
DHCP	DHCP address assignment	dhcp								

Web: IPv6 Mode UCI: network.3G.ipv6mode Opt: ipv6mode	Defines the IPv6 address assignment approach for mobile interfaces in Ethernet Mode. (Note, by default, mobile interfaces are in Ethernet mode). <table border="1"> <thead> <tr> <th>Option</th><th>Description</th><th>UCI</th></tr> </thead> <tbody> <tr> <td>None</td><td>No dynamic assignment</td><td>none</td></tr> <tr> <td>DHCPv6</td><td>DHCP address assignment</td><td>dhcp</td></tr> <tr> <td>RA</td><td>Router Advertisement (RA) assignment</td><td>ra</td></tr> <tr> <td>DHCPv6 after RA</td><td>Wait for RA, then start DHCP</td><td>ra_then_dhcp</td></tr> </tbody> </table>	Option	Description	UCI	None	No dynamic assignment	none	DHCPv6	DHCP address assignment	dhcp	RA	Router Advertisement (RA) assignment	ra	DHCPv6 after RA	Wait for RA, then start DHCP	ra_then_dhcp
Option	Description	UCI														
None	No dynamic assignment	none														
DHCPv6	DHCP address assignment	dhcp														
RA	Router Advertisement (RA) assignment	ra														
DHCPv6 after RA	Wait for RA, then start DHCP	ra_then_dhcp														
Web: Use DNS servers advertised by peer UCI: network.3G.peerdns Opt: peerdns	If unchecked, the advertised DNS server addresses are ignored. <table border="1"> <tr> <td>0</td><td>Use static DNS</td></tr> <tr> <td>1</td><td>Use advertised DNS</td></tr> </table>	0	Use static DNS	1	Use advertised DNS											
0	Use static DNS															
1	Use advertised DNS															
Web: Use custom DNS servers UCI: network.3G.dns Opt: dns	Specifies DNS server. Only available if Use DNS servers advertised by peer is unselected. When multiple DNS servers are required separate using space for UCI or option value. Example: <code>uci set network.3G.dns='1.1.1.1 2.2.2.2'</code>															
Web: LCP echo failure threshold UCI: network.3G.keepalive Opt: keepalive	Presumes peer to be dead after a 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: <code>uci network.3G.keepalive=<echo failure threshold> <echo interval></code> Example: <code>uci set network.3G.keepalive=15 10</code>															
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: <code>uci network.3G.keepalive=<echo failure threshold> <echo interval></code> Example: <code>uci set network.3G.keepalive=15 10</code>															
Web: Inactivity timeout UCI: network.3G.demand Opt: demand	Closes an inactive connection after the given amount of seconds. Use 0 to persist connection. <table border="1"> <tr> <td>0</td><td>Do not disconnect on inactivity</td></tr> <tr> <td>Range</td><td></td></tr> </table>	0	Do not disconnect on inactivity	Range												
0	Do not disconnect on inactivity															
Range																
Web: Dependant Interfaces UCI: network.3G.dependants Opt: dependants	Lists interfaces that are dependent on this parent interface. Dependant interfaces will go down when the parent interface is down and will start or restart when the parent interface starts. Separate multiple interfaces by a space when using UCI. Example: <code>option dependants 'PPPADSL MOBILE'</code> This replaces the following previous options in child interfaces. <table border="1"> <tr> <td>gre</td><td>option local_interface</td></tr> <tr> <td>lt2p</td><td>option src_ipaddr</td></tr> <tr> <td>iot</td><td>option wan1 wan2</td></tr> <tr> <td>6in4</td><td>option ipaddr</td></tr> <tr> <td>6to4</td><td>option ipaddr</td></tr> </table>	gre	option local_interface	lt2p	option src_ipaddr	iot	option wan1 wan2	6in4	option ipaddr	6to4	option ipaddr					
gre	option local_interface															
lt2p	option src_ipaddr															
iot	option wan1 wan2															
6in4	option ipaddr															
6to4	option ipaddr															
Web: SNMP Alias ifindex UCI: network.[..x..].snmp_alias_ifindex Opt: snmp_alias_ifindex	Defines a static SNMP interface alias index for this interface that can be polled via the SNMP interface index. (<code>snmp_alias_ifindex+1000</code>). See Configuring SNMP section for more information. <table border="1"> <tr> <td>Blank</td><td>No SNMP interface alias index</td></tr> <tr> <td>Range</td><td>0 - 4294966295</td></tr> </table>	Blank	No SNMP interface alias index	Range	0 - 4294966295											
Blank	No SNMP interface alias index															
Range	0 - 4294966295															

Table 44: Information table for general set up page

15.2.1.3 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.

Common Configuration

General Setup Advanced Settings Firewall Settings

Create / Assign firewall-zone unspecified -or- create:

Choose 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.

Back to Overview Save & Apply Save Reset

Figure 75: Firewall settings page

15.3 Configuring a mobile connection using CLI

15.3.1 UCI

To establish a basic mobile connection, enter:

```
root@GW_router:~# uci show network
network.3G=interface
network.3G.proto=3g
network.3G.monitored=0
network.3G.sim=any
network.3G.auto=1
network.3G.defaultroute=1
network.3G.metric=1
network.3G.service=autonetwork.3G.apn=test.apn
network.3G.username=username
network.3G.password=password
network.3G.ipv4mode=dhcp
network.3G.ipv6mode=none
```

15.3.2 Package options

```
root@GW_router:~#
package network

config interface '3G'
    option proto '3g'
    option monitored '0'
    option auto '1'
    option sim 'any'
    option defaultroute '1'
    option metric '1'           option service 'auto'
    option apn 'test.apn'
    option username 'username'
    option password 'password'
    option ipv4mode 'dhcp'
    option ipv6mode 'none'
```

15.4 Diagnositcs

Note: the information presented on screen and data output using UCI depends on the actual mobile hardware being used. Therefore, the interfaces or output you see may differ from the samples shown here.

15.4.1 Mobile status via the web

To view mobile connectivity information, in the top menu, select **Status -> Mobile Information**. The Mobile Information page appears. The information presented depends on the actual mobile hardware used; it might therefore differ from the samples shown here.

WAN	
Basic	Advanced
Cell Information	
SIM In	yes
SIM Slot	1
Operator	vodafone IE
Technology	UMTS
Network Status	Home network
Data Network Status	Home network
Signal (dBm)	-101
IMEI	358743040012737
IMSI	272017113618040

Figure 76: The mobile information page

WAN	
Basic	Advanced
Cell Information	
Network Status	Home network
Data Network Status	Home network
IMEI	358743040012737
IMSI	272017113618040
Operator	vodafone IE
Phone Number	+353874512040
SIM In	yes
SIM Slot	1
SIM1 ICCID	8935301140701270414
Signal (dBm)	-101
Technology	UMTS
Temperature (C)	28
Hardware Revision	R1C08

Figure 77: The advanced information page

WAN	
Basic	Advanced
Cell Information	
Cell ID	2007516
Location Area Code	3023
Mobile Country Code	272
Mobile Network Code	01

Figure 78: The cell information page

15.4.2 Mobile status using UCI

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

```
root@GW_router:~# mobile_status

Mobile Interface      : WAN
Status                : idle
SIM In               : yes
SIM Slot              : 1
Operator              : vodafone IE
Technology            : UMTS
CS Network Status     : Home network
PS Network Status     : Home network
Signal (dBm)          : -107
IMEI                 : 358743040012737
IMSI                 : 272017113618040
```

For more advanced information, enter `mobile_status -a`:

```
root@ GW_router:~# mobile_status -a

Mobile Interface      : WAN
Status                : idle
CS Network Status     : Home network
```

PS Network Status	:	Home network
IMEI	:	358743040012737
IMSI	:	272017113618040
Operator	:	vodafone IE
Phone Number	:	+353874512040
SIM In	:	yes
SIM Slot	:	1
SIM1 ICCID	:	8935301140701270414
Signal (dBm)	:	-107
Technology	:	UMTS
Temperature (C)	:	28
Hardware Revision	:	R1C08

16 Configuring mobile manager

The Mobile Manager feature allows you to configure SIM settings.

16.1 Configuration package used

Package	Sections
mobile	Main
	Callers
	Roaming template

16.2 Configuring mobile manager using the web interface

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

There are four sections in the mobile manager page

Section	Description
Basic settings	Enable SMS, configure SIM pin code, select roaming SIM, collect ICCCIDs and set IMSI.
CDMA*	CDMA configuration
Callers	Configure callers that can use SMS.
Roamin Interface Template	Configure Preferred Roaming List options

*Option available only for Telit CE910-SL module.

16.2.1 Mobile manager: basic settings

MAIN

[Basic](#) [CDMA](#)

SMS Enable

Force Mode

Collect ICCIDs [?](#) *Collect ICCIDs on startup*

IMSI

PIN-code for SIM1

PIN-code for SIM2

LTE Bands for SIM1

LTE Bands for SIM2

Temperature Polling Interval (Seconds)

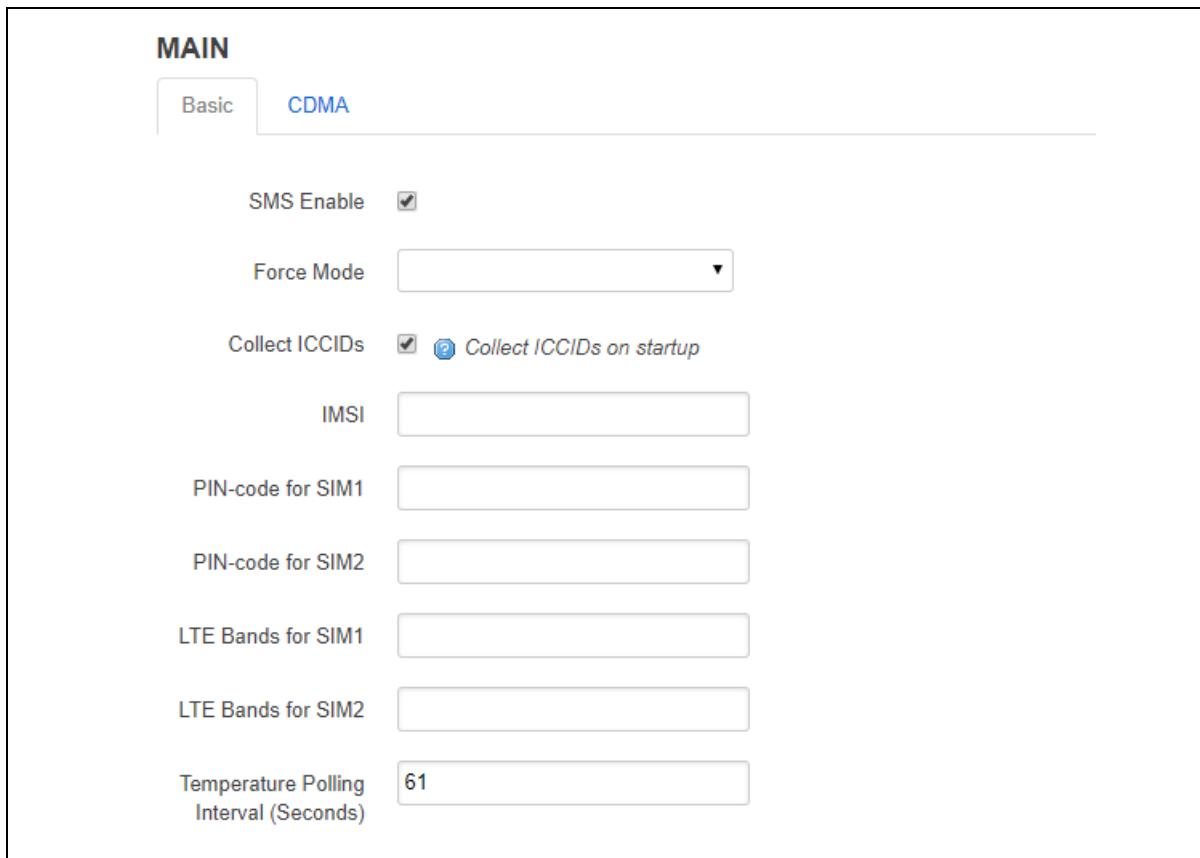


Figure 79: The mobile manager basic page

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: Force Mode UCI: mobile.main.force_mode Opt: force_mode	Defines whether to operate mobile modem in TTY or Ethernet mode. The mode will be dependent on the service provided by the mobile provider. In general, this is Ethernet mode (default).	
	Empty	Ethernet mode (option not present).
	tty	Enable TTY mode.
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: IMSI UCI: mobile.main.imsi Opt: imsi	Allows the IMSI (International Mobile Subscriber Identity) to be changed	
	Default	Programmed in module
	Digits	up to 15 digits
Web: PIN code for SIM1 UCI: mobile.main.sim1pin Opt: sim1pin	Depending on the SIM card specify 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 specify the pin code for SIM 2.	
	Blank	
	Range	Depends on the SIM provider.
Web: LTE bands for SIM1 UCI: mobile.main.sim1_lte_bands Opt: sim1_lte_bands	Depending on the SIM card specify the LTE bands for SIM 1. Comma delimiter. Example: option sim1_lte_bands '3,20' Limits LTE bands to 3 and 20. Note: currently only supported by Hucom/Wetelcom, SIMCom7100, Cellient MPL200 and Asiatel.	
	Blank	
	Range	LTE bands range from 1 to 70
Web: LTE bands for SIM2 UCI: mobile.main.sim2_lte_bands Opt: sim2_lte_bands	Depending on the SIM card specify the LTE bands for SIM 2. Comma delimiter. Example: option sim1_lte_bands '3,20' Limits LTE bands to 3 and 20. Note: currently only supported by Hucom/Wetelcom, SIMCom7100, Cellient MPL200 and Asiatel.	
	Blank	
	Range	LTE bands range from 1 to 70
Web: Temperature Polling Interval UCI: mobile.main.temp_poll_interval_sec Opt: temp_poll_interval_sec	Defines the time in seconds to poll the mobile module for temperature. Set to 0 to disable.	
	61	61 seconds
	Range	
Web: n/a UCI: mobile.main.disable_time Opt: disable_time	Defines whether to use time obtained from the mobile carrier to update the system clock when NTP is enabled.	
	0	Disabled.
	1	Enabled.

Table 45: Information table for mobile manager basic settings

16.2.2 Mobile manager: CDMA settings

This configuration page is only supported for the Telit CE910-SL CDMA module.

MAIN	
Basic	CDMA
HDR Auth User ID	<input type="text"/> <small>AN-PPP user id. Supported on Cellient modem only</small>
HDR Auth Password	<input type="text"/> <small>AN-PPP password. Supported on Cellient modem only</small>
Ordered Registration triggers module reboot	<input type="checkbox"/>
Station Class Mark	<input type="text"/>
Slot Cycle Index	<input type="text"/>
Slot Mode	<input type="text"/>
Mobile Directory Number	<input type="text"/>
MOB_TERM_HOME registration flag	<input type="checkbox"/>
MOB_TERM_FOR_SID registration flag	<input type="checkbox"/>
MOB_TERM_FOR_NID registration flag	<input type="checkbox"/>

Figure 80: The mobile manager CDMA page

Web Field/UCI/Package Option	Description	
Web: HDR Auth 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 Auth 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: Ordered Registration triggers module reboot UCI: mobile.main.mobile.main_cdma_ordered_registration_reboot_enabled Opt: cdma_ordered_registration_reboot_enabled	Enables or disables rebooting the module after Order Registration command is received from a network. 0 1	Disabled. Enabled.
Web: Station Class Mark UCI: mobile.main.cdma_station_class_mark Opt: cdma_station_class_mark	Allows the station class mark for the MS to be changed. 58 0-255	Default Range.
Web: Slot Cycle Index UCI: mobile.main.cdma_slot_cycle_index Opt: cdma_slot_cycle_index	The desired slot cycle index if different from the default 2 0-7	Default Range.
Web: Slot Mode UCI: mobile.main.cdma_slot_mode Opt: cdma_slot_mode	Specifies the slot mode 0 TBA	Default TBA
Web: Mobile Directory Number UCI: mobile.main.cdma_mobile_directory_number Opt: cdma_mobile_directory_number	Allows the mobile directory number (MDN) to be changed Default Digits	Programmed in module up to 15 digits
Web: MOB_TERM_HOME registration flag UCI: mobile.main.cdma_mob_term_home_registration_flag Opt: cdma_mob_term_home_registration_flag	The MOB_TERM_HOME registration flag 0 1	Disabled. Enabled.
Web: MOB_TERM_FOR_SID registration flag UCI: mobile.main.cdma_mob_term_for_sid_registration_flag Opt: cdma_mob_term_for_sid_registration_flag	The MOB_TERM_FOR_SID registration flag 0 1	Disabled. Enabled.
Web: MOB_TERM_FOR_NID registration flag UCI: mobile.main.cdma_mob_term_for_nid_registration_flag Opt: cdma_mob_term_for_nid_registration_flag	The MOB_TERM_FOR_NID registration flag 0 1	Disabled. Enabled.
Web: Access Overload Control UCI: mobile.main.cdma_access_overload_control Opt: cdma_access_overload_control	Allows the access overload class to be changed Default 0-7	programmed into module as part of IMSI Range.
Web: Preferred Serving System UCI: mobile.main.cdma_preferred_serving_system Opt: cdma_preferred_serving_system	The CDMA Preferred Serving System(A/B) Default value is 5.	
Web: Digital Analog Mode Preference UCI: cdma_digital_analog_mode_preference Opt: cdma_digital_analog_mode_preference	Digital/Analog Mode Preference Default: 4	

Web: Primary Channel A UCI: mobile.main.cdma_primary_channel_a Opt: cdma_primary_channel_a	Allows the primary channel (A) to be changed 283 Default 1-2016 any band class 5 channel number
Web: Primary Channel B UCI: mobile.main.cdma_primary_channel_b Opt: cdma_primary_channel_b	Allows the primary channel (B) to be changed 384 Default 1-2016 any band class 5 channel number
Web: Secondary Channel A UCI: mobile.main.cdma_secondary_channel_a Opt: cdma_secondary_channel_a	Allows the secondary channel (A) to be changed 691 Default 1-2016 any band class 5 channel number
Web: Secondary Channel B UCI: mobile.main.cdma_secondary_channel_b Opt: cdma_secondary_channel_b	Allows the secondary channel (B) to be changed 777 Default 1-2016 any band class 5 channel number
Web: Preferred Forward & Reverse RC UCI: mobile.main.cdma_preferred_forward_and_reverse_rc Opt:cdma_preferred_forward_and_reverse_rc	The Preferred Forward & Reverse RC value, this takes the form "forward_rc,reverse_rc" Format: forward radio channel, reverse radio channel Default: 0,0
Web: SID-NID pairs UCI: mobile.main.cdma_sid_nid_pairs Opt:cdma_sid_nid_pairs	Allows specification of SID:NID pairs, this takes the form "SID1,NID1,SID2,NID2, ..." Format: SID1 (0-65535),NID (0-65535) Default: 0,65535

Table 46: Information table for mobile manager CDMA settings

16.2.3 Mobile manager: callers

Callers
Configure caller numbers that may use the SMS service.

Name	<input type="text" value="CallerGroup1"/>	<small>(?) Name of the caller.</small>
Number	<input type="text" value="353*"/>	<small>(?) Number of the caller. Use * for wildcard matching.</small>
Enable	<input type="checkbox"/>	
Respond	<input type="checkbox"/>	

Figure 81: The mobile manager CDMA page

Web: Name UCI: mobile.@caller[0].name Opt:name	Name assigned to the caller.	
	Blank	
Web: Number UCI: mobile.@caller[0].number Opt:number	Range	No limit
	Number of the caller allowed to SMS the router. Add in specific caller numbers, or use the * wildcard symbol.	
	Blank	
Web: Enable UCI: mobile.@caller[0].enabled Opt:enabled	Range	No limit
	Characters	Global value (*) is accepted International value (+) is accepted
Web: Respond UCI: mobile.@caller[0].respond Opt: respond	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 47: Information table for mobile manager callers settings

16.2.4 Mobile manager: roaming interface template

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

16.3 Configuring mobile manager using command line

16.3.1 Mobile manager using UCI

The configuration files for mobile manager are stored on **/etc/config/mobile**

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

```
root@GW_router:~# uci show mobile
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=1
uci set mobile.main.hdr_password=5678
uci set mobile.main.hdr_userid=1234
uci set mobile.main.init_get_iccids=1
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=1
uci set mobile.@caller[0].respond=1
```

```
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=1
uci set mobile.@caller[1].respond=1
```

16.3.2 Mobile manager using package options

```
root@GW_router:~# uci export mobile
package mobile
config mobile 'main'
    option sim1pin '0000'
    option sim2pin '0000'
    option roaming_sim 'none'
    option sms '1'
    option hdr_password '5678'
    option hdr_userid '1234'
    option init_get_iccids '1'
config caller
    option name 'vasupport'
    option number '353871234567'
    option enabled '1'
    option respond '1'
config caller
    option name 'vasupport1'
    option number '353872345678'
    option enabled '1'
    option respond '1'
```

16.4 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 82: Example of output from system log

To monitor via SSH, login and enter:

```
logread -f &
```

An outgoing SMS message appears.

```
sendsms 353879876543 'hello'
root@SATEL:~# Aug 10 16:29:11 user.notice SATEL mobile[1737]: Queue sms to
353879876543 "hello"
```

16.5 Sending SMS from the router

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

```
sendsms 353879876543 'hello'
root@SATEL:~# Aug 10 16:29:1 user.notice SATEL mobile[1737]: Queue sms to
353879876543 "hello"
```

16.6 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
```

17 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.

17.1 Configuration packages used

Package	Sections
network	interface

17.2 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 83: 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.																										
Web: Protocol of the new interface UCI: network.<if name>.proto Opt: proto	Specifies what protocol the interface will operate on. Select GRE . <table border="1" data-bbox="690 422 1341 1010"> <thead> <tr> <th>Option</th><th>Description</th></tr> </thead> <tbody> <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> </tbody> </table>	Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.	IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.	GRE	Generic Routing Encapsulation protocol	IOT		L2TP	Layer 2 Tunnelling Protocol	PPP	Point-to-Point protocol	PPPoE	PPP over Ethernet	PPPoATM	PPP over ATM	LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.
Option	Description																										
Static	Static configuration with fixed address and netmask.																										
DHCP Client	Address and netmask are assigned by DHCP.																										
Unmanaged	Unspecified																										
IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.																										
IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.																										
GRE	Generic Routing Encapsulation protocol																										
IOT																											
L2TP	Layer 2 Tunnelling Protocol																										
PPP	Point-to-Point protocol																										
PPPoE	PPP over Ethernet																										
PPPoATM	PPP over ATM																										
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																										
Web: Create a bridge over multiple interfaces UCI: network.<if name> Opt: n/a	Not applicable for GRE.																										
Web: Cover the following interface UCI: network.<if name> Opt:n/a	Not applicable for GRE.																										

Table 48: 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.

17.2.1 GRE connection: common configuration - general setup

The screenshot shows the 'Common Configuration' page for a GRE tunnel. At the top, there are tabs for 'General Setup' (selected), 'Advanced Settings', and 'Firewall Settings'. Below the tabs, the 'Status' section shows 'gre-Tunnel1' with RX and TX counters both at 0.00 B (0 Pkts.). The 'Protocol' dropdown is set to 'GRE'. The 'Tunnel IP Address' and 'Mask Length' fields are empty. The 'Local Interface' section lists several options: 3G, ADSL, Test_BC, lan, lan2, lan3, lan4, loopback, and ethalias (with a note '(no interfaces attached)'). The 'Remote IP Address' field is empty. The 'TTL' field contains the value '128'. The 'Tunnel key' and 'MTU' fields are also empty. The 'MTU' field has a value of '1472'.

Figure 84: The GRE common configuration page

Web Field/UCI/Package Option	Description				
Web: Protocol of the new interface UCI: network.<if name>.proto Opt: proto	Shows the protocol the interface will operate on. GRE should be currently selected.				
Web: Tunnel IP Address UCI: network.<if name>.ipaddr Opt: ipaddr	Configures local IP address of the GRE interface.				
Web: Mask Length UCI: network.<if name>.mask_length Opt: mask_length	Subnet mask, in CIDR notation, to be applied to the tunnel. Typically '30' for point-to-point tunnels. <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">24</td> <td style="width: 50%;"></td> </tr> <tr> <td colspan="2">Range 0 - 30</td> </tr> </table>	24		Range 0 - 30	
24					
Range 0 - 30					

Web: Local Interface UCI: network.<if name>.local_interface Opt: local_interface	Specifies which interface is going to be linked with the GRE tunnel interface (optional).				
Web: Remote IP address UCI: network.<if name>.remote_ip Opt: remote_ip	For point to point tunnels specifies Remote IP address.				
Web: TTL UCI: network.<if name>.ttl Opt: ttl	Sets Time-To-Live value on the interface. <table border="1"><tr><td>128</td><td></td></tr><tr><td>Range</td><td></td></tr></table>	128		Range	
128					
Range					
Web: Tunnel key UCI: network.<if name>.key Opt: key	Sets GRE tunnel ID key (optional). Usually an integer.				
Web: MTU UCI: network.<if name>.mtu Opt: mtu	Configures MTU (maximum transmission unit) size of PDUs using this interface. <table border="1"><tr><td>1472</td><td></td></tr><tr><td>Range</td><td></td></tr></table>	1472		Range	
1472					
Range					

Table 49: Information table for GRE

17.2.2 GRE connection: common configuration-advanced settings

The screenshot shows the 'Common Configuration' page with the 'Advanced Settings' tab selected. Key visible fields include:

- Bring up on boot:** A checked checkbox.
- Monitor interface state:** An unchecked checkbox with a tooltip: "This interface state would be reported to VA Monitor via keep-alive".
- Dependant interfaces:** A list of interfaces with checkboxes:
 - GRETUNNEL1: (checked)
 - MOBILE_amylan: (unchecked)
 - MOBILE_voda: (unchecked)
 - PeAADSL: (unchecked)
 - SUBNET1: (no interfaces attached)
 - SUBNET2: (unchecked)
 - SUBNET3: (unchecked)
 - SUBNET4: (unchecked)
 - loopback: (unchecked)
- Check interfaces which should start after this interface is started and stop after this interface is stopped:** An unchecked checkbox.
- SNMP Alias ifindex:** A text input field containing '1'. A tooltip states: "Alias ifindex SNMP agent. Alias indexes are present at 1000 offset. So setting 1 here will create snmp ifTable entry 1001. Useful when interface creates new linux interface on every startup (e.g. ppp interface). With this set the interface could be monitored via constant snmp agent interface table entry".

Figure 85: 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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
0	Disabled.										
1	Enabled.										
Web: Dependant Interfaces UCI: network.[..x...].dependants Opt: dependants	Lists interfaces that are dependent on this parent interface. Dependant interfaces will go down when parent interface is down and will start or restart when parent interface starts. Separate multiple interfaces by a space when using UCI. Example: option dependants 'PPPADSL MOBILE' This replaces the following previous options in child interfaces. <table border="1"> <tr> <td>gre</td><td>option local_interface</td></tr> <tr> <td>lt2p</td><td>option src_ipaddr</td></tr> <tr> <td>iot</td><td>option wan1 wan2</td></tr> <tr> <td>6in4</td><td>option ipaddr</td></tr> <tr> <td>6to4</td><td>option ipaddr</td></tr> </table>	gre	option local_interface	lt2p	option src_ipaddr	iot	option wan1 wan2	6in4	option ipaddr	6to4	option ipaddr
gre	option local_interface										
lt2p	option src_ipaddr										
iot	option wan1 wan2										
6in4	option ipaddr										
6to4	option ipaddr										
Web: SNMP Alias ifindex UCI: network.[..x...].snmp_alias_ifindex Opt: snmp_alias_ifindex	Defines a static SNMP interface alias index for this interface, that can be polled via the SNMP interface index (<i>snmp_alias_ifindex+1000</i>). See <i>Configuring SNMP</i> section for more information <table border="1"> <tr> <td>Blank</td><td>No SNMP interface alias index</td></tr> <tr> <td>Range</td><td>0 - 4294966295</td></tr> </table>	Blank	No SNMP interface alias index	Range	0 - 4294966295						
Blank	No SNMP interface alias index										
Range	0 - 4294966295										

Table 50: Information table for GRE advanced settings

17.2.3 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.

Common Configuration

General Setup Advanced Settings Firewall Settings

Create / Assign firewall-zone

- lan: lan:
- wan: ADSL: 3G:
- unspecified -or- create:

Choose the firewall zone you want to assign to this interface. Select unspecified to remove the interface from a zone.

Back to Overview Save & Apply Save Reset

Figure 86: 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.

17.2.4 GRE connection: adding a static route

After you have configured the GRE interface, you must configure a static route to route the desired traffic over the GRE tunnel. To do this, browse to **Network->Static Routes**. For more information, read the chapter 'Configuring Static Routes'.

17.3 GRE configuration using command line

The configuration file is stored on **/etc/config/network**

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

17.4 GRE configuration using UCI

```
root@GW_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
```

17.5 GRE configuration using package options

```
root@GW_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.

17.6 GRE diagnostics

17.6.1 GRE interface status

To show the current running interfaces, enter:

```
root@GW_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
                      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@GW_router:~# ifconfig gre-Tunnell1
gre-Tunnell1 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@GW_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-Tunnell1
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-Tunnell1
172.19.101.3   13.13.13.1   255.255.255.255 UGH     0      0      0
gre-Tunnell1
```

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

18 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.

18.1 Configuration package used

Package	Sections
network	route

18.2 Configuring static routes using the web interface

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

Interface	Target	IPv4-Netmask	IPv4.Gateway	Metric	MTU
Host-IP or Network	<i>if target is a network</i>				

This section contains no values yet

Add

Interface	Target	IPv6-Gateway	Metric	MTU
IPv6-Address or Network (CIDR)				

This section contains no values yet

Add

Save & Apply Save Reset

Figure 87: 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. <table border="1"><tr><td>0</td><td></td></tr><tr><td>Range</td><td></td></tr></table>	0		Range	
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. <table border="1"><tr><td>Empty</td><td></td></tr><tr><td>Range</td><td></td></tr></table>	Empty		Range	
Empty					
Range					

Table 51: Information table for IPv4 static routes section

18.3 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: Example: 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. <table border="1"><tr><td>0</td><td></td></tr><tr><td>Range</td><td></td></tr></table>	0		Range	
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. <table border="1"><tr><td>Empty</td><td></td></tr><tr><td>Range</td><td></td></tr></table>	Empty		Range	
Empty					
Range					

Table 52: Information table for IPv6 routes

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

18.4 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, you can give a route 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'
```

18.5 IPv4 routes using UCI

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

```
root@GW_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
```

18.6 IPv4 routes using package options

```
root@GW_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'
```

18.7 IPv6 routes using UCI

```
root@GW_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
```

18.8 IPv6 routes using packages options

```
root@GW_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'
```

18.9 Static routes diagnostics

18.9.1 Route status

To show the current routing status, enter:

```
root@GW_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.

19 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.

19.1 Configuration package used

Package	Sections
bgpd	routing
	peer
	routemap

19.2 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.

The screenshot shows the BGP configuration page with the following sections:

- Global Settings:** Contains an "Add" button.
- BGP Route Map:** Displays a message "This section contains no values yet" and includes input fields for IP Address and Autonomous System Number, along with an "Add" button.
- BGP Neighbours:** A table with columns: IP Address, Autonomous System Number, Route Map, and Route Map Direction. It displays a message "This section contains no values yet". An "Add" button is located below the table.
- Action Buttons:** At the bottom right are "Save & Apply", "Save", and "Reset" buttons.

Figure 88: The BGP page

19.2.1 BGP global settings

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

BGP

Global Settings

BGP Enabled

Router ID

Autonomous System Number

Network

These networks will be announced to neighbors

Figure 89: The BGP global settings page

Web Field/UCI/Package Option	Description					
Web: BGP Enabled UCI: bgpd.bgpd.enabled Opt: enabled	Enables or disables BGP protocol. <table border="1"><tr><td>1</td><td>Enabled.</td></tr><tr><td>0</td><td>Disabled.</td></tr></table>		1	Enabled.	0	Disabled.
1	Enabled.					
0	Disabled.					
Web: Router ID UCI: bgpd.bgpd.router_id Opt: router_id	1	Enabled.				
Web: Autonomous System Number UCI: bgpd.bgpd.asn Opt: asn	0	Disabled.				
Web: Network UCI: bgpd.bgpd.network Opt: list network	Sets a Unique Router ID in 4 byte format 0.0.0.0. Defines the ASN for the local router. Type in the ASN. <table border="1"><tr><td>Blank</td><td></td></tr><tr><td>Range</td><td>1-4294967295</td></tr></table> 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.		Blank		Range	1-4294967295
Blank						
Range	1-4294967295					

Table 53: Information table for BGP global settings

19.2.2 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. You can configure multiple route maps.

ROUTEMAP

Order: 10

Policy Type: Permit

Match Type: IP Address

Match Value: 192.168.101.1/32 (Format depends on Match Type. In case of IP Address and BGP Community value is parsed as list of items to match. Use '-' prefix to deny match.)

Set Option: Route Weight

Set Value: 150

Figure 90: The routemap section

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

19.2.3 Configure BGP neighbours

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

BGP neighbors			
IP Address	Autonomous System Number	Route Map	Route Map Direction
10.1.10.83	1		In
<input type="button" value="Delete"/> <input type="button" value="Add"/>			

Figure 91: 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. <table border="1"><tr><td>Blank</td><td></td></tr><tr><td>Range</td><td>1-4294967295</td></tr></table>	Blank		Range	1-4294967295
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. <table border="1"><tr><td>1</td><td>In</td></tr><tr><td>0</td><td>Out</td></tr></table>	1	In	0	Out
1	In				
0	Out				

Table 55: Information table for BGP neighbours

19.3 Configuring BGP using UCI

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

```
root@GW_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.

19.4 Configuring BGP using packages options

```

root@GW_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'

```

19.5 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	FFFFFF
loopback	0:0:0:0:0:0:0/0	0:0:0:0:0:0:0/0	FFFFFF
loopback	0:0:0:0:0:0:1	0:0:0:0:0:0:0/0	0000000
LAN2	FF02:0:0:0:0:FB	0:0:0:0:0:0:0/0	0000000
(base0)	FF00:0:0:0:0:0:8	0:0:0:0:0:0:0/0	00000100
lan	FF00:0:0:0:0:0:8	0:0:0:0:0:0:0/0	00000100
LAN2	FF00:0:0:0:0:0:8	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	FFFFFF

Figure 92: The routing table

To view routes via the command line, enter:

```
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
```

20 Configuring VRRP

20.1 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. You can also configure a password authentication to protect VRRP protocol messages against spoofing.

The VRRP protocol is implemented according to internet standard RFC2338.

20.2 Configuration package used

Package	Sections
vrrp	main vrrp_group

20.3 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 93: The VRRP group configuration page

Web Field/UCI/Package Option	Description				
Web: VRRP Enabled UCI: vrrp.main.enabled Opt: Enabled	Globally enables VRRP on the router. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Group Enabled UCI: vrrp.@vrrp_group[X].enabled Opt: Enabled	Enables a VRRP group on the router. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Interface UCI: vrrp.@vrrp_group[X].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. <table border="1"> <tr> <td>lan</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	lan		Range	
lan					
Range					
Web: Track Interfaces UCI: vrrp.@vrrp_group[X].track_iface Opt: list 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. Multiple interfaces should be entered with space separator when using UCI. Example: <code>vrrp.@vrrp_group[X].track_iface=WAN MOBILE</code> <table border="1"> <tr> <td>wan</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	wan		Range	
wan					
Range					
Web: IPsec connection UCI: vrrp.@vrrp_group[X].ipsec_connection Opt: ipsec_connection	Sets which IPsec connection to bring up or down when VRRP enters 'Backup/Master' state. <table border="1"> <tr> <td>(blank)</td><td>No IPsec connection to toggle.</td></tr> <tr> <td>Range</td><td></td></tr> </table>	(blank)	No IPsec connection to toggle.	Range	
(blank)	No IPsec connection to toggle.				
Range					

Web: Start role UCI: vrrp.@vrrp_group[X].init_state Opt: init_state	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 border="1"> <tr><td>BACKUP</td><td></td></tr> <tr><td>MASTER</td><td></td></tr> </table>	BACKUP		MASTER			
BACKUP							
MASTER							
Web: Router ID UCI: vrrp.@vrrp_group[X].router_id Opt: router_id	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 border="1"> <tr><td>1</td><td></td></tr> <tr><td>Range</td><td>1-255</td></tr> </table>	1		Range	1-255		
1							
Range	1-255						
Web: Priority UCI: vrrp.@vrrp_group[X].priority Opt: priority	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 border="1"> <tr><td>100</td><td></td></tr> <tr><td>Range</td><td>0-255</td></tr> </table>	100		Range	0-255		
100							
Range	0-255						
Web: Advert intvl UCI: vrrp.@vrrp_group[X].advert_int_sec Opt: advert_int_sec	Sets the VRRP hello value in seconds. This value must match the value set on a peer. <table border="1"> <tr><td>120</td><td>120 seconds</td></tr> <tr><td>Range</td><td></td></tr> </table>	120	120 seconds	Range			
120	120 seconds						
Range							
Web: Password UCI: vrrp.@vrrp_group[X].password Opt: password	Sets the password to use in the VRRP authentication (simple password authentication method). This field may be left blank if no authentication is required.						
Web: Virtual IP UCI: vrrp.@vrrp_group[X].virtual_ipaddr Opt: virtual_ipaddr	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.						
Web: GARP delay UCI: vrrp.@vrrp_group[X].garp_delay_sec Opt: garp_delay_sec	Sets the Gratuitous ARP message sending delay in seconds. <table border="1"> <tr><td>5</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	5		Range			
5							
Range							
Web: n/a UCI: vrrp.@vrrp_group[X].track_ipsec Opt: list track_ipsec	Sets one or more IPSec connection that VRRP should monitor. If a monitored IPSec connection goes down on the Master VRRP router, it goes into 'Fault' state and the Backup VRRP router becomes the Master. Multiple IPSec connection should be entered with space separator when using UCI. Example: <table border="1"> <tr><td>vrrp.@vrrp_group[X].track_ipsec=conn1 conn2</td><td></td></tr> <tr><td>(blank)</td><td>No IPSec connection to track.</td></tr> <tr><td>Range</td><td></td></tr> </table>	vrrp.@vrrp_group[X].track_ipsec=conn1 conn2		(blank)	No IPSec connection to track.	Range	
vrrp.@vrrp_group[X].track_ipsec=conn1 conn2							
(blank)	No IPSec connection to track.						
Range							

Table 56: Information table for VRRP settings

20.4 Configuring VRRP using command line

The configuration file is stored on **/etc/config/vrrp**.

There are two config sections – **main** and **vrrp_group**.

Multiple VRRP groups can be configured. By default, all VRRP group instances are named 'vrrp_group'. It is identified by @vrrp_group then the vrrp_group position in the package as a number. For example, for the first vrrp_group in the package using UCI:

```
vrrp.@vrrp_group[0]=vrrp_group
vrrp.@vrrp_group[0].enabled=1
```

Or using package options:

```
config vrrp_group
    option enabled '1'
```

However, to better identify, it is recommended to give the vrrp_group instance a name. For example, to define a vrrp_group instance named 'g1' using UCI, enter:

```
vrrp.g1.vrrp_group
vrrp.g1.enabled=1
```

To define a named keepalive instance using package options, enter:

```
config vrrp_group 'g1'
    option enabled '1'
```

20.4.1 VRRP using UCI

To view the configuration in UCI format, enter:

```
root@GW_router:~# uci show vrrp
vrrp.main=vrrp
vrrp.main.enabled=yes
vrrp.g1=vrrp_group
vrrp.g1.enabled=yes
vrrp.g1.interface=lan
vrrp.g1.track_iface=WAN MOBILE
vrrp.g1.init_state=BACKUP
vrrp.g1.router_id=1
vrrp.g1.priority=100
vrrp.g1.advert_int_sec=120
vrrp.g1.password=secret
vrrp.g1.virtual_ipaddr=10.1.10.150/16
vrrp.g1.garp_delay_sec=5
vrrp.g1.ipsec_connection=Test
vrrp.g1.track_ipsec=conn1 conn2
```

20.4.2 VRRP using package options

To view the configuration in package option format, enter:

```
root@GW_router:~# uci export vrrp
package vrrp

config vrrp 'main'
    option enabled 'yes'

config vrrp_group 'g1'
    option enabled 'yes'
    option interface 'lan'
    list track_iface 'WAN'
    list track_iface 'MOBILE'
    option init_state 'BACKUP'
    option router_id '1'
    option priority '100'
    option advert_int_sec '120'
    option password 'secret'
    option virtual_ipaddr '10.1.10.150/16'
    option garp_delay_sec '5'
    option ipsec_connection 'Test'
    list track_ipsec 'conn1'
    list track_ipsec 'conn2'
```

21 Configuring Routing Information Protocol (RIP)

21.1 Introduction

RIP is a dynamic routing algorithm used on IP-based internet networks.

A distance-vector routing algorithm is used by RIP to assist in maintaining network convergence. It uses a metric or 'hop' count as the only routing criteria. Each route is advertised with the number of hops a datagram would take to reach the destination network. The maximum metric for RIP is 15. This limits the size of the network that RIP can support. Smaller metrics are more efficient-based on the cost associated with each metric.

RIP protocol is most useful as an Interior Gateway Protocol (IGP). An IGP refers to the routing protocol used within a single autonomous system. There may be a number of autonomous systems, using different routing protocols, combined together to form a large network.

In most networking environments, RIP is not the preferred choice for routing as its time to converge and scalability are poor compared to EIGRP or OSPF.

21.1.1 RIP characteristics

RIP is a standardised distance vector protocol, designed for use on smaller networks. RIP was one of the first true distance vector routing protocols, and is supported on a wide variety of systems.

RIP adheres to the following distance vector characteristics:

- RIP sends out periodic routing updates, every 30 seconds
- RIP sends out the full routing table every periodic update
- RIP uses a form of distance as its metric, in this case, hopcount
- RIP uses the Bellman-Ford distance vector algorithm to determine the best path to a particular destination

Other characteristics of RIP include:

- RIP supports IP and IPX routing
- RIP utilizes UDP port 520
- RIP routes have an administrative distance of 120
- RIP has a maximum hopcount of 15 hops. Any network that is 16 hops away or more is considered unreachable to RIP, thus the maximum diameter of the network is 15 hops. A metric of 16 hops in RIP is considered a poison route or infinity metric.

If multiple paths exist to a particular destination, RIP will load balance between those paths, by default, up to 4, only if the metric (hopcount) is equal. RIP uses a round-robin system of load-balancing between equal metric routes, which can lead to pinhole congestion.

For example, two paths might exist to a particular destination, one going through a 9600 baud link, the other via a T1. If the metric (hopcount) is equal, RIP will load-balance, sending an equal amount of traffic down the 9600 baud link and the T1. This will cause the slower link to become congested.

21.1.2 RIP versions

RIP has two versions, Version 1 (RIPv1) and Version2 (RIPv2).

RIPv1 (RFC 1058) is classful, and therefore does not include the subnet mask with its routing table updates. Because of this, RIPv1 does not support Variable Length Subnet Masks (VLSMs). When using RIPv1, networks must be contiguous, and subnets of a major network must be configured with identical subnet masks. Otherwise, route table inconsistencies or worse will occur.

RIPv1 sends updates as broadcasts to address 255.255.255.255.

RIPv2 (RFC 2453) is classless, and therefore does include the subnet mask with its routing table updates. RIPv2 fully supports VLSMs, allowing discontinuous networks and varying subnet masks to exist.

Other enhancements offered by RIPv2 include:

- Routing updates are sent via multicast, using address 224.0.0.9
- Encrypted authentication can be configured between RIPv2 routers
- Route tagging is supported

RIPv2 can interoperate with RIPv1. By default:

- RIPv1 routers will send only Version 1 packets
- RIPv1 routers will receive both Version 1 and 2 updates
- RIPv2 routers will both send and receive only Version 2 updates

SATEL **ripd** package supports RIP version 2 as described in RFC2453 and RIP version 1 as described in RFC1058. It is part of Quagga suite of applications for routing.

21.2 Configuration package used

Package	Sections
ripd	routing interface key_chain offset

21.3 Configuring RIP using the web interface

To configure RIP using the web interface, select **Network->RIP**. The RIP page appears.

There are four sections in the RIP page.

Section	Description
Global Settings	Enables RIP and configures the RIP routing section containing global configuration parameters. The web automatically names the routing section ripd
Interfaces Configuration	Configures the interface sections. Defines interface configuration for RIP and interface specific parameters.
Offset Configuration	Configures the offset sections for metric manipulation.
MD5 Authentication Key Chains	Configures the key_chain sections. Defines MD5 authentication settings.

21.3.1 Global settings

The web browser automatically names the routing section 'ripd'.

Figure 94: The RIP global settings configuration page

Web Field/UCI/Package Option	Description				
Web: RIP Enabled UCI: ripd.ripd.enabled Opt: enabled	Enables RIP advertisements on router. <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: RIP Version UCI: ripd.ripd.version Opt: version	Specifies the RIP version that will be used. Version 2 is recommended. <table border="1"> <tr> <td>1</td> <td>RIP version 1</td> </tr> <tr> <td>2</td> <td>RIP version 2</td> </tr> </table>	1	RIP version 1	2	RIP version 2
1	RIP version 1				
2	RIP version 2				
Web: Network/Interface UCI: ripd.ripd.network Opt: list network	Defines the list of the interfaces that will be used to advertise RIP packets. Format: A.B.C.D/mask or interface name Multiple RIP interfaces are entered using uci set and uci add_list commands. Example: uci set ripd.ripd.network=lan1 uci add_list ripd.ripd.network=lan2 or using a list of options via package options list network 'lan1' list network 'lan2'				

Web: RIP Neighbor Address UCI: ripd.ripd.neighbor Opt: list neighbor	Specifies the list of RIP neighbours. When a neighbour doesn't understand multicast, this command is used to specify neighbours. In some cases, not all routers will be able to understand multicasting, where packets are sent to a network or a group of addresses. In a situation where a neighbour cannot process multicast packets, it is necessary to establish a direct link between routers. The neighbour command allows the network administrator to specify a router as a RIP neighbour. Multiple RIP neighbours are entered using <code>uci set</code> and <code>uci add_list</code> commands. Example: <code>uci set ripd.ripd.neighbor=1.1.1.1</code> <code>uci add_list ripd.ripd.neighbor=2.2.2.2</code> or using a list of options via package options <code>list neighbor '1.1.1.1'</code> <code>list neighbor '2.2.2.2'</code>				
Web: Update Timer UCI: ripd.ripd.tb_update_sec Opt: tb_update_sec	Every update timer seconds, the RIP process is awakened to send an unsolicited response message containing the complete routing table to all neighbouring RIP routers.				
	<table border="1"> <tr> <td>30</td><td>Default</td></tr> <tr> <td>Range</td><td></td></tr> </table>	30	Default	Range	
30	Default				
Range					
Web: Timeout Timer UCI: ripd.ripd.tb_timeout_sec Opt: tb_timeout_sec	Defines timeout in seconds. Upon expiration of the timeout, the route is no longer valid; however, it is retained in the routing table for a short time so that neighbours can be notified that the route has been dropped.				
	<table border="1"> <tr> <td>180</td><td>Default</td></tr> <tr> <td>Range</td><td></td></tr> </table>	180	Default	Range	
180	Default				
Range					
Web: Garbage Collect Timer UCI: ripd.ripd.tb_garbage_sec Opt: tb_garbage_sec	Upon expiration of the <i>Garbage-Collection</i> timer, the route is finally removed from the routing table. This timer starts when <i>Timeout</i> timer expires or when route is advertised as "unreachable". The reason for using this two-stage removal method (marking-deleting) is to give the router that declared the route no longer reachable a chance to propagate this information to other routers. When the timer expires the route is deleted. If during the garbage collection period a new <i>RIP Response</i> for the route is received, then the deletion process is aborted: the <i>Garbage-Collection</i> timer is cleared, the route is marked as valid again, and a new <i>Timeout</i> timer starts.				
	<table border="1"> <tr> <td>120</td><td>Default</td></tr> <tr> <td>Range</td><td></td></tr> </table>	120	Default	Range	
120	Default				
Range					
Web: Make Default Router UCI: ripd.ripd.default_info_originate Opt: default_info_originate	Advertising a default route via RIP.				
	<table border="1"> <tr> <td>0</td><td>Disable.</td></tr> <tr> <td>1</td><td>Enable.</td></tr> </table>	0	Disable.	1	Enable.
0	Disable.				
1	Enable.				
Web: Redistribute Kernel Routes UCI: ripd.ripd.redistribute_kernel_routes Opt: redistribute_kernel_routes	Redistributes routing information from kernel route entries into the RIP tables.				
	<table border="1"> <tr> <td>0</td><td>Disable.</td></tr> <tr> <td>1</td><td>Enable.</td></tr> </table>	0	Disable.	1	Enable.
0	Disable.				
1	Enable.				
Web: n/a UCI: ripd.ripd.vty_enabled Opt: vty_enabled	Enable vty for RIPvD (telnet to localhost:2602).				

Table 57: Information table for RIP global settings

21.3.2 Offset configuration

This section is used for RIP metric manipulation. RIP metric is a value for distance in the network. Usually, ripd package increments the metric when the network information is received. Redistributed routes' metric is set to 1.

The screenshot shows a configuration page titled "Offset Configuration". It contains two input fields: "Metric" with the value "1" and "Match" with the value "1.1.1.0/24". There are "Add" and "Delete" buttons at the bottom.

Figure 95: The RIP global settings configuration page

Web Field/UCI/Package Option	Description				
Web: Metric UCI: ripd.@offset[0].metric Opt: metric	Defines the metric offset value. This modifies the default metric value for redistributed and connected routes. <table border="1"><tr><td>1</td><td></td></tr><tr><td>Range</td><td></td></tr></table>	1		Range	
1					
Range					
Web: Match UCI: ripd.@offset[0].match_network Opt: match_network	Defines the prefixes to match. Format: A.B.C.D/mask				

Table 58: Information table for RIP offset commands

21.3.3 Interfaces configuration

The screenshot shows a configuration page titled "Interfaces Configuration". It lists three interfaces: "lan", "lan2", and "lan3". Each interface has checkboxes for "Split Horizon" and "Poison Reverse", and dropdown menus for "Authentication" (with options "no", "text", and "md5"). There are also "Text Auth. Key" and "MD5 Key Chain Name" fields. "Delete" and "Add" buttons are present.

Figure 96: The RIP interfaces configuration page

Web Field/UCI/Package Option	Description				
Web: Interface UCI: ripd.@interface[0].rip_interface Opt: rip_interface	Specifies the interface name.				
Web: Split Horizon UCI: ripd.@interface[0].split_horizon Opt: split_horizon	Prohibits the router from advertising a route back onto the interface from which it was learned. <table border="1"><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr></table>	0	Disable	1	Enable
0	Disable				
1	Enable				
Web: Poison Reverse UCI: ripd.@interface[0].poison_reverse Opt: poison_reverse	Router tells its neighbour gateways that one of the gateways is no longer connected. Notifies the gateway, setting the hop count to the unconnected gateway to 16 which would mean "infinite". <table border="1"><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr></table>	0	Disable	1	Enable
0	Disable				
1	Enable				

Web: Passive UCI: ripd.@interface[0].passive Opt: passive	Sets the specified interface to passive mode. On passive mode interface, all receiving packets are processed as normal and ripd does not send either multicast or unicast RIP packets except to RIP neighbour specified with a neighbour command.						
	<table border="1"> <tr> <td>0</td><td>Disable</td></tr> <tr> <td>1</td><td>Enable</td></tr> </table>	0	Disable	1	Enable		
0	Disable						
1	Enable						
Web: Authentication UCI: ripd.@interface[0].auth_mode Opt: auth_mode	RIPv2 (only) allows packets to be authenticated via either an insecure plain text password, included with the packet, or via a more secure MD5 based HMAC (keyed-Hashing for Message AuthentiCation). Enabling authentication prevents routes being updated by unauthenticated remote routers, but still can allow routes, that is, the entire RIP routing table, to be queried remotely, potentially by anyone on the internet, via RIPv1.						
	<table border="1"> <tr> <td>no</td><td>Default value. No authentication.</td></tr> <tr> <td>md5</td><td>Sets the interface with RIPv2 MD5 authentication.</td></tr> <tr> <td>text</td><td>Sets the interface with RIPv2 simple password authentication.</td></tr> </table>	no	Default value. No authentication.	md5	Sets the interface with RIPv2 MD5 authentication.	text	Sets the interface with RIPv2 simple password authentication.
no	Default value. No authentication.						
md5	Sets the interface with RIPv2 MD5 authentication.						
text	Sets the interface with RIPv2 simple password authentication.						
Web: Text Auth. Key UCI: ripd.@interface[0].auth_key Opt: auth_key	This command sets the authentication string for text authentication. The string must be shorter than 16 characters.						
Web: MD5 Key Chain Name UCI: ripd.@interface[0].key_chain Opt: key_chain	Specify Keyed MD5 chain.						

Table 59: Information table for RIP interface configuration

21.3.4 MD5 authentication key chains

RIPv2 (only) allows packets to be authenticated using either an insecure plain text password, included with the packet, or by a more secure MD5 based HMAC (keyed-Hashing for Message AuthentiCation). Enabling authentication prevents routes being updated by unauthenticated remote routers, but still can allow routes, that is, the entire RIP routing table, to be queried remotely, potentially by anyone on the internet, using RIPv1.

This section defines key_chains to be used for MD5 authentication

MD5 Authentication Key Chains

Key Chain Name: Delete

Key ID: e.g. 1, 2... Must be unique and match at both ends

Authentication key:

Figure 97: The MD5 authentication key chains configuration section

Web Field/UCI/Package Option	Description
Web: Key Chain Name UCI: ripd.@key_chain[0].key_chain_name Opt: key_chain_name	Specifies chain name
Web: Key ID UCI: ripd.@key_chain[0].key_id Opt: key_id	Specifies key ID. Must be unique and match at both ends.
Web: Authentication key UCI: ripd.@key_chain[0].auth_key Opt: auth_key	Specify Keyed MD5 chain.

Table 60: Information table for MD5 authentication Key Chains commands

21.4 Configuring RIP using command line

RIP is configured under the ripd package **/etc/config/ripd**.

There are four config sections ripd, interface, key_chain and offset.

You can configure multiple interface, key_chain and offset sections.

By default, all RIP interface instances are named interface, it is identified by @interface then the interface position in the package as a number. For example, for the first interface in the package using UCI:

```
ripd.@interface[0]=interface
ripd.@interface[0].rip_interface=lan
```

Or using package options:

```
config interface
    option rip_interface 'lan'
```

By default, all RIP key_chain instances are named key_chain, it is identified by @key_chain then the key_chain position in the package as a number. For example, for the first key_chain in the package using UCI:

```
ripd.@key_chain[0]=key_chain
ripd.@key_chain[0].key_chain_name=Keychain1
```

Or using package options:

```
config key_chain
    option key_chain_name 'Keychain1'
```

By default, all RIP offset instances are named offset, it is identified by @offset then the offset position in the package as a number. For example, for the first offset in the package using UCI:

```
ripd.@offset[0]=offset
ripd.@offset[0].metric=1
```

Or using package options:

```
config offset
    option metric '1'
```

21.4.1 RIP using UCI

```
root@GW_router:~# uci show ripd
ripd.ripd=routing
ripd.ripd.version=2
ripd.ripd.enabled=yes
ripd.ripd.network=lan2 gre1
ripd.ripd.neighbor=10.1.1.100 10.1.2.100
ripd.ripd.tb_update_sec=30
ripd.ripd.tb_timeout_sec=180
ripd.ripd.tb_garbage_sec=120
ripd.ripd.default_info_originate=yes
ripd.ripd.redistribute_kernel_routes=yes
ripd.@interface[0]=interface
ripd.@interface[0].rip_interface=lan
ripd.@interface[0].auth_mode=no
ripd.@interface[0].split_horizon=1
ripd.@interface[0].poison_reverse=0
ripd.@interface[0].passive=0
ripd.@interface[1]=interface
ripd.@interface[1].rip_interface=lan2
ripd.@interface[1].split_horizon=1
ripd.@interface[1].poison_reverse=0
ripd.@interface[1].passive=0
ripd.@interface[1].auth_mode=text
ripd.@interface[1].auth_key=secret
ripd.@interface[2]=interface
ripd.@interface[2].rip_interface=lan3
ripd.@interface[2].split_horizon=1
ripd.@interface[2].poison_reverse=0
ripd.@interface[2].passive=0
ripd.@interface[2].auth_mode=md5
ripd.@interface[2].key_chain=Keychain1
```

```

ripd.@key_chain[0]=key_chain
ripd.@key_chain[0].key_chain_name=Keychain1
ripd.@key_chain[0].key_id=1
ripd.@key_chain[0].auth_key=123
ripd.@offset[0]=offset
ripd.@offset[0].metric=1
ripd.@offset[0].match_network=10.1.1.1/24

```

21.4.2 RIP using package options

```

root@GW_router:~# uci export ripd
package ripd

config routing 'ripd'
    option version '2'
    option enabled 'yes'
    list network 'lan2'
    list network 'gre1'
    list neighbor '10.1.1.100'
    list neighbor '10.1.2.100'
    option tb_update_sec '30'
    option tb_timeout_sec '180'
    option tb_garbage_sec '120'
    option default_info_originate 'yes'
    option redistribute_kernel_routes 'yes'

config interface
    option rip_interface 'lan'
    option auth_mode 'no'
    option split_horizon '1'
    option poison_reverse '0'
    option passive '0'

config interface
    option rip_interface 'lan2'
    option split_horizon '1'
    option poison_reverse '0'

```

```

option passive '0'
option auth_mode 'text'
option auth_key 'textsecret'

config interface
    option rip_interface 'lan3'
    option split_horizon '1'
    option poison_reverse '0'
    option passive '0'
    option auth_mode 'md5'
    option key_chain 'keychain1'

config key_chain
    option key_chain_name 'Keychain1'
    option key_id '1'
    option auth_key '123'

config offset
    option metric '1'
    option match_network '10.1.1.1/24'

```

21.5 RIP diagnostics

21.5.1 Route status

To show the current routing status, enter `route -n`:

Destination	Gateway	Genmask	Flags	Metric	Ref	Use
0.0.0.0	10.205.154.65	0.0.0.0	UG	1	0	0
usb0						
10.1.0.0	0.0.0.0	255.255.0.0	U	0	0	0
eth1						
10.205.154.64	0.0.0.0	255.255.255.252	U	0	0	0
usb0						

11.11.11.0	0.0.0.0	255.255.255.248	U	0	0	0
gre-GRE						
89.101.154.151	10.205.154.65	255.255.255.255	UGH	0	0	0
usb0						
192.168.100.0	0.0.0.0	255.255.255.0	U	0	0	0
eth0						
192.168.104.1	11.11.11.4	255.255.255.255	UGH	3	0	0
gre-GRE						
192.168.154.154	11.11.11.1	255.255.255.255	UGH	2	0	0
gre-GRE						

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

21.5.2 Tracing RIP packets

RIP uses UDP port 520. To trace RIP packets on any interface on the router, enter:

```
tcpdump -i any -n -p port 520 &
```

```
root@GW_router:~# tcpdump -i any -n -p port 520 &
root@GW_router:~# tcpdump: verbose output suppressed, use -v or -vv for
full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked), capture size 65535
bytes
```

To stop tracing enter `fg` to bring tracing task to foreground, and then **<CTRL-C>** to stop the trace.

```
root@GW_router:~# fg
tcpdump -i any -n -p port 67
^C
33 packets captured
33 packets received by filter
0 packets dropped by kernel
```

21.5.3 Quagga/zebra console

Quagga is the routing protocol suite embedded in the router firmware. Quagga is split into different daemons for implementation of each routing protocol. Zebra is a core daemon for Quagga, providing the communication layer to the underlying Linux kernel, and routing updates to the client daemons.

Quagga has a console interface to Zebra for advanced debugging of the routing protocols.

To access, enter: `telnet localhost zebra (password: zebra)`

```
root@GW_router:~# telnet localhost zebra

Entering character mode
Escape character is '^]'.

Hello, this is Quagga (version 0.99.21).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:
```

To see RIP routing information from Zebra console, enter: `sh ip route`

```
root@GW_router:~# sh ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, H - HSLS, o - OLSR,
       b - BATMAN, A - Babel,
       > - selected route, * - FIB route

K>* 0.0.0.0/0 via 10.205.154.65, usb0
C>* 10.1.0.0/16 is directly connected, eth1
C>* 10.205.154.64/30 is directly connected, usb0
C>* 11.11.11.0/29 is directly connected, gre-GRE
K>* 89.101.154.151/32 via 10.205.154.65, usb0
C>* 127.0.0.0/8 is directly connected, lo
C>* 192.168.100.0/24 is directly connected, eth0
R>* 192.168.104.1/32 [120/3] via 11.11.11.4, gre-GRE, 15:54:47
C>* 192.168.105.1/32 is directly connected, lo
R>* 192.168.154.154/32 [120/2] via 11.11.11.1, gre-GRE, 16:09:51
```

21.5.4 RIP debug console

When option **tty_enabled** (see Global settings section above) is enabled in the RIP configuration, RIP debug console can be accessed for advanced RIP debugging.

To access RIP debug console enter: `telnet localhost ripd (password zebra)`

```
root@GW_router:~# telnet localhost ripd

Entering character mode
Escape character is '^]'.

Hello, this is Quagga (version 0.99.21).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:
```

To see RIP status from RIP debug console, enter: sh ip rip

```
root@GW_router:~# show ip rip
Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
      (n) - normal, (s) - static, (d) - default, (r) - redistribute,
      (i) - interface

      Network          Next Hop          Metric From        Tag Time
C(i) 11.11.11.0/29    0.0.0.0          1 self           0
R(n) 192.168.104.1/32 11.11.11.4       3 11.11.11.1     0 02:48
C(i) 192.168.105.1/32 0.0.0.0          1 self           0
R(n) 192.168.154.154/32 11.11.11.1     2 11.11.11.1     0 02:48
```

To see RIP status from RIP debug console, enter: sh ip rip status

```
root@GW_router:~# sh ip rip status
Routing Protocol is "rip"
  Sending updates every 30 seconds with +/-50%, next due in 17 seconds
  Timeout after 180 seconds, garbage collect after 120 seconds
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Default redistribution metric is 1
```

Redistributing:

```
Default version control: send version 2, receive version 2
```

Interface	Send	Recv	Key-chain
gre-GRE	2	2	
lo	2	2	

Routing for Networks:

```
11.0.0.0/8
```

```
192.168.105.1/32
```

Routing Information Sources:

Gateway	BadPackets	BadRoutes	Distance	Last Update
11.11.11.1	0	0	120	00:00:20

```
Distance: (default is 120)
```

22 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.

Multi-WAN periodically does a health check on the interface. A health check comprises of a configurable combination of the following:

- interface state
- pings to an ICMP target
- signal level checks using signal threshold, RSCP threshold and ECIO threshold option values

A fail for any of the above health checks, results in a fail. After a configurable number of health check failures, Multi-WAN will move to the next highest priority interface. Multi-WAN will optionally stop the failed interface and start the new interface, if required.

In some circumstances, particularly in mobile environments, it is desirable for a primary interface to be used whenever possible. In this instance Multi-WAN will perform a health check on the primary interface after a configurable period. If the health checks pass for the configured number of recovery health checks then the primary will be used.

22.1 Configuration package used

Package	Sections
multiwan	config wan

22.2 Configuring Multi-WAN using the web interface

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

Multi-WAN
Multi-WAN allows for the use of multiple uplinks for load balancing and failover.

Enable

Preempt

Alternate Mode ⓘ It will use alternate interface after reboot

Figure 98: 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 by ifup_retry_sec	
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 61: 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.

The name used for Multi-WAN must be identical, including upper and lowercases, to the actual 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 appears.

WAN Interfaces

Health Monitor detects and corrects network changes and failed connections.

WAN

Health Monitor Interval	10 sec.
Health Monitor ICMP Host(s)	DNS Server(s)
Health Monitor Conntrack Test Host(s)	Default
Health Monitor ICMP Timeout	3 sec.
Health Monitor ICMP Interval	1 sec.
Attempts Before WAN Failover	3
Attempts Before WAN Recovery	5
Priority	0 <small>Higher value is higher priority</small>
Exclusive Group	0 <small>Only one interface in group could be up in the same time</small>
Manage Interface State (Up/Down)	<input checked="" type="checkbox"/>
Minimum ifup Interval	300 sec. <small>Minimum interval between two successive interface start attempts</small>
Interface Start Timeout	40 sec. <small>Time for interface to startup</small>
Signal Threshold (dBm)	-115 <small>Below is a failure</small>
RSCP Threshold for 3G (dBm)	-115 <small>Below is a failure</small>
ECIO Threshold for 3G (dB)	-115 <small>Below is a failure</small>
Signal Test	<input type="text"/> <small>Free form expression to test signal value</small>

Figure 99: 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	<p>Sets the period to check the health status of the interface. The Health Monitor interval will be used for:</p> <ul style="list-style-type: none"> interface state checks Ping interval Signal strength checks 								
Web: Health Monitor ICMP Host(s) UCI: multiwan.wan.icmp_hosts Opt: icmp_hosts	<p>Sends health ICMPs to configured value DNS servers by default. Configure to any address.</p> <table border="1" data-bbox="692 451 1394 702"> <tr> <td data-bbox="692 451 886 489">Disable</td><td data-bbox="886 451 1394 489">Disables the option.</td></tr> <tr> <td data-bbox="692 489 886 527">DNS servers</td><td data-bbox="886 489 1394 527">DNS IP addresses will be used.</td></tr> <tr> <td data-bbox="692 527 886 565">WAN Gateway</td><td data-bbox="886 527 1394 565">Gateway IP address will be used.</td></tr> <tr> <td data-bbox="692 565 886 702">Custom</td><td data-bbox="886 565 1394 702">Ability to provide IP address. Multiple pings targets can be entered, comma separated. Pings to both must fail for health check to fail. Example: option icmp_hosts '1.1.1.1,2.2.2.2'</td></tr> </table>	Disable	Disables the option.	DNS servers	DNS IP addresses will be used.	WAN Gateway	Gateway IP address will be used.	Custom	Ability to provide IP address. Multiple pings targets can be entered, comma separated. Pings to both must fail for health check to fail. Example: option icmp_hosts '1.1.1.1,2.2.2.2'
Disable	Disables the option.								
DNS servers	DNS IP addresses will be used.								
WAN Gateway	Gateway IP address will be used.								
Custom	Ability to provide IP address. Multiple pings targets can be entered, comma separated. Pings to both must fail for health check to fail. Example: option icmp_hosts '1.1.1.1,2.2.2.2'								
Web: Health Monitor Conntrack Test Host(s) UCI: multiwan.wan.conntrack_hosts Opt: conntrack_hosts	<p>Conntrack is the feature used to track if there is any traffic to and from an IP destination within the health interval.</p> <p>The Conntrack_hosts option defines the IP for conntrack to track, usually the icmp_host IP is used.</p> <p>If traffic to the conntrack_hosts IP is detected then multiwan does not send a ping health check to the icmp_host; otherwise a ping is sent as normal to the icmp_host.</p> <p>By default the conntrack_hosts is checked if the health interval is greater than 5 minutes. This time threshold currently cannot be manipulated.</p> <p>Conntrack is generally used to limit the traffic sent on a GSM network.</p> <table border="1" data-bbox="692 1057 1394 1221"> <tr> <td data-bbox="692 1057 822 1125">Default</td><td data-bbox="822 1057 1394 1125">Conntrack checks for traffic from icmp_host IP when health_interval is greater than 5 minutes.</td></tr> <tr> <td data-bbox="692 1125 822 1163">Disable</td><td data-bbox="822 1125 1394 1163">Conntrack disabled.</td></tr> <tr> <td data-bbox="692 1163 822 1221">Custom</td><td data-bbox="822 1163 1394 1221">Specifies an IP other than the icmp_host for conntrack to track.</td></tr> </table>	Default	Conntrack checks for traffic from icmp_host IP when health_interval is greater than 5 minutes.	Disable	Conntrack disabled.	Custom	Specifies an IP other than the icmp_host for conntrack to track.		
Default	Conntrack checks for traffic from icmp_host IP when health_interval is greater than 5 minutes.								
Disable	Conntrack disabled.								
Custom	Specifies an IP other than the icmp_host for conntrack to track.								
Web: Health Monitor ICMP Timeout UCI: multiwan.wan.timeout Opt: timeout	<p>Sets Ping timeout in seconds. Choose the time in seconds that the health monitor ICMP will timeout at.</p> <table border="1" data-bbox="692 1282 1394 1349"> <tr> <td data-bbox="692 1282 822 1320">3</td><td data-bbox="822 1282 1394 1320">Wait 3 seconds for ping reply</td></tr> <tr> <td data-bbox="692 1320 822 1349">Range</td><td data-bbox="822 1320 1394 1349"></td></tr> </table>	3	Wait 3 seconds for ping reply	Range					
3	Wait 3 seconds for ping reply								
Range									
Web: Health Monitor ICMP Interval UCI: multiwan.wan.icmp_interval Opt: icmp_interval	<p>Defines the interval between multiple pings sent at each health check</p> <table border="1" data-bbox="692 1417 1394 1484"> <tr> <td data-bbox="692 1417 822 1455">1</td><td data-bbox="822 1417 1394 1455"></td></tr> <tr> <td data-bbox="692 1455 822 1484">Range</td><td data-bbox="822 1455 1394 1484"></td></tr> </table>	1		Range					
1									
Range									
Web: Health Monitor ICMP Count UCI: multiwan.wan.icmp_count Opt: icmp_count	<p>Defines the number of pings to send at each health check.</p> <table border="1" data-bbox="692 1545 1394 1596"> <tr> <td data-bbox="692 1545 822 1583">1</td><td data-bbox="822 1545 1394 1583"></td></tr> <tr> <td data-bbox="692 1583 822 1596">Range</td><td data-bbox="822 1583 1394 1596"></td></tr> </table>	1		Range					
1									
Range									
Web: Attempts Before WAN Failover UCI: multiwan.wan.health_fail_retries Opt: health_fail_retries	<p>Sets the amount of health monitor retries before interface is considered a failure.</p> <table border="1" data-bbox="692 1664 1394 1731"> <tr> <td data-bbox="692 1664 822 1702">3</td><td data-bbox="822 1664 1394 1702"></td></tr> <tr> <td data-bbox="692 1702 822 1731">Range</td><td data-bbox="822 1702 1394 1731"></td></tr> </table>	3		Range					
3									
Range									
Web: Attempts Before WAN Recovery UCI: multiwan.wan.health_recovery_retries Opt: health_recovery_retries	<p>Sets the number of health monitor checks before the interface is considered healthy. Only relevant if pre-empt mode is enabled.</p> <table border="1" data-bbox="692 1792 1394 1866"> <tr> <td data-bbox="692 1792 822 1830">5</td><td data-bbox="822 1792 1394 1830"></td></tr> <tr> <td data-bbox="692 1830 822 1866">Range</td><td data-bbox="822 1830 1394 1866"></td></tr> </table>	5		Range					
5									
Range									
Web: Priority UCI: multiwan.wan.priority Opt: priority	<p>Specifies the priority of the interface. The higher the value, the higher the priority.</p> <table border="1" data-bbox="692 1927 1394 1996"> <tr> <td data-bbox="692 1927 822 1965">0</td><td data-bbox="822 1927 1394 1965"></td></tr> <tr> <td data-bbox="692 1965 822 1996">Range</td><td data-bbox="822 1965 1394 1996"></td></tr> </table>	0		Range					
0									
Range									

Web: Manage Interface State (Up/Down) UCI: multiwan.wan.manage_state Opt: manage_state	Defines whether multi-wan will start and stop the interface. <table border="1"> <tr><td>1</td><td>Enabled.</td></tr> <tr><td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.												
1	Enabled.																
0	Disabled.																
Web: Exclusive Group UCI: multiwan.wan.exclusive_group Opt: exclusive_group	Defines the group to which the interface belongs, only one interface can be active. <table border="1"> <tr><td>0</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	0		Range													
0																	
Range																	
Web: Minimum ifup Interval UCI: multiwan.wan.ifup_retry_sec Opt: ifup_retry_sec	Specifies the interval in seconds before retrying the primary interface when pre-empt mode is enabled. <table border="1"> <tr><td>300</td><td>Retry primary interface every 300 seconds.</td></tr> <tr><td>Range</td><td></td></tr> </table>	300	Retry primary interface every 300 seconds.	Range													
300	Retry primary interface every 300 seconds.																
Range																	
Web: Interface Start Timeout UCI: multiwan.wan.ifup_timeout Opt: ifup_timeout	Specifies the time in seconds for interface to start up. If it is not up after this period, it will be considered a fail. <table border="1"> <tr><td>40</td><td>40 seconds.</td></tr> <tr><td>Range</td><td></td></tr> </table>	40	40 seconds.	Range													
40	40 seconds.																
Range																	
Web: Signal Threshold (dBm) UCI: multiwan.wan.signal_threshold Opt: signal_threshold	Specifies the minimum signal strength in dBm before considering if the interface fails signal health check. Uses the value stored for sig_dbm in mobile diagnostics.-115. <table border="1"> <tr><td></td><td>Disabled</td></tr> <tr><td>Range</td><td>-46 to -115 dBm</td></tr> </table>		Disabled	Range	-46 to -115 dBm												
	Disabled																
Range	-46 to -115 dBm																
Web: RSCP Threshold (dBm) UCI: multiwan.wan.rscp_threshold Opt: rscp_threshold	Specifies the minimum RSCP signal strength in dBm before considering if the interface fails signal health check. Uses the value stored for rscp_dbm in mobile diagnostics. <table border="1"> <tr><td>-115</td><td>Disabled</td></tr> <tr><td>Range</td><td>-46 to -115 dBm</td></tr> </table>	-115	Disabled	Range	-46 to -115 dBm												
-115	Disabled																
Range	-46 to -115 dBm																
Web: ECIO Threshold (dB) UCI: multiwan.wan.ecio_threshold Opt: ecio_threshold	Specifies the minimum ECIO signal strength in dB before considering if the interface fails signal health check. Uses the value stored for ecio_db in mobile diagnostics. <table border="1"> <tr><td>-115</td><td>Disabled</td></tr> <tr><td>Range</td><td>-46 to -115 dB</td></tr> </table>	-115	Disabled	Range	-46 to -115 dB												
-115	Disabled																
Range	-46 to -115 dB																
Web: Signal Test UCI: multiwan.wan.signal_test Opt: signal_test	Defines a script to test various signal characteristics in multiwan signal test. For example: <pre>option signal_test '(tech == 0) then (sig_dbm > -70) else (rscp_dbm > -105 and ecio_db > -15)'</pre> <p>This states that when technology is GSM, a health fail is determined when signal strength is less than -70dBm. When technology is not GSM a health fail occurs when either rscp_dbm falls below -105dBm or ecio_db falls below -15dB</p> <p>Tech values are:</p> <table border="1"> <tr><td>0</td><td>GSM</td></tr> <tr><td>1</td><td>GSM Compact</td></tr> <tr><td>2</td><td>UTRAN</td></tr> <tr><td>3</td><td>GSM w/EGPRS</td></tr> <tr><td>4</td><td>UTRAN w/HSPDA</td></tr> <tr><td>5</td><td>UTRAN w/HSUPA</td></tr> <tr><td>6</td><td>UTRAN w/HSUPA and HSDPA</td></tr> <tr><td>7</td><td>E-UTRAN</td></tr> </table>	0	GSM	1	GSM Compact	2	UTRAN	3	GSM w/EGPRS	4	UTRAN w/HSPDA	5	UTRAN w/HSUPA	6	UTRAN w/HSUPA and HSDPA	7	E-UTRAN
0	GSM																
1	GSM Compact																
2	UTRAN																
3	GSM w/EGPRS																
4	UTRAN w/HSPDA																
5	UTRAN w/HSUPA																
6	UTRAN w/HSUPA and HSDPA																
7	E-UTRAN																

Table 62: Information table for multi-WAN interface page

22.3 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 100: The multi-WAN traffic rules page

22.4 Configuring Multi-WAN using UCI

Multi-WAN UCI configuration settings are stored on **/etc/config/multiwan**

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

```
root@GW_router:~# 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 health_fail_retries '3'
    option health_recovery_retries '5'
    option priority '2'
    option manage_state 'yes'
    option exclusive_group '0'
    option ifup_retry_sec '40'
    option icmp_hosts 'disable'
    option icmp_interval '1'
    option timeout '3'
```

```

option icmp_count '1'
option conntrack_hosts 'disable'          option signal_threshold '-111'
option rscp_threshold '-90'
option ecio_threshold '-15'
option ifup_timeout_sec '120'

root@GW_router:~# 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=10multiwan.wan.health_fail_retries=3
multiwan.wan.health_recovery_retries=5
multiwan.wan.priority=2
multiwan.wan.manage_state=yes
multiwan.wan.exclusive_group=0
multiwan.wan.ifup_retry_sec=36000
multiwan.wan.icmp_hosts=disable
multiwan.wan.timeout=3
multiwan.wan.icmp_interval '1'
multiwan.wan.timeout '3'
multiwan.wan.icmp_count '1'
multiwan.wan.conntrack_hosts 'disable'
multiwan.wan.signal_threshold=-111
multiwan.wan.rscp_threshold=-90
multiwan.wan.ecio_threshold=-15

```

22.5 Multi-WAN diagnostics

The multiwan 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@GW_router:~# uci export 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@GW_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.

23 Automatic operator selection

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

23.1 Configuration package used

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

23.2 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 multiwan package is used to run failover between interfaces. Typically these auto-generated interfaces are sorted by signal strength.

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.

Multi-WAN periodically does a health check on the active interface. A health check comprises of a configurable combination of the following:

- interface state
- pings to an ICMP target
- signal level checks using signal threshold, RSCP threshold and ECIO threshold option values

A fail for any of the above health checks results in an overall fail. After a configurable number of health check failures, multiwan will move to the next highest priority interface. Multi-WAN will optionally stop the failed interface and start the new interface, if required.

In some circumstances, particularly in mobile environments, it is desirable for a primary interface to be used whenever possible. In this instance, if the active interface is not the primary interface, multiwan will perform a health check on the primary interface after a configurable period. If the health checks pass for the configured number of recovery health checks then the primary interface will be used.

There are typically three scenarios:

- Primary Mobile Provider (PMP) + roaming: pre-empt enabled
- PMP + roaming: pre-empt disabled
- No PMP + roaming

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

23.2.1.1 Overview

In this scenario, the PMP interface is used whenever possible.

The PMP interface is attempted first. When the health checks fail on the PMP interface, and Multi-WAN moves to an autogenerated interface, a timer is started `multiwan option ifup_retry_sec`. On expiration of this timer, multiwan will disconnect the current interface and retry the PMP interface.

The PMP interface will then be used if the configurable number of health checks pass the checks.

23.2.1.2 Software operation

6. Multiwan first attempts to bring up the PMP interface. If the PMP interface connects within the time set by multiwan option `ifup_timeout` continue to step 2. Otherwise go to step 4.
7. A health check is periodically done on the PMP interface as determined by the multiwan option `health_interval`. If the health check fails for the number of retries (multiwan option `health_fail_retries`), disconnect the PMP interface.
8. Connect the first auto-generated interface.
9. If the interface connects within the time set by multiwan option `ifup_timeout` continue to step 5, otherwise multiwan moves to the next auto-generated interface.
10. Wait until the health check fails on the auto-generated interface, or until the PMP interface is available to connect after it was disconnected in step 2. (multiwan option `ifup_retry_sec`).
11. Disconnect auto-generated interface.
12. If the interface was disconnected due to health check failure then connect the next auto-generated interface and repeat step 4. If the interface was disconnected because `ifup_retry_sec` of PMP interface timed out, then go back to step 1 and repeat the process.

The PMP predefined interface is defined in the network package. Ensure the interface name matches the interface name defined in the multiwan package.

23.2.1.3 Create a primary predefined interface

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

The screenshot shows the 'Interfaces' section of a web-based configuration interface. At the top, there's a 'LAN' header. Below it, the 'Interfaces' section has a 'Interface Overview' heading. A table lists two interfaces:

Network	Status	Actions
LAN eth0	Uptime: 6h 37m 34s MAC Address: 00:E0:C8:10:0E:E6 RX: 431.31 MB (4672877 Pkts.) TX: 1.68 MB (21023 Pkts.) IPv4: 10.1.10.93/16	<input type="button" value="Connect"/> <input type="button" value="Stop"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>
LOOPBACK lo	Uptime: 6h 37m 38s MAC Address: 00:00:00:00:00:00 RX: 9.99 MB (109997 Pkts.) TX: 9.99 MB (109997 Pkts.) IPv4: 127.0.0.1/8 IPv6: 0:0:0:0:0:0:0:1/128	<input type="button" value="Connect"/> <input type="button" value="Stop"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>

At the bottom left of the table area is a button labeled 'Add new interface...'. The entire interface is enclosed in a light gray border.

Figure 101: The interface overview page

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

The screenshot shows the 'Create Interface' page. It includes the following fields:

- Name of the new interface: A text input field with a note: "The allowed characters are: A-Z, a-z, 0-9 and _".
- Protocol of the new interface: A dropdown menu set to "Static address".
- Create a bridge over multiple interfaces: A checkbox.
- Cover the following interface: A radio button group with four options:
 - Ethernet Adapter: "eth0" (lan)
 - Ethernet Adapter: "gre0"
 - Ethernet Adapter: "lo" (loopback)
 - Custom Interface: An input field for entering a custom interface name.
- Note: If you choose an interface here which is part of another network, it will be moved into this network.

Figure 102: The create interface page

Web Field/UCI/Package Option	Description
Web: Name of the new interface UCI: network.3g_s<sim-number>_<short-operator-name>. Opt: 3g_s<sim-number>_<short-operator-name>.	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:
Operator name	First four alphanumeric numbers
Vodafone UK	voda
O2 – UK	o2uk
Orange	oran

Web: Protocol of the new interface UCI: network.[..x..].proto Opt: proto	Protocol type. Select LTE/UMTS/GPRS/EV-DO . <table border="1"> <thead> <tr> <th>Option</th><th>Description</th></tr> </thead> <tbody> <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> </tbody> </table>	Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	IPv6-in-IPv4 (RFC4213)	IPv4 tunnels that carry IPv6.	IPv6 over IPv4	IPv6 over IPv4 tunnel.	GRE	Generic Routing Encapsulation.	IOT		L2TP	Layer 2 Tunnelling Protocol.	PPP	Point to Point Protocol.	PPPoE	Point to Point Protocol over Ethernet.	PPPoATM	Point to Point Protocol over ATM.	LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.
Option	Description																										
Static	Static configuration with fixed address and netmask.																										
DHCP Client	Address and netmask are assigned by DHCP.																										
Unmanaged	Unspecified																										
IPv6-in-IPv4 (RFC4213)	IPv4 tunnels that carry IPv6.																										
IPv6 over IPv4	IPv6 over IPv4 tunnel.																										
GRE	Generic Routing Encapsulation.																										
IOT																											
L2TP	Layer 2 Tunnelling Protocol.																										
PPP	Point to Point Protocol.																										
PPPoE	Point to Point Protocol over Ethernet.																										
PPPoATM	Point to Point Protocol over ATM.																										
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																										
Web: Create a bridge over multiple interfaces UCI: network.[..x..].typeOpt: type	Enables bridge between two interfaces. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.																						
0	Disabled.																										
1	Enabled.																										
Web: Cover the following interface UCI: network.[..x..].ifname Opt: ifname	Select interfaces for bridge connection.																										

Table 63: Information table for the create interface page

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

Common Configuration

General Setup **Advanced Settings** Physical Settings Firewall Settings

Status: 3g-3g_s2_voda RX: 0.00 B (0 Pkts.)
TX: 0.00 B (0 Pkts.)

Protocol: UMTS/GPRS/EV-DO

Service Type: UMTS/GPRS

SIM: 1

APN: internet

PIN:

PAP/CHAP username: internet

PAP/CHAP password:

[Back to Overview](#) **Save & Apply** **Save** **Reset**

Figure 103: The common configuration page

Web Field/UCI/Package Option	Description	
Web: Protocol UCI: network.[..x..].proto Opt: proto	Protocol type. Select LTE/UMTS/GPRS/EV-DO .	
Option	Description	
Static	Static configuration with fixed address and netmask.	
DHCP Client	Address and netmask are assigned by DHCP.	
Unmanaged	Unspecified	
IPv6-in-IPv4 (RFC4213)	IPv4 tunnels that carry IPv6.	
IPv6 over IPv4	IPv6 over IPv4 tunnel.	
GRE	Generic Routing Encapsulation.	
IOT		
L2TP	Layer 2 Tunnelling Protocol.	
PPP	Point to Point Protocol.	
PPPoE	Point to Point Protocol over Ethernet.	
PPPoATM	Point to Point Protocol over ATM.	
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.	
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: network.[..x..].apn Opt: apn	APN name of Mobile Network Operator.	
Web: APN username UCI: network.[..x..].username Opt: username	Username used to connect to APN.	
Web: APN password UCI: network.[..x..].password Opt: password	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 64: Information table for the general set up sectionClick **Save & Apply**.

23.2.1.4 Set multi-WAN options for primary predefined interface

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

The screenshot shows the 'Multi-WAN' configuration page. At the top, a header reads 'Multi-WAN' with a sub-instruction: 'Multi-WAN allows for the use of multiple uplinks for failover.' Below this is a button labeled 'Add'. The main section is titled 'WAN Interfaces' with the sub-instruction: 'Health Monitor detects and corrects network changes and failed connections.' A note below states: 'This section contains no values yet.' There is an empty input field followed by an 'Add' button. At the bottom right are three buttons: 'Save & Apply' (blue), 'Save' (blue), and 'Reset' (grey).

Figure 104: 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.

Multi-WAN

Multi-WAN allows for the use of multiple uplinks for failover.

Enable

Preempt

Alternate Mode ⓘ It will use alternate interface after reboot

WAN Interfaces

Health Monitor detects and corrects network changes and failed connections.

Delete

3G_S1_VODA

Health Monitor Interval: 10 sec.

Health Monitor ICMP Host(s): DNS Server(s)

Health Monitor ICMP Timeout: 3 sec.

Attempts Before WAN Failover: 3

Attempts Before WAN Recovery: 5

Priority: 0 ⓘ Higher value is higher priority

Manage Interface State (Up/Down)

Exclusive Group: 0 ⓘ Only one interface in group could be up in the same time

Minimum ifup Interval: 300 sec. ⓘ Minimum interval between two successive interface start attempts

Interface Start Timeout: 40 sec. ⓘ Time for interface to startup

Signal Threshold (dBm): 115 ⓘ Below is a failure

Add

Save & Apply Save Reset

Figure 105: The multi-WAN page

Web Field/UCI/Package Option	Description				
Web: Enable UCI: multiwan.config.enabled Opt: enabled	Enables multiwan. <table border="1"><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Preempt UCI: multiwan.config.preempt Opt: preempt	Enables or disables pre-emption for multiwan. If enabled the router will keep trying to connect to a higher priority interface depending on timer set. <table border="1"><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Alternate Mode UCI: multiwan.config.alt Opt: alt	Enables or disables alternate mode for multiwan. If enabled the router will use an alternate interface after reboot. <table border="1"><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: WAN Interfaces UCI: multiwan.3g_s<sim-number>_<short-operator-name> Opt: 3g_s<sim-number>_<short-operator-name>	Provide the same interface name as chosen in multiwan section below and click Add.				

Web: Health Monitor Interval UCI: multiwan.[..x..].health_interval Opt: health_interval	Sets the period to check the health status of the interface. The Health Monitor interval will be used for: interface state checks ping interval signal strength checks								
Web: Health Monitor ICMP Host(s) UCI: multiwan.[..x..].icmp_hosts Opt: icmp_hosts	Specifies target IP address for ICMP packets. <table border="1"> <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>	Disable	Disables the option.	DNS servers	DNS IP addresses will be used.	WAN Gateway	Gateway IP address will be used.	custom	Ability to provide IP address.
Disable	Disables the option.								
DNS servers	DNS IP addresses will be used.								
WAN Gateway	Gateway IP address will be used.								
custom	Ability to provide IP address.								
Web: Health Monitor Conntrack Test Host(s) UCI: multiwan.wan.conntrack_hosts Opt: conntrack_hosts	Conntrack is the feature used to track if there is any traffic to and from an IP destination within the health interval. Conntrack_hosts option defines the IP for conntrack to track – usually the icmp_host IP is used. If traffic to the conntrack_hosts IP is detected then multiwan does not send a ping health check to the icmp_host otherwise a ping is sent as normal to the icmp_host. By default the conntrack_hosts is checked if the health interval is greater than 5 minutes. This time threshold currently cannot be manipulated. Conntrack is generally used to limit the traffic sent on a GSM network <table border="1"> <tr> <td>Default</td><td>Conntrack checks for traffic from icmp_host IP when health_interval is greater than 5 minutes</td></tr> <tr> <td>Disable</td><td>Conntrack disabled</td></tr> <tr> <td>Custom</td><td>Specifies an IP other than the icmp_host for conntrack to track</td></tr> </table>	Default	Conntrack checks for traffic from icmp_host IP when health_interval is greater than 5 minutes	Disable	Conntrack disabled	Custom	Specifies an IP other than the icmp_host for conntrack to track		
Default	Conntrack checks for traffic from icmp_host IP when health_interval is greater than 5 minutes								
Disable	Conntrack disabled								
Custom	Specifies an IP other than the icmp_host for conntrack to track								
Web: Health Monitor ICMP Timeout UCI: multiwan.[..x..].timeout Opt: timeout	Sets ping timeout in seconds. Choose the time in seconds that the health monitor ICMP will timeout at. <table border="1"> <tr> <td>3</td><td>Wait 3 seconds for ping reply</td></tr> <tr> <td>Range</td><td></td></tr> </table>	3	Wait 3 seconds for ping reply	Range					
3	Wait 3 seconds for ping reply								
Range									
Web: Health Monitor ICMP Interval UCI: multiwan.wan.icmp_interval Opt: icmp_interval	Defines the interval between multiple pings sent at each health check <table border="1"> <tr> <td>1</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	1		Range					
1									
Range									
Web: Health Monitor ICMP Count UCI: multiwan.wan.icmp_count Opt: icmp_count	Defines the number of pings to send at each health check. <table border="1"> <tr> <td>1</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	1		Range					
1									
Range									
Web: Attempts Before WAN Failover UCI: multiwan.[..x..].health_fail_retries Opt: health_fail_retries	Sets the amount of health monitor retries before interface is considered a failure. <table border="1"> <tr> <td>3</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	3		Range					
3									
Range									
Web: Attempts Before WAN Recovery UCI: multiwan.[..x..].health_recovery_retries Opt: health_recovery_retries	Sets the number of health monitor checks before the interface is considered healthy. Only relevant if pre-empt mode is enabled. <table border="1"> <tr> <td>5</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	5		Range					
5									
Range									
Web: Priority UCI: multiwan.[..x..].priority Opt: priority	Specifies the priority of the interface. The higher the value, the higher the priority. This multiwan 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 border="1"> <tr> <td>0</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	0		Range					
0									
Range									
Web: Exclusive Group UCI: multiwan.[..x..].exclusive_group Opt: exclusive_group	Defines the group to which the interface belongs, only one interface can be active. <table border="1"> <tr> <td>0</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	0		Range					
0									
Range									

Web: Manage Interface State (Up/Down) UCI: multiwan.[...x...].manage_state Opt: manage_state	Defines whether multi wan will start and stop the interface. Select Enabled. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.												
0	Disabled.																
1	Enabled.																
Web: Minimum ifup Interval UCI: multiwan.[...x...].ifup_retry_sec Opt: ifup_retry_sec	Specifies the interval in seconds before retrying the primary interface when pre-empt mode is enabled.																
Web: Interface Start Timeout UCI: multiwan.[...x...].ifup_timeout Opt: ifup_timeout	Specifies the time in seconds for interface to start up. If it is not up after this period, it will be considered a fail. Choose timer greater than 120 seconds. <table border="1"> <tr><td>40</td><td>40 seconds</td></tr> <tr><td>Range</td><td></td></tr> </table>	40	40 seconds	Range													
40	40 seconds																
Range																	
Web: Signal Threshold (dBm) UCI: multiwan.[...x...].signal_threshold Opt: signal_threshold	Specifies the minimum signal strength in dBm before considering if the interface fails signal health check. Uses the value stored for sig_dbm in mobile diagnostics. <table border="1"> <tr><td>-115</td><td>Disabled</td></tr> <tr><td>Range</td><td>-46 to -115 dBm</td></tr> </table>	-115	Disabled	Range	-46 to -115 dBm												
-115	Disabled																
Range	-46 to -115 dBm																
Web: RSCP Threshold (dBm) UCI: multiwan.[...x...].rscp_threshold Opt: rscp_threshold	Specifies the minimum RSCP signal strength in dBm before considering if the interface fails signal health check. Uses the value stored for rscp_dbm in mobile diagnostics. <table border="1"> <tr><td>-115</td><td>Disabled</td></tr> <tr><td>Range</td><td>-46 to -115 dBm</td></tr> </table>	-115	Disabled	Range	-46 to -115 dBm												
-115	Disabled																
Range	-46 to -115 dBm																
Web: ECIO Threshold (dB) UCI: multiwan.[...x...].ecio_threshold Opt: ecio_threshold	Specifies the minimum ECIO signal strength in dB before considering if the interface fails signal health check. Uses the value stored for ecio_db in mobile diagnostics. <table border="1"> <tr><td>-115</td><td>Disabled</td></tr> <tr><td>Range</td><td>-46 to -115 dB</td></tr> </table>	-115	Disabled	Range	-46 to -115 dB												
-115	Disabled																
Range	-46 to -115 dB																
Web: Signal Test UCI: multiwan.[...x...].signal_test Opt: signal_test	Defines script to test various signal characteristics in multiwan signal test. For example: <pre>option signal_test '(tech == 0) then (sig_dbm > -70) else (rscp_dbm > -105 and ecio_db > -15)'</pre> <p>This states that when technology is GSM a health fail is determined when signal strength is less than -70dBm. When technology is not GSM a health fail occurs when either rscp_dbm falls below -105dBm or ecio_db falls below -15dB.</p> <p>Tech values are:</p> <table border="1"> <tr><td>0</td><td>GSM</td></tr> <tr><td>1</td><td>GSM Compact</td></tr> <tr><td>2</td><td>UTRAN</td></tr> <tr><td>3</td><td>GSM w/EGPRS</td></tr> <tr><td>4</td><td>UTRAN w/HSPDA</td></tr> <tr><td>5</td><td>UTRAN w/HSUPA</td></tr> <tr><td>6</td><td>UTRAN w/HSUPA and HSDPA</td></tr> <tr><td>7</td><td>E-UTRAN</td></tr> </table>	0	GSM	1	GSM Compact	2	UTRAN	3	GSM w/EGPRS	4	UTRAN w/HSPDA	5	UTRAN w/HSUPA	6	UTRAN w/HSUPA and HSDPA	7	E-UTRAN
0	GSM																
1	GSM Compact																
2	UTRAN																
3	GSM w/EGPRS																
4	UTRAN w/HSPDA																
5	UTRAN w/HSUPA																
6	UTRAN w/HSUPA and HSDPA																
7	E-UTRAN																

Table 65: Information table for Multi-WAN page

Click **Save**.

23.2.2 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 106: 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.

23.2.2.1 Basic settings

Web Field/UCI/Package Option	Description				
Web: SMS Enable UCI: mobile.main.sms Opt: sms	Enables SMS. <table border="1" style="margin-left: 20px;"> <tr> <td>no</td><td>Disabled.</td></tr> <tr> <td>yes</td><td>Enabled.</td></tr> </table>	no	Disabled.	yes	Enabled.
no	Disabled.				
yes	Enabled.				
Web: Collect ICCIDs UCI: mobile.main.init_get_iccid Opt: init_get_iccid	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 <table border="1" style="margin-left: 20px;"> <tr> <td>no</td><td>Disabled.</td></tr> <tr> <td>yes</td><td>Enabled.</td></tr> </table>	no	Disabled.	yes	Enabled.
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 66: Information table for mobile manager basic settings

23.2.2.2 Caller settings

Web Field/UCI/Package Option	Description				
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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

Table 67: Information table for caller settings

23.2.3 Roaming interface template

The screenshot shows the 'Roaming Interface Template' configuration page. At the top, there are navigation links: Status, System, Services, Network, and Logout. A blue bar indicates 'UNSAVED CHANGES: 12'. The main title is 'Roaming Interface Template' with a subtitle 'Common config values for interfaces created by Automatic Operator Selection'. On the right, there is a 'Delete' button.

Interface Signal Sort: A checked checkbox with a tooltip: 'Sort interfaces by signal strength so those having better signal strength at the startup would be tried first'.

Roaming SIM: A dropdown menu set to '1' with a tooltip: 'In which slot roaming sim-card is inserted'.

Firewall Zone: A radio button group with three options: 'lan:' (selected), 'wan:' (disabled), and 'unspecified -or- create:'. A tooltip for 'wan:' says: 'Append all the generated interfaces to this zone'.

Service Type: A dropdown menu set to 'UMTS/GPRS'.

APN: A text input field containing 'vpn.amylan.co.uk'.

PIN: An empty text input field.

PAP/CHAP username: A text input field containing 'campen1'.

PAP/CHAP password: A masked text input field showing '*****' with a green eye icon.

Health Monitor Interval: A dropdown menu set to 'Disable'.

Health Monitor ICMP Host(s): A dropdown menu set to 'Disable'.

Health Monitor ICMP Timeout: A dropdown menu set to '1 sec.'.

Attempts Before WAN Failover: A dropdown menu set to '3'.

Attempts Before WAN Recovery: A dropdown menu set to '5'.

Priority: A dropdown menu set to '5' with a tooltip: 'Higher value is higher priority'.

Minimum ifup Interval: A dropdown menu set to '120 sec.' with a tooltip: 'Minimum interval between two successive interface start attempts'.

Interface Start Timeout: A dropdown menu set to '180' with a tooltip: 'Time for interface to startup'.

Signal Threshold (dBm): A dropdown menu set to '-105' with a tooltip: 'Below is a failure'.

At the bottom left is an 'Add' button. At the bottom right are 'Save & Apply', 'Save', and 'Reset' buttons.

Figure 107: 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	<p>Sorts interfaces by signal strength priority so those that have a better signal strength will be tried first.</p> <table border="1" data-bbox="692 294 1399 361"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
0	Disabled.										
1	Enabled.										
Web: Roaming SIM UCI: mobile.main.roaming_sim Opt: roaming_sim	<p>Sets in which slot to insert roaming SIM card.</p> <table border="1" data-bbox="692 417 1399 485"> <tr> <td>1</td><td>SIM slot 1.</td></tr> <tr> <td>2</td><td>SIM slot 2.</td></tr> </table>	1	SIM slot 1.	2	SIM slot 2.						
1	SIM slot 1.										
2	SIM slot 2.										
Web: Firewall Zone UCI: mobile.@roaming_template[0].firewall_zone Opt: firewall_zone	<p>Adds all generated interfaces to this zone. Select existing zone or click unspecified or create to create new zone.</p>										
Web: Service Type UCI: mobile.@roaming_template[0].service Opt: service	<p>Specifies the service type that will be used to connect to the network.</p> <table border="1" data-bbox="692 709 1399 1012"> <tr> <td>Auto</td><td>GSM module will automatically detect the best available technology code.</td></tr> <tr> <td>lte_only</td><td>Allows GSM module to only connect to LTE network.</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>	Auto	GSM module will automatically detect the best available technology code.	lte_only	Allows GSM module to only connect to LTE network.	umts_only	Allows GSM module to only connect to 3G network.	gprs_only	Allows GSM module to only connect to GPRS network.	cdma	Allows GSM module to only connect to cdma network.
Auto	GSM module will automatically detect the best available technology code.										
lte_only	Allows GSM module to only connect to LTE network.										
umts_only	Allows GSM module to only connect to 3G network.										
gprs_only	Allows GSM module to only connect to GPRS network.										
cdma	Allows GSM module to only connect to cdma network.										
Web: APN UCI: mobile.@roaming_template[0].apn Opt: apn	APN name of Mobile Network Operator.										
Web: PIN UCI: mobile.@roaming_template[0].pincode Opt: pincode	SIM card's PIN number.										
Web: PAP/CHAP username UCI: mobile.@roaming_template[0].username Opt: username	Username used to connect to APN.										
Web: PAP/CHAP password UCI: mobile.@roaming_template[0].password Opt: password	Password used to connect to APN.										
Web: Health Monitor Interval UCI: mobile.@roaming_template[0].health_interval Opt: health_interval	<p>Sets the period to check the health status of the interface. The Health Monitor interval will be used for:</p> <ul style="list-style-type: none"> interface state checks ping interval signal strength checks 										
Web: Health Monitor ICMP Host(s) UCI: mobile.@roaming_template[0].icmp_hosts Opt: icmp_hosts	<p>Specifies target IP address for ICMP packets.</p> <table border="1" data-bbox="692 1697 1399 1843"> <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>	Disable	Disables the option.	DNS servers	DNS IP addresses will be used.	WAN gateway	Gateway IP address will be used.	custom	Ability to provide IP address.		
Disable	Disables the option.										
DNS servers	DNS IP addresses will be used.										
WAN gateway	Gateway IP address will be used.										
custom	Ability to provide IP address.										
Web: Health Monitor ICMP Timeout UCI: mobile.@roaming_template[0].timeout Opt: timeout	<p>Specifies the time in seconds that Health Monitor ICMP will timeout at.</p> <p>Sets ping timeout in seconds. Choose the time in seconds that the health monitor ICMP will timeout at.</p> <table border="1" data-bbox="692 1967 1399 2034"> <tr> <td>3</td><td>Wait 3 seconds for ping reply</td></tr> <tr> <td>Range</td><td></td></tr> </table>	3	Wait 3 seconds for ping reply	Range							
3	Wait 3 seconds for ping reply										
Range											

Web: Attempts Before WAN Failover UCI: mobile.@roaming_template[1].health_fail_retries Opt: health_fail_retries	Defines the number of health check failures before interface is disconnected. <table border="1"><tr><td>3</td><td></td></tr><tr><td colspan="2">Range</td></tr></table>	3		Range	
3					
Range					
Web: Attempts Before WAN Recovery UCI: mobile.@roaming_template[0].health_recovery_retries Opt: health_recovery_retries	Sets the number of health check passes before the interface is considered healthy. This field is not used for a roaming template. <table border="1"><tr><td>5</td><td></td></tr><tr><td colspan="2">Range</td></tr></table>	5		Range	
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 lower than the one specified in the priority field for the PMP interface. <table border="1"><tr><td>0</td><td></td></tr><tr><td colspan="2">Range</td></tr></table>	0		Range	
0					
Range					
Web: Minimum ifup interval UCI: multiwan.wan.ifup_retry_sec Opt: ifup_retry_sec	Not used for a roaming interface. <table border="1"><tr><td>300</td><td>Retry primary interface every 300 seconds</td></tr><tr><td colspan="2">Range</td></tr></table>	300	Retry primary interface every 300 seconds	Range	
300	Retry primary interface every 300 seconds				
Range					
Web: Interface Start Timeout UCI: mobile.@roaming_template[0].ifup_timeout_sec Opt: ifup_timeout	Specifies the time in seconds for interface to start up. If it is not up after this period, it will be considered a fail. <table border="1"><tr><td>40</td><td>40 seconds</td></tr><tr><td colspan="2">Range</td></tr></table>	40	40 seconds	Range	
40	40 seconds				
Range					
Web: Signal Threshold (dBm) UCI: mobile.@roaming_template[0].signal_threshold Opt: signal_threshold	Specifies the minimum RSCP signal strength in dBm before considering if the interface fails signal health check. Uses the value stored for rscp_dbm in mobile diagnostics. <table border="1"><tr><td>Range</td><td>-46 to -115 dBm</td></tr><tr><td colspan="2">-115dBm</td></tr></table>	Range	-46 to -115 dBm	-115dBm	
Range	-46 to -115 dBm				
-115dBm					

Table 68: 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 108: The reboot page

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

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

As in the previous section, multi-WAN connects the PMP interface and uses auto created interfaces for failover.

However, in this scenario, the auto-created interface will not be disconnected as soon as the `ifup_retry_sec` expires for the PMP interface. The primary interface will be reconnected when the current auto-created interface fails multiwan health checks after expiration of the `ifup_retry_sec` timer.

Follow the instructions in the section above for creation of the PMP interface, multi-WAN and Mobile Manager roaming interfaces. The only change in configuration compared to the PMP + roaming: pre-empt enabled scenario is that you must disable the pre-empt option in the multi-WAN package.

23.2.4.1 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.

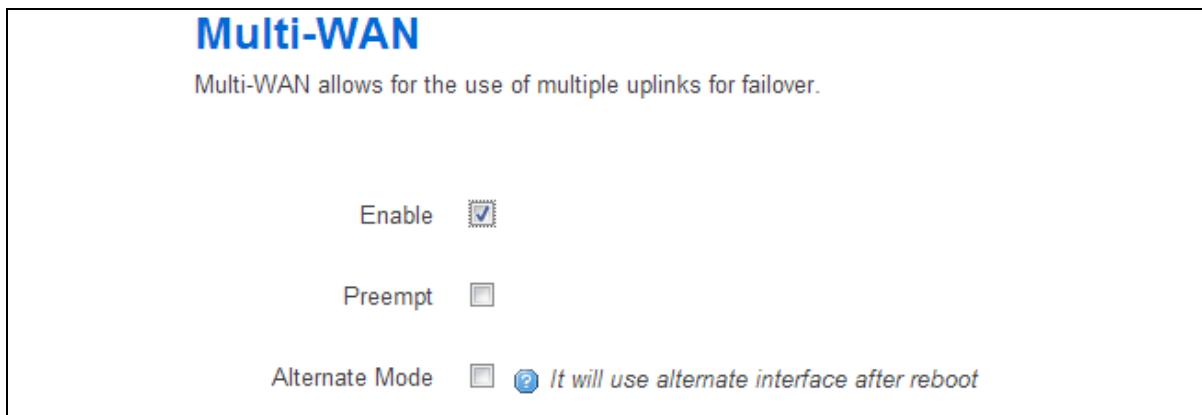


Figure 109: The multi-wan page, pre-empt not selected

Click **Save & Apply**.

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



Figure 110: The system reboot page

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

23.2.5 Scenario 3: No PMP + roaming

In this scenario there is no PMP interface that can be used for a connection. The router scans the available mobile networks at boot and sorts the networks according to signal strength.

The network that offers the best signal strength will be the first to connect. Multi-WAN then controls the failover between the available networks.

Multiwan periodically does a health check on the interface. A health check comprises of a configurable combination of the following:

- interface state
- pings to an ICMP target
- signal level checks using signal threshold, RSCP threshold and ECIO threshold option values

A fail for any of the above health checks results in a fail. After a configurable number of health check failures, Multi-WAN will disconnect the failed interface and attempt to connect to the next best roaming interface.

23.2.6 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 ICCIDs
Callers	Configure callers that can use SMS.
Roaming Interface Template	Configure common values for interface created by Automatic Operator Selection.

23.2.6.1 Basic settings

Web Field/UCI/Package Option	Description				
Web: SMS Enable UCI: mobile.main.sms Opt: sms	Enables SMS. <table border="1" style="margin-left: 20px;"> <tr> <td>no</td> <td>Disabled.</td> </tr> <tr> <td>yes</td> <td>Enabled.</td> </tr> </table>	no	Disabled.	yes	Enabled.
no	Disabled.				
yes	Enabled.				
Web: Collect ICCIDs UCI: mobile.main.init_get_iccid Opt: init_get_iccid	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 displayed under mobile stats. <table border="1" style="margin-left: 20px;"> <tr> <td>no</td> <td>Disabled.</td> </tr> <tr> <td>yes</td> <td>Enabled.</td> </tr> </table>	no	Disabled.	yes	Enabled.
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. <table border="1" style="margin-left: 20px;"> <tr> <td>blank</td> <td></td> </tr> <tr> <td>range</td> <td></td> </tr> </table>	blank		range	
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. <table border="1" style="margin-left: 20px;"> <tr> <td>blank</td> <td></td> </tr> <tr> <td>range</td> <td></td> </tr> </table>	blank		range	
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. <table border="1" style="margin-left: 20px;"> <tr> <td>blank</td> <td></td> </tr> <tr> <td>range</td> <td></td> </tr> </table>	blank		range	
blank					
range					

Table 69: Information table for mobile manager basic settings

23.2.6.2 Caller settings

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

Table 70: Information table for mobile manager caller settings

23.2.7 Roaming interface template

The screenshot shows the 'Roaming Interface Template' configuration page. At the top, there are navigation links: Status, System, Services, Network, and Logout. A blue bar indicates 'UNSAVED CHANGES 12'. Below the header, the title 'Roaming Interface Template' is displayed, followed by a note: 'Common config values for interfaces created by Automatic Operator Selection'. On the right side, there is a 'Delete' button. The main configuration area includes:

- Interface Signal Sort:** A checkbox with a tooltip: 'Sort interfaces by signal strength so those having better signal strength at the startup would be tried first'.
- Roaming SIM:** A dropdown menu set to '1' with a tooltip: 'In which slot roaming sim-card is inserted'.
- Firewall Zone:** A dropdown menu with three options: 'lan: lan:' (selected), 'wan: 3g_s1_voda:' (disabled), and 'unspecified -or- create:'. There is also a checkbox: 'Append all the generated interfaces to this zone'.
- Service Type:** A dropdown menu set to 'UMTS/GPRS'.
- APN:** A text input field containing 'vpn.amyilan.co.uk'.
- PIN:** A text input field.
- PAP/CHAP username:** A text input field containing 'campen1'.
- PAP/CHAP password:** A text input field containing '*****' with a visibility icon.
- Health Monitor Interval:** A dropdown menu set to 'Disable'.
- Health Monitor ICMP Host(s):** A dropdown menu set to 'Disable'.
- Health Monitor ICMP Timeout:** A dropdown menu set to '1 sec.'
- Attempts Before WAN Failover:** A dropdown menu set to '3'.
- Attempts Before WAN Recovery:** A dropdown menu set to '5'.
- Priority:** A text input field containing '5' with a tooltip: 'Higher value is higher priority'.
- Minimum ifup Interval:** A dropdown menu set to '120 sec.' with a tooltip: 'Minimum interval between two successive interface start attempts'.
- Interface Start Timeout:** A dropdown menu set to '180' with a tooltip: 'Time for interface to startup'.
- Signal Threshold (dBm):** A dropdown menu set to '-105' with a tooltip: 'Below is a failure'.

At the bottom, there are buttons for 'Add', 'Save & Apply' (highlighted in blue), 'Save', and 'Reset'.

Figure 111: 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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>1</td> <td>SIM slot 1.</td> </tr> <tr> <td>2</td> <td>SIM slot 2.</td> </tr> </table>	1	SIM slot 1.	2	SIM slot 2.				
1	SIM slot 1.								
2	SIM slot 2.								
Web: Firewall Zone UCI: mobile.@roaming_template[0].firewall_zone Opt: firewall_zone	Adds all generated interfaces to this zone. Select existing zone or click unspecified or create to create a new zone.								
Web: Service Type UCI: mobile.@roaming_template[0].service Opt: service	Specifies the service type that will be used to connect to the network. <table border="1" style="margin-left: auto; margin-right: auto;"> <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>	UMTS/GPRS	GSM module will automatically detect the best available technology code.	Umts_only	Allows GSM module to only connect to 3G network.	GPRS_only	Allows GSM module to only connect to GPRS network.	cdma	Allows GSM module to only connect to cdma network.
UMTS/GPRS	GSM module will automatically detect the best available technology code.								
Umts_only	Allows GSM module to only connect to 3G network.								
GPRS_only	Allows GSM module to only connect to GPRS network.								
cdma	Allows GSM module to only connect to cdma network.								
Web: APN UCI: mobile.@roaming_template[0].apn Opt: apn	APN name of Mobile Network Operator.								
Web: PIN UCI: mobile.@roaming_template[0].pincode Opt: pincode	SIM card's PIN number.								
Web: PAP/CHAP username UCI: mobile.@roaming_template[0].username Opt: username	Username used to connect to APN.								
Web: PAP/CHAP password UCI: mobile.@roaming_template[0].password Opt: password	Password used to connect to APN.								
Web: Health Monitor Interval UCI: mobile.@roaming_template[0].health_interval Opt: health_interval	Sets the period to check the health status of the interface. The Health Monitor interval will be used for: interface state checks ping interval signal strength checks								
Web: Health Monitor ICMP Host(s) UCI: mobile.@roaming_template[0].icmp_hosts Opt: icmp_hosts	Specifies target IP address for ICMP packets. <table border="1" style="margin-left: auto; margin-right: auto;"> <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>	Disable	Disables the option	DNS servers	DNS IP addresses will be used.	WAN gateway	Gateway IP address will be used.	custom	Ability to provide IP address.
Disable	Disables the option								
DNS servers	DNS IP addresses will be used.								
WAN gateway	Gateway IP address will be used.								
custom	Ability to provide IP address.								
Web: Health Monitor ICMP Timeout UCI: mobile.@roaming_template[0].timeout Opt: timeout	Sets ping timeout in seconds. Choose the time in seconds that the health monitor ICMP will timeout at. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>3</td> <td>Wait 3 seconds for ping reply</td> </tr> <tr> <td>Range</td> <td></td> </tr> </table>	3	Wait 3 seconds for ping reply	Range					
3	Wait 3 seconds for ping reply								
Range									

Web: Attempts Before WAN Failover UCI: mobile.@roaming_template[1].health_fail_retries Opt: health_fail_retries	Defines the number of health check failures before interface is disconnected. <table border="1"><tr><td>3</td><td></td></tr><tr><td>range</td><td></td></tr></table>	3		range	
3					
range					
Web: Attempts Before WAN Recovery UCI: mobile.@roaming_template[0].health_recovery_retries Opt: health_recovery_retries	Sets the number of health check passes before the interface is considered healthy. This field is not used for a roaming template.				
Web: Priority UCI: mobile.@roaming_template[0].priority Opt: priority	Type the priority number. The higher the value, the higher the priority. <table border="1"><tr><td>0</td><td></td></tr><tr><td>range</td><td></td></tr></table>	0		range	
0					
range					
Web: Minimum ifup interval UCI: mobile.@roaming_template[0].ifup_retry_sec Opt: ifup_retry_sec	Specifies the interval in seconds before retrying the primary interface when pre-empt mode is enabled. <table border="1"><tr><td>300</td><td>Retry primary interface every 300 seconds</td></tr><tr><td>Range</td><td></td></tr></table>	300	Retry primary interface every 300 seconds	Range	
300	Retry primary interface every 300 seconds				
Range					
Web: Interface Start Timeout UCI: mobile.@roaming_template[0].ifup_timeout_sec Opt: ifup_timeout	Specifies the time in seconds for interface to start up. If it is not up after this period, it will be considered a fail. It is recommended to configure a value greater than 120 seconds. <table border="1"><tr><td>40</td><td></td></tr><tr><td>range</td><td></td></tr></table>	40		range	
40					
range					
Web: Signal Threshold (dBm) UCI: mobile.@roaming_template[0].signal_threshold Opt: signal_threshold	Specifies the minimum signal strength in dBm before considering if the interface fails signal health check. Uses the value stored for sig_dbm in mobile diagnostics. -115 dBm <table border="1"><tr><td></td><td>Disabled</td></tr><tr><td>range</td><td>-46 to -115 dBm</td></tr></table>		Disabled	range	-46 to -115 dBm
	Disabled				
range	-46 to -115 dBm				

Table 71: Information table for roaming interface template

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

23.2.7.1 Set multi-WAN operation

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

Multi-WAN

Multi-WAN allows for the use of multiple uplinks for failover.

Add

WAN Interfaces

Health Monitor detects and corrects network changes and failed connections.

This section contains no values yet

Add

Save & Apply Save Reset

Figure 112: The multi-WAN page

In the Multi-WAN section click **Add**.

Web Field/UCI/Package Option	Description				
Web: Enable UCI: multiwan.config.enabled Opt: enabled	Enables multiwan. Select this option. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">0</td> <td>Disabled.</td> </tr> <tr> <td style="text-align: center;">1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Preempt UCI: multiwan.config.preempt Opt: pre-empt	Enables or disables pre-emption for multiwan. If enabled the router will keep trying to connect to a higher priority interface depending on timer set by ifup_retry_sec. Leave this option unselected. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">0</td> <td>Disabled.</td> </tr> <tr> <td style="text-align: center;">1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Alternate Mode UCI: multiwan.config.alt Opt: alt	Enables or disables alternate mode for multiwan. If enabled the router will use an alternate interface after reboot. Leave this option unselected. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">0</td> <td>Disabled.</td> </tr> <tr> <td style="text-align: center;">1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

Table 72: Information table for multi-WAN operation

23.3 Configuring via UCI

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

23.3.1.1 PMP interface configuration

The PMP interface is configured in the network package /etc/config/network. To view the network configuration file, enter:

```
root@GW_router:~# 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 '3g_s1_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:

```

root@GW_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=static
network.lan.ipaddr=192.168.100.1
network.lan.netmask=255.255.255.0
network.3g_s1_voda=interface
network. 3g_s1_voda.auto=0
network. 3g_s1_voda.proto=3g
network. 3g_s1_voda.service=umts
network. 3g_s1_voda.apn=test IE
network. 3g_s1_voda.username=test
network. 3g_s1_voda.password=test
network. 3g_s1_voda.sim=1
network. 3g_s1_voda.operator=vodafone IE

```

23.3.1.2 Roaming interface configuration

The roaming interface configurations are stored in the mobile package /etc/config/mobile. To view the mobile configuration file, enter: root@GW_router:~# uci export 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:

```

root@GW_router:~#uci show mobile
mobile.main=mobile
mobile.main.sms=yes
mobile.main.roaming_sim=1
mobile.main.init_get_iccid=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

```

23.3.1.3 Multi-WAN configuration using UCI

The configuration file for package multiwan is stored on **/etc/config/multiwan**

To see configuration file of mobile package, enter:

```

root@GW_router:~# cat /etc/config/multiwan
config multiwan 'config'
    option enabled '1'
    option preempt '1'

config interface '3g_s1_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'

```

To view the uci command of package multiwan, enter:

```

root@GW_router:~# uci show 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.3g_s1_voda.timeout=1
multiwan.3g_s1_voda.icmp_hosts=disable
multiwan.3g_s1_main_voda.priority=10
multiwan.3g_s1_voda.exclusive_group=3g
multiwan.3g_s1_voda.signal_threshold=-95
multiwan.3g_s1_voda.ifup_retry_sec=350
multiwan.3g_s1_voda.ifup_timeout_sec=180
multiwan.3g_s1_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

23.4 Configuring no PMP + roaming using UCI

The roaming interface configuration file is stored in the mobile package **/etc/config/mobile**. To view the mobile package, enter:

```

root@GW_router:~# 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 the mobile package via uci commands, enter:

```

root@GW_router:~# uci show 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 multiwan package is stored on **/etc/config/multiwan**. To view the multiwan package, enter:

```

root@GW_router:~# uci export multiwan
package multiwan

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

To see multiwan package via uci, enter:
root@GW_router:~# uci show multiwan
multiwan.config=multiwan
multiwan.config.enabled=yes
multiwan.config.preempt=no
multiwan.config.alt_mode=no

```

23.5 Automatic operator selection diagnostics via the web interface

23.5.1 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**.

Interface Overview		
Network	Status	Actions
3G_S1_O2IR 3g-3g_s1_o2IR	RX: 0.00 B (0 Pkts.) TX: 0.00 B (0 Pkts.)	Connect Stop Edit Delete
3G_S1_VODA 3g-3g_s1_voda	Uptime: 7h 31m 26s RX: 62.00 B (8 Pkts.) TX: 23.44 KB (329 Pkts.) IPv4: 10.140.1.23/32	Connect Stop Edit Delete
WCLIENT Client "0"	MAC Address: 00:00:00:00:00:00 RX: 0.00 B (0 Pkts.) TX: 0.00 B (0 Pkts.)	Connect Stop Edit Delete
LAN eth0	Uptime: 7h 35m 24s MAC Address: 00:E0:C8:10:1A:82 RX: 67.25 KB (502 Pkts.) TX: 132.29 KB (157 Pkts.) IPv4: 10.1.1.9/29	Connect Stop Edit Delete
LOOPBACK lo	Uptime: 7h 35m 30s MAC Address: 00:00:00:00:00:00 RX: 41.72 KB (516 Pkts.) TX: 41.72 KB (516 Pkts.) IPv4: 127.0.0.1/8 IPv6: 0:0:0:0:0:0:1/128	Connect Stop Edit Delete

Figure 113: 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.

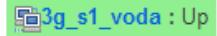
<i>There are no active leases.</i>
<p>Multi-WAN Status</p> <div style="display: flex; justify-content: space-around;">  3g_s1_voda : Up  3g_s1_O2IR : Down(standby backup) </div>

Figure 114: The status page: multi-WAN status section page

23.6 Automatic operator selection diagnostics via UCI

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

```
root@GW_router:~# 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 115: Example of output from the command: cat /var/const_stat/multiwan

To check interfaces created in the network package, enter:

```
root@GW_router:~# cat /var/const_state/network
```

```
root@GW_GW0000 :~# 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 116: Example of output from the command cat /var/const_state/network

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

```
root@GW_router:~# cat /var/const_state/mobile
```

```
root@GW_GW0000 :~# cat /var/const_state/network
mobile.3g_0=status
mobile.3g_0.sim1_iccid=89314404000039480265
root@GW_GW0000 :~#
root@GW_GW0000 :~#
root@GW_GW0000 :~# 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@GW_GW0000 :~#
```

Figure 117: Example of output from the command cat /vat/const_state/mobile

24 Configuring Connection Watch (cwatch)

Connection Watch is a recovery feature to enable dynamic recovery of an interface. You can configure multiple instances of Connection Watch.

Connection Watch consists of the following configurable instances:

- Interface(s) to be monitored
- Failure periods
- Recovery actions

If no data is received over the monitored interface during the configured duration, then the recovery action is performed. If more than one interface is specified under a single Connection Watch, the recovery action will be performed only if no data is received on **both** of the interfaces for the defined period.

Currently three configurable periods and associated recovery actions can be defined.

24.1 Configuration package used

Package	Sections
cwatch	watch

24.2 Configuring Connection Watch using the web interface

To configure Connection Watch using the web interface, select **Services->Connection Watch**. The Connection Watch page appears.

If no Connection Watch configuration exists in the configuration file, first enter a name for the Connection Watch instance and select **Add**.

Figure 118: The add connection watch configuration page

Connection Watch
Configuration of Connection Watch.

Watch

WATCH_MOBILE

Enabled	<input type="checkbox"/>
Status	unknown
Interfaces	<input type="checkbox"/> LAN1: (no interfaces attached) <input type="checkbox"/> LAN2: <input type="checkbox"/> LAN3: <input checked="" type="checkbox"/> MOBILE1: <input type="checkbox"/> PoADSL: <input type="checkbox"/> loopback:
Failure Time for Action 1	1h
Failure Action 1	ifup MOBILE1
Failure Time for Action 2	10h
Failure Action 2	/etc/init.d/usb_startup restart
Failure Time for Action 3	24h
Failure Action 3	reboot

Figure 119: The connection watch configuration page

Web Field/UCI/Package Option	Description				
Web: Enabled UCI: cwatch.@watch[0].enabled Opt: enabled	Enables a cwatch instance. <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Interfaces UCI: cwatch.@watch[0].test_ifaces Opt: test_ifaces	Defines the interface name(s) to monitor. Multiple interfaces are delimited by space separator. Example: <pre>option test_ifaces 'WANADSL WANMOBILE'</pre> If multiple interfaces are defined the failure action will only be triggered if no traffic is received on all interfaces for the defined period.				
Web: Failure Time for Action 1 UCI: cwatch.@watch[0].failure_time_1 Opt: failure_time_1	Defines a duration to monitor an interface for receive traffic. Duration can be specified in seconds, minutes, hours, days. <table border="1"> <tr> <td>1h</td> <td></td> </tr> <tr> <td>Range</td> <td>s; m; h; d;</td> </tr> </table>	1h		Range	s; m; h; d;
1h					
Range	s; m; h; d;				
Web: Failure Action 1 UCI: cwatch.@watch[0].failure_action_1 Opt: failure_action_1	Defines the failure action associated with failure_time_1. Example to force up interface: <pre>option failure action 1 'ifup wan'</pre> <table border="1"> <tr> <td>blank</td> <td></td> </tr> <tr> <td>Range</td> <td></td> </tr> </table>	blank		Range	
blank					
Range					

Web: Failure Time for Action 2 UCI: cwatch.@watch[0].failure_time_2 Opt: failure_time_2	Defines a second duration to monitor an interface for receive traffic. Duration can be specified in seconds, minutes, hours, days. <table border="1"> <tr><td>10h</td><td></td></tr> <tr><td>Range</td><td>s; m; h; d;</td></tr> </table>	10h		Range	s; m; h; d;
10h					
Range	s; m; h; d;				
Web: Failure Action 2 UCI: cwatch.@watch[0].failure_action_2 Opt: failure_action_2	Defines the failure action associated with failure_time_2. Example to reset usb: option failure_action_1 '/etc/init.d/usb_startup restart' <table border="1"> <tr><td>blank</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	blank		Range	
blank					
Range					
Web: Failure Time for Action 3 UCI: cwatch.@watch[0].failure_time_3 Opt: failure_time_3	Defines a third duration to monitor an interface for receive traffic. Duration can be specified in seconds, minutes, hours, days. <table border="1"> <tr><td>24h</td><td></td></tr> <tr><td>Range</td><td>s; m; h; d;</td></tr> </table>	24h		Range	s; m; h; d;
24h					
Range	s; m; h; d;				
Web: Failure Action 3 UCI: cwatch.@watch[0].failure_action_3 Opt: failure_action_3	Defines the failure action associated with failure_time_3. Example to reset usb: option failure_action_3 'reboot' <table border="1"> <tr><td>blank</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	blank		Range	
blank					
Range					

Table 73: Information table for cwatch section

24.3 Configuring cwatch using command line

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

```
cwatch.@watch[0]=watch
cwatch.@watch[0].enabled=1
```

Or using package options:

```
config watch
    option enabled '1'
```

However, to better identify it, we recommend giving the cwatch instance a name. For example, a watch named 'WATCH_MOBILE' will be cwatch.WATCH_MOBILE.

To define a named cwatch instance using UCI, enter:

```
cwatch.WATCH_MOBILE=watch
cwatch.WATCH_MOBILE.enabled=1
```

To define a named cwatch instance using package options, enter:

```
config watch 'WATCH_MOBILE'
    option 'enabled' '1'
```

24.3.1 cwatch using UCI

```
root@GW_router:~# uci show cwatch
cwatch.WATCH_MOBILE=watch
cwatch.WATCH_MOBILE.enabled=1
cwatch.WATCH_MOBILE.test_ifaces=wan
cwatch.WATCH_MOBILE.failure_time_1=1h
cwatch.WATCH_MOBILE.failure_action_1=ifup wan
cwatch.WATCH_MOBILE.failure_time_2=10h
cwatch.WATCH_MOBILE.failure_action_2=/etc/init.d/usb_startup restart
cwatch.WATCH_MOBILE.failure_time_3=24h
cwatch.WATCH_MOBILE.failure_action_3=reboot
```

24.3.2 cwatch using package options

```
root@GW_router:~# uci export cwatch
package cwatch

config watch 'WATCH_MOBILE'
    option enabled '1'
    option test_ifaces wan
    option failure_time_1 '1h'
    option failure_action_1 'ifup wan'
    option failure_time_2 '10h'
    option failure_action_2 '/etc/init.d/usb_startup restart'
    option failure_time_3 '24h'
    option failure_action_3 'reboot'
```

24.4 cwatch diagnostics

24.4.1 Syslog

A syslog message will be generated when cwatch starts:

```
cwatch[x]: cwatch configuration OK. Entering main loop...
```

Syslog messages will be generated when the failure action is triggered:

```
cwatch[x]: Watch WATCH_MOBILE executed action 1
cwatch[x]: Watch WATCH_MOBILE executed action 2
cwatch[x]: Watch WATCH_MOBILE executed action 3
```

25 Configuring DHCP server and DNS (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.

25.1 Configuration package used

Package	Sections
dhcp	dnsmasq
	dhcp
	host

25.2 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 120: The DHCP and DNS page

25.2.1 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.					
	<table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>		1	Enabled.	0	Disabled.
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.					
	<table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>		1	Enabled.	0	Disabled.
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.					
	<table border="1"> <tr> <td>Lan</td><td>Serve only on LAN interface</td></tr> <tr> <td>Range</td><td></td></tr> </table>		Lan	Serve only on LAN interface	Range	
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.					
	<table border="1"> <tr> <td>/lan/</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>		/lan/		Range	
/lan/						
Range						
Web: Local Domain UCI: dhcp.@dnsmasq[0].domain Opt: domain	Specifies local domain suffix appended to DHCP names and hosts file entries.					
	<table border="1"> <tr> <td>lan</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>		lan		Range	
lan						
Range						
Web: Log Queries UCI: dhcp.@dnsmasq[0].logqueries Opt: logqueries	Writes received DNS requests to syslog.					
	<table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>		0	Disabled.	1	Enabled.
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.					
	<table border="1"> <tr> <td></td><td>No DNS server configured.</td></tr> <tr> <td>Range</td><td></td></tr> </table>			No DNS server configured.	Range	
	No DNS server configured.					
Range						
Web: Rebind Protection UCI: dhcp.@dnsmasq[0].rebind_protection Opt: rebind_protection	Enables DNS rebinding attack protection by discarding upstream RFC1918 responses.					
	<table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>		0	Disabled.	1	Enabled.
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 rebinding protection is enabled.					
	<table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>		0	Disabled.	1	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 rebinding protection is enabled. When using UCI multiple servers should be entered with a space between them.					
	<table border="1"> <tr> <td></td><td>No list configured.</td></tr> <tr> <td>Range</td><td></td></tr> </table>			No list configured.	Range	
	No list configured.					
Range						

Table 74: Information table for general server settings

25.2.2 Dnsmasq: resolv and host files

The screenshot shows the 'DHCP and DNS' configuration page. At the top, there are tabs for 'General Settings', 'Resolv and Hosts Files' (which is selected), 'TFTP Settings', and 'Advanced Settings'. Below these tabs, there are several configuration options:

- Use /etc/ethers:** A checkbox labeled 'Read /etc/ethers to configure the DHCP-Server'.
- Leasefile:** A dropdown menu set to '/tmp/dhcp.leases'.
- Ignore resolve file:** A checkbox.
- Resolve file:** A dropdown menu set to '/tmp/resolv.conf.auto'.
- Ignore Hosts files:** A checkbox.
- Additional Hosts files:** An input field containing a list of hosts.

Figure 121: 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. <table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
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. <table border="1"> <tr> <td>/tmp/dhcp.leases</td><td>Store DHCP leases in this file.</td></tr> <tr> <td>Range</td><td></td></tr> </table>	/tmp/dhcp.leases	Store DHCP leases in this file.	Range	
/tmp/dhcp.leases	Store DHCP leases in this file.				
Range					
Web: Ignore resolve file UCI: dhcp.@dnsmasq[0].noresolv Opt: noresolv	Defines whether to use the local DNS file for resolving DNS. <table border="1"> <tr> <td>0</td><td>Use local DNS file.</td></tr> <tr> <td>1</td><td>Ignore local DNS file.</td></tr> </table>	0	Use local DNS file.	1	Ignore local DNS file.
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. <table border="1"> <tr> <td>0</td><td>Use local hosts file.</td></tr> <tr> <td>1</td><td>Ignore local hosts file.</td></tr> </table>	0	Use local hosts file.	1	Ignore local hosts file.
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 75: Information table for resolv and host files section

25.2.3 Dnsmasq: TFTP settings

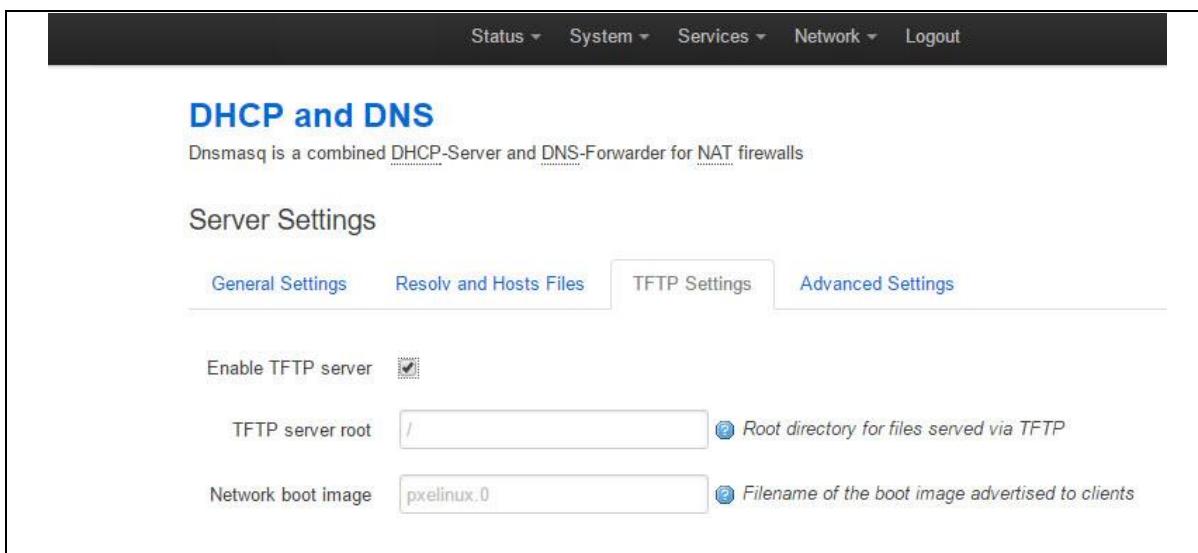


Figure 122: The TFTP settings section

Web Field/UCI/Package Option	Description				
Web: Enable TFTP Server UCI: dhcp.@dnsmasq[0].enable_tftp Opt: enable_tftp	Enables the TFTP server. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
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 76: Information table for TFTP settings

25.2.4 Dnsmasq: advanced settings

DHCP and DNS

Dnsmasq is a combined DHCP-Server and DNS-Forwarder for NAT firewalls

Server Settings

General Settings Resolv and Hosts Files TFTP Settings Advanced Settings

Filter private Do not forward reverse lookups for local networks

Filter useless Do not forward requests that cannot be answered by public name servers

Localise queries Localse hostname depending on the requesting subnet if multiple IPs are available

Expand hosts Add local domain suffix to names served from hosts files

No negative cache Do not cache negative replies, e.g. for not existing domains

Strict order DNS servers will be queried in the order of the resolvfile

Bogus NX Domain
Override List of hosts that supply bogus NX domain results

DNS server port Listening port for inbound DNS queries

DNS query port Fixed source port for outbound DNS queries

Max. DHCP leases Maximum allowed number of active DHCP leases

Max. EDNS0 packet size Maximum allowed size of EDNS.0 UDP packets

Max. concurrent queries Maximum allowed number of concurrent DNS queries

Figure 123: 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. <table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
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. <table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
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. <table border="1"> <tr><td>1</td><td>Enabled.</td></tr> <tr><td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
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. <table border="1"> <tr><td>1</td><td>Enabled.</td></tr> <tr><td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
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. <table border="1"> <tr><td>1</td><td>Enabled.</td></tr> <tr><td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
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. <table border="1"> <tr><td>1</td><td>Enabled.</td></tr> <tr><td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
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. <table border="1"> <tr><td>Empty list</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	Empty list		Range	
Empty list					
Range					
Web: DNS server port UCI: dhcp.@dnsmasq[0].port Opt: port	Listening port for inbound DNS queries. <table border="1"> <tr><td>53</td><td>Set to 0 to disable DNS functionality.</td></tr> <tr><td>Range</td><td>0 - 65535</td></tr> </table>	53	Set to 0 to disable DNS functionality.	Range	0 - 65535
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. <table border="1"> <tr><td>any</td><td></td></tr> <tr><td>Range</td><td>any; 0 - 65535</td></tr> </table>	any		Range	any; 0 - 65535
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. <table border="1"> <tr><td>unlimited</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	unlimited		Range	
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. <table border="1"> <tr><td>1280</td><td>1280 bytes</td></tr> <tr><td>Range</td><td></td></tr> </table>	1280	1280 bytes	Range	
1280	1280 bytes				
Range					
Web: Max concurrent queries UCI: dhcp.@dnsmasq[0].dnsforwardmax Opt: dnsforwardmax	Maximum allowed number of concurrent DNS queries. <table border="1"> <tr><td>150</td><td>1280 bytes</td></tr> <tr><td>Range</td><td></td></tr> </table>	150	1280 bytes	Range	
150	1280 bytes				
Range					

Table 77: Information table for advanced settings

25.2.5 Active leases

This section displays all currently active leases.

Active Leases			
Active Leases			
Hostname	IPv4-Address	MAC-Address	Leasetime remaining
There are no active leases.			

Figure 124: 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 78: Information table for active leases section

25.2.6 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.

The screenshot shows a web-based configuration interface for static leases. At the top, there's a heading 'Static Leases' with a brief description: 'Static leases are used to assign fixed IP addresses and symbolic hostnames to DHCP clients. They are also required for non-dynamic interface configurations where only hosts with a corresponding lease are served.' Below this, there's a table with three columns: 'Hostname', 'MAC-Address', and 'IPv4-Address'. Each column has an input field. To the right of the table are 'Delete', 'Add', and 'Save & Apply' buttons. At the bottom right are 'Save' and 'Reset' buttons.

Figure 125: The static leases section

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

Table 79: Information table for static leases

25.3 Configuring DHCP and DNS using UCI

25.3.1 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@GW_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@GW_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.

25.4 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@GW_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@GW_router:~# uci export dhcp
config 'dhcp' 'lan'
    option 'interface'    'lan'
    option 'start'        '100'
    option 'limit'         '150'
    option 'leasetime'     '12h'
    option ignore          0
```

UCI/Package Option	Description				
Web: n/a UCI: <code>dhcp.<pool_name>.interface</code> Opt: <code>interface</code>	Defines the interface that is served by this DHCP pool. This must be one of the configured interfaces. <table border="1"> <tr> <td>lan</td><td>Enabled.</td></tr> <tr> <td>Range</td><td></td></tr> </table>	lan	Enabled.	Range	
lan	Enabled.				
Range					
Web: n/a UCI: <code>dhcp.<pool_name>.start</code> Opt: <code>start</code>	Defines the offset from the network address for the start of the DHCP pool. It may be greater than 255 to span subnets <table border="1"> <tr> <td>100</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	100		Range	
100					
Range					
Web: n/a UCI: <code>dhcp.<pool_name>.limit</code> Opt: <code>limit</code>	Defines the offset from the network address for the end of the DHCP pool <table border="1"> <tr> <td>150</td><td></td></tr> <tr> <td>Range</td><td>0 - 255</td></tr> </table>	150		Range	0 - 255
150					
Range	0 - 255				

Web: n/a UCI: dhcp.<pool_name>.leasetime Opt: leasetime	Defines the lease time of addresses handed out to clients, for example 12h or 30m. <table border="1"><tr><td>12h</td><td>12 hours</td></tr><tr><td colspan="2">Range</td></tr></table>	12h	12 hours	Range	
12h	12 hours				
Range					
Web: n/a UCI: dhcp.<pool_name>.ignore Opt: ignore	Defines whether this DHCP pool is enabled. <table border="1"><tr><td>0</td><td>DHCP pool enabled.</td></tr><tr><td>1</td><td>DHCP pool disabled.</td></tr></table>	0	DHCP pool enabled.	1	DHCP pool disabled.
0	DHCP pool enabled.				
1	DHCP pool disabled.				
Web: n/a UCI: dhcp.<pool_name>.force Opt: force	Forces DHCP serving on the specified interface even if another DHCP server is detected on the same network segment. <table border="1"><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
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. <table border="1"><tr><td></td><td>No options defined</td></tr><tr><td>Syntax</td><td>Option_number, option_value.</td></tr></table>		No options defined	Syntax	Option_number, option_value.
	No options defined				
Syntax	Option_number, option_value.				
Web: n/a UCI: dhcp.<pool_name>.dynamicdhcp Opt: dynamicdhcp	Defines whether to allocate DHCP leases. <table border="1"><tr><td>1</td><td>Dynamically allocate leases.</td></tr><tr><td>0</td><td>Use /etc/ethers file for serving DHCP leases.</td></tr></table>	1	Dynamically allocate leases.	0	Use /etc/ethers file for serving 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 80: Information table for DHCP pool UCI and package options

25.5 Configuring static leases using UCI

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

```
root@GW_router:~# uci show dhcp.mypc
dhcp.mypc=host
root@GW_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@GW_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.

26 Configuring DHCP client

This section describes how to configure an interface as a DHCP client. This section will only detail the configuration for DHCP client. For information on how to configure other interface options such as firewall zone, mapping of switch ports, etc, refer to standard interface configuration document.

26.1 Configuration packages used

Package	Sections
network	interface

26.2 Configuring DHCP client using the web interface

DHCP client is configured under the interface configuration by setting the interface protocol to DHCP Client. To create and edit interfaces via the web interface, in the top menu, click **Network -> Interfaces**. The Interfaces overview page appears.

Figure 126: The interfaces overview page

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 character 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.

26.2.1 Editing an existing interface for DHCP client

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.

26.2.2 Creating a new interface for DHCP client

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

Figure 127: 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 _																																							
Web: Protocol of the new interface UCI: network.<if name>.proto Opt: proto	Specifies what protocol the interface will operate on. Select DHCP Client . <table border="1" data-bbox="690 415 1397 1066"> <thead> <tr> <th>Option</th><th>Description</th><th>UCI</th></tr> </thead> <tbody> <tr> <td>Static</td><td>Static configuration with fixed address and netmask.</td><td>Static</td></tr> <tr> <td>DHCP Client</td><td>Address and netmask are assigned by DHCP.</td><td>dhcp</td></tr> <tr> <td>Unmanaged</td><td>Unspecified</td><td>Empty</td></tr> <tr> <td>IPv6-in-IPv4 (RFC4213)</td><td>Used with tunnel brokers.</td><td></td></tr> <tr> <td>IPv6-over-IPv4</td><td>Stateless IPv6 over IPv4 transport.</td><td></td></tr> <tr> <td>GRE</td><td>Generic Routing Encapsulation protocol</td><td></td></tr> <tr> <td>IOT</td><td></td><td></td></tr> <tr> <td>L2TP</td><td>Layer 2 Tunnelling Protocol</td><td></td></tr> <tr> <td>PPP</td><td>Point to Point Protocol</td><td></td></tr> <tr> <td>PPPoE</td><td>PPP over Ethernet</td><td></td></tr> <tr> <td>PPPoATM</td><td>PPP over ATM</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><td></td></tr> </tbody> </table>	Option	Description	UCI	Static	Static configuration with fixed address and netmask.	Static	DHCP Client	Address and netmask are assigned by DHCP.	dhcp	Unmanaged	Unspecified	Empty	IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.		IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.		GRE	Generic Routing Encapsulation protocol		IOT			L2TP	Layer 2 Tunnelling Protocol		PPP	Point to Point Protocol		PPPoE	PPP over Ethernet		PPPoATM	PPP over ATM		LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.	
Option	Description	UCI																																						
Static	Static configuration with fixed address and netmask.	Static																																						
DHCP Client	Address and netmask are assigned by DHCP.	dhcp																																						
Unmanaged	Unspecified	Empty																																						
IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.																																							
IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.																																							
GRE	Generic Routing Encapsulation protocol																																							
IOT																																								
L2TP	Layer 2 Tunnelling Protocol																																							
PPP	Point to Point Protocol																																							
PPPoE	PPP over Ethernet																																							
PPPoATM	PPP over ATM																																							
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																																							
Web: Create a bridge over multiple interfaces UCI: network.<if name>.type Opt: type	If you select this option, then the new logical interface created will act as a bridging interface between the chosen existing physical interfaces. <table border="1" data-bbox="690 1145 1341 1246"> <tr> <td>Empty</td><td></td></tr> <tr> <td>Bridge</td><td>Configures a bridge over multiple interfaces.</td></tr> </table>	Empty		Bridge	Configures a bridge over multiple interfaces.																																			
Empty																																								
Bridge	Configures a bridge over multiple interfaces.																																							
Web: Cover the following interface UCI: network.<if name>.ifname Opt: ifname	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'																																							

Table 81: 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	Assign multiple IP addresses to the interface.
DHCP Server	Configure DHCP server settings for this interface.

26.2.3 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

Only **General setup** and **Advanced Settings** have DHCP client option configuration options

26.2.3.1 Common configuration – general setup

Common Configuration	
General Setup Advanced Settings Physical Settings Firewall Settings	
Status	MAC Address: 00:E0:C8:D3:18:20 eth3 RX: 0.00 B (0 Pkts.) TX: 0.00 B (0 Pkts.)
Protocol	DHCP client
Hostname to send when requesting DHCP	VA_router
Accept router advertisements	<input type="checkbox"/>
Send router solicitations	<input type="checkbox"/>

Figure 128: The interface general setup configuration page for DHCP client protocol

Web Field/UCI/Package Option	Description																										
Web: Status UCI: network.<if name>.proto Opt: proto	<p>Shows the current status of the interface.</p> <p>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.</p> <p>Select DHCP Client.</p> <table border="1"> <thead> <tr> <th>Option</th><th>Description</th></tr> </thead> <tbody> <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> </tbody> </table>	Option	Description	Static	Static configuration with fixed address and netmask.	DHCP Client	Address and netmask are assigned by DHCP.	Unmanaged	Unspecified	IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.	IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.	GRE	Generic Routing Encapsulation protocol	IOT		L2TP	Layer 2 Tunnelling Protocol.	PPP	Point-to-Point protocol	PPPoE	PPP over Ethernet	PPPoATM	PPP over ATM	LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.
Option	Description																										
Static	Static configuration with fixed address and netmask.																										
DHCP Client	Address and netmask are assigned by DHCP.																										
Unmanaged	Unspecified																										
IPv6-in-IPv4 (RFC4213)	Used with tunnel brokers.																										
IPv6-over-IPv4	Stateless IPv6 over IPv4 transport.																										
GRE	Generic Routing Encapsulation protocol																										
IOT																											
L2TP	Layer 2 Tunnelling Protocol.																										
PPP	Point-to-Point protocol																										
PPPoE	PPP over Ethernet																										
PPPoATM	PPP over ATM																										
LTE/UMTS/GPRS/EV-DO	CDMA, UMTS or GPRS connection using an AT-style 3G modem.																										
Web: Hostname to send when requesting DHCP UCI: network.<if name>.hostname Opt: hostname	Defines the hostname to include in DHCP requests																										
Web: Accept router advertisements UCI: network.<if name>.accept_ra Opt: accept_ra	<p>Specifies whether to accept IPv6 Router Advertisements on this interface (optional).</p> <p>Note: default is 1 if protocol is set to DHCP, otherwise the setting defaults to 0.</p> <table border="1"> <tr> <td>0</td><td>Do not accept IPv6 Router advertisements</td></tr> <tr> <td>1</td><td>Accept IPv6 Router Advertisements</td></tr> </table>	0	Do not accept IPv6 Router advertisements	1	Accept IPv6 Router Advertisements																						
0	Do not accept IPv6 Router advertisements																										
1	Accept IPv6 Router Advertisements																										
Web: Send router solicitations UCI: network.<if name>.send_rs Opt: send_rs	<p>Specifies whether to send Router Solicitations on this interface (optional).</p> <p>Note: defaults to 1 for Static protocol, otherwise the setting defaults to 0.</p> <table border="1"> <tr> <td>0</td><td>Do not send router solicitations</td></tr> <tr> <td>1</td><td>Send router solicitations</td></tr> </table>	0	Do not send router solicitations	1	Send router solicitations																						
0	Do not send router solicitations																										
1	Send router solicitations																										

Table 82: Information table for general setup configuration settings for DHCP client protocol

26.2.3.2 Common configuration: advanced settings

The screenshot shows the 'Common Configuration' page with the 'Advanced Settings' tab selected. The page includes the following configuration options:

- Bring up on boot:** A checked checkbox.
- Monitor interface state:** An unchecked checkbox with a note: "This interface state would be reported to VA Monitor via keep-alive".
- Use broadcast flag:** An unchecked checkbox with a note: "Required for certain ISPs, e.g. Charter with DOCSIS 3".
- Use default gateway:** A checked checkbox with a note: "If unchecked, no default route is configured".
- Use DNS servers advertised by peer:** A checked checkbox with a note: "If unchecked, the advertised DNS server addresses are ignored".
- Use gateway metric:** A dropdown menu set to "0".
- Client ID to send when requesting DHCP:** An empty input field.
- Vendor Class to send when requesting DHCP:** An empty input field.
- Override MAC address:** An input field containing "00:E0:C8:D3:18:20".
- Override MTU:** An input field containing "1500".
- Dependant interfaces:** Two checkboxes: "ADSL" (unchecked) and "LAN3" (unchecked).

Figure 129: The interface advanced settings page for DHCP client protocol

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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Use broadcast flag UCI: network.<if name>.broadcast Opt: broadcast	Enables the broadcast flag in DHCP requests (required for certain ISPs) <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Use default gateway UCI: network.<if name>.gateway Opt: gateway	Defines whether to suppress the DHCP assigned default gateway. When disabled via web option gateway is set to 0.0.0.0 <table border="1"> <tr> <td>0</td><td>Disabled (option gateway set to 0.0.0.0)</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled (option gateway set to 0.0.0.0)	1	Enabled.
0	Disabled (option gateway set to 0.0.0.0)				
1	Enabled.				
Web: Use DNS servers advertised by peer UCI: n/a Opt: n/a	Defines whether to override DHCP assigned DNS servers with configured list of DNS servers. When unchecked allows configuration of custom DNS servers via web. There is no uci option set when checking or unchecking this option.				

Web: Use custom DNS servers UCI: network.<if name>.dns Opt: dns	Defines whether to override DHCP assigned DNS servers with configured list of DNS servers. Multiple DNS Servers are separated by a space if using UCI. Example: option dns '1.1.1.1 2.2.2.2' <table border="1"> <tr><td>0</td><td>Disabled (option gateway set to 0.0.0.0)</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled (option gateway set to 0.0.0.0)	1	Enabled.						
0	Disabled (option gateway set to 0.0.0.0)										
1	Enabled.										
Web: Use gateway metric UCI: network.<if name>.metric Opt: metric	Specifies the default route metric to use for this interface. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>Range</td><td></td></tr> </table>	0	Disabled.	Range							
0	Disabled.										
Range											
Web: Client ID to send when requesting DHCP UCI: network.<if name>.clientid Opt: clientid	Defines whether to override the client identifier in DHCP requests <table border="1"> <tr><td>(empty)</td><td>Do not override.</td></tr> <tr><td>Range</td><td>Override.</td></tr> </table>	(empty)	Do not override.	Range	Override.						
(empty)	Do not override.										
Range	Override.										
Web: Vendor Class to send when requesting DHCP UCI: network.<if name>.vendorid Opt: vendorid	Defines whether to override the vendor class in DHCP requests <table border="1"> <tr><td>(empty)</td><td>Do not override.</td></tr> <tr><td>Range</td><td>Override.</td></tr> </table>	(empty)	Do not override.	Range	Override.						
(empty)	Do not override.										
Range	Override.										
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. <table border="1"> <tr><td>1500</td><td>1500 bytes</td></tr> </table>	1500	1500 bytes								
1500	1500 bytes										
Web: Dependant Interfaces UCI: network.[if_name].dependants Opt: dependants	Lists interfaces that are dependent on this parent interface. Dependant interfaces will go down when parent interface is down and will start or restart when parent interface starts. Separate multiple interfaces by a space when using UCI. Example: option dependants 'PPPADSL MOBILE' This replaces the following previous options in child interfaces. <table border="1"> <tr><td>gre</td><td>option local_interface</td></tr> <tr><td>lt2p</td><td>option src_ipaddr</td></tr> <tr><td>iot</td><td>option wan1 wan2</td></tr> <tr><td>6in4</td><td>option ipaddr</td></tr> <tr><td>6to4</td><td>option ipaddr</td></tr> </table>	gre	option local_interface	lt2p	option src_ipaddr	iot	option wan1 wan2	6in4	option ipaddr	6to4	option ipaddr
gre	option local_interface										
lt2p	option src_ipaddr										
iot	option wan1 wan2										
6in4	option ipaddr										
6to4	option ipaddr										
Web: SNMP Alias ifIndex UCI: network.@interface[X].snmp_alias_ifindex Opt: snmp_alias_ifindex	Defines a static SNMP interface alias index for this interface, that can be polled using via the SNMP interface index (<i>snmp_alias_ifindex+1000</i>) <table border="1"> <tr><td>Blank</td><td>No SNMP interface alias index</td></tr> <tr><td>Range</td><td>0 - 4294966295</td></tr> </table>	Blank	No SNMP interface alias index	Range	0 - 4294966295						
Blank	No SNMP interface alias index										
Range	0 - 4294966295										

Table 83: Information table for advanced settings for DHCP client protocol

26.3 Configuring DHCP client using command line

The configuration files for DHCP client are stored on **/etc/config/network**

26.3.1 DHCP client using UCI

```
root@GW_router:~# uci show network
...
network.DHCPCCLIENTLAN=interface
network.DHCPCCLIENTLAN.proto=dhcp
```

```
network.DHCPClientLAN.ifname=eth3
network.DHCPClientLAN.monitored=0
network.DHCPClientLAN.broadcast=0
network.DHCPClientLAN.accept_ra=1
network.DHCPClientLAN.send_rs=0
network.DHCPClientLAN.metric=1
```

26.3.2 DHCP client using package options

```
root@GW_router:~# uci export network
package network
.....
config interface 'DHCPClientLAN'
    option proto 'dhcp'
    option ifname 'eth3'
    option monitored '0'
    option broadcast '0'
    option accept_ra '1'
    option send_rs '0'
    option metric '1'
```

26.4 DHCP client diagnostics

26.4.1 Interface status

To see IP address of DHCP client interface, enter **ifconfig**:

```
root@GW_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:

```

root@GW_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)

```

26.4.2 ARP table status

To show the current ARP table of the router, enter:

```

root@GW7314:~# arp
? (10.67.253.141) at 30:30:41:30:43:36 [ether] on eth8
? (10.47.48.1) at 0a:44:b2:06 [ether] on gre-gre1

```

26.4.3 Route status

To show the current routing status, enter:

```
root@GW_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.

27 Configuring DHCP forwarding

This section describes how to configure the router to forward DHCP requests from an interface to a network DHCP server.

27.1 Configuration packages used

Package	Sections
dhcp_fwd	dhcpfwd

27.2 Configuring DHCP forwarding using the web interface

To configure DHCP forwarding via the web interface, in the top menu, click **Network -> DHCP-Forwarder**.

The DHCP forwarder page appears. The web GUI creates a dhcpfwd section called main so this will be used in the uci examples below.

The screenshot shows the 'DHCP Forwarder' configuration page. It has a title 'DHCP Forwarder' and a subtitle 'Configuration of DHCP Forwarder'. Under 'Basic Settings', there is an 'Enable' checkbox which is checked. Below it, there is a list of 'Interfaces' with checkboxes next to them. The checked interfaces are 'ADSL' (unchecked), 'LAN3' (checked), and 'lan' (checked). Other interfaces listed are 'lan2' (unchecked), 'loopback' (unchecked), 'main2_voda' (unchecked), and 'wan' (unchecked). A note below says 'Interfaces to listen to for DHCP requests'. There is also a 'DHCP Servers' input field containing '1.1.1.1' with a plus icon to its right. At the bottom right, there are three buttons: 'Save & Apply', 'Save', and 'Reset'.

Figure 130: The DHCP Forwarder configuration page

Web Field/UCI/Package Option	Description				
Web: Enabled UCI: dhcp_fwd.main.enabled Opt: enabled	Defines whether DHCP forwarding is enabled or disabled. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Do not send router solicitations</td></tr> <tr> <td>1</td><td>Send router solicitations</td></tr> </table>	0	Do not send router solicitations	1	Send router solicitations
0	Do not send router solicitations				
1	Send router solicitations				
Web: Interfaces UCI: dhcp_fwd.main.listen_interface Opt: list listen_interface	Defines a list of the source interface name(s) to forward DHCP messages from. Multiple interface_name(s) are entered using uci set and uci add_list commands. Example: <pre>uci set dhcp_fwd.main.listen_interface=LAN1 uci add_list dhcp_fwd.main.listen_interface=LAN2</pre> or using a list of options via package options <pre>list listen_interface 'LAN1' list listen_interface 'LAN2'</pre>				
Web: DHCP Servers UCI: dhcp_fwd.main.server Opt: list server	Defines a list of the network DHCP servers to forward DHCP messages to. Multiple interface_name(s) are entered using uci set and uci add_list commands. Example: <pre>uci set dhcp_fwd.main.server=1.1.1.1 uci add_list dhcp_fwd.main.main.server=2.2.2.2</pre> or using a list of options via package options <pre>list server '1.1.1.1' list server '2.2.2.2'</pre>				

Table 84: Information table for the DHCP Forwarder section

27.3 Configuring DHCP forwarding using command line

The configuration files for DHCP client are stored in **/etc/config/dhcp_fwd**

27.3.1 DHCP forwarding using UCI

```
root@GW_router:~# uci show dhcp_fwd
dhcp_fwd.main=dhcpfwd
dhcp_fwd.main.enabled=1
dhcp_fwd.main.listen_interface=LAN3 lan2
dhcp_fwd.main.server=1.1.1.1
```

27.3.2 DHCP forwarding using package options

```
root@GW_router:~# uci export dhcp_fwd
package dhcp_fwd

config dhcpfwd 'main'
    option enabled '1'
    list listen_interface 'LAN3'
    list listen_interface 'lan2'
    list server '1.1.1.1'
```

27.4 DHCP forwarding over IPSec

DHCP messages are forwarded over the WAN interface using the IP address of the WAN interface as the source IP for the transmitted packet. This means that when forwarding over an IPSec tunnel a source NAT firewall rule is required to change the source IP to match an IPSec connection rule.

27.4.1 Configuration packages used

Package	Sections
firewall	redirect

27.4.2 Configuring source NAT for DHCP forwarding over IPsec

To enter a source NAT rule, browse to **Network -> Firewall**. Select **Traffic Rules** tab. The Firewall - Traffic Rules page appears. A source NAT rule that changes the source IP for UDP destination port 67 from the required LAN should be configured.

For more information on configuring a source NAT rule please refer to the 'Configuring Firewall' section of the User Manual.

The screenshot shows the 'Source NAT' section of the Firewall – Traffic Rules configuration page. It includes a table header for Name, Protocol, Source, Destination, SNAT, Enable, and Sort. Below the table, a message states 'This section contains no values yet'. A 'New source NAT:' form is present, with fields for Name (DHCPMessages), Source zone (lan), Destination zone (wan), To source IP (192.168.100.1), and To source port (left empty). Buttons at the bottom include 'Save & Apply', 'Save', and 'Reset'.

Figure 131: The Firewall – Traffic Rules configuration page

Web Field/UCI/Package Option	Description
Web: Name UCI: firewall.@redirect[X].name Opt: name	Defines a name for the source NAT rule
Web: Source Zone UCI: firewall.@redirect[X].src Opt: src	Defines the source interface for the source NAT rule. Select the interface where the DHCP requests are originating
Web: Destination Zone UCI: firewall.@redirect[X].dest Opt: dest	Defines destination interface for the source NAT rule. Select the interface where the DHCP requests are intended to be transmitted
Web: To source IP UCI: firewall.@redirect[X].src_dip Opt: src_dip	Defines the IP address to rewrite matched traffic source IP. Select the source IP address to match the required IPSec rule
Web: To source port UCI: firewall.@redirect[X].src_dport Opt: src_dport	Defines the port number to rewrite matched traffic source port number. Leave empty

Table 85: Information table for the source NAT configuration

Firewall - Traffic Rules - SNAT DHCPMessages

This page allows you to change advanced properties of the traffic rule entry, such as matched source and destination hosts.

Rule is enabled

Name:

Protocol: ⓘ You may specify multiple by selecting "custom" and then entering protocols separated by space.

Source zone:

Source MAC address:

Source IP address:

Source port: ⓘ Match incoming traffic originating from the given source port or port range on the client host.

Destination zone:

Destination IP address:

Destination port: ⓘ Match forwarded traffic to the given destination port or port range.

SNAT IP address: ⓘ Rewrite matched traffic to the given address.

SNAT port: ⓘ Rewrite matched traffic to the given source port. May be left empty to only rewrite the IP address.

Extra arguments: ⓘ Passes additional arguments to iptables. Use with care!

Figure 132: The Firewall – Traffic Rules – SNAT configuration page

Web Field/UCI/Package Option	Description																					
Web: Rule is enabled UCI: firewall.@redirect[X].enabled Opt: enabled	Defines whether source NAT rule is enabled <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled</td> </tr> <tr> <td>1</td> <td>Enabled</td> </tr> </table>	0	Disabled	1	Enabled																	
0	Disabled																					
1	Enabled																					
Web: Name UCI: firewall.@redirect[X].name Opt: name	Defines a name for the source NAT rule																					
Web: Protocol UCI: firewall.@redirect[X].proto Opt: proto	Defines the protocol for the source NAT rule to match. Select UDP . <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Option</th> <th>Description</th> <th>UCI</th> </tr> </thead> <tbody> <tr> <td>All protocols</td> <td>Match all protocols</td> <td>all</td> </tr> <tr> <td>TCP+UDP</td> <td>Match TCP and UDP protocols</td> <td>tcp upd</td> </tr> <tr> <td>TCP</td> <td>Match TCP protocol</td> <td>tcp</td> </tr> <tr> <td>UDP</td> <td>Match UDP protocol</td> <td>udp</td> </tr> <tr> <td>ICMP</td> <td>Match ICMP protocol</td> <td>icmp</td> </tr> <tr> <td>Custom</td> <td>Enter custom protocol</td> <td></td> </tr> </tbody> </table>	Option	Description	UCI	All protocols	Match all protocols	all	TCP+UDP	Match TCP and UDP protocols	tcp upd	TCP	Match TCP protocol	tcp	UDP	Match UDP protocol	udp	ICMP	Match ICMP protocol	icmp	Custom	Enter custom protocol	
Option	Description	UCI																				
All protocols	Match all protocols	all																				
TCP+UDP	Match TCP and UDP protocols	tcp upd																				
TCP	Match TCP protocol	tcp																				
UDP	Match UDP protocol	udp																				
ICMP	Match ICMP protocol	icmp																				
Custom	Enter custom protocol																					
Web: Source Zone UCI: firewall.@redirect[X].src Opt: src	Defines the source interface for the source NAT rule. Select the interface where the DHCP requests are originating																					
Web: Destination Zone UCI: firewall.@redirect[X].dest Opt: dest	Defines destination interface for the source NAT rule. Select the interface where the DHCP requests are intended to be transmitted																					
Web: Destination port UCI: firewall.@redirect[X].port Opt: port	Defines the destination port number to match. Select 67																					
Web: SNAT IP address UCI: firewall.@redirect[X].src_dip Opt: src_dip	Defines the IP address to rewrite matched traffic. Select the source IP address to match the required IPSec rule																					

Table 86: Information table for the advanced source NAT configuration

27.4.3 Configuring source NAT for DHCP forwarding over IPSec using command line

27.4.3.1 Source NAT for DHCP forwarding over IPSec using UCI

```
root@GW_router:~# uci show firewall
.....
firewall.@redirect[0]=redirect
firewall.@redirect[0].target=SNAT
firewall.@redirect[0].src=lan
firewall.@redirect[0].dest=wan
firewall.@redirect[0].src_dip=192.168.100.1
firewall.@redirect[0].name=DHCPMessages
firewall.@redirect[0].proto=udp
firewall.@redirect[0].dest_port=67
```

27.4.3.2 Source NAT for DHCP forwarding over IPSec using package options

```
root@GW_router:~# uci export firewall
package firewall
.....
config redirect
    option target 'SNAT'
    option src 'lan'
    option dest 'wan'
    option src_dip '192.168.100.1'
    option name 'DHCPMessages'
    option proto 'udp'
    option dest_port '67'
```

27.5 DHCP forwarding diagnostics

27.5.1 Tracing DHCP packets

To trace DHCP packets on any interface on the router, enter **tcpdump -i any -n -p port 67 &**

```
root@GW_router:~# tcpdump -i any -n -p port 67 &
root@GW_router:~# tcpdump: verbose output suppressed, use -v or -vv for
full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked), capture size 65535
bytes
16:39:20.666070 IP 0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP, Request
from 00:e0:c8:13:02:3d, length 360
16:39:20.666166 IP 0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP, Request
from 00:e0:c8:13:02:3d, length 360
```

To trace stop tracing enter **fg** (to bring tracing task to foreground), and then **<CTRL-C>** to stop the trace.

```
root@GW_router:~# fg
tcpdump -i any -n -p port 67
^C
33 packets captured
33 packets received by filter
0 packets dropped by kernel
```

```
16:39:20.666166 IP 0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP, Request  
from 00:e0:c8:13:02:3d, length 360
```

27.5.2 ARP table status

To show the current ARP table of the router, enter **arp**

```
root@GW7314:~# arp  
? (10.67.253.141) at 30:30:41:30:43:36 [ether]  on eth8  
? (10.47.48.1) at 0a:44:b2:06 [ether]  on gre-gre1
```

28 Configuring Dynamic DNS

28.1 Overview

Dynamic DNS (DDNS) functionality on a SATEL router will dynamically perform DDNS updates to a server so it can associate an IP address with a correctly associated DNS name. Users can then contact a machine, router, device and so on with a DNS name rather than a dynamic IP address.

An account is required with the provider, and one or more domain names are associated with that account. A dynamic DNS client on the router monitors the public IP address associated with an interface and whenever the IP address changes, the client notifies the DNS provider to update the corresponding domain name.

When the DNS provider responds to queries for the domain name, it sets a low lifetime, typically a minute or two at most, on the response so that it is not cached. Updates to the domain name are thus visible throughout the whole Internet with little delay.

Note: most providers impose restrictions on how updates are handled: updating when no change of address occurred is considered abusive and may result in an account being blocked. Sometimes, addresses must be refreshed periodically, for example, once a month, to show that they are still in active use.

28.2 Configuration packages used

Package	Sections
ddns	service

28.3 Configuring Dynamic DNS using the web interface

In the top menu, select **Services -> Dynamic DNS**. The Dynamic DNS Configuration page appears.

Name	Value
------	-------

Figure 133: The Dynamic DNS configuration page

Enter a text name that will be used for the dynamic DNS section in the configuration. Select **Add**. The Dynamic DNS configuration options appear.

28.3.1 Dynamic DNS settings

Dynamic DNS

Dynamic DNS allows that your router can be reached with a fixed hostname while having a dynamically changing IP address.

DDNS1

Enable	<input type="checkbox"/>
Service	-- custom --
Custom update-URL	<input type="text"/>
Hostname	<input type="text"/>
Username	<input type="text"/>
Password	<input type="password"/>
Source of IP address	network
Network	lan
Check for changed IP every	10
Check-time unit	min
Force update every	72
Force-time unit	h
Listen on	<input type="radio"/> dialin:

Figure 134: The dynamic DNS main settings page

Web Field/UCI/Package Option	Description						
Web: Enable UCI: ddns.<name>.enabled Opt: enabled	Enables a Dynamic DNS entry on the router. <table border="1" style="margin-left: 20px;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled</td></tr> </table>	0	Disabled.	1	Enabled		
0	Disabled.						
1	Enabled						
Web: Service UCI: ddns.<name>.service_name Opt: service_name	Defines the Dynamic DNS provider						
Web: Customer update-URL UCI: ddns.<name>.update_url Opt: update_url	Defines the customer DNS provider. Displayed when the service is set to custom in the web UI.						
Web: Hostname UCI: ddns.<name>.domain Opt: domain	Defines the fully qualified domain name associated with this entry. This is the name to update with the new IP address as needed.						
Web: Username UCI: ddns.<name>.username Opt: username	Defines the user name to use for authenticating domain updates with the selected provider.						
Web: Password UCI: ddns.<name>.password Opt: password	Defines the password to use for authenticating domain name updates with the selected provider.						
Web: Source of IP address UCI: ddns.<name>.ip_source Opt: ip_source	Defines the type of interface whose IP needs to be updated. <table border="1" style="margin-left: 20px;"> <tr> <td>network</td><td>IP is associated with a network configuration.</td></tr> <tr> <td>interface</td><td>IP is associated with an interface.</td></tr> <tr> <td>web</td><td>IP is associated with a URL.</td></tr> </table>	network	IP is associated with a network configuration.	interface	IP is associated with an interface.	web	IP is associated with a URL.
network	IP is associated with a network configuration.						
interface	IP is associated with an interface.						
web	IP is associated with a URL.						

Web: Network UCI: ddns.<name>.ip_network Opt: ip_network	Defines the network whose IP needs to be updated. Displayed when the Source of IP address option is set to network. All the configured network interfaces will be shown.				
Web: Interface UCI: ddns.<name>.ip_interface Opt: ip_interface	Defines the interface whose IP needs to be updated. Displayed when the Source of IP address option is set to interface. All the configured interfaces will be shown.				
Web: URL UCI: ddns.<name>.ip_url Opt: ip_url	Defines the URL where the IP downloaded from. Displayed when the Source of IP address option is set to URL.				
Web: Check for changed IP every UCI: ddns.<name>.check_interval Opt: check_interval	Defines how often to check for an IP change. Used in conjunction with check_unit. <table border="1"> <tr> <td>10</td> <td>.</td> </tr> <tr> <td>Range</td> <td></td> </tr> </table>	10	.	Range	
10	.				
Range					
Web: Check-time unit UCI: ddns.<name>.check_unit Opt: check_unit	Defines the time unit to use for check for an IP change. Used in conjunction with check_interval. <table border="1"> <tr> <td>minutes</td> <td></td> </tr> <tr> <td>hours</td> <td></td> </tr> </table>	minutes		hours	
minutes					
hours					
Web: Force update every UCI: ddns.<name>.force_interval Opt: force_interval	Defines how often to force an IP update to the provider. Used in conjunction with force_unit. <table border="1"> <tr> <td>72</td> <td>Disabled.</td> </tr> <tr> <td>Range</td> <td>Enabled</td> </tr> </table>	72	Disabled.	Range	Enabled
72	Disabled.				
Range	Enabled				
Web: Force-time unit UCI: ddns.<name>.force_unit Opt: force_unit	Defines the time unit to use for check for an IP change. Used in conjunction with force_interval. <table border="1"> <tr> <td>minutes</td> <td></td> </tr> <tr> <td>hours</td> <td></td> </tr> </table>	minutes		hours	
minutes					
hours					
Web: Listen on UCI: ddns.<name>.interface Opt: interface	Defines the interface for ddns monitoring. Typically this will be the same as the interface whose IP is being updated – as defined ip_network or ip_interface. All configured interfaces will be displayed.				

Table 87: Information table for dynamic DNS settings

28.4 Dynamic DNS using UCI

Dynamic DNS uses the ddns package **/etc/config/ddns**

28.4.1 UCI commands for DDNS

```
root@GW_router:~# uci show ddns
ddns.ddns1=service
ddns.ddns1.enabled=1
ddns.ddns1.service_name=dyndns.org
ddns.ddns1.domain=fqdn_of_interface
ddns.ddns1.username=testusername
ddns.ddns1.password=testpassword
ddns.ddns1.ip_source=network
ddns.ddns1.ip_network=dsl0
ddns.ddns1.check_interval=10
```

```
ddns.ddns1.check_unit=minutes
ddns.ddns1.force_interval=72
ddns.ddns1.force_unit=hours
ddns.ddns1.interface=dsl0
Package options for DDNS
root@GW_router:~# uci export ddns
package ddns

config service 'ddns1'
    option enabled '1'
    option service_name 'dyndns.org'
    option domain 'fqdn_of_interface'
    option username 'test'
    option password 'test'
    option ip_source 'network'
    option ip_network 'dsl0'
    option check_interval '10'
    option check_unit 'minutes'
    option force_interval '72'
    option force_unit 'hours'
    option interface 'dsl0'
```

29 Configuring hostnames

29.1 Overview

Hostnames are human-readable names that identify a device connected to a network.

There are several different ways in which hostnames can be configured and used on the router.

- Local host file records
- PTR Records
- Static DHCP leases

29.2 Local host file records

The hosts file is an operating system file that maps hostnames to IP addresses. It is used preferentially to other name resolution methods such as DNS.

The hosts file contains lines of text consisting of an IP address in the first text field followed by one or more host names. Each field is separated by white space – tabs are often preferred for historical reasons, but spaces are also used. Comment lines may be included; they are indicated by an octothorpe (#) in the first position of such lines. Entirely blank lines in the file are ignored.

By default, the routers local host file contains:

```
127.0.0.1 localhost
::1 ip6-localhost ip6-loopback
```

The local host file is stored at **/etc/hosts**

29.2.1 Configuration packages used

Package	Sections
network	host

29.2.2 Configuring local host files entries using the web interface

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

Browse to **Host Records** section at the bottom of the page.

Host Records	
Hostname	IP-Address
Hostname1	1.1.1.1

This section contains no values yet

Add

Figure 135: The host records add page

Select **Add**. Enter a hostname and IP address and select **Save & Apply**.

Host Records	
Hostname	IP-Address
Hostname1	1.1.1.1

Delete

Add

Figure 136: The host records configuration page

Web Field/UCI/Package Option	Description
Web: Hostname UCI: network.host.hostname Opt: hostname	Defines the hostname.
Web: IP-Address UCI: network.host.addr Opt: addr	Defines the IP address associated with the hostname.

Table 88: Information table for host records settings

29.2.3 Local host records using command line

Local host records are configured in the **host** section of the network package **/etc/config/network**.

Multiple **host** can be configured.

By default, all host instances are named host, it is identified by `@host` then the host position in the package as a number. For example, for the first host in the package using UCI:

```
network.@host[0]=host
network.@host[0].hostname=Device1
```

Or using package options:

```
config host
    option hostname 'Device1'
```

29.2.3.1 Local host records using uci

```
root@GW_router:~# uci show network
.....
```

```
network.@host[0]=host
network.@host[0].hostname=Device1
network.@host[0].addr=1.1.1.1
```

29.2.3.2 Local host records using package option

```
root@GW_router:~# uci export network
package network
.....
config host
    option hostname 'Device1'
    option addr '1.1.1.1'
```

29.2.4 Local host records diagnostics

29.2.4.1 Hosts file

Local host records are written to the local hosts file stored at **/etc/hosts**. To view the local hosts file, enter:

```
root@GW_router:~# cat /etc/hosts
127.0.0.1 localhost
::1 ip6-localhost ip6-loopback
1.1.1.1 Device1
```

29.3 PTR records

PTR records are used for Reverse DNS.

The primary purpose for DNS is to map domains to IP addresses. A pointer record works in the opposite way; it associates an IP address with a domain name.

29.3.1 Configuration packages used

Package	Sections
dhcp	domain

29.3.2 Configuring PTR records using the web interface

In the top menu, select **Network -> Hostnames**. The Hostnames configuration page appears.

Hostnames	
Host entries	
Hostname	IP address
Domain1	2.2.2.2

This section contains no values yet

Add

Figure 137: The hostnames add page

Select **Add**. Enter a hostname and IP address for the PTR record and select **Save & Apply**.

Hostnames	
Host entries	
Hostname	IP address
Domain1	2.2.2.2

Add

Figure 138: The hostnames configuration page

Web Field/UCI/Package Option	Description
Web: Hostname UCI: dhcp.domain.name Opt: name	Defines the domain name for the PTR record.
Web: IP-Address UCI: dhcp.domain.ip Opt: ip	Defines the IP address associated with the domain name.

Table 89: Information table for hostnames settings

29.3.3 PTR records using command line

PTR records are configured in the **domain** section of the **dhcp** package.
/etc/config/dhcp.

Multiple **domains** can be configured.

By default, all domain instances are named domain. It is identified by `@domain` then the domain position in the package as a number. For example, for the first domain in the package using UCI:

```
dhcp.@domain[0]=domain
dhcp.@domain[0].name=Domain1
```

Or using package options:

```
config domain
    option name 'Domain1'
```

29.3.3.1 PTR records using uci

```
root@GW_router:~# uci show dhcp
.....
dhcp.@domain[0]=domain
dhcp.@domain[0].name=Domain1
dhcp.@domain[0].ip=2.2.2.2
```

29.3.3.2 PTR records using package option

```
root@GW_router:~# uci export dhcp
package dhcp
.....
config domain
    option name 'Domain1'
    option ip '2.2.2.2'
```

29.3.4 PTR records diagnostics

29.3.4.1 PTR records table

To view PTR records, enter:

```
root@GW_router:~# pgrep -fl dnsmasq
4724 /usr/sbin/dnsmasq -K -D -y -Z -b -E -s lan -S /lan/ -l
/tmp/dhcp.leases -r /tmp/resolv.conf.auto --stop-dns-rebind --rebind-
localhost-ok -A /Device1.lan/1.1.1.1 --ptr-record=1.1.1.1.in-
addr.arpa,Device1.lan -A /Device2.lan/2.2.2.2 --ptr-record=2.2.2.2.in-
addr.arpa,Device2.lan
```

29.4 Static leases

Static leases are used to assign fixed IP addresses and symbolic hostnames to DHCP clients based on their MAC (hardware) address.

They are also required for non-dynamic interface configurations where only hosts with a corresponding lease are served.

29.4.1 Configuration packages used

Package	Sections
dhcp	host

29.4.2 Configuring static leases using the web interface

In the top menu, select **Network -> DHCP and DNS**. The DHCP and DNS configuration page appears.

Browse to **Static leases** section.

Hostname	MAC-Address	IPv4-Address
<i>This section contains no values yet</i>		
<input type="button" value="Add"/>		

Figure 139: The static leases add page

Select **Add**. Enter a hostname, MAC address and IP address for the static lease. Select **Save & Apply**.

Hostname	MAC-Address	IPv4-Address	
host1	aa:bb:cc:dd:ee:ff	4.4.4.4	<input type="button" value="Delete"/>
<input type="button" value="Add"/>			

Figure 140: The static leases configuration page

Web Field/UCI/Package Option	Description
Web: Hostname UCI: dhcp.host.name Opt: name	Defines the symbolic hostname to assign.
Web: MAC-Address UCI: dhcp.host.mac Opt: mac	Defines the MAC address for this host. MAC addresses should be entered in the format aa:bb:cc:dd:ee:ff
Web: IPv4-Address UCI: dhcp.host.ip Opt: ip	Defines the IP address to be used for this host.

Table 90: Information table for static leases settings

29.4.3 Static leases using command line

Static leases are configured in the **host** section of the **dhcp** package **/etc/config/dhcp**.

Multiple **hosts** can be configured.

By default, all **dhcp** host instances are named **host**. It is identified by **@host** then the host position in the package as a number. For example, for the first host in the package using UCI:

```
dhcp.@host[0]=host
dhcp.@host[0].name=Host1
```

Or using package options:

```
config host
    option name 'Host1'
```

29.4.3.1 Static leases using uci

```
root@GW_router:~# uci show dhcp
.....
dhcp.@host[0]=host
dhcp.@host[0].name=Host1
dhcp.@host[0].mac=aa:bb:cc:dd:ee:ff
dhcp.@host[0].ip=4.4.4.4
```

29.4.3.2 Static leases using package option

```
root@GW_router:~# uci export dhcp
package dhcp
.....
config host
    option name 'Host1'
    option mac 'aa:bb:cc:dd:ee:ff'
    option ip '4.4.4.4'
```

30 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'.

30.1 Configuration package used

Package	Sections
firewall	

30.2 Configuring firewall using the web interface

In the top menu, select **Network -> Firewall**. The Firewall page appears. It is divided into four sections:

Section	Description
General Zone Settings	Defines the firewall zones, both global and specific.
Port Forwards	Port Forwards are also known as Redirects. This section creates the redirects using DNAT (Destination Network Address Translation) with Netfilter.
Traffic Rules	Defines rules to allow or restrict access to specific ports, hosts or protocols.

30.2.1 Firewall: zone settings

Zone settings is divided into two sections:

Section	Description
General Settings	Defines the global firewall settings that do not belong to any specific zones.
Zones	The zones 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.

30.2.1.1 Firewall general settings

The General Settings page, 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.

The screenshot shows the 'Firewall - Zone Settings' page. At the top, there are tabs for 'General Settings', 'Port Forwards', and 'Traffic Rules'. The 'General Settings' tab is selected. It contains sections for 'Enable SYN-flood protection' (checkbox checked), 'Drop invalid packets' (checkbox unchecked), and three dropdown menus for 'Input' (accept), 'Output' (accept), and 'Forward' (accept). Below this is a 'Zones' section with a table showing two entries. The first entry is 'lan: LAN1: LAN2: LAN3: > wan' with policies 'accept', 'accept', 'accept', and 'drop'. The second entry is 'wan: MOBILE1: PoADSL: > lan' with policies 'accept', 'accept', 'accept', and 'mss clamping' (checkbox checked). There are 'Edit' and 'Delete' buttons for each row, and an 'Add' button at the bottom left.

Figure 141: The firewall zone general 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. 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: Forward UCI: firewall.defaults.forward Opt: forward	Default policy for the Forward 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.	

Table 91: Information table for general zone general settings page

30.2.1.2 Firewall zones

The Zones 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. To view a zone's settings, click **Edit**.

The number of concurrent dynamic/static NAT entries of any kind (NAT/PAT/DNAT/SNAT) is not limited in any way by software; the only hardware limitation is the amount of RAM installed on the device.

30.2.1.3 Firewall zone: general settings

General Settings **Port Forwards** **Traffic Rules**

Firewall - Zone Settings - Zone "lan"

Zone "lan"

This section defines common properties of "lan". The *input* and *output* options set the default policies for traffic entering and leaving this zone while the *forward* option describes the policy for forwarded traffic between different networks within the zone. *Covered networks* specifies which available networks are member of this zone.

General Settings **Advanced Settings**

Name	lan
Input	accept
Output	accept
Forward	accept
Masquerading	<input type="checkbox"/>
MSS clamping	<input type="checkbox"/>
Covered networks	<input checked="" type="checkbox"/> LAN1: (no interfaces attached) <input checked="" type="checkbox"/> LAN2: <input checked="" type="checkbox"/> LAN3: <input type="checkbox"/> MOBILE1: <input type="checkbox"/> PoAADS1: <input type="checkbox"/> loopback:

Figure 142: The firewall zone general settings

Web Field/UCI/Package Option	Description						
Web: name UCI: firewall.<zone label>.name Opt: name	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.						
Web: Input UCI: firewall.<zone label>.input Opt: input	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 border="1"> <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>	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.
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.<zone label>.output Opt: output	<p>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.</p> <table border="1"> <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>	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.
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: Forward UCI: firewall.<zone label>.forward Opt: forward	<p>Default policy for internal zone traffic between interfaces. Forward rules for a zone describe what happens to traffic passing between different interfaces within that zone.</p> <table border="1"> <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>	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.
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: Masquerading UCI: firewall.<zone label>.masq Opt: masq	Specifies whether outgoing zone traffic should be masqueraded (NATTED). This is typically enabled on the wan zone.						
Web: MSS Clamping UCI: firewall.<zone label>.mtu_fix Opt: mtu_fix	<p>Enables MSS clamping for outgoing zone traffic. Subnets are allowed.</p> <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.		
0	Disabled.						
1	Enabled.						
Web: Covered networks UCI: firewall.<zone label>.network Opt: network	<p>Defines a list of interfaces attached to this zone, if omitted, the value of name is used by default.</p> <p>Note: use the uci list syntax to edit this setting through UCI.</p>						

Table 92: Information table for firewall zone general settings

30.2.1.4 Firewall zone: advanced settings

Zone "lan"

This section defines common properties of "lan". The *input* and *output* options set the default policies for traffic entering and leaving this zone while the *forward* option describes the policy for forwarded traffic between different networks within the zone. *Covered networks* specifies which available networks are member of this zone.

General Settings	Advanced Settings
Restrict to address family	IPv4 and IPv6
Restrict Masquerading to given source subnets	0.0.0.0/0
Restrict Masquerading to given destination subnets	0.0.0.0/0
Force connection tracking	<input type="checkbox"/>
Enable logging on this zone	<input type="checkbox"/>
Allow NAT Reflections	<input checked="" type="checkbox"/>

Figure 143: Firewall zone advanced settings

Web Field/UCI/Package Option	Description												
Web: Restrict to address family UCI: firewall.<zone label>.family Opt: family	Restricts zone to IPv4, IPv6 or both IPv4 and IPv6. <table border="1"> <thead> <tr> <th>Option</th><th>Description</th><th>UCI</th></tr> </thead> <tbody> <tr> <td>IPv4 and IPv6</td><td>Any address family</td><td>any</td></tr> <tr> <td>IPv4 only</td><td>IPv4 only</td><td>ipv4</td></tr> <tr> <td>IPv6 only</td><td>IPv6 only</td><td>Ipv6</td></tr> </tbody> </table>	Option	Description	UCI	IPv4 and IPv6	Any address family	any	IPv4 only	IPv4 only	ipv4	IPv6 only	IPv6 only	Ipv6
Option	Description	UCI											
IPv4 and IPv6	Any address family	any											
IPv4 only	IPv4 only	ipv4											
IPv6 only	IPv6 only	Ipv6											
Web: Restrict Masquerading to given source subnets. UCI: firewall.<zone label>.masq_src Opt: masq_src	Limits masquerading to the given source subnets. Negation is possible by prefixing the subnet with '!'. Multiple subnets are allowed.												
Web: Restrict Masquerading to given destination subnets. UCI: firewall.<zone label>.masq_dest Opt: masq_dest	Limits masquerading to the given destination subnets. Negation is possible by prefixing the subnet with '!'. Multiple subnets are allowed. Multiple IP addresses/subnets should be separated by a space, for example:option masq_dest '1.1.1.1 2.2.2.0/24'												
Web: Force connection tracking UCI: firewall.<zone label>.conntrack Opt: conntrack	Forces connection tracking for this zone. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>If masquerading is used. Otherwise, default is 0.</td></tr> </table>	0	Disabled.	1	If masquerading is used. Otherwise, default is 0.								
0	Disabled.												
1	If masquerading is used. Otherwise, default is 0.												
Web: Enable logging on this zone UCI: firewall.<zone label>.log Opt: log	Creates log rules for rejected and dropped traffic in this zone.												
Web: Allow NAT reflections UCI: firewall.<zone label>.reflection Opt: reflection	Enable/disable all NAT reflections for this zone. <table border="1"> <tr> <td>0</td><td>Disable reflection.</td></tr> <tr> <td>1</td><td>Enable reflection.</td></tr> </table>	0	Disable reflection.	1	Enable reflection.								
0	Disable reflection.												
1	Enable reflection.												
Web: n/a UCI: firewall.<zone label>.log_limit Opt: log_limit	Limits the amount of log messages per interval.												

Table 93: Information table for firewall zone advanced settings

30.2.1.5 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.

The options below control the forwarding policies between this zone (lan) and other zones. Destination zones cover forwarded traffic originating from "lan". Source zones match forwarded traffic from other zones targeted at "lan". The forwarding rule is *unidirectional*, e.g. a forward from lan to wan does *not* imply a permission to forward from wan to lan as well.

Allow forward to destination zones:	<input checked="" type="checkbox"/> wan: MOBILE1: PoADSL:
Allow forward from source zones:	<input checked="" type="checkbox"/> wan: MOBILE1: PoADSL:

Figure 144: 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 94: 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.

30.2.2 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.

To edit an existing port forward select **edit**.

To add a new port forward select **add**.

The screenshot shows the 'Firewall - Port Forwards' section of a web-based configuration interface. At the top, there are three tabs: 'General Settings', 'Port Forwards' (which is selected), and 'Traffic Rules'. Below the tabs, the title 'Firewall - Port Forwards' is displayed, followed by a brief description: 'Port forwarding allows remote computers on the Internet to connect to a specific computer or service within the private LAN.' A table lists existing port forwards, with columns for Name, Protocol, Source, Via, Destination, Enable, and Sort. One entry is shown: 'HTTPS' (Protocol: TCP) from 'any host in wan' to 'any router IP at port 443' forwarded to 'IP 192.168.100.100, port 443 in lan'. Below the table is a 'New port forward:' form with fields for Name, Protocol (set to 'TCP+UDP'), External port, Internal IP address, and Internal port. Buttons for 'Add' and 'Cancel' are at the bottom of the form.

Figure 145: 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. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Option</th> <th>Description</th> <th>UCI</th> </tr> </thead> <tbody> <tr> <td>tcp+udp</td> <td>Match either TCP or UDP packets</td> <td>tcp udp</td> </tr> <tr> <td>tcp</td> <td>Match TCP packets only</td> <td>tcp</td> </tr> <tr> <td>udp</td> <td>Match UDP packets only</td> <td>udp</td> </tr> </tbody> </table>	Option	Description	UCI	tcp+udp	Match either TCP or UDP packets	tcp udp	tcp	Match TCP packets only	tcp	udp	Match UDP packets only	udp
Option	Description	UCI											
tcp+udp	Match either TCP or UDP packets	tcp udp											
tcp	Match TCP packets only	tcp											
udp	Match UDP packets only	udp											
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. <table border="1" style="margin-left: 20px;"> <tr> <td>(empty)</td> <td>Match traffic to any port</td> </tr> <tr> <td>Range</td> <td>1 - 65535</td> </tr> </table>	(empty)	Match traffic to any port	Range	1 - 65535								
(empty)	Match traffic to any port												
Range	1 - 65535												
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.												

Table 95: 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).

Firewall - Port Forwards - (Unnamed Entry)

This page allows you to change advanced properties of the port forwarding entry. In most cases there is no need to modify those settings.

Rule is enabled

Name: Forward

Protocol: TCP+UDP

Source zone: wan: MOBILE1: PoAADS:

Source MAC address: any Only match incoming traffic from these MACs.

Source IP address: any Only match incoming traffic from this IP or range.

Source port: any Only match incoming traffic originating from the given source port or port range on the client host.

External IP address: any Only match incoming traffic directed at the given IP address.

External port: Match incoming traffic directed at the given destination port or port range on this host.

Internal zone: wan: MOBILE1: PoAADS:

Internal IP address: Redirect matched incoming traffic to the specified internal host.

Internal port: any Redirect matched incoming traffic to the given port on the internal host.

Enable NAT Loopback:

Extra arguments: Passes additional arguments to iptables. Use with care!

Figure 146: The firewall port forwards edits page

Web Field/UCI/Package Option	Description													
Web: Rule is enabled	Specifies if this redirect should be enabled or disabled.													
UCI: firewall.<redirect label>.enabled	0	Disabled.												
Opt: enabled	1	Enabled.												
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.													
	<table border="1"> <thead> <tr> <th>Option</th><th>Description</th><th>UCI</th></tr> </thead> <tbody> <tr> <td>tcp+udp</td><td>Match either TCP or UDP packets</td><td>tcp udp</td></tr> <tr> <td>tcp</td><td>Match TCP packets only</td><td>tcp</td></tr> <tr> <td>udp</td><td>Match UDP packets only</td><td>udp</td></tr> </tbody> </table>	Option	Description	UCI	tcp+udp	Match either TCP or UDP packets	tcp udp	tcp	Match TCP packets only	tcp	udp	Match UDP packets only	udp	
Option	Description	UCI												
tcp+udp	Match either TCP or UDP packets	tcp udp												
tcp	Match TCP packets only	tcp												
udp	Match UDP packets only	udp												
Web: Source zone 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.													

Web: Source MAC address UCI: firewall.<redirect label>.src_mac Opt: list src_mac	Defines the list of source MAC addresses that this redirect will match Format: aa:bb:cc:dd:ee:ff Multiple RIP interfaces are entered using <code>uci set</code> and <code>uci add_list</code> commands. Example: <code>uci set firewall.@rediect[0].src_mac=aa:bb:cc:dd:ee:ff</code> <code>uci add_list</code> <code>firewall.@rediect[0].src_mac=12:34:56:78:90:12</code> or using a list of options via package options <code>list network 'aa:bb:cc:dd:ee:ff'</code> <code>list network '12:34:56:78:90:12'</code>				
Web: Source IP address UCI: firewall.<redirect label>.src_ip Opt: src_ip	Defines a source IP address that this redirect will match. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>(empty)</td><td>Match traffic from any source IP.</td></tr> <tr> <td>Range</td><td>A.B.C.D/mask</td></tr> </table>	(empty)	Match traffic from any source IP.	Range	A.B.C.D/mask
(empty)	Match traffic from any source IP.				
Range	A.B.C.D/mask				
Web: Source port UCI: firewall.<redirect label>.src_port Opt: src_port	Defines a source IP port that this redirect will match. Multiple ports can be entered using a space separator. Example: <code>option src_port '22 23'</code> *see note below on use with options src_dport and dest_port <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>(empty)</td><td>Match traffic from any source port.</td></tr> <tr> <td>Range</td><td>1 - 65535</td></tr> </table>	(empty)	Match traffic from any source port.	Range	1 - 65535
(empty)	Match traffic from any source port.				
Range	1 - 65535				
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 in format start:stop, for example, 2001:2020. Multiple ports can be entered using a space separator. Example: <code>option src_dport '22 23'</code> *see note below on use with options src_port and dest_port <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>(empty)</td><td>Match traffic to any port.</td></tr> <tr> <td>Range</td><td>1 - 65535</td></tr> </table>	(empty)	Match traffic to any port.	Range	1 - 65535
(empty)	Match traffic to any port.				
Range	1 - 65535				
Web: Internal zone UCI: firewall.<redirect label>.dest Opt: dest	Specifies the traffic destination zone, must refer to one of the defined zone names.				
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. Multiple ports can be entered using a space separator. *For example: <code>option dest_port '22 23'</code> *see note below table on use with options src_port and src_dport.				
Web: Enable NAT Loopback UCI: firewall.<redirect label>.reflection Opt: reflection	Enable or disable NAT reflection for this redirect. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>reflection disabled</td></tr> <tr> <td>1</td><td>reflection enabled</td></tr> </table>	0	reflection disabled	1	reflection enabled
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 <code>-m policy --dir in</code> for IPSec. The arguments are entered as text strings.				

Table 96: Information table for port forward edits fields

***Note:** redirect rule options src_port and src_dport/dest_port accept space-separated lists of ports. If src_port is a list, then src_dport/dst_port cannot be, to avoid ambiguity.

If src_dport/dest_port are lists of different lengths, then the missing values of the shorter list default to the corresponding port in the other list. For example, if configuration file is:

```
option src_dport '21 22 23'
option dest_port '21 22 23 24'
```

then the firmware will interpret the values as:

```
option src_dport '21 22 23 24'
option dest_port '21 22 23 24'
```

30.2.3 Firewall traffic rules

Rules can be defined to allow or restrict access to specific ports, hosts or protocols.

The screenshot shows the 'Firewall - Traffic Rules - (Unnamed Rule)' configuration page. The page has tabs at the top: General Settings, Port Forwards, and Traffic Rules (which is selected). Below the tabs, there is a note: 'This page allows you to change advanced properties of the traffic rule entry, such as matched source and destination hosts.' The configuration fields include:

- Rule is enabled:** A button labeled 'Disable'.
- Name:** An input field containing a dash '-'.
- Restrict to address family:** A dropdown menu set to 'IPv4 and IPv6'.
- Protocol:** A dropdown menu set to 'TCP+UDP'.
- Match ICMP type:** A dropdown menu set to 'any'.
- Source zone:** A radio button group where 'Any zone' is selected.
- Source MAC address:** An input field containing 'any'.
- Source address:** An input field containing 'any'.
- Source port:** An input field containing 'any'.
- Destination zone:** A radio button group where 'Device (input)' is selected.
- Destination address:** An input field containing 'any'.
- Destination port:** An input field containing 'any'.
- Action:** A dropdown menu set to 'accept'.
- Extra arguments:** An input field with a note: 'Passes additional arguments to iptables. Use with care!' followed by a help icon.

Figure 147: The firewall traffic rules page

Web Field/UCI/Package Option	Description																		
Web: Rule is enabled UCI: firewall.<rule label>.enabled Opt: enabled	Enables or disables traffic rule. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Rule is disabled.</td></tr> <tr> <td>1</td><td>Rule is enabled.</td></tr> </table>	0	Rule is disabled.	1	Rule is enabled.														
0	Rule is disabled.																		
1	Rule is enabled.																		
Web: Name UCI: firewall.<rule label>.name Opt: name	Select a descriptive name limited to less than 11 characters.																		
Web: Restrict to address family UCI: firewall.<rule label>.family Opt: family	Restrict to protocol family. <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Option</th><th>Description</th><th>UCI</th></tr> </thead> <tbody> <tr> <td>IPv4 and IPv6</td><td>Traffic rule applies to any address family</td><td>any</td></tr> <tr> <td>IPv4 only</td><td>IPv4 only</td><td>ipv4</td></tr> <tr> <td>IPv6 only</td><td>IPv6 only</td><td>Ipv6</td></tr> </tbody> </table>	Option	Description	UCI	IPv4 and IPv6	Traffic rule applies to any address family	any	IPv4 only	IPv4 only	ipv4	IPv6 only	IPv6 only	Ipv6						
Option	Description	UCI																	
IPv4 and IPv6	Traffic rule applies to any address family	any																	
IPv4 only	IPv4 only	ipv4																	
IPv6 only	IPv6 only	Ipv6																	
Web: Protocol UCI: firewall.<rule label>.proto Opt: proto	Matches incoming traffic using the given protocol. <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Option</th><th>Description</th><th>UCI</th></tr> </thead> <tbody> <tr> <td>TCP+UDP</td><td>Applies rule to TCP and UDP only</td><td>tcp udp</td></tr> <tr> <td>TCP</td><td>Applies rule to TCP only</td><td>tcp</td></tr> <tr> <td>UDP</td><td>Applies rule to UDP only</td><td>udp</td></tr> <tr> <td>ICMP</td><td>Applies rule to ICMP only</td><td>icmp</td></tr> <tr> <td>custom</td><td>Specify protocol from /etc/protocols</td><td></td></tr> </tbody> </table>	Option	Description	UCI	TCP+UDP	Applies rule to TCP and UDP only	tcp udp	TCP	Applies rule to TCP only	tcp	UDP	Applies rule to UDP only	udp	ICMP	Applies rule to ICMP only	icmp	custom	Specify protocol from /etc/protocols	
Option	Description	UCI																	
TCP+UDP	Applies rule to TCP and UDP only	tcp udp																	
TCP	Applies rule to TCP only	tcp																	
UDP	Applies rule to UDP only	udp																	
ICMP	Applies rule to ICMP only	icmp																	
custom	Specify protocol from /etc/protocols																		
Web: Match ICMP type UCI: firewall.<rule label>.icmp_type Opt: icmp_type	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 ICMP Options table below.																		
Web: Source zone UCI: firewall.<rule label>.src Opt: src	Specifies the traffic source zone, must refer to one of the defined zone names. For typical port forwards, this is usually WAN.																		
Web: Source MAC address UCI: firewall.<rule label>.src_mac Opt: src_mac	Matches incoming traffic from the specified MAC address. The MAC address must be entered in the following format: aa:bb:cc:dd:ee:ff: To only match the first portion of the MAC address append /prefix to the option value, where prefix defines the bits from the start of the MAC to match on. Example: option src_mac 00:E0:C8:12:34:56/24 will match on all packets with prefix 00:E0:C8.																		
Web: Source address UCI: firewall.<rule label>.src_ip Opt: src_ip	Matches incoming traffic from the specified source IP address.																		
Web: Source port UCI: firewall.<rule label>.src_port Opt: src_port	Matches incoming traffic originating from the given source port or port range on the client host.																		
Web: Destination zone UCI: firewall.<rule label>.dest Opt: dest	Specifies the traffic destination zone. Must refer to one of the defined zone names.																		
Web: Destination address UCI: firewall.<rule label>.dest_ip Opt: dest_ip	For DNAT, redirects matched incoming traffic to the specified internal host. For SNAT, matches traffic directed at the given address.																		

Web: Destination port UCI: firewall.<rule label>.dest_port Opt: dest_port	For DNAT, redirects matched incoming traffic to the given port on the internal host. For SNAT, matches traffic directed at the given ports.															
Web: Action UCI: firewall.<rule label>.target Opt: target	Action to take when rule is matched. <table border="1" data-bbox="666 348 1373 595"> <thead> <tr> <th>Option</th> <th>Description</th> <th>UCI</th> </tr> </thead> <tbody> <tr> <td>drop</td> <td>Drop matching traffic</td> <td>DROP</td> </tr> <tr> <td>accept</td> <td>Allow matching traffic</td> <td>ACCEPT</td> </tr> <tr> <td>reject</td> <td>Reject matching traffic</td> <td>REJECT</td> </tr> <tr> <td>don't track</td> <td>Disable connection tracking for the rule. See the Connection tracking section below for more information.</td> <td>NOTRACK</td> </tr> </tbody> </table>	Option	Description	UCI	drop	Drop matching traffic	DROP	accept	Allow matching traffic	ACCEPT	reject	Reject matching traffic	REJECT	don't track	Disable connection tracking for the rule. See the Connection tracking section below for more information.	NOTRACK
Option	Description	UCI														
drop	Drop matching traffic	DROP														
accept	Allow matching traffic	ACCEPT														
reject	Reject matching traffic	REJECT														
don't track	Disable connection tracking for the rule. See the Connection tracking section below for more information.	NOTRACK														
Web: Extra arguments UCI: firewall.<rule 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.															
Web: n/a UCI: firewall.<rule label>.reflection Opt: reflection	Disables NAT reflection for this redirect if set to 0. Applicable to DNAT targets.															
Web: n/a UCI: firewall.<rule label>.limit Opt: limit	Sets maximum average matching rate; specified as a number, with an optional /second, /minute, /hour or /day suffix. Example 3/hour.															
Web: n/a UCI: firewall.<rule label>.limit_burst Opt: limit_burst	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.															
Web: n/a UCI: firewall.<rule label>.recent Opt: recent	Sets number of allowed connections within specified time. This command takes two values e.g. recent=2 120 will allow 2 connections within 120 seconds.															

Table 97: Information table for firewall traffic rules

ICMP Options	ICMP Options	ICMP Options	ICMP Options
address-mask-reply	host-redirect	pong	time-exceeded
address-mask-request	host-unreachable	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 98: Information table for match ICMP type drop-down menu

30.3 Configuring firewall using UCI

Firewall is configured under the firewall package /etc/config/firewall.

There are three config sections: defaults, zone, forwarding, redirect, rule and include.

You can configure multiple zone, forwarding and redirect sections.

30.3.1 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.

30.3.2 Firewall zone settings

By default, all firewall zone instances are named zone, instances are identified by @zone then the zone position in the package as a number. For example, for the first zone in the package using UCI:

```
firewall.@zone[0]=zone
firewall.@zone[0].name=lan
```

Or using package options:

```
config zone
    option name 'lan'
```

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
```

30.3.3 Inter-zone forwarding

By default, all inter-zone instances are named forwarding, instances are identified by @forwarding then the forwarding position in the package as a number. For example, for the first forwarding in the package using UCI:

```
firewall.@forwarding[0]=forwarding
firewall.@forwarding[0].src=lan
```

Or using package options:

```
config forwarding
    option src 'lan'
```

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
```

30.3.4 Firewall port forwards

By default, all port forward instances are named redirect, instances are identified by @redirect then the redirect position in the package as a number. For example, for the first redirect in the package using UCI:

```
firewall.@redirect[0]=redirect
firewall.@redirect[0].name=Forward
```

Or using package options:

```
config redirect
    option name 'Forward'
```

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
```

30.3.5 Firewall traffic rules

By default, all traffic rule instances are named rule, instances are identified by @rule then the rule position in the package as a number. For example, for the first rule in the package using UCI:

```
firewall.@rule[0]=rule
firewall.@rule[0].enabled=1
```

Or using package options:

```
config rule
    option enabled '1'
```

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
```

30.3.5.1 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.

30.4 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.

30.5 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)

30.6 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`.

30.7 Firewall examples

30.7.1 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.

30.7.2 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'
```

30.7.3 Source NAT (SNAT)

Source NAT changes an outgoing packet destined for the system so that it 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.

30.7.4 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
```

30.7.5 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
```

30.7.6 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
```

30.7.7 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
```

30.7.8 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
```

30.7.9 Denial of service protection rule

The example below shows a sample configuration of SSH DoS attack where if more than two SSH connections are attempted within 120 seconds, every further connection will be dropped. You can configure this for any port number.

```
config rule 'sshattack'
    option src 'lan'
    option dest_port '22'
    option proto 'tcp'
    option recent '2 120'
    option target 'DROP'
```

30.7.10 IP spoofing prevention mechanism

Configure IP spoofing protection on a per interface basis in the /etc/config/network configuration file. The example below shows the ipv4_rp_filter option enabled on the Vlan12 interface in the network file. When reverse path filtering mechanism is enabled, the router will check whether a receiving packet source address is routable.

If it is routable through the interface from which it came, then the machine will accept the packet

If it is not routable through the interface from which it came, then the machine will drop that packet.

```
config interface 'Vlan12'
    option type 'bridge'
    option proto 'static'
    option monitored '0'
    option ipaddr '10.1.28.122'
    option netmask '255.255.0.0'
    option ifname 'eth1 eth3.12'
    option ipv4_rp_filter '1'
```

30.7.11 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
```

30.7.12 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

```

30.7.13 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

```

30.7.14 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

```

30.7.15 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.

30.7.16 Firewall management

After a configuration change, to rebuild firewall rules, enter:

```
root@GW_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@GW_router:/# /etc/init.d/firewall stop
```

To manually start the firewall, enter:

```
root@GW_router:/# /etc/init.d/firewall start
```

To permanently disable the firewall, enter:

```
root@GW_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@GW_router:/# /etc/init.d/firewall enable
```

30.7.17 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@GW_router:/# FW_TRACE=1 fw reload
```

To direct the output to a file for later inspection, enter:

```
root@GW_router:/# FW_TRACE=1 fw reload 2>/tmp/iptables.lo
```

31 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. SATEL 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)'.

The number of IPSec tunnels supported by SATEL' routers is not limited in any way by software; the only hardware limitation is the amount of RAM installed on the device.

31.1 Configuration package used

Package	Sections
strongswan	general connection secret

31.2 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	
Secret Settings	Together, these sections define the required parameters for a two-way IKEv1 tunnel.

31.2.1 Configure common settings

The screenshot shows the 'strongSwan IPsec VPN' configuration page. The title is 'strongSwan IPsec VPN' and the subtitle is 'Configuration of the strongSwan IPsec VPN system.' The page contains several configuration options:

- Enable StrongSwan IPsec:** A checked checkbox.
- Strict CRL Policy:** A dropdown menu set to 'no'. A tooltip explains: 'Defines if a fresh CRL must be available in order for the peer authentication based on RSA signatures to succeed. IKEv2 additionally recognizes 'furi' which reverts to 'yes' if at least one CRL URI is defined and to 'no' if no URI is known.'
- Unique IDs:** A dropdown menu set to 'yes'. A tooltip explains: '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. The IKEv2 daemon also accepts the value 'replace' which is identical to 'yes' and the value 'keep' to reject new IKE SA setups and keep the duplicate established earlier.'
- Cache CRLs:** A checked checkbox. A tooltip says: 'CRLs fetched via HTTP or LDAP will be cached.'
- Disable Revocation (CRL and OCSP):** An unchecked checkbox.
- Send INITIAL CONTACT by default:** A checked checkbox. A tooltip says: 'Send INITIAL CONTACT notification when first connection attempt for all connections'
- Debug:** A dropdown menu set to 'none'.

Figure 148: 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.cachecharts Opt: cachecharts	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: Disable Revocation UCI: strongswan.general.revocation_disabled Opt: revocation_disabled	Defines whether disable CRL and OCSP checking for revoked certificates.	
	0	Disabled.
	1	Enabled.
Web: Send INITIAL CONTACT by default UCI: strongswan.general.initial_contact Opt: initial_contact	Defines whether the first attempt to contact a remote peer by this strongswan instance sets the initial_contact flag, which should cause compliant peers to automatically bring down any previous sessions. This can also be enabled/disabled per connection.	
	0	Does not set initial contact flag.
	1	Sets initial contact flag on first attempt.
Web: Debug UCI: strongswan.general.debug Opt: debug	Enables 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 99: Information table for IPSec common settings

31.2.2 Common settings: configure connection

Status ▾ System ▾ Services ▾ Network ▾ Logout

UNSAVED CHANGES 10
00E0C8122C89
VIE-16.00.55
Image2/config2

Connections

Enabled

Aggressive Mode

Name

Autostart Action Operation on startup. **add** loads a connection without starting it. **route** loads a connection and installs kernel traps. If traffic is detected between localan and remotelan, a connection is established. **start** loads a connection and brings it up immediately. **ignore** do nothing

Connection Type

Delete

Figure 149: The configuring IPSec settings

Web Field/UCI/Package Option	Description										
Web: Enabled UCI: strongswan.@connection[X].enabled Opt: enable	Enables or disables IPSec connection. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
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. <table border="1"> <tr> <td>start</td><td>On start up.</td></tr> <tr> <td>route</td><td>When traffic routes this way.</td></tr> <tr> <td>add</td><td>Loads a connection without starting it.</td></tr> <tr> <td>ignore</td><td>Ignores the connection.</td></tr> <tr> <td>always</td><td>Actively retries to establish the tunnel if it went down.</td></tr> </table>	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.
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. <table border="1"> <tr> <td>tunnel</td><td>Connection uses tunnel mode.</td></tr> <tr> <td>transport</td><td>Connection uses transport mode.</td></tr> <tr> <td>pass</td><td>Connection does not perform any IPSec processing.</td></tr> <tr> <td>drop</td><td>Connection drops all the packets.</td></tr> </table>	tunnel	Connection uses tunnel mode.	transport	Connection uses transport mode.	pass	Connection does not perform any IPSec processing.	drop	Connection drops all the packets.		
tunnel	Connection uses tunnel mode.										
transport	Connection uses transport mode.										
pass	Connection does not perform any IPSec processing.										
drop	Connection drops all the packets.										

Table 100: Information table for connection settings

31.2.3 Common settings: IP addressing

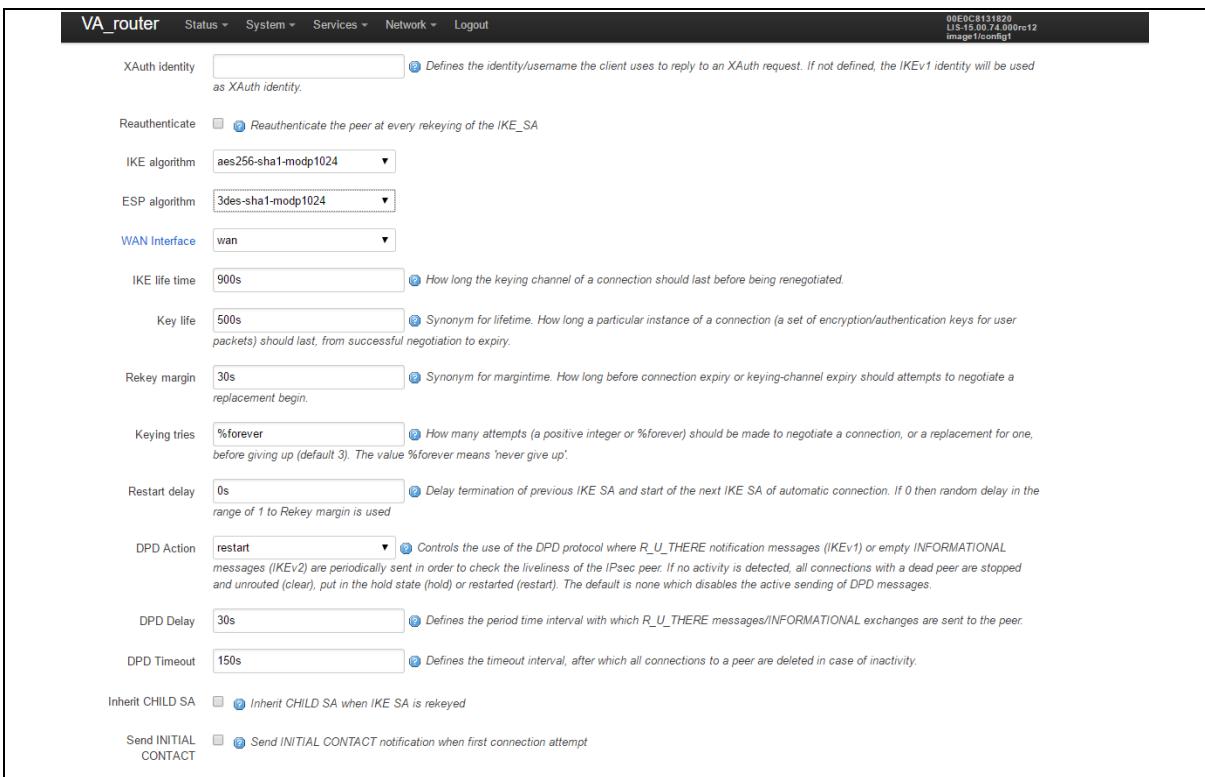
Figure 150: The IP addressing settings

Web Field/UCI/Package Option	Description
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.

Web: Local Port UCI: strongswan.@connection[X].localport Opt: localport	Restricts the connection to a single port on the local side.														
Web: Remote Protocol UCI: strongswan.@connection[X].remoteproto Opt: remoteproto	Restricts the connection to a single protocol on the remote side.														
Web: Remote Port UCI: strongswan.@connection[X].remoteport Opt: remoteport	Restricts the connection to a single port on the remote side.														
Web: Authby UCI: strongswan.@connection[X].authby Opt: authby	<p>Defines how the two secure gateways should authenticate. Note: using aggressive mode along with PSK authentication is unsecure and should be avoided.</p> <table border="1"> <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>	Pubkey	For public key signatures.	Rsasig	For RSA digital signatures.	ecdsasig	For Elliptic Curve DSA signatures.	Psk	Using a preshared key.	xauthrsasig	Enables eXtended Authentication (XAuth) with addition to RSA signatures.	xauthpsk	Using extended authentication and preshared key.	never	Can be used if negotiation is never to be attempted or accepted (shunt connections).
Pubkey	For public key signatures.														
Rsasig	For RSA digital signatures.														
ecdsasig	For Elliptic Curve DSA signatures.														
Psk	Using a preshared key.														
xauthrsasig	Enables eXtended Authentication (XAuth) with addition to RSA signatures.														
xauthpsk	Using extended authentication and preshared key.														
never	Can be used if negotiation is never to be attempted or accepted (shunt connections).														

Table 101: Information table for IP addressing settings

31.2.4 Common settings: IPSec settings



The screenshot shows the 'VA_router' configuration interface with the 'Status' tab selected. The main title is 'IPSec connections settings'. The configuration fields include:

- XAuth identity: A text input field with a tooltip explaining it defines the identity/username the client uses to reply to an XAuth request. If not defined, the IKEv1 identity will be used as XAuth identity.
- Reauthenticate: A checkbox with a tooltip explaining it reauthenticates the peer at every rekeying of the IKE_SA.
- IKE algorithm: A dropdown menu set to 'aes256-sha1-modp1024'.
- ESP algorithm: A dropdown menu set to '3des-sha1-modp1024'.
- WAN Interface: A dropdown menu set to 'wan'.
- IKE life time: A text input field set to '900s' with a tooltip explaining it is how long the keying channel of a connection should last before being renegotiated.
- Key life: A text input field set to '500s' with a tooltip explaining it is a synonym for lifetime. How long a particular instance of a connection (a set of encryption/authentication keys for user packets) should last, from successful negotiation to expiry.
- Rekey margin: A text input field set to '30s' with a tooltip explaining it is a synonym for margintime. How long before connection expiry or keying-channel expiry should attempts to negotiate a replacement begin.
- Keying tries: A text input field set to '%forever' with a tooltip explaining it is how many attempts (a positive integer or %forever) should be made to negotiate a connection, or a replacement for one, before giving up (default 3). The value %forever means 'never give up'.
- Restart delay: A text input field set to '0s' with a tooltip explaining it is the delay termination of previous IKE SA and start of the next IKE SA of automatic connection. If 0 then random delay in the range of 1 to Rekey margin is used.
- DPD Action: A dropdown menu set to 'restart' with a tooltip explaining it controls the use of the DPD protocol where R_U_THERE notification messages (IKEv1) or empty INFORMATIONAL messages (IKEv2) are periodically sent in order to check the liveness of the IPsec peer. If no activity is detected, all connections with a dead peer are stopped and unrouted (clear), put in the hold state (hold) or restarted (restart). The default is none which disables the active sending of DPD messages.
- DPD Delay: A text input field set to '30s' with a tooltip explaining it defines the period time interval with which R_U_THERE messages/INFORMATIONAL exchanges are sent to the peer.
- DPD Timeout: A text input field set to '150s' with a tooltip explaining it defines the timeout interval, after which all connections to a peer are deleted in case of inactivity.
- Inherit CHILD SA: A checkbox with a tooltip explaining it inherits CHILD SA when IKE SA is rekeyed.
- Send INITIAL CONTACT: A checkbox with a tooltip explaining it sends INITIAL CONTACT notification when first connection attempt.

Figure 151: The IPSec connections settings

Web Field/UCI/Package Option	Description
Web: XAuth Identity UCI: strongswan.@connection[X].xauth_identity Opt: xauth_identity	Defines Xauth ID.
Web: IKE Algorithm UCI: strongswan.@connection[X].ike Opt: ike	<p>Specifies the IKE algorithm to use. The format is: encAlgo authAlgo DHGroup</p> <p>encAlgo:</p> <ul style="list-style-type: none"> 3des aes128 aes256 serpent twofish blowfish <p>authAlgo:</p> <ul style="list-style-type: none"> md5 sha sha2 <p>DHGroup:</p> <ul style="list-style-type: none"> modp1024 modp1536 modp2048 modp3072 modp4096 modp6144 modp8192 <p>For example, a valid IKE algorithm is aes128-sha-modp1536.</p>
Web: ESP algorithm UCI: strongswan.@connection[X].esp Opt: esp	<p>Specifies the esp algorithm to use. The format is: encAlgo authAlgo DHGroup</p> <p>encAlgo:</p> <ul style="list-style-type: none"> 3des aes128 aes256 serpent twofish blowfish <p>authAlgo:</p> <ul style="list-style-type: none"> md5 sha sha2 <p>DHGroup:</p> <ul style="list-style-type: none"> modp1024 modp1536 modp2048 modp3072 modp4096 modp6144 modp8192 <p>For example, a valid encryption algorithm is: aes128-sha-modp1536.</p> <p>If no DH group is defined then PFS is disabled.</p>

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. <table border="1"> <tr> <td>3h</td> <td></td> </tr> <tr> <td>Timespec</td> <td>1d, 3h, 25m, 10s.</td> </tr> </table>	3h		Timespec	1d, 3h, 25m, 10s.				
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). <table border="1"> <tr> <td>1h</td> <td></td> </tr> <tr> <td>Timespec</td> <td>1d, 1h, 25m, 10s.</td> </tr> </table>	1h		Timespec	1d, 1h, 25m, 10s.				
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. <table border="1"> <tr> <td>9m</td> <td></td> </tr> <tr> <td>Timespec</td> <td>1d, 2h, 9m, 10s.</td> </tr> </table>	9m		Timespec	1d, 2h, 9m, 10s.				
9m									
Timespec	1d, 2h, 9m, 10s.								
Web: Restart Delay UCI: strongswan.@connection[X].restartdelay Opt: restartdelay	Defines specific delay when re-establishing a connection. Previously if <code>close_action=restart</code> , then new option <code>restartdelay</code> controls how many seconds it waits before attempting to re-establish the tunnel (to allow head-end some time to tidy up). If not set, it defaults to zero, which means that the previous behaviour of choosing a random time interval in the range 0.. <code>RekeyMargin</code> seconds takes effect. Relevant only locally, other end need not agree on it. <table border="1"> <tr> <td>0</td> <td></td> </tr> <tr> <td>Timespec</td> <td>1d, 2h, 9m, 10s.</td> </tr> </table>	0		Timespec	1d, 2h, 9m, 10s.				
0									
Timespec	1d, 2h, 9m, 10s.								
Web: Keying 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. <table border="1"> <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>	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.
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. <table border="1"> <tr> <td>30s</td> <td></td> </tr> <tr> <td>Timespec</td> <td>1d, 2h, 25m, 10s.</td> </tr> </table>	30s		Timespec	1d, 2h, 25m, 10s.				
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. <table border="1"> <tr> <td>150s</td> <td></td> </tr> <tr> <td>Timespec</td> <td>1d, 2h, 25m, 10s.</td> </tr> </table>	150s		Timespec	1d, 2h, 25m, 10s.				
150s									
Timespec	1d, 2h, 25m, 10s.								

Web: Inherit CHILD SA UCI: strongswan.@connection[X].inherit_child Opt: inherit_child	Defines whether the existing phase two IPSEC SA is maintained through IKE rekey for this tunnel. This is normally set to match the behaviour on the IPSEC headend. <table border="1"><tr><td>0</td><td>Delete the existing IPSEC SA on IKE rekey</td></tr><tr><td>1</td><td>Maintain the existing IPSEC SA on IKE rekey</td></tr></table>	0	Delete the existing IPSEC SA on IKE rekey	1	Maintain the existing IPSEC SA on IKE rekey
0	Delete the existing IPSEC SA on IKE rekey				
1	Maintain the existing IPSEC SA on IKE rekey				
Web: Send INITIAL CONTACT UCI: strongswan.@connection[X].initial_contact Opt: initial_contact	Defines whether the first attempt to contact a remote peer by this strongswan instance sets the initial_contact flag which should cause compliant peers to automatically bring down any previous sessions. <table border="1"><tr><td>0</td><td>Do not set initial contact flag</td></tr><tr><td>1</td><td>Set initial contact flag on first attempt</td></tr></table>	0	Do not set initial contact flag	1	Set initial contact flag on first attempt
0	Do not set initial contact flag				
1	Set initial contact flag on first attempt				

Table 102: Information table for IPsec connections settings

31.2.5 Configure secret settings

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

The screenshot shows a web-based configuration interface for 'Secrets'. It has a table with columns: Enabled, ID selector, Secret Type, and Secret. There are two rows of data:

- Row 1: Local IP: 192.168.208.1, Remote IP: 189.101.154.151, Secret Type: psk, Secret: secret
- Row 2: Local IP: 192.168.208.1, Remote IP: 192.168.100.2, Secret Type: psk, Secret: secret

Below the table are 'Add', 'Save & Apply', 'Save', and 'Reset' buttons.

Figure 152: 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. <table border="1"><tr><td>0</td><td>Disabled.</td></tr><tr><td>1</td><td>Enabled.</td></tr></table>	0	Disabled.	1	Enabled.
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.				

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

Table 103: Information table for IPSec secrets settings

31.3 Configuring IPSec using UCI

31.3.1 Common settings

```
# 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.cachecls=no
uci set strongswan.general.debug=none
uci set strongswan.general.initial_contact=0
uci commit
```

This will create the following output:

```
config general 'general'
    option enabled 'yes'
    option strictcrlpolicy 'no'
    option uniqueids 'yes'
    option cachecls 'no'
    option debug 'none'
    option initial_contact '0'
```

31.3.2 Connection settings

```
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].restartdelay=0
uci set strongswan.@connection[0].dpdaction=none
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].inherit_child=0
uci set strongswan.@connection[0].initial_contact=0
uci commit

```

This will create the following output:

```

config connection
    option ikelifetime '3h'
    option keylife '1h'
    option rekeymargin '9m'
    option keyingtries '3'
    option restartdelay '0'
    option dpdaction 'none'
    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 inherit_child '0'
option initial_contact '0'

```

31.3.3 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.

31.3.4 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=userfqn
uci set strongswan.@secret[1].userfqn=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 output:

```
config secret
    option enabled 'yes'
    option idtype 'userfqn'
    option userfqn 'testxauth'
    option remoteaddress '100.100.100.100'
    option secret 'xauth'
    option secrettype 'XAUTH'
```

31.4 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	

31.4.1 Configure common settings

The screenshot shows the 'strongSwan IPsec VPN' configuration page. At the top, there are links for 'Services', 'Network', and 'Logout'. A blue bar at the top right indicates 'UNSAVED CHANGES 3'. The main section is titled 'strongSwan IPsec VPN' with the subtitle 'Configuration of the strongSwan IPsec VPN system.' Below this, there are several configuration options:

- Enable StrongSwan IPsec:** A checked checkbox.
- Strict CRL Policy:** A dropdown menu set to 'no'. A tooltip explains: 'Defines if a fresh CRL must be available in order for the peer authentication based on RSA signatures to succeed. IKEv2 additionally recognizes 'ifuri' which reverts to 'yes' if at least one CRL URI is defined and to 'no' if no URI is known.'
- Unique IDs:** A dropdown menu set to 'yes'. A tooltip explains: '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. The IKEv2 daemon also accepts the value 'replace' which is identical to 'yes' and the value 'keep' to reject new IKE SA setups and keep the duplicate established earlier.'
- Cache CRLs:** A dropdown menu set to 'none'. A tooltip explains: 'CRLs fetched via HTTP or LDAP will be cached.'
- Debug:** A dropdown menu set to 'none'.

Figure 153: 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.cachecls Opt: cachecls	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 104: Information table for IPSec common settings

31.4.2 Configure connection settings

Scroll down to view the connection settings section.

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

Enabled	<input checked="" type="checkbox"/>
Aggressive Mode	<input checked="" type="checkbox"/>
Name	DMVPN_VDF
Autostart Action	ignore <small>(Operation on startup. add loads a connection without starting it. route loads a connection and installs kernel traps. If traffic is detected between localan and remotelan, a connection is established. start loads a connection and brings it up immediately. ignore do nothing)</small>
Connection Type	transport
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	
Local Protocol	gre <small>(Restrict the traffic selector to a single protocol on the local side)</small>
Local Port	
Remote Protocol	gre <small>(Restrict the traffic selector to a single protocol on the remote side)</small>
Remote Port	
Authby	psk <small>(How the two security gateways should authenticate each other.)</small>
XAuth identity	
IKE algorithm	aes128-sha1-modp1024
ESP algorithm	3des-md5
WAN Interface	3GVDF
IKE life time	3h <small>(How long the keying channel of a connection should last before being renegotiated.)</small>
Key life	1h <small>(Synonym for lifetime. How long a particular instance of a connection (a set of encryption/authentication keys for user packets) should last, from successful negotiation to expiry.)</small>
Rekey margin	9m <small>(Synonym for margintime. How long before connection expiry or keying-channel expiry should attempts to negotiate a replacement begin.)</small>
Keying tries	3 <small>(How many attempts (a positive integer or %forever) should be made to negotiate a connection, or a replacement for one, before giving up (default 3). The value %forever means 'never give up').</small>
DPD Action	none <small>(Controls the use of the DPD protocol where R_U_THERE notification messages (IKEv1) or empty INFORMATIONAL messages (IKEv2) are periodically sent in order to check the liveness of the IPsec peer. If no activity is detected, all connections with a dead peer are stopped and unrouteed (clear), put in the hold state (hold) or restarted (restart). The default is none which disables the active sending of DPD messages.)</small>
DPD Delay	30s <small>(Defines the period time interval with which R_U_THERE messages/INFORMATIONAL exchanges are sent to the peer.)</small>
DPD Timeout	30s <small>(Defines the timeout interval, after which all connections to a peer are deleted in case of inactivity.)</small>

Figure 154: The connections settings section

Web Field/UCI/Package Option	Description										
Web: Enabled UCI: strongswan.@connection[X].enabled Opt: enable	Enables or disables IPSec connection. <table border="1" style="margin-left: 20px;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
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. <table border="1" style="margin-left: 20px;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
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. <table border="1" style="margin-left: 20px;"> <tr> <td>start</td><td>On start up.</td></tr> <tr> <td>route</td><td>When traffic routes this way.</td></tr> <tr> <td>add</td><td>Loads a connection without starting it.</td></tr> <tr> <td>ignore</td><td>Ignores the connection.</td></tr> <tr> <td>always</td><td>Actively retries to establish the tunnel if it went down.</td></tr> </table>	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.
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. <table border="1" style="margin-left: 20px;"> <tr> <td>tunnel</td><td>Connection uses tunnel mode.</td></tr> <tr> <td>transport</td><td>Connection uses transport mode.</td></tr> <tr> <td>pass</td><td>Connection does not perform any IPSec processing.</td></tr> <tr> <td>drop</td><td>Connection drops all the packets.</td></tr> </table>	tunnel	Connection uses tunnel mode.	transport	Connection uses transport mode.	pass	Connection does not perform any IPSec processing.	drop	Connection drops all the packets.		
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 blank for DMVPN.										
Web: Local ID UCI: strongswan.@connection[X].localid Opt: localid	Defines the local peer identifier. Leave blank for DMVPN.										
Web: Remote ID UCI: strongswan.@connection[X].remoteid Opt: remoteid	Defines the remote peer identifier. Leave blank for DMVPN.										
Web: Local LAN IP Address UCI: strongswan.@connection[X].locallan Opt: locallan	Defines the local IP of LAN. Leave blank for DMVPN.										
Web: Local LAN IP Address Mask UCI: strongswan.@connection[X].locallanmask Opt: locallanmask	Defines the subnet of local LAN. Leave blank for DMVPN.										
Web: Remote LAN IP Address UCI: strongswan.@connection[X].remotelan Opt: remotelan	Defines the IP address of LAN serviced by remote peer. Leave blank for DMVPN.										
Web: Remote LAN IP Address Mask UCI: strongswan.@connection[X].remotelanmask Opt: remotelanmask	Defines the Subnet of remote LAN. Leave blank for DMVPN.										
Web: Local Protocol UCI: strongswan.@connection[X].localproto Opt: localproto	Restricts the connection to a single protocol on the local side.										

Web: Local Port UCI: strongswan.@connection[X].localport Opt: localport	Restricts the connection to a single port on the local side.														
Web: Remote Protocol UCI: strongswan.@connection[X].remoteproto Opt: remoteproto	Restricts the connection to a single protocol on the remote side.														
Web: Remote Port UCI: strongswan.@connection[X].remoteport Opt: remoteport	Restricts the connection to a single port on the remote side.														
Web: Authby UCI: strongswan.@connection[X].authby Opt: authby	<p>Defines how the two secure gateways should authenticate. Note: using aggressive mode along with PSK authentication is unsecure and should be avoided.</p> <table border="1"> <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>	Pubkey	For public key signatures.	Rsasig	For RSA digital signatures.	ecdsasig	For Elliptic Curve DSA signatures.	Psk	Using a preshared key.	xauthrsasig	Enables eXtended Authentication (XAuth) with addition to RSA signatures.	xauthpsk	Using extended authentication and preshared key.	never	Can be used if negotiation is never to be attempted or accepted (shunt connections).
Pubkey	For public key signatures.														
Rsasig	For RSA digital signatures.														
ecdsasig	For Elliptic Curve DSA signatures.														
Psk	Using a preshared key.														
xauthrsasig	Enables eXtended Authentication (XAuth) with addition to RSA signatures.														
xauthpsk	Using extended authentication and preshared key.														
never	Can be used if negotiation is never to be attempted or accepted (shunt connections).														
Web: XAuth Identity UCI: strongswan.@connection[X].xauth_identity Opt: xauth_identity	Defines Xauth ID.														
Web: IKE Algorithm UCI: strongswan.@connection[X].ike Opt: ike	<p>Specifies the IKE algorithm to use. The format is: encAlgo authAlgo DHGroup: encAlgo: 3des aes128 aes256 serpent twofish blowfish authAlgo: md5 sha sha2 DHGroup: modp1024 modp1536 modp2048 modp3072 modp4096 modp6144 modp8192 For example, a valid IKE algorithm is: aes128-sha-modp1536.</p>														

<p>Web: ESP algorithm UCI: strongswan.@connection[X].esp Opt: esp</p>	<p>Specifies the esp algorithm to use. The format is: encAlgo authAlgo DHGroup encAlgo: 3des aes128 aes256 serpent twofish blowfish authAlgo: md5 sha sha2 DHGroup: modp1024 modp1536 modp2048 modp3072 modp4096 modp6144 modp8192 For example, a valid encryption algorithm is: aes128-sha-modp1536. <u>If no DH group is defined then PFS is disabled.</u></p>				
<p>Web: WAN Interface UCI: strongswan.@connection[X].waniface Opt: waniface</p>	<p>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'.</p>				
<p>Web: IKE Life Time UCI: strongswan.@connection[X].ikelifetime Opt: ikelifetime</p>	<p>Specifies how long the keyring channel of a connection (ISAKMP or IKE SA) should last before being renegotiated.</p> <table border="1" data-bbox="716 1304 1410 1372"> <tr> <td>3h</td> <td></td> </tr> <tr> <td>Timespec</td> <td>1d, 3h, 25m, 10s.</td> </tr> </table>	3h		Timespec	1d, 3h, 25m, 10s.
3h					
Timespec	1d, 3h, 25m, 10s.				
<p>Web: Key Life UCI: strongswan.@connection[X].keylife Opt: keylife</p>	<p>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).</p> <table border="1" data-bbox="716 1522 1410 1596"> <tr> <td>1h</td> <td></td> </tr> <tr> <td>Timespec</td> <td>1d, 1h, 25m, 10s.</td> </tr> </table>	1h		Timespec	1d, 1h, 25m, 10s.
1h					
Timespec	1d, 1h, 25m, 10s.				
<p>Web: Rekey Margin UCI: strongswan.@connection[X].rekeymargin Opt: rekeymargin</p>	<p>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.</p> <table border="1" data-bbox="716 1686 1410 1760"> <tr> <td>9m</td> <td></td> </tr> <tr> <td>Timespec</td> <td>1d, 2h, 9m, 10s.</td> </tr> </table>	9m		Timespec	1d, 2h, 9m, 10s.
9m					
Timespec	1d, 2h, 9m, 10s.				
<p>Web: Keyring Tries UCI: strongswan.@connection[X].keyringtries Opt: keyringtries</p>	<p>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.</p>				

Web: DPD Action UCI: strongswan.@connection[X].dpdaction Opt: dpdaction	Defines DPD (Dead Peer Detection) action. <table border="1"> <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>	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.
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. <table border="1"> <tr><td>30s</td><td></td></tr> <tr><td>Timespec</td><td>1d, 2h, 25m, 10s.</td></tr> </table>	30s		Timespec	1d, 2h, 25m, 10s.				
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. <table border="1"> <tr><td>150s</td><td></td></tr> <tr><td>Timespec</td><td>1d, 2h, 25m, 10s.</td></tr> </table>	150s		Timespec	1d, 2h, 25m, 10s.				
150s									
Timespec	1d, 2h, 25m, 10s.								

Table 105: Information table for IPSec connections settings

31.4.3 Configure secret settings

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

Secrets			
Enabled	ID selector	Secret Type	Secret
<i>To match local/remote ip enter local ip followed by space followed by remote ip</i>			
<i>This section contains no values yet</i>			
<input type="button" value="Add"/> <input type="button" value="Save & Apply"/> <input type="button" value="Save"/> <input type="button" value="Reset"/>			

Figure 155: 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. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
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.										
Web: Secret Type UCI: strongswan.@secret[X].secrettype Opt: secrettype	Specifies the authentication mechanism to be used by the two peers. <table border="1" data-bbox="690 348 1373 527"> <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>	Psk	Preshared secret	Pubkey	Public key signatures	Rsasig	RSA digital signatures	Ecdsasig	Elliptic Curve DSA signatures	Xauth	Extended authentication
Psk	Preshared secret										
Pubkey	Public key signatures										
Rsasig	RSA digital signatures										
Ecdsasig	Elliptic Curve DSA signatures										
Xauth	Extended authentication										
Web: Secret UCI: strongswan.@secret[X].secret Opt: secret	Defines the secret.										

Table 106: Information table for IPSec secret settings

31.5 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.cachecls=yes
uci set strongswan.general.nat traversal=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-shal-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'
```

31.6 IPSec diagnostics using the web interface

31.6.1 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 156: 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.

31.7 IPSec diagnostics using UCI

31.7.1 IPSec configuration

To view IPSec configuration via UCI, enter:

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

To restart strongSwan, enter:

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

31.7.2 IPSec status

31.7.3 To view IPSec status, enter:

```
root@GW_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@GW_router:~# ipsec -help
```

32 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.

32.1 Prerequisites for configuring DMVPN

Before configuring DMVPN, you must first configure:

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

32.2 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 SATEL 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.

32.3 DMVPN scenarios

32.3.1 Scenario 1

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

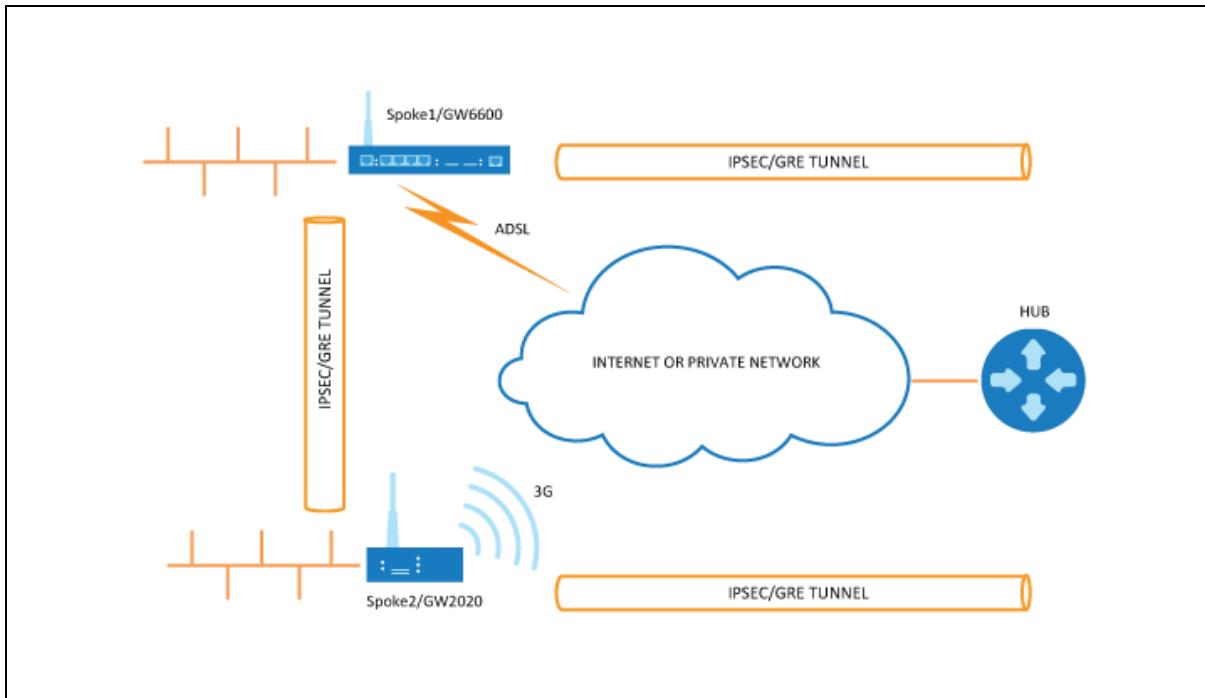


Figure 157: Network diagram for DMVPN spoke to spoke

- 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.

32.3.2 Scenario 2

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

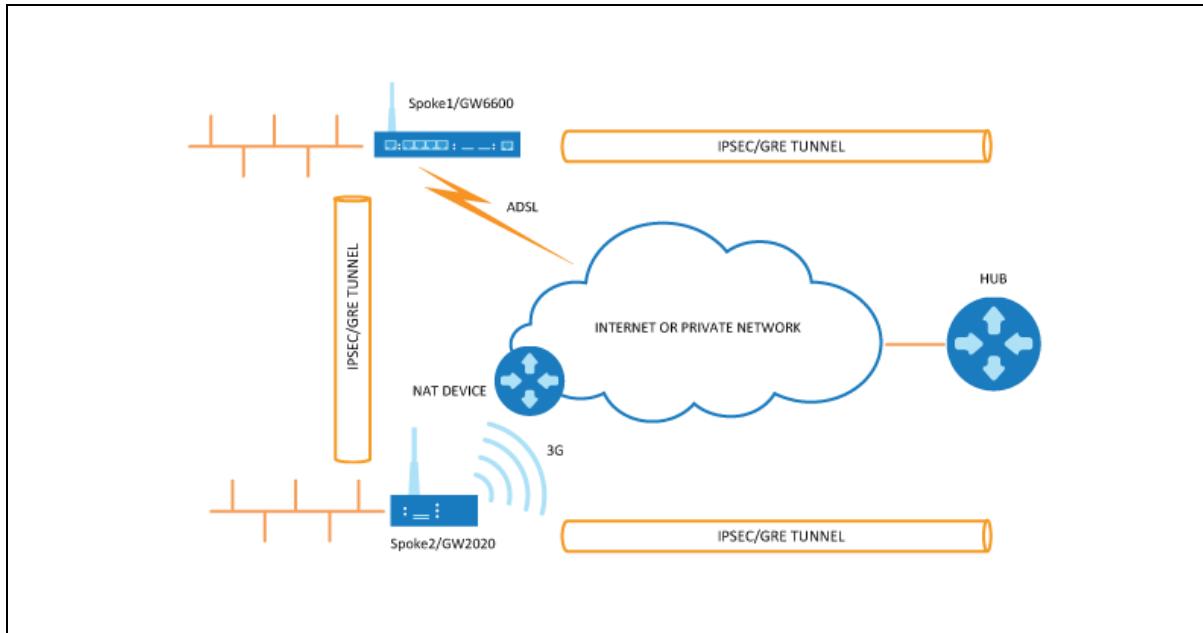


Figure 158: Network diagram for DMVPN spoke behind NAT

- 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.

32.4 Configuration packages used

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

32.5 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.

32.5.1 DMVPN general settings

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

Figure 159: The DMVPN general section

Web Field/UCI/Package Option	Description				
Web: Enable DMVPN UCI: dmvpn.common.enabled Opt: enable	Enables DMVPN. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
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 107: Information table for DMVPN general settings

32.5.2 DMVPN hub settings

Figure 160: 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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
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 108: Information table for DMVPN hub settings

32.5.3 Configuring an IPSec template for DMVPN using the web interface

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

32.6 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 161: 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 162: 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

```

Interface	Description	
Type	incomplete	Resolution request sent.
	negative	Negative cached.
	cached	Received/relayed resolution reply.
	shortcut_route	Received/relayed resolution for route.
	dynamic	NHC resolution.
	dynamic_nhs	Dynamic NHS from dns-map.
	static	Static mapping from config file.
	dynamic_map	Static dns-map from config file.
	local_route	Non-local destination, with local route.
	local_addr	Local destination (IP or off-NBMA subnet).
Protocol Address	Tunnel IP address	
NBMA-Address	Pre-NAT IP address if NBMA-NAT-OA-Address is present or real address if NAT is not present.	
NBMA-NAT-OA-Address	Post NAT IP address. This field is present when Address is translated in the network.	
Flags	up	Can send all packets (registration ok).
	unique	Peer is unique.
	used	Peer is kernel ARP table.
	lower-up	openhrp script executed successfully.
Expires-In	Expiration time.	

Table 109: Information table for DMVPN status

You can check IPSec status using UCI commands.

```

root@GW-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
```

33 Open VPN

The following is a guide to configuring a basic client mode OpenVPN connection.

33.1 Client configuration

To configure a basic client mode, rrowse to the router's IP address and login.

Select **Service tab > OpenVPN**.

The screenshot shows the 'OpenVPN' configuration page. At the top, there is a header 'OpenVPN'. Below it, a section titled 'OpenVPN instances' displays a message: 'Below is a list of configured OpenVPN instances and their current state'. A table header row includes columns for 'Enabled', 'Started', 'Start/Stop', 'Port', and 'Protocol'. Below the table, a message says 'This section contains no values yet'. Underneath the table, there is a dropdown menu with 'VPN' selected, followed by a note 'Simple client configuration for a' and an 'Add' button. At the bottom right of the page are three buttons: 'Save & Apply', 'Save', and 'Reset'. The footer of the page indicates it was 'Powered by LuCI LIS-15.00.52r13 00E0C812269F image1 config2'.

Figure 163: The openVPN page

Enter a relevant name and select the instance from the drop down list. The options are:

- Client configuration for an Ethernet bridge VPN
- Client configuration for a routed multi-client VPN
- Simple client configuration for a routed point-to-point VPN
- Server configuration for an Ethernet bridge VPN
- Server configuration for a routed multi-client VPN
- Simple server configuration for a routed point-to-point VPN

This document outlines how to configure a 'simple client configuration for a routed point-to-point VPN'. Click **Add**. The instance Overview page appears.

Overview » Instance "client_tun_ptp"

Switch to advanced configuration »

verb: 3 Set output verbosity

port: 2009 TCP/UDP port # for both local and remote

tun_ipv6: Make tun device IPv6 capable

ifconfig: 10.8.0.86 10.8.0.85 Set tun/tap adapter parameters

nobind: Do not bind to local address and port

comp_lzo: Use fast LZO compression

client: Configure client mode

client_to_client: Allow client-to-client traffic

remote: 1.1.1.1 Remote host name or ip address

secret: /etc/openvpn/michaelb.txt Enable Static Key encryption mode (non-TLS)

Additional Field – Add

Figure 164: The overview -> instance page

For this scenario, a secret key is used, which is loaded into /etc/openvpn.

33.1.1 Load secret key

To load the secret key, in the top menu, browse to **System tab > Administration**. Scroll towards the bottom of the page and select **Choose File for OpenVPN Certificates and Keys**.

Certificates & Private Keys

Certificates and private keys used for various services could be uploaded here

IPsec Certificates and Keys

Choose File No file chosen

Upload a *.tar.gz file containing certificates and/or private keys. All the ipsec certs previously uploaded will be deleted when new ones uploaded. Archive structure should match this of /etc/ipsec.d folder. Every file should be in one of 8 subfolders according to its purpose:

- private (private keys) certs (entity certs)
- crls (revocation lists)
- cacerts (CA certs)
- ocspcerts (OCSP signer certs)
- aacerts (Authorization Authority certs)
- acerts (attribute certs)
- reqs (PKCS#10 cert requests)

More info

OpenVPN Certificates and Keys

Choose File No file chosen

Upload a *.tar.gz file containing certificates and/or private keys. All the openvpn certs previously uploaded will be deleted when new ones uploaded. OpenVPN requires no special folder structure, hence files will be installed into the openvpn folder as they are in archive

VA Certificates and Keys

Choose File No file chosen

Upload a *.tar.gz file containing certificates and/or private keys. All the va certs previously uploaded will be deleted when new ones uploaded. Archive structure should match this of /etc/certs folder which is similar to /etc/ipsec.d folder.

Figure 165: The certificates and private keys page

When the key has been uploaded, you can select it as a secret option in the OpenVPN configuration page.

33.1.2 Add routes to a VPN connection

To add routes to the VPN connection, select **Switch to advanced configuration**. Select the Networking tab, scroll to the bottom of the page.

Click the drop down button and select the route you require and then click **Add**.

The route parameter will be available as below.

« Switch to basic configuration

Configuration category: Service | Networking | VPN | Cryptography

Networking

port ⓘ TCP/UDP port # for both local and remote

float ⓘ Allow remote to change its IP or port

nobind ⓘ Do not bind to local address and port

dev ⓘ tun/tap device

tun_ipv6 ⓘ Make tun device IPv6 capable

ifconfig ⓘ Set tun/tap adapter parameters

ifconfig_noexec ⓘ Don't actually execute ifconfig

ifconfig_nowarn ⓘ Don't warn on ifconfig inconsistencies

route ⓘ Add route after establishing connection

route_noexec ⓘ Don't add routes automatically

mtu_test ⓘ Empirically measure MTU

comp_lzo ⓘ Use fast LZO compression

comp_noadapt ⓘ Don't use adaptive lzo compression

ping_timer_rem ⓘ Only process ping timeouts if routes exist

Figure 166: The networking page

Select **Save & Apply**.

34 Configuring multicasting using PIM and IGMP interfaces

34.1 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.

34.2 Configuration package used

Package	Sections
pimd	pimd interface

34.3 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 167: The global settings interface

34.3.1 Global settings

Web Field/UCI/Package Option	Description	
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 110: Information table for PIM global settings

34.3.2 Interfaces configuration

Interfaces Configuration				
Enabled	Interface	Enable IGMP	Enable SSM	
<input checked="" type="checkbox"/>	gre1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Delete</button>
<input checked="" type="checkbox"/>	wlan_ap	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<button>Delete</button>
<button>Add</button>				

Figure 168: The interfaces configuration section

Web Field/UCI/Package Option	Description				
Web: Enabled UCI: pimd.interface[x].enabled Opt: enabled	Enables multicast management of the given interface by the PIM application.				
	<table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Interface UCI: pimd.interface[x].interface Opt: interface	Selects the interface to apply PIM settings to.				
Web: Enable IGMP UCI: pimd.interface[x].igmp Opt: igmp	<p>Enable IGMP on given interface.</p> <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table> <p>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.</p>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Enable SSM UCI: pimd.interface[x].ssm Opt: ssm	<p>Enable SSM on given interface.</p> <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

Table 111: Information table for interface settings

To save your configuration updates, click **Save & Apply**.

34.4 Configuring PIM and IGMP using UCI

You can configure PIM and IGMP through CLI using UCI.

The configuration file is stored on **/etc/config/pimd**

To view the configuration file, enter:

```
uci export pimd
root@GW_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@GW_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.

35 QoS: VLAN 802.1Q PCP tagging

35.1 Configuring VLAN PCP tagging

SATEL routers have the capability to respect and set PCP priority values inside 802.1Q VLAN tagged frames. The following partial export of network configuration shows how to configure VLAN priorities for specific interfaces (VLANs).

```
root@GW_router:~# uci export network package network

config va_switch

    option eth0 'A E'
    option eth1 'B F'
    option eth2 'C G'
    option eth3 'D'
    option eth4 'H'

config interface 'VLAN_1'

    option type 'bridge'
    option proto 'static'
    option ipaddr '10.1.28.99'
    option netmask '255.255.0.0'
    option ifname 'eth0 eth4'

config interface 'VLAN_2'

    option type 'bridge'
    option proto 'static'
    option ipaddr '192.168.2.1'
    option netmask '255.255.255.0'
    option ifname 'eth1 eth4.2'
    option vlan_qos_map_ingress '1:1'
    option vlan_qos_map_egress '0:1'

config interface 'VLAN_3'

    option ifname 'eth2 eth4.3'
    option type 'bridge'
    option proto 'static'
    option ipaddr '192.168.3.1'
    option netmask '255.255.255.0'
```

```

        option vlan_qos_map_ingress '3:3'
        option vlan_qos_map_egress '0:3'

config interface 'VLAN_4'
    option ifname 'eth3 eth4.4'
    option type 'bridge'
    option proto 'static'
    option ipaddr '192.168.3.1'
    option netmask '255.255.255.0'
    option vlan_qos_map_ingress '5:5'
    option vlan_qos_map_egress '0:5'

```

UCI/Package Option	Description
UCI: network.<if name>.vlan_qos_map_ingress Opt: list vlan_qos_map_ingress	VLAN priority code point to socket buffer mapping. Example: network.<if name>.vlan_qos_map_ingress =1:1
UCI: network.<if name>.vlan_qos_map_egress Opt: list vlan_qos_map_egress	Socket buffer to VLAN priority code point mapping. Example: network.<if name>.vlan_qos_map_egress =0:1

The above sample configuration specifies that any frames on VLAN2, VLAN3 and VLAN4 will be processed or have their PCP value adjusted according to QoS values set.

VLAN1

- VLAN1 is an untagged VLAN so there are no 802.1Q tags on the frames.

VLAN2

- Any frames received on VLAN2 destined to VLAN2 with PCP priority of 1 will be forwarded without altering the priority; it will be still set to 1.
- Any frames received on VLAN2 destined to VLAN2 with a PCP priority set to 0 will have a priority of 1 set as they leave the router on VLAN2.

VLAN3

- Any frames received on VLAN3 destined to VLAN3 with a PCP priority of 3 will be forwarded without altering the priority; it will be still set to 3.
- Any frames received on VLAN3 destined to VLAN2 with PCP priority set to 0 will have a priority of 3 set as they leave the router on VLAN3.

VLAN4

- Any frames received on VLAN4 destined to VLAN2 with PCP priority of 5 will be forwarded without altering the priority; it will be still set to 5.
- Any frames received on VLAN4 destined to VLAN2 with PCP priority set to 0 will have a priority of 5 set as they leave the router on VLAN4.

Four queues are supported and are structured as follows:

- Queue 1: PCP values 0 and 1 - Default
- Queue 2: PCP values 2 and 3 - Normal
- Queue 3: PCP values 4 and 5 - High
- Queue 4: PCP values 6 and 7 - Express

Value 7 is the highest priority and 0 is the lowest. These queues prioritise 802.1Q tagged frames as they are received on the port, these are hardware defined.

When 802.1Q frames are received on the port they are processed according to the above queues on arrival (even if not defined in the configuration). Then if value 'vlan_qos_map_ingress' is configured you can modify the PCP priority for egress if the frame was to be forwarded on another tagged interface.

When frames are received on an untagged VLAN interface configured with 'vlan_qos_map_egress' and are destined to tagged interface, 802.1Q tag will be created with a default priority of 0 and then the priority will be set according to the PCP value specified as the frames leave port.

36 QoS: type of service

SATEL routers are capable of implementing quality of service configurations on a per interface basis, which allows traffic prioritisation based on type of service criteria parameters.

36.1 QoS configuration overview

A minimal QoS configuration usually consists of:

- One interface section
- Some rules allocating packets to at least two buckets
- Configuration of the buckets

36.2 Configuration packages used

Package	Sections
qos	interface
	classgroup
	class
	classify

36.3 Configuring QoS using the web interface

Browse to the router's IP address and login.

Select **Network tab -> QoS**. The QoS page appears. From this page you can configure interfaces that QoS is applied to as well as classification rules.

The screenshot shows the 'Quality of Service' configuration page. It includes sections for 'Interfaces' and 'Classification Rules', both of which are currently empty. At the bottom of the page are 'Save & Apply', 'Save', and 'Reset' buttons.

Figure 169: The quality of service page

To configure an interface, enter a relevant interface name and click **Add**. The Quality of Service page for that interface appears.

Quality of Service

With QoS you can prioritize network traffic selected by addresses, ports or services.

Interfaces

WAN

Enable

Classification group: default

Calculate overhead

Half-duplex

Download speed (kbit/s): 8000

Upload speed (kbit/s): 1000

Add

Figure 170: The quality of service page for WAN interface

The following parameters can be configured for the interface you have chosen. The name of the interfaces should match with the logical name given to the interface in the network configuration.

Web Field/UCI/Package Option	Description				
Web: Enabled UCI: qos.[interface].enabled Opt: enabled	Enables or disables QoS interface. <table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
1	Enabled.				
0	Disabled.				
Web: Classification group UCI: qos. [interface].classgroup Opt: classgroup	Creates a mapping before previously created classgroup and interface to which it should be assigned to.				
Web: Calculate overhead UCI: qos. [interface].overhead Opt: overhead	Decreases upload and download ratio to prevent link saturation.				
Web: Half-duplex UCI: qos [interface].halfduplex Opt: halfduplex	Enables or disables half-duplex operation. <table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
1	Enabled.				
0	Disabled.				
Web: Download speed UCI: qos.[interface].download Opt: download	Download speed limit in kbits/sec.				
Web: Upload speed UCI: qos.[interface].upload=2000 Opt:upload	Upload speed limit in kbits/sec.				

Table 112: Information table for QoS page

To add classification rules, click **Add**. The Classification Rules section appears.

Configure each classification rule with the following parameters.

Figure 171: Parameters for classification rules

Web Field/UCI/Package Option	Description								
Web: Target UCI: Opt:	Creates and configures selected target bucket. <table border="1"><tr><td>Normal</td><td></td></tr><tr><td>Priority</td><td></td></tr><tr><td>Low</td><td></td></tr><tr><td>Express</td><td></td></tr></table>	Normal		Priority		Low		Express	
Normal									
Priority									
Low									
Express									
Web: Source host UCI: Opt:	Source host.								
Web: Destination host UCI: Opt:	Destination host.								
Web: Service UCI: Opt:	Selectable service.								
Web: Protocol UCI: Opt:	Protocol to classify.								
Web: Ports UCI: Opt:	Upload speed kbits/sec.								
Web: Number of bytes UCI: Opt:	Number of bytes for bucket.								

Table 113: Information table for classification rules

36.4 Configuring QoS using UCI

You can also configure QoS using UCI. The configuration file is stored on:

/etc/config/qos

36.4.1 Interface

Defines the interface on which configured QoS settings will take place.

Each interface can have its own buffer. The interface section declares global characteristics of the connection on which the specified interface is communicating. The following options are defined within this section:

```
config interface 'ADSL'
    option classgroup 'Default'
    option enabled '1'
    option overhead '1'
    option halfduplex '0'
    option download '900'
    option upload '245'
```

Web Field/UCI/Package Option	Description				
Web: Enabled UCI: qos.[interface].enabled Opt: enabled	Enables or disables QoS interface. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
1	Enabled.				
0	Disabled.				
Web: Classification group UCI: qos. [interface].classgroup Opt: classgroup	Creates a mapping before previously created classgroup and interface to which it should be assigned to.				
Web: Calculate overhead UCI: qos. [interface].overhead Opt: overhead	Decrease upload and download ratio to prevent link saturation.				
Web: Half-duplex UCI: qos [interface].halfduplex Opt: halfduplex	Enables or disables half-duplex operation. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
1	Enabled.				
0	Disabled.				
Web: Download speed UCI: qos.[interface].download Opt: download	Download speed limit in kbits/sec.				
Web: Upload speed UCI: qos.[interface].upload=2000 Opt:upload	Upload speed limit in kbits/sec.				

36.4.2 Classgroup

As there is more than one interface you can have more than one classgroup.

```
config classgroup 'Default'
    option classes 'Express Normal'
    option default 'Normal'
```

UCI/Package Option	Description
UCI: qos.Default=classgroup Opt: Default	Specifies name of classgroup.
UCI: qos.Default.classes=Express Normal Opt: classes	Specifies the list of names of classes which should be part of classgroup.
qos.Default.default=Normal Opt: default	Defines which class is considered default.

36.4.3 Classes

Each bucket has its own configuration.

```
config class 'Normal'
    option packetsize '1500'
    option avgrate '30'
    option priority '5'

config class 'Express'
    option packetsize '1000'
    option maxsize '800'
    option avgrate '50'
    option priority '10'
    option limitrate '10'
```

UCI/Package Option	Description
UCI: qos.Normal=class Opt: Normal	Specifies class name.
UCI: qos.Normal.packetsize=1500 Opt: packetsize	Specifies packet size for the class in bytes.
UCI: qos.Normal.avgrate=30 Opt: avgrate	Average rate for this class, value in % of bandwidth in %.
UCI: qos.Normal.priority=5 Opt: priority	Specifies priority for the class in %.
UCI: qos.Express=class Opt: Express	Specifies class name.
UCI: qos.Express.packetsize=1000 Opt: packetsize	Specifies packet size for the class in bytes.
UCI: qos.Express.maxsize=800 Opt: maxsize	Specify max packet size in bytes.
UCI: qos.Express.avgrate=50 Opt: avgrate	Average rate for this class, value in % of bandwidth in %.
UCI: qos.Express.priority=10 Opt: priority	Specifies priority for the class in %.
UCI: qos.Express.limitrate=10 Opt: limitrate	Defines to how many % of the available bandwidth this class is capped to.

36.4.4 Classify

Classifiers match the traffic for desired class.

```
config classify
    option target 'Express'
    option proto 'udp'
```

UCI/Package Option	Description
UCI: qos.@classify[0]=classify Opt: classify	Part of classify rule.
UCI: qos.@classify[0].target=Express Opt: target	Specifies target class.
UCI: qos.@classify[0].proto=udp Opt: proto	Specifies protocol.

36.5 Example QoS configurations

```

config interface 'ADSL'
    option classgroup 'Default'
    option enabled '1'
    option overhead '1'
    option download '900'
    option upload '245'

config classgroup 'Default'
    option classes 'Express Normal'
    option default 'Normal'

config class 'Normal'
    option packetsize '1500'
    option avgrate '30'
    option priority '5'

config class 'Express'
    option packetsize '1000'
    option maxsize '800'
    option avgrate '50'
    option priority '10'
    option limitrate '10'

config classify
    option target 'Express'
    option proto 'udp'
```

37 Management configuration settings

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

37.1 Activator

Activator is a SATEL 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.

- Autoload activation of firmware and configuration files on router boot up:
 - Autoload is generally used for router installation. In this scenario the router will initiate the request for firmware and configuration files when it boots up. The router is 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 initiates the process. This process, known as Active Updates, 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.

37.2 Monitor

Monitor is a SATEL 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.

37.3 Configuration packages used

Package	Sections
autoload	main
httpclient	default
management_users	user

37.4 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.

37.5 Autoload packages

Package	Sections
autoload	main

37.5.1 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.

The screenshot shows the 'Autoload' configuration page. At the top, there are tabs for Status, System, Services, Network, and Logout. Below the tabs, the title 'Autoload' is displayed with the subtitle 'Configuration of the VA Autoload Service'. Under 'Basic Settings', there are fields for 'Enabled' (checkbox), 'Start Timer' (10), 'Retry Timer' (30), 'Boot Using Config' (altconfig), and 'Boot Using Image' (altimage). A 'Delete' button is located to the right of these fields. The 'Entries' section contains a table with three rows, each representing an autoload entry. The columns are 'Configured', 'Segment Name', and 'Remote Filename'. The first row has a note: 'Download destination' and 'Use \$\$ for the serial number.' The second row has a checked checkbox, segment name 'altconfig', and remote filename '\$\$.ini'. The third row has a checked checkbox, segment name 'altimage', and remote filename '\$\$.img'. Both rows have a 'Delete' button to their right. Below the table is an 'Add' button. At the bottom right are 'Save & Apply', 'Save', and 'Reset' buttons.

Figure 172: 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. <table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
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. <table border="1"> <tr> <td>10</td><td></td></tr> <tr> <td>Range</td><td>0-300 secs</td></tr> </table>	10		Range	0-300 secs
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. <table border="1"> <tr> <td>30</td><td></td></tr> <tr> <td>Range</td><td>0-300 secs</td></tr> </table>	30		Range	0-300 secs
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. <table border="1"> <tr> <td>5</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	5		Range	
5					
Range					
Web: N/A UCI: autoload.main.BackoffTimer Opt: Backofftimer	Defines how many minutes to back off for if a download and all retries fail. After the backoff period, the entire autoload sequence will start again. <table border="1"> <tr> <td>15</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	15		Range	
15					
Range					

Web: Boot Using Config UCI: autoload.main.BootUsingConfig Opt: BootUsingConfig	Specifies which configuration to boot up with after the activation sequence. <table border="1"> <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>	Altconfig	Alternative configuration	Config1	Configuration 1	Config2	Configuration 2	Factconf	Factory configuration
Altconfig	Alternative configuration								
Config1	Configuration 1								
Config2	Configuration 2								
Factconf	Factory configuration								
Web: Boot Using Image UCI: autoload.main.BootUsingImage Opt: BootUsingImage	Specifies which image to boot up with after the activation sequence completes successfully. <table border="1"> <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>	Altimage	Alternative image	Image 1	image 1	Image 2	image 2		
Altimage	Alternative image								
Image 1	image 1								
Image 2	image 2								
Entries									
Web: Configured UCI: autoload.@entry[x].Configured Opt: Configured	Enables the autoload sequence to process this entry. <table border="1"> <tr><td>1</td><td>Enabled.</td></tr> <tr><td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.				
1	Enabled.								
0	Disabled.								
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 border="1"> <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> <p>Note: \$.vas should always be requested last.</p>	\$.vas	Notifies activator sequence is complete.	\$.ini	Request configuration	\$.img	Request firmware		
\$.vas	Notifies activator sequence is complete.								
\$.ini	Request configuration								
\$.img	Request firmware								

Table 114: Information table for autoload

37.6 Autoload using UCI

```

root@GW_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
Autoload using package options
root@GW_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"

```

37.7 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.

37.7.1 HTTP Client configuraton packages

Package	Sections
Httpclient	default

37.7.2 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 173: The HTTP client page

Web Field/UCI/Package Option	Description				
Basic settings					
Web: Enabled UCI: httpclient.default.enabled Opt: Enabled	Enables the HTTP client. <table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
1	Enabled.				
0	Disabled.				
Web: Server IP Address UCI: httpclient.default.Fileserver Opt: list Fileservers	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: list SecureFileServer	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). <table border="1"> <tr> <td>1</td><td>Enabled.</td></tr> <tr> <td>0</td><td>Disabled.</td></tr> </table>	1	Enabled.	0	Disabled.
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. <table border="1"> <tr> <td>/Activator/Sessionless/Httpserver.asp</td><td></td></tr> <tr> <td>Range</td><td></td></tr> </table>	/Activator/Sessionless/Httpserver.asp		Range	
/Activator/Sessionless/Httpserver.asp					
Range					

Web: Check Server Certificate UCI: httpclient.default.ValidateServerCertificate Enabled Opt: ValidateServerCertificateEnabled	Checks for the certificates presence and validity. <table border="1"><tr><td>1</td><td>Enabled.</td></tr><tr><td>0</td><td>Disabled.</td></tr></table>	1	Enabled.	0	Disabled.
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. <table border="1"><tr><td>1</td><td>Enabled.</td></tr><tr><td>0</td><td>Disabled.</td></tr></table>	1	Enabled.	0	Disabled.
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. <table border="1"><tr><td>PEM</td><td></td></tr><tr><td>DER</td><td></td></tr></table>	PEM		DER	
PEM					
DER					
Web: Certificate File Path UCI: httpclient.default.CertificateFile Opt: CertificateFile	Defines the directory/location of the certificate. <table border="1"><tr><td>/etc/httpclient.crt</td><td></td></tr><tr><td>Range</td><td></td></tr></table>	/etc/httpclient.crt		Range	
/etc/httpclient.crt					
Range					
Web: Certificate Key File Path UCI: httpclient.default.CertificateKey Opt: CertificateKey	Specifies the directory/location of the certificate key. <table border="1"><tr><td>/etc/httpclient.key</td><td></td></tr><tr><td>Range</td><td></td></tr></table>	/etc/httpclient.key		Range	
/etc/httpclient.key					
Range					
Web: N/A UCI: ValidateServerCertificateFieldEnabled Opt: ValidateServerCertificate	Defines the field in the server certificate that the client should check. <table border="1"><tr><td>1</td><td>Enabled.</td></tr><tr><td>0</td><td>Disabled.</td></tr></table>	1	Enabled.	0	Disabled.
1	Enabled.				
0	Disabled.				
Web: N/A UCI: httpclient.default.ActivatorChunkyDownloadPath Opt: ActivatorChunkyDownloadPath	Enables partial download activations and active updates. The default value is httpclient.default.ActivatorChunkyDownloadPath=/activator/partial/download The url (on activator) to which the client should send requests for chunky image download.				
Web: N/A UCI: httpclient.default.ChunkSize Opt: ChunkSize	Specifies the size of each packet payload <table border="1"><tr><td>100k</td><td>100K Bytes</td></tr><tr><td>1-infinite</td><td>Available values</td></tr></table>	100k	100K Bytes	1-infinite	Available values
100k	100K Bytes				
1-infinite	Available values				
Web: N/A UCI: httpclient.default.RateLimit Opt: RateLimit	Throttle activation/active updates traffic received by device to specified limit <table border="1"><tr><td>None</td><td>By default there is no limit</td></tr><tr><td>1-infinite</td><td>Available values in kbps</td></tr></table>	None	By default there is no limit	1-infinite	Available values in kbps
None	By default there is no limit				
1-infinite	Available values in kbps				
Web: N/A UCI: httpclient.default.CAFile Opt: CAFile	Defines path to the certificate authority file stored on the router				
Web: N/A UCI: httpclient.default.IgnoreServerCertificateStatus Opt: IgnoreServerCertificateStatus	Defines whether to skip the status check on the server certificate. <table border="1"><tr><td>1</td><td>Enabled.</td></tr><tr><td>0</td><td>Disabled.</td></tr></table>	1	Enabled.	0	Disabled.
1	Enabled.				
0	Disabled.				

Table 115: Information table for HTTP client

37.8 Httpclient: Activator configuration using UCI

```
root@GW_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
httpClient.default.ActivatorChunkyDownloadPath=/activator/partial/download
httpClient.default.ChunkSize=100k
httpClient.default.RateLimit=2
httpClient.default.CAFile='\''
httpClient.default.IgnoreServerCertificateStatus=0

```

37.9 Httpclient: Activator configuration using package options

```

root@GW_router:~# uci export httpclient
package httpclient

config core 'default'
    option Enabled 'yes'
    list FileServer '1.1.1.1:80'
    list FileServer '1.1.1.2:80'
    listSecureFileServer '1.1.1.1:443'
    list SecureFileServer '1.1.1.2:443'
    option ActivatorDownloadPath '/Activator/Sessionless/Httpserver.asp'
    option SecureDownload 'no'
    option PresentCertificateEnabled 'no'
    option ValidateServerCertificateEnabled 'no'
    option CertificateFile '/etc/httpclient.crt'
    option CertificateFormat 'PEM'
    option CertificateKey '/etc/httpclient.key'
    option ActivatorChunkyDownloadPath '/activator/partial/download'
    option ChunkSize '100k'
    option RateLimit '2'
    option CAFile '\''

```

```
option IgnoreServerCertificateStatus '0'
```

37.10 User management using UCI

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

37.10.1 User management packages

Package	Sections
management_users	Users

37.10.2 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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: n/a UCI: management_users.@user[x].papuser Opt: papuser	Specifies PAP access permissions for the PPP connection. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: n/a UCI: management_users.@user[x].srpuser Opt: srpuser	Specifies SRP access permissions for the PPP connection. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: n/a UCI: management_users.@user[x].smsuser Opt: smsuser	Specifies SMS access permissions for the user. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: n/a UCI: linuxuser Opt: linuxuser	Specifies linuxuser access permissions for the user. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
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 116: 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.

37.11 Configuring the management user password using UCI

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

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

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

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

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

37.12 Configuring management user password using package options

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

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

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

If you are changing the password using 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.

37.13 User management using UCI

```
root@GW_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
```

37.14 User management using package options

```
root@GW_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'
    option smsuser '0'
```

37.15 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.

38 Configuring Monitor

38.1 Introduction

SATEL monitoring system (Monitor) is a secure portal that provides:

- Centralised monitoring of devices
- Device status
- GPS location
- Syslog reporting
- Real time diagnostics
- Email notification
- Advanced statistics
- Dashboard graph reporting

You must configure each router in the network to send the required information to Monitor. This chapter explains how to configure the different information that can be sent to Monitor, including the required router configuration for:

- Reporting device status to Monitor
- Reporting GPS location to Monitor
- Reporting syslog to Monitor
- Configuration of interface statistics collection (ISAD)

For detailed information on operating Monitor, read the Monitor User Manual.

38.2 Reporting device status to Monitor

To allow Monitor to track the IP address and ongoing presence of a device, a keepalive heartbeat SNMP trap is sent from the router. The router is capable of sending SNMP in version 1, 2c and 3.

The SNMP keepalive heartbeat sends basic information on interface status but can also be configured to contain more detailed information such as GPS location.

The basic heartbeat configuration consists of two parts:

- enabling the heartbeat keepalive
- enabling the interface(s) to be monitored

38.2.1 Configuration package used

Package	Sections
monitor	keepalive
network	interface

38.2.2 Configuring keepalive heartbeat using the web interface

Select **Services -> Monitor**. The Monitor Keepalive & ISAD page appears.

The keepalive heartbeat is configured under the **Basic Settings** section.

A single instance keepalive can be configured to multiple monitor address using the same reference, heartbeat interval and other options. Or alternatively multiple keepalive instances can be configured with unique options.

Monitor Keepalive & ISAD

Configuration of the VA Monitor Keepalive Service and Interface Stats Upload.

Basic Settings

Basic settings should be checked according to your network.

KEEPALIVE1

Enabled	<input checked="" type="checkbox"/>
Dev Reference	router1
Monitor Address	1.1.1.1
Monitor Heartbeat Interval	1
SNMP Protocol Version	1

, or alternative

Figure 174: The Monitor & ISAD keepalive page

38.2.2.1 Basic settings

Web Field/UCI/Package Option	Description						
Web: Enabled UCI: monitor.@keepalive[0].enabled Opt: Enabled	Enables Monitor to send heartbeats to the router. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.		
0	Disabled.						
1	Enabled.						
Web: Dev Reference UCI: monitor.@keepalive[0].dev_reference Opt: dev_reference	Sets a unique identification for this device known to Monitor.						
Web: Monitor Address UCI: monitor.@keepalive[0].monitor_ip Opt: list monitor_ip	Defines the IP address of Monitor. It is possible to specify multiple addresses to which SNMP heartbeat traps will be sent. To configure via UCI use a space separator. Example: monitor.@keepalive[0].monitor_ip=1.1.1.1 2.2.2.2						
Web: Monitor Heartbeat Interval UCI: monitor.@keepalive[0].interval_min Opt: interval_min	Specifies the interval, in minutes, at which traps are sent. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>1</td> <td>Trap set every 1 minute</td> </tr> <tr> <td>Range</td> <td></td> </tr> </table>	1	Trap set every 1 minute	Range			
1	Trap set every 1 minute						
Range							
Web: SNMP Protocol Version UCI: monitor.@keepalive[0].snmp_version Opt: snmp_version	Specifies what SNMP version is sent to remote Manager. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>1</td> <td>snmp version 1</td> </tr> <tr> <td>2c</td> <td>SNMP version 2c</td> </tr> <tr> <td>3</td> <td>SNMP version 3</td> </tr> </table>	1	snmp version 1	2c	SNMP version 2c	3	SNMP version 3
1	snmp version 1						
2c	SNMP version 2c						
3	SNMP version 3						

Table 117: Information table for Monitor & ISAD basic configuration

The figure below shows options that are relevant only if you have selected SNMP version 3.

SNMP Protocol Version	3
User Name	
Authentication Protocol	SHA
Authentication Password	
Privacy Protocol	AES
Privacy Password	
SNMPv3 Context	
SNMPv3 Context Engine ID	
SNMPv3 Security Engine ID	

Figure 175: The Monitor & ISAD keepalive page for SNMP v3

Web Field/UCI/Package Option	Description						
Web: User Name UCI: monitor.@keepalive[0].snmp_uname Opt: snmp_uname	Specifies user name <table border="1"> <tr> <td>Blank</td> <td>Default value</td> </tr> <tr> <td colspan="2">String</td> </tr> </table>	Blank	Default value	String			
Blank	Default value						
String							
Web: Authentication Password UCI: monitor.@keepalive[0].snmp_auth_pass Opt: snmp_auth_pass	Specifies snmpv3 authentication password.						
Web: Authentication Protocol UCI: monitor.@keepalive[0].snmp_auth_proto Opt: snmp_auth_proto	Specifies snmpv3 authentication protocol. <table border="1"> <tr> <td>Blank</td> <td>Default value</td> </tr> <tr> <td>MD5</td> <td>MD5 as authentication protocol</td> </tr> <tr> <td>SHA</td> <td>SHA as authentication protocol</td> </tr> </table>	Blank	Default value	MD5	MD5 as authentication protocol	SHA	SHA as authentication protocol
Blank	Default value						
MD5	MD5 as authentication protocol						
SHA	SHA as authentication protocol						
Web: Privacy Protocol UCI: monitor.@keepalive[0].snmp_priv_proto Opt: snmp_priv_proto	Specifies snmpv3 privacy protocol <table border="1"> <tr> <td>Blank</td> <td>Default value</td> </tr> <tr> <td>AES</td> <td>AES as privacy protocol</td> </tr> <tr> <td>DES</td> <td>MD5 as privacy protocol</td> </tr> </table>	Blank	Default value	AES	AES as privacy protocol	DES	MD5 as privacy protocol
Blank	Default value						
AES	AES as privacy protocol						
DES	MD5 as privacy protocol						
Web: Privacy Password UCI: monitor.@keepalive[0].snmp_priv_pass Opt: snmp_priv_pass	Specifies snmpv3 privacy password.						
Web: SNMPv3 Context UCI: monitor.@keepalive[0].snmp_context Opt: snmp_context	Specifies snmpv3 context name.						
Web: SNMPv3 Context Engine ID UCI: monitor.@keepalive[0].snmp_context_eid Opt: snmp_context_eid	Specifies snmpv3 context engine ID.						
Web: SNMPv3 Security Engine ID UCI: monitor.@keepalive[0].snmp_sec_eid Opt: snmp_sec_eid	Specifies snmpv3 security engine ID.						

Table 118: Information table for SNMP v3 reporting device commands

38.2.3 Configuring keepalive heartbeat using command line

Keepalive is configured under the monitor package.

By default, all keepalive instances are named ‘keepalive’, it is identified by @keepalive then the keepalive position in the package as a number. For example, for the first keepalive in the package using UCI:

```
monitor.@keepalive[0]=keepalive
monitor.@ keepalive[0].enabled=1
```

Or using package options:

```
config keepalive
    option enabled '1'
```

However, to better identify, it is recommended to give the keepalive instance a name. For example, to create a keepalive instance named keepalivev1.

To define a named keepalive instance using UCI, enter:

```
monitor.keepalivev1=keepalive
monitor.keepalivev1.enable=1
```

To define a named keepalive instance using package options, enter:

```
config keepalive 'keepalivev1'
    option enabled '1'
```

38.2.4 Keepalive using UCI

```
root@GW_router:~# uci show monitor
monitor.keepalivev1=keepalive
monitor.keepalivev1.enabled=1
monitor.keepalivev1.interval_min=1
monitor.keepalivev1.dev_reference=router1
monitor.keepalivev1.monitor_ip=10.1.83.36
monitor.keepalivev1.snmp_version=1
monitor.keepalivev2=keepalive
monitor.keepalivev2.enable=1
monitor.keepalivev2.interval_min=1
monitor.keepalivev2.monitor_ip=172.16.250.100
monitor.keepalivev2.dev_reference=TEST
monitor.keepalivev2.snmp_version=2c
monitor.keepalivev3=keepalive
monitor.keepalivev3.enable=1
monitor.keepalivev3.interval_min=1
monitor.keepalivev3.monitor_ip=172.16.250.101
monitor.keepalivev3.dev_reference=TEST
monitor.keepalivev3.snmp_version=3
monitor.keepalivev3.snmp_uname=TEST
monitor.keepalivev3.snmp_auth_pass=vasecret
monitor.keepalivev3.snmp_auth_proto=MD5
monitor.keepalivev3.snmp_priv_pass=vasecret
monitor.keepalivev3.snmp_priv_proto=DES
```

38.2.5 Keepalive using package options

```
root@GW_router:~# uci export monitor
package 'monitor'

config keepalive 'keepalivev1'
    option enabled '1'
    option interval_min '1'
    option dev_reference 'router1'
    option enabled 'yes'
    list monitor_ip '10.1.83.36'

config keepalive 'keepalivev2'
    option enable '1'
    option interval_min '1'
    list monitor_ip '172.16.250.100'
    option dev_reference 'TEST'
    option snmp_version '2c'

config keepalive 'keepalivev3'
    option enable '1'
    option interval_min '1'
    list monitor_ip '172.16.250.101'
    option dev_reference 'TEST'
    option snmp_version '3'
    option snmp_uname 'TEST'
    option snmp_auth_pass 'vasecret'
    option snmp_auth_proto 'MD5'
    option snmp_priv_pass 'vasecret'
    option snmp_priv_proto 'DES'
```

38.2.6 Enabling interface status in keepalive heartbeat via web interface

The keepalive heartbeat can send information on multiple interfaces. In order to send an interface status to Monitor, select **Network -> Interfaces**, then under the required interface select **Edit**. Under **Advanced Settings** enable the Monitor interface state option.

Figure 176: The interface common configuration page

Web Field/UCI/Package Option	Description				
Web: Monitor interface state UCI: network.@interface[0].monitored Opt: monitored	Enables interface status to be sent in the heartbeat trap to Monitor. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 50px; height: 20px;"></td><td>Disabled.</td></tr> <tr> <td style="width: 50px; height: 20px;"></td><td>Enabled.</td></tr> </table>		Disabled.		Enabled.
	Disabled.				
	Enabled.				

Table 119: Information table for enabling interface status command

38.2.7 Enabling interface status using command line

Interface status is configured under the network package.

38.2.7.1 Enable interface status using UCI

```
root@GW_router:~# uci show network
network.@interface[0]=interface
.....
network.@interface[0].monitored=1
.....
```

38.2.7.2 Enable interface status using package option

```
root@GW_router:~# uci export network
package network
config interface 'WAN'
.....
option monitored '1'
.....
```

38.3 Reporting GPS location to Monitor

To allow Monitor to display a router GPS location, the GPS coordinates can be configured to be sent in the heartbeat keepalive from the router.

GPS location is only available in supported hardware models.

Ensure monitor keepalive heartbeat is correctly configured as in section 30.2 above.

38.3.1 Configuration package used

Package	Sections
gpsd	gpsd

38.3.2 Configuring GPS location via the web interface

Select **Services -> GPS**. The GPS configuration page appears.

The web interface configures a gpsd section named core.

Main Settings

Enable GPS

Figure 177: The GPS configuration page

Web Field/UCI/Package Option	Description				
Web: Enable GPS UCI: monitor.core.enabled Opt: enabled	Enables GPS coordinates to be sent in the heartbeat keepalive to Monitor. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

Table 120: Information table for reporting GPS commands

38.3.3 Configuring GPS using command line

GPS location is configured under the gpsd package.

38.3.3.1 GPS using UCI

```
root@GW_router:~# uci show gpsd
gpsd.core=gpsd
gpsd.core.enabled=1
```

38.3.3.2 GPS using package options

```
root@GW_router:~# uci export gpsd
package gpsd
config gpsd 'core'
    option enabled '1'
```

38.3.4 GPS diagnostics

To view information on GPS coordinates via the web interface, select **Status -> GPS Information**.

The screenshot shows the 'System' configuration page. It includes tabs for 'General Settings', 'Logging' (which is selected), and 'Language and Style'. Under 'Logging', there are fields for 'System log buffer size' (set to 400 kiB), 'External system log server' (set to 11.1.1), and 'External system log server port' (set to 514).

Figure 178: The GPS status page

To view GPS coordinates via command line, enter `gpspeek`:

```
root@GW_router:~# gpspeek
Fix: 3D,1495467700,53.342529,-
6.241236,27.700000,202.600000,0.000000,0.000000
```

38.4 Reporting syslog to Monitor

38.4.1 Configuration package used

Package	Sections
system	main

38.4.2 Configuring syslog to Monitor via the web interface

Monitor can display syslog events sent from the router. To configure the router to send syslog events, select **System -> System -> Logging** and set **External system log server** to the Monitor IP. You can also configure the syslog server port if required.

All syslog events are sent to the syslog server.

System
Here you can configure the basic aspects of your device like its hostname or the timezone.

System Properties

General Settings Logging Language and Style

System log buffer size: 400 kiB
External system log server: 34.250.103.126
External system log server port: 514

Figure 179: The system properties page

Web Field/UCI/Package Option	Description
Web: External system log server UCI: system.main.log_ip Opt: log_ip	Defines the external syslog server IP address.
Web: External system log server UCI: system.main.log_port Opt: log_port	Defines the external syslog server destination port number for syslog messages. 514 Range

Table 121: Information table for syslog properties commands

38.4.3 Configuring syslog events to Monitor using command line

Syslog is configured under the system package.

38.4.3.1 Syslog events to Monitor using UCI

```
root@GW_router:~# uci show system
system.main=system
.....
system.main.log_ip=1.1.1.1
system.main.log_port=514
.....
```

38.4.3.2 Syslog events to Monitor using package options

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

config system 'main'
.....
option log_ip '1.1.1.1'
option log_port '514'
.....
```

38.5 Configuring ISAD

ISAD is a system for collecting interface stats to be displayed on Monitor.

The following section explains how to configure interface statistics collection (iSAD).

Statistical data is collected in bins with each bin containing interface transmit and receive packets/bytes/errors for a period. Signal strength and also temperature parameters are also stored in the bins. Bins are uploaded to Monitor periodically.

Note: Ensure monitor keepalive heartbeat and interface status is correctly configured as in section 30.2 above. Interfaces should have `option monitored` enabled as part of the collection.

ISAD replaces the deprecated SLA feature.

38.5.1 Configuration package used

Package	Sections
monitor	interface_stats

38.5.2 Configuring ISAD using the web interface

Select **Services -> Monitor**. The Monitor Keepalive & ISAD page appears. ISAD is configured under the **Interface Stats** section.

The screenshot shows a configuration page titled "Interface Stats". It contains three input fields: "Enabled" (checkbox), "Bin Period" (text box containing "1h"), and "Maximum Number of Bins" (text box containing "24").

Figure 180: The Monitor Keepalive & ISAD Interface Stats page

Web Field/UCI/Package Option	Description				
Web: Enabled UCI: monitor.stats.enabled=1 Opt: enabled	Enables ISAD. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Bin Period UCI: monitor.stats.bin_period Opt: time	Specifies how long to collect data for one bin. Specifies the interval, in minutes, at which traps are sent. <table border="1"> <tr> <td>1h</td><td>Bin collected for 1 hour</td></tr> <tr> <td>Range</td><td></td></tr> </table>	1h	Bin collected for 1 hour	Range	
1h	Bin collected for 1 hour				
Range					
Web: Maximum Number of Bins UCI: monitor.stats.bin_cache_size Opt: bin_cache_size	Specifies the maximum number of bins to store. <table border="1"> <tr> <td>Empty</td><td>24</td></tr> <tr> <td>Range</td><td></td></tr> </table>	Empty	24	Range	
Empty	24				
Range					

Table 122: Information table for ISAD Monitor Keepalive & ISAD Interface Stats section

38.5.3 Configuring ISAD using the command line

ISAD is configured under the Monitor package.

38.5.3.1 ISAD using UCI

```
root@GW_router:~# uci show monitor
monitor.keepalivev1=keepalive
monitor.keepalivevlenabled=1
monitor.keepalivev1.interval_min=1
monitor.keepalivev1.dev_reference=router1
monitor.keepalivev1.monitor_ip=10.1.83.36
monitor.keepalivev1.snmp_version=1
monitor.stats=interface_stats
monitor.stats.enabled=1
monitor.stats.bin_period=1h
monitor.stats.bin_cache_size=24
```

38.5.3.2 ISAD using package options

```
root@GW_router:~# uci export monitor
package monitor

config keepalive 'keepalivev1'
    option interval_min '1'
    option enabled '1'
    list monitor_ip '10.1.83.36'
    option dev_reference 'router1'

config interface_stats 'stats'
    option enabled '1'
    option bin_period '1h'
    option bin_cache_size '24'
```

38.5.4 ISAD diagnostics

38.5.4.1 Checking process

To check to see if ISAD is running, enter pgrep -fl isad:

```
root@GW_router:~# pgrep -fl isad
5303 /usr/sbin/isad -b 60 -s 10 -c 200 -u /var/state /var/const_state
```

38.5.4.2 Checking bin statistics

To check if stats are being collected, enter cat /var/state/monitor:

```
root@GW_router:~# cat /var/state/monitor
monitor.bin_0=isad
monitor.bin_0.end_ts=85020
monitor.bin_0.start_ts=84960
monitor.bin_1=isad
monitor.bin_1.end_ts=85080
monitor.bin_1.start_ts=85020
monitor.bin_2=isad
monitor.bin_2.end_ts=85140
monitor.bin_2.start_ts=85080
```

38.5.5 ISAD operation

The bin statistics stored on the router must be periodically pushed statistics to Monitor.

This is normally done centrally when statistics are enabled on Monitor. Monitor contacts each router and auto-generates a script that will automatically schedule the upload of the bin statistics.

However, if Monitor cannot access the router WAN IP, you must do this manually on each router using a UDS script. An example is shown below where the bins are uploaded every hour to a Monitor server IP 89.101.154.154 using TFTP.

```
package uds

config script 'isb_upload_scr'
    option enabled '1'
    option exec_type 'periodic'
    option period '1h'
    list text '/usr/sbin/isb_upload.lua 89.101.154.154:69'
```

39 Configuring SNMP

SNMP (Simple Network Management Protocol) is an internet-standard protocol for managing devices on IP networks. SNMP exposes management data in the form of a hierarchy of variables in a MIB (Management Information Base). These variables can be queried individually, or in groups using their OIDs (Object Identifiers) defined in MIBs. In addition, information from the router can be pushed to a network management station in the form of SNMP traps.

39.1 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.
usm_user	Define a user for SNMPv3 USM
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

39.2 Configuring SMNP using the web interface

In the top menu, select **Services -> SNMP**. The SNMP Service page appears.

The screenshot shows the 'SNMP Service' configuration page. At the top, there is a navigation bar with links for Status, System, Services, Network, and Logout. Below the navigation bar, the title 'SNMP Service' is displayed, followed by the subtitle 'Configuration of the SNMP service.' The page is divided into two main sections: 'System Settings' and 'Agent Settings'. Under 'System Settings', there are three input fields: 'System Location' (value: Desk_Joe), 'System Contact' (value: joe.brown@company.com), and 'System Name' (value: Test Router). Under 'Agent Settings', there are four input fields: 'Agent Address' (value: UDP 161), 'Enable Authentication' (checkbox checked), 'Traps' (checkbox checked), and 'Enable Link State Notification' (checkbox checked). There is also a note below the last checkbox stating 'Generate Trap/Info when interface go up or down'.

Figure 181: The SNMP service page

39.2.1 System and agent settings

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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table> <p>Note: this is the SNMP poll authentication trap to be set when there is a community mismatch.</p>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

Table 123: Information table for system and agent settings

39.2.2 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.

A community string is a password that is applied to a device to restrict both read-only and read-write access to the SNMP data on the device. These community strings should be chosen carefully to ensure they are not trivial. They should also be changed at regular intervals and in accordance with network security policies.

The screenshot shows a web-based configuration interface for 'COM2SEC Settings'. At the top, there are navigation links: Status, System, Services, Network, and Logout. A blue bar at the top right indicates 'UNSAVED CHANGES'. Below the header, the title 'COM2SEC Settings' is displayed. A table lists two entries:

Security Name	Source	Community	
public	ro	default	public Delete
private	rw	localhost	private Delete

At the bottom left is an empty input field, and at the bottom right is a 'Add' button.

Figure 182: The COM2Sec settings section

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 124: Information table for Com2Sec settings

39.2.3 Group settings

Group settings assign community names and SNMP protocols to groups.

The screenshot shows a web-based configuration interface for 'Group Settings'. At the top, there are navigation links: Status, System, Services, Network, and Logout. Below the header, the title 'Group Settings' is displayed. A table lists five entries:

Group	Version	Security Name	
public_v1	v1	ro	Delete
public_v2c	v2c	ro	Delete
public_usm	usm	ro	Delete
private_v1	v1	rw	Delete
private_v2c	v2c	rw	Delete

Figure 183: The group settings section

Web Field/UCI/Package Option	Description								
Web: Group UCI: snmpd.group[x].group Opt: group	Specifies an arbitrary group name.								
Web: Version UCI: snmpd.group[x].version Opt: version	Specifies the SNMP version number being used in the request: v1, v2c and usm (User-based Security Module) are supported. <table border="1" style="margin-left: auto; margin-right: auto;"> <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>	v1	SNMP v1	v2v	SNMP v2	usm	SNMP v3	any	Any SNMP version
v1	SNMP v1								
v2v	SNMP v2								
usm	SNMP v3								
any	Any SNMP version								
Web: Security Name UCI: snmpd.group[x].secname Opt: secname	An already defined security name that is being included in this group.								

Table 125: Information table for group settings

39.2.4 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 184: 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.				
Web: Type UCI: snmpd.view[x].type Opt: type	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 border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>included</td> <td></td> </tr> <tr> <td>excluded</td> <td></td> </tr> </table>	included		excluded	
included					
excluded					
Web: OID UCI: snmpd.view[x].oid Opt: oid	OID to be included in or excluded from the view. Only numerical representation is supported. Example <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>1</td> <td>Everything</td> </tr> <tr> <td>1.3.6.1.2.1.2</td> <td>Interfaces table</td> </tr> </table>	1	Everything	1.3.6.1.2.1.2	Interfaces table
1	Everything				
1.3.6.1.2.1.2	Interfaces table				

Table 126: Information table for view settings

39.2.5 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.

Access Settings								
	group	context	version	level	prefix	read	write	notify
public_access	public	none	any	noauth	exact	all	none	none
private_access	private	none	any	noauth	exact	all	all	all
<input type="button" value="Delete"/>								
<input type="button" value="Add"/>								

Figure 185: The access settings section

Web Field/UCI/Package Option	Description								
Web: Group UCI: snmpd.access[x].group Opt: group	Specifies the group to which access is being granted.								
Web: Context UCI: snmpd.access[x].context Opt: context	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 border="1"> <tr> <td>none</td> <td></td> </tr> <tr> <td>all</td> <td></td> </tr> </table>	none		all					
none									
all									
Web: Version UCI: snmpd.access[x].version Opt: version	Specifies the SNMP version number being used in the request: any, v1, v2c and usm are supported. <table border="1"> <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>	v1	SNMP v1	v2v	SNMP v2	usm	SNMP v3	any	Any SNMP version
v1	SNMP v1								
v2v	SNMP v2								
usm	SNMP v3								
any	Any SNMP version								
Web: Level UCI: snmpd.access[x].level Opt: level	Specifies the security level. For SNMP v1 and SNMP v2c level must be noauth . <table border="1"> <tr> <td>noauth</td> <td></td> </tr> <tr> <td>auth</td> <td></td> </tr> <tr> <td>priv</td> <td></td> </tr> </table>	noauth		auth		priv			
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. <table border="1"> <tr> <td>exact</td> <td></td> </tr> <tr> <td>any</td> <td></td> </tr> <tr> <td>all</td> <td></td> </tr> </table>	exact		any		all			
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 127: Information table for access settings

39.2.6 Trap receiver

Trap receiver settings define a notification receiver that should be sent SNMPv1 TRAPS and SNMPv2c TRAP2.

Trap Receiver			
Host	Port	Version	Community
192.168.100.254		v1	public
<input type="button" value="Add"/>			

Figure 186: 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 an FQDN.				
Web: Port UCI: snmpd.trapreceiver[x].port Opt: port	UDP port to be used for sending traps. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>Range</td> <td></td> </tr> <tr> <td>162</td> <td></td> </tr> </table>	Range		162	
Range					
162					
Web: Version UCI: snmpd.trapreceiver[x].version Opt: version	SNMP version. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>v1</td> <td></td> </tr> <tr> <td>V2</td> <td></td> </tr> </table>	v1		V2	
v1					
V2					
Web: Community UCI: snmpd.trapreceiver[x].community Opt: community	Community to use in trap messages for this host.				

Table 128: Information table for trap receiver settings

39.2.7 Inform receiver

Inform receiver settings define a notification receiver that should be sent SNMPv2c INFORM notifications.

Figure 187: 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 an FQDN.				
Web: Port UCI: snmpd.informreceiver[x].port Opt: port	UDP port to be used for sending traps. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>Range</td> <td></td> </tr> <tr> <td>162</td> <td></td> </tr> </table>	Range		162	
Range					
162					
Web: Community UCI: snmpd.informreceiver[x].community Opt: community	Community to use in inform messages for this host.				

Table 129: Information table for trap receiver settings

39.3 Configuring SNMP using command line

The configuration files are stored on **/etc/config/snmpd**.

39.3.1 System settings using UCI

```
root@GW_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

```

39.3.2 System settings using package options

```

root@GW_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'

```

39.3.3 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.

39.3.3.1 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
```

39.3.3.2 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'
```

39.3.4 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.

39.3.4.1 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
```

39.3.4.2 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'
```

39.3.5 View settings

The following example defines two views, one for the entire system and another for only mib2.

39.3.5.1 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
```

39.3.5.2 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'
```

39.3.6 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. Although it is possible to write some settings using SNMP write permission, it is not recommended, as any changes to the configuration made through an snmpset command may conflict with the UCI configuration. In this instance the changes will be overwritten by other processes and will not persist after a reboot.

39.3.6.1 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'

```

39.3.7 SNMP traps settings

39.3.7.1 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

```

39.3.7.2 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'

```

39.4 Configuring SNMP interface alias with static SNMP index

A Linux interface index changes dynamically. This is not ideal for SNMP managers that require static interface indexes to be defined.

The network package interface section allows defining a static SNMP interface alias index for this interface.

An alias entry is created in the SNMP ifEntry table at index (**snmp_alias_ifindex + 1000**). This entry is a shadow of the real underlying Linux interface corresponding to the UCI definition. You may use any numbering scheme you wish; the alias values do not need to be consecutive.

39.4.1 Configuration package used

Package	Sections
network	interface

39.4.2 Configuring SNMP interface alias

To enter and SNMP alias for an interface, select **Network->Interfaces->Edit->Common Configuration->Advanced Settings**.

Enter a small index value for **SNMP Alias ifindex** that is unique to this interface. To retrieve SNMP statistics for this interface, the SNMP manager should be configured to poll (**snmp_alias_ifindex + 1000**). For example, if an interface is configured with an **snmp_alias_ifindex** of 11, then the SNMP manager should poll **ifIndex=1011**. The ifIndex will remain fixed regardless of how many times the underlying interface is added or removed.

If the Linux interface associated with the UCI entry is active when the alias index is polled, the normal ifEntry information for that interface is reported. Otherwise, a dummy entry is created with the same ifDescr, and its ifOper field set to **DOWN**.

Note: if you are using SIM roaming, where mobile interfaces are created dynamically, you need to specify a fixed **snmp_alias_ifindex** value and a fixed **ifName** value in the roaming template. All roaming entries will then map to the same Linux interface name and underlying device.

(2) Alias ifindex SNMP agent. Alias indexes are present at 1000 offset. So setting 1 here will create snmp ifTable entry 1001.

Useful when interface creates new linux interface on every startup (e.g. ppp interface). With this set the interface could be monitored via constant snmp agent interface table entry

Figure 188: The interface SNMP Alias ifindex field advanced settings page

UCI/Package Option	Description				
Web: SNMP Alias ifindex UCI: network.@interface[X].snmp_alias_ifindex Opt: snmp_alias_ifindex	Defines a static SNMP interface alias index for this interface, that can be polled using via the SNMP interface index (snmp_alias_ifindex+1000) <table border="1" style="margin-top: 5px; width: fit-content;"> <tr> <td style="padding: 2px;">Blank</td><td style="padding: 2px;">No SNMP interface alias index</td></tr> <tr> <td style="padding: 2px;">Range</td><td style="padding: 2px;">0 - 4294966295</td></tr> </table>	Blank	No SNMP interface alias index	Range	0 - 4294966295
Blank	No SNMP interface alias index				
Range	0 - 4294966295				
Web: n/a UCI: network.@interface[X].snmp_alias_ifdescr Opt: snmp_alias_ifdescr	Defines an alias name to be reported for the UCI name in the enterprise MIB for UCI interfaces, and in alias entries in the ifIndex table. If present, this option supercedes the default ifDescr value (usually the UCI interface name, or configured ifName) <table border="1" style="margin-top: 5px; width: fit-content;"> <tr> <td style="padding: 2px;">Blank</td><td style="padding: 2px;">No SNMP interface alias name</td></tr> <tr> <td style="padding: 2px;">Range</td><td style="padding: 2px;"></td></tr> </table>	Blank	No SNMP interface alias name	Range	
Blank	No SNMP interface alias name				
Range					

Table 130: Information table for static SNMP alias interface

39.4.3 Configuring SNMP interface alias using the command line

SNMP interface alias is configured under the network package **/etc/config/network**

The following examples use an interface section named MOBILE.

39.4.3.1 SNMP interface alias using UCI

```
root@GW_router:~# uci show network
network.MOBILE=interface
.....
network.MOBILE.snmp_alias_ifindex=11
network.MOBILE.snmp_alias_ifdescr=primary_mobile
.....
```

39.4.3.2 SNMP interface alias using package options

```
root@GW_router:~# uci show network
config interface 'MOBILE'
.....
option snmp_alias_ifindex '11'
option snmp_alias_ifdescr 'primary_mobile'
.....
```

39.4.4 SNMP interface alias MIBS

OID Name	OID
interface alias table	.1.3.6.1.2.1.2.2.1.1.
snmp_alias_ifindex	.1.3.6.1.2.1.2.2.1.1.<snmp_alias_ifindex+1000>
snmp_alias_ifdescr	1.3.6.1.4.1.2078.3.2.66.1.1.<index>.{5,6}

39.5 SNMP diagnostics

39.5.1 SNMP process

To check the SNMP process is running correctly, enter **pgrep -fl snmpd**.

```
root@GW_router:~# pgrep -fl snmpd
6970 /usr/sbin/snmpd -Lsd0-6 -p /var/run/snmpd.pid -m -c
/var/conf/snmpd.conf
```

39.5.2 SNMP port

To check that SNMP service is listening on the configured port, enter **netstat -pantu | grep snmp**

```
root@GW_router:~# netstat -pantu | grep snmp
udp    0 0 0.0.0.0:161    0.0.0.0:*        6970/snmpd
```

39.5.3 Retrieving SNMP values

SNMP values can be queried by an snmpwalk or snmpget either locally or remotely.

39.5.3.1 snmpwalk

To do an snmpwalk locally, use **snmpwalk**. An example snmpwalk is shown below:

```
root@GW_router:~# snmpwalk -c public -v 1 localhost .1.3.6.1.2.1.1
iso.3.6.1.2.1.1.1.0 = STRING: "SATEL GWXXXX, SN# 00E0C812D1A0, EDG-
21.00.07.008"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.2078
iso.3.6.1.2.1.1.3.0 = Timeticks: (71816) 0:11:58.16
iso.3.6.1.2.1.1.4.0 = STRING: "info@SATEL.com"
iso.3.6.1.2.1.1.5.0 = STRING: "GWXXXX"
iso.3.6.1.2.1.1.6.0 = STRING: "UK"
iso.3.6.1.2.1.1.7.0 = INTEGER: 79
iso.3.6.1.2.1.1.8.0 = Timeticks: (60) 0:00:00.60
iso.3.6.1.2.1.1.9.1.2.1 = OID: iso.3.6.1.2.1.4
iso.3.6.1.2.1.1.9.1.2.2 = OID: iso.3.6.1.6.3.1
iso.3.6.1.2.1.1.9.1.2.3 = OID: iso.3.6.1.2.1.49
iso.3.6.1.2.1.1.9.1.2.4 = OID: iso.3.6.1.2.1.50
iso.3.6.1.2.1.1.9.1.2.5 = OID: iso.3.6.1.6.3.16.2.2.1
iso.3.6.1.2.1.1.9.1.2.6 = OID: iso.3.6.1.6.3.10.3.1.1
iso.3.6.1.2.1.1.9.1.2.7 = OID: iso.3.6.1.6.3.11.3.1.1
iso.3.6.1.2.1.1.9.1.2.8 = OID: iso.3.6.1.6.3.15.2.1.1
iso.3.6.1.2.1.1.9.1.2.9 = OID: iso.3.6.1.2.1.10.131
iso.3.6.1.2.1.1.9.1.4.4 = Timeticks: (35) 0:00:00.35
iso.3.6.1.2.1.1.9.1.4.5 = Timeticks: (38) 0:00:00.38
iso.3.6.1.2.1.1.9.1.4.6 = Timeticks: (38) 0:00:00.38
iso.3.6.1.2.1.1.9.1.4.7 = Timeticks: (38) 0:00:00.38
iso.3.6.1.2.1.1.9.1.4.8 = Timeticks: (38) 0:00:00.38
iso.3.6.1.2.1.1.9.1.4.9 = Timeticks: (60) 0:00:00.60
....
```

39.5.3.2 snmpget

To do an snmpget locally, use **snmpget**. An example snmpget is shown below.

```
root@GW_router:~# snmpget -c public -v 1 localhost .1.3.6.1.4.1.2078.3.14.2  
iso.3.6.1.4.1.2078.3.14.2 = STRING: "EDG-21.00.07.008"
```

39.5.4 SNMP status

To view an overview including tx/rx packets and uptime of the SNMP process, enter **snmpstatus**.

```
root@GW_router:~# snmpstatus -c public -v 2c localhost  
[UDP: [0.0.0.0]->[127.0.0.1]:161]=>[SATEL GWXXXX, SN# 00E0C812D1A0, EDG-  
21.00.07.008] Up: 0:17:05.87  
Interfaces: 21, Recv/Trans packets: 47632/9130 | IP: 15045/8256  
15 interfaces are down!
```

40 Event system

SATEL routers include a event system feature called as "varying actions". 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.

40.1 Configuration package used

Package	Section
va_eventd	main
	forwarding
	target
	conn_tester

40.2 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.

40.3 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**'.

40.4 Supported targets

The table below describes the targets currently supported.⁸

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.
SMS	Event sent via SMS.

Table 131: Targets currently supported

The attributes of a target vary significantly depending on its type.

40.5 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 132: Event system - supported connection tester methods

40.6 Configuring the event system using the web interface

Configuring the event system using the web interface is not currently supported.

40.7 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.

40.7.1 Va_eventd: main section

40.7.1.1 Main using UCI

```
root@GW_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
```

40.7.1.2 Main using package options

```
root@GW_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'
```

40.7.1.3 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	/tmp/event_buffer	File where the events will be stored before being processed. Default file is /tmp/event_buffer.
UCI: va_eventd.main.event_queue_size Opt: event_queue_size	Range	Maximum size of the event queue in bytes. Default value is 128k. 128K 128 kilobytes Range

Table 133: Information table for event settings main section

40.7.2 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, enter:

```
va_eventd.Monitor=forwarding
```

In the examples below, no forwarding label has been defined.

40.7.3 Forwarding using UCI

```
root@GW_router:~# uci show va_eventd
va_eventd.@forwarding[0]=forwarding
va_eventd.@forwarding[0].enabled=1
va_eventd.@forwarding[0].className=etherenet
```

```
va_eventd.@forwarding[0].eventName=LinkUp
va_eventd.@forwarding[0].severity=warning-critical
va_eventd.@forwarding[0].target=syslog1
```

40.7.4 Forwarding using package options

```
root@GW_router:~# uci export va_eventd
config forwarding
    option enabled '1'
    option className 'ethernet'
    option eventName 'LinkUp'
    option severity 'warning-critical'
    option target 'syslog1'
```

40.7.5 Forwarding table options

UCI/Package Option	Description																
UCI: va_eventd.<forwarding label>.enabled Opt: enabled	Enables or disables event generation. <table border="1" style="margin-left: 20px;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.												
0	Disabled.																
1	Enabled.																
UCI: va_eventd.<forwarding label>.className Opt: className	Only generate events with the given className. Available class names can be viewed using 'vae_cli -d' command. <table border="1" style="margin-left: 20px;"> <tr><td>ClassName</td></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>	ClassName	internal	mobile	ethernet	isdn	power	usage	pvc	l2tp	auth	ipsec	wifi	ppp	adsl	system	ntp
ClassName																	
internal																	
mobile																	
ethernet																	
isdn																	
power																	
usage																	
pvc																	
l2tp																	
auth																	
ipsec																	
wifi																	
ppp																	
adsl																	
system																	
ntp																	
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.																

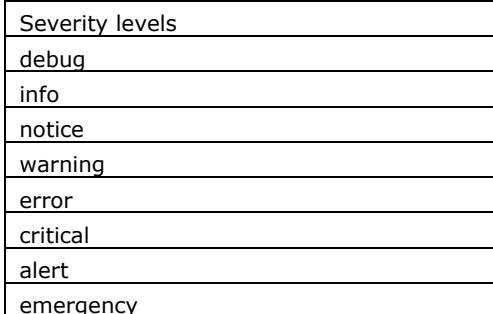
UCI: va_eventd.<forwarding label>.severity Opt: severity	Only generate events with a severity in the severity range. This is optional. Severity must be a range in the form severity1-severity2. Example: va_eventd.@forwarding[0].severity=emergency-warning 
UCI: va_eventd.<forwarding label>.target Opt: target	Target to send the event to. This parameter refers to the target name as defined in a target config section.

Table 134: Information table for event system forwarding rules

40.7.6 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, enter:

```
config conn_tester 'Tester1'
```

To define a forwarding label of Tester1 using UCI, enter:

```
va_eventd.Tester1=conn_tester
```

In the examples below no connection tester label has been defined.

40.7.6.1 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.

40.7.6.2 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
```

40.7.6.3 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'
```

40.7.6.4 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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
UCI: va_eventd.<conn_tester_label>.type Opt: type	Set to ping for a ping connection tester. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>ping</td><td>Ping connection tester.</td></tr> <tr> <td>link</td><td>Link connection tester.</td></tr> </table>	ping	Ping connection tester.	link	Link 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 135: Information table for ping connection tester settings

40.7.6.5 Link connection tester

A link connection tester tests a connection by checking the status of the interface being used.

40.7.6.6 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
Link connection tester using package options
```

```

config conn_tester
    option name 'linktest'
    option enabled '1'
    option type 'link'
    option link_iface 'eth0'

```

40.7.6.7 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. <table border="1"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
UCI: va_eventd.<conn_tester label>.type Opt: type	Set to 'link' for a link connection tester. <table border="1"> <tr> <td>ping</td><td>Ping connection tester.</td></tr> <tr> <td>link</td><td>Link connection tester.</td></tr> </table>	ping	Ping connection tester.	link	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 136: Information table for link connection tester settings

40.7.7 Supported targets

There are four possible targets:

- Syslog target
- Email target
- SNMP target
- Exec target
- SMS 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, enter:

```
config target 'Target1'
```

To define a target label of Target1 using UCI, enter:

```
va_eventd.Target1=target
```

40.7.7.1 Syslog target

When a syslog target receives an event, it sends it to the configured syslog server. In the examples below no target label has been defined.

40.7.7.2 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].tcp_syslog=0
va_eventd.@target[0].addr=192.168.0.1:514
va_eventd.@target[0].conn_tester=pinger
va_eventd.@target[0].snmp_version=3
```

40.7.7.3 Syslog target using package options

```
config target
    option name syslog
    option enabled '1'
    option type 'syslog'
    option tcp_syslog '0'
    option target_addr '192.168.0.1:514'
    option conn_tester 'pinger'
    option snmp_version '3'
```

40.7.7.4 Syslog target table options

UCI/Package Option	Description										
UCI: va_eventd.<target label>.name Opt: name	Name of the target. This is to be used in the forwarding section.										
UCI: va_eventd.<target label>.enabled Opt: enabled	Enable this target. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
0	Disabled.										
1	Enabled.										
UCI: va_eventd.<target label>.type Opt: type	Must be 'syslog' for a syslog target. <table border="1" style="margin-left: auto; margin-right: auto;"> <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> <tr> <td>sms</td><td>SMS target.</td></tr> </table>	Syslog	Syslog target.	email	Email target.	snmptrap	SNMP target.	exec	Exec target.	sms	SMS target.
Syslog	Syslog target.										
email	Email target.										
snmptrap	SNMP target.										
exec	Exec target.										
sms	SMS target.										
UCI: va_eventd.<target label>.tcp_syslog Opt: tcp_syslog	Defines whether to use TCP for delivery of syslog messages. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
0	Disabled.										
1	Enabled.										
UCI: va_eventd.<target label>.target_addr Opt: target_addr	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.										

UCI: va_eventd.<target label>.conn_tester Opt: conn_tester	Name of the connection tester to use for this target.						
UCI: va_eventd.<target label>.snmp_version Opt: snmp_version	Ability to change snmp version. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>1</td> <td>Version1</td> </tr> <tr> <td>2c</td> <td>Version 2c</td> </tr> <tr> <td>3</td> <td>Version 3</td> </tr> </table>	1	Version1	2c	Version 2c	3	Version 3
1	Version1						
2c	Version 2c						
3	Version 3						

Table 137: Information table for syslog target settings

40.7.7.5 Email target

When an email target receives an event, it sends it to the configured email address.

40.7.7.6 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=%{serial} %{severityName} %{eventName}
}!!!
va_eventd.@target[0].body_template=%{eventName} (%{class}.%{subclass})
happened!
va_eventd.@target[0].conn_tester=pinger

```

40.7.7.7 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 "%{serial} %{severityName} %{eventName}!!!"
option body_template "%{eventName} (%{class}.%{subclass}) happened!"

```

40.7.7.8 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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>		0	Disabled.	1	Enabled.						
0	Disabled.											
1	Enabled.											
UCI: va_eventd.<target label>.type Opt: type	Must be 'email' for a syslog target. <table border="1" style="margin-left: auto; margin-right: auto;"> <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> <tr> <td>sms</td> <td>SMS target.</td> </tr> </table>		syslog	Syslog target.	email	Email target.	snmptrap	SNMP target.	exec	Exec target.	sms	SMS target.
syslog	Syslog target.											
email	Email target.											
snmptrap	SNMP target.											
exec	Exec target.											
sms	SMS target.											
UCI: va_eventd.<target label>.smtp_addr Opt: smtp_addr	IP address or FQDN and 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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>		0	Disabled.	1	Enabled.						
0	Disabled.											
1	Enabled.											
UCI: va_eventd.<target label>.tls_starttls Opt: tis_starttls	Enable StartTLS support. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>		0	Disabled.	1	Enabled.						
0	Disabled.											
1	Enabled.											
UCI: va_eventd.<target label>.tls_forcessl3 Opt: tis_forcessl3	Force SSLv3 for TLS. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>		0	Disabled.	1	Enabled.						
0	Disabled.											
1	Enabled.											
UCI: va_eventd.<target label>.timeout_sec Opt: timeout_sec	Email send timeout in seconds. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>10</td> <td>10 seconds</td> </tr> <tr> <td>Range</td> <td></td> </tr> </table>		10	10 seconds	Range							
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. Supported parameters: Serial number: %{serial}; Severity: %{severityName}; Event Name: %{eventName}. Example: option subject_template '%{serial} %{severityName} %{eventName}!'
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 138: Information table for email target settings

40.7.8 SNMP target

When a SNMP target receives an event, it sends it in a trap to the configured SNMP manager.

40.7.8.1 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
```

40.7.8.2 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'
```

40.7.8.3 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. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td><td>Disabled.</td></tr> <tr> <td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

UCI: va_eventd.<target label>.type Opt: type	Must be snmptrap for a snmp target. <table border="1"> <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> <tr><td>sms</td><td>SMS target.</td></tr> </table>	syslog	Syslog target.	email	Email target.	snmptrap	SNMP target.	exec	Exec target.	sms	SMS target.
syslog	Syslog target.										
email	Email target.										
snmptrap	SNMP target.										
exec	Exec target.										
sms	SMS 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 139: Information table for snmp target settings

40.7.8.4 Exec target

When an exec target receives an event, it executes a shell command.

40.7.8.5 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}

```

40.7.8.6 Exec target using package options

```

config target
    option name 'logit'
    option enabled '1'
    option type 'exec'
    option cmd_template "logger -t eventer %{eventName}"

```

40.7.8.7 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. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

UCI: va_eventd.<target label>.type Opt: type	Must be exec for an exec target. <table border="1"> <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> <tr><td>sms</td><td>SMS target.</td></tr> </table>	syslog	Syslog target.	email	Email target.	snmptrap	SNMP target.	exec	Exec target.	sms	SMS target.
syslog	Syslog target.										
email	Email target.										
snmptrap	SNMP target.										
exec	Exec target.										
sms	SMS target.										
UCI: va_eventd.<target label>.cmd_template Opt: cmd_template	Template of the command to execute.										

Table 140: Information table for exec target settings

40.7.8.8 SMS target

When SMS target receives an event, it sends SMS message.

40.7.8.9 SMS target using UCI

```

va_eventd.@target[0]=target
va_eventd.@target[0].name=sms
va_eventd.@target[0].enabled=1
va_eventd.@target[0].type=sms
va_eventd.@target[0].callee=0123321123321
va_eventd.@target[0].template=%{eventName}

```

40.7.8.10 SMS target using package options

```

config target
    option name 'sms'
    option enabled '1'
    option type 'sms'
    option callee '0123321123321'
    option template '%{eventName}'

```

40.7.8.11 SMS 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. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
0	Disabled.										
1	Enabled.										
UCI: va_eventd.<target label>.type Opt: type	Must be sms for an sms target. <table border="1"> <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> <tr><td>sms</td><td>SMS target.</td></tr> </table>	syslog	Syslog target.	email	Email target.	snmptrap	SNMP target.	exec	Exec target.	sms	SMS target.
syslog	Syslog target.										
email	Email target.										
snmptrap	SNMP target.										
exec	Exec target.										
sms	SMS target.										
UCI: va_eventd.<target label>.callee Opt: callee	Defines the SMS number to send to. <table border="1"> <tr><td>blank</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	blank		Range							
blank											
Range											

UCI: va_eventd.<target label>.template Opt: template	Template of the command to execute. Uses template associated with that particular event, which can be listed under "vae_cli -d".
---	--

Table 143: Information table for SMS target settings

40.8 Event system diagnostics

40.8.1 Displaying VA events

To view a list of all available class names, events and severity levels, enter:

```
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 error: %{p1}	warning SMS send
mobile 10 SMSSent to %{p1}: %{p2}	notice Sent SMS
ethernet 1 LinkUp	notice Ethernet %{p1} up
ethernet 2 LinkDown down	notice Ethernet %{p1}
auth 2 BadPasswordSSH from %{p2}: ba..	warning SSH login attempt
auth 3 BadUserConsole attempt on %{p1}: ..	warning Console login
auth 4 BadPasswordConsole attempt on %{p2}: ..	warning Console login
auth 5 BadUserTelnet attempt: bad username	warning Telnet login
auth 6 BadPasswordTelnet attempt: bad passwo..	warning Telnet login
auth 7 BadUserLuCI attempt: bad username..	warning LuCI login
auth 8 BadPasswordLuCI attempt: bad password..	warning LuCI login
auth 9 LoginSSH user %{p2} from %{p3}	notice SSH login:
auth 10 LogoffSSH user %{p1} due to "%..	notice SSH logoff:
auth 11 LoginConsole user %{p1} on %{p2}	notice Console login:
auth 12 LogoffConsole on %{p1}	notice Console logoff
auth 13 LoginTelnet user %{p1}	notice Telnet login:
auth 14 LoginLuCI user %{p1}	notice LuCI login:
auth 15 ConsoleCommand	informat %{p1}@%{p2} %{p3}
auth 16 LuCIAction %{p1}@%{p2} %{p3} %{p4} %{p5}	informat
ipsec 6 IPSecInitIKE established	informat IPSec IKE %{p1}
ipsec 7 IPSecInitSA established	informat IPSec SA %{p1}
ipsec 8 IPSecCloseIKE deleted	informat IPSec IKE %{p1}
ipsec 9 IPSecCloseSA closed	informat IPSec SA %{p1}

```

| 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} (proto..

| ppp        | 2  | LinkDown                 | informat | PPP for
interface %{p2} (proto..

| 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}
```

40.8.2 Viewing the event system config

To view the event system configuration via UCI, enter:

```
root@GW_router:~# uci show va_eventd
```

To view the event system config via package options

```
root@GW_router:~# uci export va_eventd
```

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 '%{serial} %{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}'
```

41 Configuring Terminal Server

41.1 Overview

Terminal Server is a background application whose main task is to forward data between TCP connections or UDP streams and asynchronous or synchronous serial ports.

Terminal Server application serves up to 4 sessions simultaneously one for each serial port, depending on the device. Each Terminal Server session has an IP endpoint and an associated specific serial port.

You can configure the IP endpoint of each Terminal Server session to be a:

- TCP server: each session is listening on a unique port.
- TCP client: Terminal Server makes a TCP connection to external TCP server.
- UDP endpoint: Terminal Server forwards data between a UDP stream and a serial port.

41.2 Configuration packages used

Package	Sections
Tservd	Main
	Port

41.3 Configuring Terminal Server using the web interface

In the top menu, select **Services -> Terminal Server**. The Terminal Server Configuration page appears. You must configure two main sections: Main Settings and Port Settings.

41.3.1 Configure main settings

The screenshot shows the 'Terminal Server' configuration page. At the top, it says 'Terminal Server' and 'Configuration of the Terminal Server'. Below that is a 'Main Settings' section with the following options:

- Enable**: A checkbox labeled 'enable terminal server'.
- Debug Enable**: A checkbox labeled 'enables detailed debug logging (state transitions, data transfer etc)'.
- Syslog severity**: A dropdown menu set to 'Informational'.
- Log RX-TX**: A checkbox labeled 'enable logging data transfers'.

Figure 189: The terminal server main settings page

Web Field/UCI/Package Option	Description																
Web: Enable UCI: tservd.main.enable Opt: enable	Enables Terminal Server on the router. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.												
0	Disabled.																
1	Enabled.																
Web: Debug Enable UCI: tservd.main.debug_ev_enable Opt: debug_ev_enable	Enables detailed debug logging. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.												
0	Disabled.																
1	Enabled.																
Web: Syslog severity UCI: tservd.main.log_severity Opt: log_severity	Determines the syslog level. Events up to this priority will be logged. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Emergency</td> </tr> <tr> <td>1</td> <td>Alert</td> </tr> <tr> <td>2</td> <td>Critical</td> </tr> <tr> <td>3</td> <td>Error</td> </tr> <tr> <td>4</td> <td>Warning</td> </tr> <tr> <td>5</td> <td>Notice</td> </tr> <tr> <td>6</td> <td>Informational</td> </tr> <tr> <td>7</td> <td>Debug</td> </tr> </table>	0	Emergency	1	Alert	2	Critical	3	Error	4	Warning	5	Notice	6	Informational	7	Debug
0	Emergency																
1	Alert																
2	Critical																
3	Error																
4	Warning																
5	Notice																
6	Informational																
7	Debug																
Web: Log RX-TX UCI: tservd.main.debug_rx_tx_enable Opt: debug_rx_tx_enable	Enables logging data transfers. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.												
0	Disabled.																
1	Enabled.																

Table 141: Information table for main settings

41.3.2 Configure port settings

The Port Settings section is divided into 3 sub-sections:

- General
- Serial
- Network

41.3.2.1 Port settings: general section

In this section you can configure general port settings. The settings are usually the same for the central and the remote site.

Port Settings

PORt1

- [General](#)
- [Serial](#)
- [Network](#)

Enable [enable port](#)

Network Forwarding Buffer Size [Forwarding buffer size \(serial to network\)](#)

Network Forwarding Timeout (ms) [Forwarding timeout in milliseconds \(serial to network\)](#)

Network Forwarding timer mode [Forwarding timer mode \(serial to network\)](#)

Serial Forwarding Buffer Size [Forwarding buffer size \(network to serial\)](#)

Serial Forwarding Timeout (ms) [Forwarding timeout in milliseconds \(network to serial\)](#)

Serial Forwarding timer mode [Forwarding timer mode \(network to serial\)](#)

Proxy mode [enable proxy mode](#)

Disable remote client's local echo (Telnet option)

Telnet COM port control (RFC2217)

Enable HDLC Pseudowire over UDP (RFC4618)

Serial receive debug log size [bytes \(0=disable\)](#)

Serial transmit debug log size [bytes \(0=disable\)](#)

Figure 190: The general tab fields

Web Field/UCI/Package Option	Description	
Web: Enable UCI: tservd.@port[0].enable Opt: enable	Enables Terminal Server port.	
	<input type="checkbox"/> 0	Disabled.
	<input type="checkbox"/> 1	Enabled.
Web: Network Forwarding Buffer Size UCI: tservd.@port[0]. fwd_buffer_size Opt: fwd_buffer_size	Forwarding buffer size in bytes (serial to network).	
	<input type="text" value="256"/>	256 bytes
	<input type="text" value="Range"/>	0-2048
Web: Network Forwarding Timeout(ms) UCI: tservd.@port[0]. fwd_timeout Opt: fwd_timeout	Forwarding timeout in milliseconds (serial to network).	
	<input type="text" value="30"/>	30 ms
	<input type="text" value="Range"/>	0-10000
Web: Network Forwarding Timer Mode UCI: tservd.@port[0]. fwd_timer_mode Opt: fwd_timer_mode	Forwarding timer mode (serial to network).	
	<input type="button" value="Idle"/>	Timer is re-started on each received data.
	<input type="button" value="Aging"/>	Timer started on the first Rx.
Web: Serial Forwarding Buffer Size UCI: tservd.@port[0]. sfwd_buffer_size Opt: sfwd_buffer_size	Forwarding buffer size in bytes (network to serial). Set to 0 to use maximum possible network Rx buffer size.	
	<input type="text" value="0"/>	2048 bytes
	<input type="text" value="Range"/>	0-2048
Web: Serial Forwarding Timeout (ms) UCI: tservd.@port[0]. sfwd_timeout Opt: sfwd_timeout	Forwarding timeout in milliseconds (network to serial). Set to 0 to forward to serial immediately.	
	<input type="text" value="20"/>	20 ms
	<input type="text" value="Range"/>	0-10000

Web: Serial Forwarding Timer Mode UCI: tservd.@port[0].sfwd_timer_mode Opt: sfwd_timer_mode	Forwarding timer mode (network to serial). <table border="1"> <tr> <td>Idle</td><td>Timer is re-started on each received data</td></tr> <tr> <td>Aging</td><td>Timer started on the first Rx.</td></tr> </table>	Idle	Timer is re-started on each received data	Aging	Timer started on the first Rx.
Idle	Timer is re-started on each received data				
Aging	Timer started on the first Rx.				
Web: Proxy Mode UCI: tservd.@port[0].proxy_mode Opt: proxy_mode	Defines if special proxy mode is configured to allow 'hijacking' of the terminal server. It allows a connection to be made from a remote location and redirect terminal server data temporarily for troubleshooting. When enabled a TCP proxy server is started which listens for an incoming TCP connection from a remote peer. Once an incoming new TCP connection on the proxy server TCP port is accepted: The existing terminal server TCP client connection is disconnected. The terminal server automatically reconnects the TCP client side but this time to the local loopback address 127.0.0.1 and to the local proxies TCP port number. Once the proxy server has both local and remote TCP sessions connected it simply forwards the data between the two connections, taking into account the flow control. When either side TCP socket closes, the main terminal server client re-connects to the normal IP destination and the server proxy returns to listening for another connection from the far end. <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Disable Remote Client's Local Echo (Telnet option) UCI: tservd.@port[0]. disable_echo Opt: disable_echo	Set to 1 to send IAC WILL ECHO Telnet option to remote client forcing it to disable local echo. For server mode only. <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Telnet COM Port Control UCI: tservd.@port[0]. com_port_control Opt: com_port_control	Set to 1 to enable support for Telnet COM port control (RFC2217). <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Enable HDLC Pseudowire over UDP (RFC4618) UCI: tservd.@port[0]. hdlc_pw_enabled Opt: hdlc_pw_enabled	Set to 1 to enable HDLC pseudowire over UDP support based on RFC4618 (requires Transport Mode (udpmode) to be enabled) <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Serial Receive Debug Log Size UCI: tservd.@port[0]. serialRxLogSize Opt: serialRxLogSize	Configures serial receive log size in bytes and enables receive data logging. <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Serial Transmit Debug Log Size UCI: tservd.@port[0].serialTxLogSize Opt: serialTxLogSize	Configures serial transmit log size in bytes and enables transmit data logging. <table border="1"> <tr> <td>0</td> <td>Disabled.</td> </tr> <tr> <td>1</td> <td>Enabled.</td> </tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				

Table 142: Information table for port settings section

41.3.2.2 Port settings: serial section

In this section you can configure the serial interface settings, such as port mode, port speed, parity stip bit and so on.

Note:

- The displayed settings vary depending on options selected.
- DTR <--> DSR signalling is not available on GW600 router models.

POR1

General Serial Network

Device	/dev/ttySC0	serial device name
Portmode	rs232	serial interface mode
Speed (bps)	9600	asynchronous baud rate
Word size	8	serial device word size in bits
Parity	none	serial device parity in bits
Stop bits	1	serial device number of stop bits
Flow Control	NONE	serial device flow control type
RS485 termination	<input type="checkbox"/>	enable RS485 line termination
Auto RTS Invert	<input type="checkbox"/>	Invert RTS in auto-RTS mode
Keep serial port always open	<input type="checkbox"/>	keep serial port always activated
RS232 Half Duplex	<input type="checkbox"/>	enable RS232 half duplex mode for interfacing to external V.23 modem
RTS timeout	30	RS232 half duplex mode RTS timeout in milliseconds
POST RTS timeout	30	RS232 half duplex mode Post RTS timeout in milliseconds
Atmel USB serial card	<input type="checkbox"/>	enable support for Atmel USB serial card
Dual X.21 card bit reverse	<input type="checkbox"/>	
Dual X.21 card DTE TT Invert	<input type="checkbox"/>	
Dual X.21 card DCE TCLK Invert	<input type="checkbox"/>	
Dual X.21 card DCE RCLK Invert	<input type="checkbox"/>	
Dual X.21 card CLK Invert	<input type="checkbox"/>	
Dual X.21 card RX data delay	0	

Figure 191: The serial section fields (portmode RS232 and usb serial disabled)

Web Field/UCI/Package Option	Description											
Web: Device UCI: tservd.@port[0].devName Opt: devName	Serial device name. <table border="1"> <tr><td>/dev/ttySC0</td><td>serial port 1</td></tr> <tr><td>/dev/ttySC1</td><td>serial port 2</td></tr> <tr><td>/dev/ttySC2</td><td>serial port 3</td></tr> <tr><td>/dev/ttySC3</td><td>serial port 4</td></tr> </table>		/dev/ttySC0	serial port 1	/dev/ttySC1	serial port 2	/dev/ttySC2	serial port 3	/dev/ttySC3	serial port 4		
/dev/ttySC0	serial port 1											
/dev/ttySC1	serial port 2											
/dev/ttySC2	serial port 3											
/dev/ttySC3	serial port 4											
Web: Port mode UCI: tservd.@port[0].port_mode Opt: port_mode	Sets the serial interface mode. <table border="1"> <tr><td>rs232</td><td>RS232 mode</td></tr> <tr><td>rs485hdx</td><td>RS485 2 wire half duplex mode in which transmitter drives RTS.</td></tr> <tr><td>rs485fdx</td><td>Rs485 4 wire full duplex mode.</td></tr> <tr><td>v23</td><td>Uses V.23 leased line card driver.</td></tr> <tr><td>x21</td><td>Uses USB serial card in sync mode.</td></tr> </table>		rs232	RS232 mode	rs485hdx	RS485 2 wire half duplex mode in which transmitter drives RTS.	rs485fdx	Rs485 4 wire full duplex mode.	v23	Uses V.23 leased line card driver.	x21	Uses USB serial card in sync mode.
rs232	RS232 mode											
rs485hdx	RS485 2 wire half duplex mode in which transmitter drives RTS.											
rs485fdx	Rs485 4 wire full duplex mode.											
v23	Uses V.23 leased line card driver.											
x21	Uses USB serial card in sync mode.											
Web: Speed (bps) UCI: tservd.@port[0].speed Opt: speed	Serial device speed in baud (bps). <table border="1"> <tr><td>9600</td><td></td></tr> <tr><td>Range</td><td>115200; 57600; 38400; 19200; 9600 4800; 2400; 1800; 1200; 600; 300; 200; 150; 134; 110; 75; 50</td></tr> </table>		9600		Range	115200; 57600; 38400; 19200; 9600 4800; 2400; 1800; 1200; 600; 300; 200; 150; 134; 110; 75; 50						
9600												
Range	115200; 57600; 38400; 19200; 9600 4800; 2400; 1800; 1200; 600; 300; 200; 150; 134; 110; 75; 50											
Web: Word size UCI: tservd.@port[0].wsize Opt: wsize	Serial device word size. <table border="1"> <tr><td>8</td><td></td></tr> <tr><td>Range</td><td>5-8</td></tr> </table>		8		Range	5-8						
8												
Range	5-8											
Web: Parity UCI: tservd.@port[0].parity Opt: parity	Serial device parity. <table border="1"> <tr><td>0</td><td>None</td></tr> <tr><td>1</td><td>Even</td></tr> <tr><td>2</td><td>Odd</td></tr> <tr><td>3</td><td>Space</td></tr> </table>		0	None	1	Even	2	Odd	3	Space		
0	None											
1	Even											
2	Odd											
3	Space											
Web: Stop Bits UCI: tservd.@port[0].stops Opt: stops	Serial device number of stop bits. <table border="1"> <tr><td>1</td><td></td></tr> <tr><td>Range</td><td>1-2</td></tr> </table>		1		Range	1-2						
1												
Range	1-2											
Web: Flow Control UCI: tservd.@port[0].fc_mode Opt: fc_mode	Serial flow control mode. <table border="1"> <tr><td>0</td><td>None</td></tr> <tr><td>1</td><td>RTS/CTS</td></tr> <tr><td>2</td><td>XON/XOFF</td></tr> </table>		0	None	1	RTS/CTS	2	XON/XOFF				
0	None											
1	RTS/CTS											
2	XON/XOFF											
Web: RS485 Termination UCI: tservd.@port[0].rs485_line_termination Opt: rs485_line_termination	Enables or disable RS485 termination. Applies only if port mode is set to RS485. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>		0	Disabled.	1	Enabled.						
0	Disabled.											
1	Enabled.											
Web: Auto RTS Invert UCI: tservd.@port[0].rtsinvert Opt: rtsinvert	Invert RTS in auto-RTS mode, if portmode is set to RS485. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>		0	Disabled.	1	Enabled.						
0	Disabled.											
1	Enabled.											
Web: Keep Serial Port Always Open UCI: tservd.@port[0].tty_always_open Opt: tty_always_open	Keep serial port always open. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>		0	Disabled.	1	Enabled.						
0	Disabled.											
1	Enabled.											
Web: RS232 Half Duplex UCI: tservd.@port[0].hd_mode Opt: hd_mode	Defines whether to enable special mode in the asynchronous serial driver for communication to an externally connected V.23 half-duplex modem. Note: this setting does not enable half-duplex mode in the serial hardware of the router. <table border="1"> <tr><td>0</td><td>Full duplex mode.</td></tr> <tr><td>1</td><td>Half duplex mode.</td></tr> </table>		0	Full duplex mode.	1	Half duplex mode.						
0	Full duplex mode.											
1	Half duplex mode.											
Web: RTS Timeout UCI: tservd.@port[0].rts_timeout Opt: rts_timeout	In RS232 half-duplex mode, time in milliseconds between raising RTS and enabling the transmitter. For use with externally connected V.23 modem. <table border="1"> <tr><td>30</td><td>30ms</td></tr> <tr><td>Range</td><td></td></tr> </table>		30	30ms	Range							
30	30ms											
Range												

Web: POST RTS Timeout UCI: tservd.@port[0].post_rts_timeout Opt: post_rts_timeout	In RS232 half duplex mode, time in milliseconds between dropping RTS (transmission finished) and enabling the receiver. For use with externally connected V.23 modem. <table border="1"> <tr><td>20</td><td>20 ms</td></tr> <tr><td colspan="2">Range</td></tr> </table>	20	20 ms	Range							
20	20 ms										
Range											
Web: n/a UCI: tservd.@port[0].v23_tx_gain Opt: v23_tx_gain	Defines the transmit gain for v23 mode. <table border="1"> <tr><td>2</td><td>Transmit samples multiplied by 2</td></tr> <tr><td colspan="2">Range</td></tr> </table>	2	Transmit samples multiplied by 2	Range							
2	Transmit samples multiplied by 2										
Range											
Web: n/a UCI: tservd.@port[0].v23_rx_loss Opt: v23_rx_loss	Defines the receive loss for v23 mode. <table border="1"> <tr><td>1</td><td>Receive samples divided by 1.</td></tr> <tr><td colspan="2">Range</td></tr> </table>	1	Receive samples divided by 1.	Range							
1	Receive samples divided by 1.										
Range											
Web: n/a UCI: tservd.@port[0].v23_rts_to_cts_delay Opt: v23_rts_to_cts_delay	Defines the v23 modem RTS to CTS delay in milliseconds. <table border="1"> <tr><td>20</td><td></td></tr> <tr><td colspan="2">Range</td></tr> </table>	20		Range							
20											
Range											
Web: n/a UCI: tservd.@port[0].v23_is_four_wire Opt: v23_is_four_wire	Defines the V23 modem LIM operation. <table border="1"> <tr><td>0</td><td>2-wire</td></tr> <tr><td>1</td><td>4-wire</td></tr> </table>	0	2-wire	1	4-wire						
0	2-wire										
1	4-wire										
Web: n/a UCI: tservd.@port[0].v23_tx_timeout Opt: v23_tx_timeout	Defines the V23 modem receive echo suppression timeout in milliseconds. <table border="1"> <tr><td>20</td><td></td></tr> <tr><td colspan="2">Range</td></tr> </table>	20		Range							
20											
Range											
Web: n/a UCI: tservd.@port[0].v23_tx_rampdown Opt: v23_tx_rampdown	Defines the time in milliseconds it takes the V23 transmitter to rampdown carrier from peak to zero. <table border="1"> <tr><td>30</td><td></td></tr> <tr><td colspan="2">Range</td></tr> </table>	30		Range							
30											
Range											
Web: n/a UCI: tservd.@port[0].v23_tx_maxfill Opt: v23_tx_maxfill	Defines the maximum transmit queue fill level in bytes. <table border="1"> <tr><td>127</td><td></td></tr> <tr><td>Range</td><td>0 - 255</td></tr> </table>	127		Range	0 - 255						
127											
Range	0 - 255										
Web: Atmel USB serial card UCI: tservd.@port[0].is_usb_serial Opt: is_usb_serial	This configures the use of tservd with the Atmel USB serial card. If portmode is X21 then it is used in synchronous mode. If port mode is RS232 it is used in asynchronous mode. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.						
0	Disabled.										
1	Enabled.										
Web: Synchronous mode UCI: tservd.@port[0].sync mode Opt: sync mode	Defines synchronous frame mode. Only displayed if Atmel USB serial card is enabled. <table border="1"> <tr><td>hdlc</td><td>HDLC frame mode.</td></tr> <tr><td>transp</td><td>Transparent mode.</td></tr> </table>	hdlc	HDLC frame mode.	transp	Transparent mode.						
hdlc	HDLC frame mode.										
transp	Transparent mode.										
Web: Use CRC32 UCI: tservd.@port[0].sync_crc32 Opt: sync_crc32	Defines whether to use CRC32 or CRC16 in HDLC mode. Only displayed if Atmel USB serial card is enabled. <table border="1"> <tr><td>0</td><td>Use CRC16.</td></tr> <tr><td>1</td><td>Use CRC32.</td></tr> </table>	0	Use CRC16.	1	Use CRC32.						
0	Use CRC16.										
1	Use CRC32.										
Web: DTR control mode UCI: tservd.@port[0].dtr_control_mode Opt: dtr_control_mode	Defines DTR line control modes. Only displayed if Atmel USB serial card is enabled and port mode is X21. <table border="1"> <tr><td>auto</td><td>DTR set to on when port is open. Off when the port is closed.</td></tr> <tr><td>on</td><td>DTR always on.</td></tr> <tr><td>off</td><td>DTR always off.</td></tr> <tr><td>app</td><td>DTR controlled by the application</td></tr> <tr><td>ontx</td><td>In HDLC mode DTR is on during frame transmission.</td></tr> </table>	auto	DTR set to on when port is open. Off when the port is closed.	on	DTR always on.	off	DTR always off.	app	DTR controlled by the application	ontx	In HDLC mode DTR is on during frame transmission.
auto	DTR set to on when port is open. Off when the port is closed.										
on	DTR always on.										
off	DTR always off.										
app	DTR controlled by the application										
ontx	In HDLC mode DTR is on during frame transmission.										

<p>Web: RTS control mode UCI: tservd.@port[0].rts_control_mode Opt: rts_control_mode</p>	<p>Defines RTS line control modes. Only displayed if Atmel USB serial card is enabled and port mode is X21.</p> <table border="1"> <tr><td>auto</td><td>RTS set to on when port is open. Off when the port is closed.</td></tr> <tr><td>on</td><td>RTS always on.</td></tr> <tr><td>off</td><td>RTS always off.</td></tr> <tr><td>app</td><td>RTS controlled by the application.</td></tr> <tr><td>ontx</td><td>In HDLC mode RTS is on during frame transmission.</td></tr> </table>	auto	RTS set to on when port is open. Off when the port is closed.	on	RTS always on.	off	RTS always off.	app	RTS controlled by the application.	ontx	In HDLC mode RTS is on during frame transmission.
auto	RTS set to on when port is open. Off when the port is closed.										
on	RTS always on.										
off	RTS always off.										
app	RTS controlled by the application.										
ontx	In HDLC mode RTS is on during frame transmission.										
<p>Web: Synchronous rate UCI: tservd.@port[0].sync_speed Opt: sync_speed</p>	<p>Defines the synchronous speed in bps. Set to 0 for external clock. If not set to 0 an internal clock is used. Only displayed if Atmel USB serial card is enabled.</p> <table border="1"> <tr><td>64000</td><td>64 kbps</td></tr> <tr><td>Range</td><td>2048000; 1024000; 768000; 512000; 384000; 256000; 128000; 19200; 9600</td></tr> </table>	64000	64 kbps	Range	2048000; 1024000; 768000; 512000; 384000; 256000; 128000; 19200; 9600						
64000	64 kbps										
Range	2048000; 1024000; 768000; 512000; 384000; 256000; 128000; 19200; 9600										
<p>Web: Invert receive clock UCI: tservd.@port[0].sync_invert_rxclk Opt: sync_invert_rxclk</p>	<p>Defines receive clock inversion. Normal clock data is sampled on falling edge. Inverted clock data is sampled on rising edge. Only displayed if Atmel USB serial card is enabled.</p> <table border="1"> <tr><td>0</td><td>Normal</td></tr> <tr><td>1</td><td>Invert</td></tr> </table>	0	Normal	1	Invert						
0	Normal										
1	Invert										
<p>Web: Invert transmit clock UCI: tservd.@port[0].sync_invert_txclk Opt: sync_invert_txclk</p>	<p>Defines transmit clock inversion. Normal clock data transmitted on falling edge. Inverted clock data transmitted on rising edge. Only displayed if Atmel USB serial card is enabled.</p> <table border="1"> <tr><td>0</td><td>Normal</td></tr> <tr><td>1</td><td>Invert</td></tr> </table>	0	Normal	1	Invert						
0	Normal										
1	Invert										
<p>Web: RX MSBF UCI: tservd.@port[0].sync_rx_msbf Opt: sync_rx_msbf</p>	<p>Defines whether most significant bit is received first. Only displayed if Atmel USB serial card is enabled.</p> <table border="1"> <tr><td>0</td><td>Receive least significant bit first.</td></tr> <tr><td>1</td><td>Receive most significant bit first.</td></tr> </table>	0	Receive least significant bit first.	1	Receive most significant bit first.						
0	Receive least significant bit first.										
1	Receive most significant bit first.										
<p>Web: TX MSBF UCI: tservd.@port[0].sync_tx_msbf Opt: sync_tx_msbf</p>	<p>Defines whether most significant bit is transmitted first. Only displayed if Atmel USB serial card is enabled.</p> <table border="1"> <tr><td>0</td><td>Transmit least significant bit first.</td></tr> <tr><td>1</td><td>Transmit most significant bit first.</td></tr> </table>	0	Transmit least significant bit first.	1	Transmit most significant bit first.						
0	Transmit least significant bit first.										
1	Transmit most significant bit first.										
<p>Web: RX data delay UCI: tservd.@port[0].sync_rxdata_dly Opt: sync_rxdata_dly</p>	<p>Defines the number of bit positions to delay sampling data from the detecting clock edge. Only displayed if Atmel USB serial card is enabled.</p> <table border="1"> <tr><td>0</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	0		Range							
0											
Range											
<p>Web: TX data delay UCI: tservd.@port[0].sync_txdata_dly Opt: sync_txdata_dly</p>	<p>Defines the number of bit positions to delay output of data from the detecting clock edge. Only displayed if Atmel USB serial card is enabled.</p> <table border="1"> <tr><td>0</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	0		Range							
0											
Range											
<p>Web: Dual X.21 card bit reverse UCI: tservd.@port[0].bit_reverse Opt: bit_reverse</p>	<p>Enables bit reversal of all bits in 8 byte word during transmission.</p> <table border="1"> <tr><td>0</td><td>Normal.</td></tr> <tr><td>1</td><td>Reverse.</td></tr> </table>	0	Normal.	1	Reverse.						
0	Normal.										
1	Reverse.										
<p>Web: Dual X.21 card DTE TT Invert UCI: tservd.@port[0].dte_tt_inv Opt: dte_tt_inv</p>	<p>Enables X.21 TT clock signal inversion.</p> <table border="1"> <tr><td>0</td><td>Normal.</td></tr> <tr><td>1</td><td>Invert.</td></tr> </table>	0	Normal.	1	Invert.						
0	Normal.										
1	Invert.										
<p>Web: Dual X.21 card DCE TCLK Invert UCI: tservd.@port[0].dce_tclk_inv Opt: dce_tclk_inv</p>	<p>Enables X.21 DCE TCLK signal inversion.</p> <table border="1"> <tr><td>0</td><td>Normal.</td></tr> <tr><td>1</td><td>Invert.</td></tr> </table>	0	Normal.	1	Invert.						
0	Normal.										
1	Invert.										
<p>Web: Dual X.21 card DCE RCLK Invert UCI: tservd.@port[0].dce_rclk_inv Opt: dce_rclk_inv</p>	<p>Enables X.21 DCE RCLK signal inversion.</p> <table border="1"> <tr><td>0</td><td>Normal.</td></tr> <tr><td>1</td><td>Invert.</td></tr> </table>	0	Normal.	1	Invert.						
0	Normal.										
1	Invert.										

Web: Dual X.21 card CLK Invert UCI: tservd.@port[0].x21_clk_invert Opt: x21_clk_invert	Enables X.21 DCE CLK signal inversion. <table border="1"> <tr><td>0</td><td>Normal.</td></tr> <tr><td>1</td><td>Invert.</td></tr> </table>	0	Normal.	1	Invert.				
0	Normal.								
1	Invert.								
Web: Dual X.21 card RX data delay UCI: tservd.@port[0].x21_data_delay Opt: x21_data_delay	Sets X.21 card RX data delay in number of bit positions. <table border="1"> <tr><td>0</td><td></td></tr> <tr><td>Range</td><td>0 – 7</td></tr> </table>	0		Range	0 – 7				
0									
Range	0 – 7								
Web: n/a UCI: tservd.@port[0].sync_tx_idle Opt: sync_tx_idle	Defines the value of idle character (decimal) to transmit in case of transmit underrun. In HDLC mode this configures inter-frame fill. <table border="1"> <tr><td>0</td><td>Transmit 0 (in HDLC mode)</td></tr> <tr><td>126</td><td>Transmit flags (in HDLC mode)</td></tr> <tr><td>255</td><td>Transmit 1 (in HDLC mode)</td></tr> <tr><td>Range</td><td>0 – 255</td></tr> </table>	0	Transmit 0 (in HDLC mode)	126	Transmit flags (in HDLC mode)	255	Transmit 1 (in HDLC mode)	Range	0 – 255
0	Transmit 0 (in HDLC mode)								
126	Transmit flags (in HDLC mode)								
255	Transmit 1 (in HDLC mode)								
Range	0 – 255								
Web: n/a UCI: tservd.@port[0].v23_inband_carrier_sign_alling Opt: v23_inband_carrier_signalling	Enables signalling of carrier by sending special characters. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.				
0	Disabled.								
1	Enabled.								
Web: n/a UCI: tservd.@port[0].v23_inband_carrier_on_char Opt: v23_inband_carrier_on_char	Defines the character decimal to signal remote carrier on. <table border="1"> <tr><td>255</td><td></td></tr> <tr><td>Range</td><td>0 - 255</td></tr> </table>	255		Range	0 - 255				
255									
Range	0 - 255								

Table 143: Information table for port settings serial section

41.3.2.3 Port settings: network section

In this section you can configure the network side of the Terminal Server. Note: the displayed settings vary depending on options selected.

The screenshot shows the configuration interface for PORT1. The 'Network' tab is active. The configuration fields include:

- Transport mode: TCP
- Local IP: 0.0.0.0
- TCP mode: Server
- TCP listen port: 888
- Remote IP 1: 0.0.0.0
- Remote IP 2: 0.0.0.0
- Enable TCP keepalives:
- TCP Keepalive interval (seconds): 5
- TCP Keepalive timeout (seconds): 2
- TCP Keepalive count: 1
- TCP User timeout: 20000
- TCP nodelay:
- TCP always on:
- Close TCP on DSR:
- Reconnect time (ms): 5000

Figure 192: The port settings network fields (TCP server mode)

Web Field/UCI/Package Option	Description	
Web: Transport Mode UCI: tservd.@port[0].udpMode Opt: udpMode	Selects the transport mode.	
	0	TCP
	1	UDP
Web: Local IP UCI: tservd.@port[0].local_ip Opt: local_ip	Local IP address to listen on.	
	0.0.0.0	Listen on any interface.
	Range	IPv4 address.
Web: TCP Mode UCI: tservd.@port[0].server_mode Opt: server_mode	Select between server and client modes of TCP. Only displayed if Transport Mode is TCP.	
	0	Client Mode.
	1	Server Mode.
Web: TCP Listen Port UCI: tservd.@port[0].listen_port Opt: listen_port	TCP listen port for server mode. Only displayed if Transport Mode is TCP and server mode is enabled.	
	999	
	Range	1 - 65535
Web: Remote TCP Port 1 UCI: tservd.@port[0].ip_port1 Opt: ip_port1	Destination peer port IP 1 number. Only displayed if client mode enabled.	
	951	
	Range	1 - 65535
Web: Remote TCP Port 2 UCI: tservd.@port[0].ip_port2 Opt: ip_port2	Destination peer port IP 2 number for failover. Only displayed if client mode enabled.	
	951	
	Range	1 - 65535
Web: Remote IP 1 UCI: tservd.@port[0].remote_ip1 Opt: remote_ip1	Destination peer IP 1 address.	
	0.0.0.0	
	Range	IPv4 address
Web: Remote IP 2 UCI: tservd.@port[0].remote_ip2 Opt: remote_ip2	Destination peer IP 2 address. Only displayed if Transport Mode is TCP.	
	0.0.0.0	
	Range	IPv4 address
Web: Enable TCP Keepalives UCI: tservd.@port[0].tcp_keepalives_enabled Opt: tcp_keepalives_enabled	Enable or disables TCP keepalives. Only displayed if Transport Mode is TCP.	
	0	Disabled.
	1	Enabled.
Web: TCP Keepalive Interval UCI: tservd.@port[0].tcp_keepalive_interval Opt: tcp_keepalive_interval	Interval in seconds between TCP keepalive probes. Only displayed if Transport Mode is TCP.	
	5	5 seconds
	Range	0-65535
Web: TCP Keepalive Timeout UCI: tservd.@port[0].tcp_keepalive_timeout Opt: tcp_keepalive_timeout	Time in seconds to wait for response to a TCP keepalive probe. Only displayed if Transport Mode is TCP.	
	2	2 seconds
	Range	0-65535
Web: TCP Keepalive Count UCI: tservd.@port[0].tcp_keepalive_count Opt: tcp_keepalive_count	Number of TCP keepalive probes to send before connection is closed. Only displayed if Transport Mode is TCP.	
	1	
	Range	0-65535
Web: TCP User Timeout UCI: tservd.@port[0].tcp_user_timeout Opt: tcp_user_timeout	Maximum time in milliseconds for TCP to wait for transmitted data to be acked before closing connection in established state. Set to 0 to use kernel defaults. Only displayed if Transport Mode is TCP.	
	20000	20 seconds
	Range	0-65535

Web: TCP Nodelay UCI: tservd.@port[0].tcp_nodelay Opt: tcp_nodelay	Sets TCP to delay behaviour. Only displayed if Transport Mode is TCP. <table border="1"> <tr><td>0</td><td>Normal operation.</td></tr> <tr><td>1</td><td>Disable TCP Nagle algorithm. Only displayed if Transport Mode is TCP.</td></tr> </table>	0	Normal operation.	1	Disable TCP Nagle algorithm. Only displayed if Transport Mode is TCP.
0	Normal operation.				
1	Disable TCP Nagle algorithm. Only displayed if Transport Mode is TCP.				
Web: TCP Always on UCI: tservd.@port[0].tcp_always_on Opt: tcp_always_on	Keep TCP session always connected. Only displayed if Transport Mode is TCP and client mode is enabled. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Close TCP on DSR UCI: tservd.@port[0].close_tcp_on_dsr Opt: close_tcp_on_dsr	Close TCP session on detection of DSR signal low. Only displayed if Transport Mode is TCP and client mode is enabled. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>1</td><td>Enabled.</td></tr> </table>	0	Disabled.	1	Enabled.
0	Disabled.				
1	Enabled.				
Web: Reconnect Time (ms) UCI: tservd.@port[0].disc_time_ms Opt: disc_time_ms	Time in milliseconds to start reconnecting after setting DTR low. <table border="1"> <tr><td>5000</td><td>5 seconds.</td></tr> <tr><td>Range</td><td>0 – 10000</td></tr> </table>	5000	5 seconds.	Range	0 – 10000
5000	5 seconds.				
Range	0 – 10000				
Web: UDP Keepalive Interval UCI: tservd.@port[0].udpKaIntervalMs Opt: udpKaIntervalMs	Defines time in milliseconds to send UDP keepalives (empty UDP packets) when no data to send. Only displayed if transport mode is UDP. <table border="1"> <tr><td>0</td><td>Disabled.</td></tr> <tr><td>Range</td><td>0-65535</td></tr> </table>	0	Disabled.	Range	0-65535
0	Disabled.				
Range	0-65535				
Web: UDP Keepalive Count UCI: tservd.@port[0].udpKaCount Opt: udpKaCount	Defines the maximum number of remote UDP keepalive not received before UDP stream is considered broken. Only displayed if transport mode is UDP. <table border="1"> <tr><td>3</td><td></td></tr> <tr><td>Range</td><td>0-65535</td></tr> </table>	3		Range	0-65535
3					
Range	0-65535				
Web: local UDP Port UCI: tservd.@port[0].udpLocalPort Opt: udpLocalPort	Local UDP port used by terminal server. Only displayed if transport mode is UDP. <table border="1"> <tr><td>0</td><td></td></tr> <tr><td>Range</td><td></td></tr> </table>	0		Range	
0					
Range					
Web: remote UDP Port UCI: tservd.@port[0].udpRemotePort Opt: udpRemotePort	Remote UDP port used by terminal server. Only displayed if transport mode is UDP. <table border="1"> <tr><td>0</td><td></td></tr> <tr><td>Range</td><td>0-65535</td></tr> </table>	0		Range	0-65535
0					
Range	0-65535				

Table 144: Information table for port settings network section

41.4 Terminal Server using UCI

```
root@GW_router:~# uci show tservd
tservd.main=tservd
tservd.main.log_severity=0
tservd.main.debug_rx_tx_enable=1
tservd.main.debug_ev_enable=1
tservd.@port[0]=port
tservd.@port[0].devName=/dev/ttySC0
tservd.@port[0].remote_ip1=0.0.0.0
tservd.@port[0].remote_ip2=0.0.0.0
```

41.5 Terminal Server using package options

```
root@GW_router:~# uci export tservd
package tservd

config tservd 'main'
    option log_severity '0'
    option debug_rx_tx_enable '1'
    option debug_ev_enable '1'

config port
    option devName '/dev/ttysC0'
    option remote_ip1 '0.0.0.0'
    option remote_ip2 '0.0.0.0'
```

41.6 Terminal Server diagnostics

The tservd process has to be running otherwise diagnostics options for terminal server will not be available.

41.6.1 Checking Terminal Server process

To check if Terminal Server is running, enter:

```
root@GW_router:~# ps | grep tservd
1264 root      1032 S  tservd
1769 root      1496 S  grep tservd
```

If Terminal Server is running it will be shown with its process ID.

41.6.2 Terminal Server statistics

To see the terminal server statistics, enter:

```
root@GW_router:~# tserv show stats
TERMINAL 1, Dev: /dev/ttysC0
State:          LISTENING
Serial Bytes   Rx (0)   Tx (0)   TxErrs (0)
TCP Packets    Rx (0)   Tx (0)   TxErrs (0)       TxBlocked (0)
TCP Bytes      Rx (0)   Tx (0)
UDP Datagrams  Rx (0)   Tx (0)   TxErrs (0)
UDP Bytes      Rx (0)   Tx (0)
DSR            Up (0)   Down (0)
```

41.6.3 Terminal Server debug statistics

To see debug statistics about the terminal server, enter:

```
root@GW_router:~# tserv show debug all

TERMINAL 1, Dev: /dev/ttysC0
State: LISTENING
netRxBuf length=0 offset=0 hdrsz=0
ttyRxBuf length=0 offset=16 hdrsz=16
line_status_mask = 0x0 line_status = 0x0
RFC2217 negotiated=0
Tcp tx last error: 0
```

41.6.4 Terminal Server advanced debugging

To see advanced debug commands for the terminal server, enter:

```
root@GW_router:~# tserv
== Termserv diagnostics. Command syntax: ==
tserv show stats - show statistics
tserv clear stats - clear statistics
tserv show serial - show serial interface status
tserv send serial0 <data>- send data to serial port 0
tserv start capture N, N=port number (0 to 3) - start capturing rx serial
data
tserv print capture N, N=port number (0 to 3) - print captured rx serial
data
tserv show serial txlog-hex <Port> [length], Port=port cfg index (0 to 3),
length=length to show
tserv show serial rxlog-hex <Port> [length], Port=port cfg index (0 to 3),
length=length to show
tserv show serial txlog-asc <Port> [length], Port=port cfg index (0 to 3),
length=length to show
tserv show serial rxlog-asc <Port> [length], Port=port cfg index (0 to 3),
length=length to show
tserv show debug - show debug info
tserv show userial stats - show USB serial card statistics
tserv clear userial stats - clear USB serial card statistics
tserv start userial rxlog - start USB serial card rx log
tserv show userial rxlog <offs> <length> - show USB serial card rx log
```

```
tserv show userial version - show USB serial card firmware version  
tserv show userial cpld status - show USB serial card CPLD programming  
status  
tserv upgrade userial - initiate upgrade of the USB serial card  
tserv quit - terminate termserv process
```

You can configure your router using either the router's web interface or via the command