RedFox Series
Wolverine Series
Lynx+ Series
6101-3201

Westermo OS Management Guide

WeOS

RedFox

Wolverine

Lynx+

www.westermo.com

Legal information

The contents of this document are provided ”as is”. Except as required by applicable law, no warranties of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose, are made in relation to the accuracy and reliability or contents of this document. Westermo reserves the right to revise this document or withdraw it at any time without prior notice.

Under no circumstances shall Westermo be responsible for any loss of data or income or any special, incidental, and consequential or indirect damages howsoever caused. More information about Westermo can be found at the following Internet address:

http://www.westermo.com

Contents

Chapter 1

Introduction

This guide describes the functionality and the management features of the Westermo Operating System (WeOS). WeOS is the *firmware* controlling the operation on the following series of Westermo switches:

- RedFox Industrial
- RedFox Rail
- Wolverine (DDW-225 and DDW-226)
- Lynx+
- Lynx 1400G

WeOS will be available for an increasing set of Westermo products in the future.

WeOS delivers an extensive set of functionality including layer-2 (basic switching, VLAN, IGMP snooping, etc.), layer-3 (routing, firewall, NAT, etc.), and higher-level services (DHCP, DNS, etc.). Furthermore, WeOS provides easy management via a Web interface, via the Westermo IPConfig tool, and via a USB stick. To satisfy even more advanced customer needs, WeOS provides flexible management via a command line interface (CLI), as well as via SNMP.

1.1 Getting Started

The dedicated *User Guide*[8, 9, 10, 11] of your product includes information on how to get started with WeOS on your specific product. That is the ideal place to

start if you wish to do the least possible configuration of your switch (i.e., assign appropriate IP settings) before putting it into your network infrastructure.

If the user guide of your specific product lacks a section on how to get started with WeOS, please visit the chapter 2 (*Quick Start*) of this document.

1.2 Where to find more information

At http://www.westermo.com you can find the latest updated version of this document - the *WeOS management guide*. There you can also find, application notes, user guides, and other support information for your product.

1.3 How to read this document

This remainder of this guide is structured in the following parts:

- Chapters 3-6 introduces the three main methods to manage a switch running WeOS (IPConfig, Web and CLI). If you need recommendations of which method to use, please read chapter 3. To find in-depth information of how to use the Westermo IPConfig tool, see chapter 4. Chapters 5 and 6 contain general information on how manage the switch via the Web and the CLI respectively.
- Chapters 7-28 hold in-depth information on how to manage your switch. Each chapter contains information related to a specific area of functionality. For example, if you wish to configure virtual LANs (VLANs), chapter 13 is where you should look (see the table of contents for information on what topic is treated in the different chapters).

  Each of these chapters starts with a section providing a *general feature description*. This is followed by sections on how to manage the features via the Web and via the CLI.
- Chapter 29 contains information on how to manage the switch using SNMP.

1.4 Differences between products running WeOS

1.4.1 Hardware differences affecting WeOS functionality

The WeOS functionality described in the management guide generally applies to all Westermo products running WeOS. However, where functionality assumes the

presence of certain hardware (such as a USB port), those functions are limited to products including that hardware. The table below provides a summary of hardware differences affecting the availability of certain WeOS functions. For a more definite description of hardware specifications you are referred to the dedicated *User Guide* of each product[8, 9, 10, 11].

	RedFox Industrial	RedFox Rail	DDW-225	DDW-226	Lynx+	Lynx 1400G
Ethernet ports	X	X	X	X	X	X
DSL ports			X	X		
Serial ports				X		
Console port	X		X	X	X	
Digital In/Out	X		X	X	X	X
USB Port	X	X	X	X		

1.4.2 Port naming conventions

The convention to name communication ports such as Ethernet ports and DSL ports differs between Westermo products. RedFox Rail and Lynx 1400G use a simple *port ID* to refer to the ports.

- *RedFox Rail:* Ethernet ports on RedFox Rail are named *X1*, *X2*, *X3*, . . .
- *Lynx+ and Lynx 1400G:* Ethernet ports on Lynx+ and Lynx 1400G are named *1*, *2*, *3*, . . .

RedFox and Wolverine: RedFox and Wolverine use a slotted architecture, and ports are named according to the *slot ID* and the *port's position* within that slot. For example, port *1/2* would denote the second port in the first slot.

This name convention is used irrespective of port type, e.g., DDW-225 (Wolverine) has two SHDSL ports (1/1-1/2) and 4 Ethernet ports (2/1-2/4). Details on the name convention and the slotted architecture on RedFox is described further below.

The *RedFox Industrial* switches come in a two-slot and a three-slot version. Figure 1.1 shows a sample three-slot RedFox Industrial equipped with a 4-port Gigabit/SFP card (middle slot) and an 8-port 10/100BaseTX card (right slot). The leftmost slot contains the Power/CPU card, which is present on all RedFox Industrial switches.

RedFox Industrial makes use of a slotted architecture with different combinations of interface modules. As mentioned above WeOS numbers the ports based on *slotID/portID*, where the

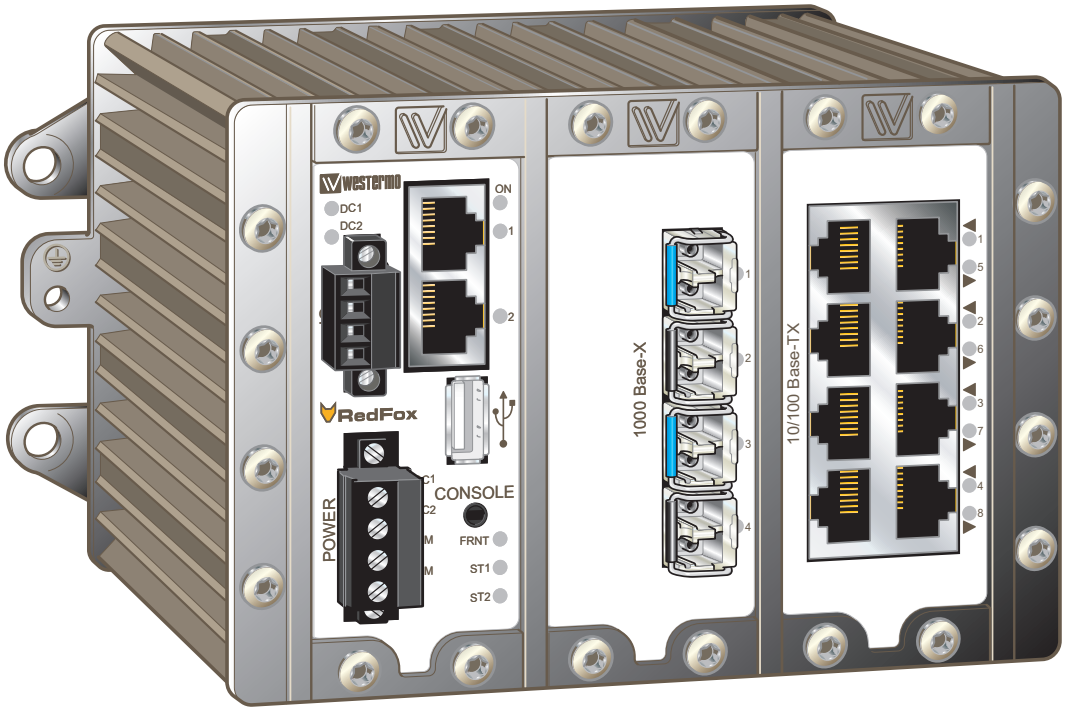


Figure 1.1: Three-slot RedFox Industrial equipped with a 4-port Gigabit/SFP card (middle slot), and an 8-port 10/100BaseTX card (right slot).

- the *slotID* denotes the slot's position within the rack (left to right), and
- the *portID* denotes the port's position within the slot (left to right, up to down).

For example, the two Ethernet ports in the leftmost slot (slot 1) are named *1/1* (top) and *1/2* (bottom). The ports in the second slot are named *2/1-2/4*, and the ports in slot 3 are named *3/1-3/4* (left side) and *3/5-3/8* (right side).

1.4.3 Factory default settings

Factory default settings may vary between products. Information on the factory default settings is provided in the *Getting Started* section of the dedicated *User Guide* of each product.

Chapter 2

Quick Start

This section provides a guide to quickly get started with your switch. Only simple configuration procedures will be covered[1] The steps covered concern:

- Get familiar with the factory default setting
- Configuring an appropriate IP address

2.1 Starting the Switch for the First Time - Factory Default Setting

When booting the switch for the first time the switch will use the factory default setting. The factory default setting makes the switch operate as a manageable layer-2 switch, where all Ethernet ports belong to the same virtual LAN (VLAN).

- Manageable: The switch is manageable via any of the Ethernet ports. To manage the switch via an Ethernet port you need to know the IP address of the switch (see table 2.1). For switches equipped with a console port, the switch can as well be managed via that port without knowing the IP address of the switch.
- Single VLAN: By default all ports on the switch will belong to the same VLAN. Thus, devices connected to different ports of the switch should be able to communicate with each other right away. For more advanced setups, the ports of the switch can be grouped into different VLANs. In the factory default setting all ports belong to VLAN 1.

[1] For more advanced settings, we refer to the remaining chapters of this guide as well as the online help provided via the Web configuration tool and the Command Line Interface (CLI).

The default IP setting for the switch is as shown in table 2.1. Before you put your switch into your network infrastructure you should change its IP setting according to your network topology.

IP Parameter	Default Setting
IP address	192.168.2.200
Netmask	255.255.255.0
Default gateway	Disabled

Table 2.1: Default IP settings.

2.2 Modifying the IP Setting

The switch can be configured with a static IP setting, or it can get its IP address dynamically via DHCP. The latter case is useful if you are running a DHCP server on the same LAN as the switch will be located.

WeOS provides several management tools, which will be presented further in later chapters of this guide. In this chapter we limit the scope to describe how these tools can be used to update the IP settings of the switch.

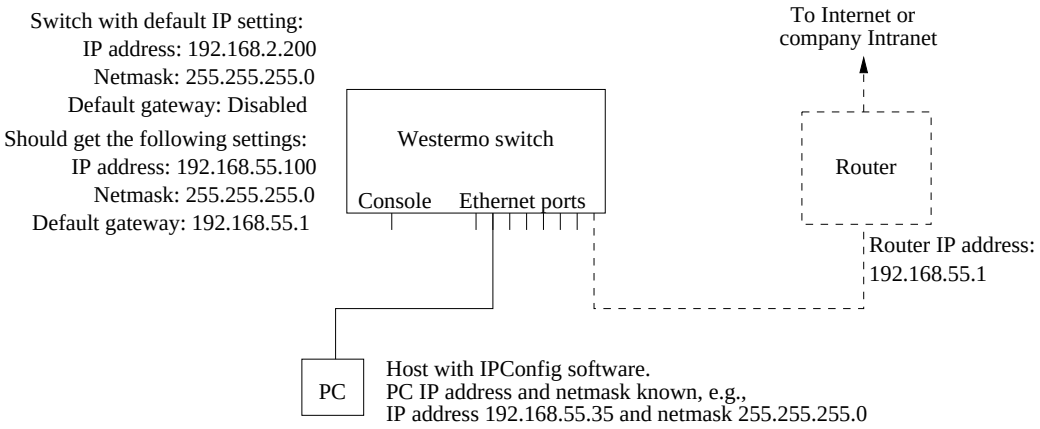
- *IPConfig*: IPConfig is a custom Westermo tool used for *discovery* of attached Westermo switches, and for various management tasks. Configuration of IP settings via IPConfig is described in section 2.2.1.
- *Web*: Configuration of IP settings via the Web interface is described in section 2.2.2.
- *CLI*: Configuration of IP settings via the Command Line Interface (CLI) is described in section 2.2.3.

Hint: *If you are not sure what IP address your switch has, use the* IPConfig *method (section 2.2.1) or the* CLI via console *method (section 2.2.3.1). If neither of these methods work, please visit section 7.1.4 for information on how to conduct a factory reset.*

2.2.1 Using the IPConfig tool to Update the Switch IP Settings

1. *Installation*: To use the IPConfig tool to scan for switches and manage their IP address setting, you first need to install the IPConfig software on a PC with Microsoft Windows™ (2000/XP) operating system. You can find the IPConfig software on the software CD bundled with your switch, or download it online from http://www.westermo.com.

2. *Connect your PC to the switch*: IPConfig can only scan for Westermo switches attached to the same LAN. The figure below shows a simple setup where the PC is attached directly to an Ethernet port of a Westermo switch.

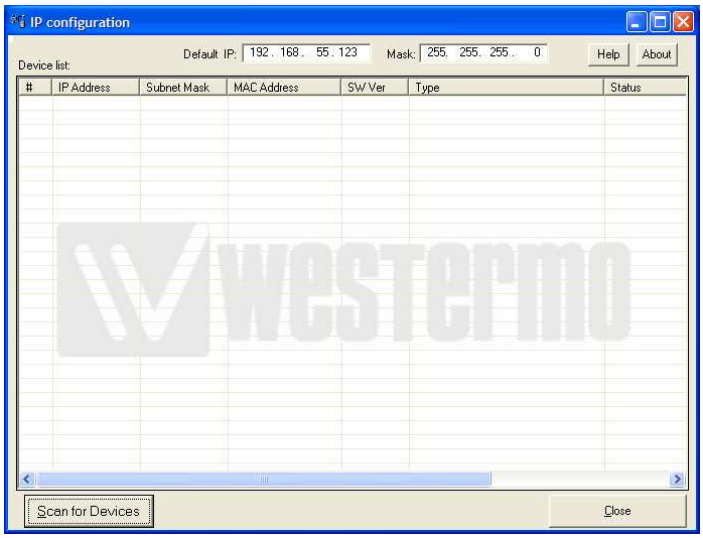


3. *Determine the PC's IP address and subnet mask*: To scan for switches with IPConfig, you need to know the PC's IP address and netmask. You also need to know a *free* IP address on that IP subnet. Ask your system administrator if you need help with this. In the example network shown in step 2 the answer would be as follows:

   - PC IP address: 192.168.55.35
   - PC netmask: 255.255.255.0
   - Free IP address on that subnet: 192.168.55.123 (just an example)

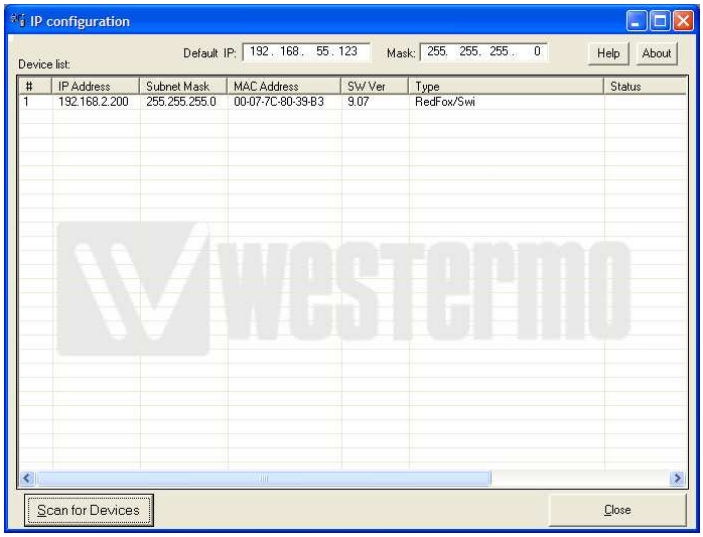
   In this example, IP address 192.168.55.123 will be used in step 5 below.

4. *Start the IPConfig tool*: You should then see a window similar to what is shown below.

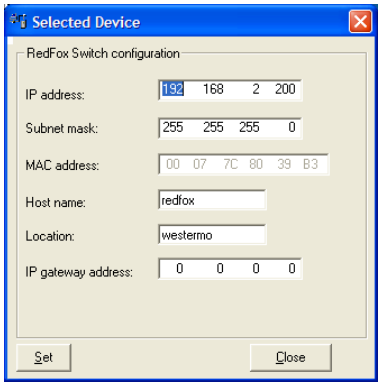


5. *Fill in IP address and netmask*: In the startup window, fill in the *free* IP address (see step 3) in the **`Default IP`** text field, and your PC's netmask in the *Mask* text field.

6. Scan for devices: Press the **`Scan for devices`** button to scan for Westermo devices supporting IPConfig. The devices in the locally broadcast domain should appear as shown below. Each one detected is displayed as a row in the IPConfig Startup window. For each device information such as base MAC and LAN IP address is shown.



7. *Change the IP addresses of a switch*: To change the IP address setting of a switch, double-click on the row for that switch in the IPConfig Startup window. A configuration window for the selected device should appear as shown below. To change the IP settings (**`IP address`**, **`Subnetmask`** and **`IP gateway address`**) update the text fields appropriate for your network setup and press the **`Set`** button.



   Example: In our sample setup (see the figure in step 2 we may like to give the switch an IP address on the same subnet as our PC. This is just an example - consult your system administrator if you do not know what IP address to assign.

   - *IP address*: 192.168.55.100
   - *Netmask*: 255.255.255.0
   - *Default Gateway*: 192.168.55.1

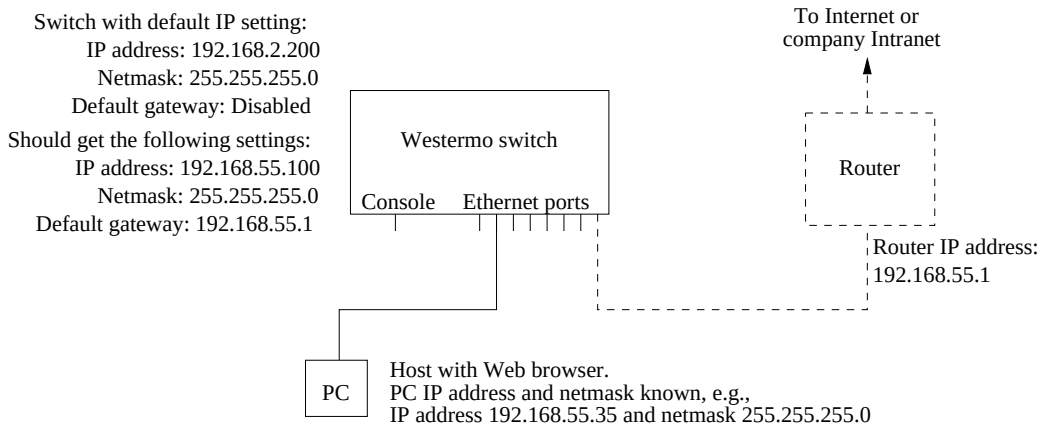
   After filling in all the configuration settings, click the **`Set`** button.

8. *Done!* You have now updated the IP settings of the switch to a configuration suitable for your specific network setup.

Further management of the switch can be performed via any of the available management tools - IPConfig, Web, SSH/CLI or SNMP.

2.2.2 Using the Web Interface to Update the Switch IP Settings

To configure the IP settings via web your switch is required to be located on the same IP subnet as your PC.

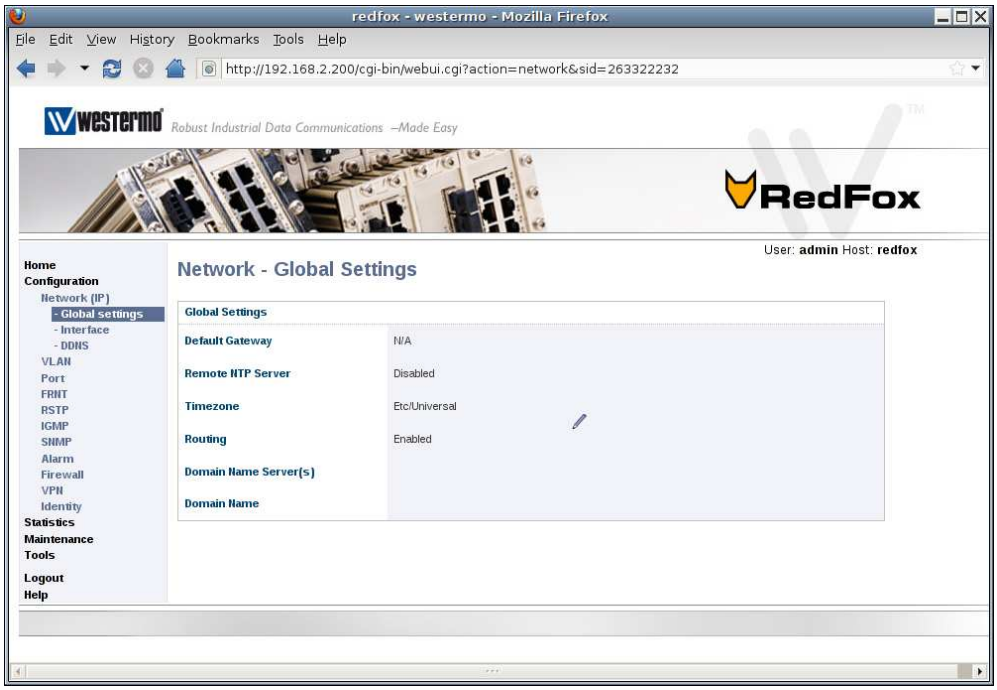


In this *example* the switch shall be assigned the IP address 192.168.55.100, netmask 255.255.255.0 and default gateway 192.168.55.1. To achieve this you must (temporarily) change the IP address of the PC in order to be able to communicate with the switch.

The steps to configure the IP settings via the web interface are as follows:

1. *Connect your PC to the switch*: Connect your PC to the switch as shown in the figure above.
2. *Modifying IP Settings on PC*: The IP settings on the PC must be updated to match the default settings on the switch, i.e., the PC should be assigned an IP address on the 192.168.2.0/24 network, e.g.,
   - PC IP address: 192.168.2.1
   - PC Netmask: 255.255.255.0
3. *Access switch via web browser*: Open your web browser and enter URL **`http://192.168.2.200`** in the browser's address field. You will be asked to enter a *username* and a *password*. Use the the factory default account settings shown below:
   - Login username: **admin**
   - Password: **westermo**

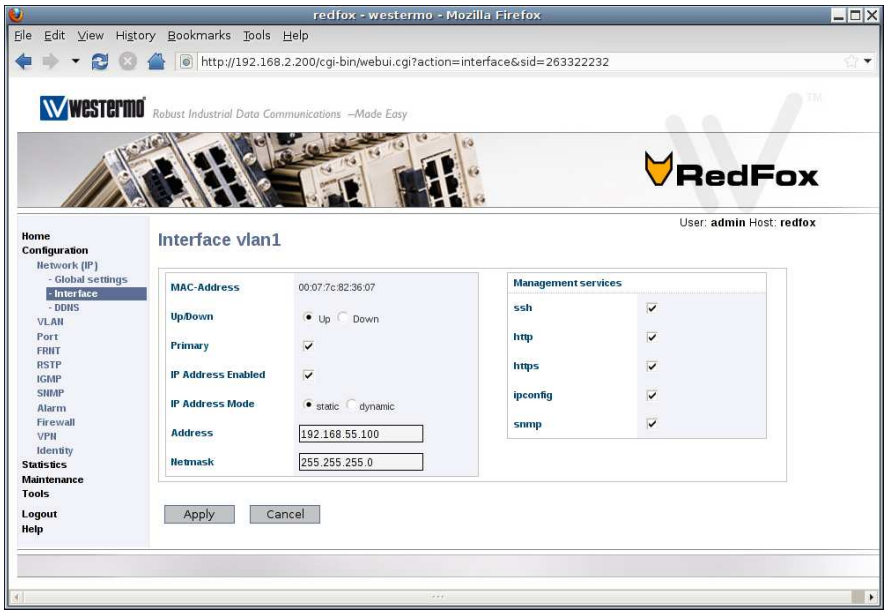
4. *Open the Network(IP) configuration page*: Click on the **`Configuration`** top-menu and then on the **`Network (IP)`** sub-menu and then the **`Global settings`** menu.



5. *Configure Default Gateway*: Now click the edit icon (✎) in the **`Global Settings`** frame. The following page should appear.

Fill in the appropriate address in the **`Default Gateway`** field. In this example, the default gateway is 192.168.55.1. Click the **`Apply`** button. Your switch is configured with a new default gateway.

6. *Open Interface Configuration Page*: Click on the **Configuration** top-menu and then on the **Network (IP)** sub-menu and then the **Interface** sub menu. In the **Interface** page, click the *edit* icon (✎) on the row for the interface named **vlan1** . The *Interface Configuration Page* will appear:



7. *Configure Interface IP Settings*: Enter the appropriate IP settings for your switch. In this example we fill in **192.168.55.100** in the **IP address** field, and keep **255.255.255.0** in the **Netmask** field.

   Click the **Apply** button and your switch is configured with a new IP address.

8. *Reconfigure PC's IP Settings*: As the IP address is changed on the switch, you cannot reach it from your PC any longer. To access the switch from the PC, the PC's IP settings must be changed again. In this case, we assume it is changed back to its original settings:

   - PC IP address: 192.168.55.35
   - PC Netmask: 255.255.255.0
   - PC Default Gateway: 192.168.55.1

Further management of the switch can be performed via any of the available management tools - IPConfig, Web, SSH/CLI or SNMP.

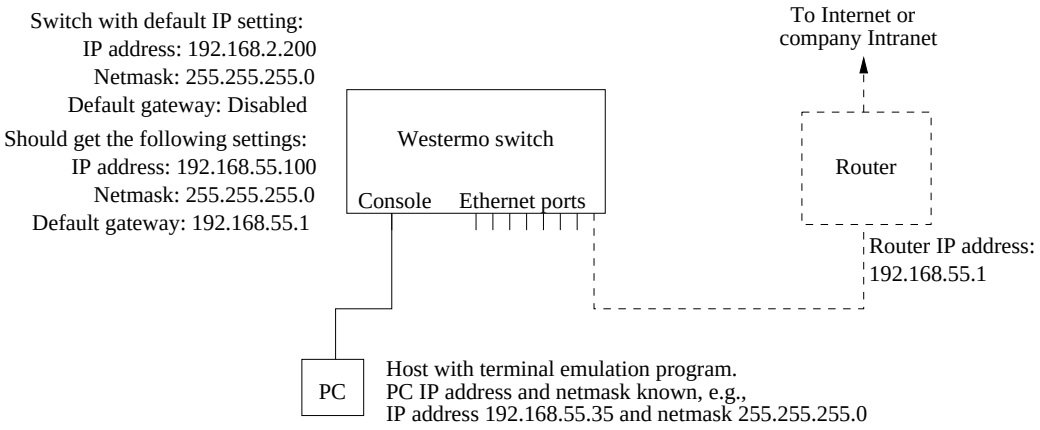
2.2.3 Using the CLI to Update the Switch IP Settings

The CLI can be accessed in two ways: via the console port (given that the switch is equipped with a console port) or via the Ethernet ports using the Secure Shell (SSH) protocol. Section 2.2.3.1 explains how to access the CLI via the console port, and how to update the IP settings. Section 2.2.3.2 explains how to access the CLI via SSH.

2.2.3.1 Accessing the CLI via the console port

For Westermo switches equipped with a console port, this port can be used to change the switch IP address.

1. *Connect your PC to the switch*: Connect your PC to the switch as shown in the figure below.



> **Important notice for Westermo Switches equipped with a console port**: *To access the console port on Westermo Switches equipped with a 2.5 mm jack console port (see, e.g., fig. 1.1), the Westermo Diagnostic Cable, 1211-2027, must be used to connect a USB port on your PC to the switch console port. See the User Guide of your specific product for more information[8, 9, 10, 11].*

2. *Terminal program*: To communicate with the switch via the console port, you need to use a terminal emulation program on your PC, such as *Hyperterminal*. Ask your system administrator if you need help to *install* or *configure* your terminal emulation program.

On Wolverine and RedFox Industrial the following console port settings are used:

Console Port Parameter	Setting
Data rate	115200 bits/s
Data bits	8
Stop bits	1
Parity	Off
Flow control	Off

3. *Activating the console*: When the switch has finished booting, you will be asked to press the **Enter** key on your keyboard to activate the console.

4. *Logging in*: Now you will be asked to enter a *username* and thereafter a *password*. For a switch using the factory default settings, use the following login username and password:

   - Login username: **admin**
   - Password: **westermo**

   Below you see a sample printout when logging in on a RedFox Industrial switch. (The password is not ”echoed” back to the screen.)

```
redfox login:  admin
Password:
.--.--.--.-----.-----.------.-----.-.--.--------.-----.
|  |  |  |  -__|__ --|_    _|  -__|   _|  .  .  |  _  | http://www.westermo.com
 \__/\__/|_____|_____| |__| |_____|__| |__|__|__|_____|        info@westermo.se
   Robust Industrial Data Communications -- Made Easy

Westermo/RedFox Version 9.99 cricket/trunk@15038 -- Oct 16 07:12 CEST 2009
redfox:/#>
```

5. *Listing IP address*: Use the CLI command **”show ifaces”** to list information about network interfaces.

```
redfox:/#>  show ifaces

Interface Name    Oper  IP Address       Netmask          MAC Address (auto)
----------------  ----  ---------------  ---------------  --------------------
vlan1             UP    192.168.2.200    255.255.255.0    00:07:7c:82:2f:c7
-------------------------------------------------------------------------------

redfox:/#>
```

6. *Changing IP address and netmask*: To change the switch IP address and netmask, use CLI commands **”configure”**, **”iface vlan1 inet static”**,

”address <IPV4ADDRESS/LEN>” and **”end”** as shown below. This example is based on the setup in step 1, and configures the switch with an address (192.168.55.100/24) on the same IP subnet as the PC. (Prefix length ’/24’ corresponds to netmask 255.255.255.0 - ask your system administrator if you need help to find out the prefix length of your IP subnet.)

```
redfox:/#> configure
redfox:/config/#> iface vlan1 inet static
redfox:/config/iface-vlan1/#> address 192.168.55.100/24
redfox:/config/iface-vlan1/#> end
redfox:/config/#> end
redfox:/#> show ifaces

Interface Name    Oper  IP Address       Netmask          MAC Address (auto)
---------------   ----  ---------------  ---------------  --------------------
vlan1             UP    192.168.55.100   255.255.255.0    00:07:7c:82:2f:c7
-------------------------------------------------------------------------------

redfox:/#>
```

7. *Set default gateway IP address*: The figure below shows the same network setup, but with a router attached to the IP subnet.

   With this setup you would like to configure a *default gateway* IP address to allow management of the switch from outside the local network. This can be achieved using CLI commands **”configure”**, **”ip”**, **”default-gateway <IPADDRESS>”**, and **”end”** as shown below.

   ```
   redfox:/#> configure
   redfox:/config/#> ip
   redfox:/config/ip/#> default-gateway 192.168.55.1
   redfox:/config/ip/#> end
   redfox:/config/#> end
   redfox:/#>
   ```

8. *Save configuration*: Although the configuration changes has been activated, the running configuration must be stored to the startup configuration. Otherwise the changes will be lost if the switch is rebooted.

   ```
   redfox:/#> copy running-config startup-config
   redfox:/#>
   ```

9. You are now done setting the IP address, subnet mask and default gateway of your switch. Logout from the CLI using the **”logout”** command.

Further management of the switch can be performed via any of the available management tools - IPConfig, Web, SSH/CLI or SNMP.

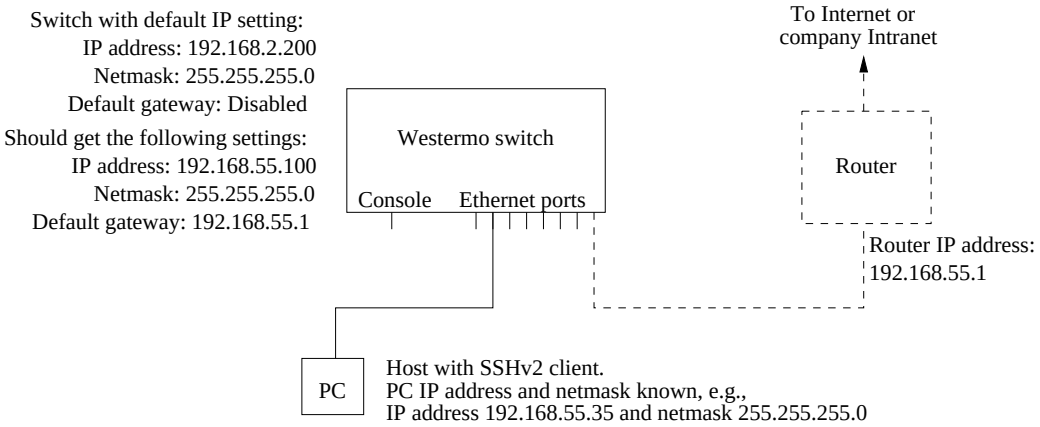
2.2.3.2 Accessing the CLI via SSH

Configuring the IP settings via SSH/CLI is very similar to configuring them via the console port. The major differences are:

- The IP address of the PC must (temporarily) be changed in order to be able to communicate with the switch, i.e., the PC should have an address on network 192.168.2.0/24, e.g., 192.168.2.1/24.
- After the IP settings have been changed on the switch, the PC is likely to loose contact with the switch. The PC must therefore change its IP address again, and login to the switch again in order to copy the running configuration to the startup configuration.

The steps to configure the IP settings via SSH/CLI are as follows:

1. *Connect your PC to the switch*: Connect your PC to the switch as shown in the figure below. In this example we assume the switch will get IP address 192.168.55.100, netmask 255.255.255.0 and default gateway 192.168.55.1.



2. *Modifying IP Settings on PC*: The IP settings on the PC must be updated to match the default settings on the switch, i.e., the PC should be assigned an IP address on the 192.168.2.0/24 network, e.g.,
   - PC IP address: 192.168.2.1
   - PC Netmask: 255.255.255.0
   - PC Default Gateway: Not needed
3. *Connecting and Logging in*: When connecting via SSH you will be asked to enter a *username* and thereafter a *password*. For a switch using the factory default settings, use the following login username and password:

- Login username: **admin**
- Password: **westermo**

The procedure to connect may vary slightly depending on what SSH client you are using. The example below show the connection procedure using Unix OpenSSH[2]. (On Windows one can use Putty[3].)

```
user@pc:~\$ ssh admin@192.168.2.200
The authenticity of host '192.168.2.200 (192.168.2.200)' can't be established.
RSA key fingerprint is 9b:28:d8:7e:c0:9b:68:67:41:9a:1a:5e:b0:92:ff:f4.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.2.200' (RSA) to the list of known hosts.
admin@192.168.2.200's password:
.--.--.--.-----.-----.------.-----.-.--.--------.-----.
|  |  |  |  -__|__ --|_    _|  -__|   _|  .  .  |  _  | http://www.westermo.com
 \__/\__/|_____|_____| |__| |_____|__| |__|__|__|_____|        info@westermo.se
   Robust Industrial Data Communications -- Made Easy

Westermo/RedFox Version 9.99 cricket/trunk@15038 -- Oct 16 07:12 CEST 2009
redfox:/#>
```

4. *Changing IP settings*: The switch IP settings are changed with the same commands as described when accessing the CLI via the console port (section 2.2.3.1). In this example we assign IP address, netmask and default gateway.

```
redfox:/#> configure
redfox:/config/#> iface vlan1 inet static
redfox:/config/iface-vlan1/#> address 192.168.55.100/24
redfox:/config/iface-vlan1/#> end
redfox:/config/#> ip
redfox:/config/ip/#> default-gateway 192.168.55.1
redfox:/config/ip/#> end
redfox:/config/#> end
```

The configuration is now changed, but not yet saved to the startup configuration. However, as the IP address is changed, the SSH connection will be broken.

5. *Logging in again to save configuration*: To login again, the PC's IP settings must be changed again. In this case, we assume it is changed back to its original settings:

- PC IP address: 192.168.55.35
- PC Netmask: 255.255.255.0
- PC Default Gateway: 192.168.55.1

[2] OpenSSH, http://www.openssh.com
[3] Putty, http://www.chiark.greenend.org.uk/~sgtatham/putty/

We can then login again to copy the running configuration to startup configuration.

```
user@pc:~\$  ssh admin@192.168.55.100
The authenticity of host '192.168.55.100 (192.168.55.100)' can't be established.
RSA key fingerprint is 9b:28:d8:7e:c0:9b:68:67:41:9a:1a:5e:b0:92:ff:f4.
Are you sure you want to continue connecting (yes/no)?  yes
Warning: Permanently added '192.168.55.100' (RSA) to the list of known hosts.
admin@192.168.55.100's password:
.--.--.--.-----.-----.-----.-----.-.--.--------.-----.
|  |  |  |  -__|__ --|_    _|  -__|   _|  .  .  |  _  | http://www.westermo.com
 \__/\__/|_____|_____| |__| |_____|__| |__|__|__|_____|        info@westermo.se
   Robust Industrial Data Communications -- Made Easy

Westermo/RedFox Version 9.99 cricket/trunk@15038 -- Oct 16 07:12 CEST 2009
redfox:/#>
redfox:/#>  copy running-config startup-config
redfox:/#>
```

You are now done setting the IP address, subnet mask and default gateway of your switch. Logout from the CLI using the **"logout"** command.

Further management of the switch can be performed via any of the available management tools - IPConfig, Web, SSH/CLI or SNMP.

Chapter 3

Management Tools

WeOS is managed and monitored using the following tools and interfaces:

- **IPConfig:** IPConfig is a custom Westermo tool for management of basic device settings. The IPConfig tool is typically used for discovery of attached Westermo switches, and when setting the device IP address for the first time.
- **Web:** The WeOS Web interface provides management of essential features. The Web interface should satisfy the needs of all common use cases.
- **CLI:** The WeOS Command Line Interface is an industry standard CLI, and provides the most complete management support. The CLI is intended for advanced users requiring fine grain control of the system.

In addition, WeOS provides device management via SNMP (v1/v2c/v3). A set of standard MIBs and the WeOS private MIB are supported, as described in chapter 29.

Task	IPConfig	Web	CLI	SNMP
Discover Westermo Devices	X	(X)	(X)	
Set Device IP Address	X	X	X	X
Upgrade primary firmware	X	X	X	
Common management tasks		X	X	X
All management tasks			X	
Secure management		X	X	X

3.1 Selecting a Management tool

In the following sections the properties of the IPConfig tool, the Web Interface, and the CLI are presented further. These sections give information about what management tool to use for a specific need. For more information on SNMP we refer to chapter 29.

3.1.1 When to use the IPConfig Tool

The IPConfig tool can be used to manage Westermo switches from a PC attached to the same LAN. IPConfig is suitable to use in the following situations:

- Discover Westermo switches: With IPConfig you can discover Westermo switches attached to the local LAN.
- Determine the IP address of Westermo switches: The IPConfig tool can be used to determine the IP address of switches attached to the same LAN. Once you know the IP address of a switch, you could go on managing that switch via your preferred management interface (Web, CLI, SNMP).
- Set the IP address of a switch: With IPConfig you could change the IP address, the IP netmask and the default gateway IP address. Once you have given your switch suitable IP settings, you could go on managing that switch *remotely* via your preferred management interface (Web, CLI, SNMP).
- Upgrade switch firmware: It is possible to upgrade the (primary) firmware of the switch via IPConfig.

The IPConfig tool can be used to discover and manage different types of Westermo switches, both those switches running WeOS and some which do not.

Chapter 4 further describes IPConfig management capabilities on switches running WeOS.

3.1.2 When to use the Web Management Tool

The Web interface would be the management interface of choice for most users. The main advantages of the Web Interface are:

- *Easy to use*: The Web management interface provides an *easy to use* method to manage the switch.
- *All common features*: The web interface includes support for all essential management features, and should therefore meet the needs of most users.

- *Secure management:* The web interface can be accessed via regular HTTP and secure HTTP (HTTPS). Secure management is also possible via the CLI (SSHv2) and and SNMP (SNMPv3).
- *Discover other Westermo Switches*: The Web contains a discovery service similar to what IPConfig provides. (Note, you must still be able to login to one switch in order to make use of this service.)

To use the Web interface, you must know the IP address of your switch. To find out the switch IP address you may need to use the Westermo IPConfig tool[1], but once you know it you can do the rest of the management via the Web interface.

The Web interface is introduced in chapter 5.

3.1.3 When to use the Command Line Tool

The WeOS CLI aims to serve advanced users. Furthermore, the CLI is the only management tool which cannot be disabled.

Below we list the situations where the CLI is the most suitable management tool.

- *Complete set of management features*: The CLI includes all the management features available on the switch. If you cannot accomplish your task with any of the other management tools, the CLI may provide the feature you need.
- *Discover other Westermo Switches*: The CLI contains a discovery service similar to what IPConfig provides. (Note, you must still be able to login to one switch in order to make use of this service.)
- *Secure management:* To access the CLI you must either have physical access to the switch (console port), or use the Secure Shell (SSHv2) application to access the CLI remotely. Secure management is also possible via the Web interface (HTTPS) and SNMP (SNMPv3).
- *Configuration scripting:* With a CLI it is possible to develop automatic configuration scripts, e.g., using the *Expect* automation and testing tool. *Expect* extensions exist for many common scripting languages (Ruby, Perl, Tcl).

[1] For more information about finding the IP address of your switch we refer to the *Getting Started* guide in chapter 2, and the general presentation of IPConfig in chapter 4.

As with the Web interface, you must know the IP address of your switch before you can access the CLI remotely via SSH (access via the console port is possible without knowing the switch IP address). To find out the switch IP address you may need to use the Westermo IPConfig tool[1], but once you know it you can do the rest of the management via SSH/CLI.

The WeOS CLI is introduced in chapter 6.

Chapter 4

The IPConfig management tool

Bundled with switch is a software CD including the Westermo IP Configuration Tool (IPConfig). IPConfig is a Microsoft Windows (2000/XP/Vista)[1] based application mainly used for identifying Westermo devices on a network. It can also be used to configure a small subset of the available switch settings. Currently IPConfig can be used to change the following settings for switches running WeOS:

- IP address
- Subnet Mask
- Gateway Address
- Hostname
- Location

In addition, the IPConfig tool can be used to upgrade the firmware of the switch.

4.1 Important Notice

Be aware that the PC with IPConfig needs to be connected to the same LAN (broadcast domain) as the Westermo switch in order for IPConfig to work. The reason for this is that the application uses local UDP broadcasts and those cannot be forwarded through a gateway or a router.

[1] Microsoft, Windows, Windows 2000, Windows XP and Windows Vista are trademarks of Microsoft Corporation.

4.2 Installation

This guide assumes the use of IPConfig version 10.3.0 or later. The IPConfig software uses a standard Windows installation procedure. Just ”double click” on the executable file, for instance *WestermoIpconfig-10.3.0.exe*, and follow the on-screen installation instructions.

4.3 Usage

After starting the IP Configuration Tool the window depicted in fig. 4.1 will appear.

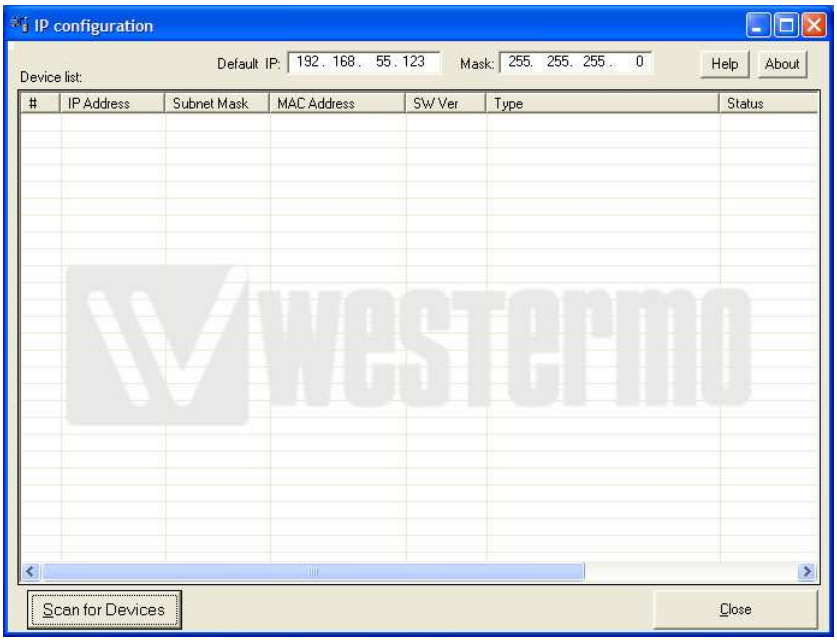


Figure 4.1: IPConfig startup window.

Once you have installed the IPConfig software on a PC, you must determine what IP subnet the PC’s network interface belongs to. For instance, if you know the PC has IP address *192.168.55.35* and a subnet mask of *255.255.255.0* (24 bits), then the subnet consists of all IP addresses in the range *192.168.101.X* where ’X’ is typically a number between 1 and 254 (0 and 255 are usually reserved).

After determining the PC’s subnet you can use any *free* IP address within that subnet, for instance 192.168.55.123 and set it as the **`Default IP:`** text field at

the top of the IPConfig startup window, see fig. 4.1. We use the IP subnet mask of the PC's network interface (here *255.255.255.0*, and insert it into the **Mask** text field, also located the top of the IPConfig startup window (see fig. 4.1). See table 4.1 for a summary.

Parameter	Value	Enter in IPConfig Tool
PC's IP Address	192.168.55.35	No
PC's IP Subnet Mask	255.255.255.0	Yes (**Mask:** text field)
"Free" IP Address	192.168.55.123	Yes (**Default IP:** text field)

Table 4.1: Sample values to enter in IP Config Startup Window.

When **Default IP** and **Mask** are configured correctly it is time to press the **Scan for devices** button at the bottom of the IPConfig startup window (see fig. 4.1). The result of the scan depends on what Westermo devices we have in our network. A sample result is shown in fig. 4.2.

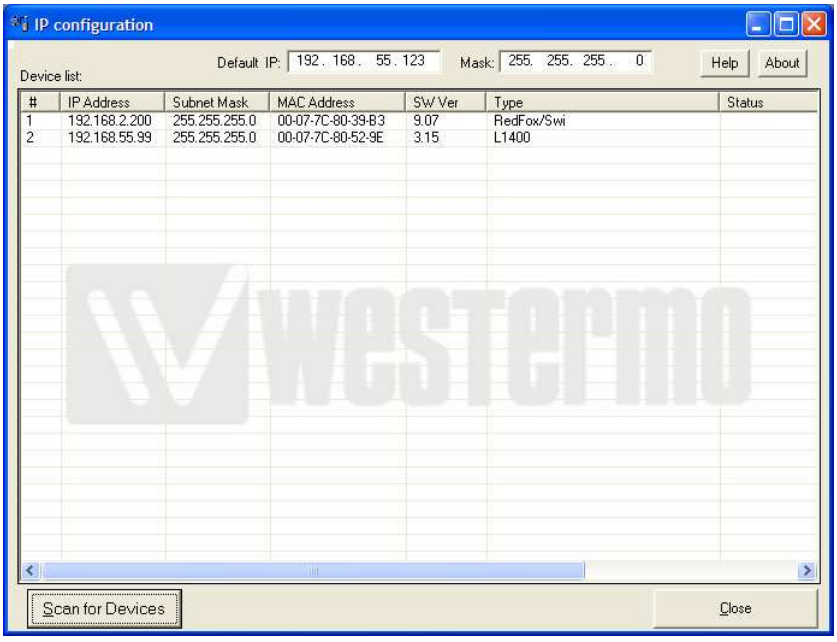


Figure 4.2: Result after a successful device scan.

The screen in fig. 4.2 displays two RedFox units in our connected network.

Each entry provides a brief summary of IP Address, Subnet Mask, MAC Address, Software Version, and the Type of Westermo unit. To change a setting on a unit we simply click on the row for the unit we want to change. For Westermo switches running WeOS the window presented in fig. 4.3 appears.

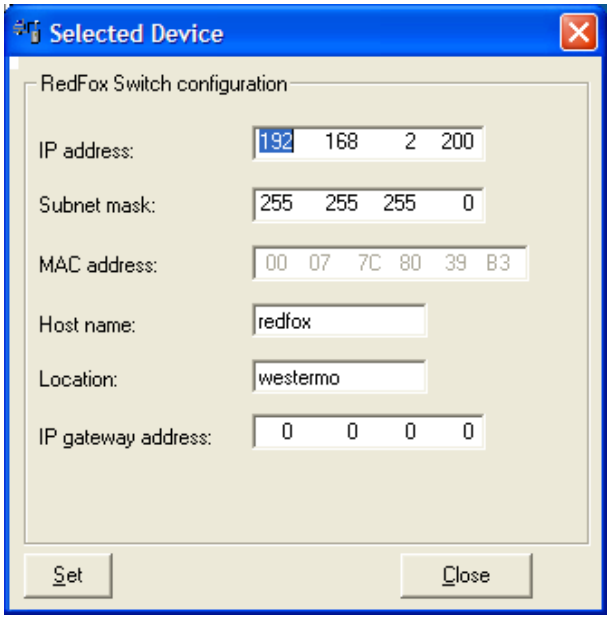


Figure 4.3: Selected Device view in IPConfig.

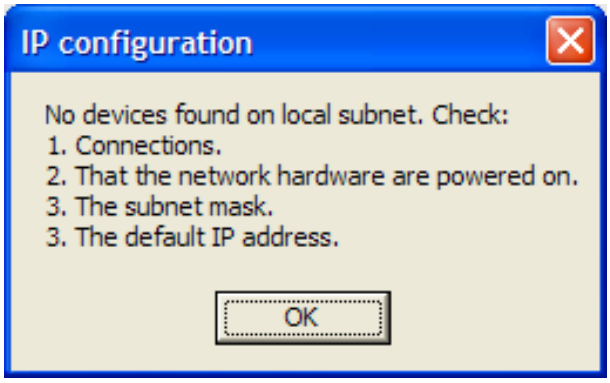
In this dialogue window we can configure everything but the MAC address. For instance, we might want to change the ”Location” information of the unit. All we need to do is type the new location in the **`Location`** text field, and then press the **`Set`** button.

Please note that a correct **`IP address`**, **`Subnet mask`** and **`IP Gateway`** mapping must be used. The IPConfig tool will evaluate the input and make sure everything matches; otherwise it will notify the user that something is incorrect.

4.3.1 Troubleshooting

As described above, IPConfig can be used to scan for Westermo switches on your local network, and also to manage a subset of all available switch settings.

If IPConfig is unable to find any Westermo switch on your local network, the following error message will appear (instead of the successful result shown in fig. 4.2).



The error message provides a check list to determine what may have gone wrong:

1. Check that your PC (with IPConfig) is properly connected the LAN you want to scan for switches (typically you connect the PC directly to a Westermo switch with an Ethernet cable).
2. Ensure that the Westermo switches you are scanning for are powered on.
3. Make sure you have filled in the proper subnet mask in the **Mask** text field of the IPConfig Startup Window (see fig. 4.1).
4. Make sure you have filled in the proper IP address in the **Default IP** text field of the IPConfig Startup Window (see fig. 4.1).

If nothing of this helps, the reason may be that IPConfig has been disabled on the switch you are scanning for, or that the firewall or IP settings of the switch blocks the IPConfig traffic. In this case, please visit chapter 7 (in particular section 7.1.4) for information on how to proceed.

4.3.2 Upgrading primary firmware using IPConfig

In addition to configurating a basic set of switch settings, IPConfig can be used to upgrade the *primary firmware* of the switch (for more information on switch firmware, see section 7.1.1).

> **Note**: *To upgrade the switch primary firmware via IPConfig, the PC running IPConfig must also run a FTP server or a TFTP server. When using a FTP server, the primary firmware image must be located in the anonymous FTP directory on the PC.*

The following fields in the IPConfig configuration window (fig. 4.4) are used when upgrading the primary firmware:

- *Hostname*: Enter *upgrade* to upgrade via FTP/TFTP. FTP will be tried first, with fall-back to TFTP if FTP fails.
- *Location*: Enter the name of the file with the new primary firmware, e.g., *rw430.img*.

After filling in the appropriate values in the **Hostname** and **Location** text fields, press the **Set** button to start the firmware upgrade process.

Note: *Make sure you load a primary firmware applicable for your specific Westermo product. See section 7.1.1.1 for details.*

Fig. 4.4 shows an example where IPConfig is used to upgrade the primary firmware of the switch (file *rw430.img*) from a FTP (or TFTP) server, located on the PC running IPConfig.

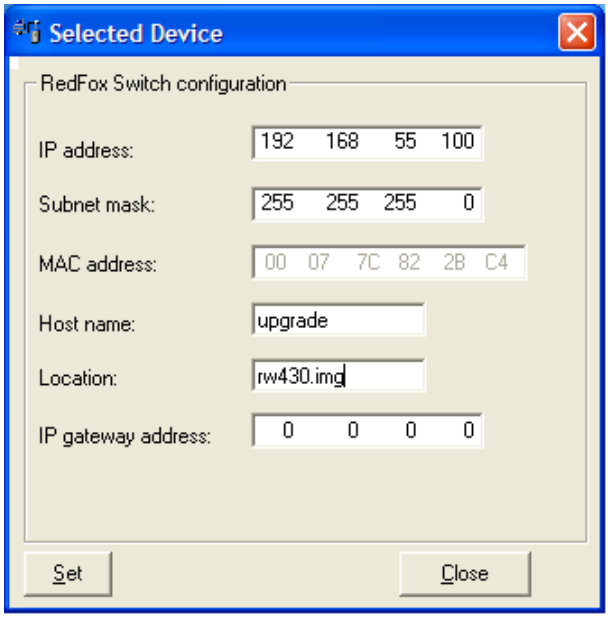


Figure 4.4: Upgrading Primary Firmware via IPConfig.

Chapter 5

The Web Management Tool

WeOS supports device management via web interface. Both HTTP and HTTPS[1] are supported. The design is optimised for style sheet and JavaScript[2] capable web browsers. In addition, the design allows users to access the web interface and all settings *without* a style sheet and JavaScript capable browser, but then with less guidance and support from the user interface. Westermo recommends using Internet Explorer 6 (or later) or Firefox 2 (or later).

When using the Web Management Tool you have to be aware of the following:

- Only one user can be logged in at a time (see section 5.2 for more information).
- You are automatically logged out after ten (10) minutes of inactivity (see section 5.2 for more information).
- When you click **`Apply`** on a page, the settings on that page are immediately activated.
- When you click **`Apply`** on a page, all settings are stored in the *startup configuration* and therefore survive a reboot (see chapter 7 for more information).

Section 5.2 explains how to access the Web Management Tool and section 5.3 describes the web menu hierarchy. In section 5.3 the *system overview* web pages are presented. Other pages and settings are described per topic in chapter 8 and following chapters.

[1] For HTTPS server authentication, a self-signed certificate is uses as of WeOS v4.3.0.
[2] JavaScript is a trademark of Sun Microsystems.

5.1 Document Conventions

Specific conventions for the web part of this document.

`Button Text`	Buttons are indicated by use of bold type-writer style.
Menu path: Top Item ⇒ Sub Item	For each page the menu path to the page is described with this syntax. It means: First click the *Top Item* menu item and in the sub-menu revealed, click the *Sub Item* menu item. See also section 5.3.
Menu path: Top Item ⇒ Sub Item ⇒ **`Button Text`**	This is an extension to the *Menu path: Top Item ⇒ Sub Item* version described above. It tells you to click a button with the text *Button Text* on the page navigated to by *Top Item ⇒ Sub Item*.
Top Item ⇒ Sub Item ⇒  (ctx)	The button may be an icon. In this case the icon is shown. Additionally in parenthesis a sub-context (ctx) may be described which will identify a context on the page, normally identified by its header.

5.2 Logging in

To access the switch through the web interface, enter the appropriate URL (e.g., the factory default IP-address http://192.168.2.200) in the address field of your web-browser. You will then be presented to the login page where you fill in the *username* and *password*, see figure 5.1.

Currently there is only a single user account defined, the *administrator* user account. Note that it is the same user account used for login in CLI. Factory default user account and password are as follows :

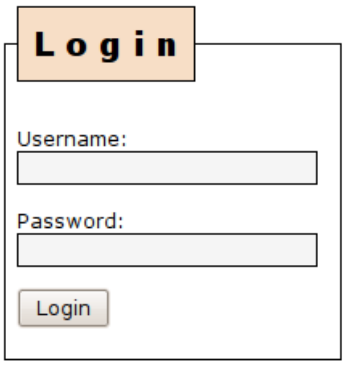


Figure 5.1: Web login window

- Login: **admin**
- Password: **westermo**

Your web session will last for ten (10) minutes after your latest ”web action”. Clicking a link or button at least every 10 minutes will let you keep the session forever. The same goes for pages with an automatic refresh option, given that a refresh interval of 10 minutes or shorter is selected.

Only *one user at a time* can be logged into the switch Web Management Tool. If a new user tries to log in the currently logged in user will automatically be logged out.

5.3 Navigation

After logging in you will be redirected to the *start page*, see fig. 5.2. In the page header you find the menus used to navigate between different tasks. The menu consists of two rows, the *top-menu* row, and the *sub-menu*. For some items you will be presented to a third level sub-menu below the second level sub-menu. Its function is analogously to the second level sub-menu .

To navigate in the menu, click on the *top-menu* to reveal the associated *sub-menu*. Then click on the desired *sub-menu* item. For example, fig. 5.2 shows the selection of top-menu *Home* and sub-menu *Summary* (i.e., Home ⇒ Summary).

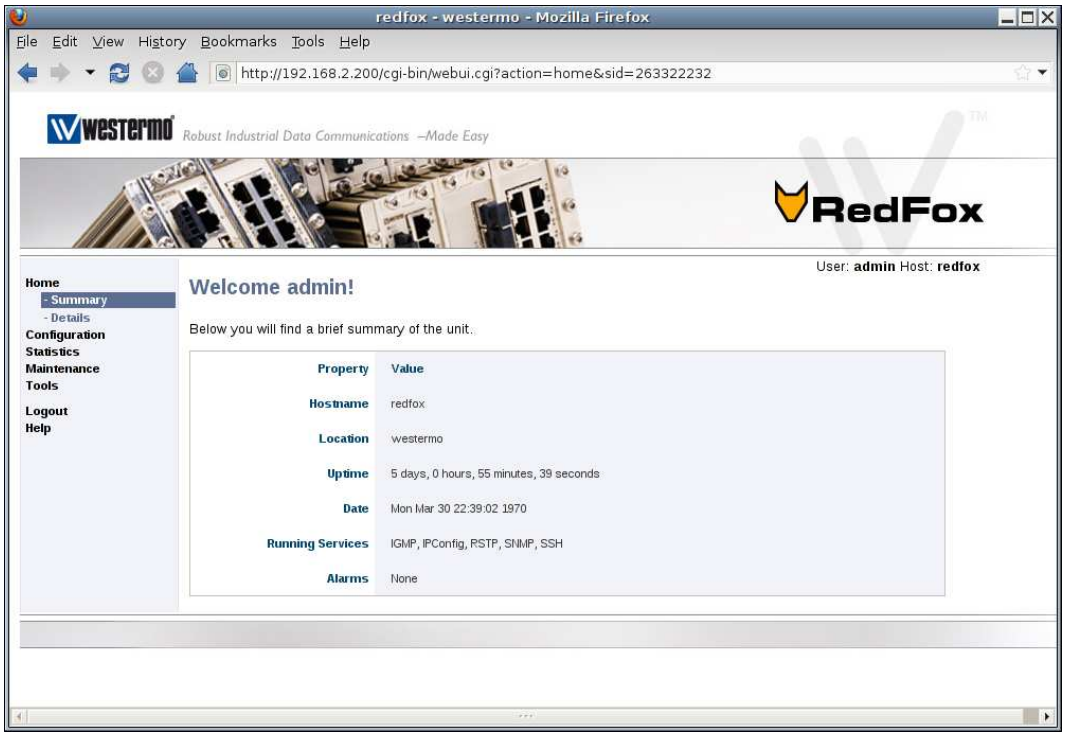


Figure 5.2: Unit Summary - the first page after logging in.

The menu structure is described below:

- Home
  - Summary - Basic switch overview
  - Details - Detailed switch overview
- Configuration
  - Network (IP) - Network related IP-settings
    - Global settings - Global IP-settings
    - Interface - Interface settings
    - DDNS - Dynamic DNS settings
  - VLAN - VLAN settings and port assignment
    - VLANS - VLANS settings
    - Dynamic - Dynamic VLAN settings
  - Port - Port settings
  - FRNT - FRNT settings
  - RSTP - RSTP settings
  - IGMP - Global IGMP settings
  - SNMP - SNMP settings
  - Firewall - Firewall related settings, see subcontexts below.
    - Common - Common firewall settings.
    - NAT - Network address translation settings.
    - Port Forwarding - Setting up port forwarding rules.
    - Access - Setting up firewall rules to allow access through the firewall.
  - Identity - Hostname, location and contact settings
- Statistics
  - Port - Port statistics (RMON etc)
- Maintenance
  - Date & Time - Set the date and time
  - Backup & Restore - Backup and restore switch configuration
  - F/W Upgrade - Firmware upgrade, using FTP/TFTP or file upload
  - Port Monitoring - Port monitoring (a.k.a. port mirroring) for debugging

  - Password - Change user password
  - View Log - Show system logs
  - Restart - Restart the switch
- Tools
  - Ping - Ping tool
  - Trace - Traceroute tool
  - IPConfig - IPConfig tool
- Logout - Logout from the session
- Help - Online help for current page/context.

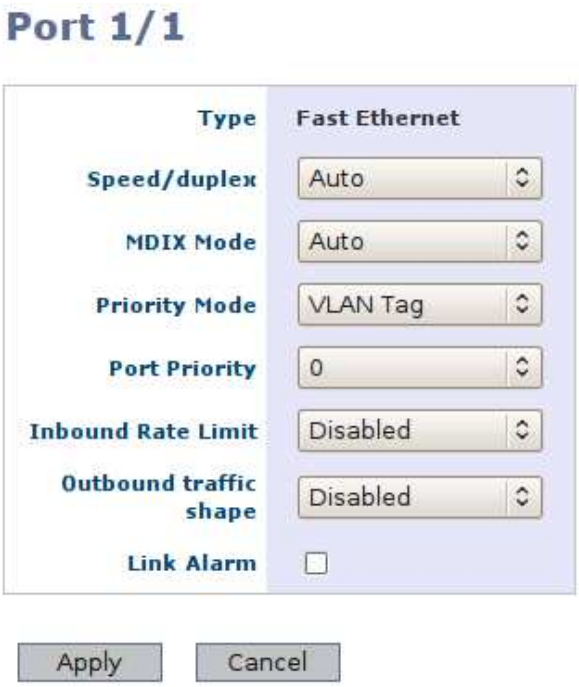


Figure 5.3: Sample web page containing **Apply** and **Cancel** buttons.

Pages where you can change settings generally contains an **Apply** and a **Cancel** button, as shown in fig. 5.3. The semantics of the **Apply** and **Cancel** buttons are provided below:

Apply	Applies the changes on the current page. Changes are applied immediately (i.e., no reboot needed), and are also stored in the startup configuration.
Cancel	Discards changes and either returns to an overview page for the context, or reloads current page and thus shows the current settings.

5.4 System Overview

There are two levels of system information, *summary* and *detailed*.

5.4.1 System Overview - Summary

Menu path: Home ⇒ Summary

Fig 5.4 shows the first page you will be presented to after logging into the switch. It provides a quick overview of the system, including a list of current alarms.

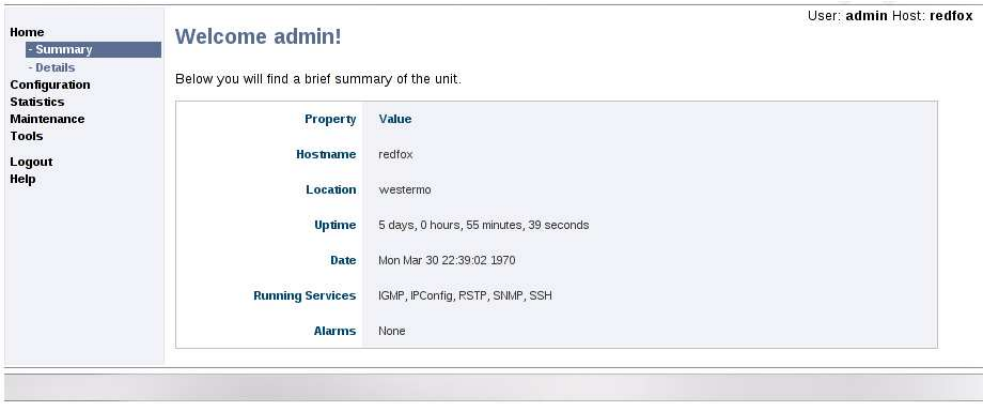


Figure 5.4: The basic system overview page (here on a RedFox Industrial switch).

Hostname	An arbitrary name to identify this unit.
Location	An arbitrary description to identify where the unit is located.
Uptime	The time passed since last reboot of the unit.
Date	The current date and time. System time is configured manually or set by using a NTP-server.
Running Services	A list of services currently running on the unit.
Alarms	Currently active port and FRNT alarms. *Link alarms* are only shown for ports where link alarm is enabled and when the link is down. *FRNT alarms* are only shown for FRNT ports with link down.

5.4.2 System Overview - Detailed

Menu path: Home ⇒ Details

To get more information about the switch you go to the detailed page shown in Fig 5.5. This page contains more information on hardware (e.g. versions, article number, etc.) and system status (e.g. memory usage and CPU load).

Details

Property	Value
Hostname	redfox
Location	westermo
Contact	support@westermo.com
Uptime	0 days, 0 hours, 45 minutes, 21 seconds
Base Mac Address	00:07:7c:82:09:90
System Default Gateway Address	0.0.0.0
Article Number	3641-3100-0
Main Firmware Version	9.99
Backup Firmware Version	9.99
Main FPGA Version	20080626
Boot Loader Version	2.01
Serial Number	3568
Product	RedFox Industrial
Model	RFI-18P
Card #1	
Type	CPU
Article No.	5013-0000
Batch ID	081211-00012345-00000
Revision	0
Card #2	
Type	10/100TX
Article No.	5013-0100
Batch ID	081211-00012345-00000
Revision	0
Card #3	
Type	10/100TX
Article No.	5013-0100
Batch ID	081211-00012345-00000
Revision	0
Card #4	
Type	BACKPLANE
Article No.	5010-0910
Batch ID	090126-00000000-00036
Revision	0
Card #5	
Type	POWER
Article No.	5013-0200
Batch ID	081211-00012345-00000
Revision	0
Enabled Redundancy Protocol(s)	None
VLANs With IGMP	Disabled
SNMP	Enabled
Alarms	None
Load Average	
1 minute	0.15
5 minutes	0.05
15 minutes	0.01
Memory Usage (%)	23

Figure 5.5: Detailed system overview page (here on a RedFox Industrial switch)

Hostname	An arbitrary name to identify this unit.
Location	An arbitrary description to identify unit location.
Contact	An arbitrary description to identify a contact person who has more information about management of the unit and the network.
Uptime	The time passed since last reboot of the unit.
Base MAC Address	The base MAC address defines the starting point of the MAC address range used within the unit. This is a unique number assigned to each unit.
System Default Gateway Address	The operational default gateway for all VLANs on the unit. Either retreived dynamically or set statically. This is the IP-address of the gateway to send packages to when no more specific route can be found in the routing table.
Article Number	The article number for the unit.
Main Firmware Version	The version number of the main firmware.
Build Details	The build string of the currently running firmware.
Backup Firmware Version	The version number of the backup firmware.
Main FPGA Version	The version number of the FPGA software.
Boot Loader Version	The version number of the boot loader software.
Serial Number	The units serial number.
Product	The product name.
Model	The product model.
Type	Description for the card in the specified slot.
Article No.	The article number of the card in the specified slot.
Batch ID	The batch identification of the card in the specified slot.
Revision	The revision of the card in the specified slot.
Enabled Redundancy Protocol(s)	A list of the redundancy protocols currently enabled on the unit.
VLANs With IGMP	A list of VLANs on which IGMP is enabled.
SNMP	Shows if SNMP support is enable or disabled.
Alarms	Currently active port and FRNT alarms. *Link alarms* are only shown for ports where link alarm is enabled and link is down.*FRNT alarms* are only shown for FRNT ports where link alarm is enabled and when the link is down.
Load Average	The load average is a standard Linux way of measuring system load.
Memory Usage (%)	A snapshot of RAM (Random Access Memory) usage as percentage of total RAM.

Chapter 6

The Command Line Management Tool

This chapter introduces the command line interface (CLI) tool. All Westermo switches running the WeOS software include a CLI similar to what is provided by major vendors of network equipment. The CLI provides a more complete set of management features than the Web interface, the IPConfig tool or SNMP. Thus, when advanced management operations are required, the CLI is the management interface of choice.

The CLI can be accessed via the console port (on devices equipped with a console port) and remotely via secure shell (SSHv2).

Section 6.1 introduces the CLI hierarchy and its various contexts. Section 6.2 explains how to access the CLI interface, and section 6.3 provides general information on how to use the CLI.

The last section (section 6.4) presents CLI commands available in *all* CLI contexts as well as their syntax. Other CLI commands are described per topic in the chapters to follow.

6.1 Overview of the WeOS CLI hierarchy

The WeOS CLI is organised in a hierarchical structure. For management purposes, the use of a hierarchical structure limits the available commands to those relevant for a certain topic. This in turn simplifies switch operation.

Fig. 6.1 shows an overview of the CLI hierarchy. When the user logs in as ”admin” the user will enter the CLI with ”administrator” privileges in Admin Exec

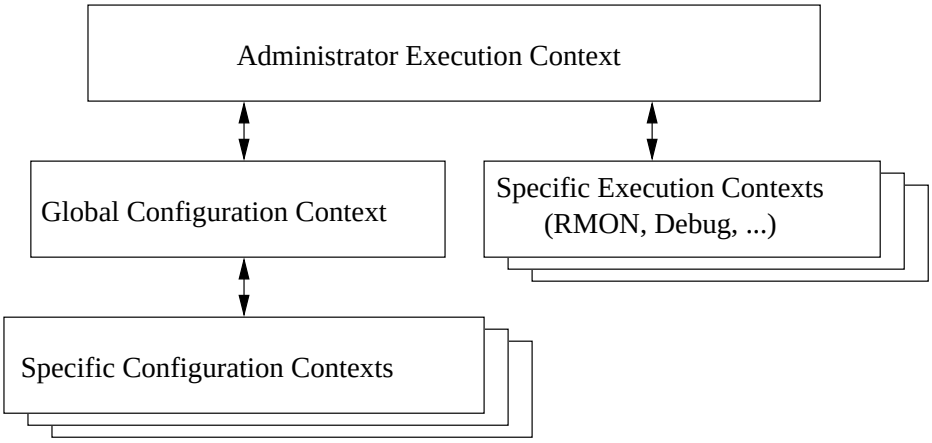


Figure 6.1: CLI hierarchy

context. (In addition to the ”admin” user, future versions of WeOS are likely to support a ”guest” account with limited privileges.)

Admin Exec context In Admin Exec context the user can execute a set of general monitoring and diagnostic functions, and also manage configuration files and firmware versions. From Admin Exec context the user can enter a set of specific execution contexts, e.g., to view RMON statistics.

Global Configuration context From the Admin Exec context the user can enter the Global Configuration context. In Global Configuration the user can configure device parameters of global significance, such as *hostname* and *location* of the device. From Global Configuration the user can reach contexts specific to certain protocols or device entities such as *port*, *vlan*, *interface*, and *FRNT* contexts.

A simple example on CLI usage is given below. There you can see how the CLI prompt changes to match the current context.

```
redfox:/#> configure
redfox:/config/#> vlan 100
redfox:/config/vlan-100/#> untagged 1/1,1/2
redfox:/config/vlan-100/#> end
redfox:/config/#> end
redfox:/#>
```

6.2 Accessing the command line interface

To login via the console port you need the username and password. Currently there is only a single user account defined, the *administrator* user account. Factory default account and password:

- Login: **admin**
- Password: **westermo**

The same account is used for management via CLI and Web (see section 5). To reset the *administrator* password to the default setting, see chapter 7.

6.2.1 Accessing CLI via console port

For Westermo switches equipped with a console port, that port can be used to access the CLI. (For information on which WeOS devices that have a console port, see section 1.4.1).

Console cable: *To access the console port on Westermo Switches equipped with a 2.5 mm jack console port (see, e.g., fig. 1.1), the Westermo Diagnostic Cable, 1211-2027, must be used to connect a USB port on your PC to the switch console port. See the User Guide of your specific product for more information[8, 9, 10, 11].*

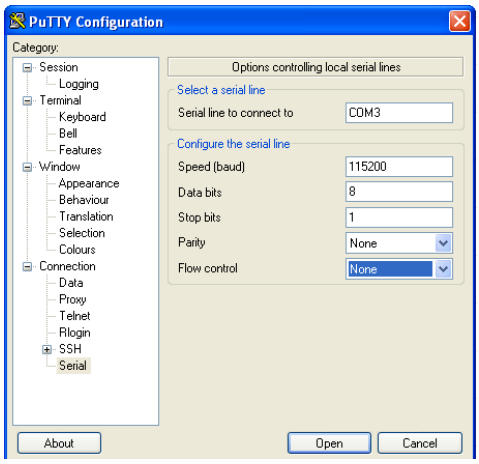
Recommended Terminal Emulation programs:

- **Win32**: *PuTTY*, http://www.chiark.greenend.org.uk/~sgtatham/putty/
- **UNIX**: There are different terminal emulation programs for different Unix dialects. On Linux Westermo recommends *minicom*.

The following console port settings are used:

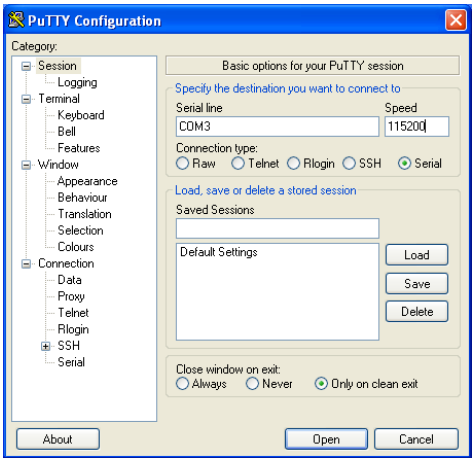
Data rate	115200 bits/s
Data bits	8
Stop bits	1
Parity	None
Flow control	None

The example in below shows how to login via the console port using the *PuTTY* application. Once you have installed and started *PuTTY*, configure the appropriate *Serial* settings.

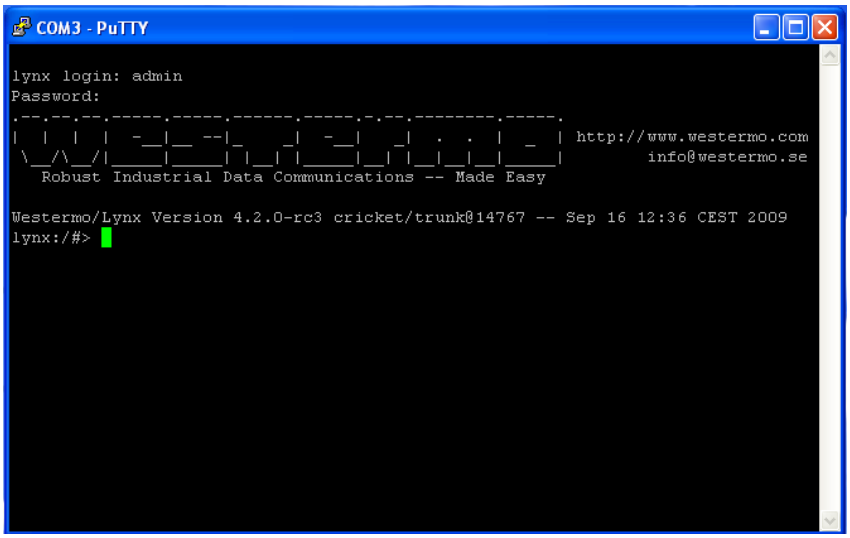


Hint: *In this example, the switch is accessible via the logical port "COM3", but the USB/serial adapter may be mapped to a different COM port on your PC. Please check "Ports (COM and LPT)" in the Windows "Device Manager" to get information on what COM port to specify.*

When the appropriate serial settings have been configured, select the "Session" view. Select *Serial* as *Connection type* as shown in the figure below.



To start the serial connection, press the **0pen** button. The figure below shows the console prompt when logging in via the CLI after a system boot.



6.2.2 Accessing the CLI via SSH

To gain access to the CLI via SSH you need a *SSH client*, the switch IP address, and the account information (username and password).

Recommended SSH Clients:

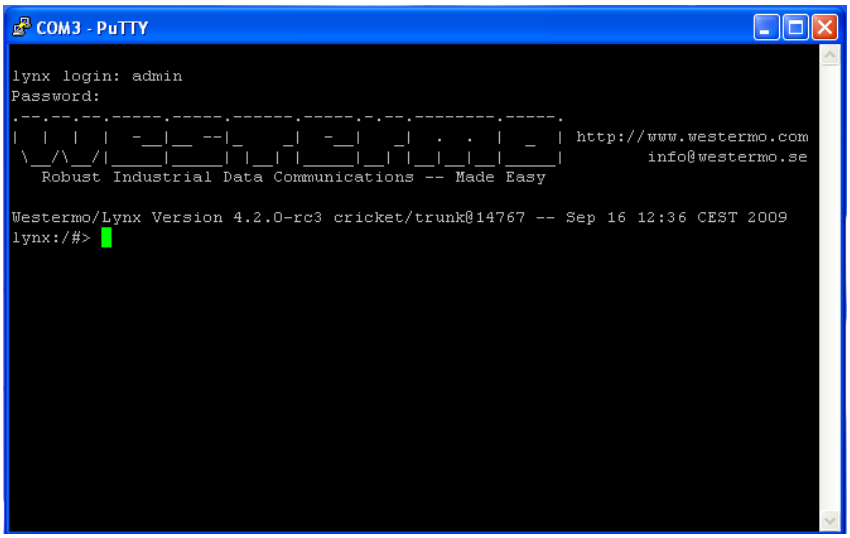
- **Win32**: PuTTY, http://www.chiark.greenend.org.uk/~sgtatham/putty/
- **UNIX** OpenSSH, http://www.openssh.com

The switch IP address can be found using the IPConfig tool, see section 4 (additional methods are listed in section 7.1.4).

The following example illustrates how to login to the switch using PuTTY from a Windows based host system as user *admin*. In this example, the switch is a RedFox Industrial switch with IP address 192.168.2.200 (the factory default IP address). See section 6.2 for information about user accounts and passwords.

In the PuTTY session view, select *SSH* as *Connection type*, and enter the IP address of the switch (here 192.168.2.200).

Click the **Open** button to start the SSH session. You will be presented to a login prompt (see below), and enter login *admin* and the associated password.



6.3 Using the CLI

6.3.1 Starting out with the CLI

When first entering the CLI you end up in the *Admin Exec* context. In the *Admin Exec* you can view system *status* information using various **”show”** commands, upgrade system firmware, etc., as well as other functions, which do not affect the system *configuration*.

To be able to modify the switch configuration you should enter the *Global Configuration* context, by using the **”configure”** command as shown below. From the *Global Configuration* you are able to configure system parameters such as its **”hostname”** or its **”date”**.

```
redfox:/#> configure
redfox:/config/#>
```

As described in section 6.3.2 you can reach other, specific configuration contexts from the *Global Configuration* context.

```
redfox:/#> configure
redfox:/config/#> vlan 100
redfox:/config/vlan-100/#> untagged 1/1,1/2
redfox:/config/vlan-100/#> end
redfox:/config/#> end
redfox:/#>
```

To get help on what commands are available in the current context, use the **”help”** command (see example below). First the context specific configuration commands are shown, followed by the commands to *show* the current configuration settings. At the end, commands available in all contexts are shown (see also section 6.4.).

```
redfox:/config/vlan-100/#> help
Available Commands
================================================================================
enable                    Enable, or disable this VLAN
name <ARG>                Set name of VLAN
tagged <ARG>              Set tagged ports
untagged <ARG>            Set untagged ports
channel <ARG>             Set VLAN channel interface
priority <ARG>            Set VLAN priority, overrides port priority
igmp                      Enable, or disable IGMP Snooping

show enable               Show if VLAN is active or not
show name                 Show name of VLAN
show tagged               Show tagged ports
show untagged             Show untagged ports
show channel              Show VLAN channel interface
show priority             Show VLAN priority setting
```

```
show igmp                Show IGMP Snooping status

no <ARG>                 Prefix, used to disable services or settings.
do                       Shortcut to EXEC mode, e.g. do ping <IP>.
end                      Save settings and return to previous mode.
leave                    Save settings and return to EXEC mode.
abort                    Cancel all changes and leave this mode.
show <ARG>               Show summary, or status.
repeat <ARG>             Repeat next command every second, until Ctrl-C
help <ARG>               This help text.
tutorial                 Brief introduction to the CLI
================================================================================
<ARG> - Command takes argument(s), see help <command> for further information.
Short forms of commands are possible, see the tutorial for more help.
redfox:/config/vlan-100/#>
```

The **”help”** command can also be used to get information on a specific command as shown below.

```
redfox:/config/vlan-100/#>  help igmp
Syntax:
        [no] igmp

Description:
        Enable, or disable IGMP Snooping

================================================================================
The [no] keyword is when you want to disable a service or remove a property.
redfox:/config/vlan-100/#>
```

The CLI supports basic *TAB-completion*, which can come in handy when you do not know the exact command name, e.g., writing **”fi[TAB]”** within the *IP* context will expand to **”firewall”**.
TAB-completion is only able to expand the full command when there is no ambiguity. Otherwise the available alternatives will be listed.

```
redfox:/#>  d[TAB]
do      debug   date    dir     delete
redfox:/#> d
```

Furthermore, when there is no ambiguity it is possible to use an abbreviation of a command instead of the full command (i.e., without using *TAB-completion*).

```
redfox:/#>  con
redfox:/config/#>
```

6.3.2 Entering and leaving CLI contexts

Fig. 6.2 gives a general overview of how to enter and leave the various context in the CLI hierarchy. The commands to move between contexts are further discussed in the text below.

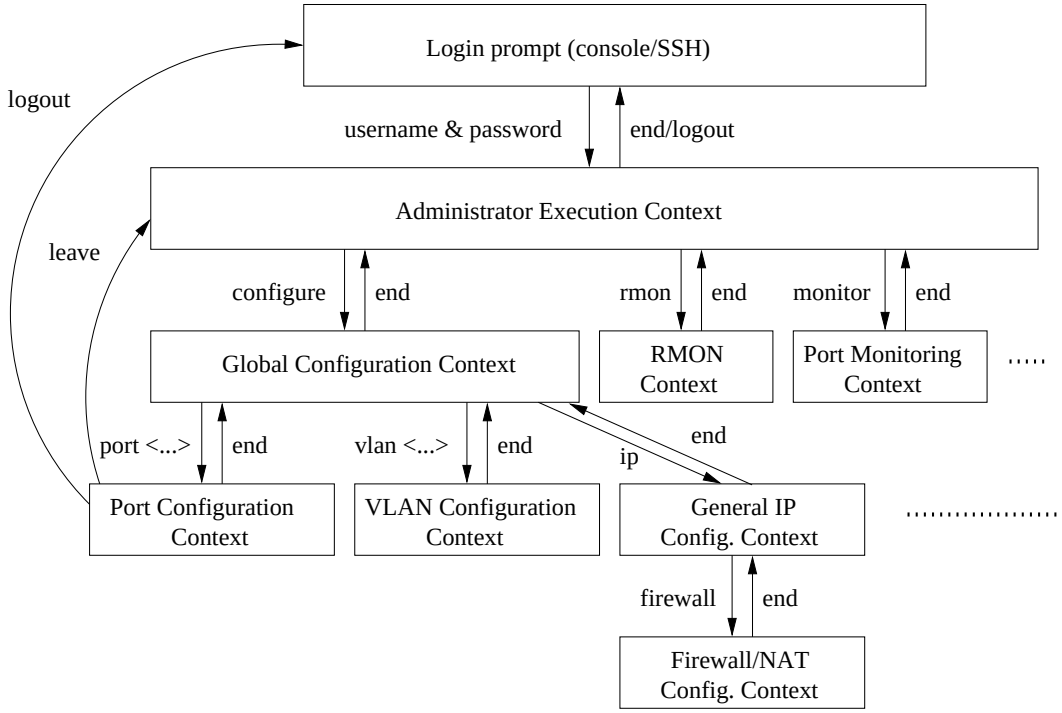


Figure 6.2: Moving between CLI contexts. Only a subset of the available contexts is shown. Although not shown, the *leave* and *logout* commands can be used from all contexts.

To enter Global Configuration context from Admin Exec context, the **"configure"** command is used. From Global Configuration context one can reach several specific configuration contexts, and the command to enter them is context specific, e.g.,:

vlan <VID>	Manage VLAN settings for VLAN with given VID.
port <PORT>	Manage port settings for port with given PORT identifier.
interface <IFNAME>	Manage settings for the given network interface.

By entering the Global Configuration context the user is able to interactively change the device configuration, however, configuration changes will not take effect until the user leaves the configuration contexts and returns to the Admin Exec context via the **"end"** or **"leave"** commands.

When the user returns to Admin Exec context, the *running-configuration* of

the switch will be updated. To make the configuration changes permanent the *running-configuration* should be saved to the *startup-configuration* using the **"copy"** command, see also chapter 7.

It is also possible to leave the configuration contexts without updating the *running-configuration*. The commands to leave a context are listed below. More information on these and other general CLI commands can be found in section 6.4.

end Confirms configuration changes conducted in this context and returns to the context immediately above. If issued within the Global Configuration context, the user returns to the Admin Exec context and the *running-configuration* is updated.

leave Confirms configuration changes conducted in this context and returns to the Admin Exec context. The *running-configuration* is updated.

abort Discards configuration changes conducted in this context and returns to the context immediately above. If issued within the Global Configuration context, the user returns to the Admin Exec context without updating the *running-configuration*.

logout Log out from the CLI. If conducted from within any of the configuration contexts, all configuration changes are discarded (i.e., the *running configuration* is not updated).

6.3.3 CLI command conventions

This section describes the CLI command conventions used within this guide. The syntax for a sample set of CLI commands is shown below:

- `[no] default-gw <ADDRESS>`
- `igmp-interval <12|30|70|150>`
- `show iface [IFNAMELIST]`

Convention	Description
`command syntax`	Command syntax is generally written in typewriter style (fixed width)
`”command syntax”`	Commands described in running text use bold typewriter style enclosed by quotation marks.
`UPPERCASE`	A variable parameter. Enter value according to the description that follows.
`lowercase`	A keyword parameter. Enter value according to the given syntax.
`|`	Vertical bar. Used to separate alternative (mutually exclusive) parameters.
`< >`	Angle brackets. Encloses a mandatory parameter.
`[ ]`	Squared brackets. Encloses an optional parameter.
`[< >]`	Angle brackets within squared brackets. Encloses a mandatory parameter within an optional choice.

6.4 General CLI commands

The majority of the CLI commands are specific to a certain context, however, there is a set of CLI commands available in all contexts. These commands are explained further here. The **"configure"** command used to enter the Global Configuration context from the Admin Exec context, is also covered.

Command	Section
no <COMMAND>	Section 6.4.1
do	Section 6.4.2
end	Section 6.4.3
leave	Section 6.4.4
abort	Section 6.4.5
logout	Section 6.4.6
repeat <COMMAND>	Section 6.4.7
help [COMMAND]	Section 6.4.8
tutorial	Section 6.4.9
configure	Section 6.4.10

6.4.1 Negate/disable a setting

Syntax `no <COMMAND>`

Context All contexts

Usage Depending on context the **"no"** command disables or resets a setting to default.

Primarily used within configuration contexts to negate or disable a configuration setting, e.g., in *port* context **"no flow-control"** disables flow control. For some commands, "no" is used to reset to a default value, e.g., **"no polling-interval"** (SNTP context) sets the SNTP polling-interval to its default value (600 seconds).

The **"no"** command can also be used to negate/disable certain commands outside the *configuration* context, e.g., to disable debugging or port monitoring.

Default values Not applicable

Error messages None defined yet

6.4.2 Execute (do) command from Admin Exec context

Syntax `do <COMMAND>`

Context All contexts

Usage Use the **"do <COMMAND>"** to execute a COMMAND available in *Admin Exec* context from any context.

For example, when located in Global Configuration context, the user could run **"do show running-config"** to see the *running configuration*, or run **"do ping 192.168.1.1"** to "ping" IP address 192.168.1.1.

Default values Not applicable

Error messages None defined yet

6.4.3 End context

Syntax end

Context All contexts

Usage Leave this context and return to the context immediately above. If this command is issued within any of the configuration contexts, the command implies that the configuration changes conducted within that context are confirmed. If the command is issued in the Global Configuration context, the user returns to the Admin Exec context, and the *running-configuration* is updated.

Default values Not applicable

Error messages None defined yet

6.4.4 Leave context

Syntax `leave`

Context All contexts

Usage Leave this context and return to the Admin Exec context. If this command is issued within any of the configuration contexts, the command implies that the configuration changes conducted are confirmed, and the *running-configuration* is updated.

Default values Not applicable

Error messages None defined yet

6.4.5 Abort context

Syntax `abort`

Context All contexts

Usage Leave this context and return to the context immediately above. If this command is issued within any of the configuration contexts, the command implies that the configuration changes conducted within that context are discarded. If the command is issued in the Global Configuration context, the user returns to the Admin Exec context without updating the *running-configuration*.

Default values Not applicable

Error messages None defined yet

6.4.6 Logout

Syntax `logout`

Context All contexts

Usage Logout from system. If this command is issued within any of the configuration contexts, the command implies that the configuration changes conducted are discarded, i.e., the *running-configuration* is not updated.

Default values Not applicable

Error messages None defined yet

6.4.7 Repeat a command

Syntax `repeat <COMMAND>`

Context All contexts

Usage Repeat COMMAND every second until Ctrl-C is entered.

Default values Not applicable

Error messages None defined yet

6.4.8 On-line help

Syntax `help <COMMAND>`

Context All contexts

Usage Show help information specific to a certain context, or a specific command.

Default values If no COMMAND is specified, help information related to the current context is shown.

Error messages None defined yet

6.4.9 CLI tutorial

Syntax `tutorial`

Context All contexts

Usage Show CLI tutorial text.

Default values Not applicable

Error messages None defined yet

6.4.10 Entering Global Configuration Context

When a user logs in to the CLI the user will enter the *Admin Exec* context. In Admin Exec context the user can view status information and have access to tools such as *ping* and *traceroute*, but is not able to perform any configuration. To configure the device, the user can use the *configure* command to enter the Global Configuration Context.

Syntax `configure [terminal]`

Context *Admin Exec* context

Usage Enter global Configuration Context.

The optional `terminal` argument is a compatibility keyword, for advanced users. It disables all safe guards (yes-or-no questions), making it possible to paste-in configuration files into the terminal.

Pasting in configuration files can also be done with the `copy` command as `copy con run` to copy *console* to *running-config*.

Default values Not applicable

Error messages None defined yet

Chapter 7

General Switch Maintenance

7.1 Overview

The table below summarises maintenance features available for the different management tools. General descriptions of these features are presented in sections 7.1.1-7.1.6. If you are only interested in knowing how to manage maintenance features via the Web or CLI, please visit sections 7.2 or 7.3 directly.

Feature	Web (Sec. 7.2)	CLI (Sec. 7.3)	General Description
Firmware Upgrade			
Upgrade primary firmware	X	X	Sec. 7.1.1
Upgrade backup firmware		X	”
Upgrade bootloader		X	”
View firmware versions	X	X	”
Account management			
Set Admin Password	X	X	Sec. 7.1.2
Recover from lost Admin Password		X	”
Controlling Management Services			
Enable Web		X	
Enable IPConfig		X	

Continued on next page

Continued from previous page

Feature	Web (Sec. 7.2)	CLI (Sec. 7.3)	General Description
Configuration Files and Reboot			
Reboot	X	X	Sec. 7.1.3
Reset to Factory Default	X	X	”
View Configuration Files	(X)	X	”
Configuration Backup	X	X	Secs. 7.1.3 and 7.1.5
Configuration Upload	X	X	”
Alternate Configuration Files		X	”
Virtual File System			
Maintenance of Configuration		X	Sec. 7.1.5
Log and USB files	(X)	X	”
Maintenance and diagnostic tools			
Ping	X	X	Sec. 7.1.6
Traceroute	X	X	”
IPConfig Client	X	X	”
Port Monitoring	X	X	”
SSH Client		X	
Other maintenance features			
Show System Environment Sensors	X	X	
Show System Uptime	X	X	
Show Memory Usage	X	X	
Show Running Processes		X	

7.1.1 System Firmware

The system keeps three types of firmware:

- *Primary firmware*: The primary firmware contains the main system software with the features described in this document.
- *Backup firmware*: The backup firmware (also known as secondary firmware) is loaded in case an error (such as a checksum error) is encountered while loading the primary firmware. The backup firmware need not include all the functionality that the primary firmware has; the main purpose of the backup firmware is to enable the user to upload a new primary firmware to the switch in case the existing primary firmware is broken.
- *Bootloader*: The basic firmware run to bootstrap the system. The bootloader will in turn load the primary firmware.

It is possible to upgrade all three types of firmware. Most users would only be concerned with the primary firmware. Upgrading the backup firmware and the bootloader is limited to the CLI tool.

Warning: *There is no general guarantee that an older firmware can be loaded into the switch, i.e.,* downgrade *is not generally guaranteed to work. However, if the firmware is downgraded for example from version 0.95 to 0.94, it is recommended to reboot the switch once the old firmware has been installed. When the switch comes up with the old firmware (here 0.94), copy the* factory default configuration *to the* running configuration*. See section 7.1.3 for more information on configuration files.*

7.1.1.1 Firmware name conventions for different WeOS products

The WeOS firmware images and bootload images differ depending on the type of Westermo product. As of WeOS version v4.3.0 the following name conventions are used[1].

[1] As the name conventions for primary/secondary firmware and bootloader may differ for older or future versions of WeOS, please consult the release notes attached with release zip-file for definite information.

Product	Primary and secondary FW	Bootloader FW
RedFox	rwXXX.img (e.g., rw430.img)	xscale-redboot-YYY.bin (e.g., xscale-redboot-2.01.bin)
Lynx+	lwXXX.img (e.g., lw430.img)	imx27-redboot-ZZZ.bin (e.g., imx27-redboot-4.06.bin)
Wolverine (DDW-225/226)	wwXXX.img (e.g., ww430.img)	” ”
Lynx 1400G (customer specific)	lmXXX.img (e.g., lm430.img)	” ”

7.1.2 Account Management

Currently WeOS only supports a single user account, the **admin** user account. The same account is used when managing the switch via the Web or via the CLI. Factory default settings for the user account is:

- Login: **admin**
- Password: **westermo**

The *admin* password can be changed, both via the Web and the CLI interfaces. Account passwords can be at most 64 characters long (longer passwords are truncated). Printable ASCII[2] characters except ”space” (ASCII 33-126) are allowed in the password.

Section 7.1.4 provides information on how to proceed in case you forget the **admin** password.

7.1.3 Configuration Files and Reboot

The system keeps three special configuration files:

- *Startup Configuration*: The configuration file used by the switch after system boot or reboot. The *startup configuration* is stored in non-volatile memory (flash)[3].
- *Running Configuration*: The configuration currently used by the switch. The running configuration is kept in volatile memory (RAM).

[2] American Standard Code for Information Interchange (ASCII), see e.g. http://en.wikipedia.org/wiki/ASCII (accessed May 2009).

[3] As described in section 7.1.5, it is possible to keep several configuration files on flash. The startup configuration file is actually a symbolic name for one of the stored configuration files.

The *running configuration* is identical to the *startup configuration* when configuration changes are made via the Web interface, the IPConfig tool or SNMP. That is, when using these methods to manage the switch, a change in the *running configuration* is immediately copied to the *startup configuration*.

In contrast, when managing the switch via the CLI, configuration changes only affect the *running configuration*. Thus, to make CLI changes survive a reboot, you must explicitly copy the running configuration to the startup configuration.

- *Factory Default Configuration*: The system keeps a factory default configuration file. The factory default file is kept in non-volatile memory (flash) and cannot be overwritten. When the switch is shipped, and after factory reset, the startup configuration file is identical to the factory default configuration file.

In addition to these configuration files, it is possible (via CLI) to keep a set of additional configuration files on the switch, which enables easy swapping between alternate configurations.

Important: *Configuring the switch via multiple management interfaces in parallel is discouraged, since it may lead to unexpected behaviour.*

For example, consider the case when two users are accessing the switch at the same time, one user via the CLI and another user via the Web interface:

Assume the "CLI user" makes changes to the running configuration, but of some reason do not wish to copy these changes to the startup configuration (yet).

If the another *user, the "Web user", applies a single change using the web management tool, all the changes done to the running configuration (by the "CLI user") will be saved to the startup configuration. (Actually clicking the* ***Apply*** *button, even without changing any values has the same affect.)*

7.1.3.1 Account password when loading a configuration file

Configuration files contain information on user account and (hashed) passwords, e.g., for the **"admin"** account. Thus, when loading a configuration file to the switch (i.e., overwriting the *startup-configuration* or *running-configuration*), the

account passwords will also be replaced according to the setting in the new configuration file.

> **Warning**: *To copy a new configuration file to the* running-config *or* startup-config *while keeping the existing user names and passwords, the lines in the new configuration file containing the* ***"username"*** *command should be removed before installing the new configuration file.*

If you unintentionally happen to loose the *admin* password because you copied a configuration file including an unknown **admin** password, see section 7.1.4 for information on how to regain access to the switch.

7.1.4 What to do if you cannot access your switch

Occasionally you may end up in a situation where you cannot access your switch:

- *Forgetting IP address*: If you have forgotten what IP address you assigned to your switch, you will no longer be able to access it remotely (Web, SSH, SNMP). Section 7.1.4.1 presents different methods to find the IP address of your switch.
- *Forgetting password*: If you have forgotten the **admin** password you assigned to your switch, you should conduct either a *factory reset* or a *password reset*. Both alternatives require that you have *physical access* to the switch.
  - *Factory Reset*: By resetting the switch to the factory default setting the whole switch configuration (including the **"admin"** password)) will be reset to its default values. That is, the **"admin"** password will be reset to **"westermo"**, thus enabling you to login again.

    The way to accomplish a factory reset may differ if the switch has a console port (section 7.1.4.2) or if it lacks a console port (section 7.1.4.3).
  - *Password Reset*: On switches with a console port there is a possibility to reset the **"admin"** password to its default value (**"westermo"**) without affecting the rest of the configuration, see section 7.1.4.2.
- *Misconfiguration*: You may also loose the ability to access your switch remotely (Web, SSH, SNMP, IPConfig) due to *misconfiguration*, e.g., by disabling all Ethernet ports, or moving them to a VLAN where the switch has no IP address assigned. This case can be resolved by logging into the switch

via the console port, and change the configuration appropriately via the CLI (see chapter 6 on information of how to access the CLI via the console port).

However, if the switch does not have a console port, you may need to conduct a *factory reset* as described in section 7.1.4.3.

7.1.4.1 Discovering the IP address of your switch

By factory default switches are configured with IP address 192.168.2.200 and netmask 255.255.255.0. If you have forgotten what IP address you assigned your switch there are several methods to find it out:

1. *IPConfig (from PC)*: The Westermo IPConfig tool is designed to scan for (Westermo) switches on the local network. See chapter 4 for details on how to use the IPConfig tool. This option is probably the simplest method to find the IP address of a switch, but will not work if IPConfig has been disabled on your switch (see section 7.3.24 for information on how to enable/disable IPConfig on your switch).
2. *IPConfig (from switch)*: The WeOS CLI and the Web contains an IPConfig scanning facility, thus if you are logged into a switch you are to scan for neighbour switches. As in the previous step, switches can only be discovered this way if they have IPConfig enabled.
3. *Via console port*: On switches equipped with a console port, the IP address of the switch can be found using the switch Command Line Interface (CLI). See chapter 6 for more information of how to use the CLI. (If you have forgotten the **admin** password, please see section 7.1.4.2).

In case you are not able to discover the IP address by any of these methods, conducting a factory reset will take the switch back to its original configuration (IP address 192.168.2.200 and netmask 255.255.255.0). See sections 7.1.4.2 and 7.1.4.3 for information on how to conduct a factory reset.

7.1.4.2 Password or Factory Reset via Console Port

For RedFox Industrial and other switches *equipped with a console port*, it is possible to conduct a *factory reset* or just a *password reset* using the special accounts (**factory** or **password**). *For security reasons*, these special accounts can *only be used via the console port*.

- Admin password reset: It is possible to recover from a lost **admin** password by using the following login and password from the console port. The **admin**

password will be reset to its default value (**westermo**), and thereby enable you to login to the switch again.

  - Login: **password**
  - Password: **reset**

- Factory reset: It is possible to reset the switch to factory default settings by using the following login and password from the console port. The whole switch configuration (including the **admin** password) will be reset to its factory default setting.

  - Login: **factory**
  - Password: **reset**

7.1.4.3 Factory Reset without using Console Port

For switches lacking a console port, there is a different mechanism to conduct a factory reset without being logged in. (The method is available also for switches with a console port.)

1. Power off the switch and disconnect *all* Ethernet cables (including copper and fiber cables) or DSL cables.
2. Connect two Ethernet port pairs as described below (for RedFox Industrial it is only one pair). The ports need to be connected directly by Ethernet cables, i.e., **not** via a hub or switch. Use *straight* cables - not *cross-over* cables - when connecting the port pairs.

Product	**Port Pair 1**	**Port Pair 2**
RedFox Industrial	port 1/1 ⇔ port 1/2	Not applicable
RedFox Rail	port X2 ⇔ port X7	port X4 ⇔ port X5
Wolverine DDW-225/226	port 2/1 ⇔ port 2/4	port 2/2 ⇔ port 2/3
Lynx+ (10-Ethernet Ports)	port 3 ⇔ port 10	port 6 ⇔ port 7
Lynx 1400G	port 1 ⇔ port 6	port 2 ⇔ port 5

3. Power on the unit.
4. Wait for the unit to start up. Control that the ON LED is *flashing red*. The ON LED flashing indicates that the unit is now *ready* to be reset to factory default. You now have the choice to go ahead with the factory reset, or to skip factory reset and boot as normal.

- *Go ahead with factory reset*: Acknowledge that you wish to conduct the factory reset by unplugging one of the the Ethernet cable(s). The ON LED will stop flashing.

  This initiates the factory reset process, and after approximately 1 minute the unit will restart with factory default settings. When the switch has booted up, the ON LED will *typically*[4] show a *green* light.

  Note Do not power off the unit while the factory reset process is in progress.

- *Skip the factory reset*: To skip the factory reset process, just wait for approximately 30 seconds (after the ON LED starts flashing RED) without unplugging any of the Ethernet cables. The switch will conduct a normal boot with the existing settings.

The option to reset the **admin** password is only available on units with a console port, see section 7.1.4.2.

7.1.5 Virtual File System

WeOS keeps various files of interest for the operator:

- Configuration files: By default there is only one configuration file (named *config0.cfg* stored on the switch. However, it is possible to create and keep multiple configuration files on the switch, both for backup purposes of for easy shifting between configuration setups. Configuration files are commonly named with the prefix *config* and will always have *'.cfg'* as extension.

  As mentioned in section 7.1.3 there are also three special configuration files:

  - *Running Configuration:* The running configuration is only stored in RAM, thus, it is not kept over a reboot.
  - *Startup Configuration:* The startup config is *mapped* to one of the stored configurations. By default it points to *config0.cfg*, but the mapping can be changed.
  - *Factory Default Configuration:* The factory default configuration file cannot be modified (except through a firmware upgrade). Its available for the purpose of conducting a factory reset.

[4] As the ON LED during normal operation acts as a *summary alarm*, it will show a *green* light if operation is OK, or *red* if an alarm source defined in the (factory default) configuration is triggered. For more information on the ON LED and summary alarm function, see chapter 27. For information on summary alarm settings in your factory default file, use the CLI to list the factory default file as described in section 7.3.6.

- Log files: Events are logged various log files. As of WeOS v4.3.0 the following log files are used:
  - auth.log
  - messages
  - snmpd
  - user.log

For units equipped with a USB port, the operator is also able to access files on a mounted USB stick.

The files are organised in a virtual file system, and are made available both for local and remote access.

	Local File Path	Remote File Path
Configuration files	cfg://	/cfg/
Log files	log://	/log/
USB files	usb://	/usb/

Section 7.1.5.1 describes available methods for file maintenance when logged into the switch, while section 7.1.5.2 covers methods available for maintaining files remotely.

7.1.5.1 File access when logged into the switch

An operator logged in to a switch can copy, download or upload files using the CLI **"copy"** command.

Services available when logged into the system include:

- Making local backup copies of files, e.g.,
  "copy log://messages log://messages.5"
- Upload or download to/from a remote server via TFTP, FTP, and SCP. (Downloading is also available via HTTP.)

  Upload example using TFTP:
  "copy cfg://config0.cfg
  tftp://server.example.com/myswitchconfig.txt"

- Copying between systems: The CLI *copy* command can be used to copy files between remote systems via TFTP, FTP, SCP, and HTTP (HTTP can only be used as source, not destination).

  Example copying from HTTP server to TFTP server:
  ”copy http://server1.example.com/original.txt tftp://server2.example.com/backup.txt”

7.1.5.2 Remote file access

An operator is able to upload and download files to/from the switch remotely via *SCP*. This feature is convenient and saves time, since files can be maintained without the need to log into each switch.

Example with remote file upload:

```
unix> scp config1.cfg admin@redfox.example.com:/cfg/
Password for admin@redfox.example.com:
unix>
```

Example with remote file download:

```
unix> scp admin@redfox.example.com:/log/messages .
Password for admin@redfox.example.com:
unix>
```

7.1.6 Maintenance and diagnostic tools

The switch supports a set of maintenance and diagnostic tools:

Ping and Traceroute The standard Ping and Traceroute commands are available via the CLI and the Web, and are useful as basic troubleshooting tools.

Port monitoring The switch supports *port monitoring*, thus the user can monitor the traffic exchanged on one or more Ethernet ports on a dedicated monitor port. Only *correct* Ethernet packets will be forward onto the monitor destination port. To monitor occurrence of packet drops due to bad CRC, etc., we refer to the RMON statistics counters, see chapter 26.

Note: *To observe all traffic on the monitor source ports, the total amount of traffic on the monitor source ports should not exceed the capacity of the monitor destination port.*

Westermo IPConfig Client As described in chapter 4 Westermo provides the *IPConfig PC tool* for discovery and rudimentary management of Westermo switches. The CLI and the Web provides a similar mechanism, i.e., once logged into the switch, it is possible to scan for other Westermo on the same LAN.

Additional features relevant for maintenance and diagnostics are described in chapter 26 (RMON Statistics), chapter 28 (Event and Alarm Logging), chapter 29 (SNMP), and chapter 27 (Alarm handling, Digital I/O and Front-panel LEDs).

7.2 Maintenance via the Web Interface

7.2.1 Account Management via the Web Interface

Menu path: Maintenance ⇒ Password

The only account management feature in the web management tool at the moment is change of the *admin* password.

New Password	Enter the new password for the *admin* account.
Repeat New Password	To minimise risk of typing error, enter the new password for the *admin* account once again.

7.2.2 Managing switch firmware via the Web Interface

Menu path: Maintenance ⇒ F/W Upgrade

On the firmware upgrade page you are able to upgrade firmware by downloading an image using FTP/TFTP or by direct upload via the Web browser.

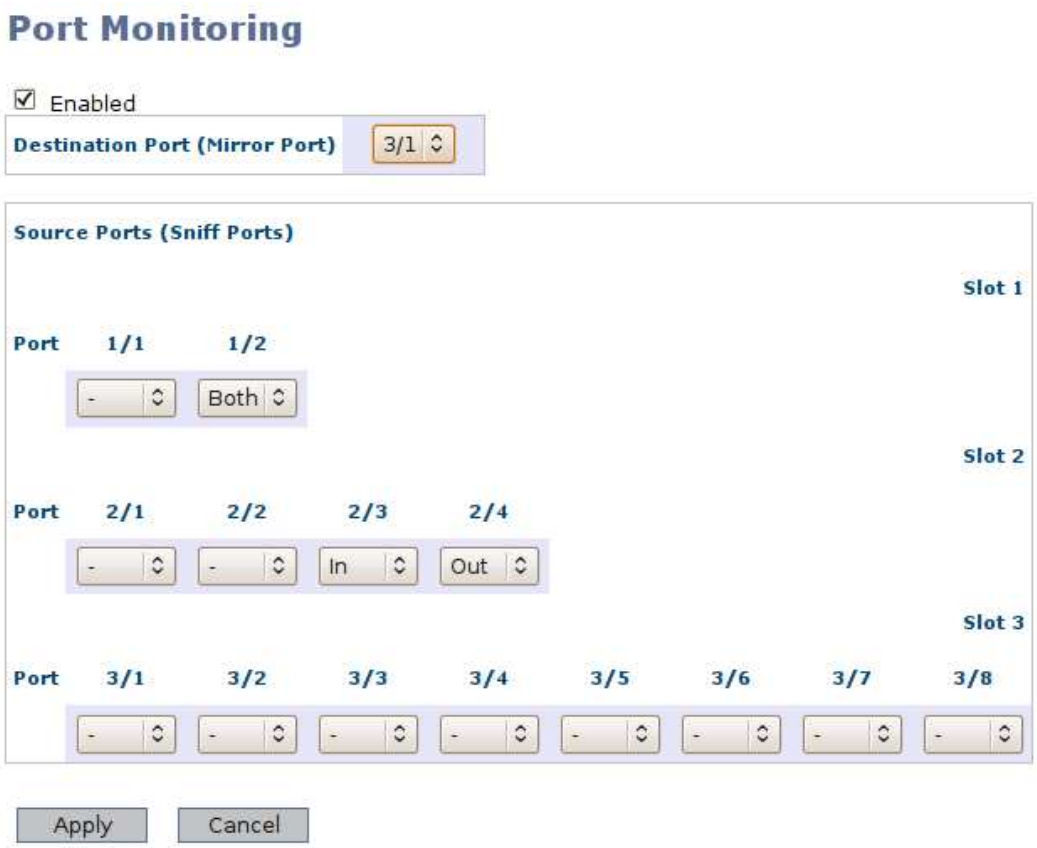
7.2.2.1 Firmware Upgrade Using File Upload

Image File	Select the file to upload (browser dependent).
Upgrade	Click the **Upgrade** button to initiate firmware upgrade.

7.2.2.2 Firmware Upgrade Using TFTP/FTP Server

Image name	The file name of the image file on the FTP/TFTP server.
Server address	The IP address of the FTP/TFTP server.
Upgrade	Click the **Upgrade** button to initiate firmware upgrade.

7.2.3 Port Monitoring



Enabled	Check the box to enable port monitoring. If you have a JavaScript enabled browser the other settings will not be displayed unless you check this box.
Destination Port (Mirror)	Select one port to which data from source ports will be copied (mirrored).
Source Ports (Sniff Ports)	Select one or more ports to monitor by selecting the ports desired sniff mode. Available modes are: **In** Inbound (ingress) traffic. **Out** Outbound (egress) traffic. **Both** Both inbound and outbound traffic.

7.2.4 Backup and Restore

Menu path: Maintenance ⇒ Backup&Restore

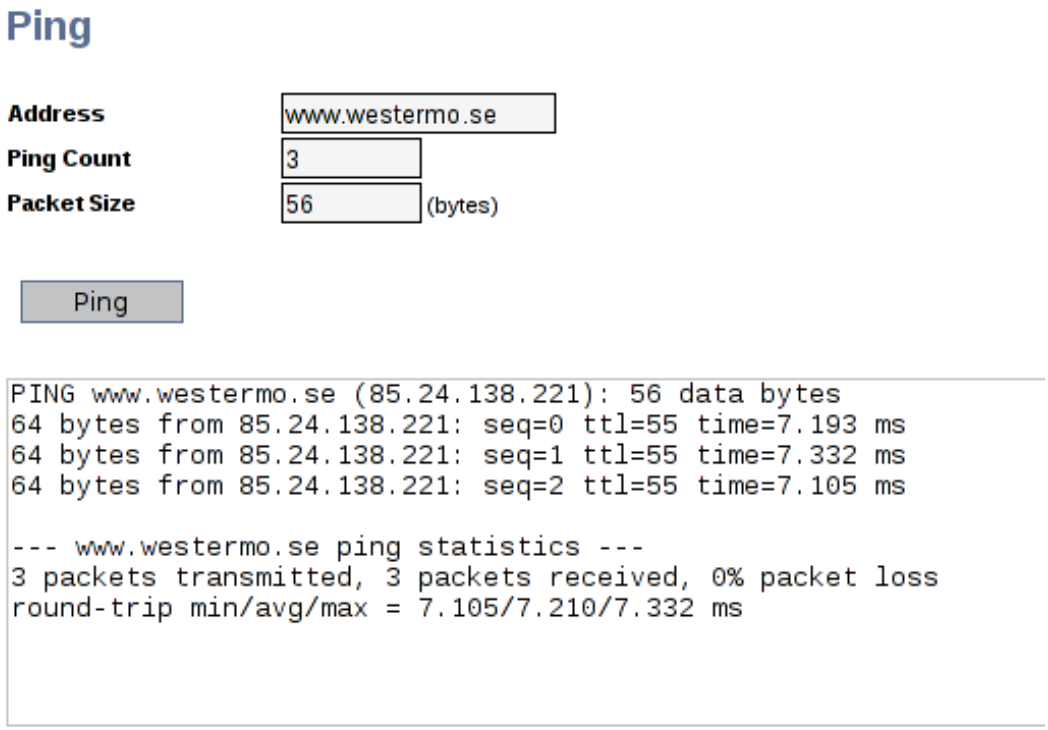
To create a backup of your switch configuration on your host, visit the *backup and restore* page.

Backup	Click this button to download a copy of the running configuration on your switch. You will be asked to open or save the file. Normally chose *save* to save the file to your host. The behaviour is web browser specific and may also depend on your current browser settings. See Fig. 7.1 for an example.
File Path	Click the **Browse** button to browse for the file. The behaviour of the file selection is browser specific.
Restore	Click this button to restore the configuration the configuration described in the file you selected in *File Path*.

Figure 7.1: Example save dialogue (this example is from a Firefox browser)

7.2.5 Ping tool

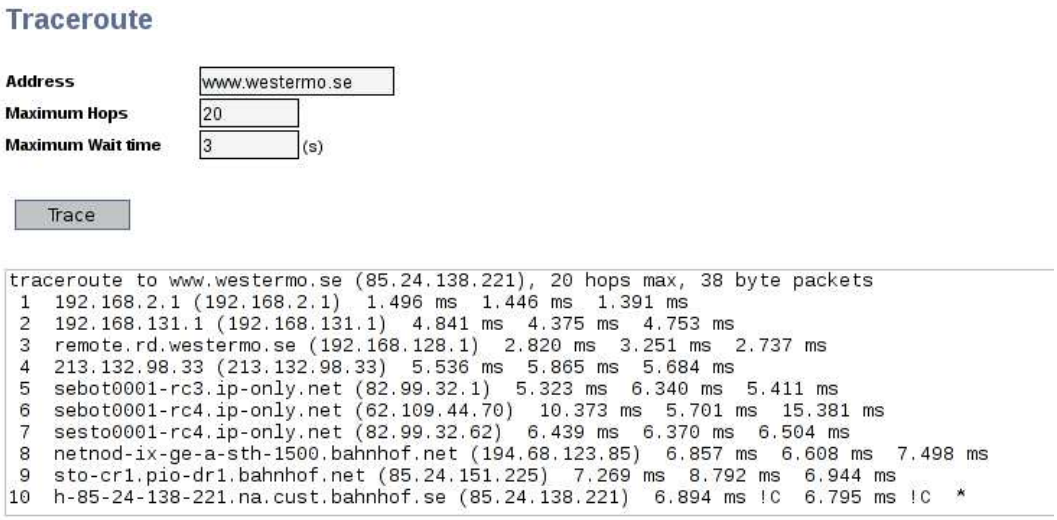
Ping is useful as a basic diagnostic tool. The output on the web is displayed once the ping command has completed. If the command takes too long to execute the web page may time out.



Address	The network host to send ICMP ECHO REQUEST packets to
Ping Count	Defines the number of ICMP packets to send.
Packet Size	Alters the default size of the ICMP packets. This only only increases the empty payload of the packet

7.2.6 Traceroute tool

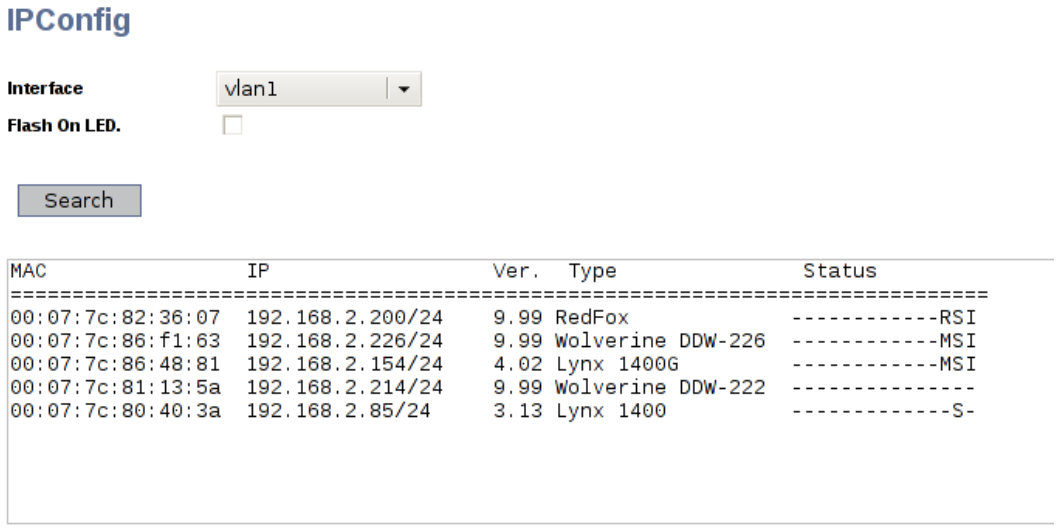
Trace the route packets take to a network host. The output on the web is displayed once the ping command has completed. If the command takes too long to execute the web page may time out.



Address	The network host
Maximum Hops	Max time-to-live (number of hops).
Maximum Wait time	Set the delay, in seconds, before timing out a probe packet

7.2.7 IPConfig scan tool

Scan network for IPConfig neighbours. The output on the web is displayed once the ping command has completed. If the command takes too long to execute the web page may time out.



Interface	The interface to scan
Flash On LED.	If enabled, this unit will flash the on LED, while scanning

7.3 Maintenance via the CLI

CLI commands for general switch maintenance are listed below:

Command	Default	Section
Firmware Upgrade		
upgrade <pri|sec|boot> <IPADDR> <FILENAME>		Section 7.3.1
show system-information		Section 7.3.2
Configuration Files and Reboot		
dir <cfg://|log://|usb://>		Section 7.3.3
copy <FROM_FILE> <TO_FILE>		Section 7.3.4
erase <file>		Section 7.3.5
show <running-config | startup-config | factory-config | [<filesys>://]FILENAME>		Section 7.3.6
reboot		Section 7.3.7
Account management		
aaa		Section 7.3.8
username <USERNAME> [hash] <PASSWORD>		Section 7.3.9
show		Section 7.3.10
show username <USERNAME>		Section 7.3.11
Maintenance and Diagnostic tools		
ping <IPADDR>		Section 7.3.12
traceroute <IPADDR>		Section 7.3.13
ssh [USER@]<IPADDR|DNAME>[/PORT]	admin/22	Section 7.3.14
show ipconfig <IFNAME>		Section 7.3.15
monitor		Section 7.3.16
[no] enable	Disabled	Section 7.3.17
destination <PORT>		Section 7.3.18
source <PORTLIST>		Section 7.3.19
show monitor		Section 7.3.20
monitor		
show mirror		Section 7.3.21
show ports		Section 7.3.22

Continued on next page

Command	Default	Section
Controlling Management Services		
[no] web	Enabled	Section 7.3.23
[no] ipconfig	Enabled	Section 7.3.24
show web		Section 7.3.25
show ipconfig		Section 7.3.26
Other maintenance commands		
date		Section 8.2.6
[no] timezone <TIMEZONE>		Section 8.2.5
show date		Section 8.2.12
show timezone [QUERY|SUBSTRING]		Section 8.2.11
show env		Section 7.3.27
show uptime		Section 7.3.28
show memory		Section 7.3.29
show processes		Section 7.3.30

7.3.1 Upgrading firmware

Syntax `upgrade <pri|sec|boot> <IPADDR> <FILENAME>`

Context *Admin Exec*

Usage Upgrade primary, secondary, or bootloader firmware via FTP or TFTP. The command first attempts to download and install *FILENAME* via FTP from a server at *IPADDR*. If no FTP server is available, the command tries to download the file using TFTP instead. After installing a *primary firmware*, the switch will automatically be rebooted.

(More precisely: after installing a *primary firmware*, the switch will automatically be rebooted given that the system booted from the primary image. Similarly, after installing a *secondary firmware*, the switch will automatically be rebooted given that the system booted from the secondary image.)

Caution! Only conduct upgrades over a stable network connection. Ensure that the switch is not powered off while the downloaded firmware is being installed.

Default values N/A

Error messages None defined yet

Example **”`upgrade primary 192.168.1.1 rx100.img`”** will download and install a new primary image (*rx100.img*) from FTP/TFTP server at *192.168.1.1*.

”upgrade boot 192.168.1.1 redboot-1.6.bin” will download and install a new bootloader image (*redboot-1.6.bin*) from a FTP/TFTP server with *192.168.1.1*.

7.3.2 Show System Information

Syntax `show system-information`

Context *Admin Exec*

Usage List general system information such as serial number, firmware version, contained hardware, etc.

Default values Not applicable

Error messages None defined yet

Example

```
redfox:/#> show system-information

System Information
================================================================================

System Name        : redfox
System Contact     : support@westermo.se
System Location    : westermo

Product Family     : RedFox          Model              : Industrial
Article number     : 3641-3100-0     Channel interfaces : 3 (0-2)
Serial Number      : 3567            Base MAC Addr      : 00:07:7c:82:09:6c

Boot loader ver.   : 1.13            Main FPGA ver.     : 20080626
Main firmware ver. : 9.89            Backup firmware ver: 9.99

Card #1 ========================================================================
Type               : CPU (1)
Article no         : 5013-0000
Revision           : 0
Batch id           : 081211-00012345-00000

Card #2 ========================================================================
Type               : 10/100TX (17)
Article no         : 5013-0100
Revision           : 0
Batch id           : 081211-00012345-00000

Card #3 ========================================================================
Type               : 10/100TX (17)
Article no         : 5013-0100
Revision           : 0
Batch id           : 081211-00012345-00000
redfox:/#>
```

7.3.3 List Configuration and Log Files

Syntax `dir [<cfg://|log://|usb://>]`

Context *Admin Exec*

Usage List files in the configuration file directory, log file directory, or files on a mounted USB memory. When listing configuration files you should be able to see which of the present configuration files that is used as startup file. To map a different configuration file as startup configuration, see the **"copy"** command (section 7.3.4).

Default values `cfg://`

Error messages None defined yet

Example

```
redfox:/#> dir
================================================================================
Contents of Config File System
================================================================================
              config0.cfg --> startup-config
              config1.cfg

redfox:/#>
```

7.3.4 Copy, Store or Restore a Configuration File

Syntax `copy <FROM_FILE> <TO_FILE>`

Several methods are available to specify <FROM_FILE> and <TO_FILE>. Local file access methods are listed below:

- Configuration files (default): **"cfg://<FILENAME>"**
- Special configuration files: **"running-config"**, **"startup-config"**, and **"factory-config"**.
- Log files: **"log://<FILENAME>"**
- USB memory: **"usb://[DIRECTORY/]<FILENAME>"**

Remote file access methods:

- TFTP: **"tftp://location[/directory]/filename"**

- FTP: **"ftp://[username[:password]@]location[:PORT] [directory]/filename"**
  If no username is provided, anonymous ftp login will be used. Default password is **"info@westermo.se"**.
- SCP: **"scp://[username@]location[:PORT][/directory] /filename"**
  By default username **"admin"** will be used.
- HTTP: **"http://location[:PORT][/directory]/filename"**

Context *Admin Exec*

Usage Copy, upload or download of configuration files, log files, etc. Files can be copied *any* to *any*, i.e., local⇒local, local⇒remote, remote⇒local, and(!) remote⇒remote. The HTTP method can only be used as *FROM_FILE*.

- Use **"copy running-config startup-config"** to make the running configuration survive a reboot.
- Copying a remote network config file directly to the the startup config is not recommended. It is strongly encouraged to copy the file into *running-config* first, inspect the settings and then save it using **"copy running-config startup-config"**.
- By default the startup-configuration is mapped to configuration file *config0.cfg*. It can be mapped to another config file (here config5.cfg) as follows:
  "copy config5.cfg startup-configuration"

Default values N/A

Error messages None defined yet

Examples

1. Restore factory default (to running configuration)

```
redfox:/#> copy factory-config running-config
Using default factory.cfg found in firmware image.
Stopping Syslog daemon ..................................... [ OK ]
Starting Syslog daemon ..................................... [ OK ]
redfox:/#>
```

2. Store running configuration to startup configuration

```
redfox:/#> copy running-config startup-config
redfox:/#>
```

3. Copy configuration file from USB to local configuration file *config3*.


```
redfox:/#> copy usb://myconfig.cfg config3
Copying myconfig.cfg to config3 ...
Done.
redfox:/#>
```

4. Copy configuration file onto remote server using FTP.

```
redfox:/#> copy cfg://config0.cfg ftp://mylogin:mypw@192.168.2.99/myconfig
redfox:/#>
```

7.3.5 Delete a Configuration File

Syntax `erase [filesys://]<FILENAME>`

filesys can be **"cfg"**, **"log"**, or **"usb"**, with **"cfg"** as default.

Context *Admin Exec*

Usage Delete a configuration file, log file or a file on a mounted USB memory.

Default values **"cfg"** is the default file system.

Error messages None defined yet

Example

```
redfox:/#> dir
================================================================================
                        Existing Configurations on System
================================================================================
                  config0 --> startup-config
                  config1

redfox:/#> erase config1
redfox:/#> dir
================================================================================
                        Existing Configurations on System
================================================================================
                  config0 --> startup-config

redfox:/#>
```

7.3.6 Show Configuration File

Syntax `show <running-config|startup-config|factory-config|[<filesys>://]<FILENAME>`

filesys can be **"cfg"**, **"log"**, or **"usb"**, with **"cfg"** as default.

Context *Admin Exec*

Usage Show content of a configuration file, log file, or file on a mounted USB memory. Special files are *running-config*, *startup-config* and *factory-config*. Use the **"dir"** command to list files (section 7.3.3).

Default values **"cfg"** is the default file system.

Error messages None defined yet

7.3.7 Rebooting the Device

Syntax `reboot`

Context *Admin Exec*

Usage Reboot the device. The switch will boot up with its *startup-config*.

Default values Not applicable.

Error messages None defined yet

7.3.8 Manage AAA Settings

Syntax `aaa`

Context *Global Configuration*

Usage Enter Authentication, Authorisation and Accounting (AAA) context. The AAA context is used for managing user account settings, etc.

Default values Not applicable.

Error messages None defined yet.

7.3.9 Changing Account Password

Syntax `username <USERNAME> [hash] <PASSWORD>`

Context *AAA* context

Usage Change password of a certain user account, e.g., the **"admin"** account. By default, the password is entered as clear-text, and saved as a hash.

The **"hash"** keyword is not intended to be used by regular users - instead it is used by the switch itself when reading a configuration file including a

hashed password. By adding the **”hash”** keyword, the system expects that a hashed password is entered (as opposed to a clear-text password).

Default values Password is entered in clear-text.

Error messages None defined yet

Example Setting the **”admin”** password to ”foobar”.

```
redfox:/config/aaa/#> username admin foobar
redfox:/config/aaa/#>
```

7.3.10 Show AAA Settings

Syntax `show aaa` Also available as **”show”** command within the AAA context.

Context *AAA* context

Usage Show overview of AAA settings.

Default values Not applicable.

Error messages None defined yet.

7.3.11 Show Account Password Hash

Syntax `show username <USERNAME>`

Context *Global Configuration*

Usage Show hashed password for the specified user.

Default values Not applicable.

Error messages None defined yet

7.3.12 Ping

Syntax `ping [-i <IFACE|IPADDR>] [-c <count>] [-s <size>] <HOST>`

Context *Admin Exec* context

Usage Ping a remote host.

Ping is useful as a basic diagnostic tool.

The -i option can be used to select the interface to send ICMP_ECHO on, which is useful in, e.g., VPN setups. The -i option can also be used with an IP address to spoof the source IP address.

You can use use the domain name or IP address as the host argument, but you need a valid name server setup for domain names to work, see section 17.4.5.

Default values Not applicable.

Error messages None defined yet

Example

```
redfox:/#> ping 192.168.131.1
Ctrl-C to abort PING 192.168.131.1 (192.168.131.1): 56 data bytes
64 bytes from 192.168.131.1: seq=0 ttl=64 time=4.832 ms
64 bytes from 192.168.131.1: seq=1 ttl=64 time=0.836 ms
64 bytes from 192.168.131.1: seq=2 ttl=64 time=0.810 ms
64 bytes from 192.168.131.1: seq=3 ttl=64 time=0.823 ms

--- 192.168.131.1 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.810/1.825/4.832 ms
redfox:/#>
```

7.3.13 Traceroute

Syntax `traceroute <HOST>`

Context *Admin Exec* context

Usage Trace the path the packets take to a remote host.

Traceroute is useful as a basic diagnostic tool.

You can use use the domain name or IP address as the host argument, but you need a valid name server setup for domain names to work, see section 17.4.5.

Default values Not applicable.

Error messages None defined yet

Example

```
redfox:/#> traceroute 192.168.130.41
traceroute to 192.168.130.41 (192.168.130.41), 30 hops max, 40 byte packets
 1  192.168.131.1  1.116 ms  0.755 ms  0.806 ms
 2  192.168.130.41  0.824 ms  0.705 ms  0.742 ms
redfox:/#>
```

7.3.14 Remote Login to another device (SSH Client)

Syntax `ssh [USER@]<IPADDR|DOMAINNAME>[/PORT]`

Context *Admin Exec* context.

Usage Login to remote device using SSH.

Default values Default user **"admin"**, default (TCP) port number **"22"**.

Error messages None defined yet.

7.3.15 Show IPConfig Neighbours

Syntax `show ipconfig <IFNAME>`

Context *Admin Exec* context.

Usage Scan network for IPConfig neighbours.

Default values Not applicable.

Error messages None defined yet.

7.3.16 Manage Port Monitoring

Syntax `monitor`

Context *Admin Exec* context

Usage Enter the port monitoring context

Default values Not applicable.

Error messages None defined yet

7.3.17 Enable/disable Port Monitoring

Syntax `[no] enable`

Context *Port monitoring* context

Usage Enable port monitoring. Use **"`no enable`"** to disable port monitoring.

Default values `no enable` (disabled)

Error messages None defined yet

7.3.18 Set Mirror Port

Syntax `[no] destination <PORT>`

Context *Port Monitoring* context

Usage Set the monitor destination port, i.e., the *mirror* port.

Default values Not applicable.

Error messages None defined yet

7.3.19 Set Monitored Ports

Syntax `[no] source <PORTLIST> [ingress] [egress]`

Context *Port Monitoring* context

Usage Add/delete/update monitor source port(s), i.e., the ports being *monitored*.

Default values By default there are no source ports. Commands apply both to `ingress` and `egress` if neither is specified.

Error messages None defined yet

7.3.20 Show Port Monitoring Settings

Syntax `show monitoring`

Context *Admin Exec* context. Also available as **"`show`"** command within the Port Monitoring context.

Usage Show port monitoring configuration.

Default values Not applicable.

Error messages None defined yet.

7.3.21 Show Monitor Destination Port

Syntax `show mirror`

Context *Port Monitoring* context.

Usage Show configured port monitoring destination port, i.e., the port to which traffic is mirrored.

Default values Not applicable.

Error messages None defined yet.

7.3.22 Show Monitor Source Ports

Syntax `show ports`

Context *Port Monitoring* context.

Usage Show configured port monitoring source ports, i.e., the list of ports being monitored, and if monitoring is being done for ingress or egress traffic, or for both.

Default values Not applicable.

Error messages None defined yet.

7.3.23 Enable/disable Web Management Interface

Syntax `[no] web`

Context *Global Configuration* context.

Usage Enable web management interface, and enter *Web* context. (The *Web* context currently does not include any additional configuration options.) Use **"no web"** to disable the web server (**warning**: The switch cannot be managed via the Web interface).

Default values Enabled (**"web"**)

Error messages None defined yet.

7.3.24 Enable/disable IPConfig Management Interface

Syntax `[no] ipconfig`

Context *Global Configuration* context.

Usage Enable IPConfig management interface, and enter *IPConfig* context. (The *IPConfig* context currently does not include any additional configuration options.) Use **"no ipconfig"** to disable the IPConfig server (**warning**: After this the switch cannot be managed (or detected) using IPConfig).

Default values Enabled (**"ipconfig"**)

Error messages None defined yet.

Examples

1. How to check whether IPConfig is enabled on my switch:
   - Alternative 1: Use **"show running"** in Admin Exec context. In the output, look for a line saying **"ipconfig"** (IPConfig enabled) or **"no ipconfig"** (IPConfig disabled).
   - Alternative 2: Enter Global Configuration context and check IPConfig configuration, e.g.:

     ```
     redfox:/#> config
     redfox:/config/#> show ipconfig
     Ipconfig is enabled
     redfox:/config/#> end
     ```

2. How to enable/disable IPConfig:
   Enter Global Configuration context, check the current IPConfig configuration, and modify it if desired. Below is an example of how to enable IPConfig.

   ```
   redfox:/#> config
   redfox:/config/#> show ipconfig
   No active ipconfig configuration available.
   redfox:/config/#> ipconfig
   Activating ipconfig with default settings, type 'abort' to cancel.
   redfox:/config/ipconfig/#> end
   redfox:/config/#> end
   redfox:/#> Starting IPConfig daemon ................................. [ OK ]

   redfox:/#>
   ```

7.3.25 Show Web Management Interface Setting

Syntax `show web`

Context *Global Configuration* context. Also available as **"show"** command within the *Web* context.

Usage Show whether the Web server is enabled or disabled.

Default values Not applicable.

Error messages None defined yet.

7.3.26 Show IPConfig Management Interface Setting

Syntax `show ipconfig`

Context *Global Configuration* context. Also available as **"show"** command within the *IPConfig* context.

Usage Show whether the IPConfig server is enabled or disabled.

Default values Not applicable.

Error messages None defined yet.

7.3.27 Show System Environment Sensors

Syntax `show env`

Context *Admin Exec* context.

Usage List available environment sensors, their index, and their current value. Examples of sensors are *power* (DC1 and DC2), Digital In, and Temperature sensors.

Default values Not applicable.

Error messages None defined yet.

7.3.28 Show System Uptime

Syntax `show uptime`

Context *Admin Exec* context.

Usage Show system uptime.

Default values Not applicable.

Error messages None defined yet.

7.3.29 Show Memory Usage

Syntax `show memory`

Context *Admin Exec* context.

Usage Show system memory usage.

Default values Not applicable.

Error messages None defined yet.

7.3.30 Show Running Processes

Syntax `show processes`

Context *Admin Exec* context.

Usage Show a list of currently running processes.

Default values Not applicable.

Error messages None defined yet.

Chapter 8

Switch Identity Information

WeOS provides management of a set of features related to *system identity*. The table below gives a summary of the features available for the different management interfaces.

System *hostname*, *location* and *contact* correspond the associated system objects of the original MIB-2 standard MIB (RFC 1213). For more information on WeOS SNMP support, see chapter 29.

Feature	Web	CLI
Set System Hostname	X	X
Set System Location	X	X
Set System Contact	X	X
Set System Time Zone	X[1]	X
Set System Date/Time	X	X
View System Identity Settings	X	X
View System Date/Time	X	X

Section 8.1 covers management of system identity features via the Web interface, and section 8.2 describes the corresponding features in the CLI.

[1] Web configuration of System Time Zone is done as part of the Network settings, see section 17.2.

8.1 Managing switch identity information via the web interface

8.1.1 Manage System Identity Information

Menu path: Configuration ⇒ Identity

Fig 8.1 shows the page where you can set hostname, location and contact information for your switch.

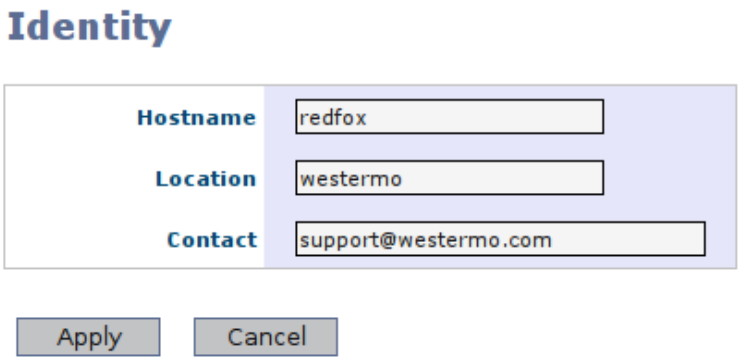


Figure 8.1: Switch identity settings (this example is from a RedFox Industrial switch)

Hostname	A name to identify this unit. Max 64 characters. Valid characters are A-Z, a-z, 0-9, and hyphen (-). The first character should be alphabetic (A-Z, a-z). Hyphen is not valid as first or last character.
Location	A description to identify where the unit is located. Max 64 characters. Valid characters are ASCII 32-126 except '#' (ASCII 35). "Space" (ASCII 32) is not valid as first or last character.
Contact	A description identifying whom to contact regarding management of the unit. Max 64 characters. Valid characters are ASCII 32-126 except '#' (ASCII 35). "Space" (ASCII 32) is not valid as first or last character.

Change the values to appropriate values for your switch and click the **Apply** button.

8.1.2 Set System Date and Time

Menu path: Maintenance ⇒ Date & Time

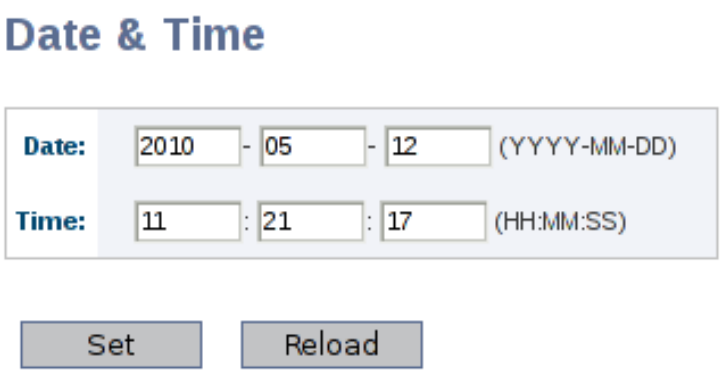


Figure 8.2: Switch date and time settings

8.2 Managing switch identity information via CLI

Command	Default	Section
Configure Identity Settings & Date/Time		
system		Section 8.2.1
hostname <ID>	redfox[1]	Section 8.2.2
location <ID>	westermo	Section 8.2.3
contact <ID>	support@westermo.se	Section 8.2.4
[no] timezone <TIMEZONE>		Section 8.2.5
date		Section 8.2.6
View Identity Settings & Date/Time		
show system		Section 8.2.7
system		
show hostname		Section 8.2.8
show location		Section 8.2.9
show contact		Section 8.2.10
show timezone [QUERY| SUBSTRING]		Section 8.2.11
show date		Section 8.2.12

8.2.1 Manage System Identity Information

Syntax `system`

Context *Global Configuration* context

Usage Enter system identity configuration context.

Default values Not applicable

Error messages None defined yet

8.2.2 System Hostname

Syntax `hostname <STRING>`

Context *system* context

[1] The default hostname will depend on the type of product WeOS runs on.

Usage Set system hostname string.

Max 64 characters. Valid characters are A-Z, a-z, 0-9, and hyphen (-). The first character should be alphabetic (A-Z, a-z). Hyphen is not valid as first or last character.

Default values redfox (The default hostname will depend on the type of product WeOS runs on.)

Error messages None defined yet

8.2.3 System Location

Syntax `location <STRING>`

Context *system* context

Usage Set system location string.

Max 64 characters. Valid characters are ASCII 32-126 except '#' (ASCII 35). "Space" (ASCII 32) is not valid as first or last character.

Default values westermo

Error messages None defined yet

8.2.4 System Contact

Syntax `contact <STRING>`

Context *Global Configuration* context

Usage Set system contact string.

Max 64 characters. Valid characters are ASCII 32-126 except '#' (ASCII 35). "Space" (ASCII 32) is not valid as first or last character.

Default values support@westermo.se

Error messages None defined yet

8.2.5 Set System Time Zone

Syntax `[no] timezone <TIMEZONE>`

Context *system* context.

Usage Set system time zone string. For information of available time zone settings, see section 8.2.11.

Default values Disabled (**"timezone"**)

Error messages None defined yet.

8.2.6 Set System Date and Time

Syntax `date [[YYYY-MM-DD ]hh:mm[:ss]]`

Context *Admin Exec* context.

Usage Set system date and time, or only time.

Default values If no date or time is given, the current date and time will be displayed (same as **"show date"**, see section 8.2.12).

Error messages None defined yet.

8.2.7 Show System Identity Information

Syntax `show system`

Also available as **"show"** command within the system identify context.

Context *Global Configuration* context

Usage Show system hostname, location, contact and Time Zone settings.

Default values See sections 8.2.2-8.2.5

Error messages None defined yet

8.2.8 Show System Hostname

Syntax `show hostname`

Context *system* context

Usage Show system hostname string.

Default values Not applicable

Error messages None defined yet

8.2.9 Show System Location

Syntax `show location`

Context *system* context

Usage Show system location string.

Default values Not applicable

Error messages None defined yet

8.2.10 Show System Contact

Syntax `show contact`

Context *system* context

Usage Show system contact string.

Default values Not applicable

Error messages None defined yet

8.2.11 Show System Time Zone

Syntax `show timezone [QUERY|SUBSTRING]`

Context *system* context.

Usage Show system time zone setting/list available time zones.

When given without any argument (**”show timezone”**), the configured time zone setting is presented.

When providing an argument, the available time zone settings matching that argument is listed, e.g., issuing the command **”show timezone asia”** will list all possible time zone configuration settings for Asia (or more precisely, all available time zones containing the substring ’asia’.) See section 8.2.5 for information of how to set the system time zone.

Default values Not applicable.

Error messages None defined yet.

8.2.12 Show System Date and Time

Syntax `show date`

Context *Admin Exec* context.

Usage Show system date and time.

Default values Not applicable.

Error messages None defined yet.

Chapter 9

Ethernet Port Management

By default all ports on the switch are enabled. Section 9.1 provides general information about the available port settings. Section 9.2 covers port settings via the Web interface and section 9.3 port settings via the CLI.

9.1 Overview of Ethernet Port Management

Feature	Web (Sec. 9.2)	CLI (Sec. 9.3)	General Description
Enable/disable port	X	X	
Speed-duplex mode	X	X	Sec. 9.1.1
Flow control	X	X	Sec. 9.1.2
Port priority (level)	X	X	Sec. 9.1.3
Port priority mode	X	X	Sec. 9.1.3
Link alarm	X	X	Sec. 9.1.4
Inbound rate limit	X	X	Sec. 9.1.5
Outbound traffic shaping	X	X	Sec. 9.1.5
MDI/MDIX	X	X	Sec. 9.1.6
Fall-back default-VID		X	Sec. 9.1.7
View port configuration	X	X	
View port status	X	X	

The table above presents available port settings. The features are presented further in the following sections.

9.1.1 Port speed and duplex modes

By default ports are configured to auto-negotiate speed (10/100/1000 Mbit/s) and duplex modes (half/full) to the ”best” common mode when a link comes up. When configured for auto-negotiation, the resulting speed and duplex mode agreed is shown as part of the port status information.

It is possible to disable auto-negotiation and instead use a static speed and duplex mode setting. When using a static speed and duplex setting, the operator should ensure that the ports on both ends of the link are configured with the same static speed and duplex settings.

Depending on Ethernet port *type*, the available port speeds will differ:

- Fast Ethernet copper ports: Fast Ethernet copper ports are capable to operate at 10 or 100 Mbit/s.
- Gigabit Ethernet copper ports: Gigabit Ethernet copper ports are capable to operate at 10, 100 or 1000 Mbit/s.
- Gigabit Ethernet fibre ports: Gigabit Ethernet fibre ports are capable to operate at 1000 Mbit/s.

9.1.2 Flow control

The ports can be configured to use *flow control*, i.e., to dynamically limit inbound traffic to avoid congestion on outbound ports.

When flow control is enabled on a *full duplex* port, the switch will send *pause frames* (IEEE 802.3x) to limit inbound traffic on this port, if that traffic is causing congestion when sent out on another switch port.

When flow control is enabled on a *half duplex* port, the switch will use a technique known as *back-pressure* to limit inbound traffic on this port, if that traffic is causing congestion when sent out on another switch port. (The *back-pressure* technique enables a switch to force its neighbour to slow down by sending *jamming signals* on that port, thus emulating a packet collision.)

9.1.3 Layer-2 priority support

Each Ethernet port has four output queues, enabling layer-2 priority support with four traffic classes. The queues are serviced according to *strict priority scheduling*, i.e., when there are traffic in multiple queues, the packets in the queue with higher priority is serviced first.

A packet's priority is determined when it enters on a port, and can be classified based on:

- VLAN ID: The switch can be configured to give specific priority to certain VLANs. This can be useful to, e.g., when providing IP telephony via a dedicated VLAN. Priority based on VLAN ID has precedence over all priority classifications described below.

  VLAN ID priority settings are further described in chapter 13.

- VLAN tag: For packets carrying a VLAN tag, the packet's priority can be based on content of the priority bits inside the VLAN tag. The VLAN tag is useful to carry packet priority information on inter-switch links.

  Use of VLAN tag priority can be configured per port (see sections 9.2 and 9.3).

- IP ToS/DiffServ: For IP packets the priority can be classified based on the content of the IP ToS bits (IPv4) or the IP TC bits (IPv6). Classification based on the IP ToS/Diffserv bits can be useful to provide higher priority to delay sensitive applications, such as *IP telephony* and *remote login*, than to bulk data applications, such as *file transfer*, however, it requires that those applications can set the IP ToS/Diffserv bits appropriately.

  Use of IP ToS/DiffServ priority can be configured per port (see sections 9.2 and 9.3).

- Port Priority: Priority can be classified based on the inbound port.

  Use of port priority can be configured per port (see sections 9.2 and 9.3). Furthermore, when priority classification is configured to be based on VLAN tag (or IP ToS/DiffServ), priority will be based on the port priority for untagged (or non-IP respectively) packets.

When priority is classified based on VLAN ID, VLAN tag, or port priority, the priority assigned to a packet will take a value in range 0-7, and be represented by 3 bits (IEEE 802.1p). The mapping of 802.1p priority (8 values) to traffic class (4 output queues) is shown in table 9.1. The rationale behind this mapping is described in IEEE 802.1Q-2005 (Annex G).

When priority is classified based on IP ToS/DiffServ, the priority assigned to a packet will take a value in range 0-63, and be represented by 6 bits (DSCP - Differentiated Services Code Point). The mapping of DSCP priority (64 values) to traffic class (4 output queues) is shown in table 9.2. This mapping is inline with the use of IP Precedence fields (RFC 1349), and IP DiffServ for *best effort* and *control* traffic (RFC 2474), *assured forwarding* (RFC 2597) and *expedited forwarding*

IEEE 802.1p priority	Queue number/ Traffic Class
0 1	0 (lowest) 0
2 3	1 1
4 5	2 2
6 7	3 3 (highest)

Table 9.1: Mapping of IEEE 802.1p priority to Queue/Traffic Class.

IP Priority bits						Queue bits		Queue number/ Traffic class
5	4	3	2	1	0	1	0	
0	0	-	-	-	-	0	0	0 (lowest)
0	1	-	-	-	-	0	1	1
1	0	-	-	-	-	1	0	2
1	1	-	-	-	-	1	1	3 (highest)

Table 9.2: Mapping of IP priority bits to Queue/Traffic Class.

(RFC 3246).

Packets sent out on a port *with* a VLAN tag will carry priority information (802.1p) within their VLAN tag.

- For packets where priority was classified based on VLAN ID, VLAN tag, or port priority, the outbound priority (3 bits) will be equal to the determined inbound priority (3 bits).
- When priority is classified based on IP ToS/DiffServ, determining the outbound priority (3 bits) is more complex: the two most significant bits of the outbound priority will be equal to the queue number (i.e., queue bits in table 9.2), while the least significant bit of the outbound priority is equal to the least significant bit of the inbound port's configured port priority.

  E.g., if the packet is put in priority queue 2 (binary '10'), and the port priority of the inbound port has an odd value (least significant bit is '1'), the packet

will carry priority value 5 (’101’) in its VLAN tag when sent on the outbound port.

Warning: *Configuration of layer-2 priority should be handled with care. In particular, mapping user traffic to the highest priority queue is discouraged, since that may affect time critical control traffic, such as FRNT traffic, already mapped to the highest priority queue. For more detailed guidelines of layer-2 priority handling, we refer to Westermo application notes, and IEEE standards 802.1D-2004 (Annex G) and 802.1Q-2005 (Annex G).*

9.1.4 Link alarm

Each Ethernet port on the switch can be configured to indicate alarm when the link comes up or goes down. The alarm is indicated in multiple ways:

- *SNMP trap:* An SNMP trap will be sent when a link changes state, i.e., both when the link comes up, or when it goes down. This assumes that SNMP is enabled, and that a trap host is configured. See chapter 29 for more information.
- *Front panel LEDs:* A link alarm may effect both the individual LED of the port, as well as the common status LED for the switch (for definite information about what functions affect the common status LED, see chapter 27):
  - *Individual LED:* Each Ethernet port has a LED, which generally indicates ’green’ if the link is up. If there is no link, the LED will indicate ’yellow’ when link alarm is configured.
  - *Common status LED:* The switch has a common status LED, labelled ’ON’ on the front panel. This LED will generally indicate ’green’ if all associated functions are OK, and ’red’ if one or more of the associated alarm sources are ’NOT OK’. E.g., if one of the ports configured with link alarm indicates link down, the common status LED will be ’red’.
- *Web interface:* Link alarms (link down) are indicated on the *main* Web page, and the *port configuration/status* page.
- *CLI:* A link alarm (link down) is indicated by an exclamation mark (’!’) when displaying the port’s status in the CLI.

- *Digital I/O:* A link alarm can affect the output level of the digital I/O port in the same way as it will affect the common status LED.

  For more information on the functionality of the Digital I/O port, see chapter 27.

9.1.5 Inbound rate limiting and outbound traffic shaping

The switch can be configured to limit rate of the traffic coming in on a port. By default a port will accept packets at a rate up to the link speed, but with inbound rate limiting activated on a port the switch will start to drop packets when data arrives above a given threshold.

The inbound rate limiting feature can be useful as a complement to layer-2 priority handling (see section 9.1.3) when congestion within the network is to be avoided. However, packet drop caused by inbound rate limiting may punish TCP traffic flows harder than desired, due to dynamics of the TCP protocol. Thus, inbound rate limiting should only be used as a means of storm prevention.

Note: *Inbound rate limiting should primarily be used as a means of storm prevention.*

The switch can be configured to limit the outbound data rate on a port (outbound traffic shaping). By default each port will send at the maximum speed of the link, but with outbound traffic shaping activated the switch will limit the outbound rate to a given threshold. Above that threshold the switch will buffer packets - *bursty* traffic will be *shaped*. In case the output buffer is full, additional packets destined for that port will be dropped.

9.1.6 MDI/MDIX crossover

By default a switch is able to sense which pin to use for reception and which to use for transmission (auto MDI/MDIX crossover), thus no external crossover cable is necessary. In addition, a port can be configured statically in MDI (Media Dependent Interface) or MDIX (crossover) mode.

9.1.7 Fall-back default-VID

The fall-back default VLAN ID is generally unnecessary to configure.

The purpose of the fall-back default-VID is to control what should happen with ”untagged” packets entering a port only configured ”tagged” on a set of VLANs. For more information on VLAN features and the VLAN related terms used throughout this section, see chapter 13.

Every port needs to have a ”default VID”. The default VID specifies the VLAN ID an ”untagged” packet should be associated with as it enters that port. A port’s default VID is determined as follows:

- If a port is associated ”untagged” with a VLAN, that VID will be the port’s default VID. E.g., if a port is associated ”untagged” to VID 10, the port will have VID 10 as its ”default VID”.
- If a port is *not* associated ”untagged” with any VLAN, the port’s default VID is determined as:
  - the port’s fall-back default VID, given that a fall-back default-VID is configured, or
  - the default VLAN (VID 1), if no fall-back default-VID is configured.

The fall-back default VID can be used to control whether ”untagged” packets should be accepted on a port (only) associated ”tagged” with a set of VLANs. If the port’s default VID is represented within that set of VLANs, the packet will be accepted. Otherwise it will be dropped.

9.2 Managing port settings via the web interface

9.2.1 List Port Settings

Menu path: Configuration ⇒ Port

When entering the port configuration page you will be presented to a list of all ports available on your switch, see Fig 9.1. Here you get an overview of the settings for all ports, and in addition two items of dynamic information - alarms and link status.

Port Configuration

	Port	Link	Type	Speed/Duplex	Link Alarm Enabled	
	1/1	Down	Fast Ethernet	Auto	–	
	1/2	Down	Fast Ethernet	Auto	–	
	2/1	Down	Gigabit Ethernet Fibre optic	1000 Full	–	
⚠	2/2	Down	Gigabit Ethernet Fibre optic	Auto	✓	
	2/3	Down	Gigabit Ethernet Fibre optic	Auto	–	
	2/4	Down	Gigabit Ethernet Fibre optic	Auto	–	
	3/1	Down	Fast Ethernet	10 Full	–	
⚠	3/2	Down	Fast Ethernet	Auto	✓	
	3/3	Down	Fast Ethernet	Auto	–	
	3/4	Down	Fast Ethernet	100 Half	–	
	3/5	Down	Fast Ethernet	Auto	–	
	3/6	Down	Fast Ethernet	Auto	–	
	3/7	Down	Fast Ethernet	Auto	–	
	3/8	Up	Fast Ethernet	Auto	–	

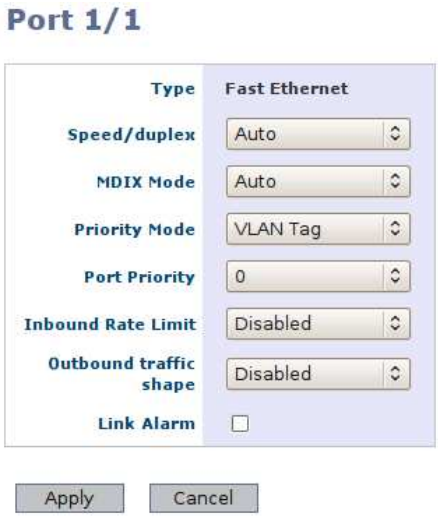
Figure 9.1: Port configuration settings overview (this example is from a RedFox Industrial switch)

Alarm	There is an active link alarm associated with the port. Only shown if link alarm is enabled and the link is down.
Port	The port label.
Link	Link status for the port. Up or down.
Type	The port type: Gigabit Ethernet Fibre optic, Gigabit Ethernet, Fast Ethernet Fibre optic or Fast Ethernet.
Speed/Duplex	The speed duplex setting. Auto means speed and duplex will be automatically negotiated. Otherwise the current setting will be shown as speed in Megabit and duplex as FDX for full duplex and HDX for half duplex. Note! This is not the negotiated speed, it is the configuration setting!
Link Alarm Enabled	When link alarm is enabled an alarm will be generated if port link is down. Alarms trigger an SNMP trap message to be sent and alarms to be shown on the administration web. In the ports overview table a green check-mark means enabled, and a dash means disabled.
Edit	Click this icon to edit a port's settings.

To change the settings for a specific port you will have to click the edit icon which will take you to the port setting edit page see Section 9.2.2.

9.2.2 Edit Port Settings

Menu path: Configuration ⇒ Port ⇒



On this page you can change the settings for the port.

Type	The port type: Gigabit Ethernet Fibre optic, Gigabit Ethernet, Fast Ethernet Fibre optic or Fast Ethernet.
Speed/Duplex	The speed duplex setting. Auto means speed and duplex will be automatically negotiated. Otherwise the current setting will be shown as speed in Megabit and duplex as FDX for full duplex and HDX for half duplex. Note! This is not the negotiated speed, it is the configuration setting!
MDIX mode	How to handle crossover cables. If you connect two units with different port settings (one with mdi and one with mdix) you need a straight-through twisted pair cabling. If you connect two units with the same setting you will need a crossover cabling. **Auto** Automatic detection **mdi** Medium dependent interface **mdix** mdi crossover
Priority Mode	Here you select on what information priority will be based: **Port Based** Based on the port's priority. See the next item (**Priority**). **IP** Based on the content of the IP ToS bits (IPv4) or the IP TC bits (IPv6). **VLAN Tag** Based on the content of the (802.1p) priority field inside the recieved packet's VLAN tag.
Priority	The port's priority level. Zero (0) is low priority and seven (7) high priority.
Inbound Rate Limit	Bandwidth limit for inbound traffic. *Disabled* means no limiting.
Outbound Traffic Shape	Bandwidth limit for outbound traffic. *Disabled* means no limiting.
Link Alarm	When link alarm is enabled an alarm will be generated if port link is down. Alarms trigger an SNMP trap message to be sent and alarms to be shown on the administration web.

9.3 Managing port settings via the CLI

The *port* configuration context can be entered using the **"port <PORT|PORTLIST>"** command from the *Global Configuration* context. When providing a list of ports, the scope of the configuration commands becomes all ports in the list. There is also a specific command, **"ports"**, to enter the port context with the scope of *all ports* of the device.

Command	Default	Section
port <PORTLIST>		Section 9.3.1
ports		Section 9.3.2
<enable|disable>	Enabled	Section 9.3.3
[no] speed-duplex <auto|10-half|10-full|100-half|100-full|. . . >	auto	Section 9.3.4
[no] flow-control	Disabled	Section 9.3.5
[no] priority <0-7>	0	Section 9.3.6
[no] priority-mode <tag|ip|port>	tag	Section 9.3.7
[no] link-alarm	Disabled	Section 9.3.8
[no] rate-limit <70-2560>	Disabled	Section 9.3.9
[no] traffic-shaping <70-2560>	Disabled	Section 9.3.10
mdix <auto|on|off>	auto	Section 9.3.11
[no] default-vid <VLAN_ID>	Disabled	Section 9.3.12
Show port configuration		
show port [PORTLIST]	All ports	Section 9.3.13
show ports		Section 9.3.14
port		
show enable		Section 9.3.15
show speed-duplex		Section 9.3.16
show flow-control		Section 9.3.17
show priority		Section 9.3.18
show priority-mode		Section 9.3.19
show link-alarm		Section 9.3.20
show rate-limit		Section 9.3.21
show traffic-shaping		Section 9.3.22
show mdix		Section 9.3.23
show default-vid		Section 9.3.24
Show port status		
show ports		Section 9.3.25

9.3.1 Managing Ports

Syntax `port <PORT|PORTLIST>`

Context *Global Configuration* context

Usage Enter Port context of the given PORT or PORTLIST.

Default values Not applicable.

Error messages None defined yet.

A **”PORTLIST”** is a comma separated list of port ranges without intermediate spaces, e.g., **”1/1-1/3,2/3”**.

9.3.2 Managing all Ports

Syntax `ports`

Context *Global Configuration* context

Usage Enter Port context with the scope of all ports.

Default values Not applicable.

Error messages None defined yet.

9.3.3 Port enabling and disabling

Syntax `<enable|disable>`

Context *port* context.

Usage Enable or disable a port.

Default values Ports are enabled by default.

Error messages None defined yet.

9.3.4 Speed and duplex setting

Syntax `[no] speed-duplex <auto|10-half|10-full|100-half|100-full|1000-half|1000-full>`

Context *port* context.

Usage Set port speed and duplex modes. **"auto"** means auto-negotiate, other modes are static configurations specifying 10, 100 or 1000 Mbit/s, and half or full duplex.

"no speed-duplex" will revert to default configuration for the speed-duplex setting, i.e., **"speed-duplex auto"**.

Default values `auto`

Error messages An attempt to set a port speed not available for this specific port type will render an error message, including information of available port speeds.

9.3.5 Flow-control setting

Syntax `[no] flow-control`

Context *port* context.

Usage Enable or disable IEEE 802.3 flow-control. For full duplex links, flow control will utilise IEEE 802.3 *pause frames*, and for half duplex links a technique known as *back-pressure* is used.

The flow control setting is only valid when the speed-duplex mode is set to **"auto"**, see section 9.1.1.

Default values Disabled (`no flow-control`)

Error messages None defined yet.

9.3.6 Port priority setting

Syntax `[no] priority <0-7>`

Context *port* context.

Usage Set the (IEEE 802.1p) priority associated with the port. Packets coming in on this port will receive this priority unless priority is based on VLAN ID, VLAN tag or IP ToS/DiffServ bits.

”no priority” will revert to default configuration for the port priority setting, i.e., **”priority 0”** (zero).

Default values `0` (zero)

Error messages None defined yet.

9.3.7 Set port priority mode

Syntax `[no] priority-mode <tag|ip|port>`

Context *port* context.

Usage Base priority classification for this port on content of VLAN tag (IEEE 802.1p priority bits), content of IP ToS/Diffserv bits, or the port priority configured for this port.

Note: VLAN priority settings (see section 13.3) will have precedence over port priority mode settings.

tag (Default) The packet’s priority is based on the content of the VLAN tag (802.1p priority bits) of the incoming packet. For packets coming in *untagged*, the priority is based on the priority associated with the port, see section 9.3.6.

ip The packet’s priority is based on the content of the IP ToS/Diffserv bit of the incoming packet. For non-IP packets coming in on the port (e.g., ARP packets), the priority is based on the priority associated with the port, see section 9.3.6.

port The packet’s priority is based on the priority associated with the port, see section 9.3.6.

Default values `tag`

Error messages None defined yet.

9.3.8 Link alarm

Syntax `[no] link-alarm`

Context *port* context.

Usage Enable or disable link-alarm for this port. When enabled, an alarm indication is activated when the link is down.

Default values Disabled (**"no link-alarm"**)

Error messages None defined yet.

9.3.9 Inbound rate limiting

Syntax `[no] rate-limit <70-256000>`

Context *port* context.

Usage Configure inbound rate limit in kbit/s. Use **"no rate-limit"** to disable inbound rate limiting.

Default values Disabled (**"no rate-limit"**)

Error messages None defined yet.

9.3.10 Outbound traffic shaping

Syntax `[no] traffic-shaping <70-256000>`

Context *port* context.

Usage Configure outbound traffic shaping in kbit/s. Use **"no traffic-shaping"** to disable outbound traffic shaping.

Default values Disabled (**"no traffic-shaping"**)

Error messages None defined yet.

9.3.11 Cable cross-over setting

Syntax `mdix <auto|on|off>`

Context *port* context.

Usage Configuration of Cable Crossover setting. **”auto”** means automatic cross-over mode, **”on”** sets port to cross-over mode (MDIX) and **”off”** sets port to MDI mode. This command is not valid for *fibre* ports.

Default values `auto`.

Error messages None defined yet.

9.3.12 Fall-back default VLAN

Syntax `[no] default-vid <VLAN_ID>`

Context *port* context.

Usage Configuration of (fall-back) default-VID for this port. The default-VID configuration is only valid when this port is not configured ”untagged” on any VLAN.

Use **”`no default-vid`”** to clear the (fall-back) default VID setting (the default-VID setting will also be cleared whenever the port is associated ”untagged” with any VLAN).

When cleared, VLAN ID 1 will be used as the port’s fall-back default-VID.

For more information see section 9.1.7.

Default values Disabled/cleared (`no default-vid`).

Error messages None defined yet.

9.3.13 Show port configuration

Syntax `show port [<PORT|PORTLIST>]`

Context *Global Configuration* context

Usage Show Port configuration information of the given PORT or PORTLIST.

Default values All ports, i.e., if no PORT or PORTLIST is provided, information on all ports will be shown.

Error messages None defined yet.

Alternatively, the command **”show”** can be run within the *port* context, to show the configuration of a port (or list of ports).

9.3.14 Show port configuration (all ports)

Syntax `show ports`

Context *Global Configuration* context

Usage Show Port configuration of all ports.

Default values Not applicable.

Error messages None defined yet.

9.3.15 Show port enable/disable setting

Syntax `show enable`

Context *port* context.

Usage Show whether the port is configured *enabled* or *disabled*.

Default values Not applicable.

Error messages None defined yet.

9.3.16 Show speed and duplex setting

Syntax `show speed-duplex`

Context *port* context.

Usage Show port speed and duplex mode settings.

Default values Not applicable.

Error messages None defined yet.

9.3.17 Show flow-control setting

Syntax `show flow-control`

Context *port* context.

Usage Show port IEEE 802.3 flow control setting.

Default values Not applicable.

Error messages None defined yet.

9.3.18 Show port priority setting

Syntax `show priority`

Context *port* context.

Usage Show port priority setting.

Default values Not applicable.

Error messages None defined yet.

9.3.19 Show priority mode setting

Syntax `show`

Context *port* context.

Usage Show whether this port is configured to classify the priority of incoming packet based on their VLAN tag (priority bits), IP ToS/DiffServ bits or the port's priority.

Default values Not applicable.

Error messages None defined yet.

9.3.20 Show link alarm setting

Syntax `show link-alarm`

Context *port* context.

Usage Show link-alarm setting.

Default values Not applicable.

Error messages None defined yet.

9.3.21 Show inbound rate limit setting

Syntax `show rate-limit`

Context *port* context.

Usage Show inbound rate limit setting.

Default values Not applicable.

Error messages None defined yet.

9.3.22 Show outbound traffic shaping setting

Syntax `show traffic-shaping`

Context *port* context.

Usage Show outbound traffic shaping setting.

Default values Not applicable.

Error messages None defined yet.

9.3.23 Show cable cross-over setting

Syntax `show mdix`

Context *port* context.

Usage Show port cable cross-over setting. Not applicable to fibre ports.

Default values Not applicable.

Error messages None defined yet.

9.3.24 Show fall-back default-vid setting

Syntax `show default-vid`

Context *port* context.

Usage Show (fall-back) default-vid setting.

Default values Not applicable.

Error messages None defined yet.

9.3.25 Show port status (all ports)

Syntax `show ports`

Context *Admin Exec* context

Usage Show Port status information for all ports.

Default values Not applicable.

Error messages None defined yet.

Chapter 10

SHDSL Port Management

DDW-225 (Wolverine family) is equipped with two SHDSL ports (Symmetric High-speed Digital Subscriber Line), enabling LAN networks to be extended over legacy copper cabling.

10.1 Overview of SHDSL Port Management

Feature	Web (Sec. 10.2)	CLI (Sec. 10.3)	General Description
CO/CPE mode selection	X	X	Sec. 10.1.1-10.1.2
DSL link rate	X	X	Sec. 10.1.1-10.1.2
DSL noise margin	X	X	Sec. 10.1.1-10.1.2
Settings in common with Ethernet ports			
Enable/disable port		X	Sec. 10.1.3
Port priority (level)		X	Sec. 10.1.3
Port priority mode		X	Sec. 10.1.3
Link alarm	X	X	Sec. 10.1.3
Inbound rate limit		X	Sec. 10.1.3
Outbound traffic shaping		X	Sec. 10.1.3
Fall-back default-VID		X	Sec. 10.1.3
View DSL port configuration	X	X	
View DSL port status	X	X	

10.1.1 SHDSL overview

With SHDSL Ethernet LANs can be extended over legacy copper cabling. Switches can be connected in a simple point-to-point setup, but also in multi-drop and ring topologies, as shown in fig. 10.1.

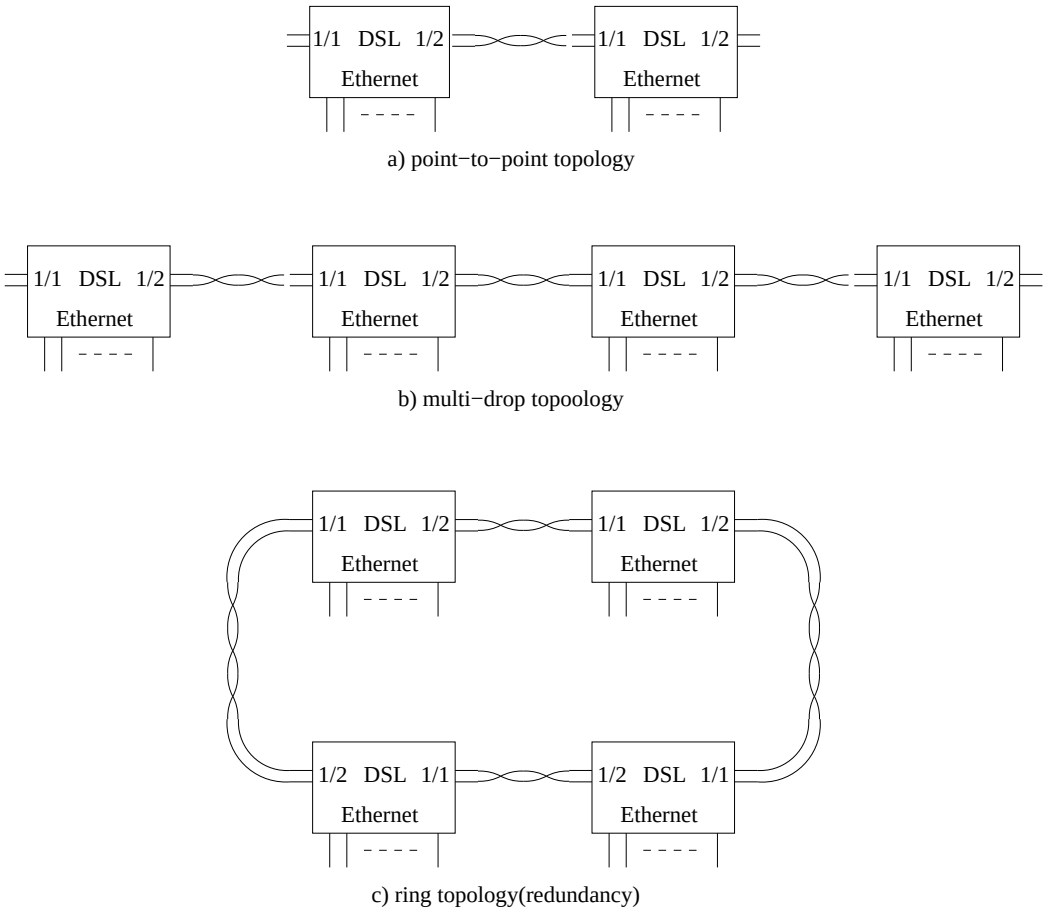


Figure 10.1: SHDSL topologies: Point-to-point (a), multi-drop (b) and ring (c).

In a SHDSL connection, the port on one unit shall be configured as *Central Office* (CO) and the port on the other unit as Customer Premises Equipment (CPE). In WeOS the SHDSL ports are named *1/1* and *1/2* (according to the slotID/portID name convention described in section 1.4.2). By default 1/1 is configured as *CPE* while 1/2 is configured as *CO*.

SHDSL support in WeOS is based on *Ethernet First Mile* (EFM) technology, and SHDSL can to a large extent be treated in the same way as Ethernet ports, e.g., you can add SHDSL ports to VLANs (chapter 13), you can run link-layer redundancy protocols such as FRNT (chapter 14) and RSTP (chapter 15) over them, etc. Settings specific to SHDSL ports are described in section 10.1.2 while port settings of more general nature is covered in section 10.1.3.

10.1.2 Settings specific to SHDSL ports

- *Port role*: One unit shall be configured as *Central Office* (CO) and the other unit as Customer Premises Equipment (CPE). CO is the answering central unit. CPE (Customer Premises Equipment) is the unit that initiates the connection. In WeOS the SHDSL ports are named *1/1* and *1/2*: by default 1/1 is configured as *CPE* and 1/2 configured as *CO*.

- *Data rate:* The SHDSL connection data rate can be achieved in the range from 192 kbit/s up to 5696 kbit/s depending on cable characteristics and communication distance. The operator can either specify a fixed data rate to be used, or let the CO and CPE discover the achievable data rate automatically.

  Using *Auto* mode will optimise the data rate for the current SNR conditions, however, the time to establish the SHDSL connection is larger when *Auto* mode is used as compared to configuring a fixed data rate.

- *Noise margin*: The noise margin is the difference between the required SNR for a certain bit rate, and the actual SNR.

  When the SHDSL connection data rate is set to auto-negotiation mode, the operator can configure an *administrative noise margin* (also referred to as *target noise margin* or *target SNR margin*). A large *administrative noise margin* gives robustness against SNR fluctuations. But as the *required SNR* increases with data rate, specifying a a large *administrative noise margin* may imply that a low data rate is negotiated.

  Thus, when configuring the *administrative noise margin* the operator can optimise the connection for *reliability* (noise margin 10dB), *high speed* (noise margin 3dB) or as a tradeoff thereof (*normal* mode, i.e., noise margin 6dB).

  To monitor the quality of the connection, WeOS enables the operator to read the *operational noise margin*, i.e., the difference between the current SNR and the required SNR.

Note: *Only the data rate and noise margin settings of the CO are used in the SHDSL connection. These parameters are passed to the CPE during the connection establishment phase.*

10.1.3 General port settings

The following parameters can be configured for SHDSL ports in the same way as for Ethernet ports. The SHDSL uses Ethernet First Mile (EFM) encapsulation, thus many Ethernet settings apply to the SHDSL ports. More detailed information is found in chapter 9.

- *Port enable/disable:* Ports can be disabled and enabled administratively.
- *Port priority mode:* Define whether incoming packets should be prioritised based on VLAN tag, VLAN ID, port ID, IP ToS, etc. See also section 9.1.3.
- *Port priority (level):* The inbound priority associated with this port. See also section 9.1.3.
- *Link alarm:* Link status can be configured as an alarm source. See also section 9.1.4.
- *Inbound rate limit:* Setting the inbound rate limit is possible on DSL ports, but is likely of less interest than on Ethernet ports, since the DSL data rates are primarily limited by the rate of the DSL line. See also sections 9.1.5 and 10.1.2.
- *Outbound traffic shaping:* Setting the outbound rate limit (traffic shaping) is possible on DSL ports, but is likely of less interest than on Ethernet ports, since the DSL data rates are primarily limited by the rate of the DSL line. See also sections 9.1.5 and 10.1.2.
- *Fall-back default-VID*: The fall-back default VID setting is only of interest for the special case when *untagged* packets are received over a link only associated with *tagged* VLANs.

Ethernet settings for *port speed/duplex* mode, and *MDI/MDIX* mode do not apply to SHDSL ports, thus are not configurable.

Note: *As of WeOS v4.3.0, enabling/disabling flow control (as described in section section 9.1.2) has no effect on SHDSL ports.*

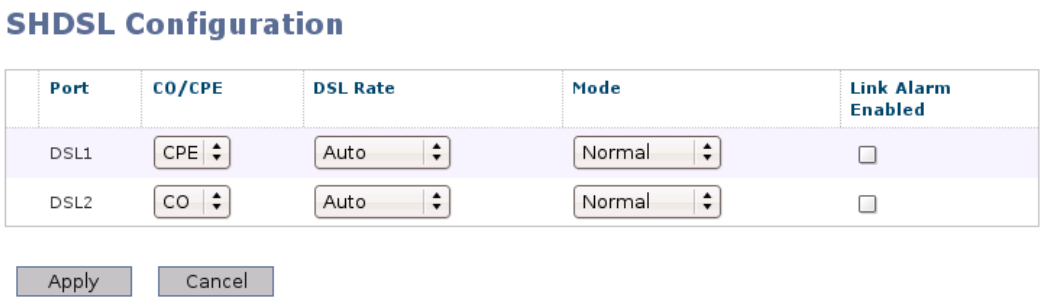
10.2 Managing SHDSL ports via the web interface

The Web interface provides configuration of SHDSL ports as well as listing of SHDSL port statistics.

The SHDSL statistics is provided in two views – an *overview* with a selection of statistics for all SHDSL ports, including some status information, and a *detailed* page with a larger set of statistics.

10.2.1 List and Edit SHDSL Port Settings

Menu path: Configuration ⇒ SHDSL



On this page you can list and change the settings for the SHDSL ports.

Port	The SHDSL port label.
CO/CPE	To establish a connection between two DSL-ports, one has to be configured as Central Office (CO) and one has to be configured as Customer Premises Equipment (CPE). Default for port 1/1 is *CPE*, and default for port 1/2 is *CO*.
DSL Rate	Speed setting is only valid if the port is configured as CO (the CPE rate setting is not used, since the CPE speed automatically follows the CO to which it becomes connected). Default is **`Auto`**.

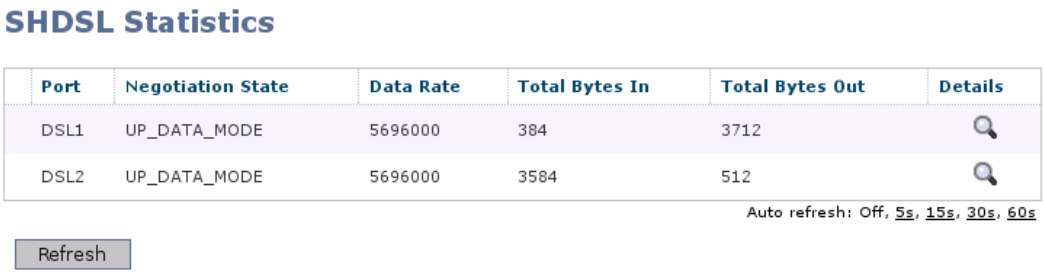
Continued on next page

	Continued from previous page
Mode	The *noise-margin mode*. The *noise-margin mode* setting is only valid if the port is configured as CO (the CPE setting is not used, since the CPE *noise-margin mode* automatically follows the CO to which it becomes connected). The CO can be configured to choose a faster less reliable speed (**`High Speed`**), a slower more reliable speed (**`Reliable`**), or a tradeoff between these two objectives (**`Normal`**). Default is **`Normal`**.
Link Alarm	When link alarm is enabled an alarm will be generated if port link is down. Alarms trigger an SNMP trap message to be sent and alarms to be shown on the administration web.

10.2.2 SHDSL statistics Overview

Menu path: Statistics ⇒ SHDSL

On the SHDSL port statistics overview page you will be presented to a selection of static data for each port. Additional statistic numbers are presented on the detailed view page.

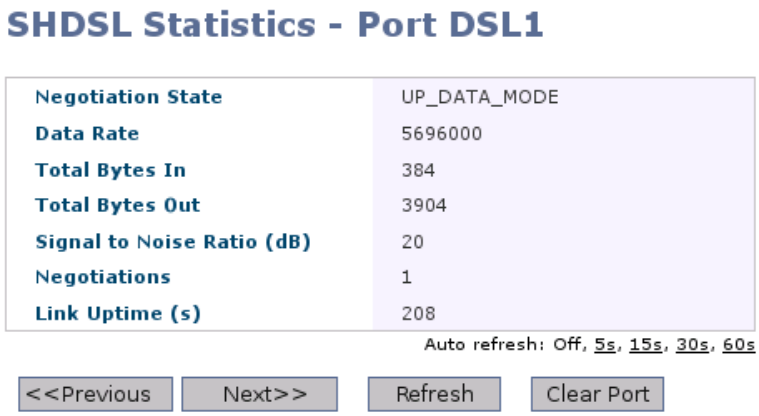


Alarm	An alarm icon appears at the start of a line if there is a link alarm on a port.
Port	The port label.
Negotiation State	Current state of the DSL-line negotiation. Possible values are UP_DATA_MODE, INITIALISING, DOWN_READY and DOWN_NOT_READY. Note: if no link is established the normal state for a CO-mode configured port is DOWN_NOT_READY, for a CPE-configured port the normal state is DOWN_READY.
Data Rate	Negotiated DSL data rate in bit/s.
Total Bytes In	Total number of bytes received on the port.
Total Bytes Out	Total number of bytes sent out on the port.
Details	Click this icon to view more detailed statistics for the port.
Auto Refresh	Click on a value to make the page reload with updated statistics automatically every 5, 15, 30 or 60 seconds. Click Off to turn off auto refresh.
Refresh	Click on this button to reload with updated statistics.

10.2.3 Detailed SHDSL Port Statistics

Menu path: Statistics ⇒ SHDSL ⇒ 

When clicking the *details*-icon in the overview page you will be presented to the detailed statistics page for the SHDSL port.



Negotiation State	Current state of the DSL-line negotiation. Possible values are UP_DATA_MODE, INITIALISING, DOWN_READY and DOWN_NOT_READY. Note: if no link is established the normal state for a CO-mode configured port is DOWN_NOT_READY, for a CPE-configured port the normal state is DOWN_READY.
Data Rate	Negotiated DSL data rate in bit/s.
Total Bytes In	Total number of bytes received on the port.
Total Bytes Out	Total number of bytes sent out on the port.
Details	Click this icon to view more detailed statistics for the port.
Signal to Noise Ratio (SNR)	Signal to Noise Ratio in dB on this link.
Negotiations	Number of negotiations since unit startup.
Link Uptime	Number of seconds since link was established.
Auto Refresh	Click on a value to make the page reload with updated statistics automatically every 5, 15, 30 or 60 seconds. Click Off to turn off auto refresh.
<<Previous	Goto statistics for previous port.
Next>>	Goto statistics for next port.
Refresh	Click on this button to reload with updated statistics.
Clear Port	Clear all statistics counters for the port shown.

10.3 Managing SHDSL ports via the CLI

The table below shows SHDSL port management features available via the CLI.

Command	Default	Section
Configure SHDSL port settings		
port [shdsl] <PORTLIST>		Section 10.3.1
[no] enable	Enabled	Section 10.3.2
[no] co		Section 10.3.3
[no] rate <0-5696k>	0 (Auto)	Section 10.3.4
[no] noise-margin	Normal	Section 10.3.5
[no] link-alarm	Disabled	Section 10.3.6
[no] priority <0-7>	0	Section 10.3.7
[no] priority-mode <tag|ip|port>	tag	Section 10.3.8
[no] rate-limit <70-2560>	Disabled	Section 10.3.9
[no] traffic-shaping <70-2560>	Disabled	Section 10.3.10
[no] default-vid <VLAN_ID>	Disabled	Section 10.3.12
Show SHDSL port settings		
show port [shdsl] [PORTLIST]	All ports	Section 10.3.13
show ports [shdsl]	All ports	Section 10.3.14
port [shdsl] [PORTLIST]		
show enable		Section 10.3.15
show co		Section 10.3.16
show rate		Section 10.3.17
show noise-margin		Section 10.3.18
show link-alarm		Section 10.3.19
show priority		Section 10.3.20
show priority-mode		Section 10.3.21
show rate-limit		Section 10.3.23
show traffic-shaping		Section 10.3.24
show default-vid		Section 10.3.25
Show SHDSL port status		
show shdsl		Section 10.3.26
show rmon		Section 26.3

10.3.1 Managing SHDSL port settings

Syntax `port [shdsl] <PORTLIST>`

Context *Global Configuration* context

Usage Enter the SHDSL port configuration context *SHDSL port*.

A **"PORTLIST"** is a comma separated list of ranges of SHDSL ports without intermediate spaces, e.g., **"1/1,1/2"**.

As of WeOS v4.3.0 the qualifier keyword *shdsl* is not used.

Default values Not applicable.

Error messages None defined yet.

10.3.2 Enable/disable SHDSL port settings

Syntax `[no] enable`

Context *SHDSL port* context

Usage Enable or disable a port.

Default values SHDSL ports are enabled by default.

Error messages None defined yet.

10.3.3 Setting SHDSL port mode (CO/CPE)

Syntax `[no] co`

Context *SHDSL port* context

Usage Set the SHDSL port to operate in *central office* (CO) or *customer premises equipment* (CPE) mode.

When connecting switches via SHDSL it is important that one side puts its SHDSL port in CO mode (**"co"**) while the other side puts its SHDSL port in CPE mode (**"no co"**).

Default values Factory default for DDW-225 is to have port 1/1 in *CPE* mode (**"no co"**), and port 1/2 in *CO* mode (**"co"**).

Error messages None defined yet.

10.3.4 Setting SHDSL port rate

Syntax `[no] rate <0-5696k>`

Context *SHDSL port* context

Usage Set SHDSL port rate, either bye specifying that auto-negotiation should be used, or that a specific fixed rate should be used.

- *Auto-negotiate:* Use **"rate 0"** to specify that the data rate is to automatically negotiated between the SHDSL peers. Alternatively, **"rate 0k"** or **"no rate"** can be used to give the same result.
  Fixed rate: Use **"rate 1k-5696k"** to specify a fixed data rate in kbit/s, or **"rate 1-5696000"** to specify a fixed data rate in bit/s.
  The following fixed rates are supported: 192k, 384k, 512k, 768k, 1024k, 1280k, 2048k, 2304k, 2688k, 3072k, 3456k, 3840k, 4224k, 4608k, 4992k, 5376k, and 5696k. If other rates are specified, WeOS will round the value upwards to the nearest supported rate.

Default values **"rate 0"** (Auto)

Error messages None defined yet.

10.3.5 Setting SHDSL port noise-margin

Syntax `[no] noise-margin <reliable|normal|high-speed>`

Context *SHDSL port* context

Usage Set SHDSL port *noise-margin*. *Note:* The noise-margin setting is only relevant when the data rate is set to *auto-negotiate* (**"rate 0"**), see section 10.3.4).

Available noise-margin modes:

- *Reliable:* Select **"noise-margin reliable"** to let the rate auto-negotiation optimise for reliability (rather than high data rate).
- *High-Speed:* Select **"noise-margin high-speed"** to let the rate auto-negotiation optimise for high data rate (rather than reliability).
- *Normal:* **"noise-margin normal"** is the default setting for the noise-margin, which gives a tradeoff between reliability and high-speed. Alternatively, the command **"no noise-margin"** can be used.

Default values **”noise-margin normal”**

Error messages None defined yet.

10.3.6 Configure DSL port link alarm

Syntax `[no] link-alarm`

Context *SHDSL port* context

Usage Enable or disable link-alarm for this SHDSL port. When enabled, an alarm indication is activated when the link is down.

Default values Disabled (**”no link-alarm”**)

Error messages None defined yet.

10.3.7 Port priority setting

Syntax `[no] priority <0-7>`

Context *SHDSL port* context.

Usage Set the (IEEE 802.1p) priority associated with the port. Packets coming in on this port will receive this priority unless priority is based on VLAN ID, VLAN tag or IP ToS/DiffServ bits.

”no priority” will revert to default configuration for the port priority setting, i.e., **”priority 0”** (zero).

Default values `0` (zero)

Error messages None defined yet.

10.3.8 Set port priority mode

Syntax `[no] priority-mode <tag|ip|port>`

Context *SHDSL port* context.

Usage Base priority classification for this port on content of VLAN tag (IEEE 802.1p priority bits), content of IP ToS/Diffserv bits, or the port priority configured for this port.

Note: VLAN priority settings (see section 13.3) will have precedence over port priority mode settings.

tag (Default) The packet's priority is based on the content of the VLAN tag (802.1p priority bits) of the incoming packet. For packets coming in *untagged*, the priority is based on the priority associated with the port, see section 10.3.7.

ip The packet's priority is based on the content of the IP ToS/Diffserv bit of the incoming packet. For non-IP packets coming in on the port (e.g., ARP packets), the priority is based on the priority associated with the port, see section 10.3.7.

port The packet's priority is based on the priority associated with the port, see section 10.3.7.

Default values `tag`

Error messages None defined yet.

10.3.9 Inbound rate limiting

Syntax `[no] rate-limit <70-256000>`

Context *SHDSL port* context.

Usage Configure inbound rate limit in kbit/s. Use **`"no rate-limit"`** to disable inbound rate limiting.

Default values Disabled (**`"no rate-limit"`**)

Error messages None defined yet.

10.3.10 Outbound traffic shaping

Syntax `[no] traffic-shaping <70-256000>`

Context *SHDSL port* context.

Usage Configure outbound traffic shaping in kbit/s. Use **`"no traffic-shaping"`** to disable outbound traffic shaping.

Default values Disabled (**`"no traffic-shaping"`**)

Error messages None defined yet.

10.3.11 Cable cross-over setting

Syntax `mdix <auto|on|off>`

Context *SHDSL port* context.

Usage Configuration of Cable Crossover setting. **”auto”** means automatic cross-over mode, **”on”** sets port to cross-over mode (MDIX) and **”off”** sets port to MDI mode. This command is not valid for *fibre* ports.

Default values `auto`.

Error messages None defined yet.

10.3.12 Fall-back default VLAN

Syntax `[no] default-vid <VLAN_ID>`

Context *SHDSL port* context.

Usage Configuration of (fall-back) default-VID for this port. The default-VID configuration is only valid when this port is not configured ”untagged” on any VLAN.

Use **”no default-vid”** to clear the (fall-back) default VID setting (the default-VID setting will also be cleared whenever the port is associated ”untagged” with any VLAN).

When cleared, VLAN ID 1 will be used as the port’s fall-back default-VID.

For more information see section 9.1.7.

Default values Disabled/cleared (`no default-vid`).

Error messages None defined yet.

10.3.13 Show port configuration

Syntax `show port [shdsl] [<PORT|PORTLIST>]`

Context *Global Configuration* context

Usage Show Port configuration information of the given PORT or PORTLIST.

Default values All ports, i.e., if no PORT or PORTLIST is provided, information on all ports will be shown.

As of WeOS v4.3.0 the qualifier keyword *shdsl* is not used.

Error messages None defined yet.

Alternatively, the command **”show”** can be run within the *SHDSL port* context, to show the configuration of a port (or list of ports).

10.3.14 Show port configuration (all ports)

Syntax `show ports [shdsl]`

Context *Global Configuration* context

Usage Show Port configuration of all ports.
As of WeOS v4.3.0 the qualifier keyword *shdsl* is not used.

Default values Not applicable.

Error messages None defined yet.

10.3.15 Show SHDSL port enable/disable setting

Syntax `show enable`

Context *SHDSL port* context.

Usage Show whether the SHDSL port is configured *enabled* or *disabled*.

Default values Not applicable.

Error messages None defined yet.

10.3.16 Show SHDSL port SHDSL port mode (CO/CPE) setting

Syntax `show co`

Context *SHDSL port* context.

Usage Show whether the SHDSL port is configured to operate as *Central Office* or *Customer Premises Equipment*.

Default values Not applicable.

Error messages None defined yet.

10.3.17 Show SHDSL port rate setting

Syntax `show rate`

Context *SHDSL port* context.

Usage Show whether the SHDSL port is configured for data rate auto-negotiation, or to use a specific fixed data rate.

Default values Not applicable.

Error messages None defined yet.

10.3.18 Show SHDSL port noise margin setting

Syntax `show noise-margin`

Context *SHDSL port* context.

Usage Show the configured *noise-margin* setting.

Default values Not applicable.

Error messages None defined yet.

10.3.19 Show SHDSL port link alarm setting

Syntax `show link-alarm`

Context *SHDSL port* context.

Usage Show whether *link alarm* is enabled or disabled for this SHDSL port.

Default values Not applicable.

Error messages None defined yet.

10.3.20 Show port priority setting

Syntax `show priority`

Context *SHDSL port* context.

Usage Show port priority setting.

Default values Not applicable.

Error messages None defined yet.

10.3.21 Show priority mode setting

Syntax `show`

Context *SHDSL port* context.

Usage Show whether this port is configured to classify the priority of incoming packet based on their VLAN tag (priority bits), IP ToS/DiffServ bits or the port's priority.

Default values Not applicable.

Error messages None defined yet.

10.3.22 Show link alarm setting

Syntax `show link-alarm`

Context *SHDSL port* context.

Usage Show link-alarm setting.

Default values Not applicable.

Error messages None defined yet.

10.3.23 Show inbound rate limit setting

Syntax `show rate-limit`

Context *SHDSL port* context.

Usage Show inbound rate limit setting.

Default values Not applicable.

Error messages None defined yet.

10.3.24 Show outbound traffic shaping setting

Syntax `show traffic-shaping`

Context *SHDSL port* context.

Usage Show outbound traffic shaping setting.

Default values Not applicable.

Error messages None defined yet.

10.3.25 Show fall-back default-vid setting

Syntax `show default-vid`

Context *SHDSL port* context.

Usage Show (fall-back) default-vid setting.

Default values Not applicable.

Error messages None defined yet.

10.3.26 Show SHDSL port status

Syntax `show shdsl`

Context *Admin Exec* context.

Usage Show the status of all SHDSL ports.

Default values Not applicable.

Error messages None defined yet.

Chapter 11

Serial Port Management

This chapter describes serial port features and management support in WeOS, and applies to Westermo products equipped with one or more serial ports.

As of WeOS v4.3.0 the serial port can be used in *serial extender* applications (Serial Over IP, see chapter 12). Future versions of WeOS will include additional serial port applications, such as *modem replacement* capabilities.

11.1 Overview of Serial Port Management

The table below presents the serial port management features in WeOS.

Feature	Web (Sec. 11.2)	CLI (Sec. 11.3)	General Description
Speed	X	X	Sec. 11.1.1
Data bits	X	X	-”-
Parity	X	X	-”-
Stop bits	X	X	-”-
Hardware flow control	X	X	Sec. 11.1.2
Software flow control	X	X	Sec. 11.1.3

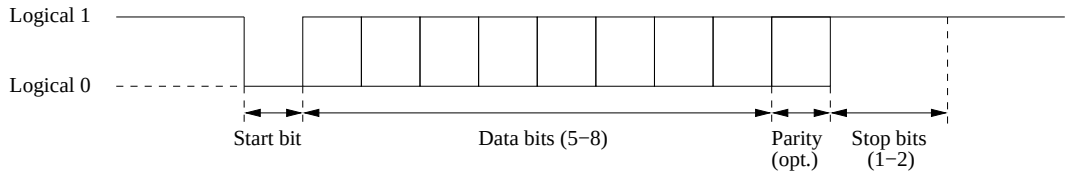
11.1.1 Serial Port Settings

The serial port settings include the following parameters:

- *Speed:* Set serial port data rate (bits/s). Possible data rates are: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800,

500000, 576000, 921600, 1000000, 1152000, 1500000, and 2000000 bits/s. Default: **115200 bits/s**

- *Data character:*



  - *Data bits:* Number of data bits per character. Possible values are 5-8 bits. Default: **8 data bits**
  - *Parity:* Parity error detection setting. Possible settings are *none* (no parity checking), *even* and *odd* parity checking. When configured to use even (or odd) parity, an additional bit (the parity bit) is transmitted after the data bits to enforce that an even (or odd, respectively) number of 1's are sent, thereby enabling the receiver to detect single[1] bit errors. Default: **No parity**
  - *Stop bits:* Number of stop bits. Possible values are 1 and 2 bits. The stop bits define the interval until the next character can be transmitted, and are sent as logical 1 (compare with the *start bit*, which is sent as a logical 0). Default: **1 stop bit**
- *Flow control*
  - *Hardware flow control:* Hardware flow control using RTS/CTS (explained further in section 11.1.2). Default: **Disabled**
  - *Software flow control:* Software flow control using XON/XOFF (explained further in section 11.1.3). Default: **Disabled**

11.1.2 Hardware flow control using RTS/CTS

RS-232 serial ports can use the request to send (RTS) and clear to send (CTS) pins to enforce flow control over the serial line. The DTE will assert the RTS to indicate to the DCE that it has data to send, and the DCE will respond by asserting the CTS when it is ready to receive data.

Similarly, the DCE asserts the CTS when it has data to send, and the DTE will respond by asserting RTS to give the DCE permission to send. The extension to allow the flow-control to work both ways is referred to as *RTS/CTS handshaking* and was not included in the original RS-232 standard.

[1] More precisely, the parity bit enables the receiver to detect an odd number of bit errors.

Serial ports on WeOS devices are typically RS-232 ports using RJ-45 sockets (EIA/TIA-561) in DCE mode, as shown in fig. 11.1 (for a definite description of the serial port on your Westermo device, see the associated product user guide).

Signal	Acronym	Dir (DCE)	Nb
Request To Send	RTS	In	8
Clear To Send	CTS	Out	7
Transmitted Data	TD	In	6
Received Data	RD	Out	5
Signal Ground	SG		4
Data Terminal Ready	DTR	In	3
Data Carrier Detect	DCD	Out	2
Data Set Ready	DSR	Out	1

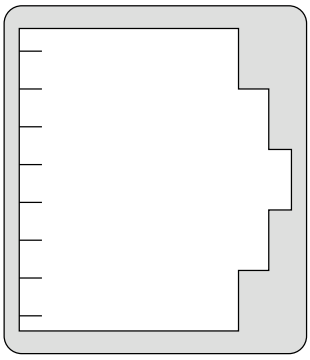


Figure 11.1: Typical RS-232 serial port on WeOS devices – RJ-45 socket (EIA/TIA-561) in DCE mode.

11.1.3 Software flow control using XON/XOFF

An alternative to hardware flow control is to use software flow control, which does not require the presence of the RTS and CTS pins. With software flow control (XON/XOFF) the receiver can stop the sender by transmitting a special character (XOFF, ASCII 19) over the data line. Once the receiver is ready to receive more data it transmits an XON character (ASCII 17).

11.2 Managing serial ports via the web interface

The Web interface provides configuration of serial ports.

11.2.1 Serial ports overview

Menu path: Configuration ⇒ Serial ⇒ Port

Figure 11.2: Serial port configuration settings overview

11.2.2 Edit Serial Port Settings

Menu path: Configuration ⇒ Serial ⇒ Port⇒

On this page you can change the settings for the serial port.

Speed	Set serial port data rate
Data bits	Set the number of data bits
Parity	Set parity error detection
Stop bits	Set the number of stop bits
HW flow control	Enable/disable hardware flow control using RTS/CTS
SW flow control	Enable/disable software flow control using XON/XOFF

11.3 Managing serial ports via the CLI interface

The table below shows serial port management features available via the CLI.

Command	Default	Section
Configure Serial port settings		
port [serial] <PORT>		Section 11.3.1
[no] speed <300-2000000>	115200	Section 11.3.2
[no] databits <5-8>	8	Section 11.3.3
[no] parity	Disabled	Section 11.3.4
[no] stopbits	1	Section 11.3.5
[no] xonxoff	Disabled	Section 11.3.7
[no] rtscts	Disabled	Section 11.3.6
Show serial port settings		
serial		
show		Section 11.3.8
show speed		Section 11.3.9
show databits		Section 11.3.10
show parity		Section 11.3.11
show stopbits		Section 11.3.12
show xonxoff		Section 11.3.13
show rtscts		Section 11.3.14
Show serial port status		
show serial		Section 11.3.15

11.3.1 Managing serial port settings

Syntax `port serial <PORT>`

Context *Global Configuration* context

Usage Enter the Serial port configuration context.

Default value Not applicable.

Error messages None defined yet.

11.3.2 Setting port speed

Syntax `[no] speed <300-2000000>`

Context *Serial port* context

Usage Set serial port data rate. Possible data rates: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 500000, 576000, 921600, 1000000, 1152000, 1500000, and 2000000 bits/s.

Use **"no speed"** to reset the serial port data rate to the default setting.

Default value 115200 (bits/s)

Error messages None defined yet.

11.3.3 Setting number of data bits

Syntax `[no] databits <5-8>`

Context *Serial port* context

Usage Set the number number of data bits.

Use **"no databits"** to reset the number of data bits to the default setting.

Default value 8

Error messages None defined yet.

11.3.4 Setting parity error detection

Syntax `[no] parity <odd|even>`

Context *Serial port* context

Usage Set parity error detection. Use command **"parity odd"** to specify *odd* parity, or **"parity even"** to specify *even* parity on this serial port.

Use **"no parity"** to disable parity checking on this port.

Default value Disabled (`no parity`).

Error messages None defined yet.

11.3.5 Setting number of stop bits

Syntax `[no] stopbits <1|2>`

Context *Serial port* context

Usage Set the number number of stop bits (1 or 2).

Use **"no stopbits"** reset the number of stop bits to the default setting.

Default value 1

Error messages None defined yet.

11.3.6 Setting Hardware flow control (RTS/CTS)

Syntax `[no] rtscts`
Context *Serial port* context
Usage Enable/disable hardware flow control using RTS/CTS
Default value Disabled (`no rtscts`)
Error messages None defined yet.

11.3.7 Setting Software flow control (XON/XOFF)

Syntax `[no] xonxoff`
Context *Serial port* context
Usage Enable/disable software flow control using XON/XOFF
Default value Disabled (`no xonxoff`)
Error messages None defined yet.

11.3.8 Show All Settings of a Serial Port

Syntax `show`
Context *Serial port* context
Usage Show all configuration settings for this serial port.
Default value Not applicable.

11.3.9 Show Serial Port Speed Setting

Syntax `show speed`
Context *Serial port* context
Usage Show the serial port speed setting.
Default value Not applicable.

11.3.10 Show Serial Port Databits setting

Syntax `show databits`
Context *Serial port* context
Usage Show the configured number of databits for this serial port.
Default value Not applicable.

11.3.11 Show Serial Port Parity Setting

Syntax `show parity`

Context *Serial port* context

Usage Show the parity checking setting for this port: *None* (i.e., Disabled), *Odd*, or *Even*.

Default value Not applicable.

11.3.12 Show Serial Port Stopbits Setting

Syntax `show stopbits`

Context *Serial port* context

Usage Show the configured number of stopbits for this serial port.

Default value Not applicable.

11.3.13 Show Software Flow Control Setting (XON/XOFF)

Syntax `show xonxoff`

Context *Serial port* context

Usage Show the software flow control setting (XON/XOFF) for this serial port.

Default value Not applicable.

11.3.14 Show Hardware Flow Control Setting (RTS/CTS)

Syntax `show rtscts`

Context *Serial port* context

Usage Show the hardware flow control setting (RTS/CTS) for this serial port.

Default value Not applicable.

11.3.15 Show Serial Port Status

Syntax `show serial`

Context *Admin Exec* context

Usage Show status of all serial ports.

Default value Not applicable.

Chapter 12

Serial Over IP

This chapter describes the *serial over IP* application available on WeOS products equipped with a serial port. *Serial over IP* enables you to:

- extend an existing serial communication channel over an intermediate IP network.
- create a virtual serial port for remote access from a PC.

For information on serial port configuration (data rate, data bits, etc.), see chapter 11.

12.1 Overview of Serial Over IP

Feature	Web (Sec. 12.2)	CLI (Sec. 12.3)	General Description
Mode (server, client, peer)	X	X	Secs. 12.1.1-12.1.3
Protocol Extensions	X	X	Secs. 12.1.1.3, 12.1.3
Packing of Data	X	X	Sec. 12.1.2
Frame separator	X	X	”
Frame size	X	X	”
Frame delay	X	X	”
Select Serial Port	X	X	Sec. 12.1.3
Addressing/Port Settings	X	X	”
Receiving (incl. multicast)	X	X	”
Sending (incl. multicast)	X	X	”

12.1.1 Serial Over IP introduction

The *Serial Over IP* application can be used in several ways, but the use cases can be divided into three typical applications: serial point-to-point, serial one-to-many (typically a Master-slaves application), and PC access to remote serial devices.

12.1.1.1 Point-to-point

In this way two serial devices can communicate over an IP network. It can be set up either as a client-server configuration using TCP, and as two peers using UDP.

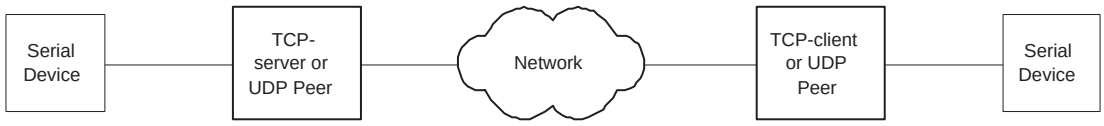


Figure 12.1: Serial Point-to-point link

12.1.1.2 One-to-many

This allows one serial device (typically a master) to communicate with multiple serial devices using UDP transport. It can be set up as IP broadcast, IP multicast, or via multiple IP unicast streams.

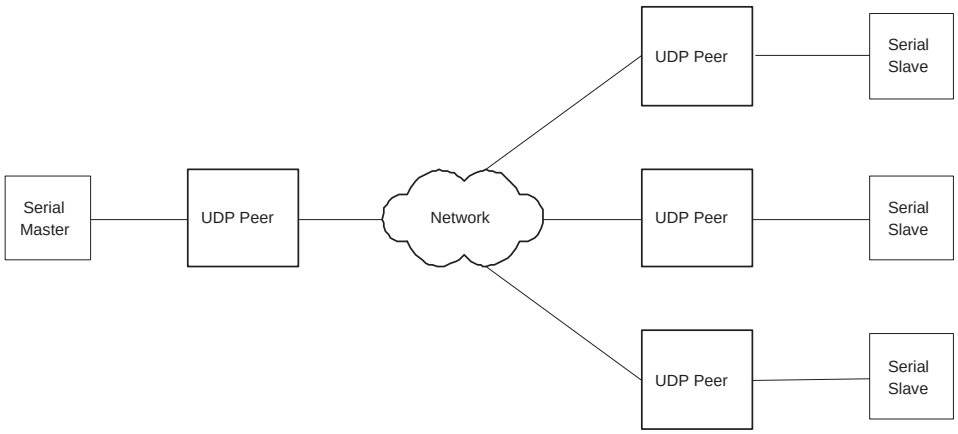


Figure 12.2: Serial one to many

12.1.1.3 Serial Port Redirector (Virtual Serial Port)

By using a serial port redirector software, an application can access remote serial devices as if the where directly connected to the PC. Westermo provides a OEM version Serial/IP® that allows up to 10 virtual serial port to be created. Note: the OEM version of Serial/IP® requires that telnet protocol extension is enabled to verify the license. There is also a possibility for an application to directly connect to the Serial Over IP.

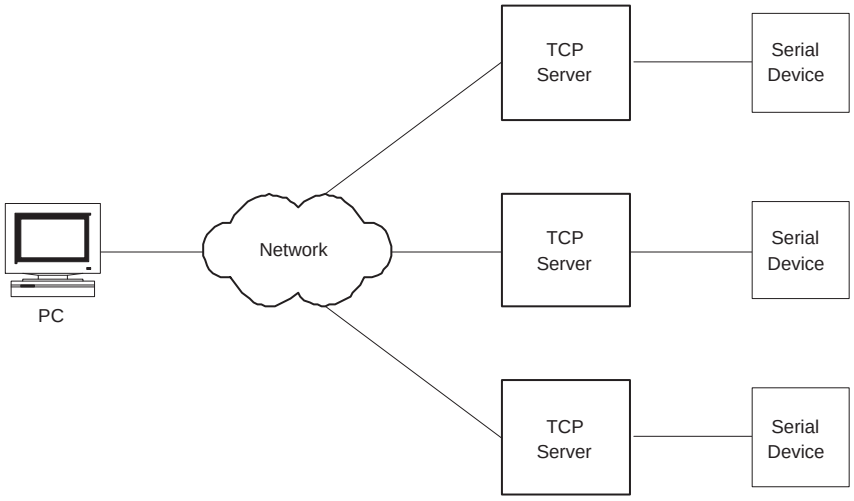


Figure 12.3: Using a serial port redirector software

12.1.2 Packing Algorithm

When data arrives at the serial port of the WeOS unit, one of the following criteria must be fulfilled before the serial data is encapsulated into a UDP/TCP packet and sent over the network.

- *Frame separator character detected:* A frame separator character can be defined. The serial data buffered will be sent over the network when this character is detected,e.g., ”13” for Carriage return). Any 8-bit ASCII character, 0-255, can be used.
- *Maximum Frame Size Reached:* A maximum frame size must be defined. This is the maximum number of serial data bytes that will be carried in each UDP/TCP frame. When the maximum number of bytes is buffered, the packet

will be transmitted over the network. Allowed values are in range 1-1460 (bytes). Values above 255 are approximate.

- *Maximum Frame Delay Reached:* A maximum frame delay can be defined. This is the time, after the last received character in the buffer, the WeOS unit will wait until the buffered serial data is sent over the network. Allowed values are in range 1-2550 ms; If *maximum frame delay* is used with low data rates (see section 11.1.1), it should be set to at least one ”character time”.

12.1.3 Serial Over IP settings

- *General settings:*
  - *Mode:* Set operating mode.
    - *Server:* This is the default setting. In *server* mode, the unit will act as a TCP server, and listen for incoming call establishments. Only a single client can connect to the serial port at time. This mode can be used both in *point-to-point* serial extension (section 12.1.1.1) and *serial port redirector* (section 12.1.1.3) applications.
    - *Client:* In *client* mode, the unit will act as a TCP client, and initiate a connection to a remote TCP server. This mode can be used in the *point-to-point* serial extension application, see section 12.1.1.1.
    - *Peer:* In *peer* mode, UDP will be used for serial data transportation. This mode can be used both in the *point-to-point* (section 12.1.1.1) and *one-to-many* (section 12.1.1.2) serial extension applications.
      In the point-to-point case both peers will specify the IP address of the remote peer as the *destination*. For the *one-to-many* case there are many addressing options, see the item on *Addressing information* below.

    Default: **Server**
  - *Protocol Extensions:* Enable protocol extensions, e.g. RFC2217 Telnet extensions[2]. Needed to verify OEM licence of Serial/IP®. As of WeOS v4.3.0 this setting is only valid in *Server* mode.
  - *Serial Port:* Select which serial port to use. Default: **Disabled**
- *Data Packing Settings*: (see section 12.1.2 for further explanation)
  - *Frame separator character:* Define frame separator character, if any. Any 8-bit ASCII character, 0-255, can be used. Default: **Disabled**
  - *Maximum Frame Size:* Define maximum frame size in number of bytes. Allowed values are in range 1-1460 (bytes). Default: **1000 (bytes)**

  - *Maximum Frame Delay:* Define maximum frame delay in milliseconds. Allowed values are in range 1-2550 (ms). Default: **20 (milliseconds)**

- *Addressing information:*
  - *Listen:* Local interface and (UDP/TCP) port to listen to. This setting is only applicable in *Server* and *Peer* modes.
    In *Server* mode, the unit will accept incoming TCP connections to the IP address of the stated interface.
    In *Peer* mode, the unit will accept incoming UDP datagrams destined to the IP address of the stated interface, as well as broadcast IP packets[1] received on that interface.
    Default: **Disabled** (once an interface is selected, the default port is 9000)
  - *Multicast group:* Multicast group to *receive* data from. This is only applicable in *Peer* (UDP) mode. IP multicast addresses are in the following range: 224.0.0.0-239.255.255.255.
    When configured, the unit will accept packets to the stated multicast address, when received on the interface and (UDP) port declared in the *Listen* setting. Note: the unit will still accept unicast and broadcast packets as described in the *Listen* item above.
    Default: **Disabled**
  - *Destination/peer:* IP address and (UDP/TCP) port numbers to relay data to/from. This setting is only applicable in *Client* and *Peer* modes. In Client mode, the destination address should be the IP address of the (remote) Server.
    In Peer mode, it is possible to specify one or more destinations/peers (maximum 32), and the address can be IP unicast, broadcast[2], or multicast [3].
    Default: **Disabled**

[1] Both IP subnet broadcast packets (e.g., 192.168.1.255 on a 192.168.2.0/24 network), and data link IP broadcast (255.255.255.255) are accepted if received on the appropriate interface.

[2] Sending to the data link IP broadcast (255.255.255.255) will only work if the unit has a default gateway configured (see section 17.2.1). IP subnet broadcast (e.g., 192.168.1.255) is preferred.

[3] Sending data to a multicast address will only work if the unit has a default gateway configured (see section 17.2.1).

12.2 Managing Serial Over IP via the web interface

The Web interface provides configuration of the Serial Over IP.

12.2.1 Serial Over IP overview

Menu path: Configuration ⇒ Serial ⇒ Serial Over IP

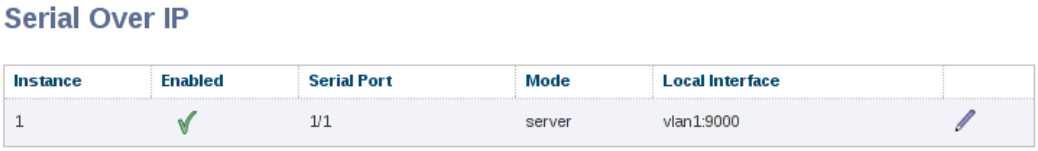
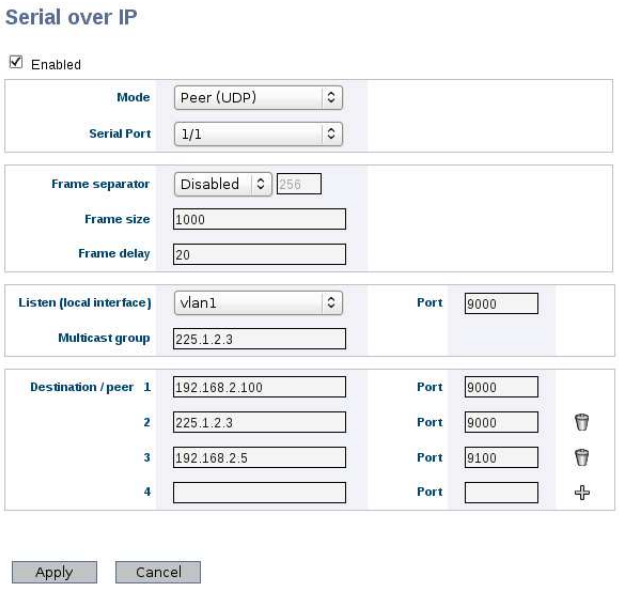


Figure 12.4: Serial Over IP configuration settings overview

Click on the Edit icon ( ) to edit the settings of a specific Serial Over IP instance.

12.2.2 Edit Serial Over IP Settings

Menu path: Configuration ⇒ Serial ⇒ Serial Over IP⇒ 



On this page you can change the settings for the Serial Over IP.

Mode	Set operating mode: Server (TCP), Client (TCP), or Peer (UDP).
Protocol Extensions	Enable/disable protocol extensions (only Server Mode).
Serial Port	Serial port to use.
Frame separator	Define frame separator character.
Frame size	Define maximum frame size in characters.
Frame delay	Define maximum frame delay in milliseconds.
Listen (local end)	Local interface and (UDP/TCP) port to listen to (only Server and Peer Modes). Default port: 9000
Multicast group	Multicast address to listen to (only Peer Mode).
Destination/peer	IP address and (UDP/TCP) port for remote peer(s)/destinations (only Client and Peer Modes). In Peer Mode, several destination/peer entries can be configured, and the destination address can be unicast, broadcast or multicast. Default port: 9000

12.3 Managing Serial Over IP via the CLI interface

The table below shows Serial Over IP management features available via the CLI.

Command	Default	Section
Configure Serial Over IP settings		
seroip		Section 12.3.1
[no] mode <server|client|peer>	server	Section 12.3.2
[no] port <SERIAL-PORT>	Disabled	Section 12.3.3
[no] protocol <raw|telnet>	Disabled	Section 12.3.4
[no] listen <IFACE[:PORT]>	Disabled	Section 12.3.5
[no] mcast-group <ADDRESS>	Disabled	Section 12.3.6
[no] frame-separator <0-255>	Disabled	Section 12.3.7
[no] frame-delay <1-2550>	20	Section 12.3.8
[no] frame-size <1-1460>	1000	Section 12.3.9
[no] peer <ADDRESS[:PORT] [,ADDRESS:PORT,...]>	Disabled	Section 12.3.10
Show Serial Over IP settings		
seroip		
show		Section 12.3.11
show mode		Section 12.3.12
show port		Section 12.3.13
show protocol		Section 12.3.14
show listen		Section 12.3.15
show mcast-group		Section 12.3.16
show frame-separator		Section 12.3.17
show frame-delay		Section 12.3.18
show frame-size		Section 12.3.19
show peer		Section 12.3.20

12.3.1 Managing Serial Over IP settings

Syntax `seroip`

Context *Global Configuration* context

Usage Enter the Serial Over IP configuration context.

Default values Not applicable.

Error messages None defined yet.

12.3.2 Setting Mode

Syntax `[no] mode <server|client|peer>`

Context *seroip* context

Usage Set Serial Over IP mode.

Default values server

Error messages None defined yet.

12.3.3 Setting Serial Port

Syntax `[no] port <SERIAL-PORT>`

Context *seroip* context

Usage Set serial port

Default values Disabled (**”no port”**)

Error messages None defined yet.

12.3.4 Setting Protocol Extensions

Syntax `[no] protocol <raw|telnet>`

Context *seroip* context

Usage Set protocol extensions. This is only applicable in *server* mode (section 12.3.2).

When accessing the serial port with Westermo’s OEM version of Serial/IP® the protocol extension setting should be **”protocol telnet”**.

Use **”no protocol”** (or **”protocol raw”**) to disable protocol extensions.

Default values Disabled (**”no protocol”**)

Error messages None defined yet.

12.3.5 Setting listen interface and port

Syntax `[no] listen <IFACE[:PORT]>`

Context *seroip* context

Usage Setting local interface and (UDP/TCP) port to listen to.

Default values Disabled (**”no listen”**) When enabled, the default port is 9000.

Error messages None defined yet.

12.3.6 Setting multicast group

Syntax `[no] mcast-group <ADDRESS>`
Context *seroip* context
Usage Multicast group to listen on. Note, this is only used in peer mode.
Default values Disabled (**"no mcast-group"**)
Error messages None defined yet.

12.3.7 Setting Frame Separator

Syntax `[no] frame-separator <0-255>`
Context *seroip* context
Usage Define frame separator character, if any. Any 8-bit ASCII character, 0-255, can be used.
Use **"no frame-separator"** to disable frame separator checking in the packging algorithm.
Default values Disabled (**"no frame-separator"**)
Error messages None defined yet.

12.3.8 Setting Frame Delay

Syntax `[no] frame-delay <1-2550>`
Context *seroip* context
Usage Define maximum frame delay in milliseconds.
Use **"no frame-delay"** to disable maximum delay checking in the packing algorithm.
Default values 20 (milliseconds)
Error messages None defined yet.

12.3.9 Setting Frame Size

Syntax `[no] frame-size <1-1460>`
Context *seroip* context
Usage Define maximum frame size in bytes (this is part of the packing algorithm).
Use **"no frame-size"** to reset the maximum frame size to the default value.
Default values 1000 (bytes)
Error messages None defined yet.

12.3.10 Setting peer address and port

Syntax `[no] peer <ADDRESS[:PORT][,ADDRESS:PORT,...]>`

Context *seroip* context

Usage Remote destinations/peer(s) to relay data to/from. Note, this is only used in client or peer mode. If PORT is omitted the default port 9000 will be used.

Default values Disabled (**"no peer"**)

Error messages None defined yet.

12.3.11 Show All Settings of a Serial Over IP

Syntax `show`

Context *seroip* context

Usage Show all configuration settings for Serial Over IP.

Default value Not applicable.

12.3.12 Show Show Serial Over IP Mode Setting

Syntax `show mode`

Context *seroip* context

Usage Show the erial Over IP mode setting

Default value Not applicable.

12.3.13 Show Show Serial Over IP Port Setting

Syntax `show port`

Context *seroip* context

Usage Show the erial Over IP port setting

Default value Not applicable.

12.3.14 Show Show Serial Over IP Protocol extensions Setting

Syntax `show protocol`

Context *seroip* context

Usage Show the serial Over IP protocol extensions setting

Default value Not applicable.

12.3.15 Show Show Serial Over IP Listen Setting

Syntax `show listen`
Context *seroip* context
Usage Show the serial Over IP listen setting
Default value Not applicable.

12.3.16 Show Show Serial Over IP Multicast group Setting

Syntax `show mcast-group`
Context *seroip* context
Usage Show the serial Over IP multicast group setting
Default value Not applicable.

12.3.17 Show Show Serial Over IP Frame Separator Setting

Syntax `show frame-separator`
Context *seroip* context
Usage Show the serial Over IP frame separator setting
Default value Not applicable.

12.3.18 Show Show Serial Over IP Frame Delay Setting

Syntax `show frame-delay`
Context *seroip* context
Usage Show the serial Over IP frame delay setting
Default value Not applicable.

12.3.19 Show Show Serial Over IP Frame Size Setting

Syntax `show frame-size`
Context *seroip* context
Usage Show the serial Over IP frame size setting
Default value Not applicable.

12.3.20 Show Show Serial Over IP Peer Setting

Syntax `show peer`

Context *seroip* context

Usage Show the serial Over IP peer setting

Default value Not applicable.

Chapter 13

Virtual LAN

WeOS supports static port based VLANs and VLAN tagging according to IEEE 802.1Q[5]. In addition, WeOS supports Westermo Adaptive VLAN Trunking (AVT) to simplify VLAN configuration in larger Westermo networks.

Section 13.1 provides general information about the VLAN properties and VLAN management features in WeOS. Section 13.2 covers VLAN settings via the Web interface and section 13.3 VLAN settings via the CLI.

13.1 Overview of VLAN Properties and Management Features

Table 13.1 summarises VLAN management features in WeOS. Section 13.1.1 provides general VLAN information and sections 13.1.2-13.1.6 contain further information on specific VLAN features.

13.1.1 Introduction to VLANs

Virtual LAN (VLAN) technology is used to create a set of separate LANs over a single physical LAN infrastructure. Each VLAN constitutes a broadcast domain, and traffic on one VLAN is (logically) isolated from traffic on another VLAN. WeOS supports creation of static port based VLANs and VLAN tagging as described further in this section. We start with two examples to explain the terms *untagged* and *tagged*.

Fig. 13.1 shows a situation where three networks, the *ADMIN* VLAN, the *OFFICE* VLAN, and the *MARKETING* VLAN share a single switch.

Feature	Web (Sec. 13.2)	CLI (Sec. 13.3)	General Description
General VLAN functionality			
Enable/disable dynamic VLAN	X	X	Sec.13.1.7
Per VLAN functionality			
Add/modify/delete VLAN	X	X	Secs.13.1.1-13.1.3
Enable/disable VLAN	X	X	
VLAN name		X	
Untagged/Tagged ports	X	X	Sec.13.1.1
VLAN priority	X	X	Sec.13.1.4
IGMP Snooping	X	X	Sec.13.1.5
VLAN CPU Channel		X	Sec.13.1.6
Forbid ports	X	X	Sec.13.1.7
View VLAN configuration	X	X	
View VLAN status	X	X	

Table 13.1: Summary of VLAN management features.

- Each VLAN is assigned a VLAN identifier, a VLAN ID (VID); in this example VIDs 1 (ADMIN), 2 (OFFICE) and 3 (MARKETING).
- Each VLAN is assigned a set of ports. In this example ports 1/1-1/2 are associated with the ADMIN VLAN, Ports 2/1-2/4 with the OFFICE VLAN, and ports 2/5-2/8 with the MARKETING VLAN.

In this example we have assumed that only regular hosts (PCs, servers, etc.; not other switches) attach to the ports of the switch. Traffic sent and received on each switch port are regular Ethernet packets (without VLAN headers), and here we refer to this by saying that the switch ports are associated with their respective VLAN *untagged*.

> A port associated *untagged* on a VLAN, will send and receive regular Ethernet packets (i.e., without VLAN header) on that port.

Consider the case where a PC attached to port 2/1 of the switch in fig. 13.1 transmits a *broadcast* packet. That packet will be forwarded onto all other ports of VLAN 2 (OFFICE), i.e., ports 2/2-2/4, but not to any of the other ports.

Fig. 13.2 shows a situation where three networks, the *ADMIN* VLAN, the *OFFICE* VLAN, and the *MARKETING* VLAN share two switches as well as the connection

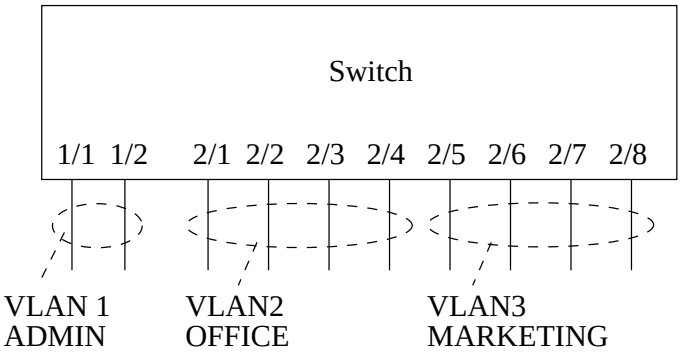


Figure 13.1: VLANs sharing a single switch.

between them.

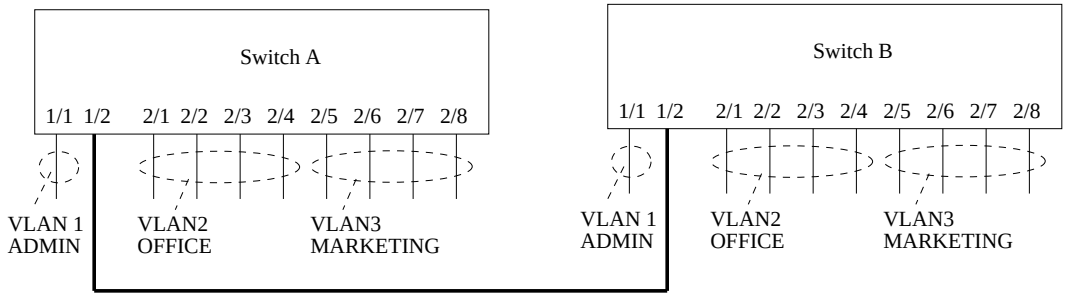


Figure 13.2: VLANs sharing two switches and the connection between them.

- As in the previous example, each VLAN is assigned a VID; in this example VIDs 1 (ADMIN), 2 (OFFICE) and 3 (MARKETING).
- Each VLAN is assigned a set of ports. (For simplicity of this example, we have chosen to use the same port assignment on both switches.) Port 1/1 is associated (untagged) with the ADMIN VLAN, Ports 2/1-2/4 are associated (untagged) with the OFFICE VLAN, and ports 2/5-2/8 are associated (untagged) with the MARKETING VLAN.

In addition, port 1/2, where the cable between the two switches is connected, is associated with all three VLANs. In order for the switches to distinguish which VLAN a packet belongs to when transmitted over a shared connection, the switch will insert a VLAN header (VLAN tag) into the packet, which includes information about the VLAN ID (here 1, 2 or 3). Thus, in this example port 1/2 would be

associated with VLAN 1, 2 and 3 *tagged*[1].

> A port associated *tagged* on a VLAN, will send and receive *tagged* Ethernet packets (i.e., Ethernet packets including a VLAN header) on that port.

Consider the case where a PC attached to port 2/1 of *switch A* in fig. 13.2 transmits a *broadcast* packet. That packet will be forwarded onto ports 2/2-2/4 of switch A *untagged*, and onto port 1/2 of switch A *tagged* with VID 2. When the *tagged* packet is received on port 1/2 on switch B, that switch can determine that the packet belongs to VLAN 2, and will forward it onto ports 2/1-2/4 *untagged*.

> A port cannot be associated with more than one VLAN *untagged*. A port cannot be associated both *untagged* and *tagged* with the same VLAN.

We refer to the VLAN with VID 1 as the *switch default VLAN*. Ports not associated with any VLAN (*untagged* or *tagged*) will automatically be associated with the default VLAN. Section 13.1.3 provides more information on the *default VLAN*.

For each VLAN on a switch, an associated network interface will be created. The name of a VLAN network interface is *vlan<VID>*, e.g., *vlan1* for VLAN 1, and *vlan100* for VLAN 100. The network interface can be assigned an IP address (IPv4), and the switch can then be managed remotely via that VLAN. It is also possible to *route* IP traffic between network interfaces. For more information on network interfaces and routing, see chapter 17.
Some Westermo switches have multiple 100 Mbit/s channels to the CPU. Section 13.1.6 describes how VLANs can be mapped to different CPU channels to achieve increased routing performance.

Layer-2 priority was described in a previous chapter, see section 9.1.3. In addition to different per port priority settings, it is possible to assign specific layer-2 priority per VLAN, see section 13.1.4.

The switch supports efficient distribution of IP multicast packets by use of *IGMP snooping*. See section 13.1.5 for more information on per VLAN IGMP snooping features.

[1] It is recommended that a port, which is shared between several VLANs, is associated *tagged* with all those VLANs, however, it is possible to configure the port *untagged* on one VLAN and *tagged* on all other VLANs without risk for ambiguity.

The switch provides support for dynamic VLANs by Westermo Adaptive VLAN Trunking (AVT). AVT can be used to simplify VLAN configuration in larger Westermo LAN infrastructures. AVT is described further in section 13.1.7.

13.1.2 Supported number of VLANs and VLAN integrity

Every VLAN needs to be associated with a unique VLAN ID (VID).

- Switches *support* configuration of up to 64 simultaneous VLANs, but have *capability* of 256 VLANs.
- Valid VIDs for configuration are in range 1-4094.
- Some VLAN IDs are reserved for specific use - currently this concerns a set of VIDs in use by the FRNT protocol, see chapter 14.

Switches only accept packets for VLANs to which the inbound port is associated. Additional rules for accepting a packet is described below:

- When an untagged packet is received on a port, that packet will be mapped to the port's default VID. If the port is associated with that VLAN (tagged or untagged), the packet will be accepted, otherwise dropped.
- The port's default VID will be the VID of the VLAN to which the port is associated *untagged*. If the port is not associated *untagged* to any VLAN, the default VID is set to the *fall-back default-VID* (see also section 9.1.7) if configured, otherwise to VID 1.
- *Priority tagged* packets, i.e., packets with VID 0, will be associated with the port's default VID.
- Typically *tagged* packets (VID in range 1-4094) or priority tagged packets (VID 0) are only accepted on ports where there is at least on VLAN associated *tagged*. In addition, the packet will only be accepted if the inbound port is associated (*untagged* or *tagged*) the the VLAN of the packet.

A common MAC address database is used for all VLANs (shared VLAN learning).

13.1.3 Switch default VLAN

In WeOS the VLAN with VID 1 (VLAN 1) is denoted as the *switch default VLAN*. Ports not associated with any VLAN (neither *untagged* nor *tagged*) will automatically be configured *untagged* to the switch default VLAN. This could happen when a port is removed from a VLAN, or when a whole VLAN is removed.

Note: *The main purpose of the switch default VLAN is to avoid loss of remote manageability of a switch due to a change in the VLAN configuration. Without a default VLAN, the user would not be able to access the switch remotely, if the ports used to connect to the switch are removed from all VLANs (unintentionally or deliberately). With the default VLAN feature, the switch is still manageable via those ports, given that proper IP and firewall settings are configured for the network interface associated with the switch default VLAN.*

The switch default VLAN cannot be removed. However, it is possible to remove all ports from the default VLAN by assigning them to other VLANs.

13.1.4 VLAN Priority

It is possible to assign an IEEE 802.1p priority to a VLAN. This feature can be useful when an operator likes to assign a higher priority to traffic on a certain VLAN, e.g., a VLAN dedicated for IP telephony.

When a *VLAN priority* is configured, all packets associated with that VLAN will be treated according to the given VLAN priority, rather than basing the packet's priority on VLAN tag priority, IP ToS/DiffServ or inbound port identifier. For more information on layer-2 priority, see section 9.1.3.

13.1.5 IGMP Snooping and VLANs

Switches use IGMP snooping for efficient distribution of IP(v4) multicast over the LAN. With IGMP snooping *enabled* on a VLAN, IP multicast packets will only be forwarded onto ports leading to a receiver of that IP multicast address, or to ports assumed to lead to an IP multicast router.

With IGMP snooping *disabled* on a VLAN, multicast traffic will be forwarded on all ports of that VLAN, i.e., it is treated similar to broadcast traffic.

By default IGMP snooping is enabled on each newly created VLAN. More information on IGMP Snooping and IGMP Snooping settings is found in chapter 17.

13.1.6 Mapping VLANs to a CPU channel

A switch can have multiple 100 Mbit/s channels to the switch CPU. By default every new VLAN (with a network interface) is mapped to CPU channel ”0” (zero).

On devices with multiple CPU channels increased routing performance may be achieved by assigning different VLANs to different CPU channels. E.g., if VLANs 1 and 2 are mapped to the same CPU channel, the maximum theoretical routing throughput between the two VLAN interfaces is 50 Mbit/s full duplex, while the maximum theoretical routing throughput would be 100 Mbit/s full duplex if these VLANs were mapped to different CPU channels.

Routing performance may also be limited by CPU performance and packet size.

A VLAN can only be mapped to a single CPU channel.

13.1.7 Dynamic VLANs

WeOS provides dynamic VLAN support via the Westermo Adaptive VLAN Trunking (AVT) protocol. With AVT enabled, VLAN configuration on *inter-switch links* is simplified - once a switch detects that it is connected to another switch, all VLANs defined on the local switch will automatically be added to that port, see fig. 13.3.

Future versions of WeOS may include dynamic VLAN support via the standard IEEE GVRP[5] protocol in addition to Westermo AVT.

13.1.7.1 Determining Inter-Switch Ports

To determine if a port on a switch is connected to another switch, AVT will utilise information from the FRNT and RSTP protocols:

- *FRNT:* If FRNT is enabled on the switch, any port configured as an FRNT port will be classified as an inter-switch port by AVT. If FRNT is disabled, or if the FRNT port configuration is changed, AVT will adapt its inter-switch port classification accordingly. For more information on FRNT, see chapter 14.
- *RSTP:* If RSTP is enabled on a port, AVT will consider the reception of an RSTP or STP message as a sign that it is connected to another switch on the receiving port. The port will continue to be classified as an inter-switch port until the link goes down or until RSTP is disabled on that port. For more information on RSTP, see chapter 15.

13.1.7.2 Dynamic addition/deletion of VLANs to Inter-Switch Ports

Once a port has been defined as an inter-switch port, that port will dynamically be associated (tagged) with all VLANs *configured on the switch*. The exception is when that port has been configured in association mode *forbid* on some VLAN(s) - the port will *not* be associated with those VLANs.

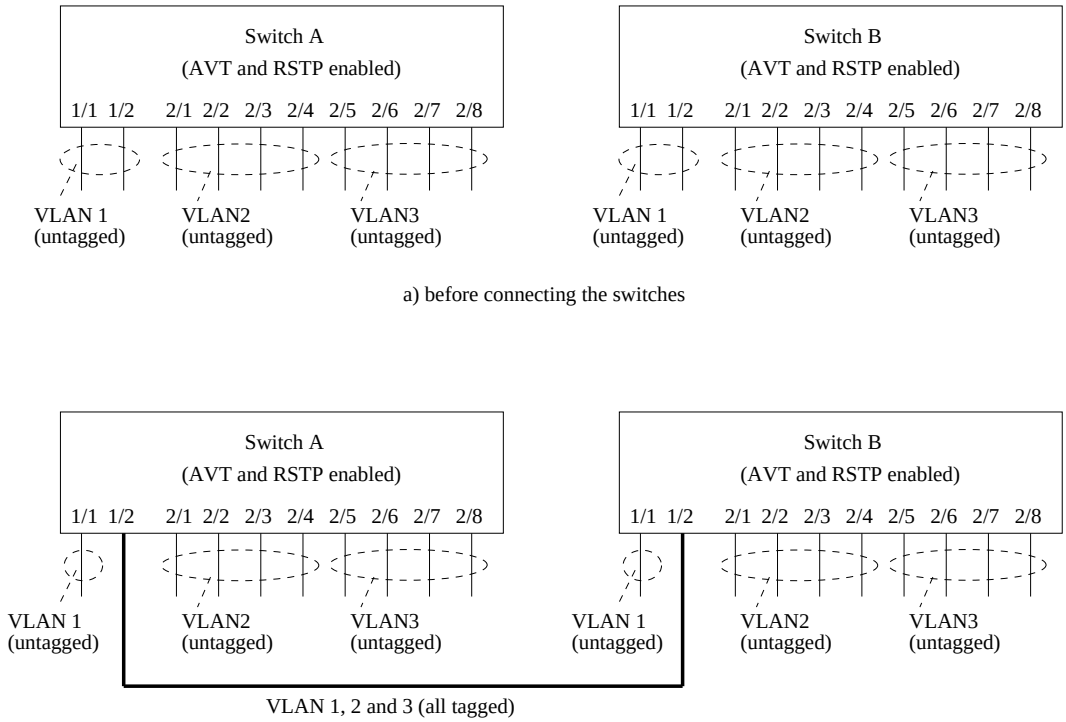


Figure 13.3: Using Adaptive VLAN trunking (AVT) to dynamically add VLANs to inter-switch ports.

Further details of the mechanism to associate VLANs dynamically to an inter-switch port are given below:

- *Association mode of dynamically added VLANs:* All VLANs configured on the switch will be associated *tagged* by AVT. This applies even to those VLANs configured *untagged* on that port. Fig. 13.3 shows an example.

  Note: *As AVT only considers the VLANs configured on the (local) switch when adding VLANs to an inter-switch port, the operator of the LAN infrastructure should ensure that all switches have the same set of VLANs defined. Otherwise the VLANs forwarded by different switches will be inconsistent, resulting in lack of full connectivity on some VLAN(s).*

- *Removing dynamically added VLANs:* When a port loses its status as inter-switch port, all VLANs dynamically added to that port will be removed. The port will then only be associated with the VLANs it has been configured with, and with association mode (tagged or untagged) according to the configuration.
- *Prohibiting that a VLAN is added to a port:* It is possible to prohibit that some VLAN(s) is dynamically added to a port even when AVT is enabled. This feature is useful when the unit acts as a routing switch, where traffic between some ports should be *routed* rather than *switched*.

  To prohibit that a VLAN is dynamically added to a port, that port should be configured with association mode *forbid* on that VLAN.

  As of WeOS version v4.3.0 the *forbid* association mode only hinders a port to be added to a VLAN dynamically via AVT. Ports not configured untagged/tagged with any VLAN will still be mapped to the switch default VLAN (VLAN 1), irrespective if that port is configured as *forbid* on VLAN 1. For more information about the switch default VLAN, see section 13.1.3.

13.1.7.3 Prohibit disabling of Inter-Switch Ports

A port determined as inter-switch port by AVT will not be possible to disable by management (Web, CLI, SNMP, etc.). This feature is added in order to avoid unintentional loss of connectivity to the switch.

13.2 Managing VLAN settings via the web interface

Menu path: Configuration ⇒ VLAN ⇒ VLANS

When entering the VLAN configuration page you will be presented to a list of all VLANs configured on your switch, see below. Here you get an overview of the settings for all VLANs and you can create or delete VLANs. The default VLAN (VID 1) cannot be removed (see section 13.3.3). To change the settings for a specific VLAN, click the edit icon which will take you to the VLAN settings edit page.

VID	The VLAN's unique identifier.
Enabled	Used to enable or disable a VLAN. Ports on a disabled VLAN are temporarily moved to the system default VLAN. A green check-mark means the VLAN is enabled, and a dash means it is disabled.
Name	The name of the VLAN. Automatically generated from VLAN identifier when the VLAN is created using the web tool.
Prio	VLAN priority setting. Values between 0-7 or disabled. See also section 13.1.4. Disabled is shown using a dash.
IGMP	In the VLAN overview table a green check-mark means enabled, and a dash means disabled on a specific VLAN. See section 13.1.5 for more information.
Interface	A list of associated interfaces.

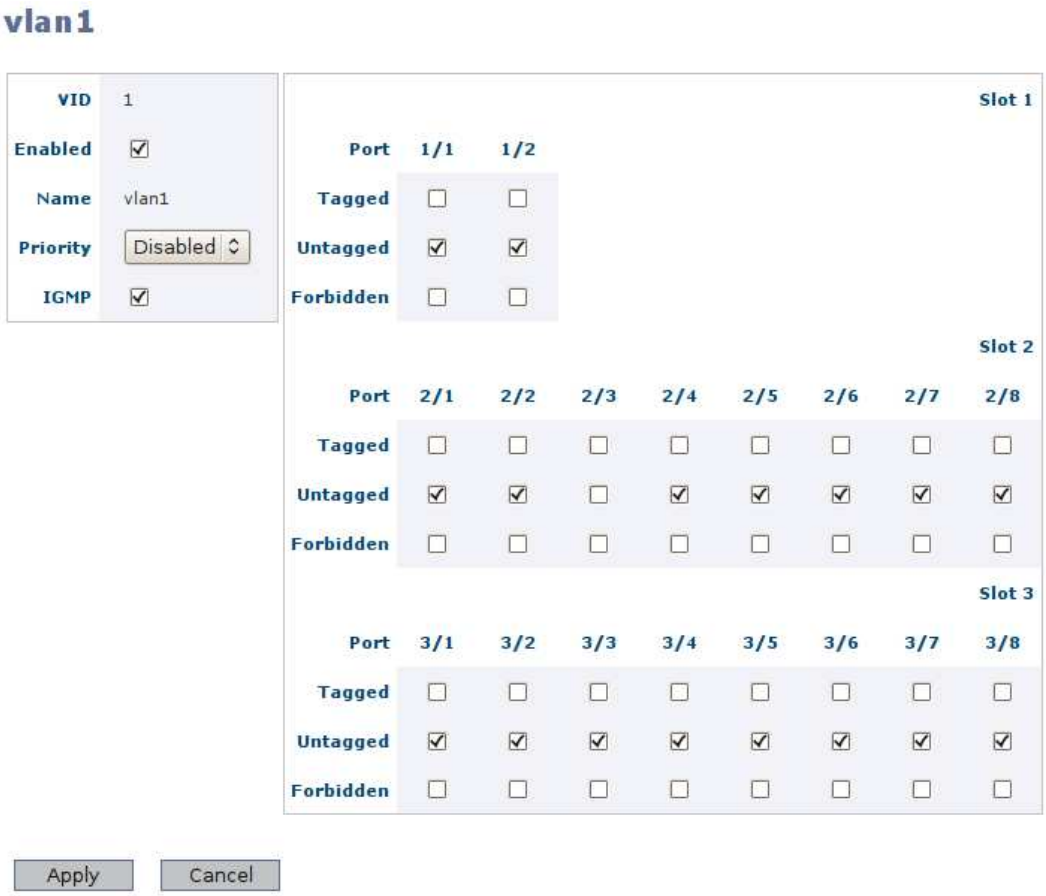
Continued on next page

	Continued from previous page
Port(s)	List of ports assigned to each VLAN. Grouped as **tagged** and **untagged** for ports configured statically to this VLAN, or as **dynamic** for ports dynamically added to this VLAN by Westermo Adaptive VLAN Trunking (AVT). (See section 13.1.7 for more information on AVT). 1/1-1/3 means port 1/1, 1/2 and 1/3, the first and last port, and all ports in-between.
New VLAN	Click this button to create a new VLAN. You will be presented to a form where you can configure the new VLAN.
 **Edit**	Click this icon to edit a VLAN.
 **Delete**	Click this icon to remove a VLAN. You will be asked to acknowledge the removal before it is actually executed.

13.2.1 Edit VLAN settings using the web interface

Menu path: Configuration ⇒ VLAN ⇒ VLANS ⇒ 

When clicking the *Edit* icon for a VLAN you will be presented to the VLAN edit page.



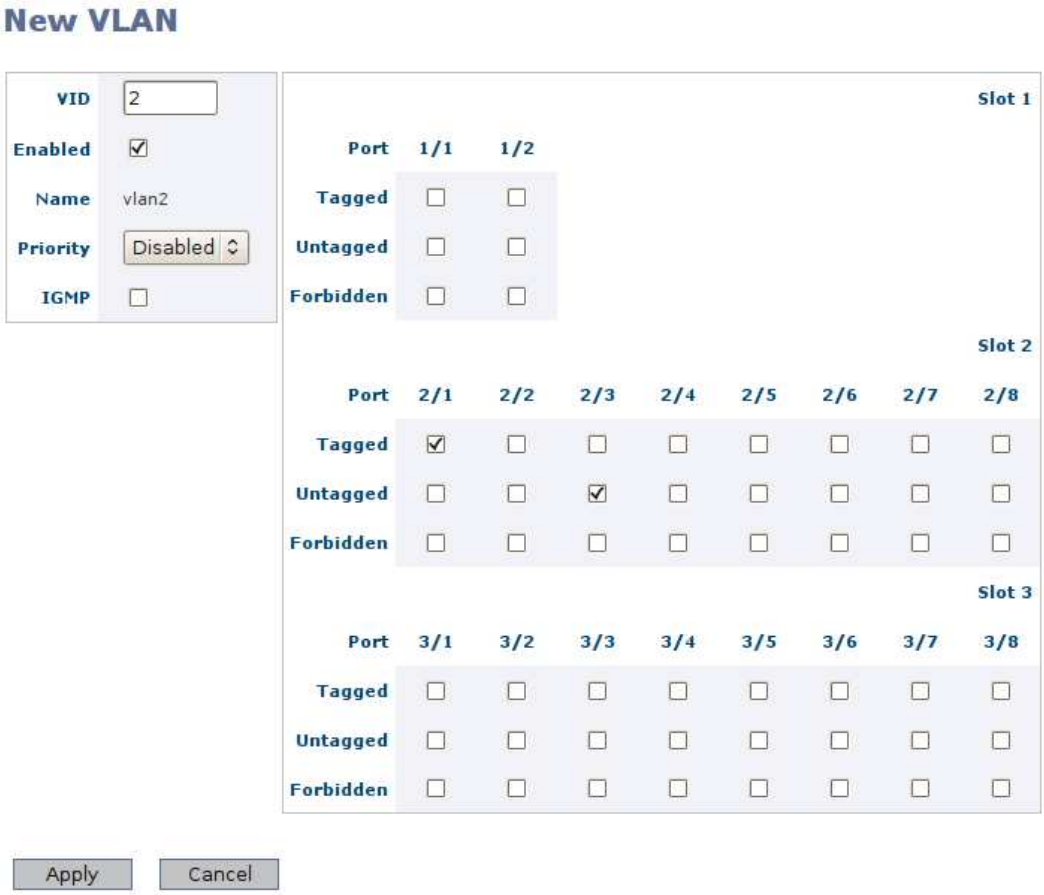
On **`VLAN Edit page`** you can change the settings for the VLAN as described below:

VID	The VLAN's unique identifier. You cannot change the VID of an already created VLAN.
Enabled	Used to enable or disable a VLAN. Ports on a disabled VLAN are temporarily moved to the system default VLAN. To enable the VLAN - check the box, to disable un-check the box.
Name	The name of the VLAN. You cannot change the VLAN name using the web tool.
Prio	VLAN priority setting. Values between 0-7 or disabled. See also section 13.1.4. Select the desired VLAN priority in the drop down list, or select disable to disable VLAN priority.
IGMP	To enable IGMP snooping on this VLAN - check the box, to disable IGMP un-check the box. See section 13.1.5 for more information.
Port	The ports on your switch is grouped as on the actual hardware, in slots. To assign a port to the VLAN, check the **Tagged** or **Untagged** check-box located underneath the port label. In the picture above you see all ports but 2/3 associated *untagged* to VLAN 1. A port may not be associated tagged and untagged to the same VLAN at the same time. It may not be associated untagged to more than one VLAN at a time. If you associate a port untagged to a VLAN any existing untagged association to another VLAN on that port will automatically be removed. You will be notified if this happens. For more information on the *tagged* and *untagged* association modes, see section 13.1.1. The **Forbidden** check-box is used to specify that this port can not be dynamically assigned to this VLAN (see section 13.1.7 for more information on dynamic VLANs).

13.2.2 Create a new VLAN using the web interface

Menu path: Configuration ⇒ VLAN ⇒ VLANS ⇒ **New VLAN**

When clicking the **New VLAN** button you will be presented to the **new VLAN** page.



The **New VLAN** and the **Edit VLAN** pages differ only by the possibility to change the VID (VLAN ID). See Section 13.2.1 for additional attribute descriptions.

VID	The VLAN's unique identifier.
Name	The VLAN name will be automatically generated when using the web management tool. The name is shown directly when you change and leave the VID field if your browser is JavaScript[1] enabled, otherwise it will be generated when you click the **Apply** button.

[1] JavaScript is a trademark of Sun Microsystems.

13.3 Managing VLAN settings via the CLI

The table below shows VLAN management features available via the CLI.

Command	Default	Section
General VLAN Configuration		
[no] vlans		Section 13.3.1
[no] dynamic <adaptive|gvrp>		Section 13.3.2
Per VLAN Configuration		
[no] vlan <VID>		Section 13.3.3
[no] enable	Enabled	Section 13.3.4
name <VLANNAME>	vlan<VID>	Section 13.3.5
[no] untagged <PORTLIST>		Section 13.3.6
[no] tagged <PORTLIST>		Section 13.3.7
[no] forbid <PORTLIST>		Section 13.3.8
[no] priority <0-7>	Disabled	Section 13.3.9
[no] igmp	Enabled	Section 13.3.10
channel <CHANNELID>	0	Section 13.3.11
Show VLAN configuration		
show vlan [VID]	All VLANs	Section 13.3.12
show vlans		Section 13.3.13
vlans		
show dynamic		Section 13.3.14
vlan <VID>		
show enable		Section 13.3.15
show name		Section 13.3.16
show untagged		Section 13.3.17
show tagged		Section 13.3.18
show priority		Section 13.3.19
show igmp		Section 13.3.20
show channel		Section 13.3.21
Show VLAN status		
show vlans		Section 13.3.22

13.3.1 Managing general VLAN settings

Syntax `[no] vlans`

Context *Global Configuration* context

Usage Enter the general VLAN context (*vlans*). The general VLAN context can be used to configure VLAN settings applicable to all VLANs.

Use **"no vlans"** to remove all VLANs except the switch default VLAN (VLAN 1). All ports will be configured *untagged* on VLAN 1.

Default values Not applicable.

Error messages None defined yet.

13.3.2 Enable dynamic VLAN

Syntax `[no] dynamic <adaptive|gvrp>`

Context *General VLAN* context (*vlans)*

Usage Use the **"dynamic adaptive"** command to enable Westermo Adaptive Dynamic Trunking (AVT) on the switch. For more information on AVT in section 13.1.7.
Future versions of WeOS may include support for dynamic VLAN via GVRP in addition to AVT, but currently only AVT is supported.

Use **"no dynamic"** to disable dynamic VLAN support.

Default values Not applicable.

Error messages None defined yet.

13.3.3 Managing individual VLANs

Syntax `[no] vlan <VID>`

Context *Global Configuration* context

Usage Enter VLAN context of the given VID. If this is a new VLAN, the VLAN will be created first upon leaving the VLAN context with *end* or *leave*.
Use **"no vlan <VID>"** to remove an existing VLAN. The default VLAN (VLAN 1) cannot be removed. Removal of a VLAN may imply that some ports will no longer be associated with any VLAN - such ports will be configured to the default VLAN (VLAN 1) untagged.

Default values Not applicable.

Error messages None defined yet.

13.3.4 Enable/disable a VLAN

Syntax `[no] enable`

Context *VLAN* context

Usage Enable or disable a VLAN. A disabled VLAN is similar to a deleted VLAN, except that its configuration is stored, and will be activated when the VLAN is *enabled*. That is, when a VLAN is disabled, its ports may be moved onto the default VLAN (unless they are associated with another VLAN), and any network interface associated with the VLAN will be disabled.

Default values *enable*

Error messages No error message defined (yet).

13.3.5 VLAN name

Syntax `name <ID>`

Context *VLAN* context

Usage Specify VLAN name, i.e., VLAN description. Max 15 characters, only alpha-numerical characters ([a-z,A-Z,0-9]) allowed.

Default values If no VLAN **"`name`"** command is given, the VLAN name defaults to *vlanVID*, e.g., *vlan100* for VID 100.

Error messages No error message defined (yet).

13.3.6 Manage untagged ports

Syntax `[no] untagged <PORT|PORTLIST>`

Context *vlan* context

Usage Associate port(s) with this VLAN VID in *untagged* mode. Only a single VLAN VID can be associated *untagged* with each port. Ports associated with a VLAN VID *untagged* will have that VID as *default VID* - this will have precedence over any (fall-back) default VID configuration set in *port* context.
Use **"`no untagged <PORTLIST>`"** to remove *untagged* ports from a VLAN. If removal of an *untagged* port implies that the port is no longer associated with any VLAN, that port will be configured to VLAN 1 *untagged*.

Default values Factory default lets all ports be associated with the default VLAN (VLAN 1) *untagged*. For new VLANs, ports must explicitly be added.

Error messages

- A notification message is given in case the addition of port as *untagged* on one VLAN implies that the same port will be removed as *untagged* on another VLAN.

- A notification message is given in case the addition of port as *untagged* on one VLAN implies that the same port will be removed as *tagged* on the same VLAN (a port cannot be associated both *tagged* and *untagged* with the same VLAN).

A **”PORTLIST”** is a comma separated list of port ranges without intermediate spaces, e.g., **”1/1-1/3,2/3”**.

13.3.7 Manage tagged ports

Syntax `[no] tagged <PORT|PORTLIST>`

Context *vlan* context

Usage Associate port(s) with this VLAN VID in *tagged* mode.
Use **”no tagged <PORTLIST>”** to remove *tagged* ports from a VLAN. If removal of a *tagged* port implies that the port is no longer associated with any VLAN, that port will be configured to VLAN 1 *untagged*.

Default values Not applicable.

Error messages A notification message is given in case the addition of port as tagged on one VLAN implies that the same port will be removed as *untagged* on the same VLAN (a port cannot be associated both *tagged* and *untagged* with the same VLAN).

A **”PORTLIST”** is a comma separated list of port ranges without intermediate spaces, e.g., **”1/1-1/3,2/3”**.

13.3.8 Manage forbidden ports

Syntax `[no] forbid <PORT|PORTLIST>`

Context *vlan* context

Usage Prohibit that ports are dynamically added (AVT) to this VLAN ID, see also sections 13.1.7 and 13.3.2.
Use **”no forbid <PORTLIST>”** to remove ports from the list of ports forbidden to be associated with this VLAN.

Default values Not applicable.

Error messages None defined.

A **”PORTLIST”** is a comma separated list of port ranges without intermediate spaces, e.g., **”1/1-1/3,2/3”**.

13.3.9 VLAN priority setting

Syntax `[no] priority <0-7>`

Context *vlan* context.

Usage Set the (IEEE 802.1p) priority associated with this VLAN. Incoming packets associated with this VLAN will receive this priority.

"no priority" will disable VLAN priority for this VLAN. Priority for packets associated with this VLAN will then be based on port priority settings.

Default values Disabled (**"no priority"**).

Error messages None defined yet.

13.3.10 VLAN IGMP Snooping

Syntax `[no] igmp`

Context *vlan* context.

Usage Enable, or disable IGMP Snooping for this VLAN.

Default values IGMP snooping enabled.

Error messages None defined yet.

13.3.11 CPU channel mapping

Syntax `channel <CHANNELID>`

Context *VLAN* context.

Usage Specify CPU channel to use for this VLAN. The channel identifier can take values in the range <0-CHANNELIDMAX>. The purpose of this command is to improve routing performance by mapping VLANs to different CPU channels, see section 13.1.6.

Default values 0 (zero), i.e., by default all VLANs will use channel 0.

Error messages None defined yet.

The number of channels, and CHANNELIDMAX can be found using the **"show system-information"** command, see section 7.3.2.

13.3.12 Show VLAN configuration

Syntax `show vlan [<VID>]`

Context *Global Configuration* context. Also available as **"show"** command within the VLAN context.

Usage Show VLAN configuration for the given VLAN VID (or all VLANs). The output format is different when showing configuration information for an individual VLAN or all VLANs.

Default values All VLANs, i.e., if no VID is provided, information on all configured VLANs will be shown.

Error messages None defined yet.

13.3.13 Show VLAN configuration (all VLANs)

Syntax `show vlans`

Context *Global Configuration* context.

Usage Show VLAN configuration for all VLANs (same as **`"show vlan"`**, see section 13.3.12).

Default values Not applicable.

Error messages None defined yet.

13.3.14 Show dynamic VLAN setting

Syntax `show dynamic`

Context General VLAN context. (*vlans*)

Usage Show whether dynamic VLAN is enabled or disabled. If enabled, the type of VLAN configured is listed (as of WeOS v4.3.0 only Westermo Adaptive VLAN Trunking is supported).

Default values Not applicable.

Error messages None defined yet.

13.3.15 Show VLAN enable/disable setting

Syntax `show enable`

Context *VLAN* context.

Usage Show whether VLAN is enabled or disabled.

Default values Not applicable.

Error messages None defined yet.

13.3.16 Show VLAN name setting

Syntax `show name`

Context *VLAN* context.

Usage Show the configured VLAN name.

Default values Not applicable.

Error messages None defined yet.

13.3.17 Show untagged ports setting

Syntax `show untagged`

Context *VLAN* context.

Usage Show the untagged ports configured for this VLAN.

Default values Not applicable.

Error messages None defined yet.

13.3.18 Show tagged ports setting

Syntax `show tagged`

Context *VLAN* context.

Usage Show the tagged ports configured for this VLAN.

Default values Not applicable.

Error messages None defined yet.

13.3.19 Show VLAN priority setting

Syntax `show priority`

Context *vlan* context..

Usage Show VLAN priority setting.

Default values Not applicable.

Error messages None defined yet.

13.3.20 Show IGMP snooping setting

Syntax `show igmp`
Context *vlan* context.
Usage Show whether IGMP snooping is *enabled* or *disabled*.
Default values Not applicable.
Error messages None defined yet

13.3.21 CPU channel mapping

Syntax `show channel`
Context *VLAN* context.
Usage Show the CPU channel ID this VLAN is mapped to. (See also section 13.1.6.)
Default values Not applicable.
Error messages None defined yet.

13.3.22 Show VLAN status (all VLANs)

Syntax `show vlans`
Context *Admin Exec* context
Usage Show VLAN status information for all VLANs.
Default values Not applicable.
Error messages None defined yet.

Chapter 14

FRNT

The Fast Reconfiguration of Network Topology (FRNT) protocol handles fast reconfiguration in switched ring topologies. When rapid convergence in case of link or switch failure is required, FRNT becomes the protocol of choice when it comes to layer-2 resilience and robustness.

In addition to FRNT, WeOS supports the standard RSTP protocol. Management of RSTP is described in chapter 15.

14.1 Overview of the FRNT protocol and its features

The table below summarises FRNT features available via the the Web and CLI interfaces. A general description of the FRNT protocol and its features are presented in sections 14.1.1 and 14.2. If you are only interested in knowing how to manage the FRNT features via the Web or CLI, please visit sections 14.3 or 14.4 directly.

Feature	Web (Sec. 14.3)	CLI (Sec. 14.4)	General Description
Enable FRNT	X	X	Sec. 14.1.1
Set FRNT mode (focal-point or member switch)	X	X	”
Set FRNT ring ports	X	X	”

14.1.1 FRNT introduction

The FRNT protocol handles fast reconfiguration in switched ring topologies. One of the switches has the role of FRNT *focal point* while the other switches are referred to as FRNT members. When the switches are connected in a ring, it is the responsibility of the focal point to break the loop by putting one of its ports (*port 1*) in *blocking* mode, see fig. 14.1.

Note: *In an FRNT ring, only one of the switches can be configured as focal point. The other switches should be configured as member switches (i.e., non-"focal-point").*

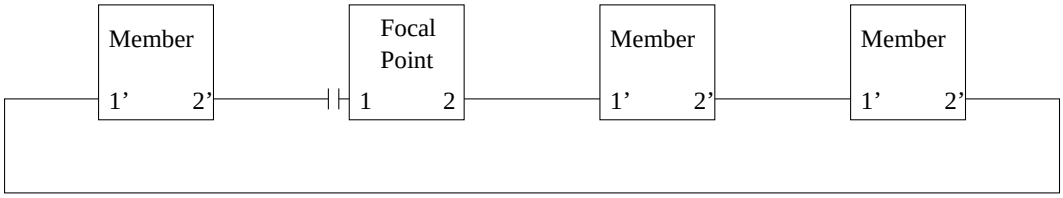


Figure 14.1: FRNT network operating in *ring mode*. Port1 on the Focal Point is in BLOCKING state.

Once a link failure is detected somewhere along the ring, the focal point will put its blocked port (*port 1*) in *forwarding* mode to establish full connectivity between the switches (see fig. 14.2). FRNT is *event based*: switches detecting a *link down* event will immediately send a *link down* FRNT message towards the focal point. Intermediate switches will forward the FRNT messages with highest priority, and the focal point will open its BLOCKED port (port1) upon receiving the *link down* message.

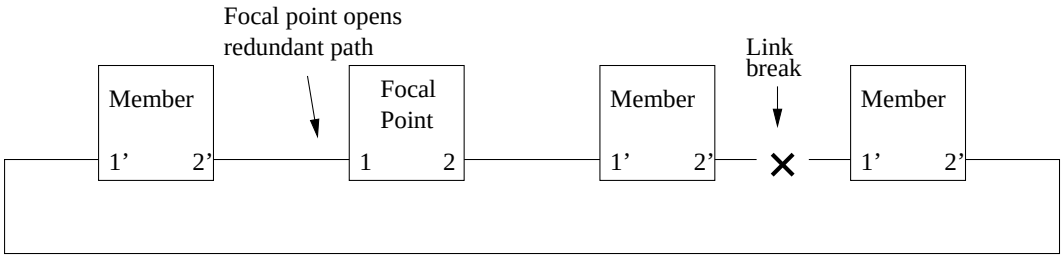


Figure 14.2: FRNT network operating in *bus mode* due to broken link.

Similarly, when a broken link comes back up again and the ring is fully connected, the focal point will react and put its *port 1* back to blocking state.

14.1.2 Guidelines when selecing FRNT ports

When enabling FRNT on a switch, you need to select two ports to use as FRNT ports – FRNT *port 1* and FRNT *port 2*. Below are some recommendations and rules when selecting and configuring the FRNT ports.

- *Fixed speed, full duplex:* When using Ethernet ports as FRNT ports, fixed speed (and full duplex) is recommended over *autonegotiation* of speed and duplex mode on the FRNT ports. Avoid using 10 Mbit/s speed.
- *Connection order:* When connecting switches, it is recommended to connect ”FRNT port 2” on one switch to ”FRNT port 1” on the next switch, and so on (. . . [1 2]⇔[1 2]⇔[1 2] . . .).
- *Avoid using copper SFPs as FRNT ports:* When using Ethernet ports as FRNT ports, choose fixed Ethernet ports or fiber SFPs. Copper SFPs may be used as FRNT ports, but will generally imply non-negligable degradation of fail-over performance.
- *Same slot:* On products with slotted architectures (RedFox Industrial and Wolverine DDW-225/226) both FRNT ports must reside within the same slot. This rule is enforced by WeOS, thus such misconfigurations should not be possible.

14.2 FRNT and RSTP coexistence

With WeOS it is possible to run FRNT and RSTP on the same switch. Fig. 14.3 shows an example of such a configuration, where two of the switches in the FRNT ring (thick lines) are running RSTP on the ”non-FRNT” ports.

As both RSTP and FRNT want to control a port’s state (FORWARDING/BLOCKING), only one of the protocols may be activated on each port to avoid protocol conflicts. Therefore, if both FRNT and RSTP are configured to operate on a certain port, FRNT will have precedence to control the port’s state.

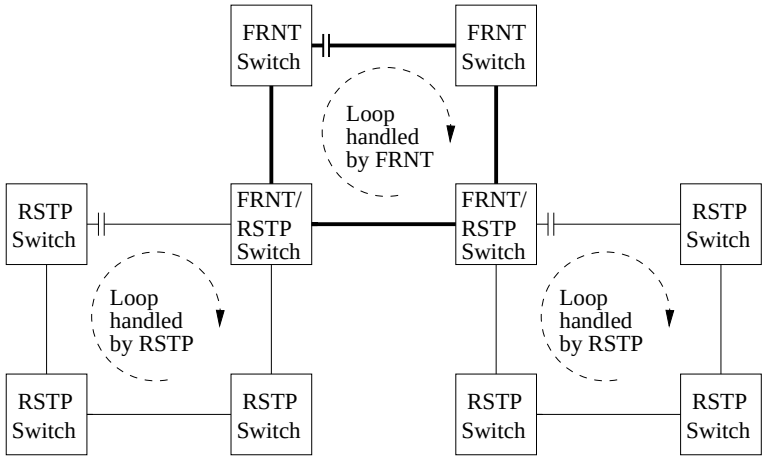


Figure 14.3: Example of coexistence of FRNT and RSTP.

Warning: *FRNT and RSTP are each able to handle loops within their respective domains, however, if a physical loop is created including some links controlled by RSTP and others by FRNT, a broadcast storm is likely to occur, since neither RSTP or FRNT is able to discover the loop, see fig. 14.4. Thus, if RSTP and FRNT is mixed in the same layer-2 network, the operator must ensure that loops across RSTP and FRNT links never occur.*

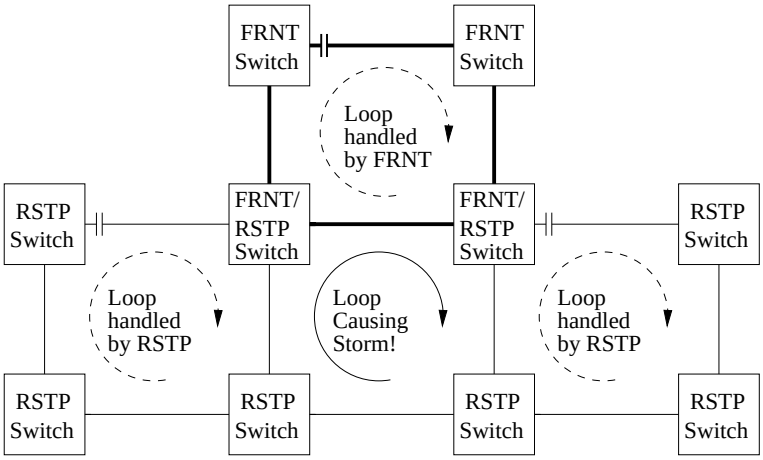
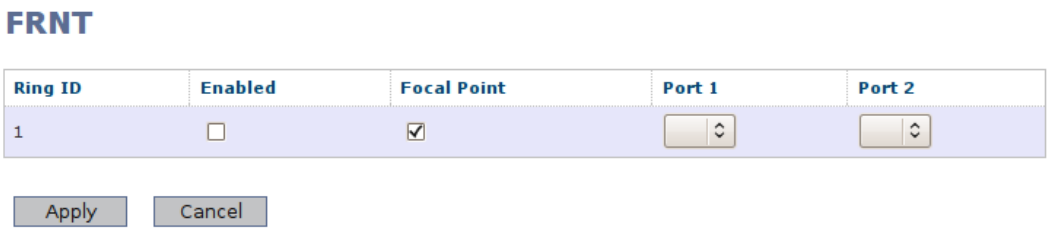


Figure 14.4: Example of loop spanning FRNT and RSTP links - a broadcast storm is likely to occur.

14.3 Managing FRNT settings via the web interface

Menu path: Configuration ⇒ FRNT

On the FRNT configuration page you will be presented to the current settings for FRNT on your switch, see below. You may change the settings by editing the page.



Ring ID	A unique identifier for the FRNT-ring. Currently only one ring is available.
Enabled	Checkbox checked if the FRNT protocol is enabled. Check/uncheck box and apply changes to enable/disable FRNT.
Focal Point	The focal point is the unit in the ring which is responsible for making decisions on topology change. Check this box if this unit should take the role as focal point in the FRNT ring. If not checked, the unit will act as a *member* unit.
Port 1/Port 2	FRNT requires two ports to be assigned FRNT-ports. These are connected to peer units participating in the FRNT ring. Select the two ports connected to other units in the FRNT ring. **Note1:** In slotted architectures (RedFox Industrial and DDW-225), the selected FRNT ports should be located in the same slot. Similar restrictions apply to RedFox Rail. **Note2:** Ports with copper SFPs should not be used as FRNT ports, due to slow link down indication on copper SFPs.

14.4 Managing FRNT settings via the CLI

Command	Default	Section
Configure FRNT settings		
[no] frnt [<ID>]	disabled	Section 14.4.1
[no] focal-point	focal-point	Section 14.4.2
ring-ports <PORT1,PORT2>	N/A	Section 14.4.3
Show FRNT settings		
show frnt [<ID>]	N/A	Section 14.4.4
frnt		
show focal-point	N/A	Section 14.4.5
show ring-ports	N/A	Section 14.4.6

14.4.1 Managing FRNT

Syntax `[no] frnt [<ID>]`

Context *Global Configuration* context

Usage Enter FRNT context of the given FRNT instance ID. Currently only a single FRNT instance is supported, thus the value of the FRNT ID is ignored.

The FRNT instance is only activated upon the selection of valid FRNT ring ports, see section 14.4.3.

Use **”no frnt [ID]”** to remove an existing FRNT instance.

Default values Default ID is 1

Error messages None defined yet.

14.4.2 FRNT focal point and member switch

Syntax `[no] focal-point`

Context *FRNT* context

Usage Configure device to act as FRNT focal point for this FRNT instance. Use **”[no] focal-point”** to configure the device to act as an FRNT member switch.

Default values focal-point

Error messages None defined yet

14.4.3 FRNT Ring Ports

Syntax `ring-ports <PORT1,PORT2>`

Context *FRNT* context

Usage For each FRNT instance, there are two FRNT ports named Port1 and Port2. On a member switch Port1 and Port2 have similar roles, however, on a focal point their roles differ - when the ring is fully connected the focal point will put its Port1 in BLOCKING state.

Note: *In slotted architectures (RedFox Industrial and DDW-225), the selected FRNT ports should be located in the same slot. Similar restrictions apply to RedFox Rail.*

Note: *Ports with copper SFPs should not be used as FRNT ports, due to slow link down indication on copper SFPs.*

Default values None defined

Error messages None defined yet

14.4.4 Show FRNT information

Syntax `show frnt [<ID>]`

Context *Global Configuration* context. Also available as **"show"** command within the FRNT context.

Usage Show FRNT configuration and status information of the given FRNT instance ID.

Default values Currently only a single FRNT instance is supported. Thus, the FRNT instance ID is ignored.

Error messages None defined yet.

14.4.5 Show FRNT focal-point/member setting

Syntax `show focal-point`

Context *frnt* context.

Usage Show whether the switch is configured as FRNT *focal-point* or *member* node (for this FRNT instance).

Default values Not applicable.

Error messages None defined yet.

14.4.6 Show FRNT ports

Syntax `show ring-ports`

Context *frnt* context.

Usage Show which ports are configures as *Port1* and *Port2* (the command gives information about both ports).

Default values Not applicable.

Error messages None defined yet.

Chapter 15

Spanning Tree Protocol - RSTP and STP

The spanning tree protocol (STP) and its successor rapid spanning tree protocol (RSTP) are the standard protocols to support redundancy while avoiding broadcast storms in switched networks. WeOS supports RSTP with fall-back to STP when connecting the switch to another device only capable of STP.

STP/RSTP does not provide the same convergence performance as FRNT, however, STP/RSTP can handle arbitrary switched topologies, while FRNT operates in a *ring* structure. For information on FRNT, and coexistence between FRNT and RSTP, see chapter 14 .

RSTP is enabled on all (Ethernet) ports at factory default.

15.1 Overview of RSTP/STP features

Table 15.1 provides a summary of available RSTP/STP features in WeOS. Further descriptions of the spanning tree protocol and the available features are provided in sections 15.1.1-15.1.3.

15.1.1 Spanning Tree Introduction

Loops in switched networks are dangerous, since packets can loop around forever and jam the network - as opposed to IP and routed networks, Ethernet frames do not include a *hop count* by which the switches could decide to drop a packet circulating around. Since a switched network may contain multiple loops, broadcast

Feature	Web (Sec. 15.2)	CLI (Sec. 15.3)	General Description
Enable STP	X	X	
Bridge priority	X	X	Section 15.1.2
Max age	X	X	Section 15.1.1
Hello time	X	X	Section 15.1.1
Forward delay	X	X	Section 15.1.1
View general RSTP/STP settings	X	X	
Per Port settings			
Enable STP		X	
Admin Edge	X	X	Section 15.1.1
Path Cost		X	Section 15.1.3
View per port RSTP/STP settings	X	X	
View RSTP/STP status	X	X	

Table 15.1: Summary of RSTP/STP features.

packets (or other packets flooded by the switches), leads to packet proliferation; this situation is generally referred to as a *broadcast storm*. On the other hand, loops in switched networks are desirable from a redundancy perspective.

Note: *The purpose of the spanning tree protocol is to ensure that an arbitrary* physical *LAN topology is turned into a* logical *tree topology (i.e., loop free) in such a way that all links in the network are still connected (i.e., a* spanning tree*). This is accomplished by having the switches put some of their ports in* blocking *state.*

Since loops in switched networks are so dangerous, layer-2 redundancy protocols such as STP and RSTP are very restrictive before putting a link in *forwarding* state. The main difference between STP and RSTP is that RSTP is able to react quicker to topology changes, thus can open an alternative path if a link in the active tree is broken, i.e., RSTP has shorter *convergence time* than STP. (FRNT has even faster convergence, see chapter 14.)

In RSTP/STP terminology, a switch is referred to as a *bridge*. Spanning tree is a *plug-and-play* protocol - bridges can use RSTP/STP to form a tree without need for any configuration. However, the protocol provides a set of parameters which

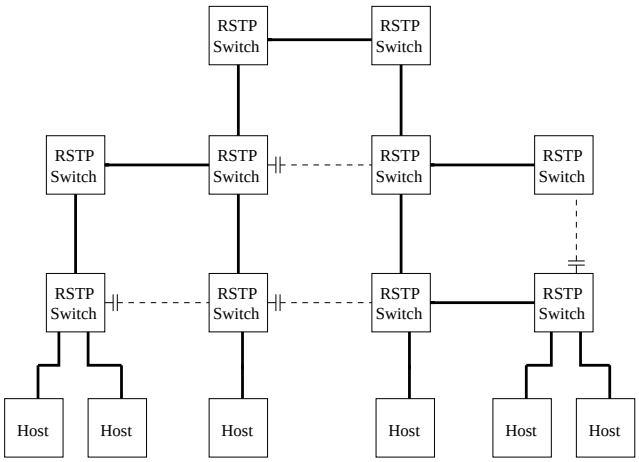


Figure 15.1: Example of RSTP creating a spanning tree. Dashed links have logically been ”cut off” from the active topology by RSTP, eliminating the loops.

the operator can use to fine-tune the network setup. Below is a list of those parameters of specific interest for the WeOS RSTP/STP implementation:

- *Bridge priority*: Used for *root bridge* and *designated bridge* election. See section 15.1.2.
- *Port/Path cost*: Each port is assigned a ”cost”. This is used by each bridge to find the *least cost* path to the *root bridge* as part of the tree establishment. See section 15.1.3.
- *Max age/Hello time*: Used to detect that a STP/RSTP neighbour is down. The *max age* also puts a protocol limit to the *size* of the network[1].
- *Forward Delay*: Used when operating in STP mode (i.e., not RSTP). Defines the time period by which the protocol can be sure that STP information on a topology change has propagated from one side of the network to the other. The STP convergence time is limited by twice the forwarding delay (plus the time it takes to detect the topology change).
- *Admin Edge:* Ports where only end nodes connect are referred to as *edge ports*. If a port is only used for connecting hosts (i.e., no risk for loops), it can be configured as an *admin edge* port.

[1] In RSTP the *Message Age* field in the *Hello Messages* effectively acts as a hop count, counting the distance from the Root. If the *Message Age* exceeds the *Max Age* the packet is dropped. Thus, the setting of the *Max Age* parameter restricts the size of the RSTP LAN.

Access ports and inter-switch ports: *It is recommended that all "inter-switch ports" (ports connecting switches) are configured as "non-edge ports" (admin edge disabled), and that all "access ports" (ports where hosts connect) are configured as "edge ports" (admin edge enabled).*

When configured as *admin edge* the port will:

- be put in *FORWARDING* state quickly after system boot, and
- be kept in *FORWARDING* state during periods when the spanning tree topology is changing.

An *admin edge* assumes the port leads to a host or a router (i.e., not another bridge), and the port is therefore put in *FORWARDING* state without first verifying that the LAN is still loop free. The bridge will still send *Hello Messages* on *admin edge* ports, and will react on any incoming *Hello Messages* as it would on regular (non-"admin edge") ports. Thus, even if loops may occur via an *admin edge* port, the bridge will generally be able to receive the high-priority RSTP messages, and cut the loop by putting the appropriate port in *BLOCKING*.

Important information on the default setting: *To limit the risk for forwarding loops when putting a new unit into the network, and still keep reasonable performance in case there are no loops, the following default settings have been chosen with respect to RSTP:*

- *Spanning Tree is enabled on all ports: This gives protection in case a loop within the LAN infrastructure unintentionally occurs.*
- *All ports are configured as "admin edge": Thereby annoying delays are avoided to get a port in FORWARDING state upon system startup. In networks designed to have loops for redundancy purposes (or when the probability of unintentional loops within the LAN cannot be ignored), the network operator* **should disable admin edge** *on all inter-switch ports (ports connecting switches).*

The IEEE std 802.1D-2004 specifies restrictions on the *Max age* parameter with respect to the *Hello time* and the *Forward delay* as shown below. This affects how these parameters can be configured.

- $Max\ age \geq 2 * (Hello\ time + 1)$
- $Max\ age \leq 2 * (Forward\ Delay - 1)$

Note: *Some of the RSTP/STP parameters (Max age, Hello time, and Forward Delay) need to be set consistently throughout all bridges with the LAN infrastructure. Therefore, bridges inherit these parameter values from the current* root bridge*, irrespective of the corresponding parameter setting in the bridge itself.*

15.1.2 Bridge Identity

Each bridge is assigned an 8 byte bridge identifier (bridge ID) as shown in fig. 15.2.

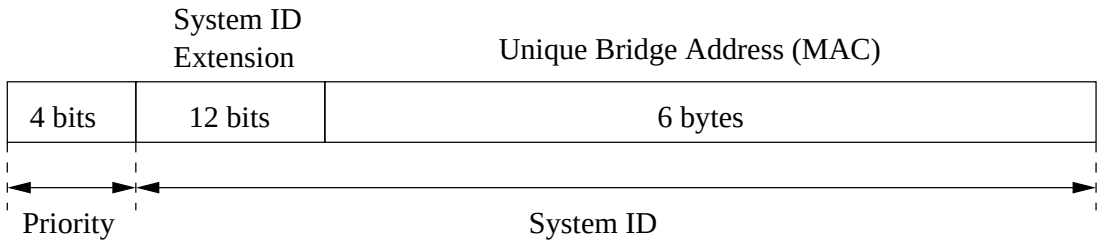


Figure 15.2: Structure of bridge ID.

The bridge ID is divided into a *priority* part (4 bits) and a *system ID* (60 bits). The bridge with the lowest bridge ID within the LAN will become the root bridge, i.e., lower *priority* means greater chance to become root bridge. The bridge ID is also used to select a *designated bridge* on a link, when multiple bridges on the link have the same ”least cost path” to the root bridge.

The format of the bridge ID follows IEEE std. 802.1D-2004 (RSTP). It differs from the structure specified in IEEE std. 802.1D-1998 (STP), where the *priority* field was 2 bytes and the *system ID* field was 6 bytes. The change in structure was made with respect to the multiple spanning tree protocol (MSTP) defined in IEEE std. 802.1Q-2005 (WeOS currently does not support MSTP).

- *Priority (4 bits)*: Can take values in range 0-15, where 8 is default. 0 (zero) means highest priority and 15 lowest priority. Compared to the ”old” 2 byte priority field of STP, this is rather a *priority factor* field, which can be multiplied by *4096* to get the ”old” STP priority.
- *System ID Extension (12 bits)*: Set to all zeros in WeOS.
- *Unique Bridge Address*: Tie-breaker ensuring the bridge ID will be unique. WeOS uses the *base MAC address* assigned to the switch for this field.

15.1.3 Path Cost

Each port is associated with a cost referred to as a *path cost*. Low-speed links are generally given a high cost, which increases the probability of the port ending up in *blocking* state (and vice versa), in case spanning tree discovers a loop.

By default, the path cost of a port is assigned dynamically with values related to the port speed (in-line with the recommendations of IEEE std 802.1D-2004). The same path costs are used irrespective if the port is operating in RSTP or STP mode.

Port Speed (Mbit/s)	RSTP path cost
10	2000000
100	200000
1000	20000

It is also possible to configure the path cost manually. That may be useful to get more fine grain control of which port in the LAN should be put in *blocking* state. Setting path costs manually may be desirable when operating a LAN including a mix of RSTP and STP capable, since STP uses a different set of default path costs.

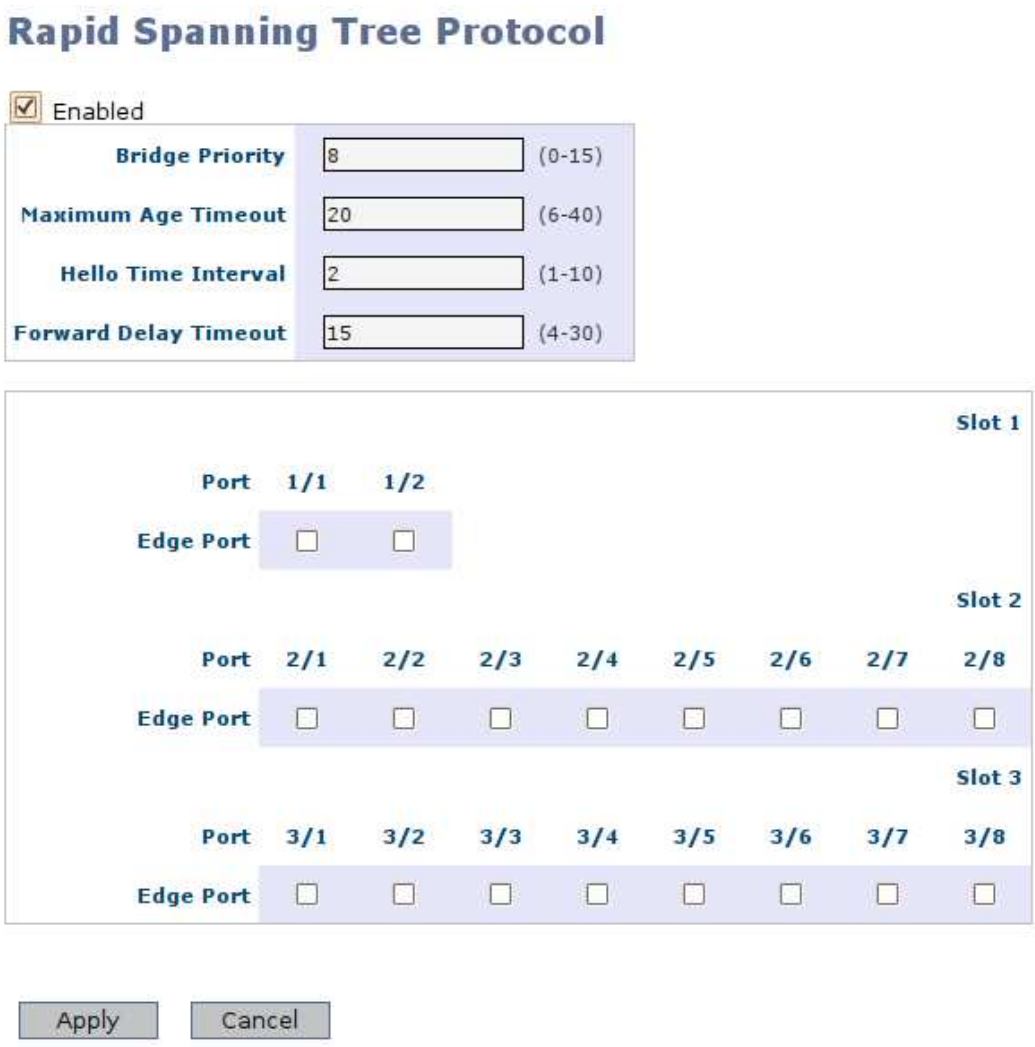
15.1.4 RSTP and STP coexistence

WeOS supports both RSTP and STP, but WeOS always attempts to run RSTP on every spanning-tree enabled port. WeOS automatically shifts to STP mode on a port, if it detects a bridge running STP on that port. Other ports continue operating in RSTP mode. When operating a network including a mix of RSTP and STP bridges, it may be necessary to configure path costs manually to get the intended spanning tree behaviour, see also section 15.1.3.

15.2 Managing RSTP via the web interface

Menu path: Configuration ⇒ RSTP

On the RSTP configuration page you will be presented to the current settings for RSTP on your switch, see below. You may change the settings by editing the page.



Enabled	Check the box to enable RSTP. If you have a JavaScript[1] enabled browser the other settings will not be displayed unless you check this box.
Bridge Priority	A priority level used in root bridge selection. A lower value increases the probability for this switch to be elected as root bridge.
Maximum Age Timeout	The time the unit will wait before considering a neighbour designated bridge is down after the last Hello message was heard from the neighbour.
Hello Time Interval	The time between two consecutive transmissions of hello messages.
Forward Delay Timeout	The time an interface takes to change from blocking to forwarding state. Only used when operating in STP mode.
Edge Port	Ports connected to end hosts and routers (i.e., not to another switch) can be set as admin-edge ports. This avoids unnecessary BLOCKING of such ports at system startup or when a topology change occurs. It is *recommended* that this box is checked for every port where it is certain that only end hosts and routers connect. Port which (may) connect to another switch should un-check this box.

[1] JavaScript is a trademark of Sun Microsystems.

15.3 Managing RSTP via the CLI

Command	Default	Section
[no] spanning-tree	Enabled	Section 15.3.1
priority <0-15|0-65536>	8 (32768)	Section 15.3.2
max-age-time <6-40>	20	Section 15.3.3
hello-time <1-10>	2	Section 15.3.4
forward-delay <4-30>	15	Section 15.3.5
show		Section 15.3.6
show priority		Section 15.3.7
show max-age-time		Section 15.3.8
show hello-time		Section 15.3.9
show forward-delay		Section 15.3.10
stp-port <PORTLIST|all>		Section 15.3.11
[no] enable	Enabled	Section 15.3.12
[no] admin-edge	Enabled	Section 15.3.13
[no] path-cost <0-20000000>	0 (Auto)	Section 15.3.14
show		Section 15.3.15
show spanning-tree		Section 15.3.16

15.3.1 Manage RSTP

Syntax `[no] spanning-tree`

Context *Global Configuration* context

Usage Enter spanning-tree configuration context, and activate spanning-tree (if not already activated). Use **"`no spanning-tree`"** to disable spanning-tree and to remove spanning-tree configurations.

Default values Enabled

Error messages None defined yet.

15.3.2 Bridge Priority Setting

Syntax `priority <0-15|0-65535>`

Context *spanning-tree* context

Usage Set bridge priority, where a low value means high priority, which increase the probability of being elected as *root bridge*. Values can be entered in

two ways, either in range 0-15, which corresponds to the 4-bit priority field specified in IEEE std 802.1D-2004, or in range 16-65535 which corresponds to the traditional 2 byte priority field defined in IEEE 802.1D-1998. In the latter case, the value is divided by 4096, and stored as a value 0-15.

See section 15.1.2 for more information.

Default values 8 (32768)

Error messages None defined yet.

15.3.3 Max Age Setting

Syntax `max-age-time <6-40>`

Context *spanning-tree* context

Usage Set spanning-tree max age timeout. Since bridges use the max age configured at the root bridge, this parameter setting only matters if this bridge becomes the root bridge.

Default values 20

Error messages An error message is given if the **"max-age-time"** is not given a valid value with respect to **"hello-time"** or **"forward-delay"**, see section 15.1.1.

15.3.4 Hello Interval

Syntax `hello-time <1-10>`

Context *spanning-tree* context

Usage Set spanning-tree hello time interval. Since bridges use the hello time configured at the root bridge, this parameter setting only matters if this bridge becomes the root bridge.

Default values 2

Error messages An error message is given if the **"hello-time"** is not given a valid value with respect to **"max-age-time"**, see section 15.1.1.

15.3.5 Forward Delay

Syntax `forward-delay <4-30>`

Context *spanning-tree* context

Usage Set spanning-tree forward delay. Since bridges use the forward delay configured at the root bridge, this parameter setting only matters if this bridge becomes the root bridge.

Default values 15

Error messages An error message is given if the **"forward-delay"** is not given a valid value with respect to **"max-age-time"**, see section 15.1.1.

15.3.6 Show General RSTP Settings

Syntax `show spanning-tree`

Context *Global Configuration* context. Also available as **"show"** command within the spanning-tree context.

Usage Show general spanning tree parameter settings, given that spanning-tree is enabled.

Default values Not applicable.

Error messages None defined yet.

15.3.7 Show Bridge Priority Setting

Syntax `show priority`

Context *spanning-tree* context

Usage Show bridge priority setting.

Default values Not applicable.

Error messages None defined yet.

15.3.8 Show Max Age Setting

Syntax `show max-age-time`

Context *spanning-tree* context

Usage Show max age timeout setting.

Default values Not applicable.

Error messages None defined yet.

15.3.9 Show Hello Interval Setting

Syntax `show hello-time`

Context *spanning-tree* context

Usage Show hello interval setting.

Default values Not applicable.

Error messages None defined yet.

15.3.10 Show Forwarding Delay Setting

Syntax `show forward-delay`

Context *spanning-tree* context

Usage Show bridge forward delay setting.

Default values Not applicable.

Error messages None defined yet.

15.3.11 Manage RSTP Ports

Syntax `stp-port <PORTLIST|all>`

Context *spanning-tree* context

Usage Manage per port spanning-tree settings for one or more ports.

Default values Not applicable.

Error messages None defined yet.

15.3.12 Enable Spanning Tree on a Port

Syntax `[no] enable`

Context *stp-port* context

Usage Enable the spanning tree protocol on a port. Use **”no enable”** to disable spanning tree protocol on a port.

Default values Enabled

Error messages None defined yet.

15.3.13 Admin Edge Setting

Syntax `[no] admin-edge`

Context *stp-port* context

Usage Configure the port as an *edge* port. Use **"no admin-edge"** to configure the port as a regular spanning tree port.

It is *recommended* that every port where it is certain that only end hosts and routers connect are configured as **"admin-edge"**. Port which (may) connect to another switch should be configured as **"no admin-edge"**.

Default values Enabled (**"admin-edge"**)

Error messages None defined yet.

15.3.14 Path Cost Setting

Syntax `[no] path-cost <0-20000000>`

Context *stp-port* context

Usage Configure the spanning tree path cost for a port. A low speed link should get a higher cost, a high speed link a lower cost. Use **"path-cost 0"** (or **"no path-cost"**) to have the path-cost assigned automatically depending on the port speed (see section 15.1.3).

Values in range 1-20000000 means a statically configured path cost of the given value.

Default values Automatic (**"path-cost 0"**)

Error messages None defined yet.

15.3.15 Show Spanning Tree Port Settings

Syntax `show stp-port [PORTLIST]`

Context *spanning-tree* context. Also available as **"show"** command within the *stp-port* context.

Usage Show per port spanning-tree parameter settings.

Default values If no port is specified, settings for all ports are shown.

Error messages None defined yet.

15.3.16 Show RSTP Status

Syntax `show spanning-tree`

Context *Admin Exec* context.

Usage Show spanning-tree status information, including current port states, root bridge ID, etc..

Default values Not applicable.

Error messages None defined yet.

Chapter 16

Link Aggregation

Note: *As of WeOS version v4.3.0, the link aggregation contains several limitations as described in the following sections.*

16.1 Overview of Link Aggregation Support in WeOS

With link aggregation it is possible to bundle multiple Ethernet links together as shown in fig. 16.1. Traffic is *load balanced* over the different member links, thus use of link aggregation enables an operator to increase the capacity between two switches. The load balancing is, however, not perfect, since packets with identical *source and destination MAC address pairs* will be forwarded through the same link.

WeOS supports basic link aggregation inline with IEEE 802.3ad. However, the current support for link aggregation contains several limitations such as:

- Aggregation control: Link aggregates can be configured statically or be managed dynamically via the Westermo FLHP protocol. LACP is currently **not** supported.
- VLAN support: There is no support to add a link aggregate to a VLAN. Instead, each of the individual member links need to be added to the appropriate VLANs.
- Port settings: There is no support to configure port settings for the link aggregate. Instead, each of the individual member ports need to be configured uniformly, e.g., with respect to port speed/duplex mode.
- Layer-2 protocols: Layer-2 redundancy protocols such as FRNT or RSTP cannot be used on a link aggregate or any of its member ports. Neither can

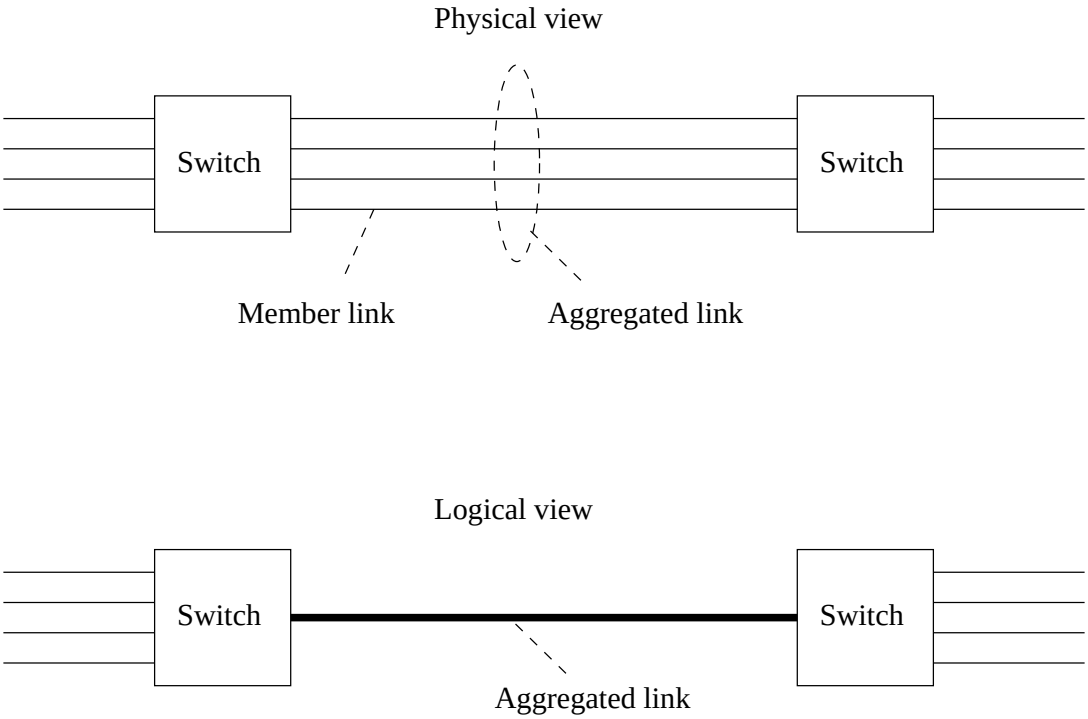


Figure 16.1: Example of link aggregation with four member links

IGMP snooping, thus VLANs where any link aggregate has a member port must have IGMP disabled.

Warning: *When configuring link aggregation on switches in an operational network, there is a potential risk for a broadcast storm to occur. WeOS currently does not support the use of RSTP or FRNT on aggregated ports. The operator must therefore ensure that no layer-2 forwarding loop is established when connecting switches via aggregated links.*

These are the recommended steps to configure link aggregation.

- It is strongly recommended that the switch is disconnected from the network while configuring link aggregation to avoid causing a broadcast storm.
- Decide which ports should be members of the link aggregate.
- Ensure that each of these ports have identical port settings, e.g., with respect to speed/duplex mode.

- Ensure that neither FRNT nor RSTP is running on any of these member ports.
- Ensure that all member ports are associated to all desired VLANs in identical association modes, i.e., for each desired VLAN, ensure that all member ports are associated tagged or that all ports are associated untagged.
- Ensure that IGMP snooping is disabled on those VLANs.
- Create the link aggregate. Add the desired ports to the member port set. We recommend the use of FLHP to handle the link aggregate.

By using FLHP (as opposed to using the static mode) the switches will be able handle link up and down events appropriately. In case a link goes down, the traffic flows earlier transmitted over that link will be mapped to some of the other member links. Similarly, if a link comes up the *load balancing* feature will ensure that traffic flows will be remapped to the links that are up. Similar functionality would be achieved by using the Link Aggregation Control Protocol (LACP); as mentioned above, LACP is not yet supported in WeOS.

Warning: *As of WeOS version v4.3.0, the use of FLHP for link aggregation control is provided as a technology preview feature. All use of the FLHP link aggregation control feature except for testing is discouraged.*

16.2 Configuring Link Aggregation Settings via the CLI

Command	Default	Section
Configure Link Aggregate		
[no] aggregate <AGGREGATE_ID>	N/A	Section 16.2.1
[no] set <PORTLIST>	N/A	Section 16.2.2
mode <static|flhp>	static	Section 16.2.3
Show Link Aggregate Settings		
show aggregate [AGGREGATE_ID]		Section 16.2.4
aggregate		
show set		Section 16.2.5
show mode		Section 16.2.6

16.2.1 Manage a Link Aggregate

Syntax `[no] aggregate <AGGREGATE_ID>`

Context *Global Configuration* context

Usage Create, modify or remove a link aggregate.
Enter link aggregate context of the given aggregate identifier (A1-A*N*, where N is limited by the number of physical Ethernet ports on the switch.) If this is a new link aggregate, the aggregate is created.
Use **"no aggregate <AGGREGATE_ID>"** to remove an existing link aggregate, or **"no aggregate"** to remove all link aggregates.

Default values When using the **"no aggregate"** form (without providing a specific aggregate ID), all link aggregates are removed.

Error messages None defined yet.

16.2.2 Configure Link Aggregation Member Set

Syntax `[no] set <PORTLIST>`

Context *Aggregate* context

Usage Add/remove a list of ports to/from the port member set of this link aggregate. Use **"no set"** (without providing a port list) to remove all ports from the member set.

Default values When using the **"no set"** form (without providing a specific PORTLIST), all ports are removed.

Error messages None defined yet.

”PORTLIST” is a comma separated list of port ranges without intermediate spaces, e.g., **”X1-X2,X4”**.

16.2.3 Configure Link Aggregate Control Mode

Syntax `mode <static|flhp>`

Context *Aggregate* context

Usage Define whether the link aggregate should be managed dynamically via FLHP (only those member links qualified as up via FLHP will be included in the aggregate) or if a static configuration should be used (all member links are included in the aggregate irrespective if they are up or not). Use of FLHP is recommended when connected to another switch also supporting FLHP.

> **Warning**: *As of WeOS version v4.3.0, the use of FLHP for link aggregation control is provided as a technology preview feature. All use of the FLHP link aggregation control feature except for testing is discouraged.*

Default values `static`

Error messages None defined yet.

16.2.4 Show Link Aggregate Settings

Syntax `show aggregate [AGGREGATE_ID]`

Context *Global Configuration* context. Also available as **”show”** command within the *Aggregate* context.

Usage Show link aggregation configuration for the given aggregate ID (or all link aggregates).

Default values All link aggregates, i.e., if no aggregate ID is provided, information on all configured link aggregates will be shown.

Error messages None defined yet.

16.2.5 Show Link Aggregation Member Set

Syntax `show set`

Context *Aggregate* context.

Usage Show member ports of this link aggregate.

Default values Not applicable.

Error messages None defined yet.

16.2.6 Show Link Aggregate Control Mode

Syntax `show mode`

Context *Aggregate* context.

Usage Show the configured control mode of this link aggregate.

Default values Not applicable.

Error messages None defined yet.

Chapter 17

General Interface and Network Settings

This chapter concerns network interface settings, such as the interface IP address setting, as well as IP settings in common for all interfaces, e.g., the default gateway IP address, DNS server and NTP server settings. There are also interface and network settings specific to various routing protocols and services (RIP, OSPF, VRRP, etc.), and this is left to chapters 18-22.

Section 17.1 describes network interfaces properties in WeOS. It also presents the *default interface* and *management interface* concepts, as well as IP related settings for DNS, NTP, etc. Section 17.2 covers management of general interface and network settings via the Web interface, while the corresponding CLI syntax description is divided into sections 17.3 (interface settings) and 17.4 (other network settings).

17.1 Overview of General Interface and Network Settings

Table 17.1 summarises general interface and network features. Sections 17.1.1-17.1.2.2 contain further information on specific interface and network features.

17.1.1 Network interfaces

A network interface is created for every VLAN configured on the switch, see fig. 17.1. Future versions of WeOS will provide additional types of network interfaces such as PPP interfaces on serial links, but currently only VLAN network

Feature	Web (Sec. 17.2	CLI (Sec. 17.3-Sec. 17.4)	General Description
Interface settings			
Enable/disable iface	X	X	Sec. 17.1.1
MAC address		X	Sec. 17.1.1.2
IP address	X	X	Sec. 17.1.1.3
Netmask (Prefix Length)	X	X	Sec. 17.1.1.3
MTU		X	
Primary interface	X	X	Sec. 17.1.1.4
Management interface	X	X	Sec. 17.1.1.5
View interface configuration	X	X	
View interface status	X	X	
General network settings			
Default gateway	X	X	Sec. 17.1.2.1
Enable/disable unicast routing	X	X	”
DNS client support			
Set DNS server	X	X	Sec. 17.1.2.3
Dynamic DNS	X	X	”
DNS search path		X	”
SNTP (NTP client)	X	X	Sec. 17.1.2.2
View general network config.	X	X	
View general network status	X	X	

Table 17.1: Interface and General Network Settings

interfaces are provided.

Every network interface can be assigned an IP(v4) address and netmask. By assigning an IP address to a VLAN interface, the operator is able to remotely manage the switch via that VLAN. Furthermore, if routing is enabled, the switch is able to *route* packets between this and other network interfaces. Section 17.1.2 gives a brief overview of WeOS routing features (chapter 19 gives a more detailed introduction to WeOS routing support, while chapters 21 and 20 covers dynamic routing with RIP and OSPF respectively).

A VLAN interface has status *up* when it is *enabled* (adminstratively configured *up*), and when its associated VLAN is up. In turn, the VLAN is up when the VLAN

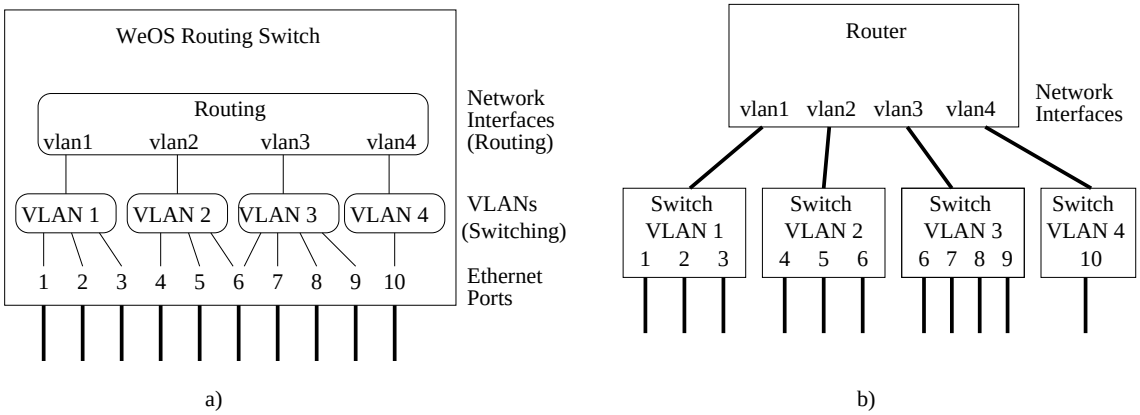


Figure 17.1: A network interface is associated with each VLAN, and VLANs are in turn associated with Ethernet (or DSL) ports as shown in figure a). The routing switch can conceptually be seen as a router connecting a set of switches, as shown in figure b). In this sample setup, port 6 is shared by VLANs 2 and 3 (by use of VLAN tagging).

is *enabled*, and when any of its associated ports have *link up* status.

17.1.1.1 Interface Default Settings

At factory default, all ports on the switch belong to VLAN 1, and the network interface associated with VLAN 1 is named *vlan1*. Thus, *vlan1* is the only network interface present at factory default, and its settings are presented below:

Interface parameters	**Factory Default Setting**
Interface	**vlan1**
Administrative Mode	**Up**
IP address	**192.168.2.200**
Netmask	**255.255.255.0**
MAC address	**Auto**
MTU	**Auto (1500 Bytes)**
Primary Interface	**Enabled**
Management Interface (SSH, HTTP, HTTPS, SNMP, IPConfig)	**Enabled**

The *primary interface* and *management interface* concepts are described in sections 17.1.1.4 and 17.1.1.5.

As shown in fig. 17.1 the switch will have one network interface for every VLAN defined on the switch. Thus, additional network interfaces can be created by creating new VLANs (see chapter 13). New interfaces will have the following settings upon creation:

Interface parameters	**Default Setting**
Interface	**vlan<VID>**
Administrative Mode	**Up**
IP address	**Disabled**
Netmask	**Disabled**
MAC address	**Auto**
MTU	**Auto (1500 Bytes)**
Primary Interface	**Disabled**
Management Interface (SSH, HTTP, HTTPS, SNMP, IPConfig)	**Enabled**

VLAN network interfaces will be named according to the associated VLAN ID, e.g., the interface of VLAN 100 will be named *vlan100*. To communicate with the switch via a newly created VLAN, an IP address has to be assigned.

The *primary interface* and *management interface* concepts are described in sections 17.1.1.4 and 17.1.1.5.

17.1.1.2 Interface MAC address

Each VLAN network interface will be assigned a MAC address (also known as the Ethernet address, the link address, the hardware address, or the IEEE EUI-48 address).

In WeOS products, each *Ethernet port* (or DSL port) is assigned a MAC address, and a *VLAN interface* will byy default inherited its MAC address from one of its member ports. It is also possible to manually configure a MAC address for a VLAN interface.

The algorithm to assign VLAN interface MAC address uses the following preference order:

1. If the interface has been configured with a specific MAC address, use that address as the interface MAC address.
2. If the VLAN has one or more ports assigned *untagged*, use the MAC address of the ”lowest” untagged port as the interface MAC address.
3. If the port has one or more ports assigned *tagged*, use the MAC address of the ”lowest” tagged port as the interface MAC address.

4. Use the MAC address of the *channel* (section 13.1.6) associated with the VLAN.

Consider the sample VLAN configuration in fig. 17.1. Assuming all interfaces get their MAC address automatically, interface *vlan1* inherits the MAC address of port 1, *vlan2* inherits its MAC from port 4, *vlan3* from port 7 (assuming port 6 is tagged on VLAN 3), and interface *vlan4* from port 10.

Note: *For the automatic MAC assignment methods (steps 2-4 above), the MAC address may change when the set of ports associated with the VLAN changes. When this happens, the WeOS device will submit a* gratuitous ARP *to update stale ARP caches in neighbor nodes.*

For VLANs created dynamically (section 13.1.7), no associated network interface is created. Thus, for such VLANs no interface MAC address is needed.

17.1.1.3 IP address settings

An interface can be configured with a static IP address and netmask, or configured to acquire its address dynamically. It is also possible to have an interface without any IP address.

The example below interface vlan1 is assigned a static IP address. In this example, the IP address *netmask* (255.255.255.0) has been written as a *prefix length* ('/24').

```
redfox:/config/#> interface vlan2
redfox:/config/iface-vlan2/#> inet static
redfox:/config/iface-vlan2/#> address 192.168.11.1/24
redfox:/config/iface-vlan2/#> end
redfox:/config/#>
```

When configured for dynamic address assignment, a VLAN network interface will attempt to get its IP address from a DHCP server. If no DHCP server is present, the interface will generally end up without any IP address. The exception is the *primary interface*, which will acquire a *link-local* IP address in absence of DHCP servers. The *primary interface* and *link-local addresses* concepts are further described in section 17.1.1.4.

17.1.1.4 Dynamic Address Assignment and Primary Interface

An interface can be configured to get its IP settings dynamically via DHCP. In addition to interface settings such as IP address and netmask, the switch can acquire general network settings such as default gateway and DNS server(s) from

the DHCP server. (More information on general network settings is given in section 17.1.2.)

Since multiple network interfaces can use DHCP to acquire its IP settings, there is a need for rules regarding which interface can update the general network settings (default gateway, etc.). The interface allowed to affect these general IP settings is in WeOS called the *primary interface*.

- Only the *primary interface* can use the parameters acquired via DHCP to set the general IP settings such as default gateway, etc..
- There can at most be one *primary interface* defined at a time. Configuring one interface to become *primary* implies the interface previously defined as *primary* will loose that property. It is possible to disable the primary interface option entirely.
- Static configuration of general IP settings has precedence over configuration acquired dynamically. That is, if for example the default gateway is set to 192.168.0.1, that will be the default gateway in use even if another gateway is learnt via DHCP on the primary interface.

  Regarding *name server* and *domain* configuration settings, they may be acquired from a DHCP server when no *name server* has been configured statically. However, configuring a domain search path does not prohibit getting name server and domain via DHCP.
- Interfaces not defined as primary interface *only* acquire their IP address and netmask via DHCP.

In the example below interface vlan3 is configured to acquire it IP address via DHCP. As vlan3 is configured as *primary interface*, it is also able to acquire default gateway, DNS server(s) and related settings via DHCP.

```
redfox:/config/#>
redfox:/config/#>  interface vlan3
redfox:/config/iface-vlan3/#>  inet dhcp
redfox:/config/iface-vlan3/#>  primary
Moved primary interface from vlan1 to vlan3, this operation cannot be undone.
redfox:/config/iface-vlan3/#>  end
redfox:/config/#>
```

If no DHCP server is present, an interface configured to use DHCP for address assignment will end up without any IP address. The exception is the *primary interface*; if the primary interface is configured to use DHCP, it will fall-back to use a *link-local* IP address if it fails to get an address via DHCP. Link-local address are taken from the 169.254.0.0/16 range in such a manner that

- address collisions are avoided,
- an interface is likely to get the same address every time it comes up.

17.1.1.5 Management Interface

The operator can manage the switch remotely in several ways: Web (HTTP/HTTPS), SSH, SNMP and IPConfig. As described in chapter 7 it is possible to completely disable individual management services, however, there are situations when an operator may wish to limit management access to a certain network interface or VLAN. WeOS provides a powerful mechanism for controlling access to management services on a *per interface basis*. An interface where one or more management services are enabled is referred to as a *management interface*.

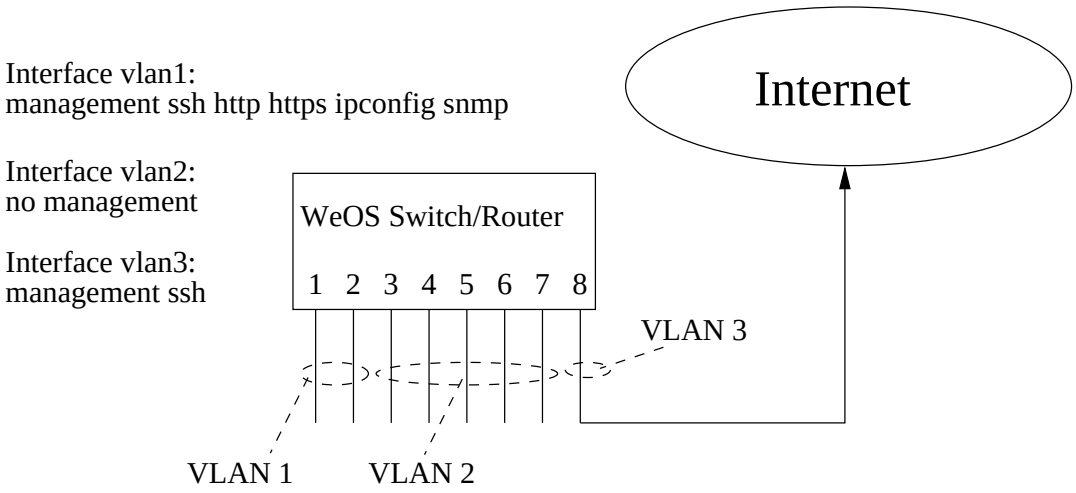


Figure 17.2: Enabling/disabling management services per interface.

Fig. 17.2 gives an example on the flexibility by the *management interface* feature in WeOS. The switch has three network interfaces - one for each VLAN. VLAN 1 is the administrator's local LAN with full management capabilities. VLAN 2 is another local LAN for regular *in-house* users, from which no management is allowed. VLAN 3 is used for the upstream connection to the Internet; in this example SSH is allowed on this network interface, while other services are disabled.

Note: *WeOS use the term " management interface" rather than "management VLAN". This is because management should not be limited to*

VLAN network interfaces. For example, the operator may wish to manage a switch remotely through a modem connection (i.e., a PPP interface on a switch equipped with a serial port).

An operator could create the equivalent of a management VLAN by disabling management on all interfaces but the network interfaces associated with that VLAN.

Sections 17.1.1.3 and 17.1.1.1 describe the network interface default settings (settings at factory default and settings for newly created interfaces). Regarding the management interface capabilities, all management services (HTTP/HTTPS, SSH, SNMP and IPConfig) are enabled both for the interface available at factory default (the *vlan1* network interface), *and* for all newly created interfaces. The default behaviour aims to avoid unintentional loss of management access to the switch.

Warning: *Enabling management services on all interfaces is convenient, but may pose a security threat if connected to an untrusted network. As the switch by default is manageable via all network interfaces, the operator must ensure to disable management services (totally or for specific management services) on interfaces connected to untrusted networks. For an interface connected to the public Internet one should consider disabling all management services, or perhaps only allow management via, e.g, SSH and HTTPS. Configuring adequately secure passwords is also crucial when providing management access via an interface connected to an untrusted/public network.*

When it comes to disabling of management services, a word of caution may be in order. The ability to select management services per interface is actually yet another way of getting locked out from the system. For systems equipped with a console port this may not be a problem, for others this is the time to be reminded about the ”crossed–cables factory reset” (section 7.1.4.3).

However, WeOS actually does implement some safeguards to prevent against locking yourself out. If all management is disabled on all interfaces, the system falls back to enabling secure shell, SSH, access on interface vlan1. Furthermore, if *Web* (for instance) is the only management service enabled on any interface, but the Web server has been entirely disabled, the same fall-back solution is triggered.

Due to the special role of interface vlan1, it is, from a security standpoint, recommended to separate the primary interface from the management interface. The primary interface is usually set on the external side of a WAN–LAN setup to ensure that default gateways or DNS servers received from a DHCP server are

set. Westermo recommends setting up, e.g., vlan2 as the system primary and external interface.

17.1.2 General IP settings

The general IP settings provided fall into three categories:

- Routing: Configuration of default gateway, static IP routes, and ability to enable/disable IP routing.
- IGMP: Configuration of IGMP snooping parameters such as *querier mode*, *query interval* and static multicast router ports.
- Services: Examples of include settings for DNS and DDNS servers, domain search path, and SNTP client settings.

17.1.2.1 Routing

To manage the switch remotely, it should generally be configured with a default gateway. It is also possible to configure additional, static IP routes.

The switch is capable of *IP forwarding*, i.e., it can *route* incoming IP packets to other interfaces and IP subnets. Both static routing and dynamic routing (RIP and OSPF) are supported. The switch acts as a router by default, i.e., IP forwarding is *enabled* in the factory default setting.

Currently, the switch is able to route *unicast* IP packets, but is unable to route IP multicast. However, WeOS devices can efficiently distribute IP multicast packets in a switched LAN by use of IGMP snooping.

This chapter only covers rudimentary routing features, such as enabling/disabling IP forwarding and configuring a default gateway. IGMP snooping is covered in chapter 18, while WeOS routing support is described in chapters 19-20.

17.1.2.2 Time synchronisation via NTP Server

The switch can synchronise its clock with an external time server via the SNTP protocol. A single SNTP server address can be configured. Time synchronisation will not be activated until a SNTP server address is configured.

17.1.2.3 DNS and dynamic DNS

For most users it is easier to refer to Internet hosts using *domain names* (e.g., *www.westermo.se*) than using IP addresses (e.g., *85.24.138.221*). To facilitate use of the Domain Name System (DNS), WeOS supports configuration of up to two DNS server entries. It is also possible to configure a *domain search path*. DNS server and domain search path settings can also be acquired dynamically via DHCP (see section 17.1.1.4).

Use of domain names on a switch can be convenient, e.g., when configuring VPN peers or when troubleshooting with tools such as *ping* or *traceroute* (section 7.1.6).

It is also convenient to communicate *with* the switch using domain names. When the switch acquires its IP address dynamically (via DHCP), maintaining the DNS server entry is cumbersome. To manage this situation, WeOS includes support for dynamic DNS (DDNS). With DDNS enabled, the switch will update its DNS server entry automatically when acquiring a new IP address via DHCP.

Supported DDNS providers are **”dyndns”** (http://www.dyndns.org), **”freedns”** (http://freedns.afraid.org), and **”no-ip”** (http://www.no-ip.com).

17.2 Managing interfaces and general IP settings via the web interface

Menu path: Configuration ⇒ Network(IP) ⇒ Global settings

When entering the Network(IP) configuration page you will be presented to a list of common network settings.

Global Settings (Default Gateway, NTP server, Timezone, Routing and DNS servers)

Default Gateway	Statically configured default gateway of the unit. This is the IP address of the gateway to send packages to when no more specific route can be found in the routing table. *N/A* indicates that no default gateway address has been statically configured.
Remote NTP Server	The IP address of a time server to be used to keep the units calendar time synchronised. The text *Disabled* is shown if no NTP server address has been entered.
Timezone	Shows current timezone region. Used to adjust local time.
Routing	Routing, also known as IP-forwarding, allows traffic to flow between VLANs. Use the firewall to protect VLANs from unwanted traffic. Texts *Enabled* and *Disabled* shows routing status.
Domain Name Server(s)	List manually configured DNS servers. An empty field indicates that no DNS server has been manually configured.
Edit	Click this icon to edit ”this part” of the global settings.

This settings is described further in section 17.2.1.

To change the settings for a specific Interface click the associated edit icon which will take you to the interface settings edit page. Interface settings are described further in section 17.2.3.

17.2.1 Edit Common Network Settings

Menu path: Configuration ⇒ Network (IP) ⇒ Global settings ⇒ 

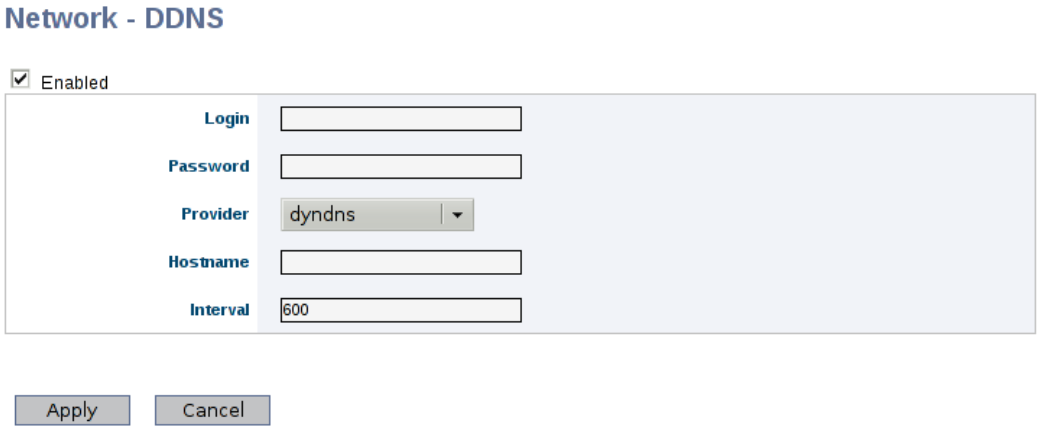
When clicking the *Edit* icon in will be the edit page.

Default Gateway	Statically configured default gateway of the unit. This is the IP address of the gateway to send packages to when no more specific route can be found in the routing table. Leave empty if no default gateway is desired.
Remote NTP Server	The IP address of a time server to be used to keep the units calendar time synchronised. Leave empty if you do not want to use a time server.
Timezone	Select a timezone region to get adjusted local time.
Routing	Routing, also known as IP-forwarding, allows traffic to flow between VLANs. Use the firewall to protect VLANs from unwanted traffic. Check this box to enable routing, uncheck to disable.
Name server 1	IP address of (primary) DNS server.
Name server 2	IP address of (secondary) DNS server.

Click the **Apply** button to save and apply the changes.

17.2.2 DDNS settings

Menu path: Configuration ⇒ Network (IP) ⇒ DDNS
Dynamic DNS (DDNS) provider settings



Dynamic DNS	Check this box to enable Dynamic DNS, uncheck to disable.
Login	Set login *username* for the account at your DDNS provider
Password	Set login *password* for the account at your DDNS provider
Provider	Select DDNS provider. Supported providers are **”dyndns”** (http://www.dyndns.org), **”freedns”** (http://freedns.afraid.org), and **”no-ip”** (http://www.no-ip.com)
Hostname	Set the DNS hostname, i.e., registered domain name which should map to the IP address of this your switch. When selecting freedns, the domain name must be followed by a hash value (**”HOSTNAME,HASH”**); the *hash* is provided by FreeDNS).
Interval	Set the interval by which DDNS verifies that the IP address mapping at your DDNS provider matches the IP address of your switch. Maximum 10 days (864000 seconds).

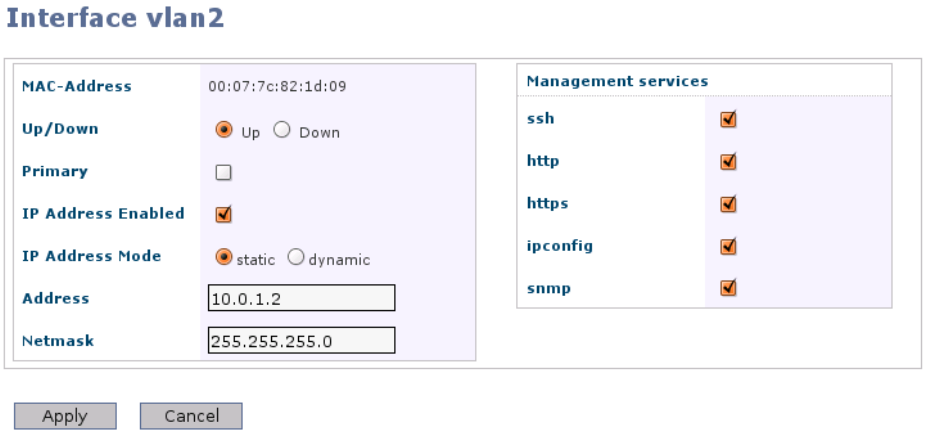
Click the **Apply** button to save and apply the changes.

17.2.3 Interface Settings

Menu path: Configuration ⇒ Network (IP) ⇒ Interface
Interfaces

Name	A unique identifier for the interface. Automatically generated from VLAN identifier when the VLAN is created.
Status	The status set on the interface, *up* or *down*.
Address	The IPv4 address assigned to the interface. Text *DISABLED* shown if IP address is disabled. *DHCPDISCOVER* shown if waiting for response from DHCP-server.
Netmask	The netmask for the IPv4 address. Identifies what IP addresses are located on the same subnet. Text *N/A* shown if IP address is disabled. *DHCPDISCOVER* shown if waiting for response from DHCP-server.
Edit	Click this icon to edit the interface.

When clicking the *Edit* icon for an interface you will be presented to its associated edit page.



Mac-Address	The media access control (MAC) address is used for controlling the communication on OSI layer 2. Shows the MAC-address associated to this interface.
Up/Down	The interface may be activated or deactivated by the up or down setting. Click the appropriate radio button to activate/deactivate the interface.
IP Address Enabled	When disabling the IP address, traffic may not be sent to the switch from units connected to the VLAN associated with this interface. The address may be disabled to e.g. prevent administration access from specific VLANs. The IP address mode field, and for static address mode the IP address and netmask fields, will not be visible unless this box has been checked (In JavaScript enabled browsers).
IP Address Mode	Choose *Static* to manually configure IP address and netmask or *Dynamic* t o let the unit query a DHCP server for address information.
Address	The IPv4 address assigned to the interface. This field will only be visible if static IP Address Mode has been selected
Netmask	The netmask for the IPv4 address. Identifies what IP addresses are located on the same subnet. This field will only be visible if static IP Address Mode has been selected

Click the **`Apply`** button to save and apply the changes.

The part of the interface edit window concerning management interfaces remains to be documents. Meanwhile, please see section 17.1.1.5 for more information.

17.3 Managing network interfaces via the CLI

The available interface settings and monitoring commands are shown in the table below:

Command	Default	Section
iface <IFNAME> inet <static|dhcp>	static	Section 17.3.1
[no] up	up	Section 17.3.2
[no] primary	Disabled	Section 17.3.3
[no] management <ssh|http|https|ipconfig|snmp>		Section 17.3.4
[no] mac <X:X:X:X:X:X>	Auto	Section 17.3.5
[no] mtu <<46-1500»	1500	Section 17.3.6
Only for inet static		
[no] address <ADDRESS/LEN| ADDRESS NETMASK>	Disabled	Section 17.3.7
Show interface configuration		
show iface [IFNAME]		Section 17.3.8
show ifaces		Section 17.3.9
iface <IFNAME> inet <static|dhcp>		
show up		Section 17.3.10
show address		Section 17.3.11
show primary		Section 17.3.12
show management		Section 17.3.13
show mac		Section 17.3.14
show mtu		Section 17.3.15
Show interface status		
show iface [IFNAME]		Section 17.3.16
show ifaces		Section 17.3.17

17.3.1 Manage Network Interfaces

Syntax `iface <IFNAME> inet <static|dhcp>`

Context *Global Configuration* context

Usage Enter *interface* context, and specify IP address assignment method.

- **"static"** means static IP address assignment. The IP address is config-

ured via the **”[no] address <ADDRESS/LEN|ADDRESS NETMASK>”** command, see section 17.3.7.

- If **”dhcp”** is selected, the switch attempts to acquire its address via DHCP. If no DHCP server is available, the interface will generally end up without an IP address. The exception is the *primary* interface, which will get a *link-local* IPv4 address if it fails to get an address via DHCP.

Default values static (That is, when an interface is created it will by default use static address assignment)

Error messages None defined yet.

17.3.2 Interface Administrative Mode (Up/Down)

Syntax `[no] up`

Context *interface* context

Usage Bring interface up/down. Note, even if an interface is configured administratively *up*, its operational status may still be *down* if the associated link is not up.

Default values Up

Error messages None defined yet.

17.3.3 Primary Interface

Syntax `[no] primary`

Context *interface* context

Usage Set this interface as primary interface. When configuring an interface as primary, the interface previously defined as primary will loose that property. Use **”no primary”** to unset this interface as primary.

For more information, see section 17.1.1.4.

Default values Disabled (`no primary`)

Error messages None defined yet.

17.3.4 Enable Management Services on Interface

Syntax `[no] management <ssh|http|https|ipconfig|snmp>`

Context *interface* context

Usage Enable and disable management services on this interface. This command controls whether it should be possible to manage the switch via this network interface, and if so, what services should be enabled.

Default values Disabled (except for the the interface associated with the primary VLAN.)

Error messages None defined yet.

17.3.5 Interface MAC address

Syntax `[no] mac <X:X:X:X:X:X>`

Context *interface* context

Usage Configure a specific MAC address for this (VLAN) interface. The address is given as a colon-separated hexadecimal string of numbers, e.g., **"mac 00:1a:4b:7b:77:24"**. Leading zeros can be ignored. Uppercase or lowercase letters can be used.

Use **"no mac"** specify that the interface should get its MAC address automatically.

For more information, see section 17.1.1.2.

Default values Auto (`no mac`)

Error messages None defined yet.

17.3.6 Interface MTU Size

Syntax `[no] mtu <68-1500>`

Context *interface* context

Usage Configure a non-default maximum transmission unit (MTU) size (in bytes) for this interface. The MTU size is the packet size a network interface will pass to the link layer for transmission, i.e., the maximum payload of the link layer protocol.

The default is to let the MTU depend on the type of link layer (*auto* mode). For interfaces associated with Ethernet and DSL links this implies a default MTU of 1500 bytes.

Use **"mtu <68-1500>"** to set a non-default MTU size. Use **"no mtu"** to specify that the interface should let its MTU be the default MTU of the associated link type.

Default values Auto (`no mtu`) For Ethernet and DSL links, this implies MTU 1500 bytes.

Error messages None defined yet.

17.3.7 IP Address

Syntax `[no] address <ADDRESS/LEN|ADDRESS NETMASK>`

Context *interface* context (only available when static address assignment is chosen, see section 17.3.1).

Usage Set static IP address and netmask for an interface.

It is possible to specify the boundary between the *network part* and the *host specific part* of the IP address either as a prefix length (e.g. **"address 192.168.0.1/24"**) or as a regular netmask (e.g., **"address 192.168.0.1 255.255.255.0"**).

Default values Disabled (`no address`). That is, newly created interfaces have no IP address configured, see also section 17.1.1.1.

Error messages None defined yet.

17.3.8 Show Network Interface Configuration

Syntax `show iface [IFNAME]`.

Context *Global Configuration* context. Also available as **"show"** command within the interface context.

Usage Show network interface configuration information of the given interface IFNAME (or all interfaces).

Default values All interfaces, i.e., if no interface IFNAME is provided, information on all interfaces will be shown.

Error messages None defined yet.

17.3.9 Show Configuration of all Interfaces

Syntax `show ifaces`

Context *Global Configuration* context.

Usage Show network interface configuration information all interfaces.

Default values Not applicable.

Error messages None defined yet.

17.3.10 Show Interface Administrative Mode

Syntax `show up`

Context *interface* context.

Usage Show whether this interface is administratively configured as enabled (up) or disabled (down).

Default values Not applicable.

Error messages None defined yet.

17.3.11 Show IP address Setting

Syntax `show address`

Context *interface* context.

Usage Show the IP address setting for this interface (static IP address, use of dynamic address assignment, or IP address disabled).

Default values Not applicable.

Error messages None defined yet.

17.3.12 Show Primary Interface Setting

Syntax `show primary`

Context *interface* context.

Usage Show the primary interface setting for this interface.

Default values Not applicable.

Error messages None defined yet.

17.3.13 Show Management Interface Setting

Syntax `show management`

Context *interface* context.

Usage Show if it is possible to manage the switch via this interface, and if so, what services (SSH, SNMP, etc.) that are enabled on this interface.

Default values Not applicable.

Error messages None defined yet.

17.3.14 Show Interface MAC Address Setting

Syntax `show mac`

Context *interface* context.

Usage Show the interface MAC address setting.

Default values Not applicable.

Error messages None defined yet.

17.3.15 Show Interface MTU Size Setting

Syntax `show mtu`

Context *interface* context.

Usage Show the interface maximum transfer unit (MTU) size setting.

Default values Not applicable.

Error messages None defined yet.

17.3.16 Show Network Interface Status

Syntax `show iface [IFNAME]`

Context *Admin Exec* context.

Usage Show status information for this interface (or all interfaces). If dynamic address assignment is configured on an interface, this command will display the IP address acquired.

Default values Unless a specific interface is specified, status for all interfaces will be shown.

Error messages None defined yet.

17.3.17 Show Status of all Interfaces

Syntax `show ifaces`

Context *Admin Exec* context.

Usage Show status information for all interfaces. If dynamic address assignment is configured on an interface , this command will display the IP address acquired.

Default values Not applicable.

Error messages None defined yet.

17.4 Managing general IP settings via the CLI

The available general IP settings and monitoring commands are shown below.

Command	Default	Section
Configure general IP settings		
ip		Section 17.4.1
[no] default-gateway <ADDRESS>	Disabled	Section 17.4.2
[no] route <NETWORK/LEN| NETWORK NETMASK>		Section 17.4.3
[no] forwarding	Enabled	Section 17.4.4
[no] name-server <ADDRESS>	Disabled	Section 17.4.5
[no] domain <DOMAIN>	Disabled	Section 17.4.6
[no] ddns	Disabled	Section 17.4.7
[no] login <USERNAME> <PASSWORD>	Disabled	Section 17.4.8
[no] provider <dyndns|freedns|no-ip>	dyndns	Section 17.4.9
[no] hostname <HOSTNAME>[,HASH]	Disabled	Section 17.4.10
[no] interval <SECONDS>	600	Section 17.4.11
icmp		Section 17.4.12
[no] broadcast-ping	Enabled	Section 17.4.13
[no] sntp	Disabled	Section 17.4.14
[no] server <ADDRESS>	Disabled	Section 17.4.15
[no] poll-interval <SECONDS>	600 sec	Section 17.4.16
Show general IP settings		
show ip		Section 17.4.17
ip		
show default-gateway		Section 17.4.18
show route		Section 17.4.19
show forwarding		Section 17.4.20
show name-server		Section 17.4.21
show domain		Section 17.4.22

Continued on next page

Continued from previous page

Command	Default	Section
Show general IP settings (cont.)		
ip		
show ddns		Section 17.4.23
icmp		
show broadcast-ping		Section 17.4.24
show sntp		Section 17.4.25
sntp		
show server		Section 17.4.26
show poll-interval		Section 17.4.27
Show general IP status		
show ip route		Section 17.4.29
show ip name-server		Section 17.4.30

17.4.1 Manage Global IP Settings

Syntax `ip`

Context *Global Configuration* context

Usage Enter *IP* context

Default values Not applicable.

Error messages None defined yet.

17.4.2 Configure IP Default Gateway

Syntax `[no] default-gateway <ADDRESS>`

Context *IP* context

Usage Add/remove default gateway. Use **"no default-gateway"** to remove default gateway.

The default gateway could alternatively be configured via the **"route"** command (e.g., **"route 0.0.0.0/0 192.168.0.1"**), see also section 17.4.3.

If a default route is configured using the **"default-gateway"** command (or **"route"** command), a default gateway acquired via DHCP on the primary interface will be ignored.

Default values Disabled (**"no default-gateway"**)

Error messages None defined yet

17.4.3 Configure Static IP Routes

Syntax `[no] route <SUBNETADDR NETMASK | SUBNETADDR/LEN> <GWADDR>`

Context *IP* context

Usage Add/remove a static IP route. The network boundary of the destination subnet can be given as a netmask (e.g., **"route 192.168.3.0 255.255.255.0 192.168.0.1"**) or as a prefix length (e.g., **"route 192.168.3.0/24 192.168.0.1"**). Use the "no"-form to remove a static route, e.g., **"no route 192.168.3.0/24 192.168.0.1"**. **"no route"** will remove all configured routes (except static route to default gateway, see the **"default-gateway"** command).

Default values Using **"no route"** (without a subnet address, etc.) removes all configured static routes (except static route to default gateway, see the **"default-gateway"** command in section 17.4.2).

Error messages None defined yet

17.4.4 Manage IP Forwarding

Syntax `[no] forwarding`

Context *IP* context

Usage Enable/disable IPv4 routing.

Default values Enabled (**"forwarding"**)

Error messages None defined yet

17.4.5 Name Server (DNS)

Syntax `[no] name-server <ADDRESS>`

Context *IP* context

Usage Add/remove name-server (DNS). Two name-servers can be configured - call the same **"name-server"** command twice.

Run **"no name-server <ADDRESS>"** to remove a specific name server, or **"no name-server"** to remove all configured name servers.

If a name server is configured using the **"name-server"** command, name server(s) (and domain search path) acquired via DHCP on the primary interface will be ignored.

Default values Disabled (**"no name-server"**) Running **"no name-server"** (without specifying any name removes all configured name servers.

Error messages None defined yet

17.4.6 Domain Search Path

Syntax `[no] domain <DOMAIN>`

Context *IP* context

Usage Add/remove domain search path. A single search path can be added. Run **"no domain"** to remove the domain search path.

If a name server is configured using the **"name-server"** command, domain(s) acquired via DHCP on the primary interface will be ignored.

Default values Disabled (**"no domain"**)

Error messages None defined yet

17.4.7 Manage DDNS Settings

Syntax `[no] ddns`

Context *IP* context

Usage Enter *ddns* context. Upon entering the context, the DDNS service will be enabled. However, it will not be activated until valid DDNS parameters (login, etc.) are configured. Use **"no ddns"** to disable the DDNS service.

Default values Disabled (**"no ddns"**)

Error messages None defined yet.

17.4.8 Set DDNS Login and Password

Syntax `[no] login <USERNAME> <PASSWORD>`

Context *ddns* context

Usage Set login *username* and *password* for your account at your DDNS provider (see section 17.4.9). Use **"no login"** to remove a configured DDNS login setting.

Default values Disabled

Error messages None defined yet.

17.4.9 Set DDNS Provider

Syntax `[no] provider <dyndns|freedns|no-ip>`

Context *ddns* context

Usage Set DDNS provider. Supported providers are **"dyndns"** (http://www.dyndns.org), **"freedns"** (http://freedns.afraid.org), and **"no-ip"** (http://www.no-ip.com). Use **"no provider"** to return to the default provider setting.

Default values dyndns

Error messages None defined yet.

17.4.10 Set DDNS Hostname

Syntax `[no] hostname <HOSTNAME>[,HASH]`

Context *ddns* context

Usage Set the DNS hostname, i.e., registered domain name which should map to the IP address of this your switch.

When selecting **"provider freedns"**, the domain name must be followed by a hash value (**"hostname HOSTNAME,HASH"**); the *hash* is provided by FreeDNS).

Default values Disabled

Error messages None defined yet.

17.4.11 Set DDNS interval

Syntax `[no] interval <SECONDS>`

Context *ddns* context

Usage Set the interval by which DDNS verifies that the IP address mapping at your DDNS provider matches the IP address of your switch. Maximum 10 days (864000 seconds).

Use **"no interval"** to return to the default provider setting.

Default values 600 (seconds)

Error messages None defined yet.

17.4.12 Manage ICMP Settings

Syntax `icmp`

Context *IP* context

Usage Enter *ICMP* context.

Default values Not applicable.

Error messages None defined yet.

17.4.13 Enable/disable Broadcast Ping

Syntax `[no] broadcast-ping`

Context *ICMP* context

Usage Define whether the switch should respond to broadcast ”ping” (ICMP Echo Request) messages or not. Responding to broadcast ping is convenient when troubleshooting the network, but can in some situations be considered a security risk.

Use **”no broadcast-ping”** to disable responding to broadcast ping messages.

Default values Enabled (**”broadcast-ping”**)

Error messages None defined yet.

17.4.14 Manage SNTP Settings

Syntax `[no] sntp`

Context *Global Configuration* context

Usage Enter *sntp* context. Upon entering the context, the SNTP service will be enabled. However, it will not be activated until valid SNTP parameters (server and polling interval) are configured. Use **”no sntp”** to disable the SNTP service.

Default values Not applicable.

Error messages None defined yet.

17.4.15 Set SNTP Server Address

Syntax `[no] server <ADDRESS>`

Context *sntp* context

Usage Set IP Address of SNTP Server. A single SNTP server IP address can be configured. Use **”no server”** to remove a configured SNTP server address.

Default values Not applicable.

Error messages None defined yet.

17.4.16 Set SNTP Poll Interval

Syntax `[no] poll-interval <30-720>`

Context *sntp* context

Usage Set SNTP server poll interval (in seconds). **"no poll-interval"** will reset the poll interval to its default (600 seconds).

Default values 600 (seconds)

Error messages None defined yet.

17.4.17 Show General IP Settings

Syntax `show ip`

Context *Global Configuration* context

Usage Show general IP settings.

Default values Not applicable.

Error messages None defined yet.

17.4.18 Show Default Gateway Setting

Syntax `show default-gateway`

Context *IP* context

Usage Show general IP settings.

Default values Not applicable.

Error messages None defined yet.

17.4.19 Show Configured Static Routes

Syntax `show routes`

Context *IP* context

Usage Show configured static routes.

Default values Not applicable.

Error messages None defined yet.

17.4.20 Show IP Forwarding Setting

Syntax `show forwarding`

Context *IP* context

Usage Show whether IP forwarding (routing) is enabled or disabled.

Default values Not applicable.

Error messages None defined yet.

17.4.21 Show Configured Name Servers

Syntax `show name-server`

Context *IP* context

Usage Show configured name servers.

Default values Not applicable.

Error messages None defined yet.

17.4.22 Show Configured Domain Search Path

Syntax `show domain`

Context *IP* context

Usage Show configured domain search path.

Default values Not applicable.

Error messages None defined yet.

17.4.23 Show DDNS settings

Syntax `show ddns`

Context *IP* context. Also available as **"show"** command within the DDNS context.

Usage Show DDNS settings.

Default values Not applicable.

Error messages None defined yet.

17.4.24 Show Broadcast Ping setting

Syntax `show broadcast-ping`

Context *ICMP* context.

Usage Show whether the switch is configured to respond to broadcast ping messages or not.

Default values Not applicable.

Error messages None defined yet.

17.4.25 Show SNTP settings

Syntax `show sntp`

Context *Global Configuration* context. Also available as **"show"** command within the SNTP context.

Usage Show SNTP settings.

Default values Not applicable.

Error messages None defined yet.

17.4.26 Show SNTP Server Setting

Syntax `show server`

Context *SNTP* context.

Usage Show SNTP server settings.

Default values Not applicable.

Error messages None defined yet.

17.4.27 Show SNTP Polling Interval Setting

Syntax `show poll-interval`

Context *SNTP* context

Usage Show configured SNTP poll interval.

Default values Not applicable.

Error messages None defined yet.

17.4.28 Show IGMP Snooping Status Information

Syntax `show ip igmp`

Context *Admin Exec* context

Usage Show IGMP snooping status information.

Default values Not applicable.

Error messages None defined yet.

17.4.29 Show IP Forwarding Table

Syntax `show ip route`

Context *Admin Exec* context

Usage Show IP Forwarding table (summary of configured routes and routes acquired dynamically).

Default values Not applicable.

Error messages None defined yet.

17.4.30 Show Name Server and Domain Status Information

Syntax `show ip name-server`

Context *Admin Exec* context

Usage Show name-server and domain search path status information (statically configured or acquired dynamically)

Default values Not applicable.

Error messages None defined yet.

Chapter 18

Multicast in Switched Networks (IGMP Snooping)

When distributing IP multicast data in a switched network, the switches within the LAN can either:

- treat the traffic as broadcast, and then forward it onto all ports (in the same VLAN), or
- limit the forwarding of multicast packets to those ports leading to subscribers of the specific IP multicast group

The latter method requires the switches to inspect Internet Group Management Protocol (IGMP) messages exchanged by hosts and routers to learn which ports leads to subscribers - this mechanism is referred to as *IGMP snooping*[1].

As part of the IGMP snooping support, WeOS also enables a *switch* to act as *IGMP querier* - a role which is usually handled by a *multicast router*. Having switches with IGMP querier capabilities enables efficient distribution of IP multicast in networks without multicast routers.

18.1 Overview of IGMP Snooping Settings

Feature	Web (Sec. 18.2-	CLI (Sec. 18.3-	General Description
IGMP Snooping Settings			
IGMP querier mode	X	X	Sec. 18.1.1
IGMP query interval	X	X	”
IGMP multicast router ports	X	X	”
View IGMP Snooping Settings	X	X	

18.1.1 IGMP Snooping

The switch is capable of efficiently distributing IP(v4) multicast traffic on LAN interfaces by means of IGMP snooping. IGMP Snooping is enabled per VLAN as described in section 13.1.5.

- With IGMP snooping *enabled* on a VLAN, IP multicast packets will only be forwarded onto ports leading to a receiver of that IP multicast address, or to ports assumed to lead to an IP multicast router.
- With IGMP snooping *disabled* on a VLAN, multicast traffic will be forwarded on all ports of that VLAN, i.e., it is treated similar to broadcast traffic.
- Port that are shared between multiple VLANs may have different IGMP snooping settings on different VLANs, i.e., one VLAN may have IGMP snooping *enabled* and another may have IGMP snooping *disabled*. The *disabled* mode has precedence on such ports, i.e., a port will ”flood/broadcast” all multicast traffic if (at least) one of the VLANs this port belongs to has IGMP Snooping *disabled*.

As part of the IGMP snooping functionality, the switch can also act as an IGMP Querier, and settings for *querier mode*, and *query interval* are provided.

- Querier mode: By default the switch will use *auto* mode, meaning that it follows the standard IGMP protocol to elect a designated IGMP querier on each LAN (the querier with the lowest IP address on each LAN becomes the querier). The switch can also be configured to always act as querier, or to act in *proxy* querier mode. In proxy mode, the switch will not send any IGMP queries by itself, but relay IGMP Queries received. The IGMP Proxy will modify the source IP address of the relayed IGMP Queries to 0.0.0.0 to indicate that it is not a multicast router.

  On VLANs where the network interface has not been assigned any IP address, the switch will revert to *proxy* mode irrespective of the querier mode setting.

  > **Warning**: *For proper multicast distribution there must be an IGMP Querier present on every VLAN where IGMP snooping is enabled. On VLANs where all switches operate in* IGMP proxy querier mode*, perhaps because none of them was assigned an IP address on that VLAN, there is a risk that multicast traffic will be blocked. If a switch is intended to act as IGMP querier on a VLAN, that switch must be assigned an IP address its associated VLAN network interface.*

- Query interval: The switch can be configured to send out queries on intervals 12, 30, 70 and 150 seconds (default 12 seconds).

When IGMP snooping is enabled, the switch will learn on which ports there are interested receivers of a certain multicast group, by listening to IGMP Report messages sent by the member nodes. Thus, the switch will only forward multicast packets on those ports leading to a member of that specific multicast group. In addition, a switch will forward *all* multicast traffic on ports which may lead to a multicast router. The current IGMP implementation considers the following ports to be *multicast router ports*:

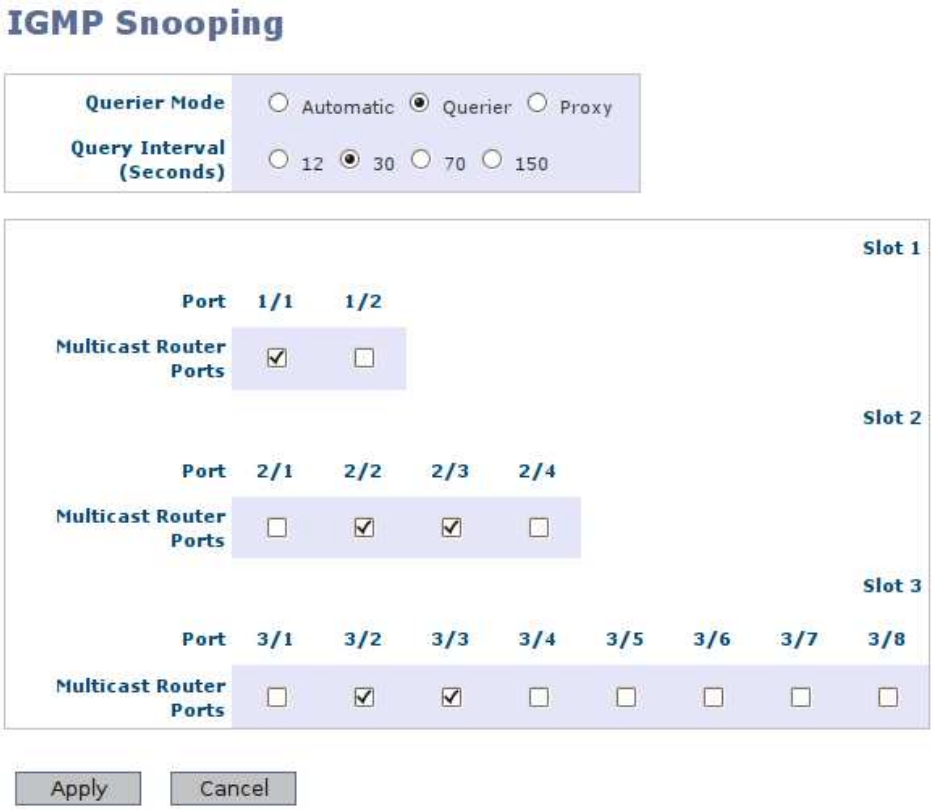
- Ports configured as multicast router ports: The operator can define ports as multicast router ports.
- Ports leading to an IGMP Querier: Ports where the switch receives IGMP Queries are dynamically added to the list of multicast router ports.
- FRNT ports: If FRNT is enabled on the switch, the FRNT ring ports are added to the list of multicast router ports. This ensures multicast traffic to perceive the benefit of FRNT's fast recovery mechanism in case the ring is broken.

When a multicast reciever attached to a switch port leaves a multicast group (i.e., stops subscribing to an IP multicast address or is simply disconnected from port), the IGMP snooping leave latency (the time until the switch stops forwarding the associated multicast data) is within 2-3 times the configured *Query Interval*.

18.2 Managing IGMP Snooping settings via the web interface

Menu path: Configuration ⇒ IGMP

When entering the IGMP configuration page you will be presented to the global settings for IGMP. Enabling of IGMP is done per VLAN, see Section 13.



Querier Mode	Select the query mode by clicking on the appropriate radio button. **Automatic** Activates automatic querier election. Recommended. **Querier** In Forced Querier mode the device always starts a new IGMP query every Query Interval seconds. **Proxy** A fall-back mode in which the switch never initiates queries by itself, only forwards queries and reports.

Continued on next page

	Continued from previous page
Query Interval	Number of seconds between each query. For the least amount of latency 12 seconds is recommended. Select the query interval by clicking on the appropriate radio button.
Multicast Router Ports	A selection of ports on which to enable multicast traffic. Useful if the device fails to automatically detect any multicast routers on the subnet. Check the box for each port that you wish to act as a multicast router port.

Click the **Apply** button to save and apply the changes.

18.3 Managing IGMP Snooping settings via the CLI

The available general IP settings and monitoring commands are shown below.

Command	Default	Section
Configure General IGMP Snooping settings		
ip		Section 17.4.1
igmp-mode <auto|querier|proxy>	auto	Section 18.3.1
igmp-interval <12|30|70|150>	12 sec	Section 18.3.2
[no] mcast-router-ports <PORTLIST>	Disabled	Section 18.3.3
show ip		Section 17.4.17
ip		
show igmp		Section 18.3.4
show igmp-mode		Section 18.3.5
show igmp-interval		Section 18.3.6
show mcast-router-ports		Section 18.3.7
Per VLAN IGMP Snooping settings		
vlan <VID>		Section 13.3.3
[no] igmp	Enabled	Section 13.3.10
show igmp		Section 13.3.20
Show IGMP Snooping Status		
show ip igmp		Section 17.4.28

18.3.1 IGMP Querier Mode

Syntax `igmp-mode <auto|querier|proxy>`

Context *IP* context

Usage Set IGMP Querier mode. In ”auto” the device will participate in the querier election process (querier with lowest IP becomes querier). In ”querier” mode the device will continue to send IGMP queries even if there are other querier present with lower IP address. In ”proxy” mode the device will act as an IGMP proxy. Note that if there is no IP address configured for an interface, the device will fall back to proxy mode regardless of the mode setting.

Default values `auto`.

Error messages None defined yet

18.3.2 IGMP Querier Interval

Syntax `igmp-interval <12|30|70|150>`

Context *IP* context

Usage Set IGMP Querier interval (seconds). The same interval is used for all interfaces.

Default values 12 (seconds).

Error messages None defined yet

18.3.3 Static Multicast Router Port Settings

Syntax `[no] mcast-router-ports <PORTLIST>`

Context *IP* context

Usage Add or remove multicast router ports. All (layer-2) multicast traffic will be forwarded on multicast router ports, see section 18.1.1.

Default values Using **"no mcast-router-ports"** (without a PORTLIST) removes all configured multicast router ports.

Error messages None defined yet

A **"PORTLIST"** is a comma separated list of port ranges without intermediate spaces, e.g., **"1/1-1/3,2/3"**.

18.3.4 Show IGMP Settings

Syntax `show igmp`

Context *IP* context

Usage Show summary of all IGMP snooping related settings.

Default values Not applicable.

Error messages None defined yet.

18.3.5 Show IGMP Querier Mode Setting

Syntax `show igmp-mode`

Context *IP* context

Usage Show configured IGMP querier mode (**"auto"**, **"querier"** or **"proxy"**)

Default values Not applicable.

Error messages None defined yet.

18.3.6 Show IGMP Query Interval Setting

Syntax `show igmp-interval`
Context *IP* context
Usage Show configured IGMP interval.
Default values Not applicable.
Error messages None defined yet.

18.3.7 Show Configured Multicast Router Ports

Syntax `show mcast-router-ports`
Context *IP* context
Usage Show configured multicast router ports. (Relates to IGMP snooping.)
Default values Not applicable.
Error messages None defined yet.

Chapter 19

IP Routing in WeOS

In addition to *switching* (layer-2), WeOS devices are capable to *route* data packets (layer-3), i.e., they are *routing switching*. The WeOS routing support includes static routing and dynamic routing via OSPF and RIP, as well as other useful router features such as firewall, NAT, VPN, VRRP, and DHCP server.

This chapter introduces the IP routing capabilities in WeOS in general. More information on dynamic routing is found in chapters 20 (OSPF) and 21 (RIP), and supplementary router services are covered in the chapters to follow.

19.1 Summary of WeOS Routing and Router Features

Table 19.1 presents the routing and router features available in WeOS.

Feature	Web	CLI (Sec. 19.4)	General Description
Enable/disable routing	X	X	Sections 19.2-19.3
Default gateway	X	X	Sections. 19.2-19.3
Static routing		X	Sections 19.2-Sec. 19.3
Dynamic routing			
- RIP (v1/v2)		X	Section 19.2, chapter 21
- OSPF		X	Section 19.2, chapter 20
Router redundancy (VRRP)		X	Section 19.2, chapter 22
Firewall and NAT	X	X	Section 19.2, chapter 23
Virtual Private Network (VPN)	X	X	Section 19.2, chapter 24
DHCP Server		X	Section 19.2, chapter 25

Table 19.1: Summary of router and routing features.

19.2 Introduction to WeOS Routing and Router Features

IP routing enables us to connect our networks together, and to let (TCP/IP) devices communicate across networks of different type and topology, and possibly over multiple network ”hops” and long distances. A router looks at the destination IP address carried within each IP packet, consults its *routing table* to make a routing decision, and forwards the packet onto the next router in the path to the destination.

The routing table can either be managed manually via *static IP routing*, or automatically by using dynamic routing protocols, or a combination of both. Static IP routing is usually fine for small IP networks, or networks with no redundant paths. To manage routing in larger networks, it is preferred to use *dynamic IP routing*. With dynamic routing, the routers will exchange routing information and build up their routing tables dynamically. Furthermore, dynamic routing utilises network redundancy; if a link goes down, routers will inform each other and packets will automatically be routed along another path. Thus, dynamic routing protocols perform a similar service in routed networks as FRNT (chapter 14) and RSTP (chapter 15) perform in switched networks. The time to react on a topology change is referred to as the *convergence time*. WeOS supports two dynamic routing protocols: Open Shortest Path First (OSPF) and Routing Information Protocol (RIP). OSPF is the recommended over RIP, due to its superior *convergence* characteristics.

OSPF and RIP are both examples of unicast Interior Gateway Protocols (IGPs), which means they can be used to handle routing *within* a routing domain, such as an corporate network. This is also referred to as *intra-domain* routing, as opposed *inter-domain* routing, which is commonly handled using the Border Gateway Protocol (BGP)[1]. OSPF and RIP are covered in chapters 20 and 21 respectively.

While dynamic routing protocols such as RIP and OSPF enable routers to find redundant paths in case a link or router goes down, it does not enable end devices (hosts) to use a second router if their *regular* router goes down. To support redundancy between hosts and routers the Virtual Router Redundancy Protocol (VRRP) is used. With VRRP, a backup router will take over if a router fails, and communication from connected hosts can continue automatically. VRRP support is covered in chapter 22.

[1] As of WeOS v4.3.0, dynamic routing is limited to intra-domain routing with RIP and OSPF. WeOS does *not* support dynamic inter-domain routing via BGP (Border Gateway Protocol), or IP multicast routing.

When a router is used as a company gateway to a public network, such as the Internet, there is an obvious need to protect the local company network against network intrusion and other attacks. It is also common that the hosts and routers within the company network use *private* IP addresses. To protect the company network and to enable the use of private IP addresses, WeOS includes *firewall* and *network address translation* (NAT) support. Chapter 23 describes the NAT and firewall features in WeOS.

Another need which occurs when connecting company networks to the Internet is to ensure communication privacy. WeOS support IPSec VPN to establish secure communication over public networks. With IPSec VPNs, a company can secure communication between a head office and different branch offices by installing a WeOS device as VPN gateway at each site. WeOS VPN support is covered in chapter 24.

WeOS includes DHCP server support, which is used to dynamically configure IP settings such as IP address, netmask, default gateway and DNS server(s) to attaches host. This removes the need to install a separate DHCP server on every IP subnet. Chapter 25 describes WeOS DHCP server support.

19.3 General IP Routing Settings and Hints

19.3.1 Using a WeOS device as a switch or as a router

WeOS devices are both able to route and to switch packets, i.e., they are *routing switches*. Switching is performed between ports in the same VLAN, while routing is performed between IP subnets or network interfaces (please see fig. 17.1 in section 17.1.1 for information on the distinction between ports, VLANs and network interfaces in WeOS). Routing can be disabled, and the WeOS device will then act as a VLAN capable *switch*.

19.3.2 Static routing

WeOS supports static IP routing. With static routing a WeOS devices can specify the next hop router to use to reach a given IP subnet. As of WeOS v4.3.0, configuration of static routes is limited to the CLI. A special case is the *default gateway*, which is also possible to configure via the Web.

19.3.3 Learning routing information from different sources

A WeOS device will learn about routing information by manual configuration (connected interfaces or static routes), or via dynamic routing protocols (OSPF and RIP). As described in chapters 20 and 21, a router is able to redistribute external routing information into an OSPF or RIP routing domain.

In some situations a router will learn the route to the same destination through different mechanisms. In this case, the route to use will depend on the *administrative distance* associated with the involved routing mechanisms. A route with a lower administrative distance will be prioritised over a router with higher administrative distance.

As of WeOS the administrative distance of connected routes, static routes, and routes learnt dynamically via RIP and OSPF will be associated with fixed administrative distances as shown below. Administrative distance configuration support is planned, but not yet implemented in WeOS.

	Administrative Distance
Connected	0
Static	0
OSPF	110
RIP	120

Static routes commonly have administrative distance 1 by default, while WeOS currently assigns static routes administrative distance 0. This is likely to change in the future.

19.3.4 Limitations When Using RSTP and Routing

As of WeOS v4.3.0 a single RSTP instance per WeOS unit is supported. This works fine in a switched environment where all VLANs on a switch can be added to inter-switch ports, see also chapters 13 (VLAN) and 15 (RSTP/STP).

However, when using RSTP in a routed environment it is often needed to run a separate instance of RSTP per VLAN. Otherwise there is a risk that RSTP incorrectly detects a loop (at layer-2) and blocks some port, even though there is a ”routing barrier”, which already handles the loop. The result of RSTP blocking ports may be loss of connectivity at layer-3.

RSTP is typically enabled on all ports by default. When using the WeOS device as a router, it is therefore recommended either to

- disable RSTP as a whole, or
- disable RSTP on all ports but one VLAN, or a group of VLANs with a shared layer-2 backbone (such as a ring).

Support for multiple RSTP/STP instances is planned but not yet implemented.

19.4 Enabling Routing and Managing Static Routing via CLI

The table below shows WeOS CLI commands relevant for handling static routing. The detailed description of these commands is found in other chapters as listed in the table.

Command	Default	Section
Configure general routing settings		
ip		Section 17.4.1
[no] default-gateway <ADDRESS>	Disabled	Section 17.4.2
[no] route <NETWORK/LEN| NETWORK NETMASK>		Section 17.4.3
[no] forwarding	Enabled	Section 17.4.4
Show general routing settings		
show ip		Section 17.4.17
ip		
show default-gateway		Section 17.4.18
show route		Section 17.4.19
show forwarding		Section 17.4.20
Show general routing status		
show ip route		Section 17.4.29

Chapter 20

Dynamic routing with OSPF

This chapter describes WeOS support for the OSPF dynamic routing protocol. As of WeOS release v4.3.0, management of OSPF is only available via the CLI.

20.1 Overview of OSPF features

The table below summarises OSPF support in WeOS.

Feature	CLI	General Description
General OSPF settings		
Router-id	X	Sec.20.1.1.1
OSPF Networks	X	Sec.20.1.1.1
Area type (regular, stub, NSSA)	X	Secs. 20.1.1.2, and 20.1.1.4-20.1.1.5
Redistribution (static, connected, RIP)	X	Sec. 20.1.1.3
Distribute default route	X	Sec. 20.1.1.3
Inter-area summarisation	X	Sec. 20.1.1.6
Inter-area filtering	X	Sec. 20.1.1.6
Passive interface default	X	Sec. 20.1.1.7
Per interface OSPF settings		
Link cost	X	Sec. 20.1.1
Passive interface	X	Sec. 20.1.1.7
Authentication (MD5, plain)	X	Sec. 20.1.1.8
Hello/Dead intervals	X	Sec. 20.1.1.9
Designated Router priority	X	Sec. 20.1.1.10

Note: *As of WeOS v4.3.0 there is no support for "load balancing" in case there are multiple paths with equal cost to reach a destination. When an OSPF configuration change is done in WeOS, OSPF will be restarted on that router. Until the OSPF routing protocol has converged, this may cause a temporary loss of connectivity in parts of your network.*

20.1.1 OSPF introduction

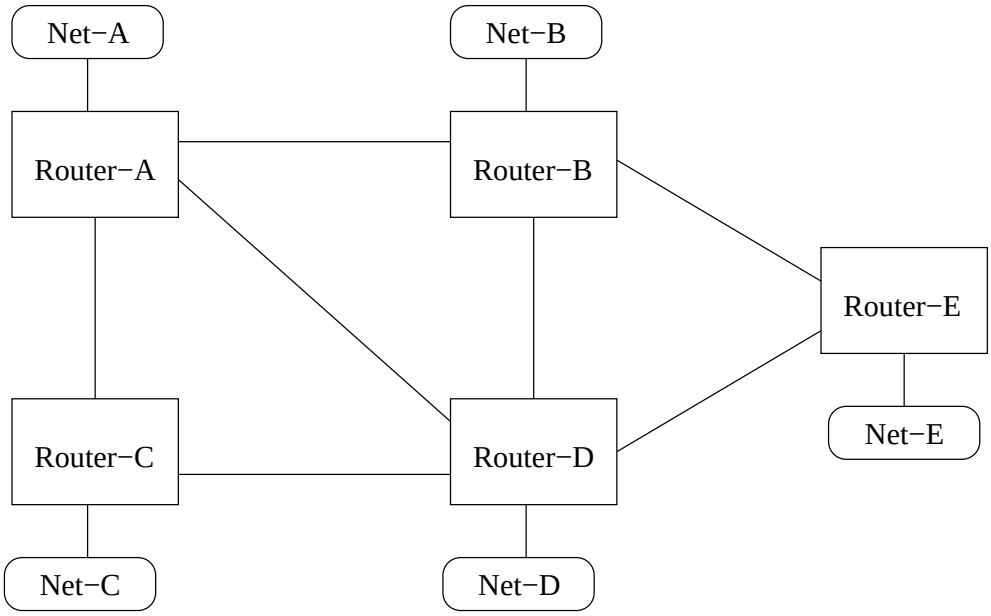


Figure 20.1: Simple network topology with interconnected routers and networks.

Dynamic routing protocols such as OSPF and RIP (chapter 21) simplifies router configuration, and improves network robustness.

- *Simplified configuration:* Manual configuration of static routes is not needed, and thereby a time consuming and error-prone procedure is avoided. In the network shown in fig. 20.1, each router would only have to be configured with information about its own identity and the IP subnets it is attached to. Routers will then exchange this information, and be able to establish the appropriate routing table by themselves.
- *Improved robustness:* If the topology changes, perhaps because a link failed, routers will automatically detect this and inform each other. The data traffic

will be forwarded other ways, given that a redundant path to the destination exists.

OSPF is an example of a *link-state* routing protocol. In a link-state routing protocol, each router announces information about its own identity (*router-id*), its directly connected networks, and its neighbour routers. This information is *flooded* throughout the OSPF domain, and each router will store the information in a local OSPF database. Each router will gain complete knowledge about every router and link in the whole topology, and is therefore able to compute the best path (the least cost path) to reach every destination[1].

For example, Router-A in fig. 20.1 would send out OSPF messages informing other routers about its *router-id*, its connected networks, i.e., Net-A and the links towards routers A, B, and C, the identity of (and link to) to its neighbour routers (A, B and C).

A major advantage of link-state routing protocols, such as OSPF, over distance vector routing protocols, such as RIP, is the *fast convergence* after a topology change. If a link goes down, information about this can be flooded rapidly to all routers within the routing domain, and each router can then update their routing table accordingly.

20.1.1.1 OSPF Router-ID and OSPF Networks

We use the example below to explain some essential OSPF parameter settings (the example is for *Router-A* in fig. 20.2).

```
router
        ospf
                router-id 10.0.11.1
                network 10.0.1.0/24 area 0.0.0.0
                network 10.0.2.0/24 area 0.0.0.0
                network 10.0.3.0/24 area 0.0.0.0
                network 10.0.11.0/24 area 0.0.0.0
                end
        end
```

The **”router-id”** line states the identity of this OSPF router, and must be unique within this OSPF routing domain.

[1] In OSPF, a cost is associated with every link. As of WeOS v4.3.0, the default cost per link is ”10”. The link cost can be configured per interface, see section 20.2.25 for details.

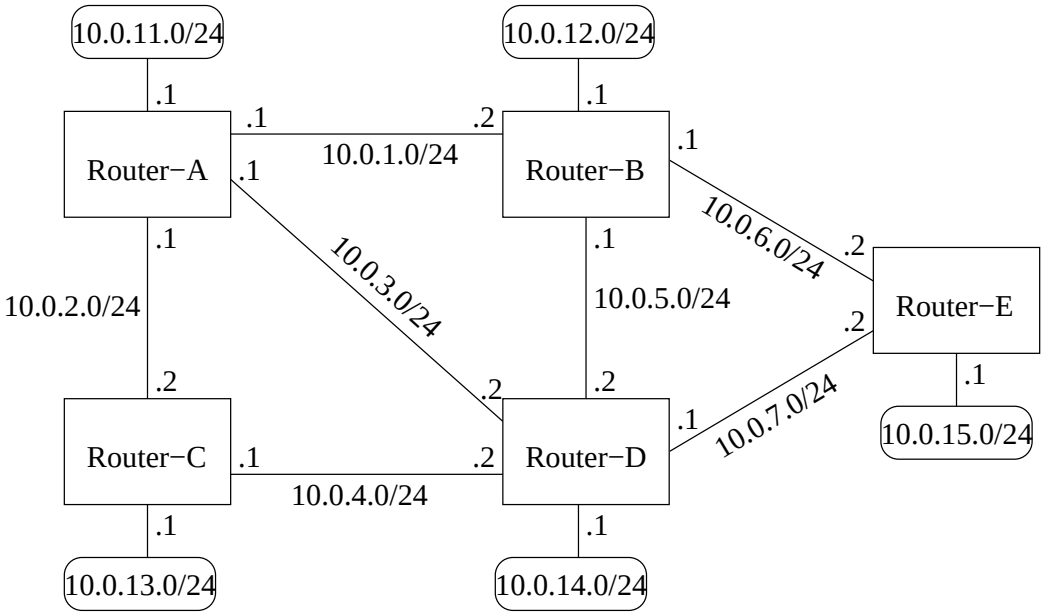


Figure 20.2: Example OSPF network with IP addresses and subnets.

- The router-id is 32-bit value, and can be specified either as a regular integer value, or in *dotted-decimal* form, just like an IP address.
- It is *common practise* to set the *router-id* to one of the IP addresses assigned to the router.
- If no router-id is configured, WeOS will pick one of the router's configured IP addresses, and use that as router-id.

As mentioned in section 20.1.1, the router should inform the other routers about its attached links and networks. However, a router will announce its networks and links first when they are declared to be within the OSPF routing domain – this is done via the **"network"** command. Furthermore, a **"network"** declaration implies that OSPF messages will be exchanged through the corresponding network interface. (In some network setups one likes to include a subnet within the OSPF domain, without activating OSPF on the corresponding interface. This can be achieved by configured that interface as *passive*, see section 20.1.1.7.)

In the example above, Router-A has been configured to include and announce all its subnets in the OSPF domain (10.0.1.0/24, 10.0.2.0/24, etc.). From the example we can also see that the **"network"** declaration contains an *area* parameter. OSPF areas are further explained in section 20.1.1.2.

20.1.1.2 OSPF hierarchy and areas

Being a link state protocol, OSPF requires routers to keep a lot of routing information in their database:

- Each OSPF router will typically keep a database with information of every router and link in the whole OSPF domain.
- OSPF routers will also redistribute and keep routing information learnt from external sources (static routes, routes learnt via other routing protocols, etc.).

To reduce the burden of keeping keeping state information about the whole OSPF domain, the domain can be split into OSPF *areas*. (For information on how to avoid the need to keep information on external routing information, see section 20.1.1.4.)

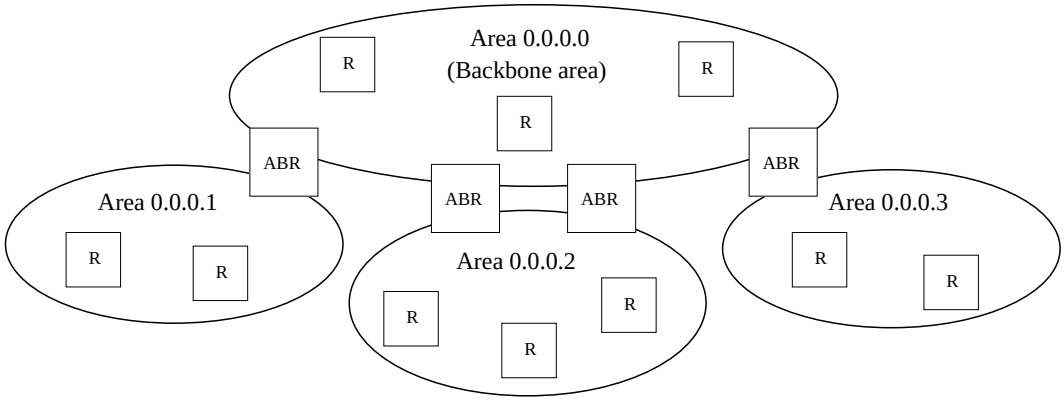


Figure 20.3: Sample OSPF hierarchy with a backbone area and three other areas.

The routers in fig. 20.3 have been divided into four areas. When splitting the network into multiple areas, each router will only have full knowledge of the topology within their respective area. Routers will also keep *summary* information about destinations outside their own area, but routers will not have knowledge about the actual topology inside other areas.

Each IP subnet can only part of one OSPF area, and when configuring OSPF networks you should also define which area it belongs to. The area identifier is a 32 bit value, which can be stated as a decimal value, but is commonly written in *dotted decimal form*. E.g., **”network 10.0.1.0/24 area 0.0.0.0”** is equivalent to writing **”network 10.0.1.0/24 area 0”**.

A router which have networks in different areas is called an *area border router* (ABR). An example is given below.

```
router
        ospf
                router-id 192.168.5.11
                network 192.168.5.0/24 area 0.0.0.0
                network 192.168.11.0/24 area 0.0.0.1
        end
```

In OSPF, areas are organised in a two-level hierarchy. At the top we have *area 0*, which is referred to as the *backbone area*. As the hierarchy is limited to two levels, every ABR must be connected to the backbone area. Direct connections between areas at lower level is prohibited; all inter-area traffic should go via the backbone area[2].

To allow for a more flexible area hierarchy, OSPF provides a feature referred to as *virtual links*, however, OSPF virtual links are not supported in WeOS v4.3.0.

20.1.1.3 Route redistribution and default route

Route information learnt from other routing protocols (RIP, BGP[3], etc.) *can* be redistributed (i.e., imported) into the OSPF domain. The same goes for static routes, and directly connected networks.

To let a router redistribute routing information into the OSPF domain, the **”redistribute”** command is used, e.g., **”redistribute rip”** to import routes learnt via RIP. An OSPF router performing route distribution into the OSPF domain is referred to as an administrative system border router (ASBR).

Routers can inject a default route (0.0.0.0/0) into the OSPF domain. This is done using the **”distribute-default [always]”** command. Without the **”always”** keyword, the router will only inject the default route if it itself has a default route.

External routes can be added at two levels, *type 1* and *type 2* external routes:

- *Type 1:* Type 1 external routes are typically used when importing routes, that are locally managed, e.g., a static routes inside your domain, or from a local RIP domain.

  The ASBR located in area 0.0.0.2 in fig. 20.4 would preferably redistribute the routes learnt via RIP as *type 1* external routes.

[2]The reason for introducing these topology limitations is to avoid the ”counting to infinity” seen in *distance vector* protocols (see chapter 21) problem to occur for OSPF inter-area routing.)

[3]As of WeOS v4.3.0 BGP is not supported.

- *Type 2:* Type 2 external routes are typically used when importing routes managed by another operator, e.g., routes learnt via BGP.

  The ASBRs located in area 0.0.0.0 in fig. 20.4 would preferably redistribute the routes learnt via BGP as *type 2* external routes.

20.1.1.4 Stub areas and totally stubby areas

In some situations one wish to limit the routing information going into an area to be limited even further, perhaps due to limited resources on the router. For this situation, OSPF provides a special area type referred to as a *stub area*.

As with other OSPF routers, routers inside a stub area will have full routing information for networks and routers within their own area and summary routes to destinations in other areas, *but* need not keep routing information learnt from *external* sources (static routes, or routes learnt via other routing protocols such as RIP, BGP, etc.). In a stub area, routing to networks outside the OSPF domain is instead based on *default routing* towards the ABR(s); i.e., the ABR will filter out all external routing information and instead inject a default route (pointing to itself) area.

To create a *stub* area, **all routers** in the area (ABRs as well as internal routers) must declare the area as stub. An example is given below.

```
router
        ospf
                router-id 192.168.5.11
                network 192.168.5.0/24 area 0.0.0.0
                network 192.168.11.0/24 area 0.0.0.1
                area 0.0.0.1
                        stub
                end

        end
```

To reduce the routing information going into a stub area even further, it is possible to prohibit *summary* routes from other areas to go into a stub area. This is done by adding the *no-summary* parameter to the stub command (**”stub no-summary”**); this is only needed on the ABR(s) of the stub area.

Such areas are referred to as *totally stubby* areas.

The cost of the default route being injected into the stub area is by default set to ”1”. The cost value can be configured via the **”default-cost”** command

within the area context.

The backbone area cannot be configured as a stub area.

20.1.1.5 Not so stubby areas (NSSAs)

In a stub area, no router can redistribute routing information learnt from external sources (static routes, BGP, etc.). That is, a stub area cannot contain an *autonomous system border router* (ASBR).

If you wish to have an ASBR in an area, but limit the amount of routing information to keep track of as in a stub area, OSPF provides an area type known as *not so stubby area* (NSSA).

Fig. 20.4 demonstrates a case where NSSAs can be a useful choice. Here we assume that area 0.0.0.1 and area 0.0.0.2 are preferably defined as *stub areas* to avoid that BGP routes (redistributed by the ASBRs in the backbone area) are propagated into those areas. But area 0.0.0.2 includes a router connected to a local RIP network. By defining area 0.0.0.2 as a NSSA, the RIP routes can be redistributed into the OSPF network.

NSSA are created in the same way as a *stub* area (see section 20.1.1.4). **All routers** in the area must declare the area as NSSA. An example is given below.

```
router
        ospf
                router-id 192.168.5.12
                network 192.168.5.0/24 area 0.0.0.0
                network 192.168.16.0/24 area 0.0.0.2
                area 0.0.0.2
                        nssa
                end
        end
```

As with stub areas, NSSAs are able to prohibit inter-area routing information to be distributed inside the area (use **"nssa no-summary"** on the ABRs of the area). Such areas are called *NSSA totally stub areas*.

The backbone area cannot be configured as a NSSA area.

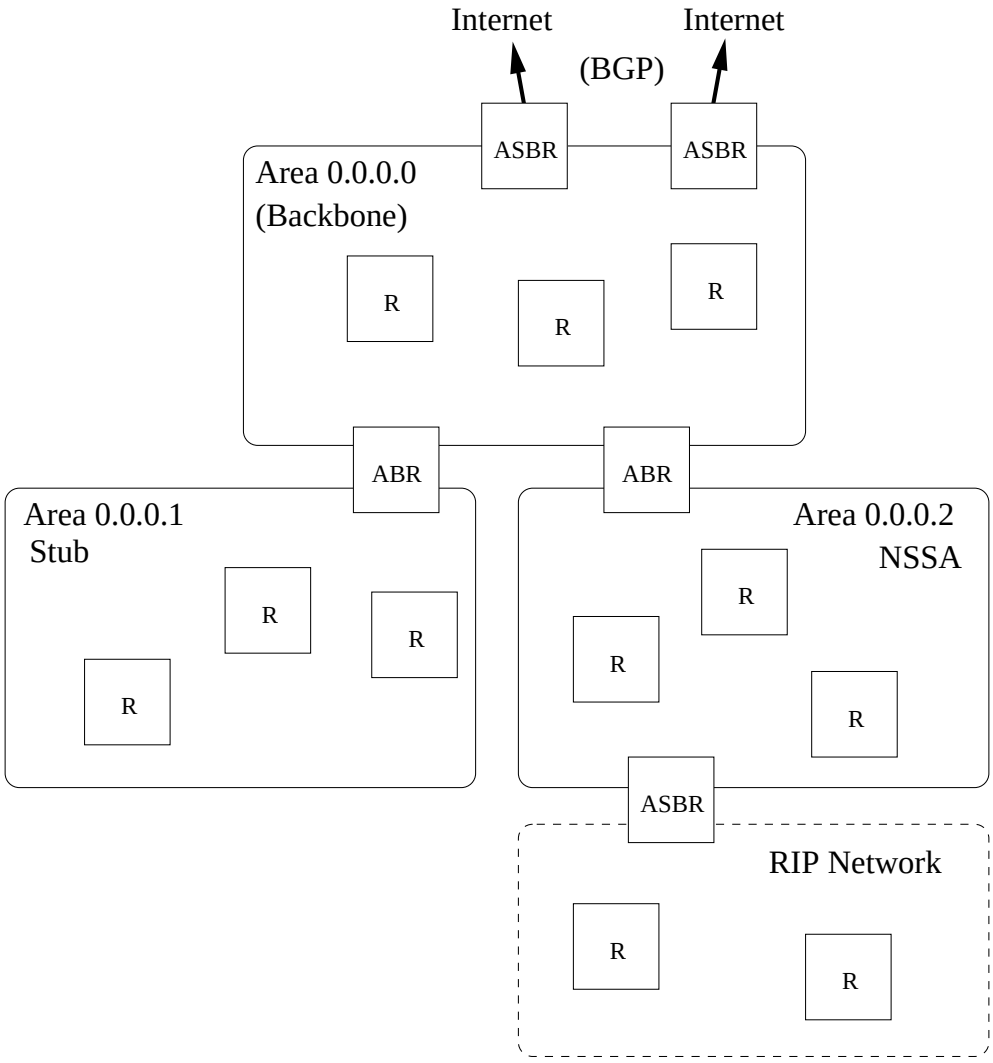


Figure 20.4: Topology where NSSA areas are useful.

20.1.1.6 Additional Area Specific Settings

ABRs are able to filter and to aggregate routing information before distributing it into another area. This is managed using the **"range <NETWORK/LEN> [not-advertise]"** command.

- *Route filtering:* With the **"not-advertise"** keyword, any route matching the given range will be filtered out when distributing routing information outside

a certain area.

- *Route summarisation:* Without the **"not-advertise"** keyword, all routes matching the given range will be summarised (aggregated) as a single destination (of given network and prefix length) outside of a certain area.

Below is and example where an ABR will filter out routes in *192.168.16.0/20* when distributing routes from *area 0.0.0.2*. Similarly, all routes inside *area 0.0.0.2* matching *172.16.0.0/16* will be summarised to single route, when distributing routes from *area 0.0.0.2*.

```
router
        ospf
                router-id 192.168.5.12
                network 192.168.5.0/24 area 0.0.0.0
                network 192.168.16.0/24 area 0.0.0.2
                network 192.168.19.0/24 area 0.0.0.2
                area 0.0.0.2
                        range 192.168.16.0/20 not-advertise
                        range 172.16.0.0/16
                end
        end
```

20.1.1.7 Passive Interfaces

In some situations you may wish to include a router's subnets as part of the OSPF routing domain without running OSPF on the associated network interface. To accomplish this the *network* should be defined in the *router ospf* context (as usual), and the related interface should be declared as *passive* in the *interface ospf* context. Below is an example where network *192.168.33.0/24* should be included in the OSPF domain, but where the associated interface (*vlan100*) is declared as passive.

```
iface vlan100 inet static
        ...
        ... Skipping lines
        ...
        address 192.168.33.1/24
        ospf
                passive
                end
        end
```

```
router
        ospf
                router-id 192.168.15.1
                network 192.168.15.0/24 area 0.0.0.0
                network 192.168.33.0/24 area 0.0.0.0
                end
        end
```

By default, OSPF will run on all interfaces which have an associated network declared as an OSPF network. If OSPF should *not* run on such an interface, that interface should be declared as passive, as described above. However, WeOS is able to support use cases where the interfaces should be passive by default. The parameters controlling the behaviour are the **"passive-interface"** setting in *router ospf* context, and the **"passive"** setting in the *interface ospf* context.

- *passive-interface:* Use the **"[no] passive-interface"** setting in *router ospf* context to control whether interfaces should be passive in OSPF by default or not. Default setting: Active (**"no passive-interface"**)
- *passive:* Use the **"[no] passive [auto]"** setting in *interface ospf* context to control whether a specific interface should be passive (**"passive"**), active (**"no passive"**), or to automatically follow (**"passive auto"**) the global OSPF setting declared by the **"[no] passive-interface"** setting in *router ospf* context. Default: Auto (**"passive auto"**)

Below is an example, with the same result as above, where interfaces are passive in OSPF by default.

```
iface vlan110 inet static
        ...
        ... Skipping lines
        ...
        address 192.168.15.1/24
        ospf
                no passive
                end
        end

router
        ospf
                router-id 192.168.15.1
                passive-interface
```

```
            network 192.168.15.0/24 area 0.0.0.0
            network 192.168.33.0/24 area 0.0.0.0
            end
    end
```

20.1.1.8 OSPF security

If an ”external” OSPF router happens to connect to your network (maliciously or by mistake) the routing inside your domain can be affected severely. E.g., if that router injects a default route into the OSPF domain, all traffic supposed to go to your Internet gateway may instead be routed towards this ”foreign” router.

To avoid that this happens, it is good practise to enable authentication of all OSPF messages inside your network. WeOS provides to forms of authentication of OSPF messages:

- *Plain:* Plain text authentication will protect against the situation when careless users attach an OSPF router to your network *by mistake*. However, since the password is sent in plain text inside the OSPF messages, it does not prohibit a deliberate attacker to inject routing information into your network. Plain text secrets are text strings of 4-8 characters.
- *MD5:* With MD5 authentication each OSPF message will include a cryptographic checksum, i.e., message authentication code (MAC), based on a secret only known by the system administrator. MD5 secrets are text strings of 4-16 characters.

Authentication of OSPF messages is configured per network interface, and is disabled by default.

Use of MD5 authentication is recommended. When using MD5 authentication, an associated *key identifier* must be specified. The purpose of the *key identifier* is to enable use of multiple MD5 keys in parallel when performing *key roll-over*. However, as of WeOS version v4.3.0 only a single OSPF secret per interface is supported.

> **Warning**: *Configuring OSPF authentication remotely in an operational network can be dangerous, since the communication towards that router can be broken if the neighbour routers do not yet have the corresponding authentication configuration. In this case it is good practice to always have a redundant routing path to the router you are configuring.*

If the you end up in the situation where you can no longer reach a router due to a change in OSPF authentication configuration, you may be able to solve the situation by first logging into a "neighbour" of the "unreachable router", and from that router use SSH (see section 7.3.14) to login to the "unreachable router", and then update the configuration appropriately.

20.1.1.9 Finding OSPF Neighbours

OSPF routers will periodically transmit OSPF *Hello* messages, and routers can thereby discover new neighbour routers, and also detect if a neighbour router is down. There two parameter settings related to the OSPF hello messages. These settings are configured per interface.

- *Hello-interval:* The interval (in seconds) at which this router is transmitting Hello messages. Default: 10 seconds
- *Dead-interval:* The interval (in seconds) after which a neighbour router is considered down if no Hello message from that router is received[4]. Default: 40 seconds

---

Note: *All routers attached to a link must have identical "hello-interval" and "dead-interval" settings. That is, an OSPF router will only accept incoming Hello messages with identical hello and dead interval values as the router itself is using on that interface.*

20.1.1.10 Designated OSPF router

In shared networks, such as Ethernets, there may be several routers attached to the same LAN. Representing a LAN as a full mesh of links between the attached routers may grow the OSPF database substantially if the number routers is large. Instead, link state protocols, such as OSPF, treats a shared link as a logical star, with a *virtual node* in the middle representing the shared network, see 20.5. The router which takes the role of network is referred to as the *designated router*.

The designated router (DR), as well as a backup designated router (BDR), are elected automatically. If no node has been elected as DR or BDR, the router with the highest configured DR election *priority* becomes the DR, using the *router-id* as tie-breaker when more than one router has highest priority.

[4] If the interface towards that neighbour goes down (e.g., if (all) the Ethernet port(s) associated with that interface goes down), the router will react immediately instead of waiting for the *dead-interval* to expire.

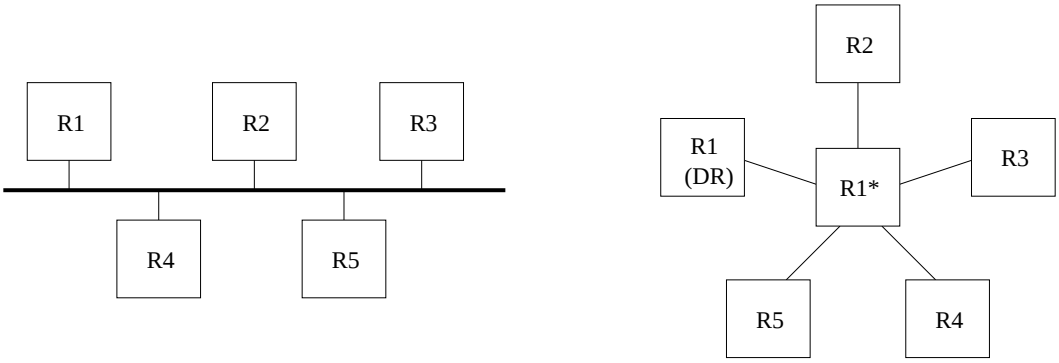


Figure 20.5: Link state protocols such as OSPF logically represent a shared link (a) as a star (b). One of the attached routers (here R1), will take the role as *designated router* and represent the ”network” in the middle.

OSPF implements a *sticky* DR election scheme. Once a router has become DR, it will keep that role even when a router with higher DR priority comes up. However, a DR will give up its role if it discovers another router, which also consider itself to be DR, *and* if that router has higher priority (with router-id as tie). Such a situation could occur if a segmented LAN becomes connected.

20.2 Managing OSPF via the CLI

The table below shows OSPF management features available via the CLI.

Command	Default	Section
Configure General OSPF Settings		
router		
[no] ospf	Disabled	Sec. 20.2.1
[no] router-id <ROUTERID>	Auto	Sec. 20.2.2
[no] network <NETWORK/LEN> [area <AREAID>]	area 0	Sec. 20.2.3
[no] passive-interface	Active	Sec. 20.2.4
[no] distribute-default [always] [metric-type <1|2>] [metric <0-16777214>]	Disabled	Sec. 20.2.5
[no] redistribute connected [metric-type <1|2>] [metric <0-16777214>]	Disabled	Sec. 20.2.6
[no] redistribute static [metric-type <1|2>] [metric <0-16777214>]	Disabled	Sec. 20.2.6
[no] redistribute rip [metric-type <1|2>] [metric <0-16777214>]	Disabled	Sec. 20.2.6
[no] area <AREAID>		Sec. 20.2.7
[no] stub [no-summary]	Disabled	Sec. 20.2.8
[no] nssa [no-summary]	Disabled	Sec. 20.2.9
[no] default-cost <0-16777215>	0	Sec. 20.2.10
[no] range <NETWORK/LEN> [<advertise|not-advertise>]	advertise	Sec. 20.2.11
View General OSPF Settings		
router		
show ospf		Sec. 20.2.12
ospf		
show router-id		Sec. 20.2.13
show network		Sec. 20.2.14
show passive-interface		Sec. 20.2.15
show distribute-default		Sec. 20.2.16
show redistribute [<connected|static|rip>]		Sec. 20.2.17

Continued on next page.

Continued from previous page.

Command	Default	Section
View General OSPF Settings (cont.)		
router		
ospf		
show area [<AREAID>]		Sec. 20.2.18
area <AREAID>		
show stub		Sec. 20.2.19
show nssa		Sec. 20.2.20
show default-cost		Sec. 20.2.21
show range		Sec. 20.2.22
Configure Interface Specific OSPF Settings		
interface <IFACE>		
[no] ospf		Sec. 20.2.23
[no] passive [auto]	Auto	Sec. 20.2.24
[no] cost <1-65535>	10	Sec. 20.2.25
[no] hello-interval <1-65535>	10	Sec. 20.2.26
[no] dead-interval <1-65535>	40	Sec. 20.2.27
[no] auth <md5 [KEYID] | plain> <SECRET>	Disabled	Sec. 20.2.28
[no] priority <0-255>	1	Sec. 20.2.29
View Interface Specific OSPF Settings		
interface <IFACE>		
show ospf		Sec. 20.2.30
ospf		
show passive		Sec. 20.2.31
show cost		Sec. 20.2.32
show hello-interval		Sec. 20.2.33
show dead-interval		Sec. 20.2.34
show auth		Sec. 20.2.35
show priority		Sec. 20.2.36
View OSPF Status		
show ip ospf		Sec. 20.2.37
show ip ospf route		Sec. 20.2.38
show ip ospf neighbor [<IFACE | detail>]		Sec. 20.2.39
show ip ospf database [asbr-summary|external| network|router|summary>		Sec. 20.2.40
show ip ospf database max-age		Sec. 20.2.40
show ip ospf database self-originate		Sec. 20.2.40

20.2.1 Activate OSPF and Manage General OSPF Settings

Syntax `[no] ospf`

Context *Router* context

Usage Enter the router OSPF configuration context, and *activate* OSPF with default settings if OSPF is not activated already. Instead of running **"ospf"** from the *Router* context, you can use **"router ospf"** directly from the Global Configuration

Use **"no ospf"** to disable OSPF and delete all existing OSPF configuration.

Default values Disabled (`no ospf`)

Error messages None defined yet.

20.2.2 Configure OSPF Router-ID

Syntax `[no] router-id <ROUTER-ID>`

Context *OSPF* context

Usage Set the OSPF router identifier, which must be unique within your OSPF domain. The router ID is a 32-bit value, and is given in a dotted 1decimal form <a.b.c.d> (where a-d are numbers in the range 0-255), or as an integer ($0..2^{32}-1$). Commonly the router ID is set equal to one of the router's IP addresses.

In *Auto* mode, the router ID is *automatically* set to the IP address of one of the router's interface (the highest IP address), and stick to that value until the OSPF process is restarted.

Default values Auto (`no router-id`)

Error messages None defined yet.

20.2.3 Enable OSPF on an Interface

Syntax `[no] network <NETWORK/LEN> [area <AREAID>`

Context *OSPF* context

Usage Enable OSPF on the router interface with the specified IP subnet (NETWORK/LEN), include that IP subnet in the OSPF routing domain, and determine the associated OSPF area.

The area ID is a 32-bit number, and is entered in dotted decimal form, or as an integer ($0..2^{32}-1$). By default, the backbone area (0.0.0.0) is assumed.

Use **"no network <NETWORK/LEN> [area <AREAID>]"** to delete a configured **"network"** entry.

Default values Disabled, i.e., no **”network”** entries exist when first activating OSPF (see section 20.2.2).

Error messages None defined yet.

20.2.4 Configure Interface Default Active/Passive Setting

Syntax `[no] passive-interface`

Context *OSPF* context

Usage Define whether OSPF should be run on the interfaces defined (implicitly) via the OSPF **”network”** command (see section 20.2.3).

If the setting is **”no passive-interface”**, the interfaces associated with the **”network”** command will automatically run OSPF, unless OSPF is explicitly disabled on the interface (see the **”passive”** command in section 20.2.24).

Similarly, if the setting is **”passive-interface”**, the interfaces associated with the **”network”** command will not run OSPF, unless OSPF is explicitly enabled on the interface (see the **”no passive”** command in section 20.2.24).

Default values Active (**”no passive-interface”**)

Error messages None defined yet.

20.2.5 Configure Distribution of Default Route into OSPF Domain

Syntax `[no] distribute-default [always] [metric-type <1|2>] [metric <0-16777214>]`

Context *OSPF* context

Usage Inject a default route into the OSPF domain, i.e., announce that this router can reach *network 0.0.0.0/0*.

Use the **”always”** keyword to make the router always advertise the default route, regardless if it has one or not. Without the "always" keyword, it will only advertise if it has one.

Default values Disabled (**”no distribute-default”**)

Error messages None defined yet.

20.2.6 Configure Redistribution of External Route Information into OSPF Domain

Syntax `[no] redistribute <connected|static|rip> [metric-type <1|2>] [metric <0-16777214>]`

Context *OSPF* context

Usage Import external routing information into the OSPF domain. Redistribution of connected routes, static routes, and routes learnt via RIP is handled independently, e.g., use **”redistribute rip”** to import routes learnt via RIP.

Use **”no redistribute”** to remove all redistribution, and **”no redistribute rip”** to remove redistribution of routes learnt via RIP, etc.

Default values Disabled (**”no redistribute”**)

Error messages None defined yet.

20.2.7 Manage area specific settings

Syntax `[no] area <AREAID>`

Context *OSPF* context

Usage Enter the area context of the specified *AREAID* to configure area specific settings, such as area type (regular, stub, nssa), inter-area route summarisation, etc.

Use **”no area <AREAID>”** to remove specific for a single area, and **”no area”** to remove specific settings for all areas.

Default values Disabled (**”no area”**)

Error messages None defined yet.

20.2.8 Configure an Area as Stub

Syntax `[no] stub [no-summary]`

Context *OSPF Area* context

Usage Configure an area as a *stub* area. To create a *stub* area, **all routers** in the area (ABRs as well as internal routers) must declare the area as stub.

To configure the area as a *totally stubby area*, all ABRs in the area should add the *no-summary* parameter to the stub command (**”stub no-summary”**.

Use **”no stub”** to let a stub (or nssa) area become a *regular* area.

Default values Disabled (i.e., areas are ”regular” OSPF areas by default)

Error messages None defined yet.

20.2.9 Configure an Area as NSSA

Syntax `[no] nssa [no-summary]`

Context *OSPF Area* context

Usage Configure an area as a *nssa* area. To create a *nssa* area, **all routers** in the area (ABRs as well as internal routers) must declare the area as nssa.

To configure the area as a *NSSA totally stub area*, all ABRs in the area should add the *no-summary* parameter to the nssa command (**"nssa no-summary"**.

Use **"no nssa"** to let a nssa (or stub) area become a *regular* area.

Default values Disabled (i.e., areas are "regular" OSPF areas by default)

Error messages None defined yet.

20.2.10 Configure default route cost in stub and NSSA areas

Syntax `[no] default-cost`

Context *OSPF Area* context

Usage Configure the cost of the default route injected into a stub area. This setting only applies to the ABRs of a stub or NSSA area.

Use **"no default-cost"** to use the *default* value for the *default cost* setting.

Default values **"default-cost 0"**

Error messages None defined yet.

20.2.11 Configure inter-area route summarisation and filtering

Syntax `[no] range <NETWORK/LEN> [<advertise|not-advertise]`

Context *OSPF Area* context

Usage Configure inter-area route *summarisation* or route *filtering*.

Use the **"range <NETWORK/LEN>"** (**"range <NETWORK/LEN> advertise"** is equivalent) to aggregate routes (within this area) matching the specified <NETWORK/LEN> range, before distributing the routes outside this area. That is, all routes within this range are *summarised* as a single route, when advertised outside this area.

Use the **"range <NETWORK/LEN> not-advertise"** to prohibit routes (within this area) matching the specified <NETWORK/LEN> range, to be distributed outside this area. That is, routes within this range are *filtered*.

Use **"no range <NETWORK/LEN>"** to remove a specific summary/filter setting, or **"no range"** to remove all summary/filter settings for this area.

Default values Disabled

Error messages None defined yet.

20.2.12 Show All General OSPF Settings

Syntax `show ospf`

Context *Router* context. Also available as **"show"** command within the *OSPF* context.

Usage Show a summary of all general OSPF settings.

Default values Not applicable

20.2.13 Show OSPF Router-ID Setting

Syntax `show router-id`

Context *OSPF* context.

Usage Show the router-ID setting.

Default values Not applicable

20.2.14 Show OSPF Network Settings

Syntax `show network`

Context *OSPF* context.

Usage Show the OSPF network settings.

Default values Not applicable

20.2.15 Show OSPF Passive Default Settings

Syntax `show passive-interface`

Context *OSPF* context.

Usage Show the default behaviour of OSPF interfaces (passive or active).

Default values Not applicable

20.2.16 Show OSPF Distribute Default Route Setting

Syntax `show distribute-default`

Context *OSPF* context.

Usage Show the whether this router is configured to inject a default route into the OSPF domain.

Default values Not applicable

20.2.17 Show OSPF Redistribute Settings

Syntax `show redistribute [<connected|static|rip>]`

Context *OSPF* context.

Usage Show the OSPF redistribution settings. Use **"show redistribute"** to show all redistribution settings, ore**"show redistribute connected"**, etc., to show redistribute settings for specific types of redistribution.

Default values Not applicable

20.2.18 Show Summary of Area Specific Settings

Syntax `show area [<AREAID>]>]`

Context *OSPF* context. (Also available as **"show"** command within the *OSPF Area* context.)

Usage Show a summary of area specific settings. Use **"show area"** to show settings for all areas, and **"show area <AREAID>"** to show settings for a specific area.

Default values All areas (if no AREAID is specified, area specific settings for all areas will be displayed.)

20.2.19 Show Stub Area Settings

Syntax `show stub`

Context *OSPF Area* context.

Usage Show whether this area is configured as *stub* or not. If this is a stub area, it will show whether the **"no-summary"** keyword is set or not, i.e., if it is a *totally stubby* area or just a *stub* area.

Default values Not applicable.

20.2.20 Show NSSA Area Settings

Syntax `show nssa`

Context *OSPF Area* context.

Usage Show whether this area is configured as *NSSA* or not. If this is a NSSA area, it will show whether the **"no-summary"** keyword is set or not, i.e., if it is a *NSSA totally stub* area or just a *NSSA* area.

Default values Not applicable.

20.2.21 Show Stub/NSSA Default Cost Setting

Syntax `show default-cost`

Context *OSPF Area* context.

Usage Show the setting of the default-cost, i.e., the cost of the default route injected by ABRs into a stub or NSSA area.

Default values Not applicable.

20.2.22 Show Area Summarise and Filtering Settings

Syntax `show range`

Context *OSPF Area* context.

Usage Show configured route summarisation and route filtering settings for this area.

Default values Not applicable.

20.2.23 Manage Interface Specific OSPF Settings

Syntax `[no] ospf`

Context *Interface* context

Usage Enter the Interface OSPF configuration context, i.e., the context where Interface specific OSPF settings are configured.

Use **"no ospf"** to remove any specific OSPF settings for this interface. .

Default values Disabled (i.e., no interface specific OSPF settings)

Error messages None defined yet.

20.2.24 Configure Interface OSPF Passive Settings

Syntax `[no] passive [auto]`

Context *Interface OSPF* context

Usage Control whether a specific interface should be passive (**"passive"**), active (**"no passive"**), or to automatically follow (**"passive auto"**) the global OSPF setting declared by the **"[no] passive-interface"** setting in *router ospf* context (see section 20.2.4).

Default values Auto (**"passive auto"**)

Error messages None defined yet.

20.2.25 Configure Interface OSPF Cost Settings

Syntax `[no] cost <1-65535>`

Context *Interface OSPF* context

Usage Configure interface OSPF cost.
Use **"no cost"** to return to the default setting.

> **Note**: *As of WeOS v4.3.0 only static configuration of the interface OSPF cost setting is available. Support to let the cost automatically depend on the interface data rate is planned, but not yet implemented.*

Default values 10 (this may be subject to change in later versions of WeOS.

Error messages None defined yet.

20.2.26 Configure Interface OSPF Hello Interval Settings

Syntax `[no] hello-interval <1-65535>`

Context *Interface OSPF* context

Usage Configure OSPF hello interval (in seconds) for this interface.
Use **"no hello-interval"** to return to the default setting.

> **Note**: *The hello interval setting must be the same on neighbour routers.*

Default values 10 (seconds)

Error messages None defined yet.

20.2.27 Configure Interface OSPF Dead Interval Settings

Syntax `[no] dead-interval <1-65535>`

Context *Interface OSPF* context

Usage Configure OSPF dead interval (in seconds) for this interface.
Use **"no dead-interval"** to return to the default setting.

> **Note**: *The dead interval setting must be the same on neighbour routers.*

Default values 40 (seconds)

Error messages None defined yet.

20.2.28 Configure Authentication of OSPF Messages

Syntax `[no] auth <md5 [KEYID] | plain> <SECRET>`

Context *Interface OSPF* context

Usage Configure authentication of OSPF messages *on this interface*. Two authentication methods are available:

- *MD5:* Use **"auth md5 <KEYID> <SECRET>"** to use a MD5 cryptographic authentication. MD5 secrets are text strings of 8-16 characters. A key identifier (0-255) is associated with MD5 keys. (Both the secret and the key identifier must be the same on neighbour routers.)
- *Plain:* Use **"auth plain <SECRET>"** to use a clear-text password as authentication. Plain text secrets are text strings of 4-8 characters. (The secret must be the same on neighbour routers.)

Use **"no auth"** to disable authentication of OSPF messages on this interface.

Default values Disabled

Error messages None defined yet.

20.2.29 Configure OSPF Designated Router Priority

Syntax `[no] priority <0-255>`

Context *Interface OSPF* context

Usage Configure the OSPF designated router priority, which affects the chance to become designated router on a broadcast network. A higher value increases the chance to become designated router.

Use **"priority 0"** to state that this router is not eligible as designated router on this interface/"IP subnet".

Use **"no priority"** to return to the default setting.

Default values 1 (**"priority 1"**)

Error messages None defined yet.

20.2.30 Show Summary of Interface OSPF Settings

Syntax `show ospf`

Context *Interface* context. (Also available as **"show"** command within the *Interface OSPF* context.)

Usage Show a summary of OSPF settings for this interface.

Default values Not applicable

20.2.31 Show Passive Interface Setting

Syntax `show passive`

Context *Interface OSPF* context.

Usage Show the OSPF passive interface setting (passive, active or ”auto”) for this interface.

Default values Not applicable

20.2.32 Show Interface OSPF Cost Setting

Syntax `show passive`

Context *Interface OSPF* context.

Usage Show OSPF cost setting for this interface.

Default values Not applicable

20.2.33 Show Interface OSPF Hello Interval Setting

Syntax `show hello-interval`

Context *Interface OSPF* context.

Usage Show the OSPF hello interval setting for this interface.

Default values Not applicable

20.2.34 Show Interface OSPF Dead Interval Setting

Syntax `show dead-interval`

Context *Interface OSPF* context.

Usage Show the OSPF dead interval setting for this interface.

Default values Not applicable

20.2.35 Show Interface OSPF Authentication Setting

Syntax `show auth`

Context *Interface OSPF* context.

Usage Show the OSPF authentication setting for this interface.

Default values Not applicable

20.2.36 Show Interface OSPF DR Priority Setting

Syntax `show auth`

Context *Interface OSPF* context.

Usage Show the OSPF designated router election priority setting for this interface.

Default values Not applicable

20.2.37 Show General OSPF Status

Syntax `show ip ospf`

Context *Admin Exec* context.

Usage Show general OSPF status information.

Default values Not applicable

20.2.38 Show OSPF Routes

Syntax `show ip ospf route`

Context *Admin Exec* context.

Usage Show the current least-cost routes learnt via OSPF. See also the command **"show ip route"** (section 17.4.29), which displays the full forwarding/routing table.

Default values Not applicable

20.2.39 Show OSPF Neighbours

Syntax `show ip ospf neighbor [<IFACE | detail>]`

Context *Admin Exec* context.

Usage Show current list of OSPF neighbours. Use **"show ip ospf neighbor IFACE"** to list OSPF neighbours for a specific interface, or the keyword **"detail"** to receive a more detailed listing.

Default values By default, neighbours on all interfaces are listed.

20.2.40 Show OSPF Database

Syntax

```
show ip ospf database [asbr-summary|external|network|router|summary>],
show ip ospf database max-age,
show ip ospf database self-originate
```

Context *Admin Exec* context.

Usage Use **”show ip ospf database”** to list the current OSPF database. Various keywords can be added to view specific parts of the database.

Default values By default, the full database is listed.

Chapter 21

Dynamic Routing with RIP

This chapter describes WeOS support for the Routing Information Protocol (RIP.)

WeOS supports dynamic routing via RIP version 1 (RIPv1) and version 2 (RIPv2). RIP is relatively simple to setup, but does not handle topology changes as rapidly as the OSPF dynamic routing protocol (support for OSPF is described in chapter 20). Therefore, OSPF is generally preferred over RIP when it is possible to select dynamic routing protocol.

21.1 Overview of RIP Support in WeOS

Feature	CLI	General Description
RIP version	X	Sec. 21.1.1
RIP Networks/Interfaces	X	-”-
RIP Neighbour	X	-”-
Redistribution (static, connected, OSPF)	X	Sec. 21.1.2
Distribute Default Route	X	-”-
Authentication (MD5, plain)	X	Sec. 21.1.3
Passive interface	X	Sec. 21.1.4

Table 21.1: Summary of RIP features.

21.1.1 Introduction to RIP

RIP is an example of a *distance vector* routing protocol, and historically it has been one of the most widely used *intra-domain* unicast routing protocol within

the Internet.

RIP is quite simple to configure; commonly you only have to enable RIP and define which interfaces to run RIP on. The router will automatically discover its neighbours and start to exchange routing information.

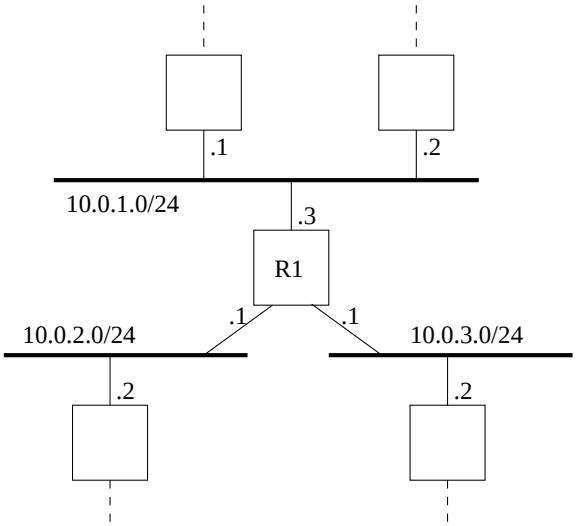


Figure 21.1: A router (R1) connected to other routers via three interfaces.

To enable RIP on all interfaces on R1 in fig. 21.1, configuration shown below would suffice.

```
router
        rip
                network 10.0.1.0/24
                network 10.0.2.0/24
                network 10.0.3.0/24
                end
        end
```

The command **”network 10.0.1.0/24”** will enable RIP on all interfaces included within the given range; in this example it states that RIP should be activated on the ”upper interface” (i.e., the interface with address 10.0.1.3/24). It is also possible to specify the interfaces explicitly; assuming the three interfaces of R1 are called *vlan1*, *vlan2*, and *vlan3*, the following configuration would give the same result:

```
router
```

```
        rip
                network vlan1
                network vlan2
                network vlan3
                end
        end
```

Both RIPv1[3] and RIPv2[7] are supported, and RIPv2 is used by default when RIP is enabled. The major difference between RIPv1 and RIPv2 is that RIPv2 supports flexible subnet masks (CIDR - classless inter-domain routing), while RIPv1 assumes that IP subnet masks follow the (deprecated) classful addressing scheme (class A, B and C). In addition, RIPv2 supports message authentication (section 21.1.3, and can therefore offer protection in situations when ”foreign RIP routers” are connected (by mistake or as a deliberate attack) to a network and inject RIP routing messages. Thus, use of RIPv2 is preferred over RIPv1, except for cases where legacy equipment require the use of RIPv1.

RIPv2 routers exchange routing information using IP multicast (IP address 224.0.0.9)[1]. In case a neighbour router is unable to handle IP multicast, the **”neighbor”** command enables the exchange of RIP messages using regular IP unicast.

21.1.2 Redistribution and Injection of Default Route

It is possible to redistribute routing information learnt externally (OSPF, connected routes or static routes) inside the RIP routing domain, using the **”redistribute”** command.

You can also let a RIP router inject a default route (0.0.0.0/0) into your RIP domain, using the **”distribute-default”**.

21.1.3 Authentication

To avoid that false routing information is injected into your network (deliberately or by mistake) it is possible to authenticate RIPv2 messages. Two authentication alternatives are available:

- *Plain:* Plain text authentication will protect against the situation when careless users attach a RIP router to your network *by mistake*. However, since the password is sent in plain text inside the RIP messages, it does not prohibit a deliberate attacker to inject routing information into your network. Plain text secrets are text strings of 4-16 characters.

[1] While RIPv2 use IP multicast, RIPv1 exchange routing information using broadcast.

- *MD5:* With MD5 authentication each RIP message will include a cryptographic checksum, i.e., message authentication code (MAC), based on a secret only known by the system administrator. MD5 secrets are text strings of 4-32 characters.

Authentication of RIP messages is configured per network interface, and is disabled by default.

Use of MD5 authentication is recommended. When using MD5 authentication, an associated *key identifier* must be specified. The purpose of the *key identifier* is to enable use of multiple MD5 keys in parallel when performing *key roll-over*. However, as of WeOS version v4.3.0 only a single RIP secret per interface is supported.

21.1.4 Passive interface

In some situations you may wish to include a router's subnets as part of the RIP routing domain without running RIP on the associated network interface. To accomplish this the *network* should be defined in the *router rip* context (as usual), and the related interface should be declared as *passive* in the *interface rip* context. Below is an example where network *10.0.3.0/24* should be included in the RIP domain, but where the associated interface (*vlan3*) is declared as passive.

```
iface vlan3 inet static
        ...
        ... Skipping lines
        ...
        address 10.0.3.1/24
        rip
                passive
                end
        end

router
        rip
                network 10.0.1.0/24
                network 10.0.2.0/24
                network 10.0.3.0/24
                end
        end
```

By default, RIP will run on all interfaces which have an associated network declared as a RIP network. If RIP should *not* run on such an interface, that interface should be declared as passive, as described above. However, WeOS is able to support use cases where the interfaces should be passive by default. The parameters controlling the behaviour are the **"passive-interface"** setting in *router rip* context, and the **"passive"** setting in the *interface rip* context.

- *passive-interface:* Use the **"[no] passive-interface"** setting in *router rip* context to control whether interfaces should be passive in RIP by default or not. Default setting: Active (**"no passive-interface"**)
- *passive:* Use the **"[no] passive [auto]"** setting in *interface rip* context to control whether a specific interface should be passive (**"passive"**), active (**"no passive"**), or to automatically follow (**"passive auto"**) the global RIP setting declared by the **"[no] passive-interface"** setting in *router rip* context. Default: Auto (**"passive auto"**)

Below is an example, with the same result as above, where interfaces are passive in RIP by default.

```
iface vlan1 inet static
        ...
        ... Skipping lines
        ...
        address 10.0.1.3/24
        rip
                no passive
                end
        end

iface vlan2 inet static
        ...
        ... Skipping lines
        ...
        address 10.0.2.1/24
        rip
                no passive
                end
        end

router
        rip
                passive-interface
```

```
                network 10.0.1.0/24
                network 10.0.2.0/24
                network 10.0.3.0/24
                end
        end
```

21.2 Managing RIP via the CLI

Command	Default	Section
Configure General RIP Settings		
router		
[no] rip	Disabled	Sec. 21.2.1
[no] version <1|2>	version 2	Sec. 21.2.2
[no] network <NETWORK | IFACE>		Sec. 21.2.3
[no] neighbor <ADDRESSLIST>		Sec. 21.2.4
[no] passive-interface	Active	Sec. 21.2.5
[no] distribute-default	Disabled	Sec. 21.2.6
[no] redistribute connected	Disabled	Sec. 21.2.7
[no] redistribute static	Disabled	Sec. 21.2.7
[no] redistribute ospf	Disabled	Sec. 21.2.7
View General RIP Settings		
router		
show rip		Sec. 21.2.8
rip		
show version		Sec. 21.2.9
show network		Sec. 21.2.10
show neighbor		Sec. 21.2.11
show passive-interface		Sec. 21.2.12
show distribute-default		Sec. 21.2.10
show redistribute [<connected|static|ospf>]		Sec. 21.2.14
Configure Interface Specific RIP Settings		
interface <IFACE>		
[no] rip		Sec. 21.2.15
[no] passive [auto]	Auto	Sec. 21.2.16
[no] split-horizon	Enabled	Sec. 21.2.17
[no] send-version <1,2>	Auto	Sec. 21.2.18
[no] receive-version <1,2>	Auto	Sec. 21.2.19
[no] auth <md5 [keyid] | plain> <SECRET>	Disabled	Sec. 21.2.20

Continued on next page.

Continued from previous page.

Command	Default	Section
View Interface Specific RIP Settings		
interface <IFACE>		
show rip		Sec. 21.2.21
rip		Sec. 21.2.21
show passive		Sec. 21.2.22
show split-horizon		Sec. 21.2.23
show send-version		Sec. 21.2.24
show receive-version		Sec. 21.2.25
show auth		Sec. 21.2.26
View RIP Status		
show ip rip		Sec. 21.2.27

21.2.1 Activate RIP and Manage General RIP Settings

Syntax `[no] rip`

Context *Router* context

Usage Enter the router RIP configuration context, and *activate* RIP with default settings if RIP is not activated already. Instead of running **"`rip`"** from the *Router* context, you can use **"`router rip`"** directly from the Global Configuration

Use **"`no rip`"** to disable RIP and delete all existing RIP configuration.

Default values Disabled (`no rip`)

Error messages None defined yet.

21.2.2 Configure Default RIP Version

Syntax `[no] version <1|2>`

Context *RIP* context

Usage Select what RIP version (1 or 2) to use by default, both with respect to sending and receiving of RIP messages. The setting can be overridden per interface using the **"`receive-version`"** (section 21.2.19) and **"`send-version`"** (section 21.2.19) respectively.

Use **"`no version`"** to return to the default setting.

Default values RIPv2 (`version 2`)

Error messages None defined yet.

21.2.3 Enable RIP on an Interface

Syntax `[no] network <NETWORK/LEN | IFACE>`

Context *RIP* context

Usage Enable RIP on the specified router interface. The interface can be specified either explicitly (**"network <IFACE>"**) or implicitly giving the IP subnet associated with the interface (**"network <NETWORK/LEN>"**).

Use **"no network <IFACE>"** and **"no network <NETWORK/LEN>"** to remove an existing **"network"** entry.

Default values Disabled, i.e., when first activating RIP (section 21.2.1, RIP will not be enabled on any interface.

Error messages None defined yet.

21.2.4 Configure Unicast Neighbor

Syntax `[no] neighbor <ADDRESSLIST>`

Context *RIP* context

Usage Configure one or more RIP neighbor routers explicitly. This is useful in case the neighbor router is unable to handle IP multicast. An **"ADDRESSLIST"** is a comma-separated list of IPv4 address, e.g, **"neighbor 192.168.1.1,192.168.3.2"**. Calling the **"neighbor"** command twice (with arguments "192.168.1.1" and "192.168.3.2" respectively) would be equivalent.

Use **"no neighbor"** to remove all configured neighbours, and **"no neighbor <ADDRESSLIST>"** to remove a specific neighbour settings.

Default values Disabled (No neighbours defined)

Error messages None defined yet.

21.2.5 Configure Interface Default Active/Passive Setting

Syntax `[no] passive-interface`

Context *RIP* context

Usage Define whether RIP should be run on the interfaces defined (implicitly) via the RIP **"network"** command (see section 21.2.3).

If the setting is **"no passive-interface"**, the interfaces associated with the **"network"** command will automatically run RIP, unless RIP is explicitly disabled on the interface (see the **"passive"** command in section 21.2.16).

Similarly, if the setting is **”passive-interface”**, the interfaces associated with the **”network”** command will not run RIP, unless RIP is explicitly enabled on the interface (see the **”no passive”** command in section 21.2.16).

Default values Active (**”no passive-interface”**)

Error messages None defined yet.

21.2.6 Configure Distribution of Default Route into RIP Domain

Syntax `[no] distribute-default`

Context *RIP* context

Usage Inject a default route into the RIP domain, i.e., announce that this router can reach *network 0.0.0.0/0*.

Use **”[no distribute-default”** to stop this router from injecting a default route into the RIP domain.

Default values Disabled (**”no distribute-default”**)

Error messages None defined yet.

21.2.7 Configure Redistribution of External Route Information into RIP Domain

Syntax `[no] redistribute <connected|static|ospf>`

Context *RIP* context

Usage Import external routing information into the RIP domain. Redistribution of connected routes, static routes, and routes learnt via OSPF is handled independently, e.g., use **”redistribute ospf”** to import routes learnt via OSPF.

Use **”no redistribute”** to remove all redistribution, and **”no redistribute ospf”** to remove redistribution of routes learnt via OSPF, etc.

Default values Disabled (**”no redistribute”**)

Error messages None defined yet.

21.2.8 Show All General RIP Settings

Syntax `show rip`

Context *Router* context. Also available as **”show”** command within the *RIP* context.

Usage Show a summary of all general RIP settings.

Default values Not applicable

21.2.9 Show Default RIP Version Setting

Syntax `show version`

Context *RIP* context.

Usage Show the default RIP version setting.

Default values Not applicable

21.2.10 Show RIP Network Settings

Syntax `show network`

Context *RIP* context.

Usage Show the RIP network settings, i.e., which interfaces/subnets that are included in the RIP routing domain.

Default values Not applicable

21.2.11 Show Configured RIP Unicast Neighbours

Syntax `show neighbor`

Context *RIP* context.

Usage Show the configured RIP Unicast Neighbours. active).

Default values Not applicable

21.2.12 Show RIP Passive Default Settings

Syntax `show passive-interface`

Context *RIP* context.

Usage Show the default behaviour of RIP interfaces (passive or active).

Default values Not applicable

21.2.13 Show RIP Distribute Default Route Setting

Syntax `show distribute-default`

Context *RIP* context.

Usage Show the whether this router is configured to inject a default route into the RIP domain.

Default values Not applicable

21.2.14 Show RIP Redistribute Settings

Syntax `show redistribute [<connected|static|rip>]`

Context *RIP* context.

Usage Show the RIP redistribution settings. Use **"show redistribute"** to show all redistribution settings, or **"show redistribute connected"**, etc., to show redistribute settings for specific types of redistribution.

Default values Not applicable

21.2.15 Manage Interface Specific RIP Settings

Syntax `[no] rip`

Context *Interface* context

Usage Enter the Interface RIP configuration context, i.e., the context where Interface specific RIP settings are configured.

Use **"no rip"** to remove any specific RIP settings for this interface. .

Default values Disabled (i.e., no interface specific RIP settings)

Error messages None defined yet.

21.2.16 Configure Interface RIP Passive Settings

Syntax `[no] passive [auto]`

Context *Interface RIP* context

Usage Control whether a specific interface should be passive (**"passive"**), active (**"no passive"**), or to automatically follow (**"passive auto"**) the global RIP setting declared by the **"[no] passive-interface"** setting in *router rip* context (see section 21.2.5).

Default values Auto (**"passive auto"**)

Error messages None defined yet.

21.2.17 Configure Split Horizon Setting

Syntax `[no] split-horizon`

Context *Interface RIP* context

Usage Enable or disable *split horizon* on this interface. Split horizon is a RIP mechanism to mitigate the *counting to infinity* issue appearing in distance vector protocols such as RIP.

Default values Enabled (**”split-horizon”**)

Error messages None defined yet.

21.2.18 Configure RIP Version for Sending on this Interface

Syntax `[no] send-version <1,2>`

Context *Interface RIP* context

Usage Control whether this interface should use the global RIP version setting (section 21.2.2) when sending RIP messages on this interface (**”no send-version”**), or to override the global setting by sending RIPv1 (**”send-version 1”**), RIPv2 (**”send-version 2”**), or both RIPv1 and RIPv2 (**”send-version 1,2”**).

Use **”no send-version”** to remove override settings and return to *auto* setting. (Override can also be removed for individual versions, e.g., **”no send-version 1”** to remove version 1 as override setting.)

Default values Auto (**”no send-version”**)

Error messages None defined yet.

21.2.19 Configure RIP Version for Receiving on this Interface

Syntax `[no] receive-version <1,2>`

Context *Interface RIP* context

Usage Control whether this interface should use the global RIP version setting (section 21.2.2) when accepting incoming RIP messages on this interface (**”no receive-version”**), or to override the global setting by accepting RIPv1 (**”receive-version 1”**), RIPv2 (**”receive-version 2”**), or both RIPv1 and RIPv2 (**”receive-version 1,2”**).

Use **”no receive-version”** to remove override settings and return to *auto* setting. (Override can also be removed for individual versions, e.g., **”no receive-version 1”** to remove version 1 as override setting.)

Default values Auto (**”no receive-version”**)

Error messages None defined yet.

21.2.20 Configure Authentication of RIP Messages

Syntax `[no] auth <md5 [KEYID] | plain> <SECRET>`

Context *Interface RIP* context

Usage Configure authentication of RIP messages *on this interface*. Two authentication methods are available:

- *MD5:* Use **"auth md5 <KEYID> <SECRET>"** to use a MD5 cryptographic authentication. MD5 secrets are text strings of 4-32 characters. A key identifier (0-255) is associated with MD5 keys. (Both the secret and the key identifier must be the same on neighbour routers.)
- *Plain:* Use **"auth plain <SECRET>"** to use a clear-text password as authentication. Plain text secrets are text strings of 4-16 characters. (The secret must be the same on neighbour routers.)

Use **"no auth"** to disable authentication of RIP messages on this interface.

Default values Disabled

Error messages None defined yet.

21.2.21 Show Summary of Interface RIP Settings

Syntax `show rip`

Context *Interface* context. (Also available as **"show"** command within the *Interface RIP* context.)

Usage Show a summary of RIP settings for this interface.

Default values Not applicable

21.2.22 Show Passive Interface Setting

Syntax `show passive`

Context *Interface RIP* context.

Usage Show the RIP passive interface setting (passive, active or "auto") for this interface.

Default values Not applicable

21.2.23 Show Split Horizon Setting

Syntax `show split-horizon`

Context *Interface RIP* context.

Usage Show whether *split horizon* is enabled on this interface or not.

Default values Not applicable

21.2.24 Show Send Version Override Setting

Syntax `show send-version`

Context *Interface RIP* context.

Usage Show RIP version override settings when sending RIP messages on this interface.

Default values Not applicable

21.2.25 Show Receive Version Override Setting

Syntax `show receive-version`

Context *Interface RIP* context.

Usage Show RIP version override settings when accepting incoming RIP messages on this interface.

Default values Not applicable

21.2.26 Show Interface RIP Authentication Setting

Syntax `show auth`

Context *Interface RIP* context.

Usage Show the RIP authentication setting for this interface.

Default values Not applicable

21.2.27 Show RIP Status Information

Syntax `show ip rip` (or simply **`"show rip"`**)

Context *Admin Exec* context.

Usage Show RIP status information, e.g., active interfaces, discovered RIP neighbours, etc.

Default values Not applicable

Chapter 22

Virtual Router Redundancy (VRRP)

This chapter describes WeOS support for the Virtual Router Redundancy Protocol (VRRP)[6]. VRRP is a standard protocol to enable redundancy between a host and its router, in case the router goes down. VRRP can also be used for load balancing purposes.

As of WeOS v4.3.0 management of VRRP settings is only available via the CLI.

22.1 Introduction to WeOS VRRP support

The table below summarises VRRP support in WeOS.

Feature	CLI	General Description
VRRP Instances	X	Sec.22.1.1
Virtual Router IP Address	X	Sec.22.1.1
Virtual Router Priority	X	Sec.22.1.1
Advertisement Interval	X	Sec.22.1.1
Preemption control	X	Sec.22.1.1
Message authentication	X	Sec.22.1.2
Load balancing	X	Sec.22.1.3

22.1.1 VRRP Overview

The primary objective of VRRP is to enable redundancy between a *host* and its next *hop router*, i.e., you can deploy additional routers on an IP subnet as backup routers, and have one of the backup routers to automatically take over if the

primary router fails. Fig. 22.1 can be use to illustrate the need for VRRP in such a scenario.

- A host will typically have an IP setting where the default gateway points to a specific router. An example is given in fig. 22.1a, where the host (H) will send all traffic towards the Internet via Router 1 (R1) with IP address 192.168.1.1. If R1 fails, the host will lose Internet connectivity even though a redundant path (R2) happens to exists.
- VRRP enables routers to share a virtual IP (VIP) address. The router with the highest priority acts as master for the VIP address, while the other routers are backups in case the master fails. Fig. 22.1b illustrates the use of VRRP. R1 and R2 are both responsible for the VIP address (192.168.1.3), with R1 as master since it has higher priority (150>100). If R1 goes down, R2 will become master of the VIP address and communication can automatically resume. Note that the default gateway of the host is configured to the VIP address.

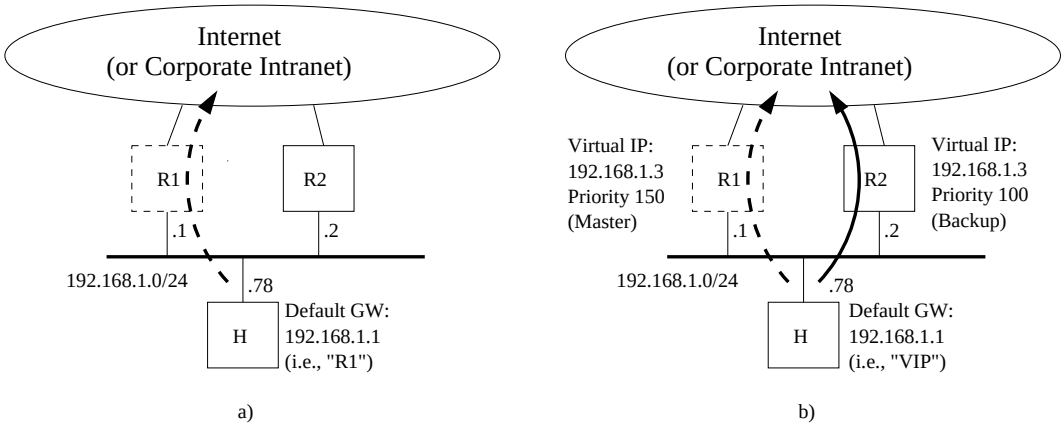


Figure 22.1: Illustrating the need for VRRP to support redundancy: a) Host (H) loses connectivity when Router 1 (R1) fails. b) Host (H) can continue to communicate even though Router 1 (R1) fails, since VRRP enables Router 2 (R2) to take over.

Note: *VRRP enables a host to have redundant routers. For redundancy "router to router", dynamic routing protocols such as OSPF (chapter 20) or RIP (chapter 21) can be used.*

Some common VRRP parameters are listed below:

- *VRRP instance:* WeOS allows you to configure up to four VRRP instances per interface. Each instance is assigned a virtual router instance identifier (VRID), which must be equal on all routers acting as virtual routers for a specific VIP address. That is, R1 and R2 in fig. 22.1b should have the same VRIP, e.g., 33. The VRIP can be in range 0-255.
- *Virtual IP address (VIP):* WeOS allows you to configure one VIP address per VRRP instance. The VIP address should be in the same IP subnet as the regular IP address assigned to the interface (e.g., the VIP address in fig. 22.1b is 192.168.1.3, which is in the same subnet as R1's and R2's IP addresses on that subnet).

  Note: *It is possible use the address assigned to a router as the VIP address, e.g., it would be possible to use 192.168.1.1 as VIP address in fig. 22.1b. If so, the router who is configured with that address is referred to as the* owner *of the VIP address, and should become master by default (priority 255, see below).*
- *Advertisement interval:* In VRRP, the master will announce its presence by sending VRRP Advertisements on a certain interval. The interval can be configured in range 1-255 seconds. All VRRP routers for a VRRP instance group must use the same advertisement interval setting.

  A low VRRP advertisement interval gives faster fail-over (the time to detect that a master is down is roughly 3 times the advertisement interval).

  Default advertisement interval: **1 (second)**
- *VRRP Priority:* The VRRP priority parameter is used to define which router should become master of the VIP address when multiple routers are available. (If two routers with the same priority transitions to master state, the router with the highest IP address will win the election.)

  The priority can be configured in range 1-255, where the value "255" should be used if (and only if) the router is also the *owner* of the VIP address (see the Note above). Default priority: **100**
- *VRRP Preemption:* The VRRP master election is not controlled by the priority setting alone; there is also a *preemption* parameter, which enables you to select to have a deterministic master election procedure (highest priority always becomes master), or if a master router should keep its role even though a router with higher priority later appears on the network. The exception to this if the new router is the VIP address *owner* (priority 255) is connected to the subnet; the VIP owner will always preempt an existing master.

  When preemption is enabled, an optional preemption delay parameter can be configured (default 0 seconds), which determines how long the router

should wait until preemption is activated. Default: **Disabled**

A sample VRRP configuration for R1 in fig. 22.1b is shown below:

```
iface vlan2 inet static
        ...
        address 192.168.1.1/24
        vrrp 33
                address 192.168.1.3
                priority 150
                end
        end
```

22.1.1.1 Limitations

VRRP specifies that the master use a specific (virtual) MAC address together with the virtual IP address[1]. As of WeOS v4.3.0 VRRP routers will instead use the physical MAC address assigned to that interface even when acting as master for a VIP address. To avoid communication disruption when a backup router becomes master, the new master router will notify local hosts about the change of MAC address by sending gratuitous ARP messages. Use of gratuitous ARP is also specified in the VRRP standard[6], however, this mechanism will fail if hosts ignores gratuitous ARP. (Support for VRRP virtual MAC addresses is planned, but not yet implemented.)

The VRRP support available in WeOS v4.3.0 provides good protection against router failure, i.e., if a router goes down. However, if your master router stays up, but ”only” loses its connection to the Internet it would be desirable for the master to lower its priority (or even decline as master) to allow for a backup router to take over. For example, if R1 in fig. 22.1b would lose its upstream connection, it could lower its priority to 30, whereby R2 would could take over if preemption is enabled. (Support for adjusting the VRRP priority depending on the status of upstream connections is planned, but not yet implemented.)

22.1.2 Authentication

WeOS supports a simple form of VRRP message authentication, but enabling the inclusion of a plain-text password in the VRRP advertisements[6]. But use of VRRP authentication is discouraged[4], as it may cause more harm than help.

[1] This virtual MAC address would depend on the identifier (VRID) used for this virtual router (00:00:5E:00:01:<VRID>).

To avoid that multiple master routers appear on an IP subnet, a WeOS VRRP router will refrain from becoming master if it hears another router with mismatching VRRP authentication information.

22.1.3 Load sharing

It is possible to use VRRP for load sharing between routers, and still provide redundancy by having the routers to act as backup for each other. Fig. 22.2 shows a load sharing example. As WeOS currently does not support multi-netting, the VIP addresses used in this example reside within the same IP subnet. (Support for multi-netting in planned but not yet supported in WeOS.)

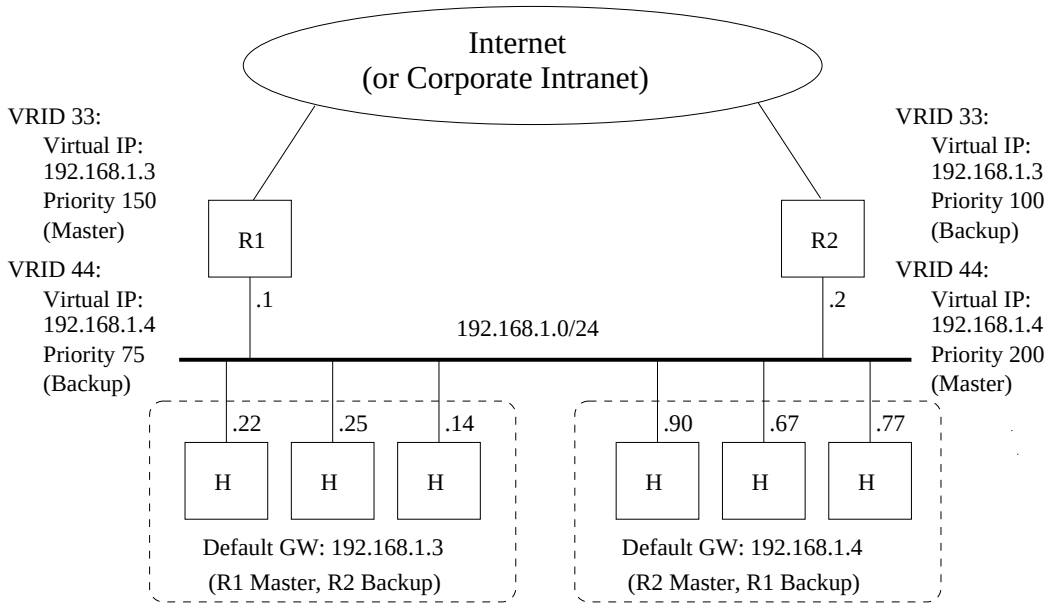


Figure 22.2: Example setup where R1 and R2 share the load from IP subnet 192.168.1.0/24, and using VRRP to backup each other.

22.2 Managing VRRP via the CLI

The table below shows VRRP management features available via the CLI.

Command	Default	Section
Configure VRRP Settings		
interface <IFACE>		
[no] vrrp <VRID>		Sec. 22.2.1
[no] address <ADDRESS>		Sec. 22.2.2
[no] interval <1..255>	1	Sec. 22.2.3
[no] priority <1..255>	100	Sec. 22.2.4
[no] preempt [delay <0..1000>]	Disabled	Sec. 22.2.5
[no] auth <plain> <SECRET>	Disabled	Sec. 22.2.6
View VRRP Settings		
interface <IFACE>		
show vrrp [VRID]		Sec. 22.2.7
vrrp <VRID>		
show address		Sec. 22.2.8
show interval		Sec. 22.2.9
show priority		Sec. 22.2.10
show preempt		Sec. 22.2.11
show auth		Sec. 22.2.12
View VRRP Status		
show vrrp		Sec. 22.2.13

22.2.1 Create and Manage a VRRP Instance

Syntax `[no] vrrp <VRID>`

Context *Interface* context

Usage Create, manage, or delete a VRRP instance. Use **"`vrrp <VRID>`"** to enter the VRRP configuration context of the specified VRRP instance (VRID can be in the range 0-255). If the instance does not already exist, it will be created.

Use **"`no vrrp <VRID>`"** to remove a specific VRRP instance, or **"`no vrrp`"** to remove all configured VRRP instances for this interface.

At most 4 VRRP instances can be created per interface.

Default values Disabled

Error messages None defined yet.

22.2.2 Configure Virtual Address

Syntax `[no] address <ADDRESS>`

Context *VRRP* context

Usage Set the virtual IP address (VIP address) used for the VRRP instance.

The VIP address should be within the same IP subnet as the regular IP address assigned to the interface (see section 17.3.7).

Only one VIP address can be configured per VRRP instance.

Default values Disabled

Error messages None defined yet.

22.2.3 Configure VRRP Advertisement Interval

Syntax `[no] interval <1..255>`

Context *VRRP* context

Usage Configure VRRP advertisement interval in seconds. A small value enables faster failover,

Use **”no interval”** to return to the default interval setting.

Default values 1 (second)

Error messages None defined yet.

22.2.4 Configure VRRP Priority

Syntax `[no] priority <1..255>`

Context *VRRP* context

Usage Configure VRRP priority. A high value increases the chance to become master of the VIP address (see also the **”preempt”** command in section 22.2.5).

Priority ”255” should be used if (and only if) this router is the *owner* of the IP address used as VIP address, i.e., if the VIP address is assigned as an IP address to this router’s interface (see section 17.3.7).

Use **”no priority”** to return to the default priority setting.

Default values 100

Error messages None defined yet.

22.2.5 Enable or Disable VRRP Master Preemption

Syntax `[no] preempt [delay <0..1000>]`

Context *VRRP* context

Usage Enable or disable VRRP master preemption. If enabled, this router will preempt an existing master if the current master has lower priority. (Note: The *owner* of a VIP address will always take over as master irrespective of the **"preempt"** setting.)

When preemption is enabled, the router will wait a time interval depending on the configured advertisement interval and a configurable preemption delay (seconds) before taking over as master.

Use **"no preempt"** to prohibit this router to preempt an existing VRRP master.

Default values Disabled (**"no preempt"**) When enabled, the delay defaults to 0 seconds.

Error messages None defined yet.

22.2.6 Configure VRRP Message Authentication

Syntax `[no] auth <plain> <SECRET>`

Context *VRRP* context

Usage Configure VRRP message authentication. Simple clear-text authentication is supported.

The associated secret can be 4-8 characters. Valid characters are ASCII characters 33-126, except '#' (ASCII 35).

Use **"no auth"** to disable VRRP message authentication.

Default values Disabled

Error messages None defined yet.

22.2.7 Show Summary of VRRP Settings

Syntax `show vrrp [VRID]`

Context *Interface* context (also available as **"show"** command within the *VRRP context*.

Usage Show summary of VRRP settings. Use **"show vrrp"** to list settings for all configured VRRP instances, and **"show vrrp VRID"** to list settings for a specific VRRP instance.

Default values By default the settings for all VRRP instances are listed.

22.2.8 Show Virtual IP Address Setting

Syntax `show interval`

Context *VRRP* context

Usage Show the configured virtual IP (VIP) address for this VRRP instance.

Default values Not applicable

22.2.9 Show VRRP Advertisement Interval Setting

Syntax `show interval`

Context *VRRP* context

Usage Show the configured advertisement interval for this VRRP instance.

Default values Not applicable

22.2.10 Show VRRP Priority Setting

Syntax `show priority`

Context *VRRP* context

Usage Show the configured VRRP priority for this VRRP instance.

Default values Not applicable

22.2.11 Show VRRP Master Preemption Setting

Syntax `show preempt`

Context *VRRP* context

Usage Show the configured VRRP master preemption setting for this VRRP instance.

Default values Not applicable

22.2.12 Show VRRP Message Authentication Setting

Syntax `show auth`

Context *VRRP* context

Usage Show the configured VRRP message authentication setting for this VRRP instance.

Default values Not applicable

22.2.13 Show VRRP Status

Syntax `show vrrp`

Context *Admin Exec* context

Usage Show the status of all configured VRRP instances.

Default values Not applicable

Chapter 23

Firewall Management

When connecting your network to the Internet (or any non-trusted network) a router with firewall functionality should be used to protect against undesired access to your local servers or other kinds of network intrusion from attackers on the Internet.

The WeOS firewall enables you to control what traffic is allowed to enter and exit your network by defining *packet filtering* rules. In addition, the firewall provides *network address translation* (NAT) gateway and *port forwarding* functionality, which are commonly needed when all hosts on your local network are sharing a single connection to the Internet.

Section 23.1 describes the firewall functionality available in WeOS. Sections 23.2 and 23.3 covers firewall management via the Web Interface and via the CLI.

23.1 Overview

Table 23.1 summarises the supported firewall functionality. Sections 23.1.1-23.1.4 provide further information on the WeOS firewall support.

23.1.1 Firewall introduction

The WeOS firewall includes support for three related types of functionality:

- *Packet Filtering*: The packet filtering support is primarily used to control what traffic is allowed to be *routed* via the switch (forward filtering), but can also be used to control accessibility to services on the switch itself (input filtering).

Feature	Web (Sec. 23.2)	CLI (Sec. 23.3)	General Description
Enable Firewall	X	X	Secs. 23.1.1-23.1.2
Packet filtering			”
Enable Packet Filtering	X	X	”
Default Forward Policy	X	X	”
Default Input Policy		X	”
Allow Rules	X	X	”
NAT	X	X	Secs. 23.1.1, 23.1.3
Port Forwarding	X	X	Secs. 23.1.1, 23.1.4
View Firewall Configuration	X	X	
View Firewall Status		X	

Table 23.1: Summary of Firewall functionality in WeOS

- *Network Address Translation (NAT)*: WeOS supports the most common NAT form, where a common (public) IP address is shared by a set of hosts in a *private* network. This form of NAT is sometimes referred to as IP Masquerading or Network Address Port Translation (NAPT).
- *Port Forwarding*: Port forwarding is commonly used together with NAT. With port forwarding a service (such as a Web Server) located in a *private* network, can be made accessible from the *public* network, typically from the Internet.

Fig. 23.1 presents an overview of the firewall mechanism. Packets are treated differently if they:

- *are destined to the switch*: Examples include HTTP/HTTPS, SSH, and SNMP traffic used to manage the switch remotely, and ICMP (Ping) traffic to check if the switch is up or not. Such packets are subject to *prerouting* and *input filtering* firewall mechanisms.
- *originate from switch*: This includes the same examples as above (HTTP/HTTPS, SSH, SNMP, ICMP, etc.) with the difference that this is the packets from the switch instead of the packets to the switch. Such packets are subject to *output filtering* and *postrouting* firewall mechanisms, however WeOS does **not** include primitives to control *output filtering*.
- *are routed via the switch*: This includes traffic that is not destined for the switch or originate from the switch. Such packets are subject to *prerouting*, *forward filtering* and *postrouting* firewall mechanisms.

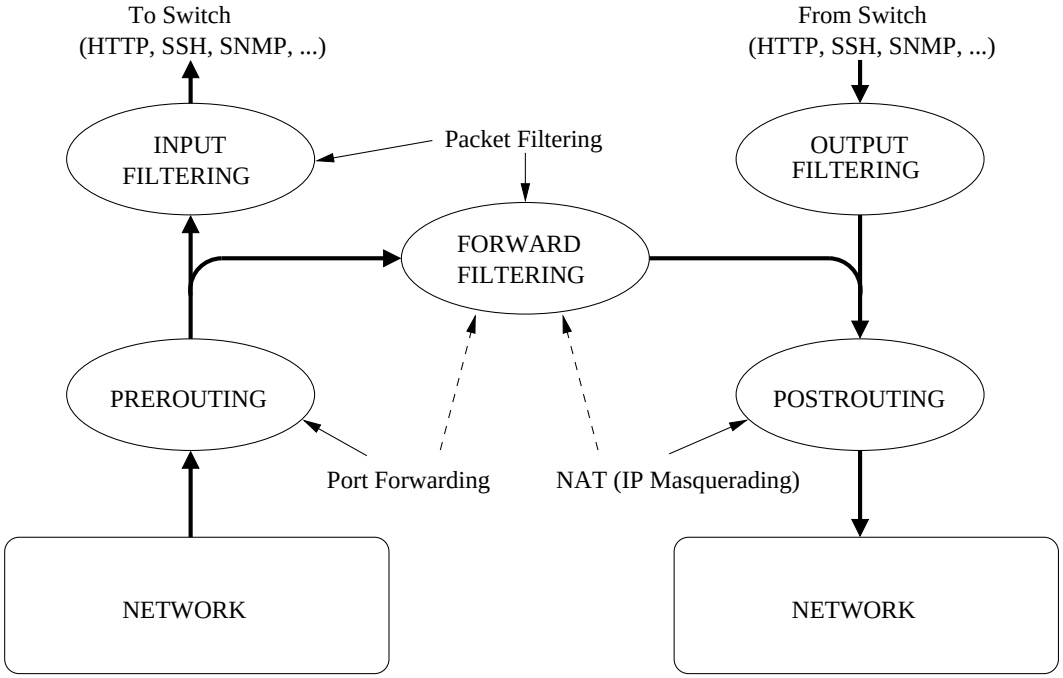


Figure 23.1: Overview of Firewall mechanism. Thick lines represent packet flows.

When the firewall is *disabled*, traffic is handled without being subject to filtering (neither input, output nor forward filtering), prerouting or postrouting mechanisms. As the firewall is *enabled*, packets destined to the switch and packet being routed via the switch start to get filtered. A set of *default rules* are automatically added to limit the risk of (unintentionally) losing management access to the switch when enabling the firewall (the added rules can be removed to limit management access to the switch). The operator is able to add additional *rules* for *NAT*, *Port Forwarding* and *Packet Filtering*.

Associated with each filtering mechanism (input, forward, output) there is a default policy, defining what to do with packets that do not match any of the defined firewall rules. When the firewall is enabled, the *default policies* for packet filtering are as follows:

- *Input Filtering*: **Deny**, i.e., packets to the switch are dropped unless they are explicitly allow. Note: the Input Filter is also affected by the *Management Interface* configuration, see section 17.1.1.5.
- *Forward Filtering*: **Deny**, i.e., when enabling the firewall no packets will be routed by the switch until such packet filter rules are explicitly defined. (As

described in sections 23.1.2-23.1.4, adding a NAT or Port Forwarding also affects *Forward Filtering*).

- *Output Filtering*: **Accept**, i.e., there are no restrictions on the traffic originating from the switch.

The WeOS firewall utilise *connection tracking*, meaning that traffic part of (or related to) a permitted connection is also accepted. Thus, if a packet filter rule is added to allow traffic to pass from your internal network to the Internet, packets associated with that connection will also be allowed to pass from the Internet to your internal network.

When a NAT rule is defined it primarily affects the *forward filtering* and *postrouting* steps (see fig. 23.1) of traffic going from your internal network to the external network, however, it is the *connection tracking* mechanisms that ensures that packets in the reverse direction (associated with those flows) are accepted and processed appropriately.

23.1.2 Packet Filtering

Packet filtering *rules* can be specified to *match* IP packets based on the following filtering parameters:

- *Inbound Interface*: The interface the packet comes in on.
- *Outbound Interface*: The interface the packet will be sent out on.
- *Source IP Address/Subnet*: The source IP address of the packet. This can be specified as a single IP address, or the rule could match a whole IP subnet.
- *Destination IP Address/Subnet*: The destination IP address of the packet. This can be specified as a single IP address, or the rule could match a whole IP subnet.
- *Protocol*: The *protocol* type of the IP payload. Typically TCP or UDP, but the filtering can also be made to match on other protocols such as ICMP and ESP[1].
- *Destination (UDP/TCP) Port*: When *protocol* is specified as UDP or TCP, the filter can match on the associated UDP/TCP port number(s).

Note: *If "outbound interface"* **and/or** *"destination IP Address/subnet" are specified in the packet filter rule, the rule will apply to the "Forwarding Packet Filter". If* **neither** *"outbound interface"* **nor** *"destination*

[1] See http://www.iana.org/assignments/protocol-numbers/ for IP for a list of defined IP protocols. As of WeOS version v4.3.0 the Web Management limits the use of IP payload protocol to UDP and TCP, while the CLI provides a larger set of protocols.

IP Address/subnet" are specified, the filter rule will apply to the "Input Packet Filter". (WeOS does not support adding rules to the "Output Packet Filter".)

When the firewall is enabled, an incoming packet will be processed according to the *rules* defined for *input filter* when the packet is destined to the switch, or the rules defined for the *forwarding filter* when the packet is being routed through the switch. The list of rules is searched (in order) until a match is found, or until the end of the list is reached.

- If the packet does not match any rule, it will be processed according to the default policy. Both the *input* and *forwarding* filter have default policy *drop* when the firewall is enabled, but the default policy can be changed to *accept* (the Web configuration capability may have limitations as compared to the CLI).
- If a packet *matches* a configured rule, the packet is accepted. That is, it is only possible to define *allow rules* as of WeOS version v4.3.0. Support for *deny rules* is left to be added in future releases.

As described in section 17.1.1.5, an operator can use the *Management Interface* feature to enable/disable services per network interface. The management interface configuration is kept separate from the firewall configuration, but both configuration methods can affect the *Input Filter*. Thus, when expecting the firewall status entries for the following services can be shown, even if no corresponding allow rule has been added in the firewall configuration.

- TCP port 22 (SSH)
- TCP port 80 (HTTP)
- TCP port 443 (HTTPS)
- UDP ports 5097-5098 (IPConfig)
- TCP/UDP port 161 (SNMP)

When enabling the firewall functionality for the first time, the operator is given the choice to add a set of default *allow* packet filter rules (*Input Filter*) to simplify switch operation.

- ICMP: Useful to check if the switch is up and reachable.
- TCP/UDP port 53 (DNS) and UDP port 67 (DHCP): Useful when the switch is running a DHCP server.
- TCP/UDP port 500 (IKE/ISAKMP) and UDP 4500 (IPSec NAT-T): Useful when the switch is operating as VPN gateway.

Note: *The "packet filter" function is the primary method to add rules to the Forward Filter. However, to simplify NAT management, adding a "NAT" rule will implicitly add a rule to allow (all) traffic to pass from the "internal" to the "external" interface. Similarly, when adding a port forwarding rule (prerouting), a matching rule is implicitly added to the "forward" filter to allow "external" hosts to access the specified service (IP address and UDP/TCP port) in the "internal" network. These implicit additions of rules are indicated with the dashed arrows in fig. 23.1.*

For more details on NAT and port forwarding, see sections 23.1.3 and 23.1.4.

23.1.3 Network Address Translation

WeOS NAT support enables hosts on a private network to share an Internet connection with a single public IP address, see fig. 23.2.

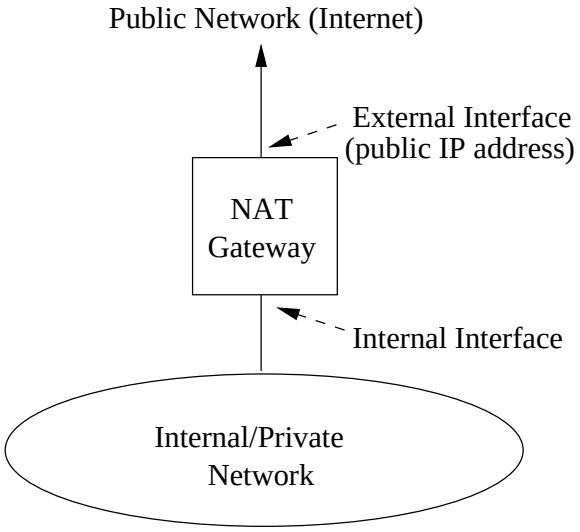


Figure 23.2: NAT gateway providing access to the Internet. All hosts in the private network share a single public IP address.

When specifying a NAT rule, all that needs to be defined with respect to the firewall functionality are the name of the *internal interface* and the *external interface*[2]. The appropriate rules will then be added to the *forward filtering* and

[2]Appropriate interface IP settings must be configured, and IP routing must also be enabled, see chapter 17.

postrouting steps (see fig. 23.1) to allow hosts in the private network to initiate connections to the Internet. *Connection tracking* will ensure packets in the reverse direction (from the Internet to the private network) are accepted and managed properly.

23.1.4 Port Forwarding

Port Forwarding is commonly used together with NAT, to enable access from the Internet to a server inside the private network. Fig. 23.3 shows a typical setup when *port forwarding* is useful:

- The switch acts as a NAT gateway to the Internet: routing is enabled (see section 17.1) and a NAT rule defining the internal and external interfaces has been configured (see section 23.1.3).
- A Web Server on the ”internal” network serves users on the Internet: A port forwarding rule has been added to allow users on the Internet to initiate connections to the Web server on host 192.168.0.2 (TCP port 80).

With port forwarding, users on the Internet will connect to the internal Web Server as if it was running on the NAT gateway, i.e., users on the Internet will connect to the Web server using the public IP address (here 1.2.3.4) and TCP port number (here 8080), without knowing that the traffic is forwarded to a server inside the internal network.

Configuration of port forwarding rules include the following parameters:

- *Inbound Interface*: Packets which are subject to port forwarding should come in on the specified interface. In the example network shown in fig. 23.3, this would be the *external interface*, i.e., the attached to the Internet.
- *Inbound Port (Range)*: Defines the range of TCP/UDP port numbers, which are to be mapped by this rule. In the example in fig. 23.3 Internet hosts would reach the Web server using TCP port 8080.
- *Source IP Address/Subnet*: Optional argument limiting the port forwarding rule to concern a limited set of Internet hosts.
- *Destination IP Address*: Specifies the IP address of the private server, i.e., where packets are to be sent. The Web server in in fig. 23.3 has IP address *192.168.0.2*.
- *Destination Port (Range)* Specifies which TCP/UDP port number(s) to use on the in the forwarded packet. The default is to use the same port number(s) as on the inbound interface. In the example, the Web server on the internal server uses TCP port 80.

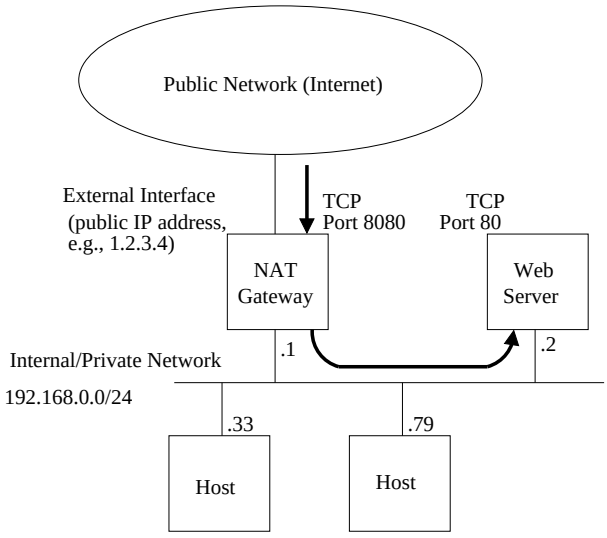


Figure 23.3: Use of port forwarding to enable Internet hosts to access a Web server inside the private network.

- *Transport Protocol (TCP/UDP)*: Specify if this rule applies to TCP, UDP or both. In the example, the rule applies only to TCP.

23.2 Firewall Management via the Web Interface

Menu path: Configuration ⇒ Firewall ⇒ Common

On the firewall common settings page you may enable or disable the firewall. When disabling the firewall all rules will be lost. A confirmation is required if you try to disable the firewall to not loose rules by accident.

Enabled	Check this box to enable firewall functionality. **Note:** When disabling the firewall, the firewall is stopped and all existing *NAT* rules, *Port Forwarding* rules and *Access* rules are deleted.

23.2.1 NAT Rules

Menu path: Configuration ⇒ Firewall ⇒ NAT

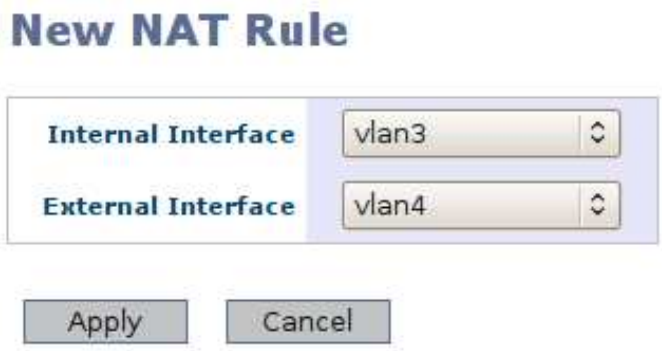
On the Firewall NAT configuration page you are presented to the list of current NAT rules. (If the firewall function is disabled or no rules have been created you will not see any list, but be presented to an information message.)

Order	The order in which the rules will be applied.
Internal Interface	The interface connected to your subnet whose addresses you want to translate (the interface to your internal/private network).
External Interface	The interface that should represent all IP addresses on the subnet of the **`internal interface`**. This is the external/public interface, typically the interface connected to the Internet.
Delete	Click this icon to remove a NAT rule. You will be asked to acknowledge the removal before it is actually executed.
New Nat Rule	Click this button to create a new NAT rule. You will be presented to a form where you can configure the new rule.

23.2.2 New NAT Rule

Menu path: Configuration ⇒ Firewall ⇒ NAT ⇒ New NAT Rule

In the **`New NAT Rule`** configuration page you can specify a new NAT rule.



Internal Interface	Mandatory. The interface connected to your subnet whose addresses you want to translate (the interface to your internal/private network).
External Interface	Mandatory. The interface that should represent all IP addresses on the subnet of the **`internal interface`**. This is the external/public interface, typically the interface connected to the Internet.

23.2.3 Port Forwarding Rules

Menu path: Configuration ⇒ Firewall ⇒ Port Forwarding

Port forwarding is e.g. used to give external units access to specific services in a subnet hidden by NAT. If firewall is disabled or no rules created you will see no list, but be presented to an information message.

Order	The order in which the rules will be applied.
Protocol	Traffic may be filtered on transport layer protocol. Available are TCP and UDP.
Incoming Interface	The interface from which inbound traffic should be allowed.
Incoming Destination Port	The transport layer port to which traffic is addressed. E.g. 80 for standard web-server access.
Incoming Source Address(es)	Optional. The source IP address(es) of packets allowed to be forwarded. Either a single address, or a subnet. Subnet mask is displayed in CIDR notation (prefix length).
Destination Address	The destination IP address to which the packets will be forwarded.
Destination New Port	Traffic is redirected to this port on the destination host. Empty means that the incoming destination port will be used.
Delete	Click this icon to remove a port forwarding rule. You will be asked to acknowledge the removal before it is actually executed.
New Forwarding Rule	Click this button to create a new port forwarding rule. You will be presented to a form where you can configure the new rule.

23.2.4 New Port Forwarding Rule

Menu path: Configuration ⇒ Firewall ⇒ Port Forwarding ⇒ New Forwarding Rule

Protocol	Mandatory. Traffic may be filtered on transport layer protocol. Available are TCP and UDP. Choose *any* to allow both TCP and UDP packets.
Incoming Interface	Mandatory. The interface from which inbound traffic should be allowed.
Incoming Destination Port	Mandatory. The transport layer port to which traffic is addressed. E.g. 80 for standard web-server access.
Source	Optional. The source IP address(es) of packets allowed to be forwarded. Either a single address, or a subnet. If single is selected, enter a single address. If subnet is selected a netmask (e.g. 255.255.255.0) must also be entered to define the subnet. If you have a JavaScript[1] enabled browser the netmask field will not be displayed unless you check the subnet radio button.
Destination Address	Mandatory. The destination IP address to which the packets will be forwarded.
New Destination Port	Optional. Traffic is redirected to this port on the destination host. If another port is used by the destination host for the service you can map the port by entering another port number. Empty means that the incoming destination port will be used.

[1] JavaScript is a trademark of Sun Microsystems.

23.2.5 Access Rules

Menu path: Configuration ⇒ Firewall ⇒ Access

Access rules are set up to allow traffic to pass through the firewall. Traffic is by default denied, except for a set of default allow rules created.
If firewall is disabled or no rules created you will see no list, but be presented to an information message.

Forward Chain Policy	The policy defines how to handle data for which no matching rule can be found. The forward chain controls traffic passing through the switch, not traffic destined to the switch itself. Possible values are: **Allow** Packets will be allowed through. **Drop** Packets will be dropped and no other actions are taken.
Rules Activated	Yes means rules are active. No means rules are deactivated and all traffic is allowed through.
Edit	Click this icon to edit the global settings.
Order	The order in which the rules will be applied.

Continued on next page

	Continued from previous page
In Interface	The interface from which traffic should be allowed to flow to the Out Interface.
Out Interface	The interface to which traffic should be allowed to flow from the In Interface. If neither *Out Interface* nor *Destination Address* (see below) are specified, the rule will apply to the INPUT chain, i.e., traffic destined to the switch itself (ICMP pings, SSH management, etc.).
Source Address(es)	A single IP-address or a subnet from which traffic should be allowed through.
Destination Address(es)	A single IP-address or a subnet to which traffic should be allowed through. If neither *Out Interface* (see above) nor *Destination Address* are specified, the rule will apply to the INPUT chain, i.e., traffic destined to the switch itself (ICMP pings, SSH management, etc.).
Destination Port	Only traffic for this (UDP/TCP) port will be allowed through.
Protocol	Only traffic using this protocol will be allowed through. TCP, UDP or any to allow TCP, UDP or any kind of traffic through (not only TCP and UDP).
Delete	Click this icon to remove an access rule. You will be asked to acknowledge the removal before it is actually executed.
New Access Rule	Click this button to create a new access rule. You will be presented to a form where you can configure the new rule.

23.2.6 Edit Access Control Common Settings

Menu path: Configuration ⇒ Firewall ⇒ Access ⇒ (Common Settings)

Here you may change the common settings for the access control filter rules.

Forward Chain Policy	The policy defines how to handle data for which no matching rule can be found. The forward chain controls traffic passing through the switch, not traffic destined to the switch itself. Possible values are: **Allow** Packets will be allowed through. **Drop** Packets will be dropped and no other actions are taken. Select the policy by clicking the radio button.
Rules Activated	Check the box to activate the rules, or uncheck to deactivate the rules. Deactivation means all traffic is allowed through (policy is changed to *allow*).

23.2.7 New Access Control Rule

Menu path: Configuration ⇒ Firewall ⇒ Access ⇒ New Access Rule

In Interface	The interface from which traffic should be allowed to flow to the Out Interface. *In Interface* and/or *Source Address* (see below) must be selected.
Out Interface	The interface to which traffic should be allowed to flow from the In Interface. If neither *Out Interface* nor *Destination Address* (see below) are specified, the rule will apply to the INPUT chain, i.e., traffic destined to the switch itself (ICMP pings, SSH management, etc.).
Protocol	Only traffic using this protocol will be allowed through. Select *TCP* or *UDP* to allow TCP or UDP traffic through (see also *Destination Port* option below). Select *any* to allow traffic from any IP Protocol (ICMP, TCP, UDP, . . .) through.
Source Address(es)	A single IP-address or a subnet from which traffic should be allowed through. Select *Single* and enter the single source address into the address field. Select *Subnet* and enter an address into the address field and a subnet mask into the *Netmask* field. *In Interface* (see above) and/or *Source Address* must be selected.
Destination Address(es)	A single IP-address or a subnet to which traffic should be allowed through. Select *Single* and enter the single source address into the address field. Select *Subnet* and enter an address into the address field and a subnet mask into the *Netmask* field. If neither *Out Interface* (see above) nor *Destination Address* are specified, the rule will apply to the INPUT chain, i.e., traffic destined to the switch itself (ICMP pings, SSH management, etc.).
Destination Port	Only traffic for this (UDP/TCP) port will be allowed through. Only valid if *Protocol* TCP or UDP has been selected (see above).

23.3 Firewall Management via the CLI

Command	Default	Section
Configure Firewall Settings		
[no] firewall	Disabled	Section 23.3.1
[no] enable	Enabled	Section 23.3.2
[no] allow [in <IFACE>] [out <IFACE>] [src <IPADDRESS[/LEN]>] [dst <IPADDRESS[/LEN]>] [proto <PROTO_NAME|PROTO_NUM>] [dport <PORTRANGE>]		Section 23.3.3
[no] nat <INT_IFACE> <EXT_IFACE>		Section 23.3.4
[no] port-forward in <IFACE>:<PORTRANGE> [src <IPADDRESS/LEN>] dst <IPADDRESS>[:PORTRANGE] [proto <tcp|udp>]		Section 23.3.5
policy [forward|input] [<deny|allow>]	Deny	Section 23.3.6
View Firewall Settings		
show firewall		Section 23.3.7
firewall		
show enable		Section 23.3.8
show allow		Section 23.3.9
show nat		Section 23.3.10
show port-forward		Section 23.3.11
show policy		Section 23.3.12
View Firewall Status		
show firewall		Section 23.3.13

23.3.1 Managing the Firewall

Syntax `[no] firewall`

Context *IP* context

Usage Enter the Firewall context. This will enable the firewall (unless it is already enabled).

Use **"no firewall"** to disable the firewall, and to delete all existing *NAT*, *Port Forwarding* and *Packet filter (allow)* rules.

Default values Disabled.

Error messages None defined yet.

23.3.2 Enable Packet Filter Rules

Syntax `[no] enable`

Context *Firewall* context

Usage Enable/disable packet filtering. This setting only affects packet filtering, not NAT or Port Forwarding rules (they are always enabled).

Use **"no enable"** to deactivate all existing packet filter rules. Use **"enable"** to reactivate them.

Default values Enabled

Error messages None defined yet.

23.3.3 Configure Packet Filter Allow Rule

Syntax `[no] allow [in <IFACE>] [out <IFACE>] [src <IPADDRESS[/LEN]>] [dst <IPADDRESS[/LEN]>] [proto <PROTO_NAME|PROTO_NUM>] [dport <PORTRANGE>]`

Context *Firewall* context

Usage Add or delete a packet filter *allow* rule.

- The **"in <IFACE>]"** and/or **"[src <IPADDRESS[/LEN]>]"** arguments must be included in the **"allow"** packet filter specification.
  The **"in <IFACE>]"** and **"[src <IPADDRESS[/LEN]>]"** are used to match the inbound interface and source IP address of a packet. If the **"LEN"** parameter is omitted the **"[src <IPADDRESS[/LEN]>]"** argument will match a single source IP address. If included it will match a whole IP subnet.
- Include the **"[out <IFACE>]"** argument to define a FORWARDING rule (i.e., packets being routed through the switch). If both the **"[out <IFACE>]"** and the **"dst <IPADDRESS[/LEN]>"** arguments are omitted, the rule will apply to the INPUT chain, i.e., traffic destined to the switch itself (ICMP pings, SSH management, etc.).
  The **"[out <IFACE>]"** argument is used to match the outbound interface of a packet.
- Use the [dst <IPADDRESS[/LEN]>] to match a single destination IP address or whole subnet. If both the **"[out <IFACE>]"** and the **"dst <IPADDRESS[/LEN]>**

arguments are omitted, the rule will apply to the INPUT chain, i.e., traffic destined to the switch itself (ICMP pings, SSH management, etc.).

- Use the **”[proto <PROTO_NAME|PROTO_NUM>]”** to match the IP protocol name, e.g., *tcp*, *udp* or *icmp*. It is also possible to specify the protocol’s assigned number, see http://www.iana.org/assignments/protocol-numbers/.
- Use the **”[dport <PORTRANGE>]”** argument to specify a UDP or TCP port number or number range. This argument is only valid if **”[proto udp]”** or **”[proto tcp]”** is included.

Default values Not applicable.

Error messages None defined yet.

23.3.4 Configure NAT Rule

Syntax `[no] nat <INTERNAL_IFACE> <EXTERNAL_IFACE>`

Context *Firewall* context

Usage Add/delete a NAT rule. Also known as IP masquerading, used to hide RFC 1918 private subnets behind a single public IP.

Use **”no nat <INTERNAL_IFACE> <EXTERNAL_IFACE>”** to remove a specific NAT rule.

Default values

Error messages None defined yet.

23.3.5 Configure Port Forwarding Rule

Syntax `[no] port-forward in <IFACE>:<PORTRANGE> [src <IPADDRESS/LEN>] dst <IPADDRESS>[:PORTRANGE] [proto <tcp|udp>]`

Context *Firewall* context

Usage Add/delete a Port Forwarding rule. This is commonly used when the switch is acting as NAT gateway, see section 23.3.4. E.g., **”port-forward in vlan1:80 dst 10.0.0.2 proto tcp”** to forward all web traffic coming in on interface *vlan1* to the Web server at IP address 10.0.0.2 (port 80).

- The argument **”<IFACE>:<PORTRANGE>”** specifies incoming interface, and what port or port range to match.
- Use the **”[src <IPADDRESS[/LEN]>]”** to match a single source IP address or whole subnet.

- Use the **”dst <IPADDRESS>[:PORTRANGE]”** to specify where the packets should be forwarded. If the **”PORTRANGE”** parameter is omitted, the same port range as specified in the **”<IFACE>:<PORTRANGE>”** argument is used.
- Use the **”[proto <tcp|udp>]”** to specify if the rule applies to TCP or UDP. If omitted, the rule applies to both.

Default values

Error messages None defined yet.

23.3.6 Configure Forwarding and Input Default Policies

Syntax `[policy [forward|input] <allow|deny>`

Context *Firewall* context

Usage Configure the default policy for *forward filtering* and *input filtering*. By default, the command applies to the *forwarding filter*, e.g., **”policy allow”** will set the default policy for forward filtering to **”allow”**.

Default values Deny (that is, both the forwarding filter and the input filter by default drop packets lacking a matching *allow* rule.)

Error messages None defined yet.

23.3.7 View Firewall Configuration Settings

Syntax `show firewall`

Context *IP* context. Also available as **”show”** command within the Firewall context.

Usage Show firewall configuration. If the firewall is enabled, the list of currently configured NAT, Port Forwarding and Packet Filtering rules are presented.

Default values Not applicable.

Error messages None defined yet.

23.3.8 View Firewall Packet Filter Enable Setting

Syntax `show enable`

Context *Firewall* context.

Usage Show whether the configured packet filters are enabled or disabled.

Default values Not applicable.

Error messages None defined yet.

23.3.9 View Packet Filter Rules

Syntax `show allow`

Context *Firewall* context.

Usage Show configured *allow* packet filter rules.

Default values Not applicable.

Error messages None defined yet.

23.3.10 View NAT Rules

Syntax `show nat`

Context *Firewall* context.

Usage Show configured *NAT* rules.

Default values Not applicable.

Error messages None defined yet.

23.3.11 View Port Forwarding Rules

Syntax `show port-forward`

Context *Firewall* context.

Usage Show configured *port forwarding* rules.

Default values Not applicable.

Error messages None defined yet.

23.3.12 View Port Forwarding Rules

Syntax `show policy`

Context *Firewall* context.

Usage Show configured default policies for the *forwarding filter* and the *input filter*.

Default values Not applicable.

Error messages None defined yet.

23.3.13 View Firewall Status

Syntax `show firewall`

Context *Admin Exec* context

Usage Show current NAT rules, Port Forwarding rules, and policies and entries in the Input and Forwarding Filters. In addition, management interface configuration (see section 17.1.1.5) will appear as entries in the *Input Filter*.

Default values Not applicable.

Error messages None defined yet.

Chapter 24

Virtual Private Network

WeOS provides virtual private network (VPN) support via IPSec VPNs. A WeOS switch can act as a VPN gateway in NETWORK-NETWORK and HOST-NETWORK scenarios. Configured as a VPN gateway, it can be used to securely connect branch office networks with a central office network, or to serve individual users wishing to ”dial in” securely over the Internet to the central office network, with their PC connected at some remote site. The data traffic will be protected by encrypted tunnels when sent over the Internet.

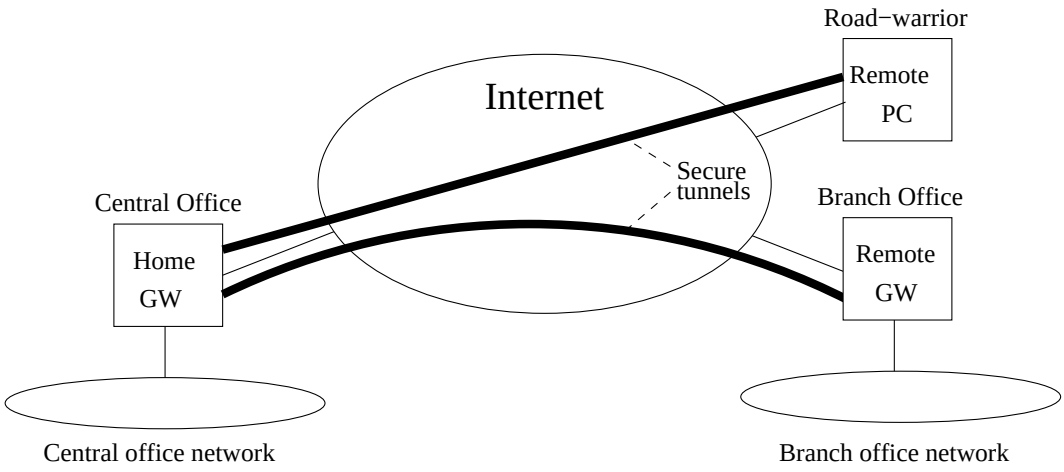


Figure 24.1: IPSec VPN tunnels can be used to securely connect hosts and networks over the Internet.

24.1 Overview of VPN Management Features

Feature	Web (Sec. 24.2)	CLI (Sec. 24.3)	General Description
VPN Configuration			
Add/Delete IPSec VPN tunnels	X	X	Sec. 24.1.1
Local/Remote Subnets	X	X	–"–
Outbound Interface	X	X	–"–
NAT Traversal	X	X	–"–
IKEv1	X	X	Sec. 24.1.2
Role (Initiator/Responder)	X	X	–"–
Mode (Main/Aggressive)	X	X	–"–
IKE Cipher Suite	X	X	–"–
Identity	X	X	–"–
Pre-Shared Key	X	X	–"–
ESP Cipher Suite	X	X	–"–
Perfect Forward Secrecy		X	Sec. 24.1.3
MTU Override	X	X	Sec. 24.1.4
Dead Peer Detection		X	Sec. 24.1.5
VPN Status			
Show IPSec Tunnel Status	X	X	

24.1.1 Introduction to IPSec VPNs

A common use case for IPSec VPNs is to connect two networks via a secure tunnel over the Internet. We refer to this scenario as NETWORK-NETWORK VPNs, and is accomplished by having two VPN gateways, one at each site, negotiate and establish a secure *tunnel*, and to forward all traffic between the two networks through this tunnel. By creating VPN tunnels you establish a secure *overlay* network on top of your regular Internet connections.

We use fig. 24.2 to explain some VPN related terminology.

- *Peers:* The two VPN gateways (Alice and Bob) are referred to as IPSec peers. The peers constitute the end-points of the secure tunnel. One of the peers will take the role of tunnel *initiator* and the other takes the *responder* role.
- *Initiator and Responder:* The VPN *initiator* is the peer that is responsible initiate the tunnel establishment by contacting the other peer - the *responder*. In fig. 24.2 we have assumed that Alice is the responder and Bob is the initiator.

  A WeOS switch configured as a VPN gateway is able to act both as *responder*

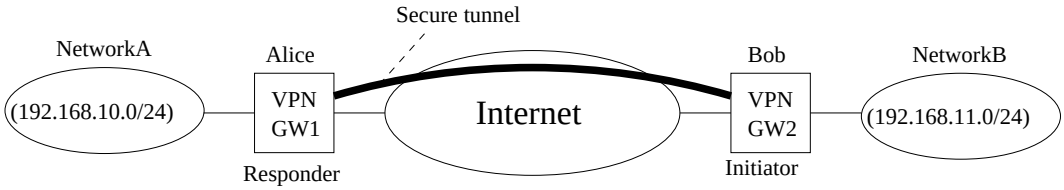


Figure 24.2: By establishing a secure IPSec Tunnel between the VPN gateways (Alice and Bob), traffic between Network-A and Network-B will be protected when sent across the Internet.

(default) and as *initiator*.

- *NAT-traversal, Peer IP addresses and DDNS:* In order to act as a responder, Alice must be assigned a *public* (routable) IP address on its interface towards the Internet. Thus, Alice generally cannot be located behind a NAT gateway, since the initiator (Bob) would not be able to initiate the tunnel. Bob will need to know Alice's IP address (or domain name) in order to know where to send the tunnel establishment messages. If Alice is assigned a fixed IP address, Bob can choose between using Alice's IP address or her domain name. But if Alice gets her address dynamically (e.g., via DHCP), Bob should use her domain name to establish the contact. WeOS supports dynamic DNS (DDNS), thus Alice can dynamically register her current IP address, see section 17.1.2.3.

  The initiator (Bob) does not need to be assigned a public IP address. Bob is able to establish the tunnel even if he is located behind a NAT gateway, given that *NAT-traversal* (NAT-T) is enabled both in Alice's and Bob's VPN configurations.

  Furthermore, it is not mandatory for Alice to know Bob's IP address beforehand. It is possible to configure the VPN tunnel such that Bob could connect to the Internet at various locations and still be able to establish the VPN tunnel. This is commonly referred to as Bob being a *road warrior*.

- *Local and Remote Subnet:* Each peer will define what traffic should be allowed to pass through the established tunnel. Each peer will define the local and remote subnet, and all traffic between this subnets is sent securely through the tunnel. To secure all traffic between networks "A" and "B", Alice would define *192.168.10.0/24* as *local subnet*, and *192.168.11.0/24* as *remote subnet* in the tunnel configuration. Bob would do the opposite, i.e.,

define *192.168.11.0/24* as *local subnet*, and *192.168.10.0/24* as *remote subnet*.

More advanced settings for the local and remote subnet parameters are possible, e.g., it is possible to configure the tunnel so that all traffic from Network B is sent through the tunnel (i.e., not only the traffic heading for Network A).

- *Outbound interface:* The *outbound interface* denotes the interface, and implicitly the IP address, a VPN gateway uses to tunnel the traffic through, and to communicate with its peer. In fig. 24.2 Alice *outbound interface* would her interface towards the Internet (and the same goes for Bob).

  By default, the *outbound interface* is set to the interface leading to the *default gateway* (see section 17.1.2).

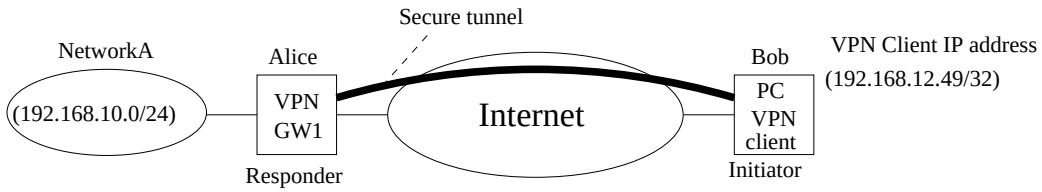


Figure 24.3: IPSec VPNs can be used to provide secure connections between individual hosts and a network behind a VPN gateway, a HOST-NETWORK VPN.

Another common use case is shown in fig. 24.3. In this case Bob is an individual host, i.e., a PC with VPN client software installed. A WeOS switch is able to act as VPN gateway in HOST-NETWORK scenarios. The host (Bob) should be assigned a VPN client IP address (*192.168.12.49* in fig. 24.3), which is used to communicate with the hosts in Network-A. For Alice the configuration is very similar to the NETWORK-NETWORK example above, with the main difference being that her remote-subnet is defines an individual IP address (*192.168.12.49/32*, i.e., *netmask 255.255.255.255*) instead of a network. As in the NETWORK-NETWORK use case, Bob's PC can be configured as a *road-warrior* connecting from different IP addresses, and with NAT-T enabled he can connect from behind a NAT gateway.

24.1.2 Authenticated Keying using Internet Key Exchange (IKE)

As part of the IPSec VPN tunnel establishment Alice and Bob will use the IKE (Internet Key Exchange) protocol to authenticate each other and create necessary session keys to protect the data traffic. WeOS supports IKE version 1 (IKEv1) with

authentication through *pre-shared keys* (PSK)[1]. In IKEv1 there are two authentication handshakes (phase-1 and phase-2):

- IKE phase-1 handshake: In this document the IKE phase-1 handshake is simply referred to as the *IKE handshake*. In the IKE handshake Alice and Bob identify themselves and use their configured PSK to authenticate each other. When configuring an IPSec tunnel, the identities of the peers should be defined. Four methods are provided:
  - IP Address (ID_IPV4_ADDR): If the IP address of the peer is known, it can be used to identify it. When using main mode (with PSK) this is the only option. When using IP address authentication, WeOS allows you to specify either an IP address or a domain name, which is then resolved via DNS.
  - Domain name (ID_FQDN): The identification can be specified as the domain name of the peer. When specifying *type* ”domain name”, the entered identity value (e.g., *foobar.westermo.com*) is sent *as is*, i.e., it is **not** resolved to an IP address. Therefore, the domain name identification type could be used as a general user name, such as *foobar*.
  - Email style (ID_USER_FQDN): The identification can be specified in email address style, e.g., *foobar@westermo.com*.
  - Key identification (ID_KEY_ID): With the key identification type, the identification can be entered as an opaque byte stream. As with the domain name type, the key identification type can be used to enter a general user name, such as *foobar*.

  The IKE handshake also creates the necessary credentials for the following ESP handshake.
- IKE phase-2 handshake: In this document the IKE phase-2 handshake is referred to as the *ESP handshake*. In the ESP handshake the *cipher suite* for the VPN tunnel is negotiated as well as the *session keys* used to encrypt and integrity protect the data send through the tunnel.

The user can also specify whether the IKE handshake should use the *main* (default) or *aggressive* mode. Aggressive mode must be used to support *road-warriors*[2], but when the initiator has a known, public IP address IKE main mode is recommended.

[1] Support for IKEv1 authentication using certificates, as well as IKEv2 is planned, but not yet included in WeOS.

[2] When WeOS includes support for IKEv1 certificate authentication, the road-warrior scenario can also be supported with *main* mode.

Both for the IKE and ESP handshakes the user can specify which cryptographic protocols to use. The following algorithms are supported by WeOS:

- *Encryption algorithm:* Supported encryption algorithms are *3DES*, *AES* (key length 128 and 256 bits), and *Blowfish*.
- *Message authentication/integrity:* Supported hash algorithms for message authentication are *MD5*, and *SHA-1*.
- *Diffie-Hellman groups:* Supported Diffie-Hellman groups are 1024 (DH group 2), 1536 (DH group 5), 2048 (DH group 14), 3072 (DH group 15) and 4096 (DH group 16).

  In WeOS, the Diffie-Hellman group used for the IKE handshake will also be used for the ESP handshake.

When using IKE *main* mode, Alice and Bob can be configured to automatically negotiate a suitable cipher suite. When using *aggressive* mode, Alice and Bob should be configured to use a specific cipher suite (same at both sides). When aggressive mode is selected, WeOS by default uses the suite *AES128-SHA1-DH1024*.

24.1.3 Perfect Forward Secrecy

Perfect Forward Secrecy (PFS) refers to the property that if an ESP session key is compromised, the attacker will only get access to the data protected by that single key. Previous and later session keys will not be revealed just because that single key was compromised, thus data encrypted by those keys is still protected.

PFS is enabled by default on all new tunnels.

> **Note**: *This setting is not supported by all IPSec implementations. It is however recommended to have it enabled, on both sides of the connection.*

If you are unsure what do to, you can safely disable PFS[3]. If the IPSec daemon receives a request with PFS, it will allow it despite PFS being disabled or not.

24.1.4 Data encapsulation and encryption

IPSec specifies two modes to encapsulate the data, a *transport* and a *tunnel* mode. WeOS IPSec VPN only support the *tunnel* mode. In the tunnel mode,

[3] As of WeOS v4.3.0 PFS can only be managed via the CLI.

the original IP packets are encapsulated within another IP packet as shown in fig. 24.4.

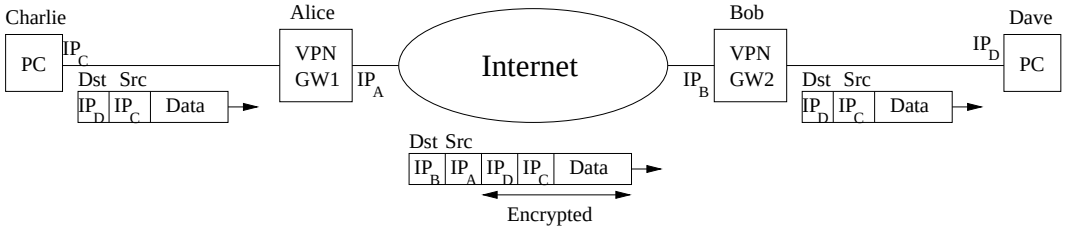


Figure 24.4: IPSec tunnel mode encapsulation. The ”inner” IP header holds the original IP addresses of Charlie and Dave, and the outer IP header contains the addresses of the VPN gateways Alice and Bob.

In IPSec there is also the choice by protecting the data using *AH* (Authentication Header), and ESP (Encapsulating Security Payload) formats. WeOS only supports ESP, which is the format to use to achieve both data *encryption* and *integrity* protection.

In order to send encapsulated data more efficiently over the Internet an operator can tune the maximum transmission unit (MTU) for VPN tunnels. By default the MTU for VPN tunnels is set to 1419 bytes.

24.1.5 Dead Peer Detection

The connectivity through an established IPSec tunnel may be broken unexpectedly, e.g., one of the peers go down or is disconnected, or if some kind of routing, NAT or firewall problem occurs on the path between them.

Dead Peer Detection (DPD) can be used to discover and manage such situations. In DPD the peers exchange keep-alive messages to monitor if the remote peer is still reachable. If a peer determines connectivity to be broken, appropriate *actions* should be taken. There are three configuration options for the DPD action:

- *Restart:* An initiator should try to reestablish an IPSec tunnel by restarting the IKE handshake.
- *Hold:* A responder can chose the *Hold* DPD action. This is often the preferred option in a NETWORK-NETWORK VPN scenario (see fig. 24.2).
- *Clear:* A responder can also chose the *Clear* DPD action. This is the preferred option if the HOST-NETWORK VPN scenario, i.e., if the initiator is a

single road-warrior (see fig. 24.3), but *Clear* may also be used in a NETWORK-NETWORK VPN scenario.

As of WeOS v4.3.0 a VPN gateway configured as initiator will use DPD action *restart* by default, while a responder by default used DPD action *clear*.

Two additional DPD parameters can be configured:

- DPD Delay: The DPD delay is the interval between DPD probing messages sent by a VPN gateway.
- DPD Timeout: If a period corresponding to the DPD timeout elapses without getting any response on the DPD probe messages, the VPN gateway considers the peer to be down.

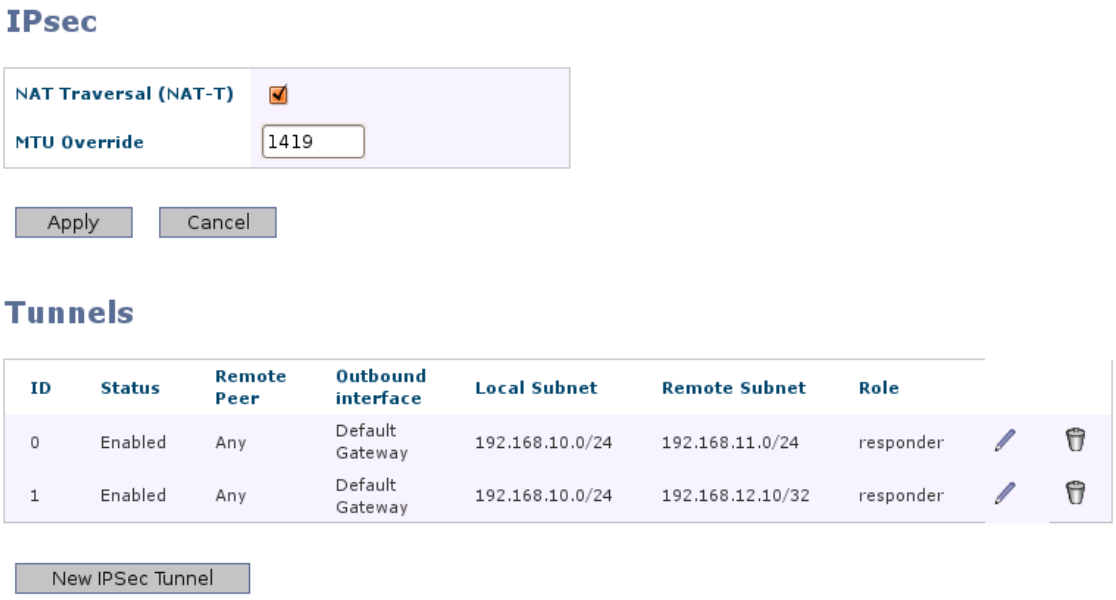
The DPD settings can be configured individually on each peer. It is even possible to disable DPD on one of the peers - that peer will still respond to DPD probing messages from the other peer.

24.2 Managing VPN settings via the web interface

24.2.1 Manage IPSec VPN via the web interface

Menu path: Configuration ⇒ VPN ⇒ IPSec

The main IPSec VPN configuration pages contains two parts: the top part lists general IPSec settings applying to all ports, the bottom part shows a list of currently configured IPSec tunnels.



General IPSec settings:

NAT Traversal (NAT-T)	Enable NAT traversal support by checking the check box, disable NAT traversal support by un-checking the checkbox. The NAT-traversal setting will apply to all IPSec tunnels. NAT Traversal can cause inter-operability problems with some IPSec clients, so the default setting is disabled. However, when NAT-T is enabled it only kicks in when the server and client detects they are being NAT'ed. So in most cases it is a safe option to set.
MTU Override	Specify the maximum transfer unit for IPSec packets. The setting affects all IPSec tunnels.

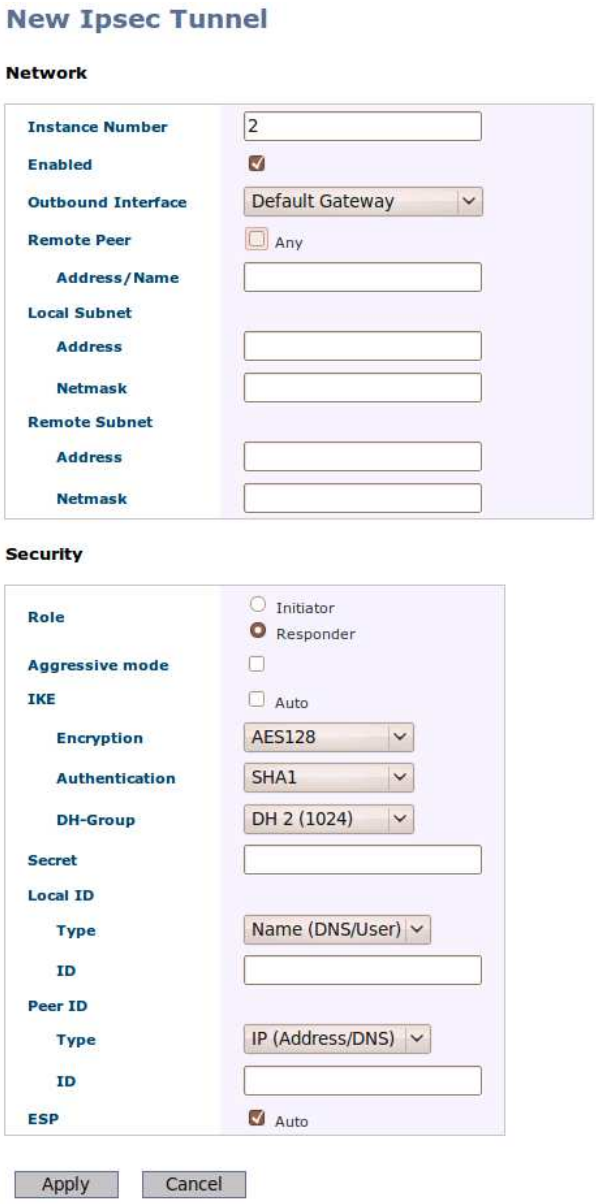
The list shows currently configured IPSec tunnels, and displays some of the tunnel settings.

ID	The IPSec tunnel index. Each configured IPSec tunnel is identified by a number for maintenance purposes. This ID is of local significance only.
Status	Shows whether a specific tunnel is **`Enabled`** or **`Disabled`**.
Remote Peer	The IP address or domain name of the remote peer. **`Any`** is shown if the remote peer is allowed to connect from any IP address.
Outbound Interface	States the outbound interface for this tunnel. The interface can either be stated explicitly (e.g., **`vlan3`**) or implicitly as the interface leading to the **`Default Gateway`**.
Local Subnet	Defines the local subnet. Only traffic from this IP range are allowed to enter the tunnel through this gateway, and traffic arriving through the tunnel is only accepted when destined to an address in this range.
Remote Subnet	Defines the local subnet. Only traffic to this IP range are allowed to enter the tunnel through this gateway, and traffic arriving through the tunnel is only accepted when destined to an address in this range.
Role	Shows whether the VPN gateway is configured as *Responder* or *Initiator* of the VPN tunnel.
Edit	Click this icon to edit the settings of a VPN tunnel.
Delete	Click this icon to remove a VPN tunnel. **Note:** Tunnels which are not intended to be used should either be *deleted* or *disabled* (section 24.2.2).

24.2.2 Configure new IPSec tunnel via the web interface

Menu path: Configuration ⇒ VPN ⇒ IPSec ⇒ **New IPSec Tunnel**

When clicking the **New IPSec Tunnel** button the window to configure a new IPSec tunnel appears.



Network part:

Instance number	The IPSec tunnel index. Each configured IPSec tunnel is identified by a number for maintenance purposes. This ID is of local significance only.
Enabled	A tunnel can be configured as **Enabled** or **Disabled**. **Note:** Tunnels which are not intended to be used should either be *deleted* (section 24.2.1) or *disabled*.
Outbound Interface	The outbound interface for this tunnel. The interface can either be stated explicitly (e.g., **vlan3**) or implicitly as the interface leading to the **Default Gateway**.
Remote Peer Any (Checkbox)	Click the **Any** checkbox if the remote peer can connect from any IP address. This is typically the case if the remote peer is a *road-warrior*, who may use different addresses every time he/she connects. A VPN gateway should only consider setting **Remote Peer** to **Any** if it is acting as **Responder** (i.e., when the remote peer is acting as **Initiator**). Un-check the **Any** checkbox to specify a specific IP address (or domain name) for the remote host, see the item below.
Remote Peer Address/Name	The IP address (e.g., **1.2.3.4**) or domain name (e.g., **foobar.westermo.com**) of the remote peer. This option is required if the node is acting as **Initiator** of the VPN tunnel. This option is only possible to set if the **Any** checkbox is *un-checked*.
Local Subnet Address & Netmask	The **Address** (e.g. **192.168.10.0**) and **Netmask** (e.g., **255.255.255.0**) define the local subnet. Only traffic from this IP range is allowed to enter the tunnel through this gateway, and traffic arriving through the tunnel is only accepted when destined to an address in this range. If no local subnet is specified, only traffic to/from the IP address of the **Outbound Interface** will be allowed through the tunnel.

Continued on next page.

Continued on next page.

Remote Subnet Address & Netmask	The **Address** (e.g. **192.168.11.0**) and **Netmask** (e.g., **255.255.255.0**) define the remote subnet. Only traffic to this IP range is allowed to enter the tunnel through this gateway, and traffic arriving through the tunnel is only accepted when destined to an address in this range. In case the remote peer is a PC (see fig. 24.3), specify the the PC's VPN client IP address (e.g., **192.168.12.49**) as **Address**, and **255.255.255.255** as **Netmask**. If no remote subnet is specified, only traffic to/from the IP address of the **Remote Peer** will be allowed through the tunnel.

Security part:

Role	Configure the VPN gateway to act as *Initiator* or *Responder* of the VPN tunnel.
Aggressive Mode	Configure whether this VPN tunnel should use *aggressive* or *main* mode for the IKE handshake. Checking the **Aggressive mode** checkbox specifies use of *aggressive* mode; un-checking the checkbox means specifies use of *main* mode.
IKE Auto (Checkbox)	The cipher suite to use for the IKE handshake can either be negotiated automatically between the peers, or a specific suite can be configured manually. Check the **Auto** checkbox to specify cipher auto-negotiation; un-check the checkbox to specify an IKE cipher suite manually (see below). Note: Cipher auto-negotiation is only valid with main mode IKE. In case of aggressive mode, a specific IKE cipher suite must be configured (see below).
IKE Encryption, Authentication & DH-Group	Configure the encryption algorithm, message authentication algorithm and Diffie-Hellman group to use for the IKE handshake. This option is only possible to set if the IKE **Auto** checkbox is *un-checked*.
Secret	The pre-shared secret (PSK) password string used to protect the IKE handshake. The password string should consist of at least 8 characters and at most 63 characters. Valid characters are ASCII characters 33-126, except '#' (ASCII 35).

Continued on next page.

	Continued from previous page.
Local ID Type & ID	The identity used by the VPN gateway during the IKE handshake. Typically the **Name(DNS/User** type with a simple ID text string (e.g., **alice**) can be used to identify the VPN gateway. For more details on available identification types and ID values, see section 24.1.2. If **Auto** is selected, the local-id will be of type **IP Address**, using the IP address of the specified **Outbound interface** as identity.
Peer ID Type & ID	The identity used by the peer VPN gateway during the IKE handshake. Typically the **Name(DNS/User** type with a simple ID text string (e.g., **bob**) can be used to identify the peer VPN gateway. For more details on available identification types and ID values, see section 24.1.2. If **Auto** is selected, the **Peer ID** will be of type **IP Address**, using the IP address from the **Remote Peer Address/Name** field as identity (a domain name will be resolved to an IP address).
ESP Auto (Checkbox)	The cipher suite to use for the ESP handshake can either be negotiated automatically between the peers, or a specific suite can be configured manually. Check the **Auto** checkbox to specify cipher auto-negotiation; un-check the checkbox to specify an ESP cipher suite manually (see below). Note: ESP cipher auto-negotiation is only valid with main mode IKE. In case of aggressive mode, a specific ESP cipher suite must be configured (see below).
ESP Encryption & Authentication	Configure the encryption algorithm and message authentication algorithm to use for the ESP handshake. This option is only possible to set if the ESP **Auto** checkbox is *un-checked*.

24.2.3 Edit existing IPSec tunnel via the web interface

Menu path: Configuration ⇒ VPN ⇒ IPSec ⇒  (IPSec Tunnel)

By clicking the **Edit** button in the list of IPSec tunnels, you reach the **Edit IPSec Tunnel** page, as shown below.

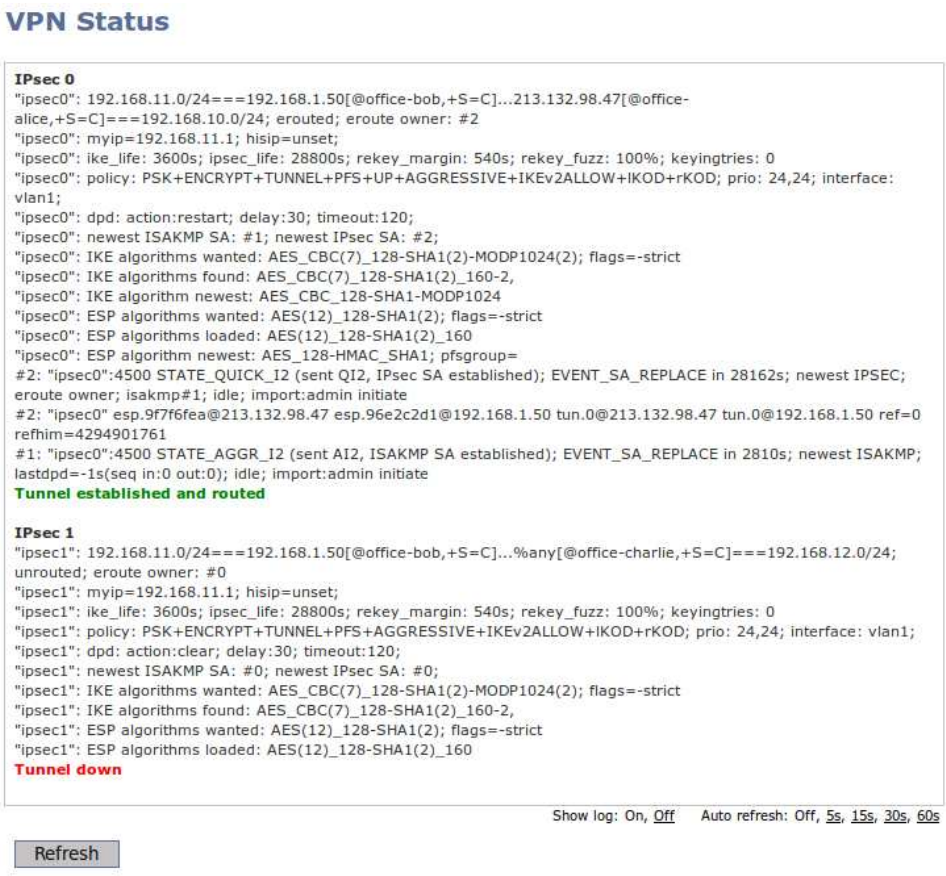
For information on the available configuration items, see section 24.2.2.

24.2.4 View IPSec Tunnel Status

Menu path: Statistics ⇒ VPN

The **IPSec Tunnel Status** page lists the status of configured IPSec tunnels.

Click the **Show log: On** link to see more verbose status information.



24.3 Managing VPN settings via the CLI

The table below shows VPN management features available via the CLI.

Command	Default	Section
Configure VPN Settings		
tunnel		Section 24.3.1
[no] ipsec-nat-traversal	Disabled	Section 24.3.2
[no] ipsec-mtu-override <BYTES>	1419	Section 24.3.3
[no] ipsec <INDEX>		Section 24.3.4
[no] enable	Enabled	Section 24.3.5
[no] aggressive	Main mode	Section 24.3.6
[no] pfs	Enabled	Section 24.3.7
[no] ike crypto <3des|aes128|. . . > auth <md5|sha1> dh <1024|. . . >	Auto	Section 24.3.8
[no] esp crypto <3des|aes128|. . . > auth <md5|sha1>	Auto	Section 24.3.9
[no] secret <PASSWORD>	Empty	Section 24.3.10
[no] peer <IPADDR|FQDN>	Any	Section 24.3.11
[no] outbound <IFACE>	Auto	Section 24.3.12
[no] local-id <inet <IPADDR|DOMAIN>| name <DOMAIN|USER> | email <USER@DOMAIN> | key <ID>>	Auto	Section 24.3.13
[no] remote-id <inet <IPADDR|DOMAIN>| name <DOMAIN|USER> | email <USER@DOMAIN> | key <ID>>	Auto	Section 24.3.14
[no] local-subnet <SUBNET/LEN | SUBNET NETMASK>	Auto	Section 24.3.15
[no] remote-subnet <SUBNET/LEN | SUBNET NETMASK>	Auto	Section 24.3.16
[no] initiator	Responder	Section 24.3.17
[no] dpd-action <clear|hold|restart>	Clear/Restart	Section 24.3.18
[no] dpd-delay <SECONDS>	30	Section 24.3.19
[no] dpd-timeout <SECONDS>	120	Section 24.3.20

Continued on next page.

Continued from next page.

Command	Default	Section
Show VPN Settings		
show tunnel		Section 24.3.21
tunnel		
show ipsec-nat-traversal		Section 24.3.22
show ipsec-mtu-override		Section 24.3.23
show ipsec <ID>		Section 24.3.24
ipsec <ID>		
show enable		Section 24.3.25
show aggressive		Section 24.3.26
show pfs		Section 24.3.27
show ike		Section 24.3.28
show esp		Section 24.3.29
show secret		Section 24.3.30
show peer		Section 24.3.31
show outbound		Section 24.3.32
show local-id		Section 24.3.33
show remote-id		Section 24.3.34
show local-subnet		Section 24.3.35
show remote-subnet		Section 24.3.36
show initiator		Section 24.3.37
show dpd-action		Section 24.3.38
show dpd-delay		Section 24.3.39
show dpd-timeout		Section 24.3.40
Show VPN Status		
show tunnel ipsec [ID]		Section 24.3.41

24.3.1 Managing Tunnels

Syntax `tunnel`

Context *Global Configuration* context

Usage Enter the *Tunnel configuration* context.

Default values Not applicable.

Error messages None defined yet.

24.3.2 Enable/disable IPSec NAT Traversal

Syntax `[no] ipsec-nat-traversal`

Context *Tunnel configuration* context

Usage Enable or disable NAT-T for *all* IPSec tunnels. NAT Traversal can cause inter-operability problems with some IPSec clients, so the default setting is disabled.

However, when NAT-T is enabled it only kicks in when the server and client detects they are being NAT'ed. So in most cases it is a safe option to set.

Use **"ipsec-nat-traversal"** to enable and **"no ipsec-nat-traversal"** to disable NAT traversal.

Default values Disabled (**"no ipsec-nat-traversal"**)

Error messages None defined yet.

24.3.3 Configure IP tunnel MTU

Syntax `[no] ipsec-mtu-override <BYTES>`

Context *Tunnel configuration* context

Usage Override default MTU for *all* IPSec tunnels.

Use **"ipsec-mtu-override <BYTES>"** to specify a specific MTU value to use for all IPSec tunnels. Use **"no ipsec-mtu-override"** to return to the default setting.

Default values 1419 (bytes)

Error messages None defined yet.

24.3.4 Managing IPSec VPN Tunnels

Syntax `[no] ipsec <INDEX>` where INDEX is a number greater or equal to 0.

Context *Tunnel configuration* context

Usage Create, delete, or modify an IPSec VPN tunnel. Use **"ipsec <INDEX>"** to create a new IPSec tunnel, or to enter the configuration context of an existing IPSec tunnel. (To find the index of configured tunnels, use **"show tunnel"** as described in section 24.3.21,)

Use **"no ipsec <INDEX>"** to remove a specific IPSec VPN tunnel, or **"no ipsec"** to remove all configured IPSec VPN tunnels.

Note: *Tunnels which are not intended to be used should either be* deleted *or* disabled *(section 24.3.5).*

Default values Not applicable.

Error messages None defined yet.

24.3.5 Enable/disable an IPSec VPN tunnel

Syntax `[no] enable`

Context *IPSec configuration* context

Usage Enable or disable an IPSec VPN tunnel. A disabled tunnel will be deactivated, but keeps its configuration settings.

Use **"enable"** to enable and **"no enable"** to disable an IPSec VPN tunnel.

Note: *Tunnels which are not intended to be used should either be* deleted *(section 24.3.4) or* disabled.

Default values Enabled

Error messages None defined yet.

24.3.6 IKE phase-1 aggressive or main mode

Syntax `[no] aggressive`

Context *IPSec configuration* context

Usage Select aggressive or main mode for the IKE phase-1 handshake.

Use **"aggressive"** to select aggressive mode, and **"no aggressive"** to select main mode.

Default values Disabled (**"no aggressive"**, i.e., *main* mode is use by default.)

Error messages None defined yet.

24.3.7 Enable/disable Perfect Forward Secrecy

Syntax `[no] pfs`

Context *IPSec configuration* context

Usage Enable or disable Perfect Forward Secrecy for this IPSec tunnel. Protects previous key exchanges even if the current one is compromised.

Note: This setting is not supported by all IPSec implementations. It is however recommended to have it enabled, on both sides of the connection.

If you are unsure what do to, you can safely disable PFS. If the IPSec daemon receives a request with PFS, it will allow it despite how your having disabled it here, because there is absolutely no reason not to use PFS if it is available.

Use **”pfs”** to enable and **”no pfs”** to disable perfect forward secrecy.

Default values Enabled (**”pfs”**)

Error messages None defined yet.

24.3.8 Configure allowed crypto algorithms for IKE phase-1

Syntax `[no] ike crypto <3des|aes128|...> auth <md5|sha1> dh <1024|...>`

Context *IPSec configuration* context

Usage Set IKE phase-1 handshake. Configure what security suite to use to protect the IKE authentication handshake. Here the security suite consists of three parameters:

- *Encryption algorithm:* Supported encryption algorithms are *3des*, *aes128*, *aes256*, and *blowfish*.
- *Message authentication/integrity:* Supported hash algorithms for message authentication are *md5*, and *sha1*.
- *Diffie-Hellman groups:* Supported Diffie-Hellman groups are 1024 (DH group 2), 1536 (DH group 5), 2048 (DH group 14), 3072 (DH group 15) and 4096 (DH group 16).

By specifying an IKE suite, e.g., **”ike crypto aes256 auth sha1 dh 2048”** you will ensure that this suite is used to secure the IKE handshake - if the remote side does not support this suite, the handshake will fail.

Use **”no ike”** to specify the *automatic* security suite negotiation. When configured as an *initiator*, this means that all combinations will be tried (starting by offering a set of suites with either AES-128 or 3DES for encryption, SHA1 or MD5 for authentication, and DH groups 1024, 1536 and 2048). When configured as a *responder* any combination of the listed algorithms will be accepted.

Default values Auto (**”no ike”**)

Note: if *aggressive* mode is selected for the IKE phase-1 handshake, the default security suite for IKE phase-1 negotiation is set to ”AES128-SHA1-DH1024” (**”esp crypto aes128 auth sha1 dh 1024”**).

Error messages None defined yet.

24.3.9 Configure allowed crypto algorithms for ESP

Syntax `[no] esp crypto <3des|aes128|...> auth <md5|sha1>`

Context *IPSec configuration* context

Usage Set IKE Phase-2 hand shake negotiation. Configure what security suite ESP should use to protect the *data traffic* in the established VPN tunnel. Here the security suite consists of two parameters:

- *Encryption algorithm:* Supported encryption algorithms are *3des*, *aes128*, *aes256*, and *blowfish*.
- *Message authentication/integrity:* Supported hash algorithms for message authentication are *md5*, and *sha1*.

By specifying an ESP suite, e.g., **"esp crypto aes256 auth sha1"** you will ensure that this suite is used to secure the data traffic in the established IPSec ESP tunnel. IKE phase-1 handshake - if the remote side does not support this suite, the handshake will fail.

Use **"no esp"** to specify the *automatic* security suite negotiation. When configured as an *initiator*, this means that all combinations will be tried. When configured as a *responder* any combination of the listed algorithms will be accepted.

Default values Auto (**"no esp"**)

Note: if *aggressive* mode is selected for the IKE phase-1 handshake, the default security suite for IKE phase-2 negotiation is set to "AES128-SHA1" (**"esp crypto aes128 auth sha1"**).

Error messages None defined yet.

24.3.10 Configure IPSec Pre-shared Secret

Syntax `[no] secret <PASSWORD>`

Context *IPSec configuration* context

Usage Set pre-shared key (shared secret). The password string should consist of at least 8 characters and at most 63 characters.

Valid characters are ASCII characters 33-126, except '#' (ASCII 35).

Use **"no secret"** to remove a configured pre-shared secret.

Default values Empty

Error messages None defined yet.

24.3.11 Specify IP Address/domain name of remote unit

Syntax `[no] peer <IPADDR|FQDN>`

Context *IPSec configuration* context

Usage Set pre-shared key (shared secret). The password string should consist of at least 8 characters and at most 63 characters.
Valid characters are ASCII characters 33-126, except '#' (ASCII 35).
Use **"no secret"** to remove a configured pre-shared secret.

Default values Empty

Error messages None defined yet.

24.3.12 Configure Outbound Interface

Syntax `[no] outbound <IFACE>`

Context *IPSec configuration* context

Usage Set the outbound interface of this tunnel.
Use **"no outbound"** to automatically select the interface leading to the *default gateway* as outbound interface.
See section 24.1.1 for more information on the outbound interface)

Default values Auto (**"no outbound"**)

Error messages None defined yet.

24.3.13 Configure Local Identifier

Syntax `[no] local-id <inet <IPADDR|DOMAIN> | name <DOMAIN|USER> | email <USER@DOMAIN> | key <ID>>`

Context *IPSec configuration* context

Usage Set the identifier (type and value) for the VPN gateway. The local-id is used by the VPN gateway during the IKE handshake. Typically the **"name"** type with a simple ID text string (e.g., **alice**) can be used to identify the VPN gateway.

For more details on available identification types and ID values, see section 24.1.2.

If **"no local-id"** is selected, the local-id will be of type **"inet"** (IPv4 address), using the IP address of the *Outbound interface* (see section 24.3.12)as identity.

Default values Auto (**”no local-id”**)

Error messages None defined yet.

24.3.14 Configure Remote Identifier

Syntax `[no] local-id <inet <IPADDR|DOMAIN> | name <DOMAIN|USER> | email <USER@DOMAIN> | key <ID>>`

Context *IPSec configuration* context

Usage Set the identifier (type and value) for the peer VPN gateway. The remote-id is used by the peer VPN gateway during the IKE handshake. Typically the **”name”** type with a simple ID text string (e.g., **”bob”**) can be used to identify the peer VPN gateway.

For more details on available identification types and ID values, see section 24.1.2.

If **”no remote-id”** is selected, the **”remote-id”** will be of type **”inet”** (IPv4 address), using the IP address from the configured *Peer* (see section 24.3.11) as identity. A peer domain name will be resolved to an IP address.

Default values Auto (**”no remote-id”**)

Error messages None defined yet.

24.3.15 Configure Local Subnet

Syntax `[no] local-subnet <SUBNET/LEN | SUBNET NETMASK>`

Context *IPSec configuration* context

Usage Set the local subnet of this tunnel.

Only traffic from this IP range is allowed to enter the tunnel through this gateway, and traffic arriving through the tunnel is only accepted when destined to an address in this range.

If **”no local-subnet”** is specified, only traffic to/from the IP address of the *outbound interface* will be allowed through the tunnel.

Default values None (**”no local-subnet”**)

Error messages None defined yet.

24.3.16 Configure Remote Subnet

Syntax `[no] remote-subnet <SUBNET/LEN | SUBNET NETMASK>`

Context *IPSec configuration* context

Usage Set the remote subnet of this tunnel.

Only traffic from this IP range is allowed to enter the tunnel through this gateway, and traffic arriving through the tunnel is only accepted when destined to an address in this range.

In case the remote peer is a PC (see fig. 24.3), specify the the PC's VPN client IP address with a "/32" prefix length, e.g., **"192.168.12.49/32"**.

If **"no remote-subnet"** is specified, only traffic to/from the IP address of the *Peer* will be allowed through the tunnel.

Default values None (**"no remote-subnet"**)

Error messages None defined yet.

24.3.17 Configure Initiator/Responder Setting

Syntax `[no] initiator`

Context *IPSec configuration* context

Usage Select whether the VPN gateway should act as initiator or responder of this IPSec tunnel.

Use **"initiator"** to make the VPN gateway act as *initiator*, and **"no initiator"** to make it act as responder.

Default values Responder (**"no initiator"**)

Error messages None defined yet.

24.3.18 Configure Dead Peer Detection Action

Syntax `[no] dpd-action <clear|hold|restart>`

Context *IPSec configuration* context

Usage Set the DPD action for this VPN gateway. The DPD action defines how the VPN gateway should react when the peer is determined to be unreachable (i.e., "dead").

Use **"no dpd-action"** to disable the DPD mechanism on this VPN gateway. When disabled, this VPN gateway will not probe the peer to check if it is down, however, this VPN gateway will still respond to DPD probing messages

from the peer. That is, it is possible for the peer to the DPD mechanism successfully even though DPD is disabled on this side.

For more information on DPD action settings, see 24.1.5.

Default values This depends on the role of this VPN gateway.

- *Initiator:* If this VPN gateway is the initiator of the tunnel, the DPD action is by default set to *restart* (**”dpd-action restart”**)
- *Responder:* If this VPN gateway is the initiator of the tunnel, the DPD action is by default set to *clear* (**”dpd-action clear”**)

Error messages None defined yet.

24.3.19 Configure Dead Peer Detection Delay

Syntax `[no] dpd-delay <SECONDS>`

Context *IPSec configuration* context

Usage Set the DPD probing interval. The DPD delay is the interval between DPD probing messages sent by this VPN gateway. (The DPD delay setting on the two peers are independent, thus they may differ.)

Use **”no dpd-delay”** to return to the default setting.

Default values 30 (seconds)

Error messages None defined yet.

24.3.20 Configure Dead Peer Detection Timeout

Syntax `[no] dpd-timeout <SECONDS>`

Context *IPSec configuration* context

Usage Set the DPD timeout. If a period corresponding to the DPD timeout elapses without getting any response on the DPD probe messages, the VPN gateway considers the peer to be down.

Use **”no dpd-timeout”** to return to the default setting.

Default values 120 (seconds)

Error messages None defined yet.

24.3.21 Show Overview of Tunnel Settings

Syntax `show tunnel`

Context *Global Configuration* context. Also available as **"show"** command within the *Tunnel configuration* context.

Usage List configured VPN tunnels.

Default values Not applicable.

Error messages None defined yet.

24.3.22 Show IPSec NAT Traversal Setting

Syntax `show ipsec-nat-traversal`

Context *Tunnel configuration* context.

Usage Show whether IPSec NAT traversal is enabled or disabled.

Default values Not applicable.

Error messages None defined yet.

24.3.23 Show IPSec MTU Override Setting

Syntax `show ipsec-mtu-override`

Context *Tunnel configuration* context.

Usage Show the configured IPSec MTU value.

Default values Not applicable.

Error messages None defined yet.

24.3.24 Show IPSec Tunnel Settings

Syntax `show ipsec <ID>`

Context *Tunnel configuration* context. Also available as **"show"** command within the *IPSec configuration* context.

Usage Show all settings of a specific IPSec tunnel.

Default values Not applicable.

Error messages None defined yet.

24.3.25 Show IPSec Tunnel Enable Setting

Syntax `show enable`

Context *IPSec configuration* context.

Usage Show whether this IPSec tunnel is enabled or disabled.

Default values Not applicable.

Error messages None defined yet.

24.3.26 Show IKE Aggressive/Main Mode Setting

Syntax `show aggressive`

Context *IPSec configuration* context.

Usage Show whether this IPSec tunnel is configured to use IKE *aggressive* or *main* mode. **”Enabled”** means *aggressive* mode, while **”Disabled”** means *main* mode.

Default values Not applicable.

Error messages None defined yet.

24.3.27 Show IPSec Perfect Forward Secrecy Setting

Syntax `show pfs`

Context *IPSec configuration* context.

Usage Show whether *perfect forward secrecy* is enabled or disabled for this tunnel.

Default values Not applicable.

Error messages None defined yet.

24.3.28 Show IKE Cipher Suite Setting

Syntax `show ike`

Context *IPSec configuration* context.

Usage Show the configured IKE Cipher suite for this tunnel, i.e., encryption algorithm, message authentication algorithm, and Diffie-Hellman group. **”Auto”** is shown if the VPN gateway is configured to auto-negotiate what IKE cipher suite to use.

Default values Not applicable.

Error messages None defined yet.

Examples The following example show the output when AES-128 is used for encryption, SHA-1 for message authentication, and Diffie-Hellman group 1024.

```
redfox:/config/tunnel/ipsec-0/#> show ike
AES128 SHA1 1024
redfox:/config/tunnel/ipsec-0/#>
```

24.3.29 Show ESP Cipher Suite Setting

Syntax `show esp`

Context *IPSec configuration* context.

Usage Show the configured ESP Cipher suite for this tunnel. **"Auto"** is shown if the VPN gateway is configured to auto-negotiate what ESP cipher suite to use.

Default values Not applicable.

Error messages None defined yet.

24.3.30 Show IKE Pre-shared Secret Setting

Syntax `show secret`

Context *IPSec configuration* context.

Usage Show the configured pre-shared secret (PSK) for this tunnel.

Default values Not applicable.

Error messages None defined yet.

24.3.31 Show IPSec Peer Setting

Syntax `show peer`

Context *IPSec configuration* context.

Usage Show the configured *peer IP address* or *peer domain name*. **"Any"** is shown if the peer can connect from any IP address.

Default values Not applicable.

Error messages None defined yet.

24.3.32 Show IPSec Outbound Interface Setting

Syntax `show outbound`

Context *IPSec configuration* context.

Usage Show the configured *outbound interface* for this tunnel. **”Default Gateway”** is shown if the interface leading to the default gateway should be used as outbound interface.

Default values Not applicable.

Error messages None defined yet.

24.3.33 Show IKE Local Identifier Setting

Syntax `show local-id`

Context *IPSec configuration* context.

Usage Show the configured *local identifier* for this tunnel, i.e., both the local-id *type* and the local-id *value*. **”Auto”** is shown if the local identifier is assigned as type **”inet”** with the IP address of the *outbound interface* (see section 24.3.33) as value.

Default values Not applicable.

Error messages None defined yet.

24.3.34 Show IKE Remote Identifier Setting

Syntax `show remote-id`

Context *IPSec configuration* context.

Usage Show the configured *remote identifier* for this tunnel, i.e., both the remote-id *type* and the remote-id *value*. **”Auto”** is shown if the local identifier is assigned as type **”inet”** with the IP address of the *peer* (see section 24.3.34) as value.

Default values Not applicable.

Error messages None defined yet.

24.3.35 Show IPSec Local Subnet Setting

Syntax `show local-subnet`

Context *IPSec configuration* context.

Usage Show the configured *local subnet* for this tunnel. **”None”** is shown if no local subnet has been configured.

Default values Not applicable.

Error messages None defined yet.

24.3.36 Show IPSec Remote Subnet Setting

Syntax `show local-subnet`

Context *IPSec configuration* context.

Usage Show the configured *local subnet* for this tunnel. **”None”** is shown if no local subnet has been configured.

Default values Not applicable.

Error messages None defined yet.

24.3.37 Show IPSec Initiator/Responder Setting

Syntax `show initiator`

Context *IPSec configuration* context.

Usage Show whether the VPN gateway acts as *Initiator* or *Responder* for this tunnel. configured.

Default values Not applicable.

Error messages

24.3.38 Show IPSec Dead Peer Detection Action Setting

Syntax `show dpd-action`

Context *IPSec configuration* context.

Usage Show the configured DPD action setting.
”off” is shown if DPD has been disabled on this VPN gateway.

Default values Not applicable.

Error messages

24.3.39 Show IPSec Dead Peer Detection Delay Setting

Syntax `show dpd-delay`

Context *IPSec configuration* context.

Usage Show the configured DPD delay setting (in seconds).

Default values Not applicable.

Error messages

24.3.40 Show IPSec Dead Peer Detection Timeout Setting

Syntax `show dpd-timeout`
Context *IPSec configuration* context.
Usage Show the configured DPD timeout setting (in seconds).
Default values Not applicable.
Error messages

24.3.41 Show IPSec Tunnel Status

Syntax `show tunnel ipsec [ID]`
Context *Admin Exec* context.
Usage Show the status for all or for a specific IPSec tunnel.
Default values If no tunnel ID is specified, the status of all tunnels is shown.
Error messages

Chapter 25

DHCP Server

25.1 Overview of DHCP Server Support in WeOS

In WeOS DHCP servers can be configured on every VLAN network interface configured with a static IP address. As of WeOS version v4.3.0 the DHCP server support is limited to handle out IP addresses dynamically from a configurable pool of addresses.

The switch is also able to act as caching name server for the DHCP clients it serves.

As of WeOS version v4.3.0 DHCP server configuration is only available via the CLI.

25.2 Configuring DHCP Server Settings via the CLI

Command	Default	Section
Configure DHCP Server		
[no] dhcp-server <IFACE>		Section 25.2.1
[no] pool <IPADDR_START> <NUM|IPADDR_END>		Section 25.2.2
[no] lease-time <60-5256000>	864000	Section 25.2.3
[no] gateway <IPADDR>	”local”	Section 25.2.4
[no] name-server <IPADDR> ”local”		Section 25.2.5
[no] domain <DOMAINNAME>	”local”	Section 25.2.6
View DHCP Server Settings		
show dhcp-server [<IFACE>]		Section 25.2.7

25.2.1 Manage DHCP Servers

Syntax `[no] dhcp-server <IFACE>`

Context *Global Configuration* context

Usage Create, modify or remove a DHCP Server.

Enter DHCP server context of the given interface. If this is a new DHCP server, the DHCP server is created on the given interface. As a side-effect, a *caching* (DNS) name server is started, which forwards incoming DNS requests to the DNS server configured for the switch (see chapter 17).

Use **"no dhcp-server <IFACE>"** to remove an existing DHCP server, or **"no dhcp-server"** to remove all link aggregates.

Default values When using the **"no dhcp-server"** form (without providing a specific IFACE argument), all DHCP servers are removed.

Error messages None defined yet.

25.2.2 Configure DHCP Server Address Pool

Syntax `[no] pool <IPADDRESS_START> <NUM|IPADDRESS_END>`

Context *DHCP server* context

Usage Specify the IP address pool from which the DHCP server will hand out leases. The *end* of the address range can be specified as an IP address (**"IPADDRESS_END"**), or as a number (**"NUM"**). **"NUM"** specifies the number of addresses in the pool, thus "IPADDRESS_END" is computed as *"IPADDRESS_START + NUM − 1"*.

Default values Not applicable.

Error messages None defined yet.

25.2.3 Configure DHCP Lease Time

Syntax `[no] lease-time <60-5256000>`

Context *DHCP server* context

Usage Specify the DHCP address lease time (seconds) for addresses handed out to DHCP clients.

Use **"no lease-time"** to reset the lease time setting to its default value.

Default values 864000 seconds (i.e., 10 days)

Error messages None defined yet.

25.2.4 Configure DHCP Default Gateway Option

Syntax `[no] gateway <IPADDRESS>`

Context *DHCP server* context

Usage Specify the IP default gateway (default router) option for leases handed to DHCP clients. A single default gateway can be specified. If no default gateway is specified, the switch IP address on this interface will be provided in the default gateway option (that is, the switch will act as default gateway for hosts on this interface). Please remember to enable routing on this (chapter 17 and enable appropriate NAT and firewall rules if necessary (chapter 23).

Use `no gateway` to remove any configured default gateway option.

Default values If no default gateway is specified, the switch IP address on this interface will be provided in the default gateway option.

Error messages None defined yet.

25.2.5 Configure DHCP Name Server Option

Syntax `[no] name-server <IPADDRESS>`

Context *DHCP server* context

Usage Specify the (DNS) name server option for leases handed to DHCP clients. A single DNS name server can be specified. If no name-server is specified, the switch IP address on this interface will be provided in the name server option (that is, the switch will act as DNS name server for hosts on this interface. In this case, the switch will act as a caching name server and forward any (non-cached) incoming requests to the name-server configured on the switch, see chapter 17).

Use `no name-server` to remove any configured name server DHCP option.

Default values If no name-server is specified, the switch IP address on this interface will be provided in the name server option.

Error messages None defined yet.

25.2.6 Configure DHCP Domain Name Option

Syntax `[no] domain <DOMAIN>`

Context *DHCP server* context

Usage Specify the domain name search path option for leases handed to DHCP clients. A single domain name option can be specified. If no name-server is specified, the switch IP address on this interface will be provided in the name server option (that is, the switch will act as DNS name server for hosts on this interface).

Use `no name-server` to reset the DNS server to remove any configured name server DHCP option.

Default values If no name-server is specified, the switch IP address on this interface will be provided in the name server option (that is, the switch will act as DNS name server for hosts on this interface).

Error messages None defined yet.

25.2.7 Show DHCP Server Settings

Syntax `show name-server [<IFACE>]` Also available as **"show"** command within the DHCP server context.

Context *Global Configuration* context

Usage Show DHCP server settings for the DHCP server on the specified interface. If no interface argument is provided, a summary of all configured DHCP servers is shown.

Default values If no interface argument is provided, a summary of all configured DHCP servers is shown.

Error messages None defined yet.

Chapter 26

Ethernet Statistics

A set of per port Ethernet statistic counters are available via the Web and via the CLI. Most of these counters correspond to standard SNMP MIB Ethernet statistics counters from the RMON MIB (RFC 2819), the Interface MIB (RFC 2863) and the Ether-Like MIB (RFC 3635)[1]. For more information about WeOS SNMP support, see chapter 29.

Section 26.1 gives a general introduction to the Ethernet statistic counters available via Web and CLI. Sections 26.2 and 26.2 present use of Ethernet statistics via the Web and CLI respectively.

26.1 Ethernet Statistics Overview

Table 26.1 provides a summary of the available Ethernet statistics counters. Sections 26.1.1-26.1.8 give more detailed information on the meaning of these counters.

26.1.1 Inbound Byte Counters

A set of byte counters (i.e., octet counters) are provided. The number of *good* bytes is also used to compute a rough estimation of the current inbound data rate.

Bytes Good The number of *good bytes/octets* received on a port, i.e., the sum of the length of all good Ethernet frames received.

[1]The Ether-Like MIB is currently not supported in WeOS.

Feature	Web	CLI	Description
Inbound			
Total Bytes	X	(X)	Sec. 26.1.1
Bytes Good		X	”
Bytes Bad		X	”
Mean rate		X	”
Total Good Packets		(X)	Sec. 26.1.2
Unicast	X	X	”
Multicast	X	X	”
Broadcast	X	X	”
Pause frames		X	”
Size statistics	X		”
Dropped	X	X	Sec. 26.1.3
Filtered		X	”
Discarded		X	”
Erroneous		(X)	Sec. 26.1.4
Undersize	X	X	”
Oversize	X	X	”
Fragments	X	X	”
Jabber	X	X	”
Checksum	X	X	”
PHY Error		X	”
Outbound			
Total Bytes	X	X	Sec. 26.1.5
Mean rate		X	”
Total Packets	(X)	(X)	Sec. 26.1.6
Unicast	X	X	”
Multicast	X	X	”
Broadcast	X	X	”
Pause frames		X	”
Dropped			Sec. 26.1.7
Filtered		X	”
Collisions and Busy Medium	X	(X)	Sec. 26.1.8
Single		X	”
Multiple		X	”
Excessive		X	”
Late	X	X	”
Other collisions		X	”
Deferred		X	”

Table 26.1: Summary of Ethernet statistics counters. Counters listed within parenthesis (i.e., as ’(X)’) are provided implicitly.

Bytes Bad The number of *bad bytes/octets* received on a port, i.e., the sum of the length of all bad Ethernet frames received.

Total Bytes The sum of good and bad bytes received on a port (see above). This would correspond to the RMON MIB *etherStatsOctets* and the Interface MIB *ifHCInOctets* objects.

Mean Rate Rough estimation of the current data rate based on the number of good bytes received during a time interval (2 seconds).

26.1.2 Inbound Counters of Good Packets

The following per port counters for *good* inbound Ethernet packets are provided.

Unicast packets The number of *good* packets with a unicast MAC address received on the port.

This would correspond to the Interface MIB *ifInUcastPkts* object.

Multicast packets The number of *good* packets with a group MAC address (excluding broadcast) received on the port.

This would correspond to the RMON MIB *etherStatsMulticastPkts* and the Interface MIB *ifInMulticastPkts* objects, except that *Pause frames* (see below) are not included.

Broadcast packets The number of *good* packets with a broadcast MAC address received on the port.

This would correspond to the RMON MIB *etherStatsBroadcastPkts* and the Interface MIB *ifInBroadcastPkts* objects.

Pause Frames The number of *good* flow control packets received.

Packet Size Statistics Counters for good Ethernet packet of the following size intervals are provided: 64 bytes, 65-127 bytes, 128-255 bytes, 256-511 bytes, 512-1023 bytes, and 1024-MAXPKTSIZE bytes, where MAXPKTSIZE is 1632.

These size intervals match the corresponding RMON statistics counters, except for the MAXPKTSIZE (1632 instead of 1518).

26.1.3 Dropped Inbound Packets

Counters for two types of dropped inbound packets are provided. Note, these packets are *good* Ethernet packets, but are dropped due to the reasons given below.

Filtered Inbound packets dropped due to VLAN mismatch or because the port was in LEARNING, LISTENING or BLOCKING state.

Discarded Packets dropped due to lack of buffer space.

26.1.4 Erroneous Inbound Packets

The following counters for received erroneous packets are provided:

Undersized packet Number of packets smaller than 64 bytes, and with a valid FCS.
This corresponds to the RMON MIB *etherStatsUndersizePkts* object.

Oversized packet Number of packets larger than 1632 bytes, and with a valid FCS.
This corresponds to the RMON MIB *etherStatsOversizePkts* object, except for the used MAXPKTSIZE (1632 instead of 1518 bytes).

Fragmented packet Number of packets smaller than 64 bytes, with an *invalid* FCS.
This corresponds to the RMON MIB *etherStatsFragments* object.

Jabber Number of packets larger than 1632 bytes, and with an *invalid* FCS.
This corresponds to the RMON MIB *etherStatsJabbers* object, except for the used MAXPKTSIZE (1632 instead of 1518 bytes).

Checksum/FCS Error Packets of valid length (64-1632), but with an incorrect FCS.
This corresponds to the RMON MIB *etherStatsCRCAlignErrors* object, except for the used MAXPKTSIZE (1632 instead of 1518 bytes).

PHY Error Signal Number of received packets generating a *receive error* signal from the Ethernet PHY. (Referred to as *InMacRcvErr* in the CLI port statistics list)

26.1.5 Outbound Byte Counters

A single outbound byte/octet counter, **Outbound Bytes**, is provided. It represents the sum of the length of all Ethernet frames sent on the port.

This would correspond to the Interface MIB *ifHCOutOctets* object.

The number of **Outbound bytes** is also used to calculate a rough estimation of the current sending data rate (**Mean Rate**, i.e., the number of bytes sent during a time interval (2 seconds).

26.1.6 Outbound Packets Counters

The following per port counters for outbound Ethernet packets are provided.

Unicast packets The number of packets with a unicast destination MAC address sent on the port.

This would correspond to the Interface MIB *ifOutUcastPkts* object.

Multicast packets The number of packets with a group destination MAC address (excluding broadcast) sent on the port.

This would correspond to the Interface MIB *ifOutMulticastPkts* objects, except that *Pause frames* (see below) are not included.

Broadcast packets The number of packets with a broadcast destination MAC address sent on the port.

This would correspond to the Interface MIB *ifOutBroadcastPkts* objects.

Pause Frames The number of flow control packets sent.

26.1.7 Dropped Outbound Packets

The counter for a single type of dropped outbound packets is described here (there is also a second kind, see *excessive collisions* in section 26.1.8).

Filtered Outbound packets dropped outbound policy rules or because the port was in LEARNING, LISTENING or BLOCKING state.

26.1.8 Outbound Collision and Busy Medium Counters

The collision and busy medium counters described here are only relevant for half-duplex links.

Single Collisions The number of packets involved in a single collision, but then sent successfully.

This would correspond to the Ether-like MIB *dot3StatsSingleCollisionFrames* object.

Multiple Collisions The number of packets involved in more than one collision, but finally sent successfully.
This would correspond to the Ether-like MIB *dot3StatsMultipleCollisionFrames* object.

Excessive Collisions The number of packets failing (i.e., dropped) due to excessive collisions (16 consecutive collisions).
This would correspond to the Ether-like MIB *dot3StatsExcessiveCollisions* object.

Late Collisions The number of collisions detected later than a *512-bits time* into the packet transmission.
This would correspond to the Ether-like MIB *dot3StatsLateCollisions* object.

Other Collisions Other collisions than *single*, *multiple*, *excessive* or *late* collisions discovered on a port.

Total Collisions Computed as the sum of *single*, *multiple*, *excessive*, *late* and *other* collisions.

Deferred (busy medium) The number of packets experiencing a busy medium on its first transmission attempt, and which is later sent successfully, and without experiencing any collision.
This would correspond to the Ether-like MIB *dot3StatsDeferredTransmissions* object.

26.2 Statistics via the web interface

Statistics shown in the web administration tool has two views. An *overview* with a selection of statistics for all ports, including some status information (e.g. if port is blocking or forwarding), and a *detailed* page with a larger set of statistics.

Note that collection of statistics is started by the first access to the statistics page, and will be halted after a short period of time (to save resources) if no one requests the statistic data. This has the effect that you may need to enter the page once again, by e.g. clicking the menu item, to ensure you are presented to updated statistics data.

26.2.1 Statistics Overview

Menu path: Statistics⇒Port

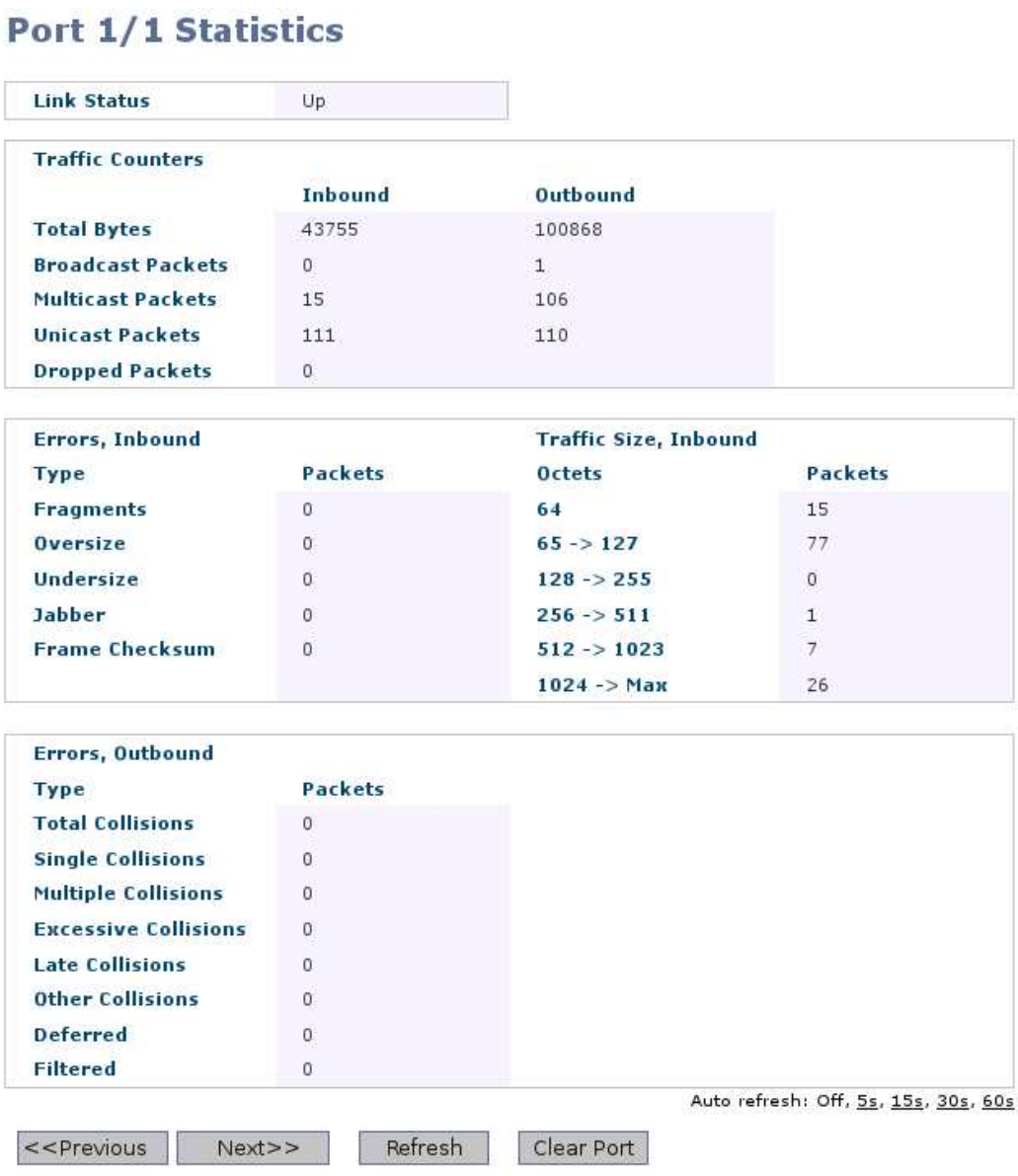
On the port statistics overview page you will be presented to a selection of static data for each port. Additional statistic numbers are presented on the detailed view page.

Alarm	An alarm icon appears at the start of a line if there is a link alarm on a port.
Port	The port label.
Link	The status of the link. Up or down.
State	**FORWARDING** Unit forwards packets. Normal operation. **LEARNING** The port is preparing itself for entering FORWARDING state. **BLOCKING** Unit does not forward any packets. **DISABLED** Port does not participate in operation.
Speed / Duplex	The current speed and duplex negotiated or set on the port.
Total Bytes In	Total number of bytes received on the port.
Total Bytes Out	Total number of bytes sent out on the port.
FCS Errors	Total number of inbound packets with check sum error received on the port.
Details	Click this icon to view more detailed statistics for the port.
Auto Refresh	Click on a value to make the page reload with updated statistics automatically every 5, 15, 30 or 60 seconds. Click Off to turn off auto refresh.
Refresh	Click on this button to reload with updated statistics.
Clear All	Clear all statistics counters for all ports.

26.2.2 Detailed Statistics

Menu path: Statistics ⇒ Port ⇒ 

When clicking the *details*-icon in the overview page you will be presented to the detailed statistics page for the port.



Total Bytes	Total number of bytes received (inbound) or transmitted (outbound) on this port.
Broadcast Packets	Total number of good broadcast packets received (inbound) or transmitted (outbound) on this port.
Multicast Packets	Total number of good multicast packets received (inbound) or transmitted (outbound) on this port.
Unicast Packets	Total number of good unicast packets received (inbound) or transmitted (outbound) on this port.
Dropped Packets	Total number of packets received that have been discarded.
Fragments	Total number of fragmented packets received on this port.
Oversize	Total number of oversized packets received on this port.
Undersize	Total number of undersized, but otherwise well formed, packets received on this port.
Jabber	Total number of packets received on this port larger than the network segment's maximum transfer unit (MTU).
Frame Checksum	Total number of packets received on this port with checksum error.
Traffic Size, Inbound	Number of octets received in different size categories.
Total Collisions	Total number of collisions detected on this port (sum of *single*, *multiple*, *excessive*, *late*, and *other* collision counters).
Single Collisions	The number of packets involved in a single collision, but then sent successfully.
Multiple Collisions	The number of packets involved in more than one collision, but finally sent successfully.
Excessive Collisions	The number of packets failing (i.e., dropped) due to excessive collisions (16 consecutive collisions).
Late Collisions	The number of collisions detected later than a *512-bits time* into the packet transmission.
Other collisions	Other collisions than *single*, *multiple*, *excessive* or *late* collisions discovered on a port.
Deferred	The number of packets experiencing a busy medium on its first transmission attempt, and which is later sent successfully, and without experiencing any collision.
Filtered	Outbound packets dropped outbound policy rules or because the port was in LEARNING, LISTENING or BLOCKING state.

Continued on next page

Auto Refresh	Click on a value to make the page reload with updated statistics automatically every 5, 15, 30 or 60 seconds. Click Off to turn off auto refresh.
«Previous	Goto statistics for previous port.
Next»	Goto statistics for next port.
Refresh	Click on this button to reload with updated statistics.
Clear Port	Clear all statistics counters for the port shown.

26.3 Statistics via the CLI

The table below shows statistic features available via the CLI.

Command	Default	Section
rmon		Section 26.3.1
statistics [PORT]		Section 26.3.2
clear-stats [PORT]		Section 26.3.3
show rmon [PORT]		Section 26.3.4

26.3.1 Managing Ethernet Statistics

Syntax `rmon`

Context *Admin Exec* context

Usage Enter Ethernet statistics context (RMON context). WeOS starts gathering statistics when this command is issued, thus there is a 2 seconds delay before the RMON context is entered.

Default values Not applicable.

Error messages None defined yet.

26.3.2 List Current Ethernet Statistics

Syntax `statistics [PORT]`

Context *RMON* context

Usage Show Ethernet statistics. If no PORT is given (**"statistics"**, a summary of statistics for all Ethernet ports is presented.

If a PORT is given as argument (e.g., **"statistics 1/1"**) detailed statistics for that port is presented.

For information about what the different statistics counters represent, see section 26.1.

Default values If no PORT argument is given, a summary of statistics for all Ethernet ports is presented.

Error messages None defined yet.

26.3.3 Clear Ethernet Statistics

Syntax `clear-stats [PORT]`

Context *RMON* context

Usage Clear Ethernet statistic counters. If no PORT is given (**”clear-stats”**, counters for all Ethernet ports are cleared.

If a PORT is given as argument (e.g., **”clear-stats 1/1”**) the counters for that port are cleared.

Default values If no PORT argument is given, counters for all Ethernet ports are cleared.

Error messages None defined yet.

26.3.4 Show Ethernet Statistics

Syntax `show rmon [PORT]`

Context *Admin Exec* context. Also available as **”show [PORT]”** command within the RMON context.

Usage Show Ethernet statistics. This command provides the same information as the **”statistics”** command (section 26.3.2). The only difference is that the **”show rmon [PORT]”** command is available from the Admin Exec context.

If no PORT is given (**”show rmon”**, a summary of statistics for all Ethernet ports is presented.

If a PORT is given as argument (e.g., **”show rmon 1/1”**) detailed statistics for that port is presented.

For information about what the different statistics counters represent, see section 26.1.

Default values If no PORT argument is given, a summary of statistics for all Ethernet ports is presented.

Error messages None defined yet.

Chapter 27

Alarm handling, Front panel LEDs and Digital I/O

This chapter describes WeOS features for alarm and event handling (sections 27.1-27.3). The chapter also covers general information on functionality related to *Digital I/O* and *front panel LEDs* (sections 27.4 and 27.5).

27.1 Alarm handling features

The table below summarises the WeOS alarm handling features.

Feature	Web (Sec. 27.2)	CLI (Sec. 27.3)	General Description
Configure alarm triggers	X	X	Sec 27.1.1-27.1.3
Configure alarm actions	X	X	Sec 27.1.1 and 27.1.4
Configure alarm targets	X	X	Sec 27.1.1 and 27.1.5
View alarm status[1]	X	X	Sec 27.1.5

27.1.1 Introduction to the WeOS alarm handling support

The WeOS alarm handling support makes use of the following terminology:

- *Alarm sources:* An *alarm source* is an object being monitored by an *alarm trigger*, e.g., the link status (up/down) of an Ethernet port, the input byte

[1] In addition to monitoring alarm status via Web and CLI, there are other ways in which an operator can get notified when an alarm is triggered.

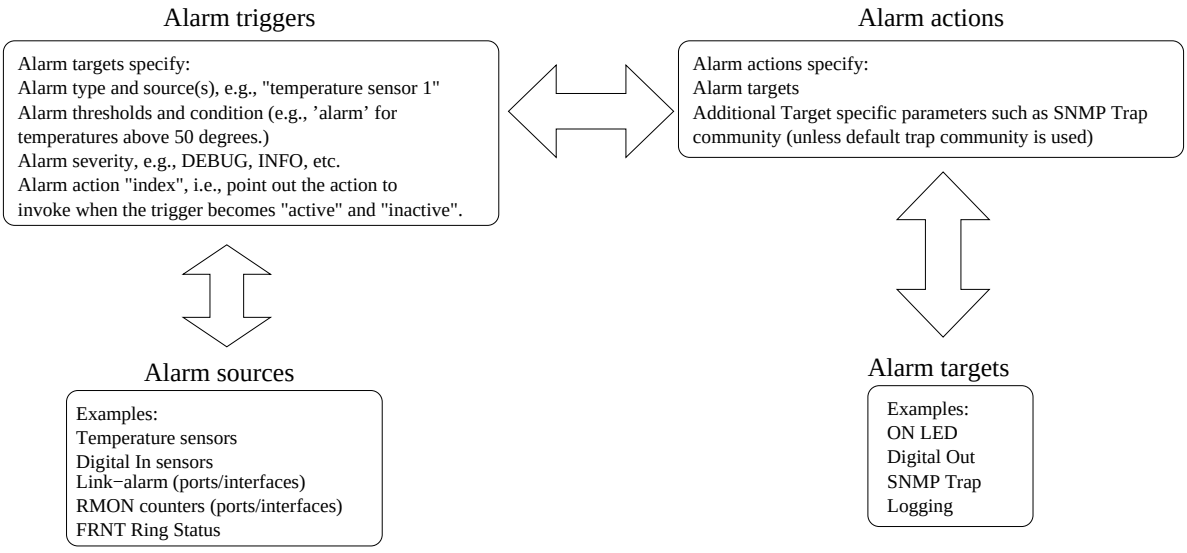


Figure 27.1: Overview of WeOS alarm entities: Alarm triggers monitor the state of alarm source, and define conditions and thresholds when to invoke an associated alarm action. The invoked alarm action specifies what alarm target(s) to use to notify the operator.

counter of a network interface, or the temperature value of a temperature sensor. Alarm sources are described further in section 27.1.2.

- *Alarm trigger:* An *alarm trigger* monitors alarm sources, and defines the conditions when alarm events occur, i.e., when the trigger becomes *active* (alarm situation) or *inactive* (normal situation).

  In addition, the alarm trigger specifies the *alarm action* to be invoked once an alarm event occurs. Alarm triggers are described further in section 27.1.3.

- *Alarm actions and alarm targets:* When an alarm event occurs, the operator can be notified via SNMP traps, logging, digital-out, and front panel status LED. These notification mechanisms are referred to as *alarm targets*.

  Instead of mapping triggers directly to targets, a trigger is mapped to an *alarm action (profile)*. The alarm action defines what specific targets to use when an alarm event occurs. For example, a link alarm trigger for ports 1/1-1/2 can be mapped to a specific alarm action, which in turn specifies *logging* and *SNMP traps* as targets. Alarm actions and targets are described further in sections 27.1.4 and 27.1.5 respectively.

27.1.2 Alarm sources

As of WeOS v4.3.0 the following alarm sources are supported:

- *Power failure:* If the unit is equipped with redundant power feed (or redundant power supply), an alarm can be triggered if one of the feeds lack input power.

  Note: if all power is lacking on all feeds, the unit is powerless and cannot trigger alarms via SNMP traps or remote logging. To detect such a situation remotely, the operator could *poll* the unit (e.g., by *pinging* the unit on a regular interval). The drawback is that it is difficult to distinguish problems in the intermediate network from problems in the monitored device.

  An alternative is to use out-of-band signalling, e.g., via GPRS equipment connected to *digital-out* to get an alarm notification instantly if a device goes down.
- *Link alarm:* It is possible to configure link alarm triggers to react when a link goes down (and up).
- *Digital-In:* Alarms can be triggered depending on the presence of input voltage/current on the *Digital-In* pins of the Digital I/O connector.
- *SHDSL SNR Margin:* On devices with SHDSL ports, alarms can be triggered when the SNR margin falls below some configured threshold.
- *Temperature sensor alarms:* Temperature alarm triggers can be configured to react when the temperature rises above (or falls below) some defined threshold.
- *FRNT status:* The FRNT ring status trigger will react when an FRNT ring is broken (bus mode) or healed (ring mode)[1].
- *Hardware failure:* Hardware alarms triggers notifies that the unit has detected a hardware failure (typically if an unsupported SFP is inserted).

27.1.3 Alarm triggers

An alarm trigger defines the rules for when alarm events should be generated for a monitored alarm source. Alarm triggers also define which *alarm action* to invoke when an alarm event occurs.

The alarm trigger classes currently supported are[2]:

- Power failure

[1] Only an FRNT focal point can determine the ring status with certainty.
[2] As of WeOS v4.3.0 there is no support for SNMP traps for *hardware* or *temperature* alarms.

- Link alarm
- Digital-In
- Temperature
- FRNT ring status
- Hardware failure
- SNR margin (SHDSL)

As the WeOS alarm handling support is designed to include triggers for additional alarm sources, the following description is of more general nature, thus contains more options than needed for the trigger types currently supported.

Note: *As of WeOS v4.3.0 there is no support for making an alarm trigger* persistent*. When an alarm condition is no longer fulfilled, the trigger status will become* inactive*. As alarms are not persistent, it is not possible for an operator to clear (i.e., acknowledge) an alarm.*

27.1.3.1 Specifying what alarm source(s) a trigger should monitor

Different types of alarm triggers operate on different types of alarm sources:

- Power failure: A power failure trigger can monitor one or more power feed sensors. The Westermo devices running WeOS today have two power feeds (single power supply), with a sensor for each power feed. Typically a single power failure trigger is used to monitor both power feed sensors.
- Digital-In: A digital-in trigger can monitor one or more digital-in sensors. The Westermo devices running WeOS today have at most one digital-in sensor.
- Link alarm: Link alarm triggers monitor the operational status (up/down) of Ethernet or DSL ports. Thus when configuring a link alarm trigger the port (or ports) to monitor should be specified.

  Note: *It is possible to define multiple link alarm triggers, where each trigger can monitor different ports and be mapped to different alarm actions.*

  In the future, link alarm triggers can be extended to monitor the operational status of *network interfaces* and *VLANs* in addition to physical ports (Ethernet and SHDSL).
- SNR Margin: An SNR Margin trigger applies to one or more SHDSL ports.

- RMON statistics (not yet supported): The alarm source for an RMON trigger is specified by two parameters: (1) the name of the statistics counter (e.g., *etherStatsPkts*), and (2) the port (or list of ports) for which this counter should be monitored.

  Note: *In WeOS the term RMON is used to refer to data traffic statistics in general; not only to the Ethernet statistics defined in the RMON MIB. Thus, if a counter from the IF-MIB (such as* ifHCInUcastPkts *is specified, the alarm source could refer to* network interfaces *or* VLANs *as well as a physical ports (Ethernet and SHDSL).*

- Temperature (not yet supported): Temperature triggers can apply to one or more temperature sensors.

Typically there would be no more than one trigger monitoring the status of a specific alarm source. However, in some cases it would make sense to have multiple triggers monitoring a single alarm source. For example, one could define two temperature triggers for a single temperature sensor, where one trigger reacts if the temperature rises above a *warning threshold* (say 60°C), and the other if the temperature gets *critically high* (say 75°C).

27.1.3.2 Alarm thresholds and trigger output

For the trigger to know when an alarm event has occurred, threshold values for the monitored alarm sources must be configured. Alarm sources which are 'binary' to their nature (link up/down, power up/down, digital-in high/low, etc.) have thresholds defined *implicitly*.

For sources which can take values in a wider range (temperature, SNR Margin, received packets within a given time interval, etc.) the alarm thresholds should be *configured*. Fig. 27.2a) illustrates use of alarm thresholds for a temperature trigger.

As can be seen in fig. 27.2a), two thresholds are used – a *rising* threshold and a *falling* threshold. Alarm events will be generated when reaching the rising threshold on the way up, and the falling threshold on the way down. However, once a rising alarm event has occurred, a new rising alarm event cannot be generated (for that alarm source) before the value has fallen down to the falling threshold (and vice versa). Thus, the use of separate rising and falling thresholds creates a *hysteresis* mechanism, which avoids generating multiple alarm events when a monitored value fluctuates around the alarm threshold.

Alarm targets such as *Digital-Out* and the *ON LED* provide a summary alarm function (see section 27.1.5.1), and these targets assume that every alarm trig-

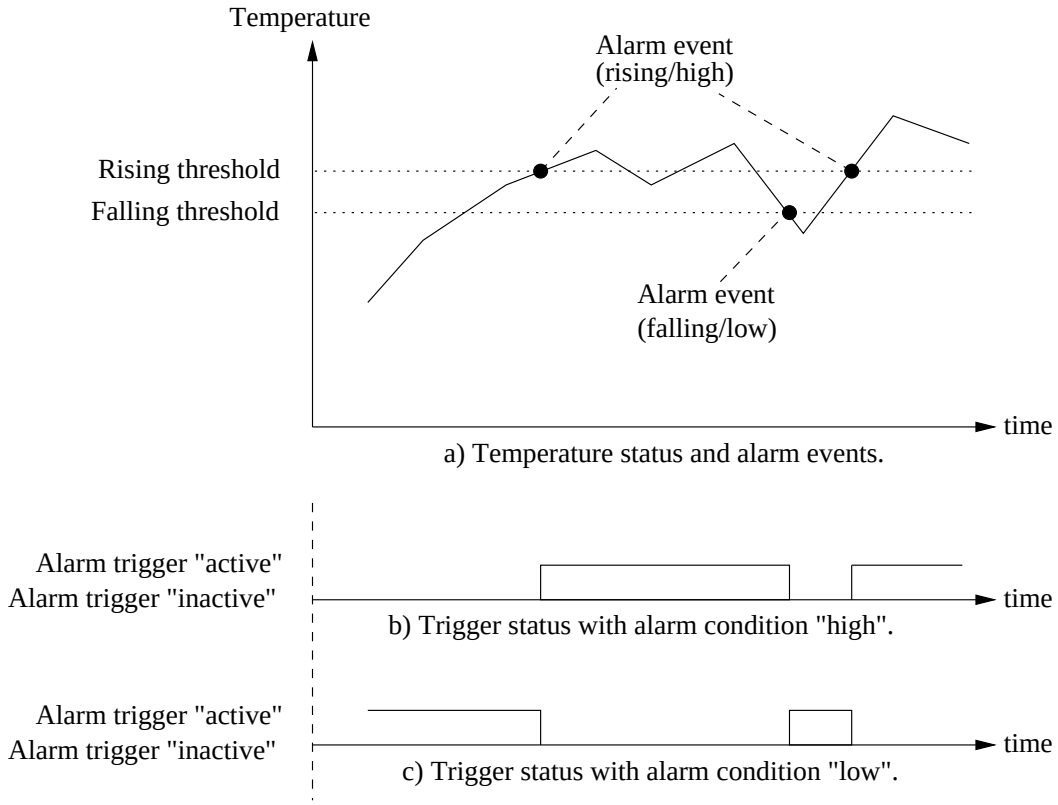


Figure 27.2: Example use of rising and falling thresholds for a temperature alarm trigger (a), and alarm condition setting to affect active and inactive trigger status (b and c).

ger define the condition when the alarm is *active* (”alarm” situation) and *inactive* (”normal” situation). To define this the alarm *condition* configuration option is used. To warn the operator for high temperatures, the alarm condition should be set to ”high”, see fig 27.2b). If we instead wish to warn the operator for low temperatures, the alarm condition should be set to ”low”, see fig 27.2c). A corresponding example for a *Digital-In* trigger is shown in fig. 27.3.

Additional details on threshold settings and properties:

- The rising threshold cannot be set lower than the falling threshold.
- It is possible to use the same value for the rising and falling thresholds.
- Rising alarm events occur if the current sample value is equal or above the rising threshold, and the previously sampled value was below the ris-

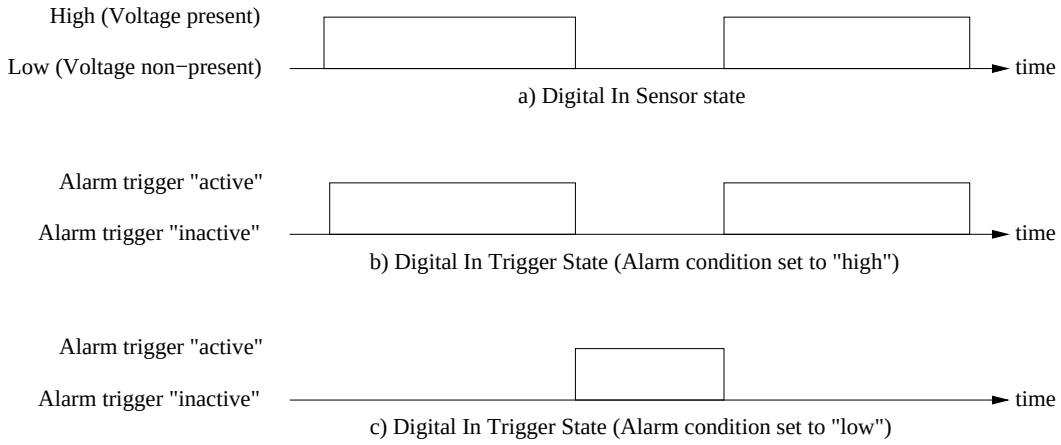


Figure 27.3: Alarm condition example: The alarm trigger for digital-in can be configured to become active when the signal is high (b) or when it is low (c).

ing threshold. A rising alarm event will also occur if the *first* sampled value is equal or above this threshold, and the *condition* variable is configured as *rising* (or any of its equivalents: *high* or *up*).

- Falling alarm events occur if the current sample value is equal or below the falling threshold, and the previously sampled value was above the falling threshold. A falling alarm event will also occur if the *first* sampled value is equal or below this threshold, and the *condition* variable is configured as *falling* (or any of its equivalents: *low* or *down*).

27.1.3.3 Sample types and interval

Two sample types are possible: *absolute* and *delta* sampling. With absolute sampling, the value is compared directly to the alarm thresholds. With delta sampling it is the difference between the current sample and the previous sample which is compared to the alarm thresholds.

Alarm sources of *counter* type, such as RMON data traffic statistics, are well suited for delta sampling. As the delta is computed over a given time interval (sample interval), the alarm thresholds should be configured with respect to the configured sample interval.

Note: *As of WeOS v4.3.0 only absolute sampling is supported, and the sampling interval is not configurable for any trigger type.*

27.1.3.4 Alarm severity

For each trigger it is possible to define the severity level of the associated alarm events. The levels defined by Unix Syslog are used:

- EMERG: System is unusable
- ALERT: Action must be taken immediately
- CRIT: Critical conditions
- ERR: Error conditions
- WARNING: Warning conditions
- NOTICE: Normal, but significant, condition
- INFO: Informational message
- DEBUG: Debug-level message

It is also possible to configure severity level ”NONE”. Alarm events with severity NONE will not cause SNMP traps to be sent or events to be logged, however, such events can still affect digital-out and ON LED targets.

Note: *Severity levels can be configured independently for the events when an alarm trigger becomes ”active” and ”inactive”. Default severity level are WARNING for ”active” alarm events and NOTICE for ”inactive” alarm events.*

27.1.3.5 Mapping triggers to actions

Each trigger specifies which alarm action (profile) to invoke when an alarm event occurs, see 27.1.4 for more information.

27.1.4 Alarm actions - mapping triggers to targets

Instead of mapping triggers directly to alarm targets, each trigger is mapped to an alarm action (alarm action profile). The alarm action specifies which targets to use (SNMP traps, Logging, ON LED, and Digital-Out) when an alarm event occurs.

In addition, the alarm action can specify target specific parameters shared by all triggers mapped to this alarm, e.g., if a specific *SNMP trap community* is to be used.

It is possible to configure several actions (action profiles). Each trigger can be mapped to an individual action, but it is also possible for multiple triggers to

share the same action. This can be particularly useful when managing several triggers of similar type, such as different types of RMON triggers.

By default a trigger is mapped to the *default alarm action* (index 1). The default alarm action cannot be removed.

27.1.5 Alarm presentation (alarm targets)

When an alarm situation occurs, such as a FRNT ring failure, WeOS enables the operator to be notified in numerous ways:

- *SNMP trap:* Alarms can be configured to generate SNMP traps[3]. See chapter 29 for general information on SNMP.
- *Log files and remote logging:* Alarms can be logged locally or passed to a remote logging server. See chapter 28 for general information on event and alarm logging.
- *Digital-Out:* On units equipped with a *Digital I/O* contact, the *Digital-Out* pins can be used as an *alarm target*. Similar to the 'ON' LED, digital-out provides a *summary alarm* function, where the 'gate' is *closed* when the switch is operating 'OK', and *open* when any of the associated alarm triggers becomes active (or when the unit has no power).

  See section 27.4 for general information on Digital I/O.
- *'ON' LED:* There are front panel LEDs which can indicate status of specific ports or protocols. There is also a *general* status LED, which shows a *green* light when the unit is operating 'OK', but shows a *red* light as soon as any of the associated alarm triggers becomes active. Thus, the 'ON' LED provides a *summary alarm* function.

  See section 27.5 for general information on front panel LEDs.

In addition, an operator can view the alarm status via the Web and CLI interfaces.

27.1.5.1 Summary alarm

The *summary alarm* in use by the *digital-out* and *ON LED* targets assumes that every alarm trigger define the condition when the alarm is *active* ("alarm" situation) and *inactive* ("normal" situation).

- For many triggers this definition is implicit, e.g., a link alarm is active when the port (or interface) is *down* and inactive it is *up*.

[3] As of WeOS v4.3.0 there is no support for SNMP traps for *hardware* or *temperature* alarms.

- Other triggers, such as temperature or digital-in sensor triggers allow for the operator to define if the alarm is active: high or low temperature, voltage signal present or not present, etc. See section 27.1.3.2, and in particular figures 27.2) and 27.3, for further information on the *active* and *inactive* trigger states.

Working as a summary alarm, digital-out as well as the ON LED will indicate 'alarm' as soon as any of the associated alarm triggers become active. For the ON LED alarm is indicated with a 'red' light, as shown in fig. 27.4. For Digital-Out, alarm is indicated by having the gate in 'open' state. See sections 27.4 and 27.5 for general information on Digital I/O and front panel LEDs.

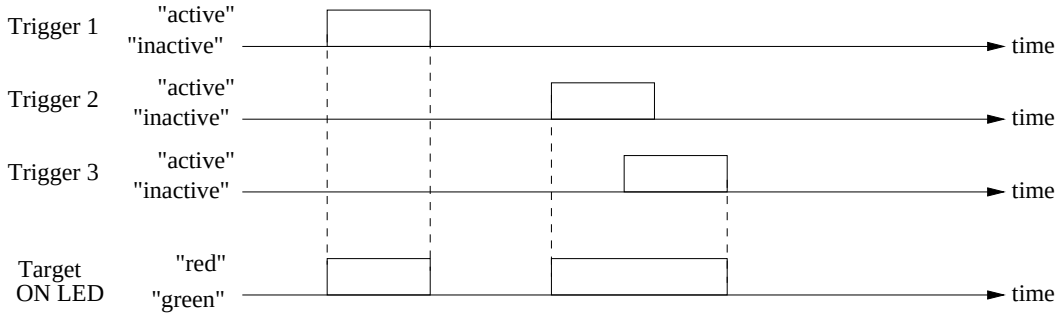


Figure 27.4: Summary alarm example with three alarm triggers mapped to the ON LED alarm target. The ON LED indicates 'alarm' (red) when any of the associated triggers are active.

27.1.5.2 Target Severity thresholds

As of WeOS v4.3.0 setting target severity thresholds is not yet supported.

For *logging* and *SNMP trap* targets it is possible to filter alarm events depending on *severity*. E.g., if the SNMP trap target configures its severity threshold to *WARNING*, only events of severity level *WARNING* or higher will cause SNMP traps to be sent.

By default, both logging and SNMP trap targets have severity threshold set to level INFO. See section 27.1.3.4 for information on how to classify the severity for alarm triggers.

27.2 Managing Alarms via the Web Interface

27.2.1 Show alarm status

Alarm status is presented in the *System Overview* and the *Detailed System Overview* web pages, which are described in sections 5.4 and 5.4.2.

Fig. 27.5 shows the *System Overview* page when a *Link Alarm* is activated.

Figure 27.5: The basic system overview page with a link alarm activated.

27.2.2 Trigger configuration overview page

Menu path: Configuration ⇒ Alarm ⇒ Triggers

When entering the Alarm configuration page you will be presented to a list of all alarm triggers configured on your unit, see below.

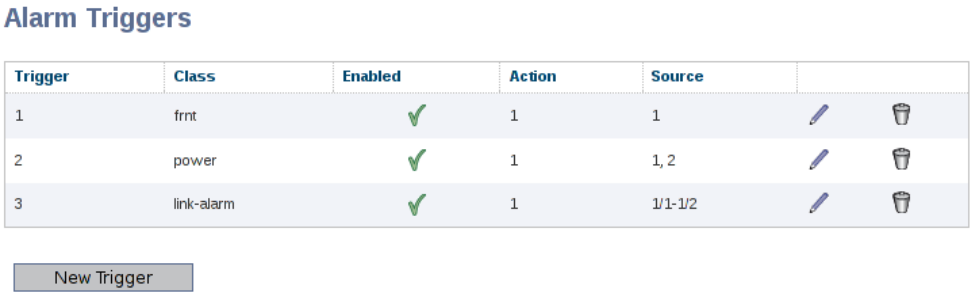


Figure 27.6: The alarm trigger configuration overview page.

Trigger	The index number of this trigger.
Class	The trigger type.
Enabled	A green check-mark means the trigger is enabled, and a dash means it is disabled.
Action	The index of the action profile associated with this trigger. The action profile controls what targets (LED, Digital Out, SNMP traps and/or Logging) to invoke for this alarm trigger.
Source	A list of alarm sources associated with this trigger. For link alarms, this is a list of port numbers, for a power alarm it is the identifiers for the associated power sensors, etc.
Edit	Click this icon to edit a trigger.
Delete	Click this icon to remove a trigger.
New Trigger	Click this button to create a new alarm trigger. You will be presented to a form where you can configure the new trigger.

27.2.3 Create a new alarm trigger using the web interface

Menu path: Configuration ⇒ Alarm ⇒ Triggers ⇒ **`New Trigger`**

When clicking the **`New Trigger`** button you will be presented to list of trigger types. Select the trigger type and click next to continue.

Figure 27.7: The trigger type selection page.

When clicking the **`Next`** button you will be presented to the **`New trigger`** page.

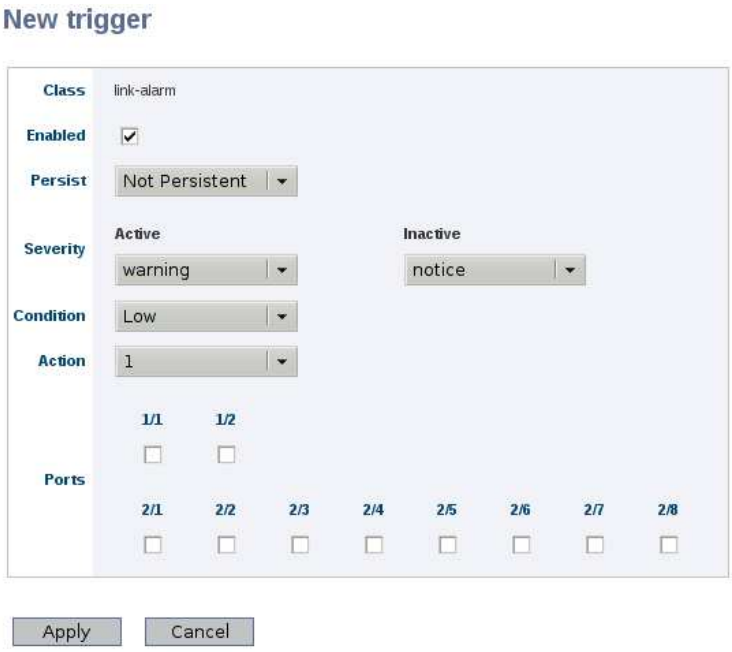


Figure 27.8: The alarm trigger creation page.

Class	The type of alarm trigger.
Enabled	To enable the trigger - check the box, to disable un-check the box.
Persist	
Severity Active	Severity level when active
Severity Inactive	Severity level when inactive
Condition	Controls the condition for triggering (High/low)
Sensors	The sensor source for this trigger
Threshold Rising	Rising Threshold
Threshold Falling	Falling Threshold
Sample Type	
Sample Interval	
Action	Selects the action for the trigger
Port	The ports on your switch is grouped as on the actual hardware, in slots. To get alarms for a a specific port, check the check-box located underneath the port label. In the picture above you see ports 1/1, 1/2 and 2/1 are marked as alarm sources for this link alarm trigger.

27.2.4 Action configuration overview page

Menu path: Configuration ⇒ Alarm ⇒ Actions

When entering the Alarm action configuration page you will be presented to a list of all alarm actions configured on your unit, see below.

Figure 27.9: The alarm action configuration overview page.

Action	The index number of this action.
Targets	The targets for this action.
Edit	Click this icon to edit an action.
Delete	Click this icon to remove an action.
New action	Click this button to add a new alarm action. You will be presented to a form where you can configure the new action.

27.3 CLI

The table below shows alarm management features available via the CLI.

Command	Default	Section
Configure Alarm Configuration Settings		
alarm		Section 27.3.1
[no] trigger <<INDEX> | <CLASS>>		Section 27.3.2
[no] enable	Enabled	Section 27.3.3
[no] <port <PORTLIST> | sensor <SENSORIDLIST> | ring <FRNTINSTANCE>		Section 27.3.4
[no] severity <<LEVEL> | [active <LEVEL>] | [inactive <LEVEL>]>		Section 27.3.5
condition <high|low>		Section 27.3.6
threshold <NUM | [rising <NUM>] | [falling <NUM>]>	rising 0 falling 0	Section 27.3.7
sample [type <abs|delta>] [intv <SECONDS>]		Section 27.3.8
[no] action <INDEX>	1	Section 27.3.9
[no] action <INDEX>		Section 27.3.10
[no] target <[log] [snmp] [led] [digout]>	log	Section 27.3.11
View Alarm Settings and trigger classes		
show alarm		Section 27.3.12
alarm		
show classes		Section 27.3.13
show triggers		Section 27.3.14
show actions		Section 27.3.15
trigger		
show enable		Section 27.3.16
show <port|sensor|ring>		Section 27.3.17
show severity		Section 27.3.18
show condition		Section 27.3.19
show threshold		Section 27.3.20
show sample		Section 27.3.21
action		
show target		Section 27.3.22

Continued on next page

Continued from previous page

Command	Default	Section
Alarm Status		
alarm		Section 27.3.23
show		Section 27.3.24

27.3.1 Managing Alarm Settings

Syntax `alarm`

Context *Global Configuration* context

Usage Enter the *alarm configuration* context.

Default values Not applicable.

Error messages None defined yet.

27.3.2 Manage Alarm Triggers

Syntax `[no] trigger <<INDEX> | <CLASS>>`

Context *Alarm Configuration* context

Usage Create, remove or update an alarm trigger.

- Use **"trigger <CLASS>"** to create a new trigger and enter the Trigger context, e.g., **"trigger link-alarm"** to create a new link-alarm trigger. Use **"show classes"** (section 27.3.13) to list supported trigger classes. An index will be assigned to each created index. This index can be used to update or remove the trigger, see items below.
- Use **"trigger <INDEX>"** to manage an existing trigger.
- Use **"no trigger <INDEX>"** to remove an existing trigger.

Default values Not applicable.

Error messages None defined yet.

Some examples of alarm trigger configurations are given in sections 27.3.2.1-27.3.2.4. Details of individual alarm trigger configuration settings are given in sections 27.3.3-27.3.9.

27.3.2.1 Link Alarm Trigger Configuration Example

Syntax `trigger link-alarm`

Context *Alarm Configuration* context

Usage Create a link-alarm trigger, and enter the configuration context for this trigger.

Additional settings for link-alarm triggers are listed below. The only mandatory setting is the list of ports - no link-alarm alarm events will occur until ports are defined.

- Port(s) (mandatory): Define the port or ports this link-alarm trigger is associated with.
- Enable/Disable: By default, the trigger is enabled.
- Severity: By default, active severity is *WARNING* and inactive severity is *NOTIFY*.
- Action: By default, the trigger is mapped to the default action profile (action 1).

Example

```
redfox:/#> configure
redfox:/config/#> alarm
redfox:/config/alarm/#> trigger link-alarm
Created trigger 2
redfox:/config/alarm/trigger-2/#> port 1/1-1/2
redfox:/config/alarm/trigger-2/#> end
redfox:/config/alarm/#> show
Trigger Class       Enabled  Action  Source
=========================================
      1 power       YES           1  1 2
      2 snr-margin  YES           1  1/1 1/2
redfox:/config/alarm/#>
```

27.3.2.2 Digital-In Trigger Configuration Example

Syntax `trigger digin`

Context *Alarm Configuration* context

Usage Create a digital-in trigger, and enter the configuration context for this trigger.

Additional settings for digital-in triggers are listed below.

- Sensor: By default, digital-in sensor with ID 1 is used. Use **"show env"** (in Admin Exec context) to list available sensors, see section 7.3.27.
- Condition: By default, the alarm condition is set to *low*. That is, *high* is considered normal and *low* is considered an alarm situation. The setting can be changed.
- Enable/Disable: By default, the trigger is enabled.
- Severity: By default, active severity is *WARNING* and inactive severity is *NOTIFY*.

- Action: By default, the trigger is mapped to the default action profile (action 1).

Example

```
redfox:/#> configure
redfox:/config/#> alarm
redfox:/config/alarm/#> trigger digin
Created trigger 2
redfox:/config/alarm/trigger-2/#> end
redfox:/config/alarm/#> show
Trigger Class       Enabled  Action  Source
=========================================
      1 power       YES           1  1 2
      2 digin       YES           1  1
redfox:/config/alarm/#>
```

27.3.2.3 Power Trigger Configuration Example

Syntax `trigger power`

Context *Alarm Configuration* context

Usage Create a power trigger, and enter the configuration context for this trigger.

Additional settings for power triggers are listed below. The only mandatory setting is the list of power sensors - no power alarm events will occur until power sensors are defined.

- Sensor: Westermo units commonly have two power sensors; sensor 1 for DC1 and sensor 2 for DC2. Use **"show env"** (in Admin Exec context) to list available sensors, see section 7.3.27.
- Enable/Disable: By default, the trigger is enabled.
- Severity: By default, active severity is *WARNING* and inactive severity is *NOTIFY*.
- Action: By default, the trigger is mapped to the default action profile (action 1).

Example

Note that a power alarm trigger is generally defined by factory default. The example below assumes there are no existing power alarm triggers.

```
redfox:/#> configure
redfox:/config/#> alarm
redfox:/config/alarm/#> trigger power
Created trigger 1
redfox:/config/alarm/trigger-1/#> sensor 1,2
redfox:/config/alarm/trigger-2/#> end
redfox:/config/alarm/#> show
Trigger Class       Enabled  Action  Source
```

```
=========================================
      1 power       YES           1  1 2
redfox:/config/alarm/#>
```

27.3.2.4 SNR-Margin Trigger Configuration Example

Note, this setting only applies to units equipped with DSL ports.

Syntax `trigger snr-margin`

Context *Alarm Configuration* context

Usage Create a SNR-margin trigger, and enter the configuration context for this trigger.

Additional settings for SNR-margin triggers are listed below. The only mandatory setting is the list of (DSL) ports - no snr-margin alarm events will occur until (DSL) ports are defined.

- Port(s) (mandatory): Define the port or ports this SNR-margin trigger is associated with.
  Note: SNR-margin alarms can only be generated for ports where a connection has been established.
- Alarm threshold: As of WeOS v4.3.0 the SNR-margin falling threshold is set to 3 (dB) by default, and the rising threshold to 6 (dB) by default.
- Enable/Disable: By default, the trigger is enabled.
- Severity: By default, active severity is *WARNING* and inactive severity is *NOTIFY*.
- Action: By default, the trigger is mapped to the default action profile (action 1).

Example

In this example an SNR-margin trigger is created for DSL ports 1/1 and 1/2, with falling threshold 4 dB and rising threshold 6 dB.

```
wolverine:/#> configure
wolverine:/config/#> alarm
wolverine:/config/alarm/#> trigger snr-margin
Created trigger 2
wolverine:/config/alarm/trigger-2/#> port 1/1-1/2
wolverine:/config/alarm/trigger-2/#> threshold falling 4 rising 6
wolverine:/config/alarm/trigger-2/#> end
wolverine:/config/alarm/#> show
Trigger Class       Enabled  Action  Source
=========================================
      1 power       YES           1  1 2
      2 snr-margin  YES           1  1/1 1/2
wolverine:/config/alarm/#>
```

27.3.3 Enable/disable a Trigger

Syntax `[no] enable`

Context *Trigger* context

Usage Enable or disable an alarm trigger. A disabled trigger will keep its configuration settings, but will not affect any alarm targets.

Use **"enable"** to enable and **"no enable"** to disable a trigger.

Default values Enabled

Error messages None defined yet.

27.3.4 Manage alarm sources

Syntax `[no] <port <PORTLIST> | sensor <SENSORIDLIST> | ring <FRNTINSTANCE>>`

Context *Trigger* context

Usage Specify which alarm sources the trigger should monitor. The command syntax differs depending on the trigger class:

- Use **"[no] port <PORTLIST>"** to specify which port(s) a *link-alarm* trigger should apply to, e.g., use **"port 1/1,2/2-2/4"** to add ports 1/1, and 2/2-2/4 to the list of ports monitored by this link-alarm trigger.
- Use **"[no] ring <FRNTINSTANCE>"** which FRNT ring an FRNT alarm trigger should apply to.
- Use **"[no] sensor <SENSORIDLIST>"** to specify which sensors a *digital in*, *power* or *temperature* trigger should apply to, e.g., use **"sensor 1,2"** to add power sensors 1 and 2 to the list of power sensors monitored by this power trigger.
  Use command `show env` (section 7.3.27) to list available sensors and their index values.

Use **"no port <PORTLIST>"** remove a specific set of ports, or **"no port"** to remove all ports from a trigger (the same goes for other source types).

If no sources are defined when exiting the trigger context, the trigger will automatically be configured as *disabled* (see section 27.3.3).

Default values

Error messages None defined yet.

27.3.5 Alarm Event Severity

Syntax `[no] severity <<LEVEL>|[active <LEVEL>]|[inactive <LEVEL>]>`

Context *Trigger* context

Usage Specify the severity level of *active* and *inactive* alarm events detected by this trigger. See section 27.1.3.4 for information on available severity levels.

Active and inactive severity levels can be configured together or independently.

"`no severity`" to will set severity to level *NONE*. Alarm events with severity NONE will not cause SNMP traps to be sent or events to be logged, however, such events can still affect digital-out and ON LED targets.

Default values `active warning` and `inactive notice`

Error messages None defined yet.

Examples The examples below show how to set severity level for active and inactive alarm events together and how to set it individually. The final example shows how to set severity 'NONE' for both active and inactive events.

```
redfox:/config/alarm/trigger-2/#>  severity err
redfox:/config/alarm/trigger-2/#>  show severity
active err, inactive err
redfox:/config/alarm/trigger-2/#>  severity inactive debug
redfox:/config/alarm/trigger-2/#>  show severity
active err, inactive debug
redfox:/config/alarm/trigger-2/#>
redfox:/config/alarm/trigger-2/#>  no severity
redfox:/config/alarm/trigger-2/#>  show severity
active none, inactive none
redfox:/config/alarm/trigger-2/#>
```

27.3.6 Configure Alarm Condition Setting

Syntax `condition <high|low>`

Alternate keywords are possible:

- *rising* and *up* are equivalents to *high*.
- *falling* and *down* are equivalents to *low*.

Context *Trigger* context

Usage Define whether the *high* or *low* trigger state should be considered the *alarm state*, while the other is considered the *normal state*.

Some triggers, such as *link-alarm* and *power* triggers have a static (predefined) alarm condition setting. (Both link-alarm and power triggers have *condition* set to *low*). For other triggers, the alarm condition setting is configurable.

See section 27.1.3.2 for more information.

Default values Differs for different trigger types

Error messages None defined yet.

27.3.7 Configure Rising and Falling Thresholds

Syntax `threshold <NUM|[rising <NUM>]|[falling <NUM>]>`

Context *Trigger* context

Usage Set falling and rising thresholds. The thresholds may be set to the same value, but by using different thresholds (rising higher than falling) one can avoid receiving multiple events when the alarm source fluctuates around the alarm threshold.

Triggers which are *binary* to their nature, such as *link-alarm*, *power*, and digital-in triggers have implicit thresholds, which cannot be configured.

See section 27.1.3.2 for more information.

Default values `rising 0` and `falling 0` (except for *binary* alarm sources)

Error messages None defined yet.

27.3.8 Configure Sampling Type and Interval

As of WeOS v4.3.0 only the ”absolute” sample type is supported, and the sample interval setting has no affect.

Syntax `sample [type <abs|delta>] [intv <SECONDS>]`

Context *Trigger* context

Usage Define sample type (absolute or delta) and the sampling interval (seconds). With absolute sampling, the sampled value is compared directly to the thresholds. For delta sampling it is the difference between the current and the previous sample values which is compared to the thresholds.

See section 27.1.3.3 for more information.

Default values

Error messages None defined yet.

27.3.9 Configure Trigger Action

Syntax `[no] action <INDEX>`

Context *Trigger* context

Usage Specify the action (profile) to be invoked when this trigger detects an alarm event.

Default values 1 (default action)

Error messages None defined yet.

27.3.10 Manage Alarm Actions

Syntax `[no] action <INDEX>`

Context *Alarm Configuration* context

Usage Create, remove or update an alarm action (profile). Use **"action <INDEX>"** to enter the Action context and create a new or update an existing action.

Use **"no action <INDEX>"** remove an existing action. The default action (index 1) cannot be removed.

Default values Not applicable.

Error messages None defined yet.

27.3.11 Manage Action Targets

Syntax `[no] target <[log] [snmp] [led] [digout]>`

Context *Action* context

Usage Add or remove alarm target to an alarm action (profile).

Default values `target log` (New action profiles has **"target log"** as default.

Error messages None defined yet.

27.3.12 Show Alarm Configuration Overview

Syntax `show alarm`

Context *Global Configuration* context. Also available as **"show"** command within the *Alarm Configuration* context.

Usage List an overview of configured alarm triggers and actions.

Default values Not applicable

Error messages None defined yet.

27.3.13 Show Supported Trigger Classes

Syntax `show classes`

Context *Alarm Configuration* context

Usage List supported trigger classes. These are the classes to be used with the **"trigger <CLASS< command"** (see section 27.3.2).

Default values Not applicable

Error messages None defined yet.

27.3.14 Show Configured Triggers

Syntax `show triggers`

Context *Alarm Configuration* context

Usage List configured alarm triggers. This is useful to find the index of a trigger, which is needed to edit (**"trigger <INDEX>"**) or remove (no **"trigger <INDEX>"**) an existing trigger, see section 27.3.2.

Default values Not applicable

Error messages None defined yet.

27.3.15 Show Configured Action Profiles

Syntax `show actions`

Context *Alarm Configuration* context

Usage List configured alarm action profiles.

Default values Not applicable

Error messages None defined yet.

27.3.16 Show Triggers Enable Setting

Syntax `show enable`

Context *Trigger* context

Usage Show whether this trigger is *enabled* or *disabled*.

Default values Not applicable

Error messages None defined yet.

27.3.17 Show Trigger Alarm Sources

Syntax `show <port|sensor|ring>`

Context *Trigger* context

Usage Show the alarm sources associated with this trigger. The type of alarm source differs depending on trigger class. See section 27.3.4 for further information.

Default values Not applicable

Error messages None defined yet.

27.3.18 Show Trigger Severity Setting

Syntax `show severity`

Context *Trigger* context

Usage Show the severity setting (*active* and *inactive* severity) for this trigger.

Default values Not applicable

Error messages None defined yet.

27.3.19 Show Trigger Condition Setting

Syntax `show condition`

Context *Trigger* context

Usage Show the alarm condition setting for this trigger.

Default values Not applicable

Error messages None defined yet.

27.3.20 Show Trigger Threshold Settings

Syntax `show threshold`

Context *Trigger* context

Usage Show the trigger threshold setting (both *rising* and *falling* thresholds) for this trigger.

Default values Not applicable

Error messages None defined yet.

27.3.21 Show Trigger Sample Type and Interval

As of WeOS v4.3.0 only the ”absolute” sample type is supported, and the sample interval setting has no affect.

Syntax `show sample`

Context *Trigger* context

Usage Show the alarm condition setting for this trigger.

Default values Not applicable

Error messages None defined yet.

27.3.22 Show Action Targets

Syntax `show target`

Context *Action* context

Usage Show the alarm target(s) configured for this action profile.

Default values Not applicable

Error messages None defined yet.

27.3.23 Handling Alarm Status

Syntax `alarm`

Context *Admin Exec* context

Usage Enter the *alarm status* context.

Default values Not applicable.

Error messages None defined yet.

27.3.24 Show overall alarm status

Syntax `show`

Context *Alarm Status* context

Usage Show status of all alarms.

Default values Not applicable.

Error messages None defined yet.

27.4 Digital I/O

Almost all Westermo products running WeOS are with a *Digital I/O* connector as the one shown in fig. 27.10. The location of the connector differs between products; on RedFox Industrial it is located on the CPU card as shown in fig. 27.11). For a detailed specification on the Digital I/O connector (including definite pin-out mapping, voltage levels, etc.), please see the *User Guide* of your specific Westermo product[8, 9, 10, 11].

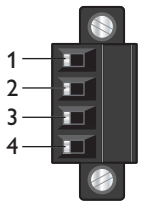
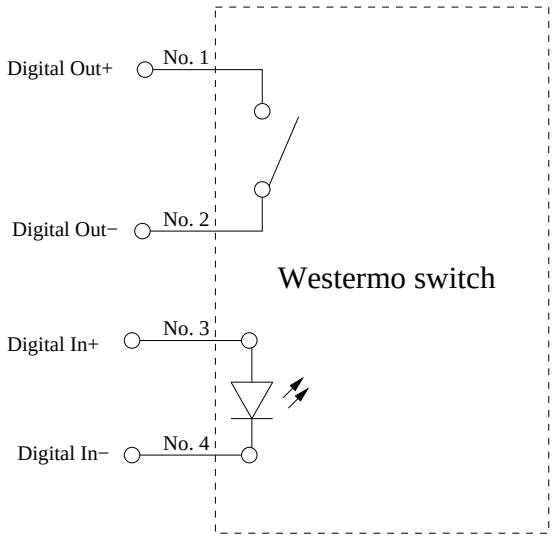


Figure 27.10: Digital I/O connector.

The Pin-Out of the Digital I/O connector is as follows:

Position	Description
1	Digital-Out + (Relay Output +)
2	Digital-Out - (Relay Output -)
3	Digital-In +
4	Digital-In -



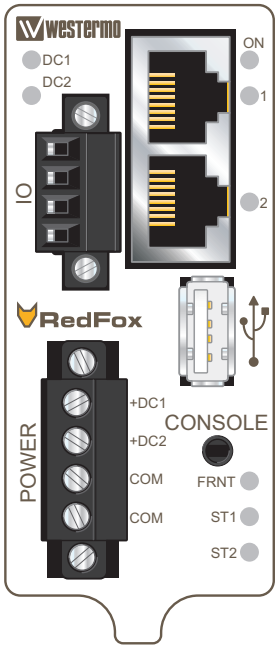


Figure 27.11: The RedFox Industrial Power and CPU module

As described in section 27.1, *Digital-In* can be used as an alarm source, while *Digital-Out* is utilised as an alarm target (*summary alarm*).

- The Digital-In alarm is triggered when there is *lack of* voltage on the Digital-In pins. For information on appropriate voltage/current levels to trigger alarms via Digital-In, see the *User Guide* of your specific product[8, 9, 10, 11].
- The Digital-Out pins are internally connected to a *gate*. The gate is *open* when the switch has no power, or when any *alarm sources* are active. When the switch is operating normally (the switch has booted up, and no alarm source is active), the gate is *closed*.

27.5 LEDs

The LED functionality when running WeOS is described in the *User Guide* of your product[8, 9, 10, 11]. Here the information on LED functionality of *all* WeOS products is summarised. Note that your product may not have all LED types listed here.

LED	Status	Description
ON	OFF	Unit has no power
	GREEN	All OK, no alarm condition.
	RED	Alarm condition, or until unit has started up. (Alarm conditions are configurable, see sections 27.1-27.3.)
	GREEN BLINK	Location indicator (”Here I am!”). Activated when connected to IPConfig Tool, or upon request from Web, or when entering the CLI configuration context. Duration of blinking: 10 seconds.
	RED BLINK	Location indicator (see previous item) **or** indication of pending cable factory reset, see section 7.1.4.3.
DC1	OFF	Unit has no power.
	GREEN	Power OK on DC1.
	RED	Power failure on DC1.
DC2	OFF	Unit has no power.
	GREEN	Power OK on DC2.
	RED	Power failure on DC2.
FRNT	OFF	FRNT disabled
	GREEN	FRNT OK. (See also the *FRNT Error* item below.)
	RED	FRNT Error. A focal point can detect and indicate local FRNT errors (FRNT link down) as well as FRNT errors elsewhere in the FRNT ring. A member switch only detects and indicates local FRNT errors (FRNT link down).
	BLINK	Unit configured as focal point.
RSTP (formerly ST1)	OFF	RSTP disabled.
	GREEN	RSTP enabled.
	BLINK	Unit elected as RSTP/STP root switch.
VPN (formerly ST2)	OFF	VPN disabled.
	GREEN	(Configurable) Default: At least one VPN tunnel up and OK.
	RED	(Configurable) Default: At VPN tunnels down.

Continued on next page.

Continued from previous page.

LED	Status	Description
Ethernet ports	OFF	No link.
	GREEN	Link established.
	GREEN FLASH	Data traffic indication.
	YELLOW	Port alarm and no link. Or if FRNT or RSTP mode, port is blocked.
SHDSL ports	OFF	No SHDSL link.
	GREEN	SHDSL link established.
	GREEN BLINK	SHDSL link negotiation.
	GREEN FLASH	Data traffic indication.
	YELLOW	Port alarm and no link. Or if FRNT or RSTP mode, port is blocked.
	YELLOW BLINK	Only during unit startup. Firmware downloading to SHDSL chip.
TD	OFF	No serial data received.
	GREEN FLASH	Serial data received.
RD	OFF	No serial data transmitted.
	GREEN FLASH	Serial data transmitted.

Additional explanations:

- BLINK means that the LED is blinking with a frequency about 1 Hz.
- FLASH means that the LED is blinking with a higher frequency.
- SHDSL LEDs only apply to products with SHDSL ports.
- TD and RD LEDs only apply to products with serial port(s). As the WeOS serial ports operate in DCE mode, TD denotes receiving, and RD denotes transmitting serial data.

Chapter 28

Logging Support

This chapter describes WeOS support for alarm and generic event logging.

In WeOS general events detected by the system (such as user login attempts), as well as alarm events defined by configured alarm triggers (see chapter 27) can be logged for further analysis. Three logging methods are available:

- *Logging to file:* General events and alarm events are always logged to a local log file.
- *Logging to console:* It is possible to direct logging messages to the console port. Messages of severity level *DEBUG* or higher are shown on the console port.
- *Logging to a remote syslog server:* Logging messages can be sent to a remote syslog server for further processing. Messages of severity level *NOTICE* or higher are forwarded to the remote syslog server(s).

As of WeOS v4.3.0 logging support is only available via the CLI. The severity thresholds for console and remote syslog logging are not configurable, however, such support is planned.

28.1 Managing Logging Support via the CLI

Command	Default	Section
Configuring Logging Settings		
[no] logging	Disabled	Section 28.1.1
[no] console		Section 28.1.2
[no] server <ADDRESS1 [ADDRESS2]>	Disabled	Section 28.1.3
View Logging Settings		
show logging		Section 28.1.4
logging		
show console		Section 28.1.5
show server		Section 28.1.6
Managing Log Files		
dir <cfg:///|log:///|usb://>		Section 7.3.3
copy <FROM_FILE> <TO_FILE>		Section 7.3.4
erase <file>		Section 7.3.5
show <running-config | startup-config | factory-config | [<filesys>://]FILENAME>		Section 7.3.6

28.1.1 Managing Logging Settings

Syntax `[no] logging`

Context *Global Configuration* context

Usage Enter Logging configuration context.
Use **"`no logging`"** to disable all logging.

Default values Disabled

Error messages None defined yet.

28.1.2 Logging to console port

Syntax `[no] console`

Context *Logging* context

Usage Enable or disable console logging.
Use **"`no console`"** to disable console logging.
When enabled, general events detected by the system, as well as alarm events associated with configured alarm triggers, will be presented on the console port.

Default values *Disabled*

Error messages None defined yet.

28.1.3 Logging to remote syslog server

Syntax `[no] server <ADDRESS1 [ADDRESS2]>`

Context *Logging* context

Usage Set remote syslog server(s) (IPv4 addresses). A maximum of two remote syslog servers are supported. The syntax allows typing them in one line or two separate lines.

Use **"no server <ADDRESS>"** to remove a single server. Use **"no server"** to remove all servers.

When enabled, general events detected by the system, as well as alarm events associated with configured alarm triggers, will be forwarded to the configured syslog server via UDP to port 514. If two servers are configured, messages are sent to both of them.

Default values *Disabled*

Error messages None defined yet.

28.1.4 Show Logging Settings

Syntax `show logging`

Context *Global Configuration* context. Also available as **"show"** command within the Logging context.

Usage Show Logging configuration settings.

Default values Not applicable

Error messages None defined yet.

28.1.5 Show Console Logging Setting

Syntax `show console`

Context *Logging* context.

Usage Show whether console port logging is enabled or disabled.

Default values Not applicable

Error messages None defined yet.

28.1.6 Show Remote Syslog Server Setting

Syntax `show server`

Context *Logging* context.

Usage Show whether remote syslog logging is enabled or disabled. If enabled, the IP address(es) of the configured server(s) are presented.

Default values Not applicable

Error messages None defined yet.

Chapter 29

SNMP

The Simple Network Management Protocol (SNMP) provides a standardised method to manage and monitor IP devices remotely. The WeOS SNMP agent supports SNMP v1, v2c and v3.

29.1 SNMP introduction and feature overview

Table 29.1 shows WeOS SNMP control features for the Web and CLI interfaces. Further description of the SNMP support is presented in the sections 29.1.1-29.1.6. If you are only interested in knowing how to manage SNMP features via the Web or CLI, please visit sections 29.2 or 29.3 directly.

29.1.1 SNMP introduction

The Simple Network Management Protocol (SNMP) provides a standardised method to manage and monitor IP devices remotely. In SNMP a *manager station* can manage a set of status and configuration objects via an SNMP *agent* on the management unit. The WeOS SNMP agent supports SNMP v1, v2c and v3.

An SNMP manager:

- can send SNMP *GET* messages to poll status and configuration information from an SNMP *agent*.
- can send SNMP *SET* messages to the SNMP *agent* to modify the device settings (or issue commands such as 'reboot').
- can get notified by an agent when specific events occur, such as link down event, via SNMP *TRAP* messages.

Feature	Web (Sec. 29.2)	CLI (Sec. 29.3)	General Description
General			
Enable/disable SNMP	X	X	
SNMPv1/v2c			
Read Community	X	X	Sec. 29.1.2
Write Community	X	X	”
Trap Community	X	X	Sec. 29.1.2-29.1.3
Trap Host	X	X	Sec. 29.1.3
SNMPv3			
Read-Only SNMPv3 User		X	Sec. 29.1.4
Read/Write SNMPv3 User		X	”

Table 29.1: WeOS control of SNMP features.

The objects manageable via SNMP are defined in a management information base (MIB). The WeOS MIB support aims at providing SNMP management primarily via standard MIBs to enable easy integration with existing SNMP management tools. In addition, WeOS includes an enterprise MIB (private MIB) to provide access to MIB objects not available via the standard MIBs.

29.1.2 SNMP Communities

An SNMP *community* is a relationship between the manager and managed station. It can be seen as a (very) basic authentication and authorisation mechanism for SNMP v1 and v2c[1]. Three types of communities are supported:

- *Read community*: The SNMP read community is used by a manager to read SNMP MIB objects from a managed station.
  Default read community: `public`
- *Write community*: The SNMP write community can be used to write (and read) SNMP MIB objects to (from) a managed station. Thus, if the agent has its write community enabled, it is possible to configure the switch via SNMP. The write community is typically named **”private”**.
  Default write community: `Disabled`

[1] See section 29.1.4 for secure management using SNMPv3.

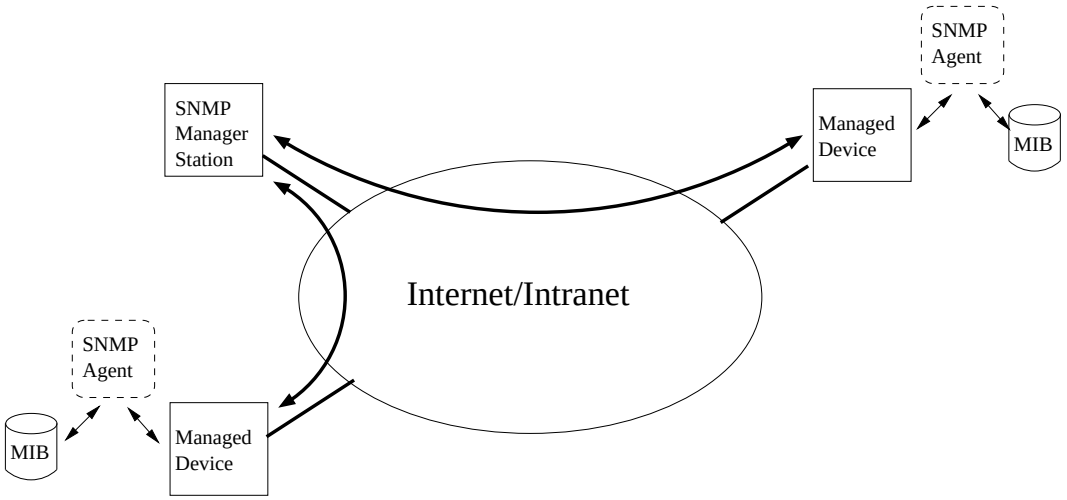


Figure 29.1: Sample SNMP setup, where one manager station controls two devices by communicating with SNMP agents running on the managed devices.

- *Trap community*: The SNMP trap community is used when an agent wants to send a notification to the manager (SNMP Trap). The trap community is typically named **”public”**.
  Default trap community: `Disabled`

Warning: *Using the well-known community strings ”public” and ”private” could pose a serious security problem.*

29.1.3 Trap Support

The WeOS SNMP trap support is integrated with the WeOS alarm handling system (see 27.1). This means that you as an operator has fine grain control of which traps to send.

The list below all traps, except *Coldstart*, are integrated with and can be controlled via the alarm handling system.

- *Link Alarm*: A trap is generated on *link up* or *link down*, given that *Link Alarm* is enabled on that specific port (see sections 27.1.3 and9.1.4).

  Link Down OID: iso(1).org(3).dod(6).internet(1).snmpV2(6).snmpModules(3).snmpMIB(1).snmpMIBObjects(1).snmpTraps(5).linkDown(3)

Link Up OID: iso(1).org(3).dod(6).internet(1).snmpV2(6).snmpModules(3).snmpMIB(1).snmpMIBObjects(1).snmpTraps(5).linkUp(4)

> **Note**: *When a port is being reconfigured, link down and link up events are likely to occur. If* link-alarm *is enabled on that port, a couple of SNMP traps are likely to be generated as a side-effect of the port reconfiguration.*

- **Cold Start**: A trap is generated when a system comes up.
  OID: *iso(1).org(3).dod(6).internet(1).snmpV2(6).snmpModules(3).snmpMIB(1).snmpMIBObjects(1).snmpTraps(5).coldStart(1)*
- **Digital-In**: A trap is generated when the voltage level on the pins of a digital-in sensor changes from *high* to *low*, or *low* to *high*.
  Digital-In High OID: *iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).westermo(16177).common(2).weos(1).notifications(6).sensorNotifications(1).sensorNotificationPrefix(0).digitalInHigh(1)*
  Digital-In Low OID: *iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).westermo(16177).common(2).weos(1).notifications(6).sensorNotifications(1).sensorNotificationPrefix(0).digitalInLow(2)*
- **Power Supply**: A trap is generated when the voltage level on any of the power feeds changes from *high* to *low*, or *low* to *high*.
  Power Supply High OID: *iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).westermo(16177).common(2).weos(1).notifications(6).sensorNotifications(1).sensorNotificationPrefix(0).powerSupplyHigh(3)*
  Power Supply Low OID: *iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).westermo(16177).common(2).weos(1).notifications(6).sensorNotifications(1).sensorNotificationPrefix(0).powerSupplyLow(4)*
- **FRNT Ring Status**: A trap is generated when a unit detects a change of FRNT ring status, i.e., ring up (ring mode) or ring down (bus mode).
  FRNT Ring Up OID: *iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).westermo(16177).common(2).weos(1).notifications(6).frntNotifications(2).frntNotificationPrefix(0).frntRingUp(1)*
  FRNT Ring Down OID: *iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).westermo(16177).common(2).weos(1).notifications(6).frntNotifications(2).frntNotificationPrefix(0).frntRingDown(2)*
- **SNR-margin**: On units with a DSL port (Wolverine-225) traps are generated when the SNR margin falls below (or rises above) a configurable threshold.

OID: *iso(1).org(3).dod(6).internet(1).mgmt(2).mib-2(1).transmission(10).hdsl2ShdslMIB(48).hdsl2ShdslNotifications(0).hdsl2ShdslSNRMarginCrossing(2)*

SNMP traps are only generated if the SNMP *trap* community is configured (see section 29.1.2), and if there is at least one *Trap Host* (i.e., SNMP management station) defined. Up to two *Trap Hosts* can be defined (if two Trap Hosts are configured, traps will be sent to both of them).

29.1.4 Secure management using SNMPv3

To manage a unit securely via SNMP, SNMPv3 should be used. SNMPv3 provides privacy and integrity (per packet authentication) to the SNMP messages.

SNMPv3 introduces the notion of a SNMPv3 *user*, as opposed to the *community* concept used in SNMPv1/v2c. The following parameters can be configured for an SNMPv3 user.

- Read-Only or Read-Write access: Defines whether the *user* should have *read* access to the SNMP variables, or be able to *read* and *modify* them.
- Security Mode: Three security modes are available:
  - *noAuthnoPriv*: No security (i.e., neither authentication, nor encryption)
  - *authNoPriv*: Authentication, but no privacy.
  - *authPriv*: Authentication and Encryption

  Note: *As of WeOS v4.3.0, the WeOS SNMP agent accepts SNMP requests of security level* authNoPriv *also for SNMPv3 users created at level* authPriv*. This feature is likely to be removed in future WeOS releases.*
- Encryption protocol: WeOS offers SNMPv3 data encryption using DES and AES-128.
- Authentication protocol: WeOS offers SNMPv3 data integrity using using MD5 and SHA1.
- Scope: A user can be restrained to only access a part of the MIB tree supported by the unit.

The encryption and authentication passwords are strings of 8-16 characters. ASCII characters 33-126 except '#' (ASCII 35) are allowed.

A maximum of 8 SNMPv3 users can be defined, each with their own parameter set.

29.1.4.1 SNMPv3 example

This example illustrates the configuration of an SNMPv3 user on the a WeOS switch. The user *alice* is grated *read-only* access to the full MIB tree. Security level *authNoPriv* is used where SHA1 is used as authentication protocol

```
redfox:/#> configure
redfox:/config/#> snmp-server
redfox:/config/snmp/#> rouser alice auth sha1 alicepwd
redfox:/config/snmp/#> leave
redfox:/#> cp running start
```

Section 29.1.6 lists recommended SNMP managment software. Those tools have graphical user interfaces and should be straight forward to use. For a simple test you could also use the (Unix) Net-SNMP **"snmpwalk"** command. (Here it is assumed that the switch is accessible on IP address *192.168.2.200* and the "walk" is limited to the mib-2 system's group).

```
mypc:~$ snmpwalk -v3 -u alice -l authNoPriv -a SHA -A alicepwd 192.168.2.200 system
SNMPv2-MIB::sysDescr.0 = STRING: Westermo RedFox Industrial,
pri. version: 9.99, back. version: 4.0.0, boot version: 2.01, fpga: 20080626
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.16177
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (1026874) 2:51:08.74
SNMPv2-MIB::sysContact.0 = STRING: support@westermo.se
SNMPv2-MIB::sysName.0 = STRING: redfox
SNMPv2-MIB::sysLocation.0 = STRING: westermo
SNMPv2-MIB::sysServices.0 = INTEGER: 79
...
mypc:~$
```

29.1.5 Supported MIBs

29.1.5.1 Standard MIBs

As of WeOS v4.3.0 the following standard MIBs are supported:

- RFC1213 MIB-2: The original MIB-2 standard MIB.
- RFC2863 Interface MIB: The *ifXTable* of the IF-MIB is supported.
- RFC2819 RMON MIB: RMON Ethernet statistics (*etherStatsTable*) is supported.
- RFC4188 Bridge MIB
- RFC4318 RSTP MIB
- RFC4363 Q-BRIDGE MIB: The *dot1qVlan* group and *dot1qVlanStaticTable* are supported, enabling support for static VLAN configuration.
- RFC4836 MAU MIB: The *dot3IfMauBasicGroup* and *dot3IfMauAutoNegGroup* of the MAU MIB are supported.
- RFC4133 Entity MIB: The *entityPhysical* group of the Entity MIB is supported. It can be used to read unit serial number, firmware version, etc.
- RFC3433 Entity Sensor MIB: The Entity Sensor MIB can be used to monitor the status of unit sensors for temperature, power supply, and ”digital-in”, etc.
- RFC 4319 HDSL2/SHDSL MIB: On products with SHDSL ports, the *hdsl2ShdslSpanConfTable*, *hdsl2ShdslSpanStatusTable*, *hdsl2ShdslInventoryTable* and *hdsl2ShdslSpanConfProfileTable* are supported (read-only).

29.1.5.2 Private MIB

To use the WeOS private MIB, two Westermo specific MIB files should be loaded into your SNMP management software (see section 29.1.6 for information on recommended management software):

- WESTERMO-OID-MIB: Defines the top level objects of the Westermo Private MIB name space.
- WESTERMO-WEOS-MIB: Defines the WeOS branch of the Westermo Private MIB.

29.1.6 Recommended Management Software

Westermo recommends the following SNMP managers:

- OidView from ByteSphere[2].
- MG-SOFT MIB Browser Pro. from MG-SOFT[3].
- SNMPc from Castlerock Computing[4].

[2] http://www.oidview.com/oidview.html. OidView is a trademark of BYTESPHERE TECHNOLOGIES LLC.
[3] http://www.mg-soft.com/mgMibBrowserPE.html.
[4] http://www.castlerock.com/. SNMPc is a trademark of Castlerock Computing.

29.2 Managing SNMP via the web interface

Menu path: Configuration ⇒ SNMP

On the SNMP configuration page you will be presented to the current settings for SNMP on your switch, see below. You may change the settings by editing the page.

Enabled	Check the box to enable SNMP. If you have a JavaScript[1] enabled browser the other settings will not be displayed unless you check this box.
Read Community	A community identifier for read access.
Write Community	A community identifier for read/write access.
Trap Community	A community identifier for traps.
Trap Host Address 1/2	IP address of SNMP trap management station. None, one or two addresses may be filled in.

[1] JavaScript is a trademark of Sun Microsystems.

29.3 Manage SNMP Settings via the CLI

Command	Default	Section
[no] snmp-server	Enabled	Section 29.3.1
[no] rocommunity <COMMUNITY>	public	Section 29.3.2
[no] rwcommunity <COMMUNITY>	Disabled	Section 29.3.3
[no] trapcommunity <COMMUNITY>	Disabled	Section 29.3.4
[no] host <IPADDR>	Disabled	Section 29.3.5
[no] rouser <USERNAME>	Disabled	Section 29.3.6
[auth <md5|sha1> <PASSPHRASE>		
[crypto <des|aes128> <PASSPHRASE>]]		
[OIDTREE]		
[no] rwuser <USERNAME>	Disabled	Section 29.3.7
[auth <md5|sha1> <PASSPHRASE>		
[crypto <des|aes128> <PASSPHRASE>]]		
[OIDTREE]		
Show SNMP settings		
show snmp-server		Section 29.3.8
snmp-server		
show rocommunity		Section 29.3.9
show rwcommunity		Section 29.3.10
show trapcommunity		Section 29.3.11
show host		Section 29.3.12
show rouser		Section 29.3.13
show rwuser		Section 29.3.14

29.3.1 Manage SNMP Server

Syntax `[no] snmp-server`

Context *Global Configuration* context.

Usage Enter *snmp-server* context. If the SNMP server is disabled, it will be enabled when issuing the **”snmp-server”** command. Use **”no snmp-server”** to disable the SNMP server.

Default values Enabled.

Error messages None defined yet.

29.3.2 Manage SNMP Read Community

Syntax `[no] rocommunity <COMMUNITY_STRING>`

Context *snmp-server* context.

Usage Configure the SNMP Read Community string. Use **"no rocommunity"** to disable the SNMP Read Community.

Default values `rocommunity public`

Error messages None defined yet.

29.3.3 Manage SNMP Write Community

Syntax `[no] rwcommunity <COMMUNITY_STRING>`

Context *snmp-server* context.

Usage Configure the SNMP Write Community string. Use **"no rwcommunity"** to disable the SNMP Read Community.

Default values Disabled.

Error messages None defined yet.

29.3.4 Manage SNMP Trap Community

Syntax `[no] trapcommunity <COMMUNITY_STRING>`

Context *snmp-server* context.

Usage Configure the SNMP Trap Community string. Use **"no trapcommunity"** to disable the SNMP Trap Community.

Default values Disabled.

Error messages None defined yet.

29.3.5 Manage SNMP Trap Hosts

Syntax `[no] host <IPV4ADDRESS>`

Context *snmp-server* context.

Usage Configure a SNMP Trap Host. Two trap hosts can be configured (issue the **"trap-host"** command twice with different IP addresses). Use **"no host <IPV4ADDRESS>"** to remove a trap-host and **"no host"** to remove all trap hosts.

Default values Disabled.

Error messages None defined yet.

29.3.6 Manage SNMPv3 Read-Only User

Syntax `[no] rouser <USERNAME> [auth <md5|sha1> <PASSPHRASE> [crypto <des|aes128> <PASSPHRASE>]] [OIDTREE]`

Context *snmp-server* context.

Usage Configure a SNMP read-only user.

- *USERNAME:* is a text string defining the user. Max 32 characters. Valid characters are ASCII 33-126 except '#' (ASCII 35).
- *Authentication:* Achieve message integrity protection by specifying MD5 or SHA1 message authentication. The authentication password is a string of 8-16 characters. ASCII characters 33-126 except '#' (ASCII 35) are allowed.
- *Encryptions:* Achieve message privacy by specifying DES or AES128 message authentication. The authentication password is a string of 8-16 characters. ASCII characters 33-126 except '#' (ASCII 35) are allowed.
- *OIDTREE:* Limit access to a certain branch of the supported MIB. Defaults to the whole tree ('1.')

Use **"no rouser <USERNAME>"** to remove a specific *read-only* user, or **"no rouser"** to remove all read-only users.

Default values Disabled.

Error messages None defined yet.

Examples

- Authentication and encryption:
  "rouser alice auth sha1 alicepwd1 crypto aes128 alicepwd2"
- Authentication with access to dot1dBridge subtree:
  "rouser bob auth md5 bobspwd1 1.3.6.1.2.1.17"

29.3.7 Manage SNMPv3 Read-Write User

Syntax `[no] rwuser <USERNAME> [auth <md5|sha1> <PASSPHRASE> [crypto <des|aes128> <PASSPHRASE>]] [OIDTREE]`

Context *snmp-server* context.

Usage Configure a SNMP read-write user.

For more information, see section 29.3.7.

Default values Disabled.

Error messages None defined yet.

Examples See section 29.3.7.

29.3.8 View SNMP Server Settings

Syntax `show snmp-server`

Context *Global Configuration* context. Also available as **"show"** command within the *snmp-server* context.

Usage Show all SNMP server settings.

Default values Not applicable.

Error messages None defined yet.

29.3.9 View SNMP Read Community Settings

Syntax `show rocommunity`

Context *snmp-server* context.

Usage Show the SNMP Read Community setting.

Default values Not applicable.

Error messages None defined yet.

29.3.10 View SNMP Write Community Settings

Syntax `show rwcommunity`

Context *snmp-server* context.

Usage Show the SNMP Write Community setting.

Default values Not applicable.

Error messages None defined yet.

29.3.11 View SNMP Trap Community Settings

Syntax `show trapcommunity`

Context *snmp-server* context.

Usage Show the SNMP Trap Community setting.

Default values Not applicable.

Error messages None defined yet.

29.3.12 View SNMP Trap Host Settings

Syntax `show host`
Context *snmp-server* context.
Usage Show the SNMP Trap Host setting.
Default values Not applicable.
Error messages None defined yet.

29.3.13 View SNMPv3 Read-Only User Settings

Syntax `show rouser`
Context *snmp-server* context.
Usage Show settings for configured SNMPv3 read-only users.
Default values Not applicable.
Error messages None defined yet.

29.3.14 View SNMPv3 Read-Write User Settings

Syntax `show rwuser`
Context *snmp-server* context.
Usage Show settings for configured SNMPv3 read-write users.
Default values Not applicable.
Error messages None defined yet.

Acronyms and abbreviations

3DES	Triple DES
AH	Authentication Header
ASCII	American Standard Code for Information Interchange
AES	Advanced Encryption Standard
CLI	Command Line Interface
CPU	Central Processing Unit
DES	Data Encryption Standard
DDNS	Dynamic DNS
DH	Diffie-Hellman
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DPD	Dead Peer Detection
DSL	Digital Subscriber Line
ESP	Encapsulating Security Payload
FRNT	Fast Reconfiguration of Network Topology
HTTP	Hypertext Transfer Protocol
HTTPS	Secure HTTP (HTTP over SSL/TLS)
I/O	Input/Output
IGMP	Internet Group Management Protocol
IKE	Internet Key Exchange
IKEv1	IKE version 1
IP	Internet Protocol
IPSec	IP Security
IPv4	IP version 4
IPv6	IP version 6
LAN	Local Area Network
LED	Light Emitting Diode
MD5	Message Digest 5
MIB	Management Information Base

MTU	Maximum Transfer Unit
NAPT	Network Address and Port Translation
NAT	Network Address Translation
NAT-T	NAT Traversal
NTP	Network Time Protcol
OID	Object Identifier
PC	Personal Computer
PFS	Perfect Forward Secrecy
PPP	Point to Point Protocol
RAM	Random Access Memory
RMON	Remote Monitoring
SHDSL	Symmetric High-speed Digital Subscriber Line
SFP	Small Form-factor Pluggable (transceiver module)
SHA	Secure Hash Algorithm
SHA-1	Secure Hash Algorithm 1
SNMP	Simple Network Management Protocol
SNR	Signal to Noise Ratio
SNTP	Simple NTP
SSH	Secure SHell
SSL	Secure Socket Layer
TLS	Transport Layer Security
USB	Universal Serial Bus
VLAN	Virtual LAN
VPN	Virtual Private Network
WAN	Wide Area Network
WeOS	Westermo Operating System

Bibliography

[1] M. Christensen, K. Kimball, and F. Solensky. Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches. rfc 4541, IETF, May 2006.

[2] G. Clark. Telnet Com Port Control Option. rfc 2217, IETF, October 1997.

[3] C.L. Hedrick. Routing Information Protocol. rfc 1058, IETF, June 1988.

[4] R. Hinden and Ed. Virtual Router Redundancy Protocol (VRRP). rfc 3768, IETF, April 2004.

[5] IEEE 802.1Q: Virtual Bridged Local Area Networks. IEEE Standard for Local and metropolitan area networks, 2005.

[6] S. Knight, D. Weaver, D. Whipple, R. Hinden, D. Mitzel, P. Hunt, P. Higginson, M. Shand, and A. Lindem. Virtual Router Redundancy Protocol. rfc 2338, IETF, April 1998.

[7] G. Malkin. RIP Version 2. rfc 2453, IETF, November 1998.

[8] DDW-225 User Guide. Westermo Teleindustri AB, Doc. number 6642-22301. Wolverine Series, See http://www.westermo.com for updates.

[9] DDW-226 User Guide. Westermo Teleindustri AB, Doc. number 6642-22401. Wolverine Series, See http://www.westermo.com for updates.

[10] Lynx+ User Guide. Westermo Teleindustri AB, Doc. number 6643-2210. See http://www.westermo.com for updates.

[11] RedFox Industrial User Guide. Westermo Teleindustri AB, Doc. number 6641-22302. RedFox Series, See http://www.westermo.com for updates.

Index

