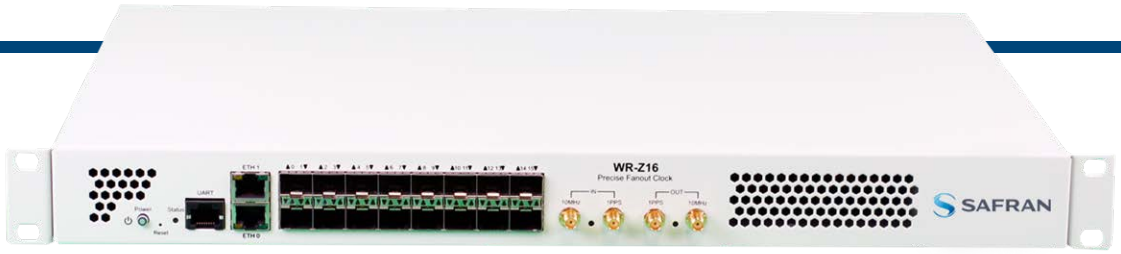
ELECTRONICS & DEFENSE

WR-Z16

MODEL



User Manual

Revision: v5.5

Date: 11-December-2025

SAFRAN

Safran Electronics & Defense

safran-navigation-timing.com

Safran Trusted 4D

• 45 Becker Road, Suite A, West Henrietta, NY 14586 USA
• 3, Avenue du Canada, 91974 Les Ulis, France

The industry-leading Seven Solutions products you depend on are now brought to you by Safran.

Do you have questions or comments regarding this User Manual?
➔ E-mail: techpubs@nav-timing.safrangroup.com

Blank page.

CONTENTS

CHAPTER 6

CHAPTER 7

CHAPTER 1

Introduction

The following topics are included in this Chapter:

1.1 About this Document

This document is the main user guide of the WR-Z16 Model . It describes the essential information about the WR-Z16 hardware, its features, and configuration options.

It is designed to allow users who have their first contact with the device to easily connect it to a management network and distribute precise timing (PTP or White Rabbit (WR), for example) through optical interfaces. It also provides advanced tips for expert users and/or details how to match the security policies defined at your company.

The set of official manuals also includes the **WRZ-OS API guide** for complete documentation about your WR-Z16.

1.2 About WR-Z16

The WR-Z16 is the reliable precise time fan-out for the most demanding time distribution applications on 1G Ethernet-based networks.

The WR-Z16 is a standalone device with 16 SFP connectors which provides sub-nanosecond accuracy time over plug-and-play fiber links. It provides very precise IEEE 1588 (PTP) in all its optical interfaces and supports NTP interoperability. Picosecond-level frequency distribution is available through digital clock. The WR-Z16 incorporates failover mechanisms which combine multi-source redundancy and holdover capabilities to ensure continued operation.

Its design is optimized for datacenter environment, where it is typically located in the top of the hierarchy level of the distribution network. The WR-Z16 can obtain the external time reference from its 10MHz and 1PPS SMA inputs, from another White Rabbit device through the SFP ports or it can work as a free running master device for the network.

A typical intra-datacenter network topology is shown in the figure below, where the WR-Z16 is working as the grandmaster and is the key element for distributing the timing through WR to each cabinet of the datacenter. Different end nodes are included in the diagram to illustrate the interoperability with different interfaces.

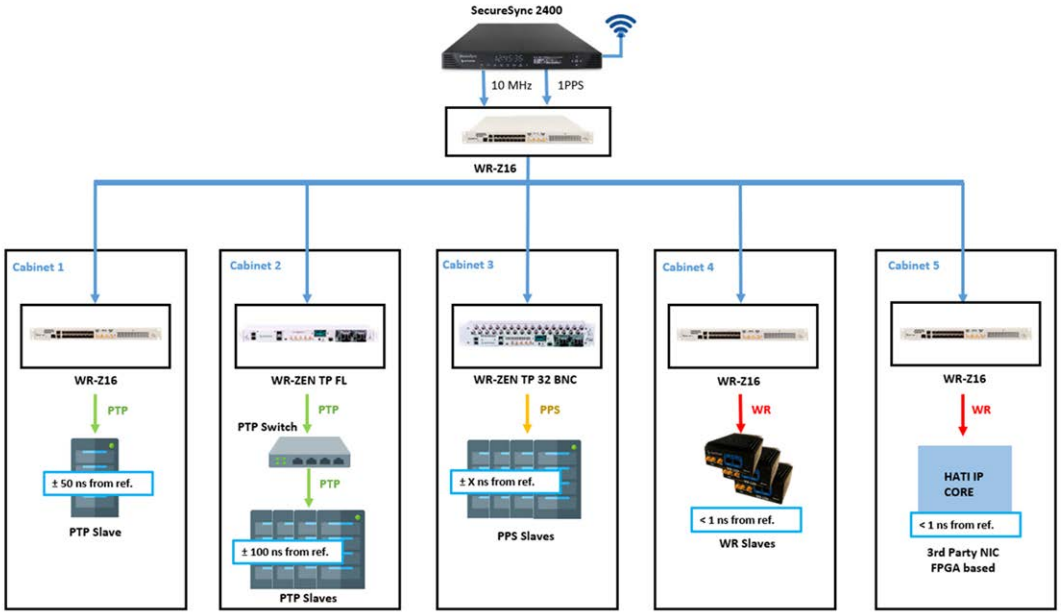


Figure 1-1: Intra-datacenter WR network topology

There are different options and licenses which enable specific functionalities of the WR-Z16. These options are described in the table below:

Table 1-1: Options and Licenses available in WR-Z16

Option /License	Description
Holdover option	An optional holdover oscillator can be included to maintain high accuracy (<1.5us/24h) even when all timing references are down.
PTP license	The device is shipped with the default profile of PTPv2/IEEE1588-2008. Other configuration including specific profiles support requires activation license.
HATI license	Enable the WR-Z16 to provide high accuracy synchronization to the HATI FPGA IP CORE. This license might be available per port or within a pack.

1.3 About WRZ-OS

The WR-Z16 is part of a full ecosystem of products which maintain sub-nano-second accuracy synchronization from an external time reference to the end nodes of the timing network, where different timing interfaces are provided to inter-operate with third-party equipment.

The **WR-Z16** devices and the **WR-ZEN family** (WR-ZEN TP, WR-ZEN TP-FL, WR-ZEN TP-32BNC) run on the same platform (**WRZ-OS**) sharing the same features, timing stack and set of tools.

All the devices running the WRZ-OS provide multiple interoperability options that include 1PPS/10 MHz signals, PTP, and NTP. They support SNMP v2/v3, rsyslog, and have an integrated web GUI for intuitive management and enhanced command line tools for advanced users. The following list contains the different form factors offered within the WR-ZEN family and a brief description of their main characteristics:

- WR-ZEN TP-FL: A fundamental, cost-effective 1U form factor version of the WR-ZEN TP. It has a front dual power supply. It includes 1PPS/10 MHz SMA outputs and PTP interoperability on its management interfaces. It can include a 4 x 1PPS expansion on demand.
- WR-ZEN TP: Standard 1U form factor version of the time provider. It accepts multiple fans and it has a rear dual power supply. It includes multiple 1PPS/10 MHz on the SMA or DB9 outputs, PTP interoperability on its management interfaces, IRIG-B, NMEA and ToD.
- WR-ZEN TP-32BNC: Expanded 2U form factor version of the WR-ZEN TP. It has a front dual power supply. It includes 1PPS/10 MHz SMA outputs and PTP interoperability on its management interfaces. The main characteristic is that it includes 32 BNC ports configured to work as 16 x 1PPS and 16 x 10 MHz outputs. A configuration with 32 x 1PPS is possible with an optional PPS expansion license.

1.4 About White Rabbit / High-Accuracy Technology

One of the key-features of the WR-Z16 family is that it fully supports the White Rabbit (WR) protocol, an extension of the IEEE 1588 (PTP), to achieve ultra-accurate sub-nanosecond synchronization in Ethernet-based networks. White Rabbit is the basis for the new High Accuracy (HA) profile in the latest PTP standard IEEE 1588-2019, and has the following characteristics:

- Time Precision: WR provides a common clock for physical layer in the entire network, allowing synchronization at sub-nanosecond level with picoseconds precision. In other words, the timing budget consumed by WR is almost insignificant.
- Scalability: WR networks are designed to be highly scalable supporting thousands of nodes and long-distance links within the range of metro area deployments. Its performance is not affected by traffic as with other PTP profiles.
- Cost effective solution: WR avoids expensive costs related to calibration and complex deployments with high requirements of maintenance, allowing plug-and-play links in Local Area Networks (LANs).

» Integration: WR is based on existing protocols and standards (such as PTP and Ethernet) so it is very easy to integrate into your existing network infrastructure.

BLANK PAGE.

CHAPTER 2

Product Description

The Chapter presents an overview of the WR-Z16 Model , its capabilities, main technical features and specifications.

The following topics are included in this Chapter:

2.1 Front panel

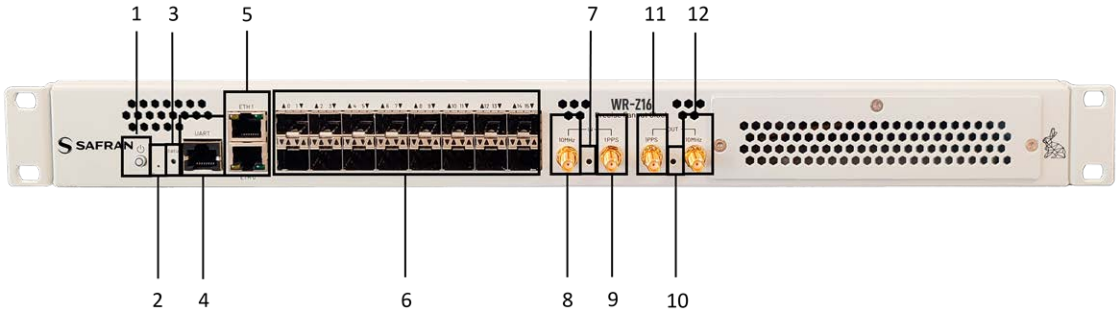


Figure 2-1: WR-Z16 front panel

Table 2-1: Front Panel Legend

#	Name	Information	Ref.
#1	Power button ⏻	Power On/Off the device	
#2	Reset Button	Button used to perform a factory reset or enter the recovery/failsafe mode	"Recovery Mode" on page 214
#3	Status LED	Green, Orange, Red	"System Status" on page 10
#4	1x Management UART (RJ45)	Serial UART RS232 on a RJ45 connector with pinout (USB-RJ45/RS232 adaptor not included)	"Logging from the UART" on page 20
		Pin #1: ✕ Pin #2: ✕ Pin #3 RXD Pin #4 GND Pin #5: ✕ Pin #6: TXD Pin #7: ✕ Pin #8: ✕	
#5	2x Management Ethernet (RJ45)	10/100/1000 ethernet network interface (eth0 & eth1)	"Product Specifications" on page 13
#6	16x SFP Fiber ports	1Gbps SFP compatible	"White Rabbit / IEEE 1588-2019 HA" on page 97"PTPv2.1 (Precision Time Protocol)" on page 103
Timing input			
#7	Timing Input LED	OK: Green; Warning: Yellow; Critical: Red	"Timing Input" on page 12

#	Name	Information	Ref.
#8	10 MHz input	● SMA connector (F) ● 50 Ω termination ● 1Veff (+/- 30%) digital or sine wave	" External Reference (GM)" on page 128
#9	PPS input	● SMA connector (F) ● 50 Ω termination ● TTL input (5V) / LVTTL input (3.3V)	" External Reference (GM)" on page 128
Timing output			
#10	Timing Output LED	OK: Green; Warning: Yellow; Critical: Red	"Timing Output" on page 11
#11	PPS output	● SMA connector (F) ● Digital output ● High level output: 3.0V +/- 0.2V (with 50 Ω termination)	" Virtual Clock Overview" on page 63
#12	10 MHz output	● SMA connector (F) ● Digital output ● High level output: 3.0V +/- 0.2V (with 50 Ω termination)	" Virtual Clock Overview" on page 63

2.2 Rear panel

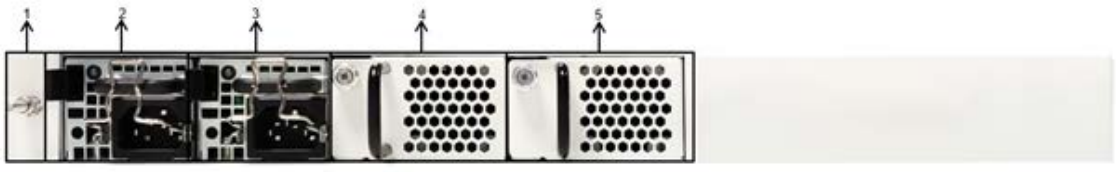


Figure 2-2: Rear panel of the WR-Z16

Table 2-2: RearPanel Legend

#	Name	Information	Ref.
#1	Ground	●Ground connector of the device 	
Power Supply			
#2	Power Supply #1	● Swappable & monitorable module: - 100- 240VAC, 50- 60Hz, (80W Max) 	
#3	Power Supply #2	● Swappable & monitorable module: - 100- 240VAC, 50- 60Hz, (80W Max) 	
Fans			
#4	Fan #1	Swappable Fan module with rear & front fans - Default airflow: blowing out	"Product Specifications" on page 13
#	Fan #2	Swappable Fan module with rear & front fans - Default airflow: blowing out	

2.3 Monitoring LEDs

The status of the WR-Z16 device can be quickly verified using the 3 visible LEDs in the front-panel. The tables below detail the behavior of each LED depending on the status of the WR-Z16.

The blinking behavior of the front panel LEDs is represented by the “Visual” column in the following tables using a sequence of three consecutive instants.

2.3.1 System Status

This LED is mainly used to inform the state of the system itself (Daemons loaded, Fans, power-supply, Temperature, CPU load, Available space, etc.). This led is also used to identify the various stages and modes of the booting procedure.

Table 2-3: Status LED behavior

Visual	Behavior	Description
🟢🟢🟢	Steady Green	Device system state is OK. There is no Warning or Critical alert.
🔴🔴🔴	Steady Red	There is a Warning or Critical alert related to the device. User might login to verify the source of the alert.
During booting procedure		
🟢⚫🟢	1x Blinking Green	Bootloader initialization OK
🟡⚫🟡	[1-15]x Blinking Yellow	Reset button is held during the booting procedure: - If released: entering recovery mode - If hold > 15s: entering reset factory mode
🔴🔴🔴	Steady Red	Device is booting in recovery mode
🟡🟡🟡	Steady Yellow	Device is booting in reset factory mode
🟢⚫🟢	2x Blinking Green	Booting in normal mode
⚫⚫⚫	Idle	Device is loading kernel and transitioning between modes
🟢⚫🟢	3x Blinking Green	FPGA initialization OK
🟡🟡🟡	Steady Yellow	The hald daemon has been loaded and it is waiting till all daemons are properly loaded
🟡⚫🟡	Nx Blinking Yellow	Device is loading in failsafe mode and for each module skipped during initialization the device blinks in orange

2.3.2 Timing Output

This LED is used to summarize the timing state (see "Timing" on page 59) of the device and if the user should expect to receive a PPS out from SMA connector according to the configuration of PPS Mode. Blinking behavior in this context refers to blinking continuously at 1Hz in parallel to the PPS output of the device.

Table 2-4: Timing Output LED behavior

Visual	Behavior	Description
🟢⚫🟢	Blinking Green	Device timing state OK
🟡⚫🟡	Blinking Yellow	Device timing state WARNING and the device is LOCKED to an active time source

Visual	Behavior	Description
	Blinking Red	Device timing state CRITICAL and PPS mode is 'Always ON'
	Steady Yellow	Device timing state is in a transitional WARNING. The device is not locked to a reference.
	Steady Red	Device timing state CRITICAL and PPS mode is 'Only Locked'
	Idle	The time manager module has not been loaded yet

2.3.3 Timing Input

The timing input LED is mainly used to quickly visualize the status of the external reference timing source (see " External Reference (GM)" on page 128) and the detection of PPS/10MHz inputs on the front-panel. Blinking behavior in this context refers to blinking continuously at 1Hz in parallel to the PPS output of the device.

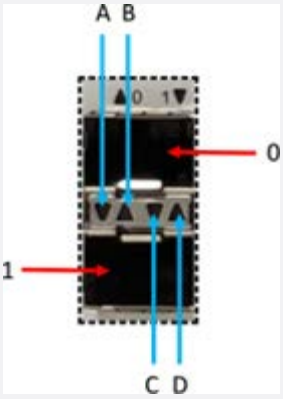
Table 2-5: Timing Input LED behavior

Visual	Behavior	Description
	Blinking Green	GM is locked, PPS and CLK signals are detected
	Blinking Yellow	GM is locked but PPS is not detected (PPS is configured as not mandatory)
	Blinking Red	The device is locking to its GM source. PPS & CLK on front panel are detected
	Steady Red	In a locking process with its GM source. The device lost the PPS signal or PPS & CLK signal at the same time on front panel
	Blinking Yellow	GM preset is active. PPS on front panel is detected.
	Idle	GM is not active and PPS on front panel is not detected

2.3.4 SFP Ports

The network ports of the device are arranged in a dual stack SFP cage. The following table represents only the two first ports but can be extrapolated to the other ones. The LEDs of these SFP ports are slightly different to standard usage as it does not differentiate TX/RX but utilizes the arrows to indicate the upper-/lower port and their corresponding states:

Table 2-6: Ports LED behavior

Visual	ID		Behavior	Description
	0		wr0	wr0 corresponds to upper SFP in the stack
	B	▲	Master / Disabled	Led B is disabled if this port is providing timing to other equipment (master mode) or disabled
		▲	Active slave	Led B is green when port is the active slave that discipline the device
		▲	Passive slave	Led B is orange when port is in passive/monitoring mode (§5.1)
	D	▲	Link down	When link is down led D is disabled
		▲	Link up	When link is up the led D stays in green
		▲	Activity	Blinks in orange each time a packet is received on this port
	1		wr1	wr1 corresponds to lower SFP in the stack
	A	▼	Link down	When link is down led A is disabled
		▼	Link up	When link is up the led A stays in green
		▼	Activity	Blinks in orange each time a packet is received on this port
	C	▼	Master / Disabled	Led C is disabled if this port is providing timing to other equipment (master mode) or disabled
		▼	Active slave	Led C is green when port is the active slave that discipline the device
		▼	Passive slave	Led C is orange when port is in passive/monitoring mode

2.4 Product Specifications

System On-Chip

- **SoC**: Xilinx Zynq 7000 series
- **CPU**: Dual ARM® A9 MP@ 1 GHz

- **Memory**:
  - 512 MB DDR3 (32-bit bus)
  - 16GB SD Card

Physical Dimension

- **Dimension**: 431 mm x 44 mm x 300 mm / (1 Rack Unit)
- **Color**: White (Metallic)
- **Certifications**: ROHS, FCC, CE

Environmental Conditions

- **Temperature**: -10ºC ~ +50ºC
- **Humidity**: 0% ~ 90% RH

Front Panel

- **UART** RS232 Serial (RJ45 connector)
- **Ethernet** 2x 100/1000 Base-T RJ45
- **SFP Ports** 16x 1GbE for timing distribution (WR/PTPv2 selectable)
- **Clocks I/O** 4x SMA connectors (3V @50Ω, TTL compatible):
  - 10MHz OUT (LVTTL)
  - PPS OUT (LVTTL)
  - PPS IN (TTL/LVTTL)
  - 10MHz IN (TTL/CMOS/ECL/clipped sine)

Back-panel

- **Power Supply** 2x Redundant & Hot-swappable
  - 100-240VAC, 50-60 Hz / 50W (max. 80W)
- **Fan** 2 x Swappable fan modules
  - Airflow: blowing out

2.5 Safety Notes

Safety: Symbols Used

Table 2-7: Safety symbols used in this document, or on the product

Symbol	Signal word	Definition
	DANGER!	Potentially dangerous situation which may lead to personal injury or death! Follow the instructions closely.
	CAUTION!	Caution, risk of electric shock.
	CAUTION!	Potential equipment damage or destruction! Follow the instructions closely.
	NOTE	Tips and other useful or important information.
	MULTIPLE POWER SOURCES	This equipment may contain more than one power source: Disconnect all power supply cords before removing the cover to avoid electric shock.
	EQUIPOTENTIALITY	Identify the terminal(s) which, when connected together, bring the various parts of the device to the same potential, not necessarily being the earth (ground) potential.
	STANDBY	Identify the switch by means of which part of the equipment is switched on in order to bring it into the stand-by condition, and to identify the control to shift to or to indicate the state of low power consumption.

SAFETY: Before You Begin Installation

DANGER!
Do not block the air vents which are located on the front panel of the device, the internal temperature might increase and damage the equipment.

DANGER!
The FAN modules must only be replaced by a skilled person. Once reinstated, its screw must be tightened up using a flat-blade screw-driver with at least 0.8Nm to avoid any manual manipulation.

DANGER!
Replacement of a power supply module has been intended only for occasional use by a skilled person. Hazardous energy inside the device might be accessible when a module is extracted. Do not make any kind of contact with any part inside the unit.

DANGER!
Installation of this product must be located in restricted access areas where only skilled persons are authorized. This product is not to be installed by the user/operator. Installation of the equipment must comply with local and national electrical codes.

DANGER!
This equipment must be earth grounded. Never defeat the ground connector or operate the equipment in the absence of a suitably installed earth ground connection. Contact the appropriate electrical inspection authority or an electrician if you are uncertain that suitable grounding is available.

Caution: To increase the lifetime of your device it is recommended to use it in a controlled temperature environment and limit to the ambient condition:
Temperature: -10°C ~ +50°C; Humidity; 0% ~ 90% RH

Note: The use of dust covers is recommended for the unused SFP/SFP+ slots.

2.6 Rack Installation

The device has been designed to be mounted in a standard 19-inch (48.3 cm) equipment rack and thus respect the following physical dimensions:

- Width: 431 mm
- Height: 44mm (1x Rack Unit)
- Depth: 300 mm

Caution:
The following guidelines are provided to prevent bodily-injury when mounting or servicing this unit in a rack:
-This unit should be mounted at the bottom of the rack if it is the only unit in the rack.
-When mounting this unit in a partially filled rack, load the rack from the bottom to the top with the heaviest component at the bottom of the rack.
-If the equipment rack is on wheels, ensure that the brakes are engaged and that the rack is stabilized.

Note:
Accessories: The screws needed to properly mount the device to the rack are not shipped with the equipment, nor the system ground kit. The device already mounts the L-brackets and is provided with a power cord C13 (European).

Note:
Airflow consideration: There are no standards for airflow in rack system but the device should be configured accordingly to the emplacement of hot and cold aisles. The default airflow of the device is from front-panel (cold) to back-panel (hot).

To properly mount the device to a rack cabinet:

1. Place the device on the floor or on a sturdy table near the rack.
2. Use a tape measure to verify the interior dimensions of the rack.
3. Carefully lift the device and position the rear of the device between the equipment rack mounting posts and slide the device into the rack until the

L brackets on the sides of the device are flush with the equipment rack front posts.

4. Align the mounting holes in the L bracket with the mounting holes in the equipment rack posts.
5. Secure the device using four 3/4-inch screws through the elongated holes in the L bracket and into the threaded holes in the mounting post (or the clip-nuts or cage-nuts).

2.7 Regulatory Compliance

2.7.1 EMC

- EN55032:2015
- AC:2016
- EN55035:2017
- EN61000-3-2:2014
- EN61000-3-3:2013
- FCC: 47 CFR Part 15B (10–1–15 Edition)
- ICES-003 Issue 6

2.7.2 Safety

- IEC 62368-1:2014
- AC:2015
- A11:2017

2.7.3 RoHS

- 2011/65/UE
- 2015/863/UE

CHAPTER 3

Device Connectivity

This chapter includes instruction to aid in device connectivity.

The following topics are included in this Chapter:

3.1 Default Configuration

The device is factory configured with the following default settings:

Table 3-1: Default Factory Settings

Port/ Service	Default Value
eth0	Waits for DHCP offer
eth1	192.168.77.100
Timing Preset	Slave on WRO (BC) WR master on all other ports
HTTPS	Disabled
SSH/web credentials	user: root password: root

The device credentials can be configured as needed. Learn more in "Security & Authentication" on page 145.

3.2 Connecting to the Device

There are two ways to connect to the WR-Z16 device using the terminal:

- Via UART
- Via SSH

This section will first introduce the general concepts on how to log to the device and then it will provide the specific steps, depending on the user OS.

3.2.1 Necessary Items for Connectivity

These are the required cables and adapters:

- RJ-45 Cat5/6/7 Ethernet cable.
- RJ45-RS232(m) and RS232(f)-USB cable.

3.2.2 Logging from the UART

In order to connect to the WR-Z16 device, it is required to connect the RJ45-RS232(m) and RS232(f)-USB cable to the RJ45 management port at the front panel.

Multiple software can be used to read from the UART in the computer depending on the operating system used for that purpose (e.g. Minicom, Picocom or Putty). Learn more in the below sections.

The following table summarizes the settings required to connect to the serial port (UART) of the device:

Table 3-2: UART Settings

Setting	Value
Baud Rate	115200 bps
Data	8 bits
Parity	None
Stop bits	1 bit
Flow Control	none

3.2.3 Logging from SSH

> **Caution:**
> Remote authentication servers (TACACS+/Radius): If there is any remote authentication server configured and it is responding, the local credentials of the user root will not be used. Only if the remote server is not working or is unreachable, will the local credentials be available again. This can render the device accessible only by UART.

There are three main ways to connect to the WR-Z16 device from SSH:

1. To use the default static IP on eth1:
   - Connect the RJ-45 Ethernet cable to the eth1 interface.
   - Connect the host interface to the same LAN and configure its IP address to be within the same netmask.
   - Access the device by typing in the host terminal: ssh root@192.168.77.100
2. To use a DHCP server on eth0:
   - Connect the RJ-45 Ethernet cable to the eth0 interface and to a LAN network with a DHCP server.
   - Connect the host interface (your PC) to the same LAN, to obtain another IP address.
   - Retrieve the IP address assigned to the device using UART or by scanning the local network.

Note: DHCP IP address: The assigned IP should persist between device reboots, as it will ask for the same address after every bootup.

3. To use a manual network configuration from UART:
   - » Connect the device using UART (see "Logging from the UART" on page 20).
   - » Follow the steps described in network configuration from CLI (see "Network configuration from CLI" on page 49).
   - » Reboot the device.
   - » Connect eth0 or eth1 (according to the user network configuration).

3.2.4 Connecting on Linux (Ubuntu 18.04 LTS)

3.2.4.1 Logging from UART

To connect to the device terminal via UART, use the RJ45-RS232(m) and RS232 (f) USB to connect to the UART management port, in the front panel of the WR-Z16 as shown in the "Front panel" on page 8 Hardware section.

The recommended software to manage UART connections on Linux is picocom, which you should be able to install just by running with super user privileges:

```
sudo apt install picocom
```

Once installed, the command to establish the connection with picocom is similar to:

```
picocom –b 115200 /dev/ttyUSB<X>
```

where ttyUSB<x> corresponds to the instance of the USB-UART driver. In most of the case, it will be ttyUSB0 (e.g., only one USB-UART cable connected to the PC) and the expected output is as follows

```
$ sudo picocom -b 115200 /dev/ttyUSB0
Calling 'sudo /usr/bin/picocom -b 115200 /dev/ttyUSB0 -b
115200'
Exiting Ctrl+A, then Ctrl+X
picocom v2.2
port is : /dev/ttyUSB0
lowcontrol : none
baudrate is : 115200
parity is : none
databits are : 8
stopbits are : 1
escape is : C-a
```

```
local echo is : no
noinit is : no
noreset is : no
nolock is : no
send_cmd is : sz -vv
receive_cmd is : rz -vv -E
imap is :
omap is :
emap is : crcrlf,delbs,
Type [C-a] [C-h] to see available commands
Terminal ready
```

Note: USB device discovery

The target device's name may vary depending on the names of other devices. The *dmesg | grep tty* command can be used to discover which name has been set to the connected device. This is an example output:

```
[4.616728] cp210x 3-6.1.2:1.0: cp210x converter detected
[4.620195] usb 3-6.1.2: cp210x converter now attached to ttyUSB0
```

In the case of the above output, `/dev/ttyUSB0` would be the device's name.

As a recommended alternative, *Putty* for Linux works properly. Programs like *minicom* or *screen* can be used, although they are not fully recommended for color compatibility issues.

3.2.4.2 Logging in from SSH

Ubuntu distributions (and many others) have already installed all ssh-related tools necessary to connect to the device. The user does not need to perform any specific steps and can directly follow instructions detailed in "Logging from SSH" on page 21.

3.2.5 Connecting on Windows

3.2.5.1 Logging in from UART

The connection to the UART in the WR-Z16 Model can be made by using Putty, the SSH and Telnet client for Windows, as it supports serial connections too.

When having connected the RJ45-RS232-USB cable to the Windows PC, a new serial port identified by COM<number> at the Device Manager, as can be checked in the figure below.

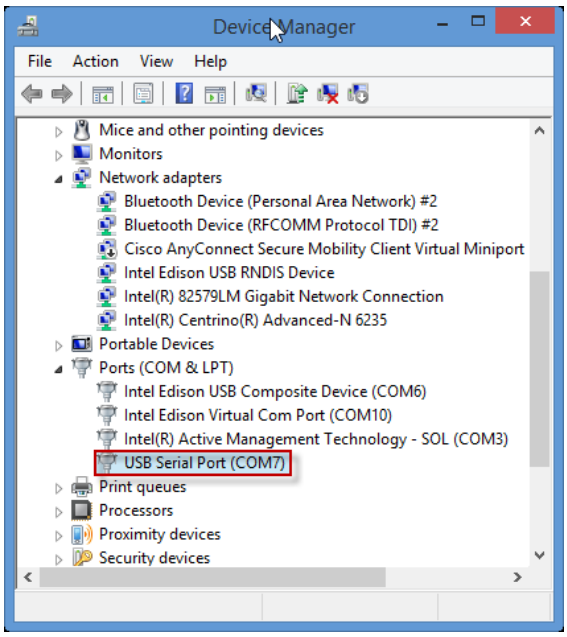


Figure 3-1: Device manager. New serial port detected.

Afterwards, the connection can be made through Putty. The connection type should be marked as Serial at <1>.The serial port name COM<number> should be placed at <2>, and the port speed (115200 Bd) at <3> (see below).

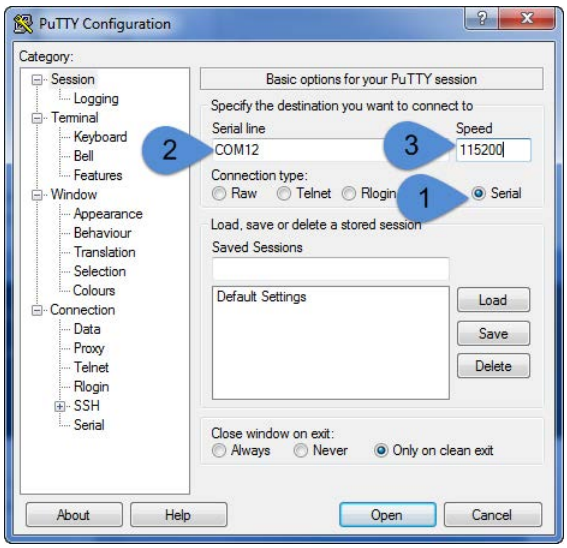


Figure 3-2: Putty configuration for serial port connection.

3.2.5.2 Logging in from SSH

It is also possible to connect to the device via SSH with Putty

The process to connect to the UART using Windows (XP, Vista, 7, 8, 10) is explained below:

1. Download and install the Putty Tool.
2. Verify that the Connection type corresponds to SSH .
3. Finally, write root@<IP> under the Host Name (or IP address) field, and click on Open.

Compatibility with wrz_config

In order to make Putty compatible with the wrz_config color scheme and avoid strangecharacters, it is recommended to try the following configuration:

1. Change remote character set to ISO-8859-1.
2. Uncheck “Override with UTF-8 if locale says so”.
3. Select “Use Unicode line drawing code points” (this is the default).

3.2.6 Logging in from web

Once the device is set with an IP address, it can be accessed by typing http://<device_ip> in the browser address bar. By default, https is disabled but if it has been enabled, the address bar should be replaced by https://<device_ip>.

Once connected the WR-Z16, the user must login by clicking on Login Page, and provide the corresponding password for the root user (see "Default Configuration" on page 20).

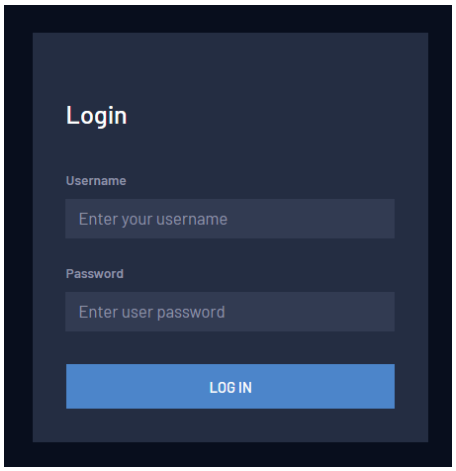


Figure 3-3: Login page of the web interface.

The main dashboard view will then be displayed.

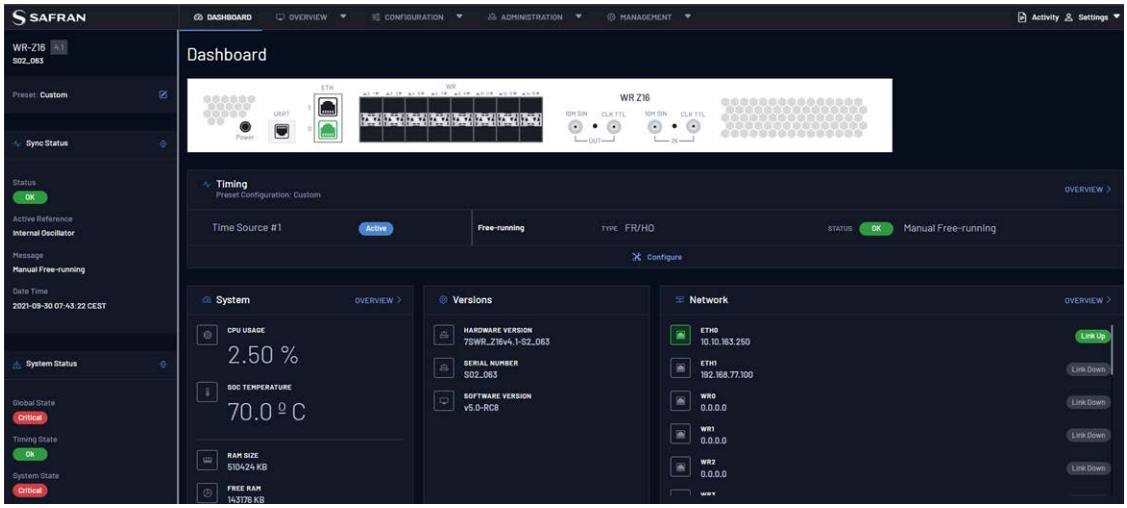


Figure 3-4: Dashboard page of the web interface

Further instructions on the available features and how to use them in the WR-Z16's web interface are detailed in the "GUI & CLI Tools" on page 27 section.

CHAPTER 4

GUI & CLI Tools

Depending on preference, a user can use the web GUI or the CLI tools to perform a standalone management of the device.

This section will briefly explain the main interactions for both methods and will also provide a detailed example of the configuration of the network interfaces.

The following topics are included in this Chapter:

4.1 Parameters API Introduction

Although the interaction using CLI & GUI differs, both approaches rely on the same mechanism handled by the “Generic Parameters Access (GPA)” core library.

In other words, most of the services running in the WRZ-OS can be configured & monitorized through common operations using the attributes of parameters. The definition of all attributes that a parameter must handle is given below:

- **OID**: a unique identifier that should be used that refers to a specific parameter. This OID is composed by three sub-indexes <M>.<D>.<P> corresponding to
  - <M>: ID of the module.
  - <D>: ID of the directory/path containing the parameter.
  - <P>: ID of the parameter inside a specific module directory.
- **Module**: The name of the corresponding module.
- **Directory**: The directory attached to the parameter.
- **Name**: The name of the parameter (Names in the GUI and CLI can slightly differ but their OID are always the same).
- **Type**: The type of value stored by the parameters.
  - String: Datatype to represent text.
  - Enum/Bool: Fixed list of String-Integer associations.
  - Integer: Integer number with different binary representations (u8, i8, u16, i16, u32, i32, u64, i64).
  - Decimal: Floating point number (f32 or f64).
  - Array: Vector of binary types handled like a separated string.
- **Unit**: Corresponding unit of the parameter including scale (i.e., s, ms, us, ns, ps).
- **Description**: Description of the parameter.
- **Access**: How the user can interact with a parameter.
  - Read: can read the value.
  - Write: can directly apply (online) the value.
  - Load: can save the value (it will be applied at next restart).
  - Disabled: Currently disabled, writing will not apply anything and the value read value might be invalid.

- **Visibility**: Expert parameters are by default hidden unless toggling the expert mode. Then, the disabled parameters are meaningless and thus temporary hidden to improve legibility.
- **Status**: Current status of a parameter.
  - Warning: The current value within a given situation corresponds to a warning alert.
  - Critical: The current value within a given situation corresponds to a critical alert.
  - Out-of-Sync: The current value could not be synchronized and is outdated.
  - Unlicensed: This parameter is invalid/unusable without the proper license.
  - Unknown: The whole module that handles the parameter is down.
- **Events**: Some relevant parameters are associated to events when they change their value (Tracked) or when their value enters an alert range (Warning, Critical) or a smart alert.

Note:
The list of all modules and corresponding parameters together with the value of their attributes can be found in the WRZ-OS API guide.

4.1.1 Table representation

For each specific feature explained through the user-guide, the following table format will be employed to describe the corresponding parameters along with their relevant attributes.

An example is given for the network interface where the same directory has been separated into two tables to follow the same structure as the web GUI panels. The following table corresponds to the parameters related to the configuration of the network interface.

Table 4-1: Configuration parameters of the network interface.

OID	Name	Value Type	Description
1.xxx0.x	/net/<iface>/xxx		Directory related to the <iface> name where OIDs follow the given pattern: wr0 → 201x, wr1 → 211x, ..., wr15 → 361x & eth0 → 680x, eth1 → 690x.
0.xxx0.7	DHCP	<Bool>	Enable/Disable the DHCP IPv4 discovery.

OID	Name	Value Type	Description
0.xxx0.3	IPv4 Address	<Array> [4 x u8]	IPv4 address of <iface> (format: [0-255].[0-255].[0-255].[0-255]).
0.xxx0.4	IPv4 Net-mask	<Array> [4 x u8]	Subnet mask.of <iface> (format: [0-255].[0-255].[0-255].[0-255]).
0.xxx0.5	IPv4 Gate-way	<Array> [4 x u8]	Default gateway for <iface> (format: [0-255].[0-255].[0-255].[0-255]).
0.xxx0.15	Description	<String>	Description of <iface>.
0.xxx0.16	Auto-nego-tiation	<Enum> 0. Enabled 1. Disabled	Enable/Disable auto-negotiation on <iface>.

The next table corresponds to the parameters that provide information (read-only) about the corresponding interface:

Table 4-2: Information related to the network interface

OID	Name	Value Type	Description
1.xxx0.x	**/net/<iface>/xxx**		**Directory related to the <iface> name where OIDs follow the given pattern: wr0 → 201x, wr1 → 211x, ..., wr15 → 351x & eth0 → 680x, eth1 → 690x.**
0.xxx0.1	Status	<Enum> 0. Link Down 1. Link Up 2. Not Found	Status of the interface.
0.xxx0.2	Ethernet Address	<Array> [6 x u8]	MAC address of the corresponding interface with upper case hexadecimal format (e.g., 64:FB:81:20:84:06).
0.xxx0.8	Speed	<String>	Auto-negotiated speed of <iface>.
0.xxx0.9	Tx Packets	<Integer> (u32)	Transmitted packets on <iface>.
0.xxx0.10	Rx Packets	<Integer> (u32)	Received packets on <iface>.
0.xxx0.11	Tx Bytes	<Integer> (u32)	Transmitted bytes on <iface>.
0.xxx0.12	Rx Bytes	<Integer> (u32	Received bytes on <iface>.
0.xxx0.13	Tx Errors	<Integer> (u32)	Transmission errors on <iface>.
0.xxx0.14	Rx Errors	<Integer> (u32	Reception errors on <iface>.

4.2 The REST-API

After upgrading the WRZ-OS to v5.0 or higher, REST API functionality is available.

4.2.1 Accessing the REST-API Documentation

The REST-API documentation is accessible in any WRZ device by accessing to `http://<ip>/API/`, where `<ip>` is a valid IP address that reach the device (e.g.`http://192.168.77.100/API/`). Once the user is logged in, a web page is shown where all the methods and its corresponding examples are described.

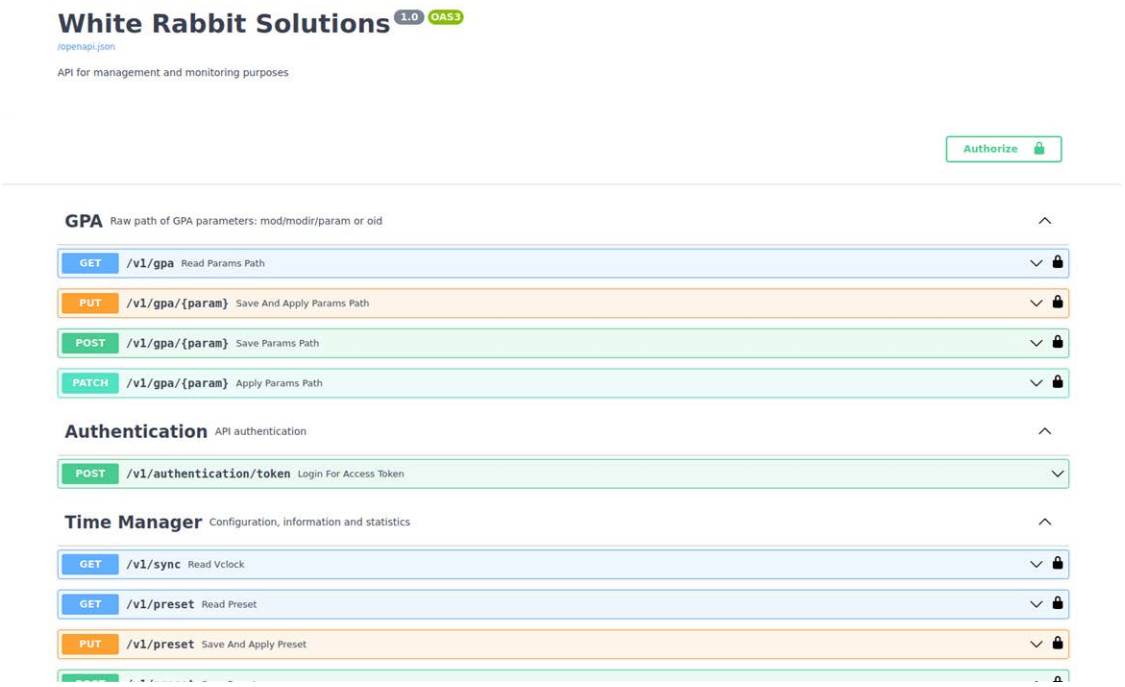


Figure 4-1: Main page of the REST-API documentation.

4.2.2 Interacting with the REST-API

The REST-API is accessible via `http://<ip>/API/`. The user can make HTTP requests with third-party applications such as curl, Postman or via its own application. Also, the REST-API is testable directly from the documentation page.

4.2.2.1 HTTP Methods

The user can interact with the device using four HTTP methods:

- **GET**: Reads the value of the parameter.
- **PUT**: Changes the parameter applying a given value, and saves it to make it persistent to reboots.
- **POST**: Saves the value of a parameter without applying it. The parameter will have the new value after the next reboot.
- **PATCH**: Applies a value to a parameter without saving it.

4.2.2.2 GET Formats

The GET method can retrieve different levels of information, called 'formats'. There are five formats:

- **value_string**: Retrieves the value as a string.
- **value_number**: Retrieves the value as a number.
- **basic**: Retrieves the value as a string and as a number, among its current status (Warning, critical,...)
- **config**: The same as the basic, plus the configured (saved) value.
- **complete**: All the retrievable information of a parameter. Is the same as the 'config' format, plus:
  - oid: Parameter's primary key in owner_id.modir_id.prm_id format.
  - access_str: The access permissions to the parameter in string format.
  - vtype_str: The type of the parameter in string format
  - unit: Unit of the parameter
  - desc: Description of the parameter
  - enum_dict: All the possible settable values of the parameter, both with its corresponding numbers numbers and strings.
  - metrics_dic: The computed metrics for the parameter.

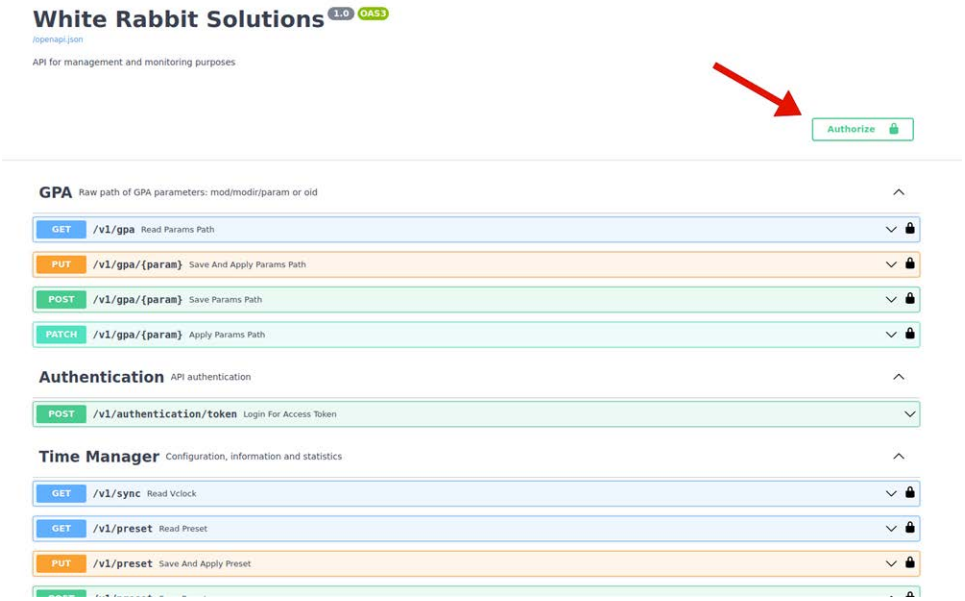
Note: To know the supported values of a parameter, the user shall make a complete query where the enum object will show the valid range.

Note: For monitoring purposes, use value_string or value_number format because of their light computation impact. Other formats require more computation time and will affect the system performance if they are requested frequently.

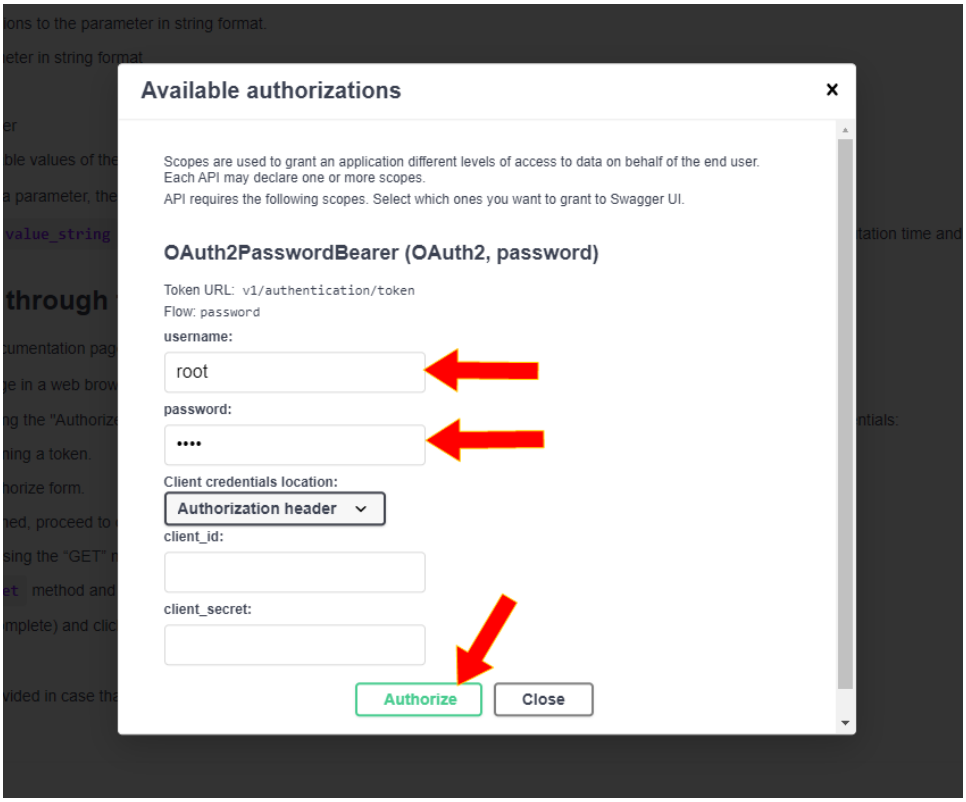
4.2.3 Accessing the REST-API through the documentation page

The REST API is testable directly from the documentation page. For doing this:

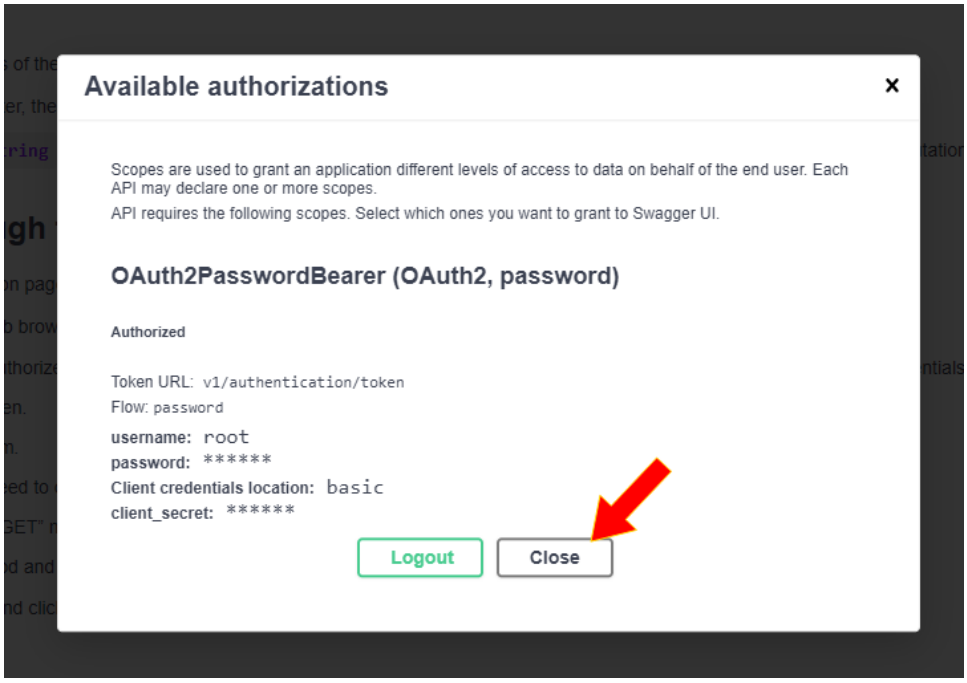
1. Open the REST-API documentation page in a web browser by navigating to `http://<ip>:8201/`
2. Obtain an authentication token by selecting the “Authorize :lock: “ button on the top of the page and inputting the credentials:
   a. Open the Authorize dialog for obtaining a token:



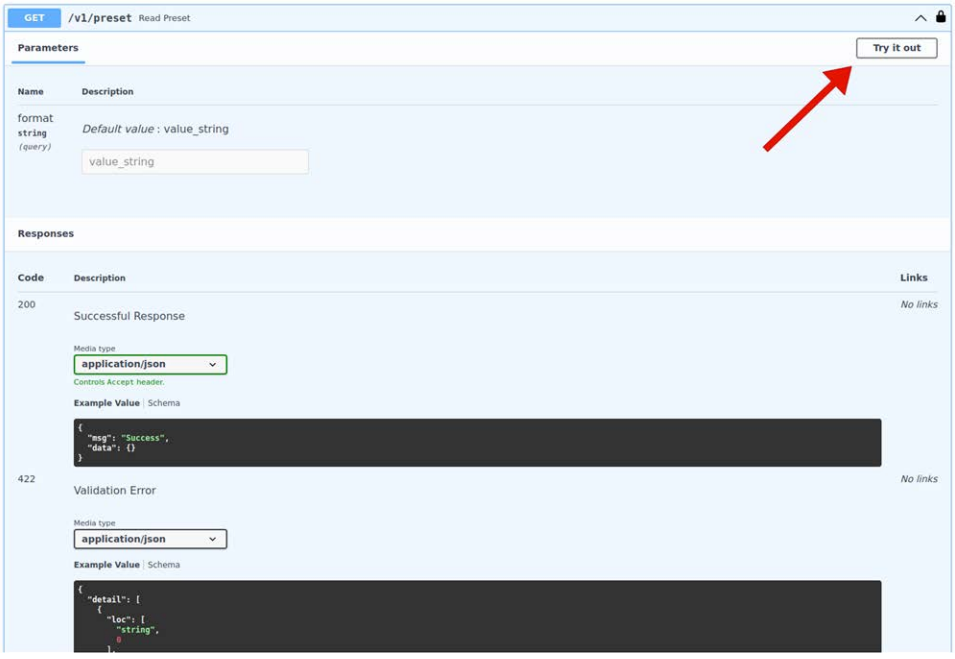
   b. Enter the credentials in the Authorize form.



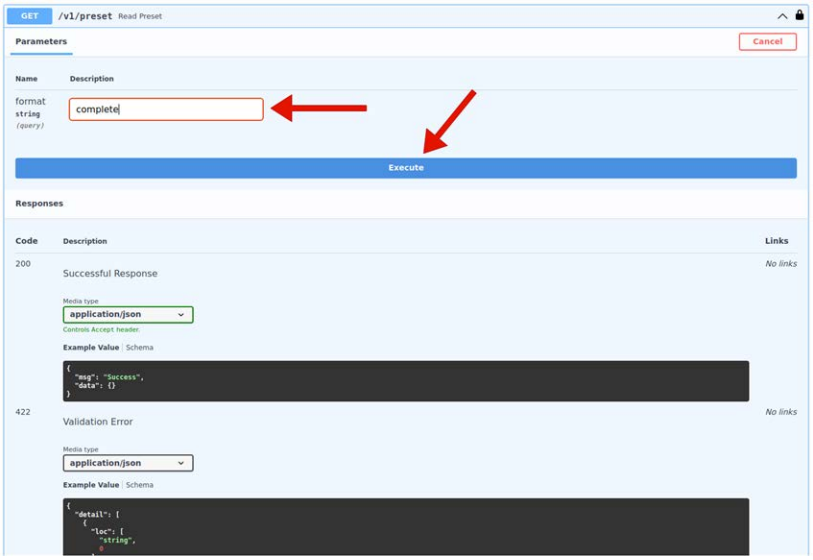
c. Once the token has been obtained, close the dialog box and proceed to test the REST-API.



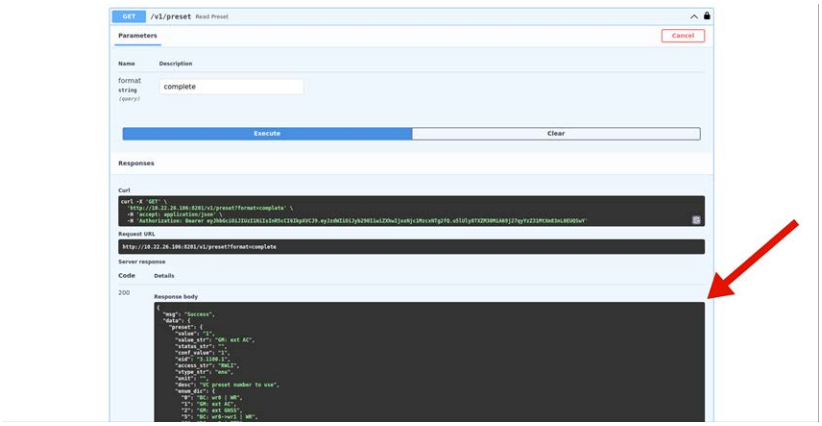
3. Choose a parameter and a method. If using the “GET” method, choose also a “format”. For example:
    a. Open the `GET /v1/preset` method and select “Try it out”.



    b. Choose a format string (for example: complete) and select “Execute”.



c. Check the response of the request.



4. An example line for curl is provided for testing purposes:

```
curl -X 'GET' \
'http://10.22.26.106:8201/v1/preset?format=complete' \
-H 'accept: application/json' \
-H 'Authorization: Bearer eyJh-
bGciOiJIUzI1NiIsInR5cCI6Ik-
pXVCJ9.eyJzdWIiOiJyb290IiwiZXhwIjoxNjc1MzY3Mjk1fQ.RGDFVCnbG-
ltHa9PfiAqLZXpaw5R_wPtOWe8R6DYZhQ'
```

4.2.4 Accessing the REST-API through Postman

4.2.4.1 About Postman

Postman is an HTTP client that serves as a development app to prototype and test APIs. The Postman app is available for Microsoft Windows™, Mac OS X or later, and Linux; there is also a web browser version: http://www.postman.com/downloads/ . Postman is a software app developed by Postdot Technologies, Inc. In its basic configuration, the tool is free of charge.

Postman can be used to send the requests to a White Rabbit Time Server unit, and it will return the JSON response from the device. This allows the user to quickly test API calls without having to develop test software, and the format of the data returned can be easily analyzed for inclusion into scripts or applications that can consume the data.

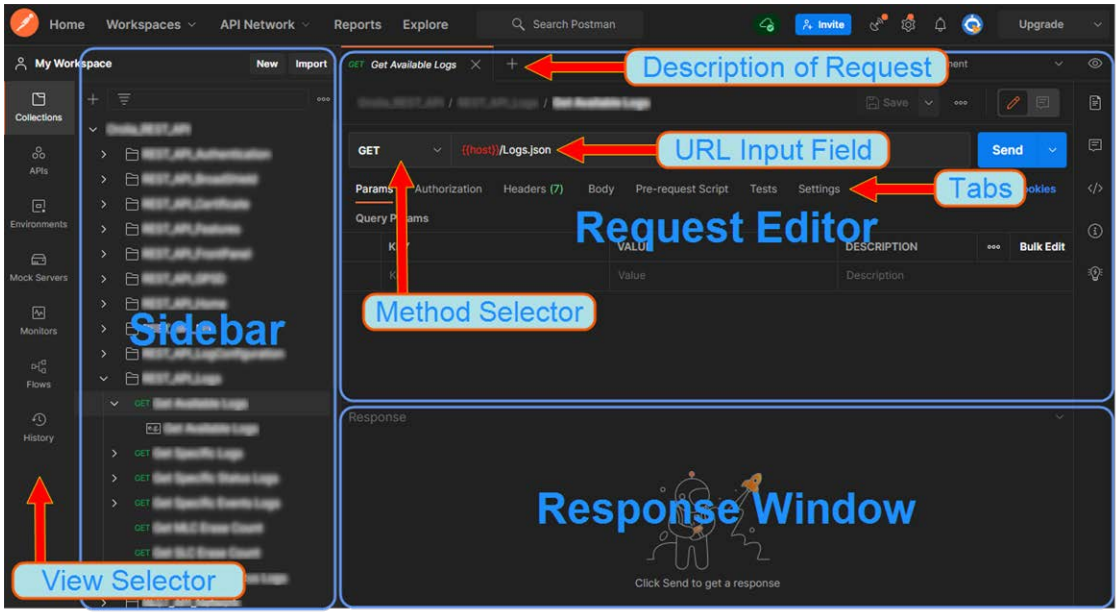
4.2.4.2 Downloading and Installing Postman

1. Navigate to http://www.postman.com/downloads/, and select "Download the App" or "Try the Web Version".
2. Install and launch Postman.
3. Create an account by signing up. This will ensure your requests, collections, environments and history data are saved for future reference.

4.2.4.3 Familiarizing Yourself with Postman

The following is a brief overview of the Postman UI. More comprehensive assistance can be found under https://learning.postman.com/docs/getting-started/introduction/.

The **View Selector** is used to switch between different functionality.



The **Sidebar** lists the requests stored in the loaded Collections tab, the parameters stored in the Environments tab, and – under the History tab – a list of recently submitted requests.

The Request Editor is used to configure elements of a request. For example:

- **URL**: Type the address of the endpoint that you want to call into the URL input field. URLs used previously will be suggested via auto-complete.

  Note that parameters entered in the URL input field bar or in the key/value editor will not automatically be URL-encoded. To manually encode the parameter value, right click on a highlighted text, and select EncodeURIComponent.

- **Method**: Select the HTTP operation that you want to use (GET, POST, etc.).
- **Tabs**: Use the tabs in the Request Editor to configure the requests. For example:
  - **Headers**: Click on the Headers tab to display the Headers key-value editor. The header frequently contains fields for authentication, Cookies, a time stamp, MD5 sums, and MIME-content type information.
  - **Body**: Click the Body toggle switch to open the Body editor. The body editor has different controls depending on the body type: form-data, urlencoded, raw, binary, and GraphQL.
  - To edit key values, click the **Params** button.

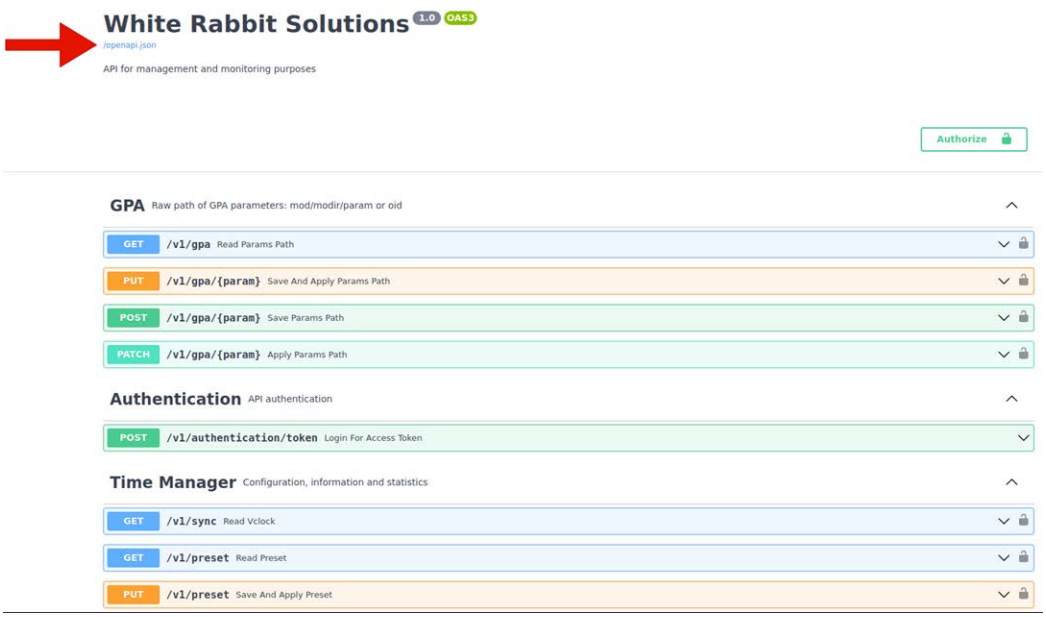
Postman functionality highlights:

- Create requests by conveniently specifying Method, URL parameters, Header and Body.
- Submit API calls quickly to test scripts; generate code snippets that can be copied and pasted.
- Specify authorization to be used.
- Display responses in different formats e.g., "pretty", "raw", or as rendered HTML pages.
- Organize and store requests in Collections.
- Store request parameters that will be used repeatedly (e.g., keys and values used as login credentials) in development project-specific Environments.
- Access history of sent requests.
- Capture documentation for requests in a description field.

4.2.4.4 Importing the White Rabbit Collection

Safran's Postman™ collection for White Rabbit Time Servers provides examples of how to pull and send data through the API. To import this collection:

1. Click on /openapi.json to download the collection:

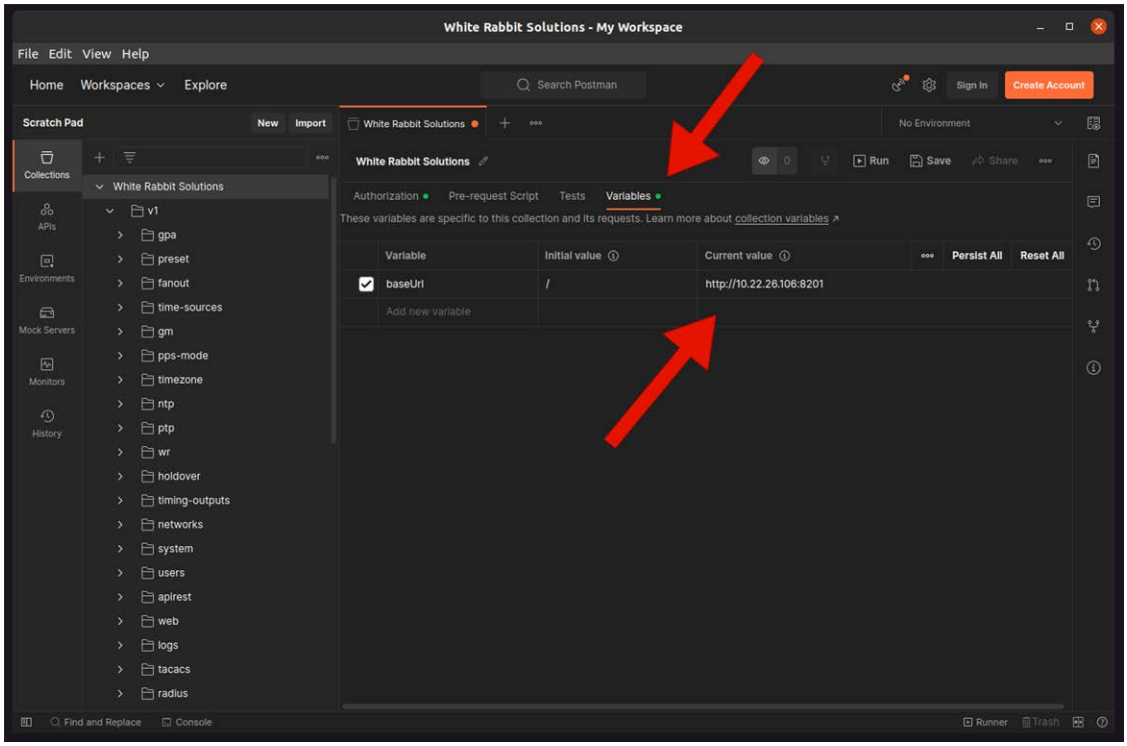


2. Open the Postman app, using the credentials of your previously created account.

3. Import the White Rabbit REST API Collection:
    a. For the standalone app, select Import.
    b. Drag and drop or click Upload Files and select the `openapi.json` file from your previously downloaded.
    c. Under the Collections tab in the Sidebar on the left, “White Rabbit Solutions” will be displayed. Click on it to display the Collection's folders. Click on any request to display it in the Request Editor.
4. It is not necessary to import an Environment into Postman, because all of the necessary variables are attached to the collection itself.

4.2.4.5 Configuring the ‘baseUrl’

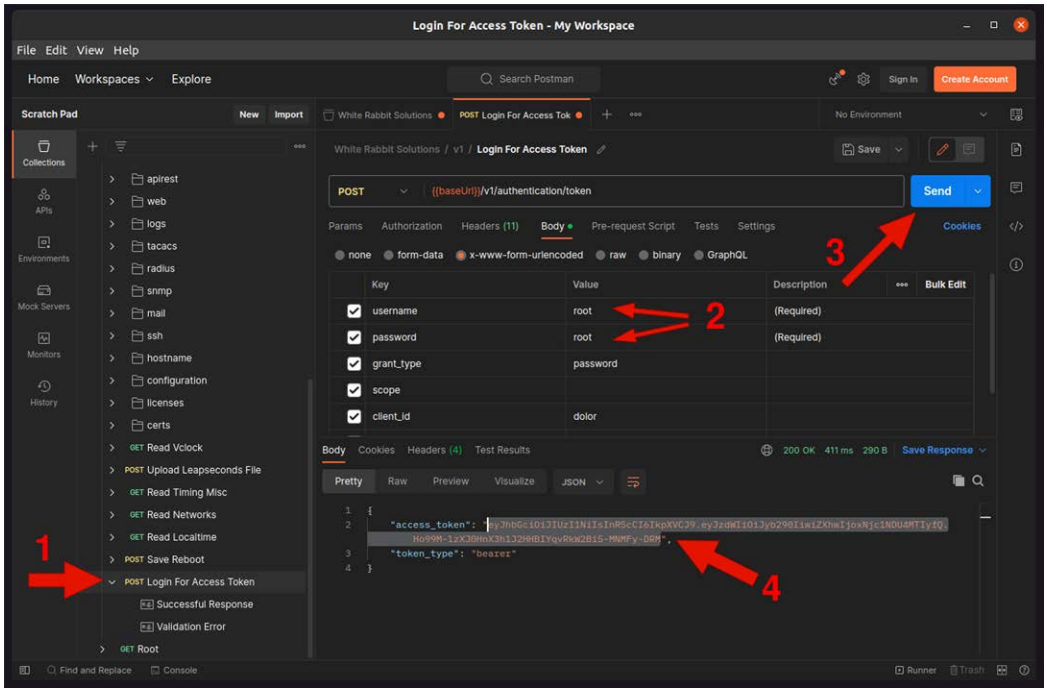
In order to indicate to Postman the device to which connect to, the user must introduce the base URL of the device’s REST-API.To do so, click on the root element on the sidebar titled “White Rabbit Solutions”, and in the “Variables” tab, edit the Current Value of the baseUrl variable and set it to `http://<ip>:8201/`, where <ip> is a valid IP address that reach the device (e.g. `http://10.22.26.106:8201/`)



4.2.4.6 Obtaining the Access Token

Once the `baseUrl` is set, the user can request an Access token to be able to make any other requests to the device. The needed steps are:

1. Open the POST request named “Login for Access Token”.
2. In the **request editor**, edit the `username` and `password` variables with the appropriate values for the device.
3. Click on the “Send” button.
4. In the **response** window, copy the value of the returned “access_token” object.

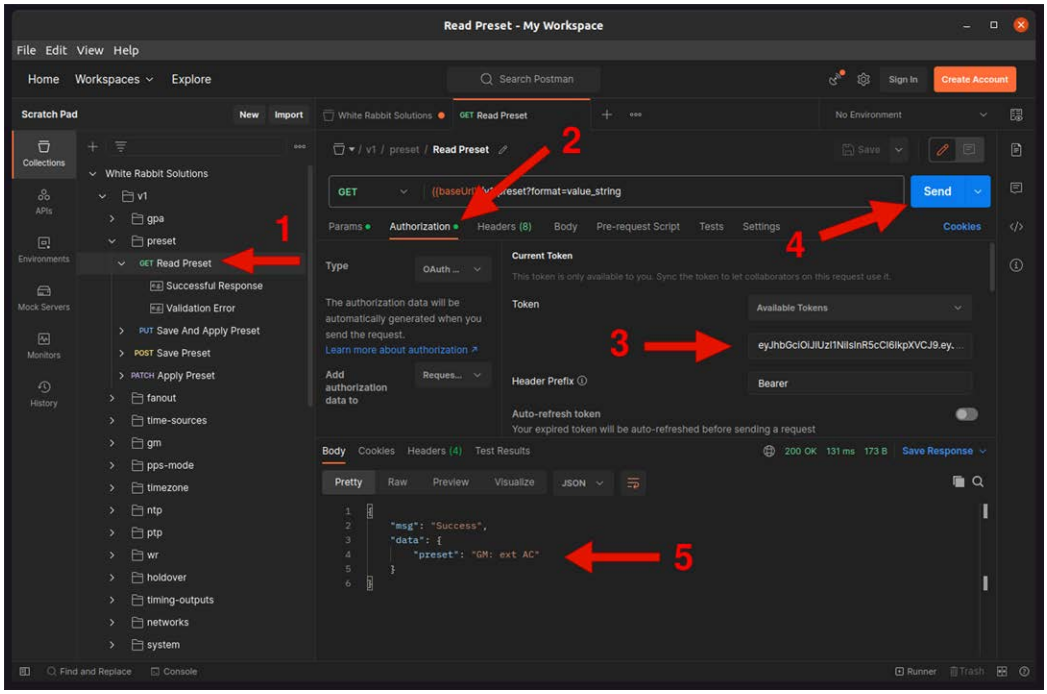


4.2.4.7 Making a Request

An authenticated user (i.e. an user with an authentication token) can make any type of request. For doing this:

1. Open the desired request in the sidebar (for example, /v1/preset/Read Preset).
2. Navigate to the “Authorization” tab.
3. Introduce the access token.
4. Click on the “Send” button.

5. Check the response in the response window.



4.3 The Web GUI

The web GUI is a user-friendly interface that allows you to monitor and manage the device through a web browser.

After navigating to the device's IP address, you will need to login using the default credentials (or your updated credentials). A dashboard tab will then be shown with the main information of the device, product image, status panel, basic timing information, system and version information, and network configuration.

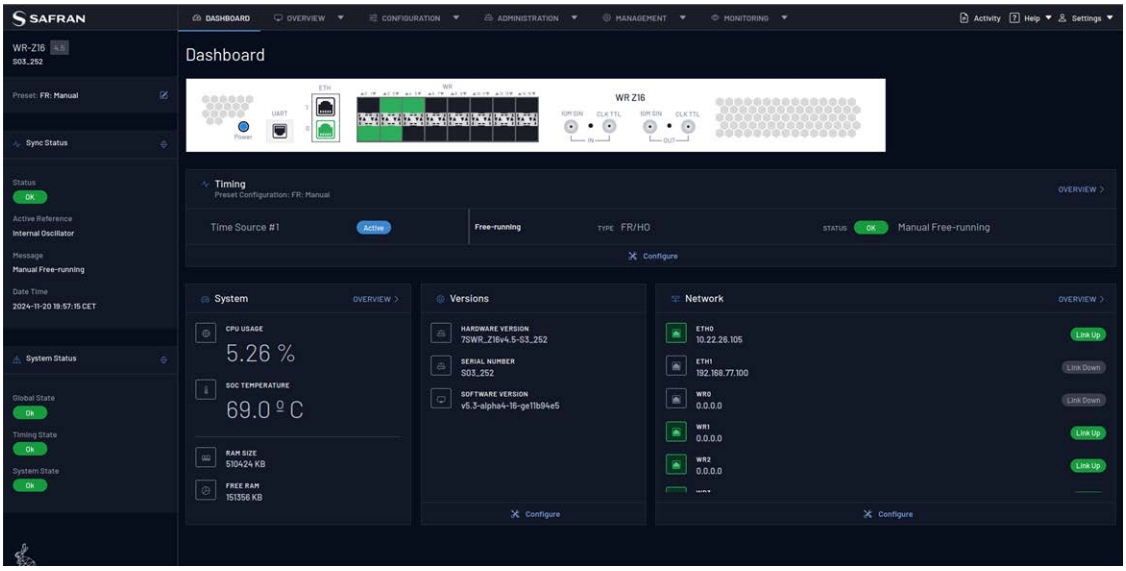
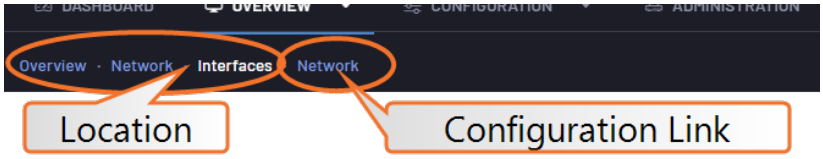


Figure 4-2: Dashboard in web interface

The Web GUI main dashboard is divided into the following main parts:

- The **Main Navigation Bar** contains the following menu links:
  - The **Overview Menu** contains links to overview status pages for the following fields:
    - Timing: General, White Rabbit, IEEE 1588- 2008 (PTPV2), External Reference (GM), Holdover, NTP, and Misc.
    - Network: Interfaces, DNS
    - Health Information: System, Power Supplies, Fan

    The Overview sub-menu will contain a link to the settings page, if applicable:



  - The **Configuration Menu** has links to the configuration of General Timing, White Rabbit, PTP, External Reference (GM), Holdover, NTP, and Misc.
  - The **Administration Menu** allows users to configure Network, SNMP, and Security settings.

- The **Management Menu** links to pages for Info, Logging, Maintenance, Firmware Update, System Alerts, Licenses, and Import/Export config.
- The **Help Menu** is located on the right side of the **Main Navigation Bar**. This drop-down menu contains the following subsections:
  - **Manuals:** contains a link to the most recent user manual of the device in use.
  - **Technical Support:** contains Application Notes, FAQs and live support.
  - **Register Product:** register any product.
  - **Leave Feedback:** allows the user to leave feedback based on their experience.
- The **Activity Menu** provides a log containing major device events.
- The **Settings Menu** allows the user to change passwords or select a light or dark theme. It also allows the user to logout from the device.

- The **Product Image** provides real-time depictions of the status of any connected ports or power supplies (functioning interfaces will appear GREEN).
- The **General Status Panel** contains at-a-glance sync status and system status information.
- The **System Status Panel** contains metrics related to system functionality (temperature, memory used, etc.) and contains a link to the overview page for system status information.
- The **Versions Status Panel** lists your current hardware version, serial number, and software version.
- The **Network Status Panel** details the IP address information for each port.

Figure 4-3:

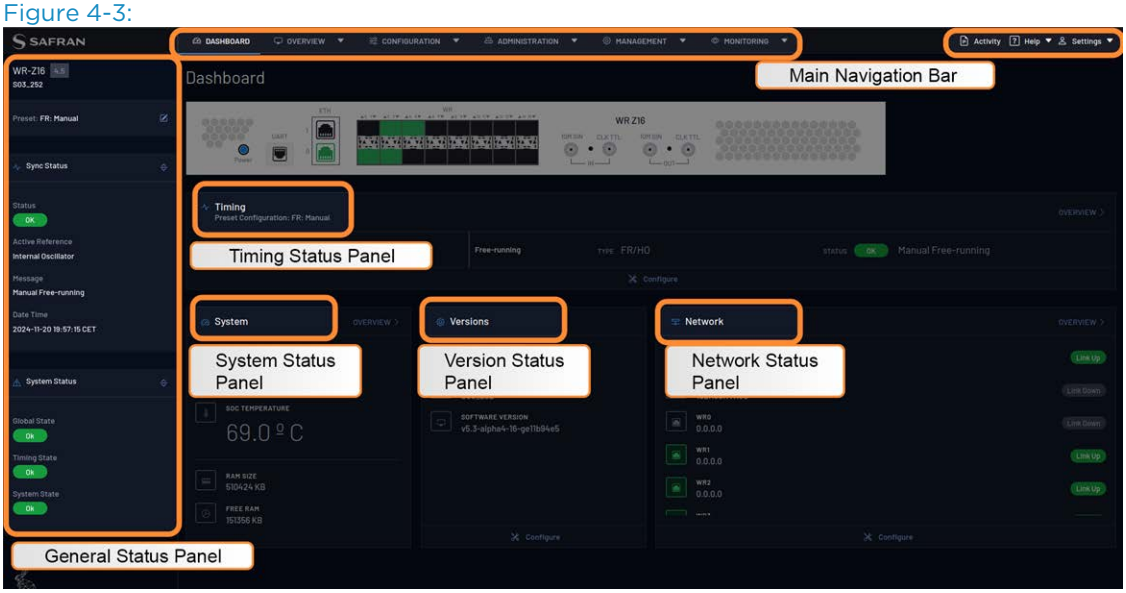


Figure 4-4: Web GUI Dashboard sections

The web session will automatically terminate after 15 minutes of inactivity. This setting can be disabled or configured by navigating to **Administration** > **Security** > **Web**.

4.3.1 Network configuration from web

To illustrate how the user should interact with the web interface, an example of the configuration of a static IP for eth0 network interface follows.

1. After logging in to the unit, select the Administration drop-down in the Main Navigation bar, and then select the Network page:

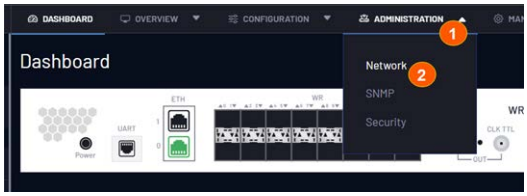


Figure 4-5: Web GUI Network Page Navigation

2. From the Interfaces tab, identify the port you wish to configure. If DHCP is enabled, the values will be read-only. To disable DHCP, select the **no** radio button. It will then be possible to fill in your network information.

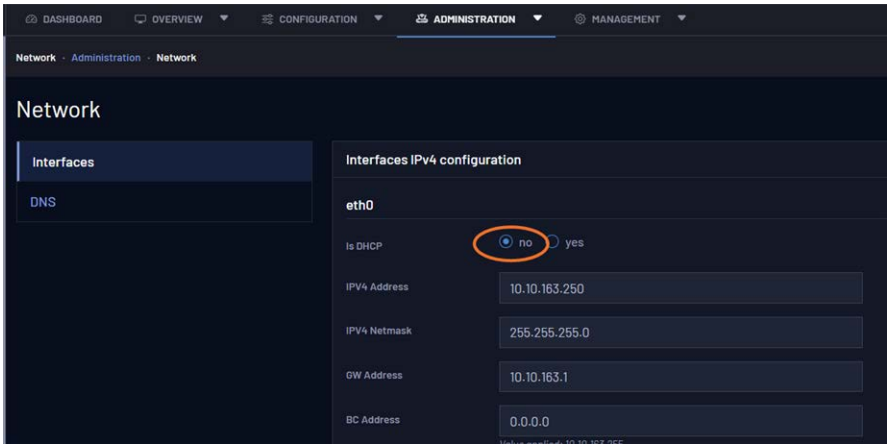


Figure 4-6: Disable DHCP via the Web GUI

3. After entering your chosen static IP address and networking information, select the **Save Configuration** button at the bottom right of the page.
4. A green confirmation banner will acknowledge your saved configuration; you will also see a warning that the saved changes will not be applied to the functionality until the unit is rebooted:

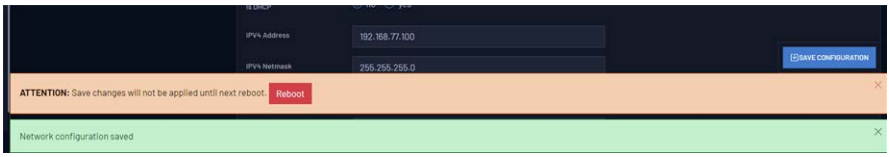
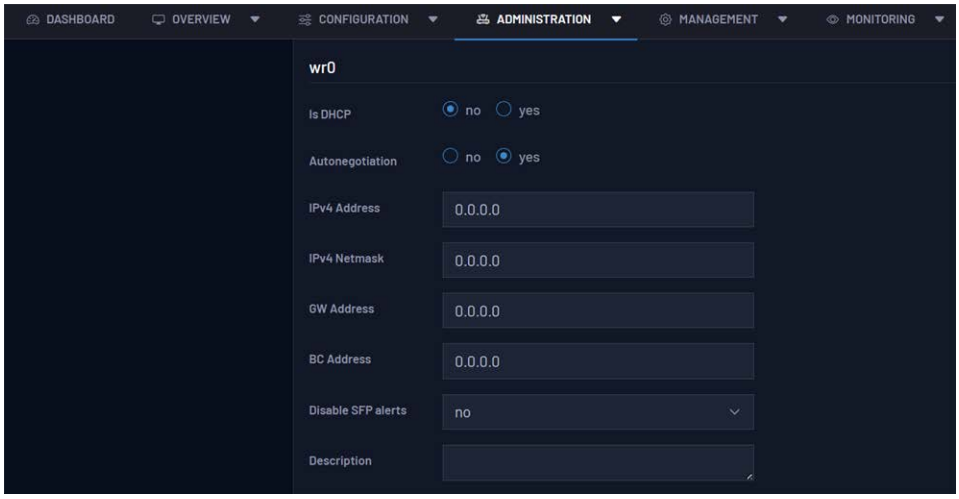


Figure 4-7: Network change banners at the bottom of the page

5. Once you have made your desired changes, you can either select the **Reboot** button from the warning banner, or navigate to **Management** > **Maintenance** and select the device Reboot button.
6. Your changes can be confirmed from the dashboard (which lists all interface IP addresses in the Network Status Panel) or you can also return to the **Administration** > **Network** page.

Note:
DNS Resolution: The last tab of network configuration entitled DNS can be used to add a custom DNS server needed to resolve IP address. This is useful in case an URL is used instead of an IP when configuring the server for a device (e.g., NTP, Auth, etc.).

Within the same page, you may also configure individual interfaces:



Note: If configuring auto-negotiation as "off" with 2 devices involved in communication, both devices should be configured without auto-negotiation.

Note: For more information on SFP alerts, see "SFP Alerts" on page 185

4.4 CLI Configuration

If the user prefers to configure the device from the command line, execute the `wrz_config` command.

The `wrz_config` tool provides an interactive menu directly from the command line with a structure similar to the web GUI. The main menus are the following:

- Timing: All timing-related configuration of the device (see "Timing" on page 59 for more information).
- Network: Network configuration of the management & timings interfaces ("Network configuration from CLI" on page 49).
- Healthing: Power Supplies & Fans related configuration ("Healthing " on page 179).
- Security: Configuration related to the security of the device ("Security & Authentication" on page 145).

» Management: Logging & Monitoring ("Monitoring & Logging" on page 157) configuration and aspects related to the maintenance of the device ("Device Maintenance" on page 201).

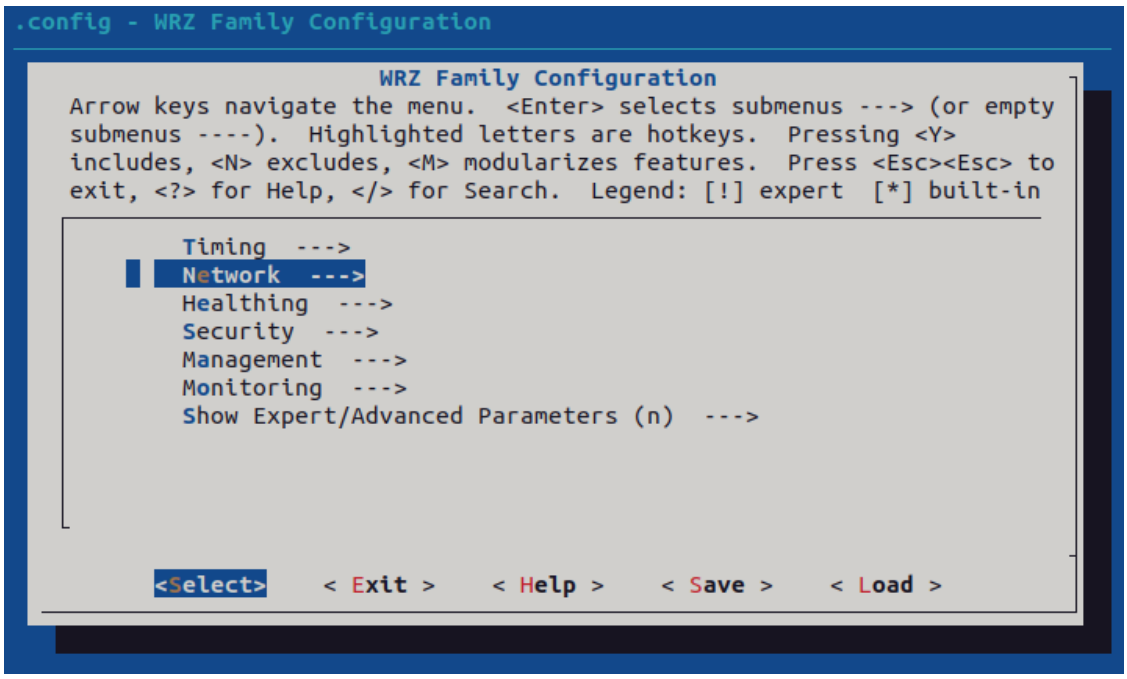


Figure 4-8: Main wrz_config interface. Modules to modify

The user can then navigate between the different menus and sub-menus using arrow keys and the <Select> and <Exit> actions.

To get more information about a specific parameter such as its description or its corresponding OID the user can press the <Help> action.

When exiting the `wrz_config` tool, the user will be asked if he wants to save the changes in the configuration or not.

Note:
Expert parameters: To ease the navigation during configuration some expert parameters are by default hidden. In the main, menu the user can toggle a configuration flag to make visible these expert parameters.

Note:
Changes applied at reboot: It is important to highlight that the changes performed through the wrz_config tool will only be applied at next reboot. Indeed, each init.d services will load their corresponding values from /root/.config file during startup.

Caution:
Avoid manual editing of .config file: In order to avoid errors such as duplicated entries, it is not recommended to manually edit the /root/.config. This might have been suggested for some specific configurations of the previous version (WRZ-OS v2.x) but this practice is now discouraged.

4.4.1 Network configuration from CLI

To illustrate the usage of `wrz_config` tool, the configuration of the network interface is detailed as a step-by-step procedure. The behavior can then be emulated in other menus in the configuration tool.

To set the IP address for the eth0 management interface:

1. The first step is to execute the wrz_config command from a terminal.
2. Then, the Network section must be selected from the main menu.

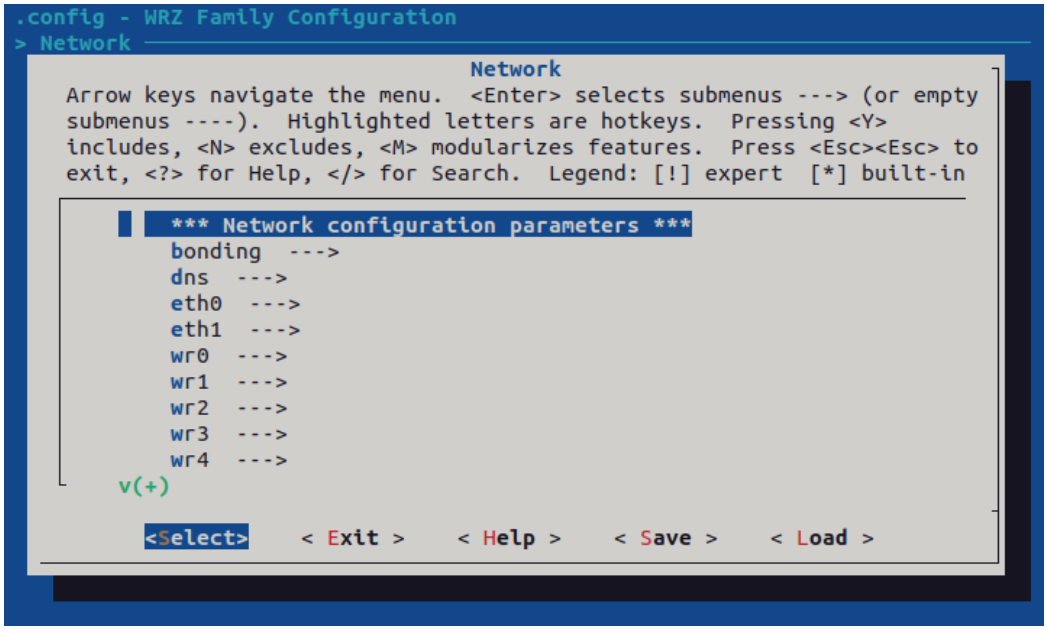


Figure 4-9: wrz_config interface. Network interfaces to change

3. Then the corresponding network interface (eth0) to be modified must be selected.
4. If the static IPv4 settings must be loaded, disable DHCP.

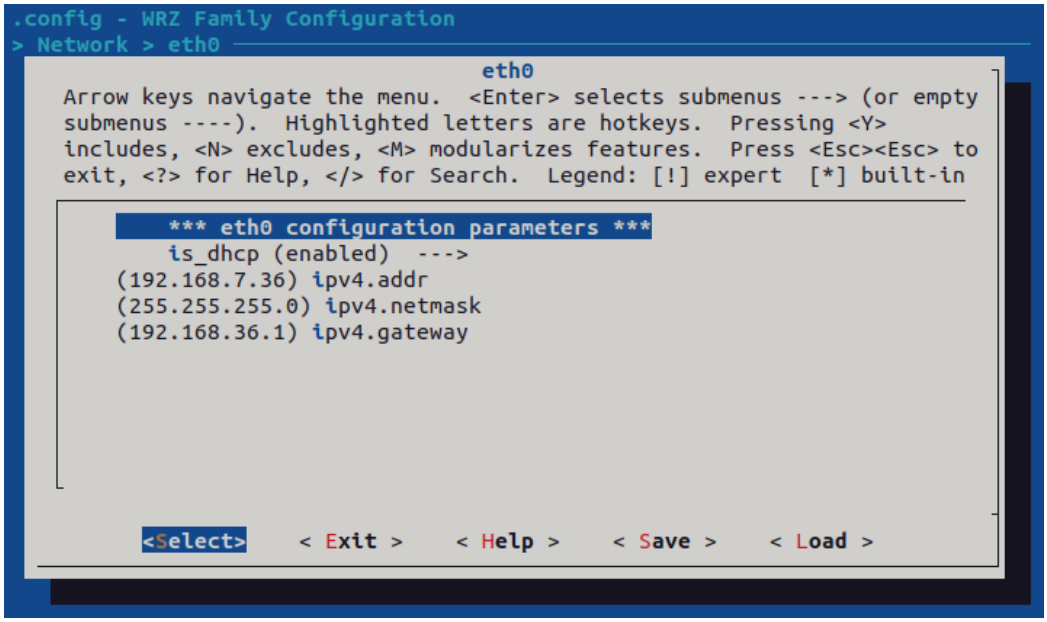


Figure 4-10: wrz_config interface. Interface parameters to change

5. Do not forget to <Save> the changes once the configuration is done. The following message will prompt. To load this configuration at next reboot the default filename (.config) must be used.

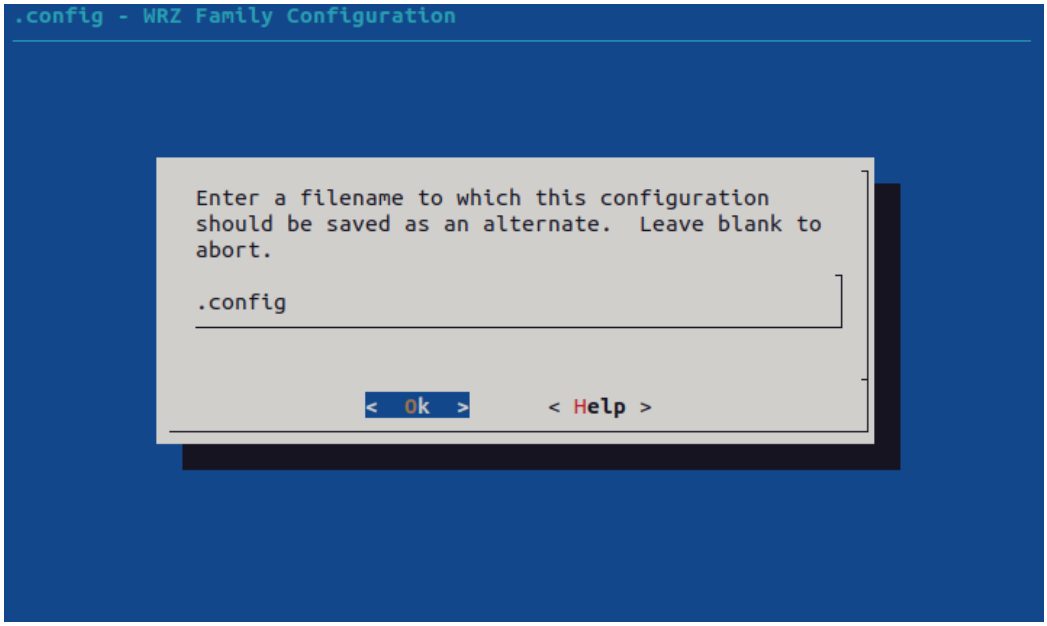


Figure 4-11: wrz_config interface. File in which to save the new applied configuration

6. Select Exit or press <Esc> to return to the command line.

Note:
Verify network configuration after reboot: Network configuration changes are only applied at startup. Thus, it is recommended to reboot the device to verify that the IP settings has been properly updated using the typical ifconfig <ifname> command or through gpa_ctrl tool.

4.4.2 Network Bonding

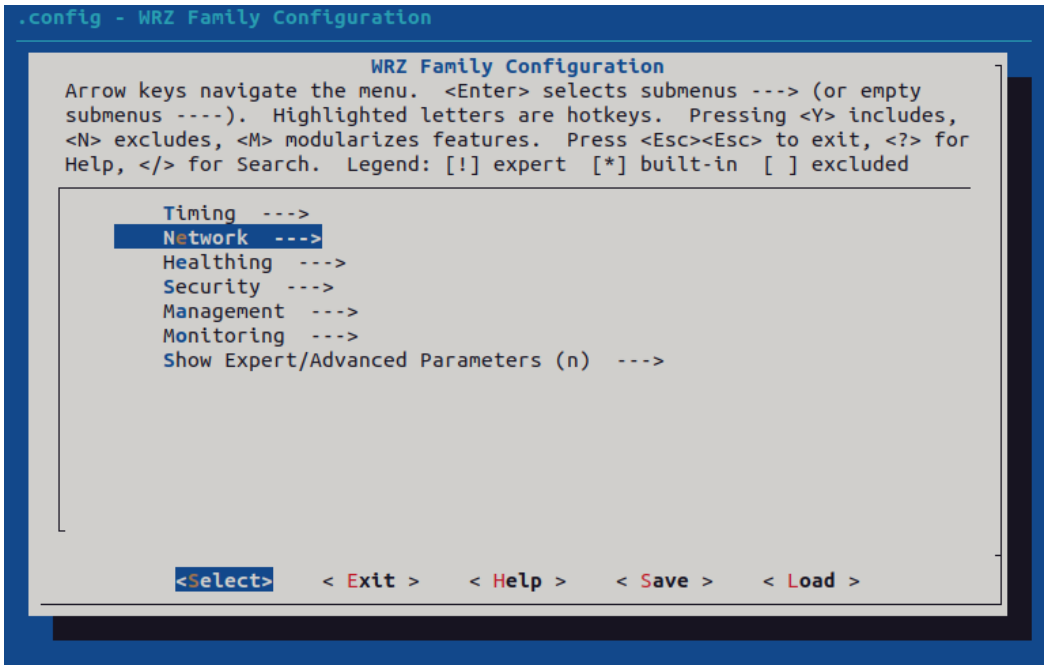
Network bonding is currently only configurable via the CLI and static configuration files. Network interface bonding is available for the eth0 and eth1 Ethernet interfaces. Bonding allows these interfaces to be combined into a single logical interface. The following bonding modes are supported:

- Round Robin (mode 0)
- Active-Backup (mode 1)
- XOR (mode 2)
- LACP (mode 4)
- TLB (Mode 5)
- ALB (Mode 6)

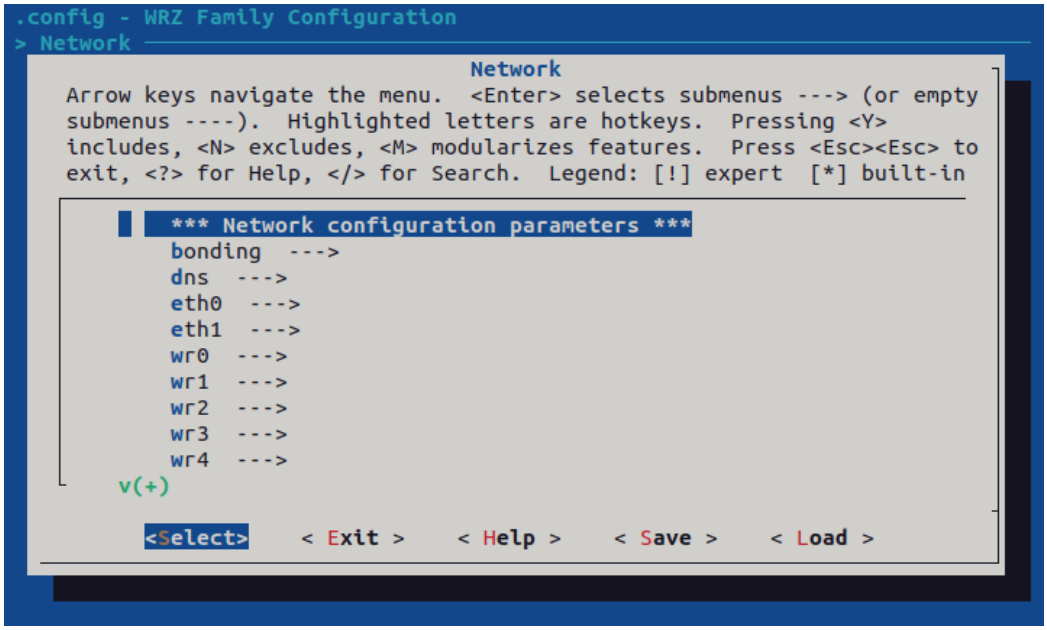
Note: All bonding modes are disabled by default.

To set up network bonding:

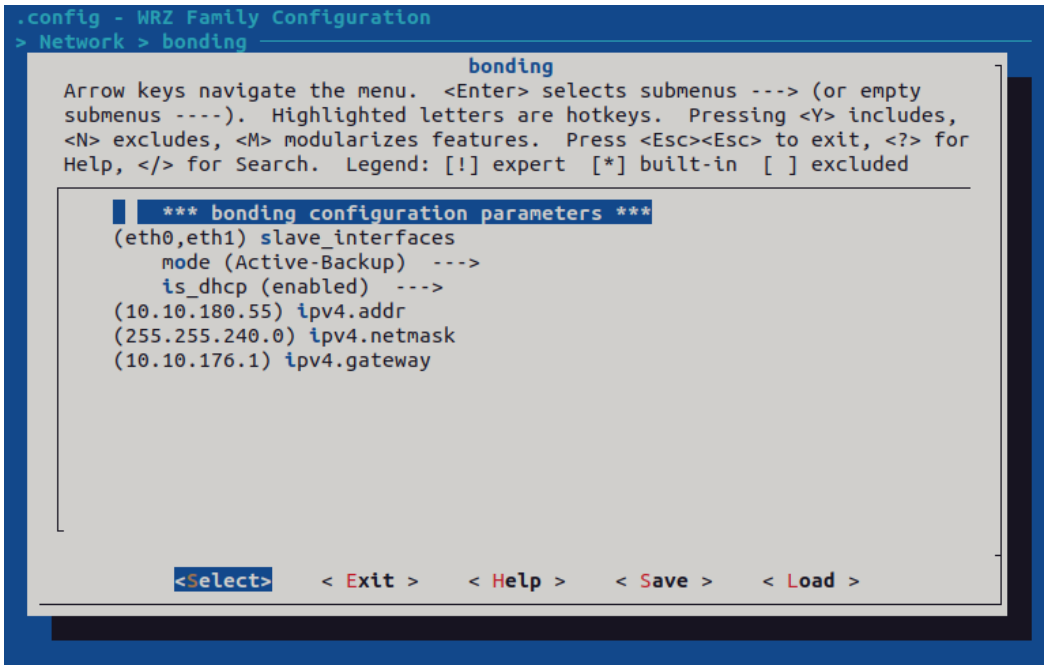
1. The first step is to execute the wrz_config command from a terminal.



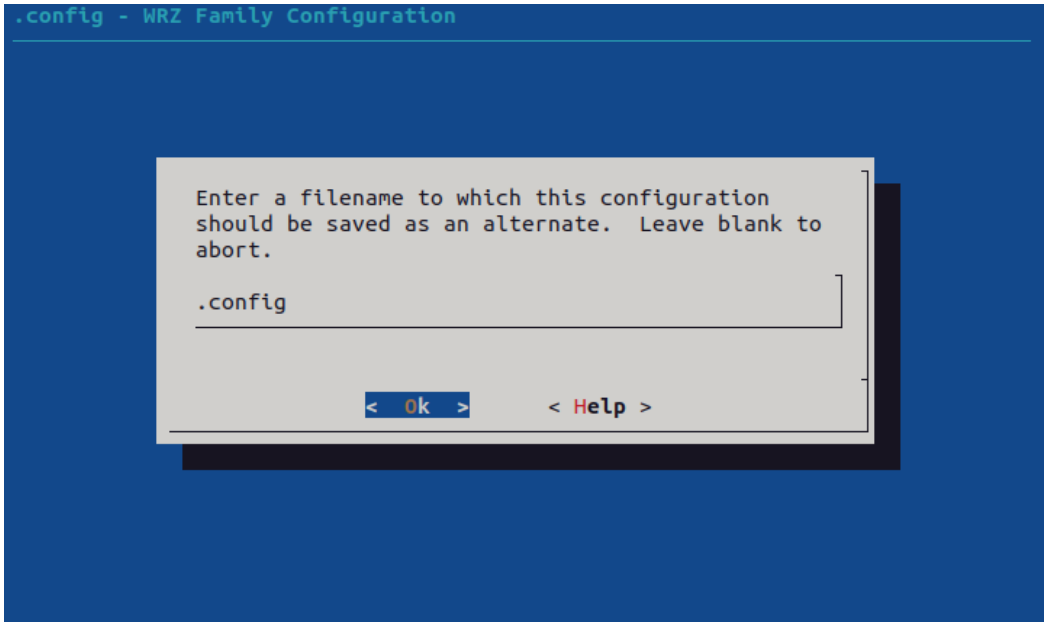
2. Then, the Network section must be selected from the main menu.
3. Select bonding to access bonding configuration.



4. Input the required parameters for bonding.



5. Do not forget to <Save> the changes once the configuration is done. The following message will prompt. To load this configuration at next reboot the default filename (.config) must be used.



The parameters for network bonding can be found in the following table:

OID	Name	Value Type	Description
0.6601.1	Slave interfaces	<String> (i.e., eth0,eth1)	Name of the slave interface to set as a bonding interface separated by commas
0.6601.2	Mode	<Enum> 0. Disabled 1. Round Robin 2. Active-Backup 3. XOR 4. 802.3ad 5. TLB 6. ALB	Bonding mode
0.6601.7	Is DHCP	<Enum> 0. Disabled 1. Enabled	Activate the dynamic IP address resolution on bonding interface using DHCP
0.6601.3	IPv4 address	<String> (i.e., 10.10.180.55)	IP version 4 address of the bonding interface

OID	Name	Value Type	Description
0.6601.4	IPv4 netmask	<String> (i.e., 255.255.240.0)	IP version 4 netmask of the bonding interface
0.6601.5	IPv4 gateway	<String> (i.e., 10.10.176.1)	IP version 4 gateway of the bonding interface

4.5 CLI Monitoring

The `gpa_ctrl` command tool can be used to monitor the current value and state of all the parameters in the WRZ-OS.

4.5.1 Listing parameters

If the user directly executes `gpa_ctrl` without any arguments, it will list all the parameters of the WRZ-OS. Then, by specifying some arguments and options the user can slightly modify its usage.

```
gpa_ctrl [OPTIONS] [<module_name> [<path> [<write_val>]]]
```

The figure below illustrates the usage of `gpa_ctrl` to monitor all parameters corresponding to the power supplies by executing the command: `gpa_ctrl hald pws/`

```
root@z16-006:~# gpa_ctrl hald pws
---              hald ------------------------------------------------------------|
   0.9110.1   pws/pwsl/status                 : OK
   0.9110.2   pws/pwsl/temperature            : 32                C
   0.9110.3   pws/pwsl/v_in                   : 232.000000        V
   0.9110.4   pws/pwsl/v_out                  : 11.949219         V
   0.9110.5   pws/pwsl/power_in               : 27.000000         W
   0.9110.6   pws/pwsl/power_out              : 20.000000         W
   0.9110.7   pws/pwsl/disable_alert          : Yes
C  0.9120.1   pws/pwsr/status                 : NOT DETECTED
   0.9120.2   pws/pwsr/temperature            : 0                 C
   0.9120.3   pws/pwsr/v_in                   : 0.000000          V
   0.9120.4   pws/pwsr/v_out                  : 0.000000          V
   0.9120.5   pws/pwsr/power_in               : 0.000000          W
   0.9120.6   pws/pwsr/power_out              : 0.000000          W
   0.9120.7   pws/pwsr/disable_alert          : No
```

Figure 4-12: Example of gpa_ctrl usage to list power supplies parameters.

The list of related parameters displays in 5 columns:

1. The state of the parameters. In this example only the status of the right power supply (0.9120.1) is in a critical state (C).
2. The OID of the parameter.
3. The path of the parameter inside the module.
4. The value of the parameter.
5. The unit of the parameter (if relevant).

The user can also list only the parameters related to the right power supply by executing:

```
gpa_ctrl hald pws/pwsr/
```

4.5.1.1 Readback a specific parameter

It might also be interesting to only readback a specific parameter. To only get the status of the left and right power supplies the user should execute:

```
root@z16-006:~# gpa_ctrl hald pws/pwsl/status; echo $?
OK
0
root@z16-006:~# gpa_ctrl hald pws/pwsr/status; echo $?
```

Warning: C pws/pwsr/status

```
NOT DETECTED
204
```

Note:
Return code and stdout/stderr: The gpa_ctrl print parameters to stdout/stderr according to their status such that an advanced user can easily filter them. It also returns specific error code depending on the status. For more information please read carefully gpa_ctrl –h.

4.5.2 Applying changes online

If a parameter is writeable, this means that it can be directly applied by using the following syntax:

```
gpa_ctrl <module_name> <param_path> <new_value>
```

For example, to disable the alert for the right power supply the user must execute:

```
gpa_ctrl hald pws/pwsr/disable_alert Yes
```

If the command returns without any errors, this mean that the changes have been properly applied. This can be checked by reading back the output of

`gpa_ctrl hald pws/pwsr/`

4.5.3 Other functionalities

To improve legibility, the parameters can be displayed in a tree view by adding the `-t` flag:

```
root@z16-006:~# gpa_ctrl -t hald pws
0             hald      LEGEND: dir writable readable expert
0.9100        pws
0.9110        ├── pwsl
0.9110.1      │   ├── status (OK)
0.9110.2      │   ├── temperature (40C)
0.9110.3      │   ├── v_in (235.500000V)
0.9110.4      │   ├── v_out (11.968750V)
0.9110.5      │   ├── power_in (29.000000W)
0.9110.6      │   ├── power_out (23.000000W)
0.9110.7      │   └── disable_alert (Yes)
0.9120        └── pwsr
0.9120.1          ├── status (NOT DETECTED)
0.9120.2          ├── temperature (0C)
0.9120.3          ├── v_in (0.000000V)
0.9120.4          ├── v_out (0.000000V)
0.9120.5          ├── power_in (0.000000W)
0.9120.6          ├── power_out (0.000000W)
0.9120.7          └── disable_alert (Yes)
```

The user can get more information about the parameters by using the verbose flag -v:

```
root@z16-006:~# gpa_ctrl -v hald pws/pwsr
---             hald ------------------------------------------------------------|
---
--- desc: The Hardware Abstraction Layer Daemon (HALD)
--- status: Running (0)
--- nparams: Warning:0 , Critical:0 , Out-of-sync:0
---
   0.9120.1   pws/pwsr/status                    : NOT DETECTED            range:[0,65535]
              └─ "Global Status. 0:ok; otherwise:error."
   0.9120.2   pws/pwsr/temperature               : 0                   C
              └─ "Temperature of PWS in ºC"
   0.9120.3   pws/pwsr/v_in                      : 0.000000            V
              └─ "Power Supply: Volts IN"
   0.9120.4   pws/pwsr/v_out                     : 0.000000            V
              └─ "Power Supply: Volts Out"
   0.9120.5   pws/pwsr/power_in                  : 0.000000            W
              └─ "Power consumed from Line in Watts"
   0.9120.6   pws/pwsr/power_out                 : 0.000000            W
              └─ "Power given from Power Supply"
   0.9120.7   pws/pwsr/disable_alert             : Yes                     range:[0,1]
              └─ "Enable/Disable the critical alert when the power supply is not plugged"
```

Or specifically list of the corresponding enum values using the `-i e` Option

```
root@z16-006:~# gpa_ctrl -i e hald pws/pwsr/status
{0:'OK', 1:'NOT DETECTED', 2:'POWER OFF', 4:'TEMP PROBLEM', 8:'IN UNDERVOLT', 16:'OUT
OVERCURR', 32:'OUT OVERVOLT', 64:'CML ERROR', 128:'DEVICE BUSY', 256
:'UNKNOWN', 512:'OTHER ERROR', 1024:'FAN PROBLEM', 2048:'POWER NOT GOOD', 4096:'MFR SPECIFIC',
8192:'VIN PROBLEM', 16384:'OUT PROBLEM', 32768:'VOUT PROBLE
M'}
```

You can also display the expert parameters by adding the -a flag.

4.6 Other CLI tools

This section enumerates some other tools that are referenced across the user-guide in order to manage the device from the console.

- `wrz_version`: Legacy tool to get information about version of firmware and hardware.
- `wrz_flashfw`: Tool used to flash an uploaded firmware. (See "Firmware Update " on page 211)
- `wrz_logdump`: Tool used to report an error log for the support team. (See " How to report an error" on page 230)

More information about each tool can be found in their respective section or simply by adding the `-h` flag to output the help message embedded in the executable.

CHAPTER 5

Timing

The following topics are included in this Chapter:

5.1 Multi-sources & Resiliency

To ensure continued operation over possible failures, the WR-Z16 incorporates an innovative system that handles multiple timing sources. It also synthesizes these timing sources into a simplified state (a.k.a Virtual Clock State) to ease the monitoring of the device and distributes a common timing information to the down layers.

5.1.1 Timing Sources

The WRZ-OS can handle multiple timing sources in order to discipline the local oscillator of the device. These timing sources can be of different types:

- External Reference (Front panel connectors)
- White Rabbit (High-Accuracy PTP)
- NTP (Survey mode only)
- Holdover (Always used as last timing source if available)

Note: PTP as timing source: A pure PTP timing source (slave) should not be selected if the timing is then re-distributed using WR (master). Indeed, the jittered correction run by PTP clock is not compatible with the precision needed for WR/HA distribution.

Note: NTP Timing source (Survey mode): Due to its poor accuracy, NTP protocol is always in Survey Mode and thus cannot actively discipline the local clock.

Then, a maximum total of 5 timing sources of the same or different types can be handled. " FOCA: The Failover Clock Algorithm" below details the common parameters shared by all the timing sources and how they are used to determine their states.

5.1.2 FOCA: The Failover Clock Algorithm

The FOCA has been designed for the purpose of automatically switching from one timing source to another by applying the following policy:

In case of failure of the active timing source, switch to the next ready timing source.

This algorithm is based on the “Best Master Clock Algorithm (BMCA)” detailed in the PTP IEEE 1588-2019 standard but acts only in case of failure and not when the “best” source appears in the network. It also enforces the evaluation of the timing sources in a rank order configured by the user. FOCA algorithm has been designed to provide a “safer” approach than BMCA or even ABMCA (Alternate BMCA) to handle switching between multi-references. Its main characteristics are:

- Provides a deterministic behavior.
- Does not allow a new (rogue) node to become the active reference.
- Recovers back to normal state must be done under the supervision of an operator.
- Allows switching between cross WR/PTP profiles and multiple external timing sources.
- Has been designed with tree network topology in mind and it is not optimized for ring topology.

The following figure depicts a configuration where the first two timing sources are employing WR protocol, followed by an external GNSS receiver connected to the front panel reference (GM) and finally ending with the holdover to slowly drift until corrective maintenance. It also illustrates how the two strategies of the FOCA algorithm behave.

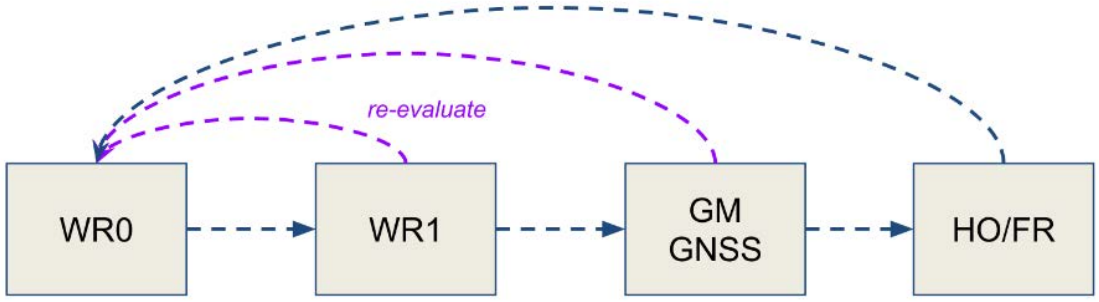


Figure 5-1: Multi-timing sources handle by FOCA policy with its two strategies: only fall-down (blue) & re-evaluation (purple)

An example of the behavior is given by the scenario illustrated in the next image where the following events are shown:

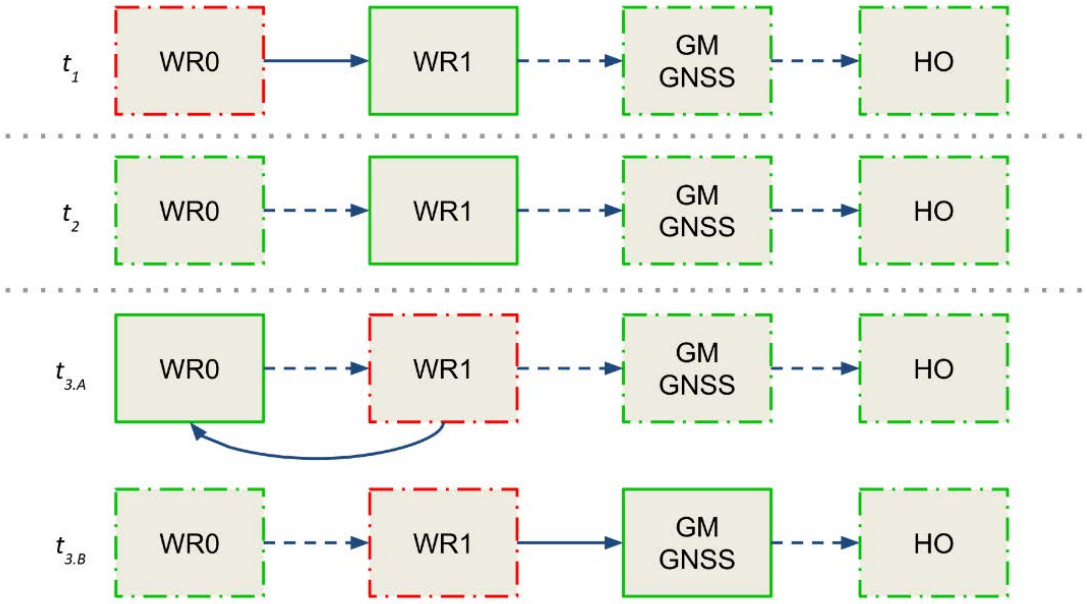


Figure 5-2: FOCA algorithm under scenario 1

- In t_1, the active reference (solid green line) is WR1 because the primary reference has reached a CRITICAL state (dashed red line).
- In t_2, the primary reference WR0 becomes available again (dashed green line) but the device keeps using WR1 as the active reference as no failure has been detected on this timing source.
- In t_3, an error is detected on WR1 and the FOCA algorithm will act differently according to the configuration of its strategy.
    A. If the strategy is to re-evaluate all timing sources when a failure occurs, and the primary reference is eligible, the WR0 will be selected as the active reference.
    B. If the strategy is to only fall-down, the FOCA algorithm will select the next available timing source in the list and will thus lock on the external GNSS reference. With this strategy the only way to use back WR0 as the active reference is to restart the devices' synchronization daemon (/etc/init.d/ppsi restart) or to reach the last timing source and wait for a critical error.

Another key aspect of FOCA is how to determine when there is a "failure" on a timing source. Some cases are obvious such as the link is down, no packets are exchange but other cases can be more complex to identify: all these cases are detailed in the appendix VCS code tables ("Grand Master (GM VCS Code)" on page 234).

For a deeper understanding of the behavior of the FOCA algorithm it is recommend reading the section "Others" on page 243 in the appendix where more scenarios are detailed.

> **Note:** FOCA is based on BMCA, thus it is compatible with all the clock quality and timing information fields. In other words, this means that a device running FOCA strategy can provide timing to a BMCA device and BMCA information is provided to FOCA algorithm.

5.1.3 Virtual Clock Overview

The concept of "Virtual Clock" has been introduced in the new version of WRZ-OS to aid monitoring of the global timing status of the device. It allows to abstract the way the timing sources discipline the local oscillator and summarizes how the device will announce its own clock information through the outputs.

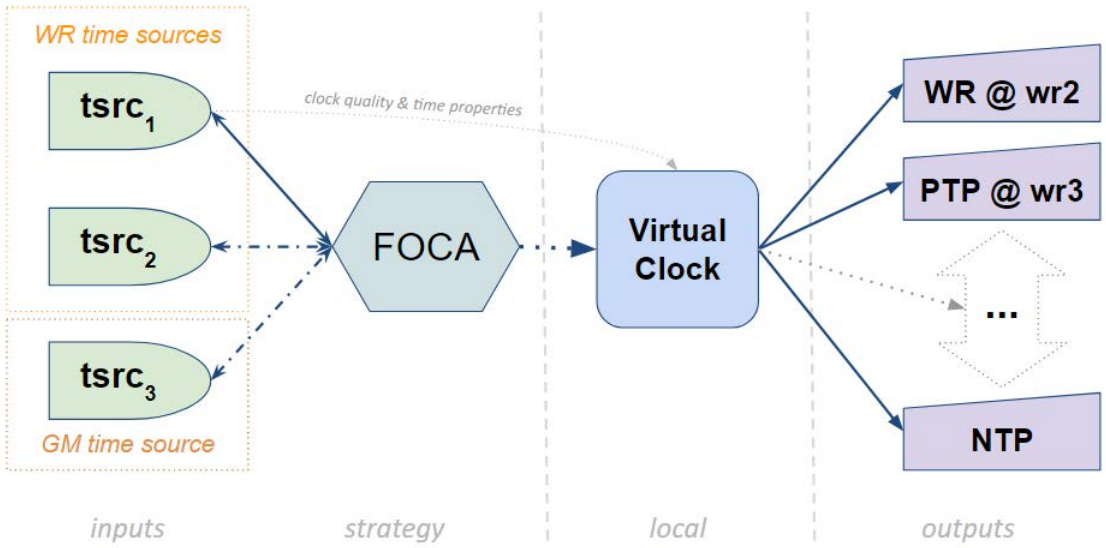


Figure 5-3: Data-flow between timing sources, virtual clock and outputs

When using the FOCA policy (see "Data-flow between timing sources, virtual clock and outputs" above), the virtual clock will be fed by the active timing source (e.g., $tsrc_1$), then this information (clock quality & time properties) will be forwarded by all the outputs:

- directly in case of PTP/WR protocol.
- by properly modifying the corresponding fields in the case of NTP, NMEA, etc.

The following figure displays the overview panel of the virtual clock information when the device is using an external reference from front-panel (GM) as the active source.

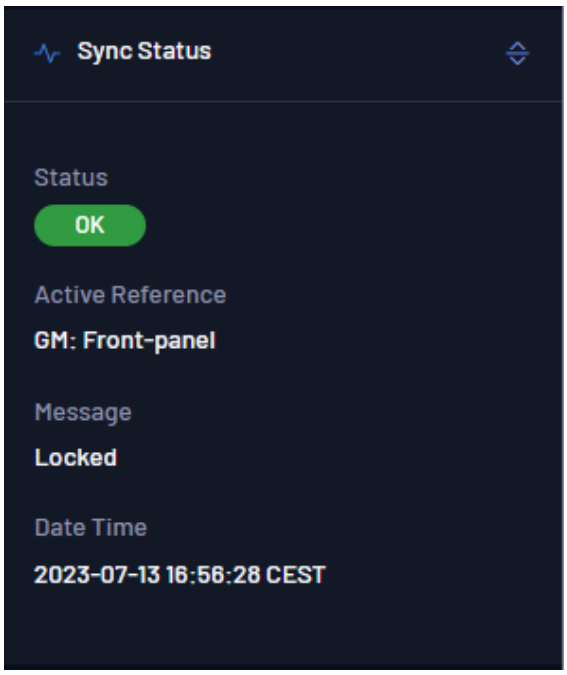


Figure 5-4: Virtual Clock Overview (Dashboard Web GUI View)

If the user wants more advanced information, the **Overview** > **General** > **Timing Sources** panel has an Advanced drop down option for each timing source (see below).

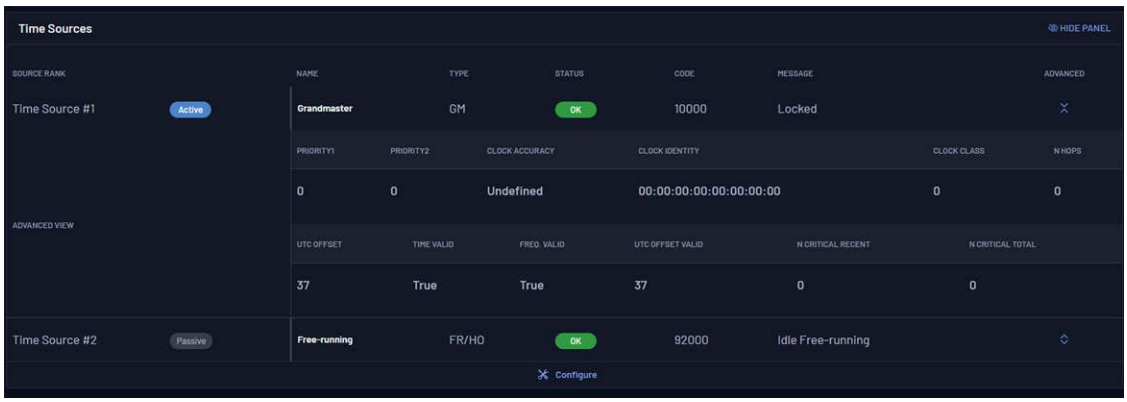


Figure 5-5: Full Timing Sources Overview

In this expanded view, all parameters related to the incoming timing source are displayed, including the **Code** field, which represents the Virtual Clock Status Code and provides a precise but simple way to identify the current timing status

of the device. The complete table with all VCS codes is detailed in the Appendix, under "VCS Code" on page 233.

5.1.4 Survey Mode

The survey mode provides the system with the capability to evaluate the synchronization performance of different time references even though they are not the active reference. It enables the possibility to compare different non-active inputs (White Rabbit/PTP/1PPS/10MHz) with the current active reference or the local oscillator when no other active references are available.

The survey mode configures the interfaces as if they were active timing sources, except that the computed error (time difference) is not applied to the system and thus, to the local oscillator.

The user can configure this mode in order to compare timing sources, using another active timing source as ground truth. Thus, the offsets, delays and time-specific parameters of the survey mode are computed regarding this reference.

The survey mode can be configured via web UI and CLI configuration.

5.1.4.1 White Rabbit (WR) survey mode

The WR survey mode allows a WR interface to measure all parameters computed in a standard WR active port mode, but without syncing the internal system clock. The interface will try to lock to a master WR reference, and all the parameters will be computed using the system clock as time base.

Survey Overview

The web user interface allows the user to have a wide view of all the parameters regarding the current status of each White Rabbit/Grand Master interface by consulting the section Overview -> White Rabbit. A prior survey configuration needs to be made in order to display these tables. To conduct a survey to obtain parameters:

1. The device should be set in custom mode. We can activate such a mode by accessing Configuration -> Timing General. Under the preset tab, select Custom preset and click Apply.
2. Under Fanout Configuration, select Fanout Source #wr0 as MASTER and Fanout Source #wr1 as SURVEY as seen in the following image:

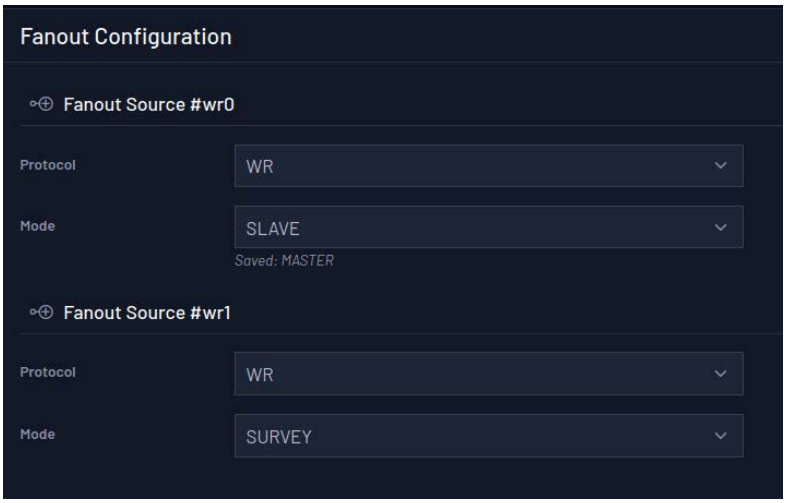


Figure 5-6: Fanout source configuration

3. Under Time Sources Configuration, set the Time Source #1 as WR/WR0:

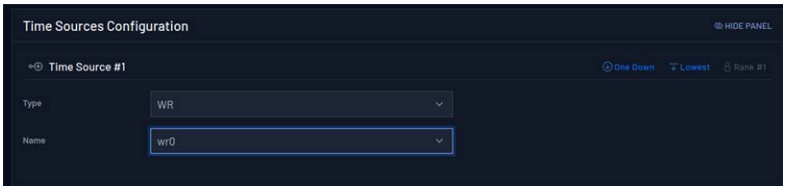


Figure 5-7: Time source configuration

4. Under External Reference (GM), make sure GM mode is set to survey, and click Apply to update the changes:

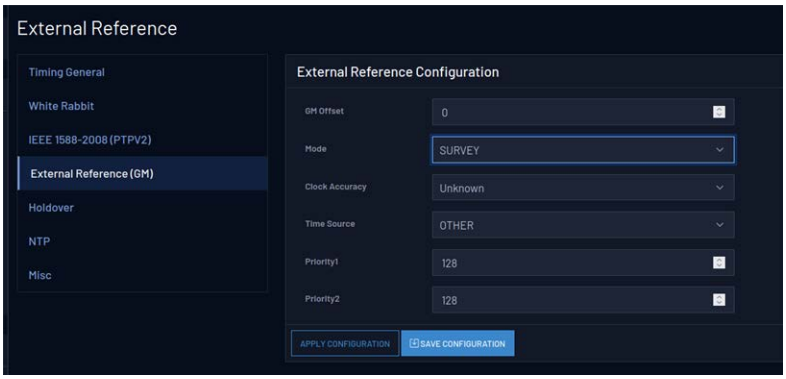


Figure 5-8: External reference configuration

5. When locked, we can access to the menu Overview -> White Rabbit or Overview -> External Reference (GM) to check the parameters regarding White Rabbit interfaces and GM respectively. By clicking the advanced option on the right side of the table we can access to the full list of parameters reported by each interface.

   The Active Servo table reports parameters regarding the slave interface:

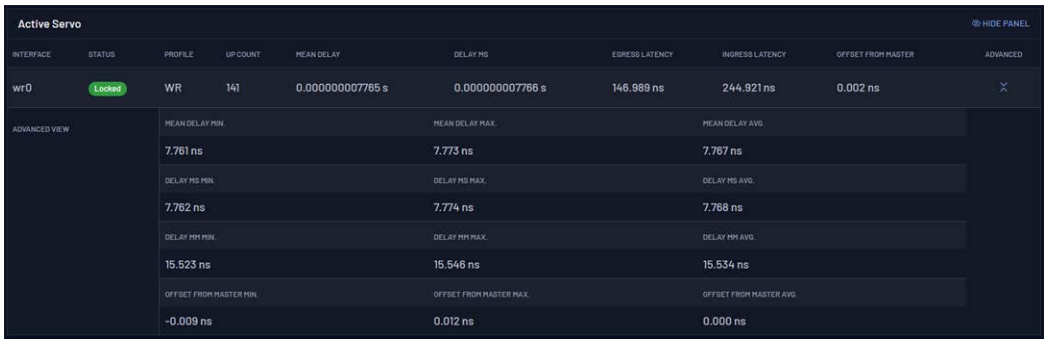


Figure 5-9: Active servo table

The Survey Servos table reports parameters regarding all interfaces that are not acting as a reference and compares those metrics with the active reference:

Figure 5-10: Survey servos table

And the External Reference survey table reports data regarding the GM:

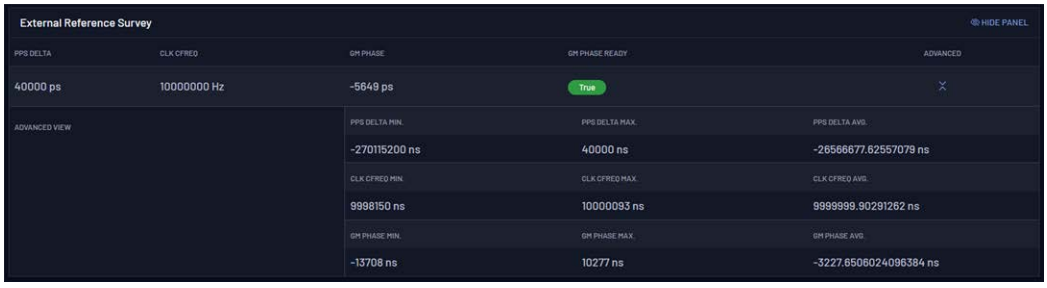


Figure 5-11: External reference survey table

WR Survey Mode Configuration

The configuration of the WR survey mode can be accessed by using the Web UI and the CLI configuration.

WR Survey Mode via the Web UI

1. Login into the device dashboard (see "The Web GUI" on page 42).
2. Navigate to Configuration->Timing General
3. Switch to the Custom profile and configure the timing sources you need in “Time Sources Configuration” section.
4. In “Fanout Configuration” section, configure the interfaces you want in survey mode by selecting “SURVEY” in “Mode” selection box of each interface:

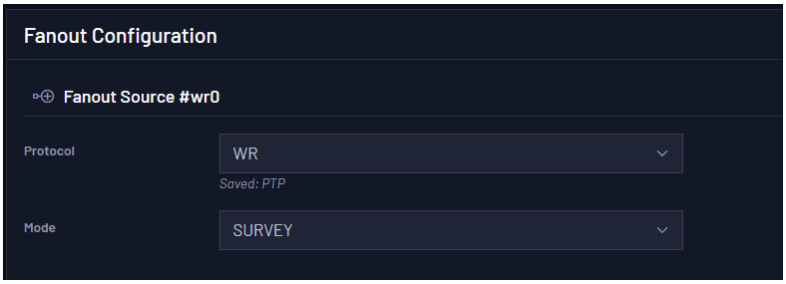


Figure 5-12: Fanout survey configuration

5. Ensure that the “Protocol” selection box marks “WR” in each WR survey interface.

WR Survey Mode via the CLI

1. Open the CLI configuration menu (as explained in "CLI Configuration" on page 47)
2. Navigate to “> Timing > Ports Configuration > “
3. For each WR interface to be configured in survey mode, enter to their configuration one by one. Select “WR” in “proto” section, and “SURVEY” in “mode” section:

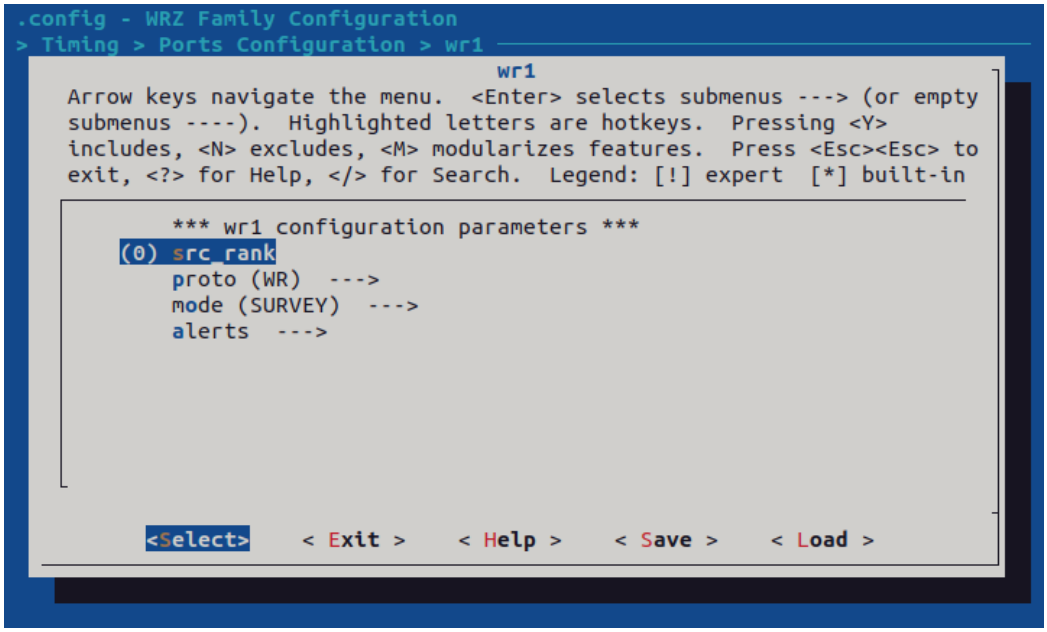


Figure 5-13: CLI survey mode configuration

4. Ensure that “src_rank” keeps in the value “0”.

Displaying WR Survey Parameters

Once one or more WR interfaces are configured in survey mode, the user can display the computed timing parameters as if they were an active nominal time source. To view these parameters in the device CLI, use the command:

```
gpa_ctrl ppsi net/wrX -A
```

where `wrX` is the desired survey interface. Once executed, all interfaces parameters are shown.

Some parameters of interest could include:

- `net/wrX/1/servo/mean_delay`: Half of the cable round trip time excluding fixed+semistatic (cRTT/2).
- `net/wrX/1/servo/delay_MS`: Delay between Master and Slave
- `net/wrX/1/servo/delay_MM`: The measured round trip time, including fixed+semistatic delays (Legacy WR:'mu')
- `net/wrX/1/servo/offset_from_master`: The time error between a Slave Clock and a Master Clock (Legacy WR: clock offset)

5.1.4.2 PTP Survey Mode

The PTP survey mode allows a PTP interface to measure all parameters computed in a PTP active port mode, but without syncing the internal system clock. The interface will try to lock to a master PTP reference, and all the parameters will be computed using the system clock as time base.

Survey Overview

The web user interface allows the user to have a wide view of all the parameters regarding the current status of each PTP interface by consulting the section Overview -> IEEE 1588-2008 (PTPV2). A prior survey configuration needs to be made in order to display these tables. To conduct a survey to obtain parameters:

1. The device should be set in custom mode. To activate, access Configuration -> Timing General. Under the preset tab, select Custom preset and click Apply.
2. Under Fanout Configuration, select Fanout Source #wr0 as MASTER and Fanout Source #wr1 as SURVEY as seen in the following image:

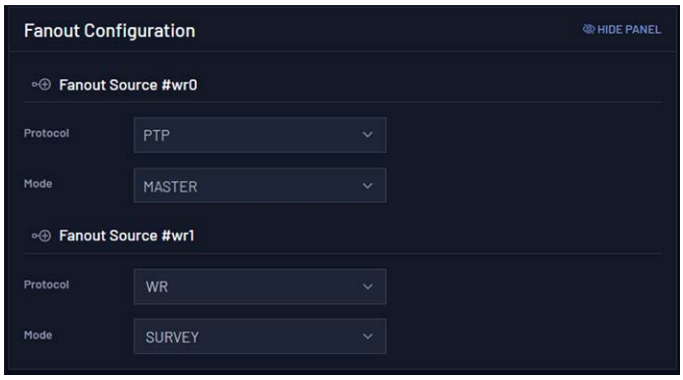


Figure 5-14: PTP survey mode fanout configuration

3. Under Time Sources Configuration, set the Time Source #1 as PTP/WR0:

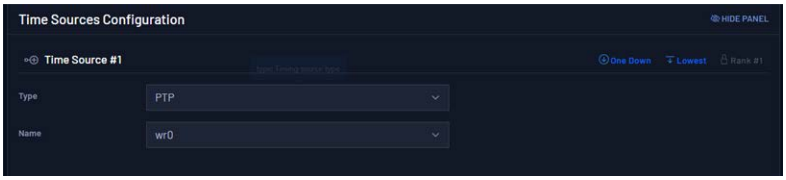


Figure 5-15: PTP survey mode time sources configuration

4. Under External Reference (GM), make sure GM mode is set to survey, and click Apply to update the changes:

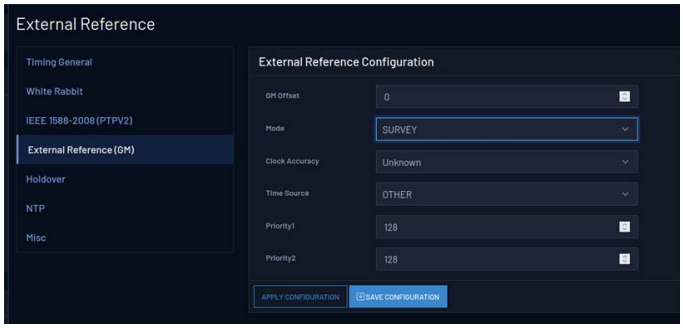


Figure 5-16: PTP survey GM settings

5. When locked, access the menu Overview -> White Rabbit or Overview -> External Reference (GM) to check the parameters regarding White Rabbit interfaces and GM respectively. Select the advanced option on the right side of the table to access the full list of parameters reported by each interface.

   The Active Servo table reports parameters regarding the slave interface:

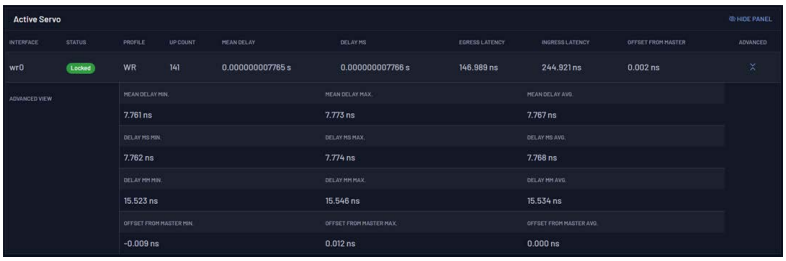


Figure 5-17: Active servo table

The Survey Servos table reports parameters regarding all interfaces that are not acting as a reference and compares those metrics with the active reference:

Figure 5-18: Survey servos table

And the External Reference survey table reports data regarding the GM:

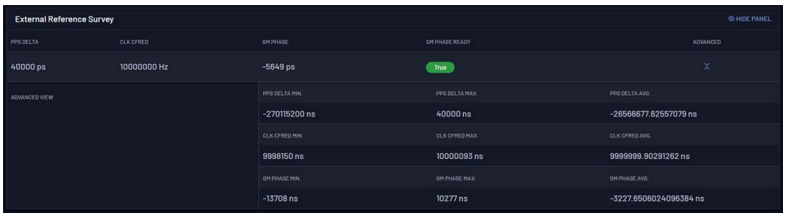


Figure 5-19: External reference survey table

PTP Survey Mode Configuration

The configuration of the PTP survey mode is accessed via the Web UI or the CLI.

PTP Survey Mode via the Web UI

1. Login into the device dashboard (see "The Web GUI" on page 42).
2. Navigate to Configuration->Timing General
3. Switch to the Custom profile and configure the timing sources you need in “Time Sources Configuration” section.
4. In “Fanout Configuration” section, configure the interfaces you want in survey mode by selecting “SURVEY” in “Mode” selection box of each interface:

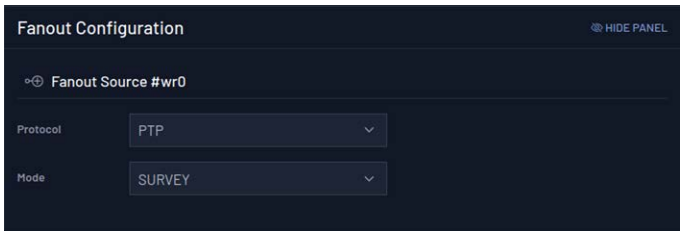


Figure 5-20: PTP survey mode fanout configuration

5. Ensure that the “Protocol” selection box marks “PTP” in each WR survey interface.

PTP Survey Mode via the CLI

1. Open the CLI configuration menu (as explained in "CLI Configuration" on page 47)
2. Navigate to “> Timing > Ports Configuration > “
3. For each PTP interface to be configured in survey mode, enter to their configuration one by one. Select “PTP” in “proto” section, and “SURVEY” in “mode” section

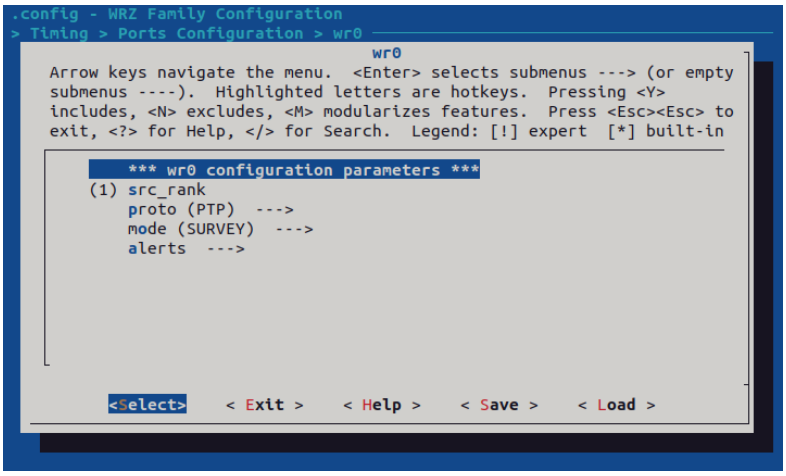


Figure 5-21: PTP survey mode CLI configuration

4. Ensure that “src_rank” keeps in the value “0”.

Displaying PTP Survey Parameters

Once one or more PTP interfaces are configured in survey mode, you can display the computed timing parameters as if they were an active nominal time source. As an example for the wr0 port, to view its metric parameters in the device CLI, use the command:

```
gpa_ctrl ppsi net/wr0 -A
```

Once executed, all interfaces parameters are shown. Some parameters of interest could include:

- `gpa_ctrl wptpd net/wr0/1/info/offset_from_master`: The calculated time error between a Slave Clock and a Master Clock.
- `gpa_ctrl wptpd net/wr0/1/info/one_way_delay`: Delay between Master and Slave.
- `gpa_ctrl wptpd net/wr0/1/info/mean_delay`: Equivalent to the Round Trip Time (RTT), it is delay between Master and Slave plus the the delay between Slave to Master.

5.1.4.3 External Reference (GM) survey mode

The Survey Mode for the external reference, named as Grand Master Survey Mode, is useful to compare an active external reference with a current PTP, NTP, or White Rabbit synchronization.

The web user interface allows the user to have a wide view of all the parameters regarding the current status of each PTP interface by consulting the section

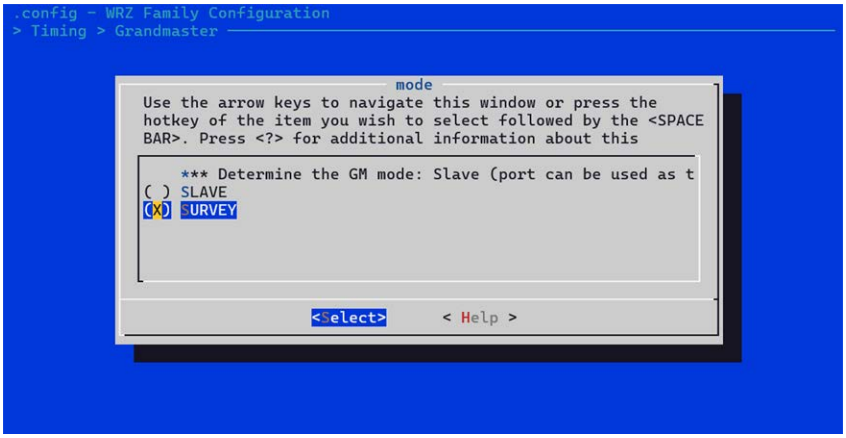
Overview -> **External Reference** (GM). A prior survey configuration needs to be made in order to display these tables. Configuration of GM Survey mode is done via the CLI or Web GUI.

GM Survey Mode via the CLI

To enable GM survey mode:

1. Open the CLI configuration menu (as explained in "CLI Configuration" on page 47)
2. Navigate to Timing > Grandmaster > mode
3. Chose the mode Survey.
4. Reboot the device.

Tip: The device must be set to Boundary Clock mode to use the Survey Mode with an External Reference.



GM Survey Mode via the Web GUI

PPS delta

This allows to measure the time difference between an external PPS signal and the active White Rabbit Reference. When the device is synchronized and has an active PPS reference in frontal panel, the device can measure the actual difference (the delta) between the internal synchronized and external signal.

Running the `gpa_ctrl` command: `gpa_ctrl hald spll/ext/fpanel/pps_delta` will show the such difference, in ps. Here there is an example:

```
root@z16-014:~# gpa_ctrl hald spll/ext/fpanel
---             hald ------------------------------------------
--------------|
   0.8410.1   spll/ext/fpanel/sig_detected         : PPS & CLK
   0.8410.4   spll/ext/fpanel/pps_delta            : -
48000               ps
   0.8410.5   spll/ext/fpanel/clk_cfreq            :
10000000             Hz
```

- `sig_detected`: This parameter reports whether both PPS and CLK 10 MHz is connected in the frontal panel.
- `pps_delta`: This parameter report the delta between PPS signals mentioned before.
- `clk_cfreq`: This parameter reports the measure of frequency of the 10 MHz input.

A PPS signal in the input (which has been detected by the sig_detected parameter) is mandatory to have a valid `pps_delta`.

10 MHz reference

The `gm_phase` reports the phase difference between the external 10 MHz signal and the internal clock, with a resolution of up to 16 picoseconds.

To enable this feature, is mandatory that the device is configured as a Boundary Clock (BC) and has an active 10 MHz input reference. Once the device is locked, the `survey_mode` for this signal have to be enabled.

Once all the requirements are met, the phase can be read using the command `gpa_ctrl hald spll/survey`. For example:

```
root@z16-014:~# gpa_ctrl hald spll/survey
---             hald ------------------------------------------
--------------|
   0.8500.1   spll/survey/gm_phase                 :
10913               ps
   0.8500.2   spll/survey/gm_phase_ready           : 1
```

- `gm_phase`: Phase difference between the external 10 MHz reference and the internal clock.
- `gm_phase_ready:` Defines if the `gm_phase` value is ready to be read.

5.1.5 WR Seamless Failover

The Seamless Failover mechanism provides switching between WR timing sources without loss of the sub-nanosecond synchronization, when there are other available and ready WR references and the offset is below a predefined threshold of 500 ns. This can be managed when:

- The next timing source uses the WR protocol and is ready and available.
- The configured policy determines that a change of timing source is required; for example, when:
  - The actual timing source loses its reference
  - The master timing source is degraded
  - A user issues a timing source switchover

The current **supported triggers** for seamless failover transition performance are the following:

- **Link down events of active WR timing sources:** a fast trigger which occurs when the communication link between the failover node and the WR master device is lost.
- **Manual switchover:** a fast trigger that can by triggered by the user in the timing sources menu.
- **Reception of a degraded clock class:** a slow trigger which occurs when a degraded clock class is received from the WR master device. In order to control the synchronization error during the transition, it is necessary to include the optional holdover module in the master device. The holdover will ensure minimal drift until the switchover is activated (the configuration of a short expiration time (<15 minutes) is recommended to enable smooth transition with optimal performance).
- **Pre-delock trigger**: If enabled, the switchover will be a smooth transition if the offset is below 500 ns, and an abrupt transition if the offset is above 500 ns. With this early detection mechanism, the Boundary Clock can anticipate when the active Grandmaster will lose lock and switch to the backup reference in advance. This minimizes error accumulation and results in an even more accurate and transparent transition.

5.1.5.1 Seamless Failover configuration via the Web UI

1. Log in into the device dashboard (see "The Web GUI" on page 42).
2. Navigate to **Configuration** > **Failover**.

3. Seamless Failover can be activated via the "Seamless Failover Transition" checkbox.

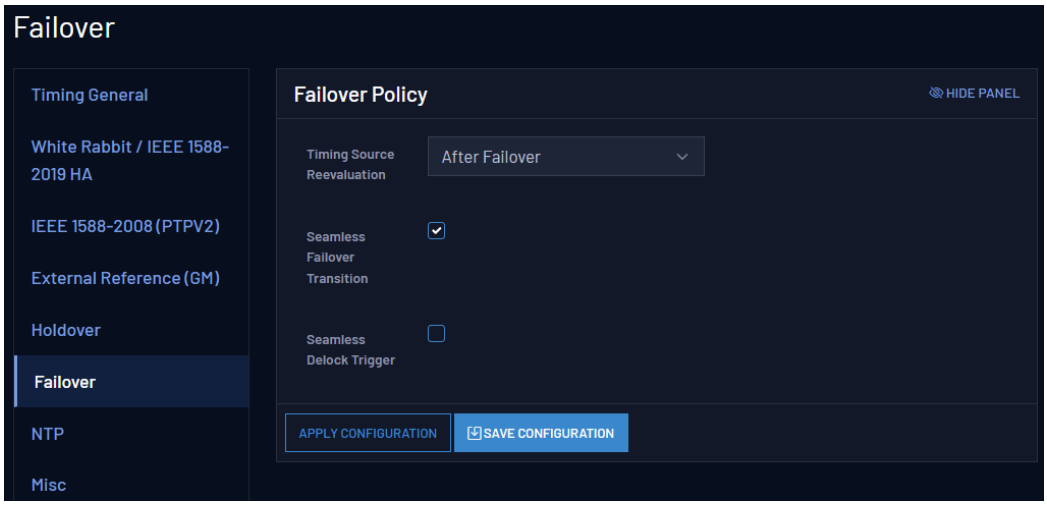


Figure 5-22: Failover Policy configuration

4. The Seamless Delock can be enabled via the "Seamless Delock Trigger'"checkbox

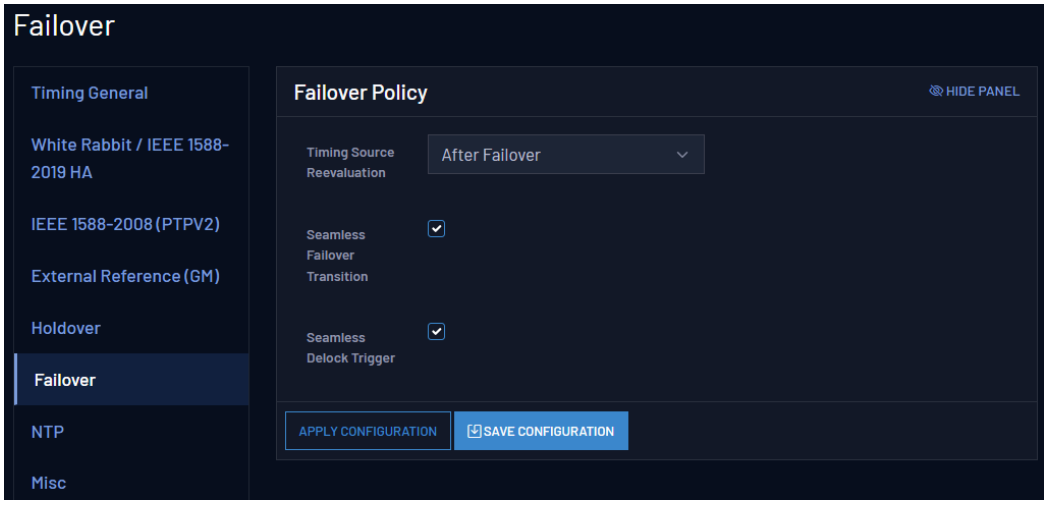


Figure 5-23: Delock Trigger configuration

In practice, both Seamless Failover and Seamless Delock Trigger are complementary. Seamless Failover defines how the transition is performed (smooth instead of abrupt), while Seamless Delock Trigger defines when the transition is initiated (early enough to avoid phase drift). For this reason, best results are obtained when both options are enabled.

5.1.5.2 Timing Source Reevaluation

As seen in " FOCA: The Failover Clock Algorithm" on page 60, the device offers several options regarding the re-evaluation of new timing sources. This behavior can be modified with the "Timing Source Reevaluation" option in the "Failover Policy" section in **Configuration** > **Failover**. The options are as follows:

- **After Failover**: Only when the active timing source fails, the device will switch to the first available timing source in the priority order. All timing sources are considered when selecting a new timing source.
- **When Available**: Whenever a timing source with higher priority than the active one becomes ready, the device will automatically select and change to that timing source. This change happens even when the active source has not failed.
- **No Reevaluation**: Only when the active timing source fails, the device will switch to the next available timing source in the priority order. Timing sources with higher priority than the active time source will not be considered when selecting a new timing source.

5.1.5.3 Manual Switchover

As described in the beginning of the section, there is an option to force the device to switch between timing sources. On the Web GUI, navigate to **Overview** > **Timing** > **General** > **Time Sources** and select the "Force Switchover" button, as shown:

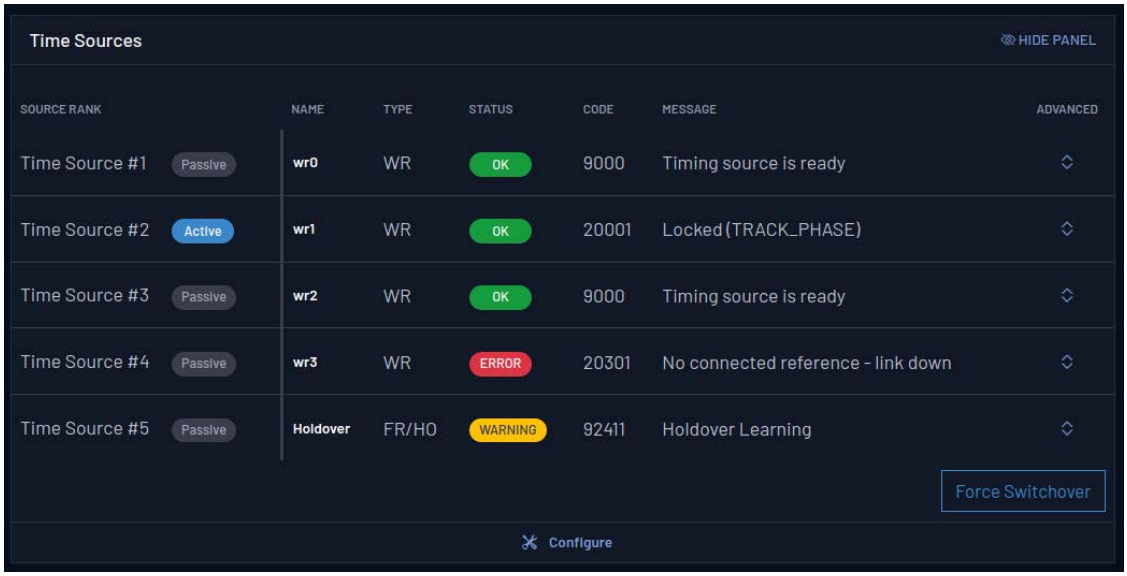


Figure 5-24: Force Switchover Button in Time Sources list

This forced switchover will be seamless (i.e. the synchronization will not be lost) if the "Seamless Failover Transition" option described above is active. If the switchover is forced without activating this option, the synchronization will be interrupted when switching timing sources.

Note: The forced switchover follows the re-evaluation behavior defined in the **Timing Source Reevaluation** option, choosing the next timing source accordingly. Forced switchover is disabled while the **When Available** re-evaluation option is selected.

5.2 General Timing Management

To ease the configuration of the device the WRZ-OS implements presets to allow a quick setup of the timing sources and of the master ports to redistribute time.

Note: If the device has been shipped with the holdover option, this timing source will be, by default, configured as the last timing source independently of the preset.

Note: PTP Master configuration: Presets only configure the role and protocol (PTP or WR) used for all network interfaces with some default settings. A specific configuration of PTP (e.g., Profile, packet rates, etc.) can then be performed under the PTPv2 configuration tab if a valid license has been detected.

5.2.1 Presets

Preset configuration is found in the Web GUI under **Configuration** > **Timing General** > **Preset**.

5.2.1.1 WR Slave @ wr0 (BC) [default]

Web GUI selection: **BC: wr0 | WR**

- The primary timing source is provided using WR protocol through interface wr0.
- The other (wr1) is configured as WR master.

This is the default preset as it is the standard/legacy configuration of most of the WR devices. This is the simplest Boundary Clock behavior where the device is disciplined by a single reference and forwards its timing to the down layers through all the other ports.

5.2.1.2 External Atomic Clock (GM)

Web GUI selection: **GM: ext AC**

- The primary timing source is provided using an external atomic clock reference through the front-panel 10 MHz and 1PPS inputs (Grand-Master).
- All timing ports (i.e., wr0-wr15) are configured as WR masters.
- Clock accuracy is announced below or equal to 25 nanoseconds.
- Alignment of PPS_in VS PPS_out must be done manually (within picoseconds)
- PPS only needed at startup

It is recommended to use this preset when the device is configured to be the Grand-Master in the timing network and is disciplined using an Atomic Clock as external reference.

Here, Atomic Clocks means that a very stable oscillator based on hyperfine transition (e.g, Caesium) that provides very low daily uncertainties (e.g., 1 ns/day) is combined with a GNSS receiver to remove its slow drift using averaging methods. For telecom, this combination is also known as ePRTC and typically provides a UTC representation accurate within 10 ns or less. Moreover, in order to guarantee the best timing performance (phase noise & determinism) the automatic alignment of the PPS output onto the PPS input has been disabled.

Caution: 10 MHz + PPS signal calibration: When using this preset, the PPS must always keep the same delay in respect to the 10 MHz signal. The user can use the GM Offset field to compensate this fixed delay.

5.2.1.3 External GNSS Receiver (GM)

Web GUI selection: **GM: ext GNSS**

- The primary timing source is provided using an external GNSS receiver reference through the front-panel 10 MHz and 1PPS inputs (Grand-Master).
- All SFP ports (i.e., wr0-wr1) are configured as WR masters.
- Clock accuracy is announced below or equal to 100 nanoseconds.

- Alignment of PPS_in VS PPS_out is done automatically (adding ~50ps of uncertainties).
- PPS_in is mandatory to announce a valid time.

It is recommended to use this preset when a third-party GNSS is providing the reference to the Grand-Master device. This preset will also ensure the automatic alignment of the PPS input to the PPS output after each time the GNSS locked itself.

Caution: GNSS PPS output: The GM can lock to the PPS from the GNSS receiver before GNSS signal locked (before its 10MHz are locked in phase to its PPS). This causes a jump in the time reference. To avoid this situation, the user should configure the GNSS to not output any PPS before locking to GNSS signals.

Note: Inform GNSS status via PPS: This preset enforces a continuous detection of the PPS input. This means that if the GNSS receiver is configured to disable its PPS when it unlocks (e.g., signal lost), the Grand-Master will then automatically degrade itself as a Free-Running Grand-Master (see VSC-10102 in "Grand Master (GM VCS Code)" on page 234).

5.2.1.4 External Atomic Clock (GM) / PTP

Web GUI selection: **GM: ext AC | PTP**

- The primary timing source is provided using an external atomic clock reference through the front-panel 10 MHz and 1PPS inputs (Grand-Master).
- All timing ports (i.e., wr0-wr15) are configured as PTP masters.
- Clock accuracy is announced below or equal to 25 nanoseconds.
- Alignment of PPS_in VS PPS_out must be done manually (within picoseconds)
- PPS only needed at startup

It is recommended to use this preset when the device is configured to be the Grand-Master in the timing network and is disciplined using an Atomic Clock as external reference.

Here, Atomic Clocks means that a very stable oscillator based on hyperfine transition (e.g, Caesium) that provides very low daily uncertainties (e.g., 1 ns/day) is combined with a GNSS receiver to remove its slow drift using averaging

methods. For telecom, this combination is also known as ePRTC and typically provides a UTC representation accurate within 10 ns or less. Moreover, in order to guarantee the best timing performance (phase noise & determinism) the automatic alignment of the PPS output onto the PPS input has been disabled.

> **Caution:** 10 MHz + PPS signal calibration: When using this preset, the PPS must always keep the same delay in respect to the 10 MHz signal. The user can use the GM Offset field to compensate this fixed delay.

5.2.1.5 External GNSS Receiver (GM) / PTP

Web GUI selection: **GM: ext GNSS | PTP**

- The primary timing source is provided using an external GNSS receiver reference through the front-panel 10 MHz and 1PPS inputs (Grand-Master).
- All SFP ports (i.e., wr0-wr15) are configured as PTP masters.
- Clock accuracy is announced below or equal to 100 nanoseconds.
- Alignment of PPS_in VS PPS_out is done automatically (adding ~50ps of uncertainties).
- PPS_in is mandatory to announce a valid time.

It is recommended to use this preset when a third-party GNSS is providing the reference to the Grand-Master device. This preset will also ensure the automatic alignment of the PPS input to the PPS output after each time the GNSS locked itself.

> **Caution:** GNSS PPS output: The GM can lock to the PPS from the GNSS receiver before GNSS signal locked (before its 10MHz are locked in phase to its PPS). This causes a jump in the time reference. To avoid this situation, the user should configure the GNSS to not output any PPS before locking to GNSS signals.

Note: Inform GNSS status via PPS: This preset enforces a continuous detection of the PPS input. This means that if the GNSS receiver is configured to disable its PPS when it unlocks (e.g., signal lost), the Grand-Master will then automatically degrade itself as a Free-Running Grand-Master (see VSC-10102 in "Grand Master (GM VCS Code)" on page 234).

5.2.1.6 WR Slave @ wr0 > wr1 (BC)

Web GUI selection: **BC: wr0->wr1|WR**

- The primary timing source is provided using WR through interface wr0. It can failover to a secondary timing source provided using WR through interface wr1.
- All ports except for wr0 and wr1 are configured as WR masters.

This preset provides multi-source redundancy by allowing to configure the two first optical ports as possible timing sources. This means that in case of failure of the first port (i.e., wr0), the device will automatically switch to wr1 as it is configured as secondary source.

5.2.1.7 PTP Slave @ wr0 > wr1 (BC)

Web GUI selection: **BC: wr0->wr1|PTP (PTP)**

- The primary timing source is provided using PTP though interface wr0. It can failover to a provided secondary timing source using PTP through interface wr1.
- All ports except wr0 and wr1 are configured as PTP slaves.

This preset provides multi-source redundancy by configuring the two first optical ports as possible timing sources. In case of failure of the first port (i.e., wro), the device will automatically switch to wr1, as it is configured as a secondary source.

5.2.1.8 WR Slave @ wr0 / PTP Fan-Out

Web GUI selection: **BC: wr0|PTP**

- The primary timing source is provided using WR protocol through interface wr0.
- All ports except wr0 and wr1 are configured as PTP slaves.

This preset targets devices used as last hop with PTP. The primary timing source is provided using WR protocol through interface wr0. The other port is configured as IEEE 1588-2008 (PTPv2) masters to distribute timing to 3rd party devices.

5.2.1.9 WR Slave @ wr0>wr1 / PTP Fan-Out

Web GUI selection: **BC: wr0->wr1|PTP**

- The primary timing source is provided using WR through interface wr0. It can failover to a secondary timing source provided using WR through interface wr1.
- All ports except wr0 and wr1 are configured as PTP slaves.

This preset targets critical devices used as last hop with PTP. The primary timing source is provided using WR through interface wr0. It can failover to a secondary timing source provided using WR through interface wr1.

5.2.1.10 Manual Free-Running

Web GUI selection: **FR: Manual**

- The primary timing source is the free running internal oscillator in the device.
- All SFP ports (i.e., wr0 & wr1) are configured as WR masters.
- The device announces itself as a Free-running GM using arbitrary timescale (ARB).

This preset is useful for laboratory and test networks where each node is disciplined by the same free-running oscillator. Selecting this preset will also silence the possible warnings in devices of the down-layers and will preclude the use of the holdover as it cannot learn from a free-running oscillator.

Caution: It is highly recommended to avoid integrating a Manual Free-Running device to a timing network in production as in some corner cases the BMCA/FOCA algorithms might select this timing source when it is not the expected choice.

5.2.1.11 Custom

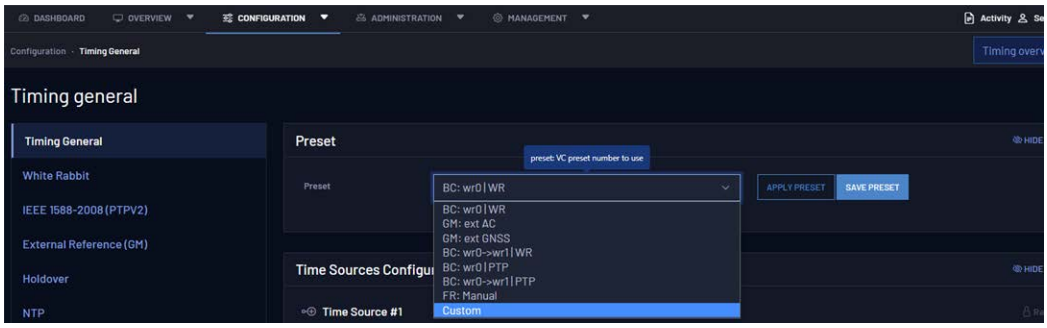
The Custom Preset has been designed to allow unique selections for timing source and fanout settings in order to meet any kind of user needs. If the user

needs a specific combination that mixes WR on some ports and PTP/IEEEE-1588 or NTP on others, he/she first select the Custom preset and then configure each interface.

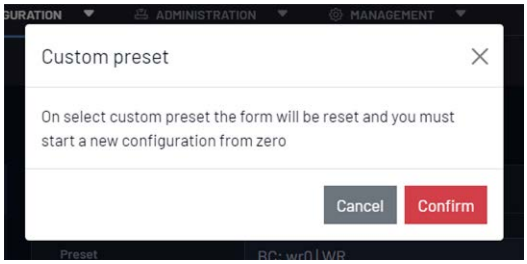
Custom Preset via the Web GUI

To configure a custom preset:

1. After logging in to the Web GUI, navigate to **Configuration** > **Timing General** > **Preset**. Select Custom from the drop down menu.



2. You will be given a warning prompt to acknowledge that your current preset settings (both Time Sources Configuration and Fanout Configuration) will be deleted from the form:



3. Modify each Timing Source in the **Timing Source Configuration** panel in the order they should be evaluated by the FOCA algorithm. If the Type field is left DISABLED, it will not be evaluated.

   The Timing Source Type options are DISABLED, WR, GM, and FR/HO. Selections in the Type field will determine available selections in the Name field.

   If you need to reorder your entries, the Rank can be adjusted here as well. To do so, use the blue rank-altering buttons for each timing source.

4. Modify each Fanout Source in the **Fanout Configuration** panel. The fanouts are listed by order of interface name.

   Set the Protocol field to determine the communication method used for that interface. The Protocol field options are DISABLED, WR, PTP, and NTP. If an interface is left as DISABLED, no timing information will be output by

that interface. The Mode field options will be determined by selections in the Protocol field.

5. After entering your information, select the Save Configuration button in the lower right of the window. (In order to test functionality, you may select Apply Configuration instead. This setting will not be persistent across reboots unless the configuration is also saved and the unit is rebooted).

   You will see a save confirmation banner and a warning that the saved changes will not be applied until the next reboot.

6. Once you have made your desired changes, you can either select the Reboot button from the warning banner, or navigate to **Management** > **Maintenance** and select the device Reboot button.

Custom Preset via the CLI

> **Note:** CLI and Custom Preset: The steps to follow in wrz_config (CLI) are slightly different than in the web as it is needed to select the Custom Preset to bring up the corresponding subset of parameters to the menu. If the Custom preset is not chosen in advance, these parameters will stay hidden and thus not configurable.

As shown in "Custom Preset with CLI tool" on the facing page, the preset must first be set to Custom to reveal each port's configurable parameters (other than alerts).

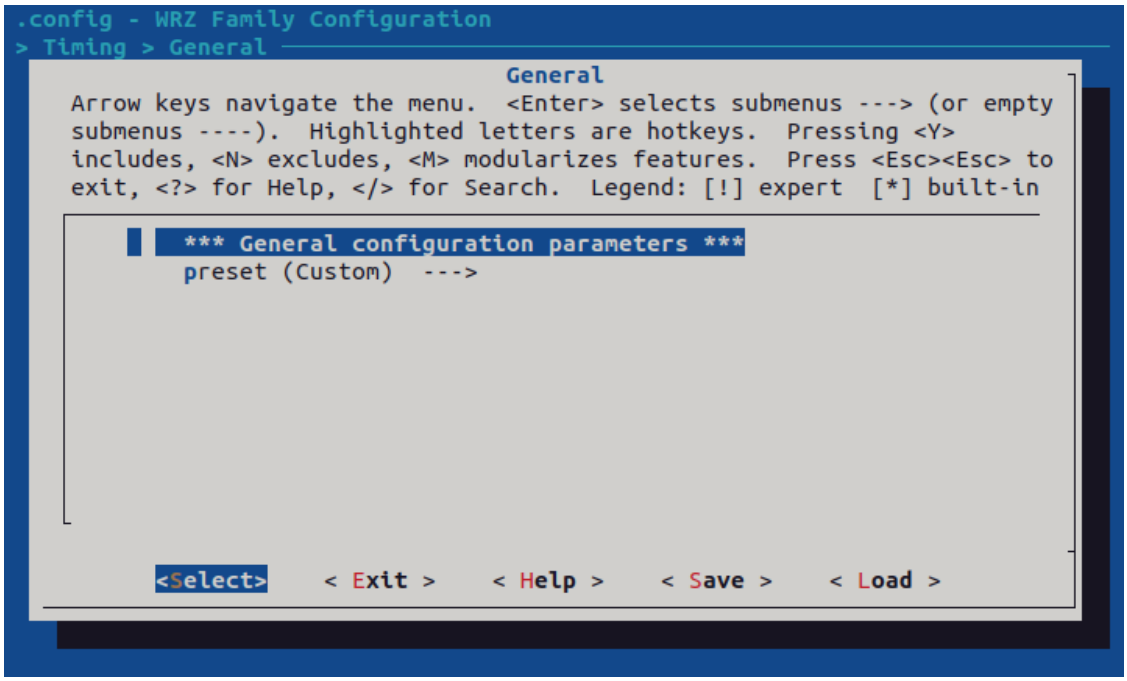


Figure 5-25: Custom Preset with CLI tool

Each port can then be configured independently with:

- Protocol: WR, PTP, NTP, Disabled
- Mode: Master, Slave, Auto , Survey
- Alerts:
  - Disable 'offset from master' alerts: Yes, No
  - Warning threshold: [0 to 2^{31}-1] Minimum value of offset from master to trigger warning.
  - Critical threshold: [0 to 2^{31}-1] Minimum value of offset from master to trigger critical state.
- Source Rank: [0-255], Order the timing source given the source rank priorities where:
  - 1 is the first source to be executed and 255 the last one.
  - If the source rank is set to 0, the port will not be included as a timing source.
  - This parameter is not used when the port role is Master.

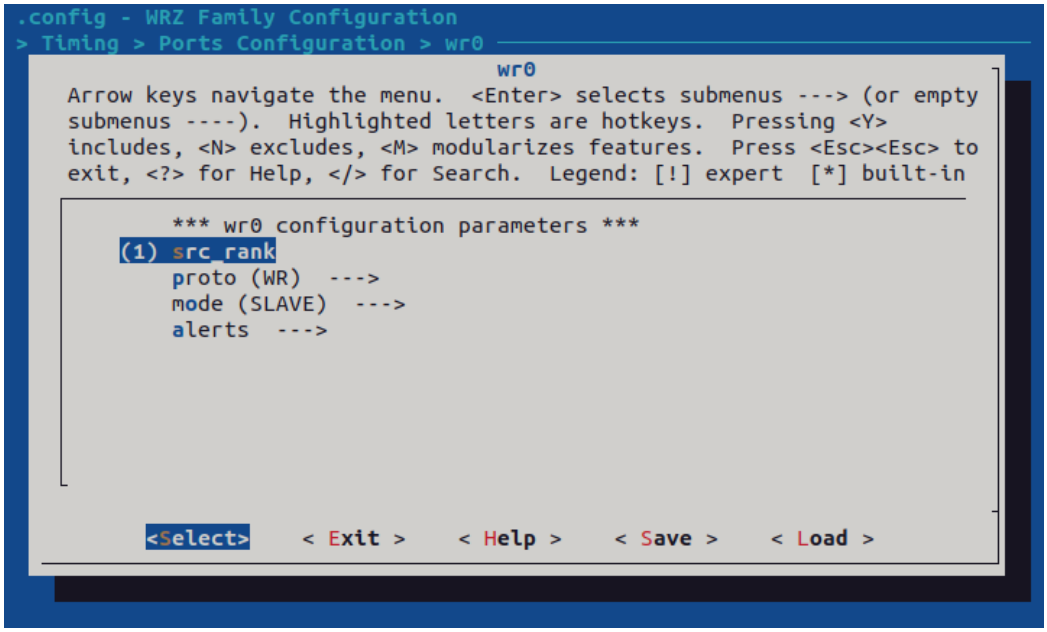


Figure 5-26: Port configuration (e.g., wr0) from CLI tool

5.2.1.12 PTP Slave @ wr0 (BC) / PTP

Web GUI selection: **BC: wr0 (PTP) | PTP**

- The primary timing source is provided using PTP through interface wr0.
- All ports except wr0 are configured as PTP masters.

5.2.1.13 PTP Slave @ wr0 (BC) / WR

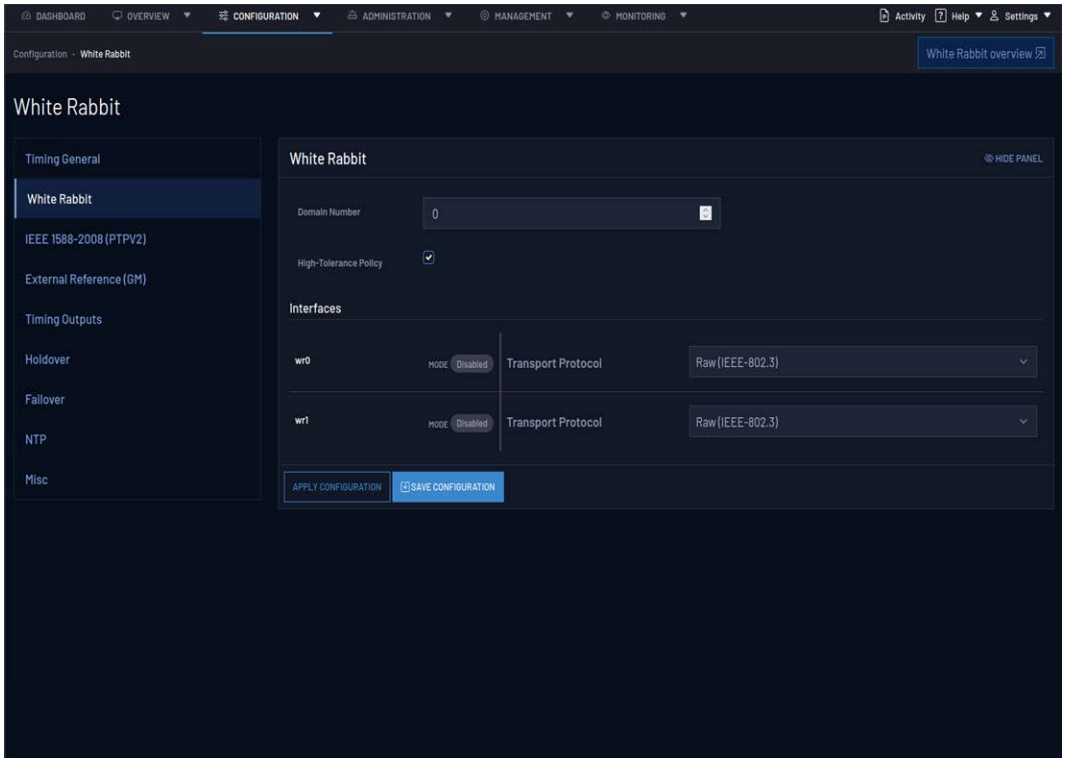
Web GUI selection **BC: wr0 (PTP) | WR**

- The primary timing source is provided using PTP through interface wr0.
- All ports except wr0 are configured as WR masters.

This preset needs to enable the High-Tolerance Policy option on all devices connected to the Grandmaster PTP reference. If you configure this preset using the WebUI or CLI, you will see a warning: "Please enable this option only on devices connected to a Grandmaster PTP reference."

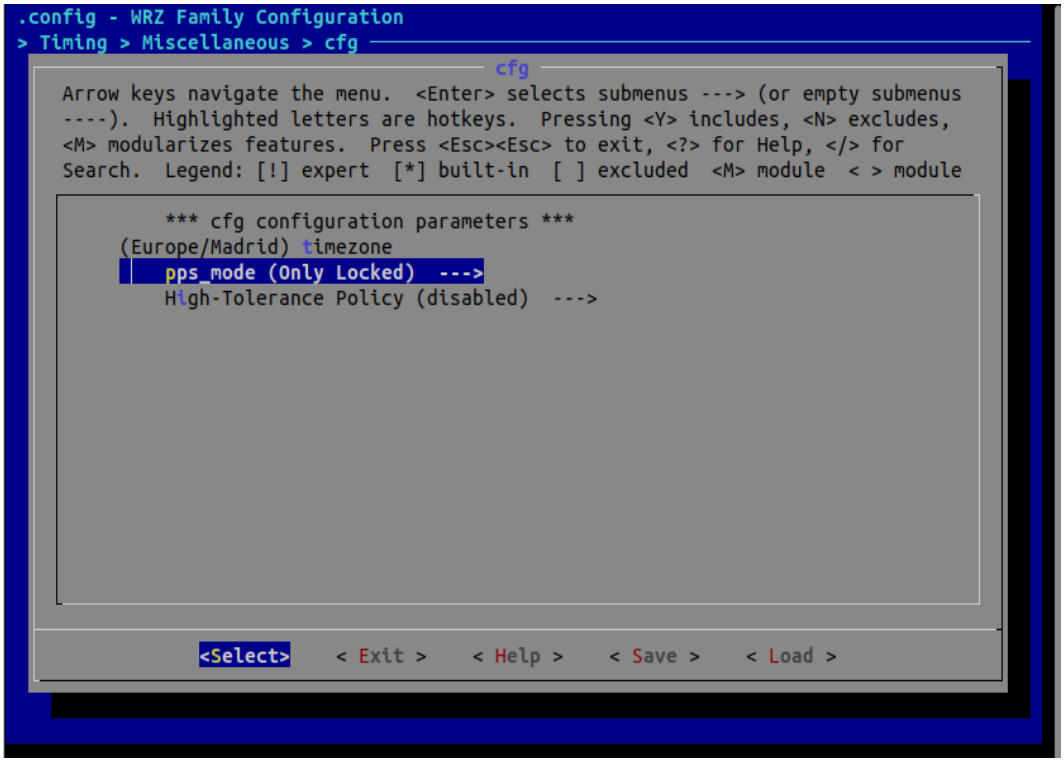
High-Tolerance Policy in the WebUI

Navigate to Configuration - White Rabbit and select the **High Tolerance Policy** checkbox.



High-Tolerance Policy in CLI

As shown in "Custom Preset with CLI tool" on the previous page, the user must navigate to Timing > Misacellaneus > cfg > and enable High-Tolerance Policy



5.2.2 Reference topology

The following figure summarizes how devices can be configured with different presets to operate on a generic timing network. To improve the comprehensibility of the reader, this reference topology has been separated in several theoretical layers:

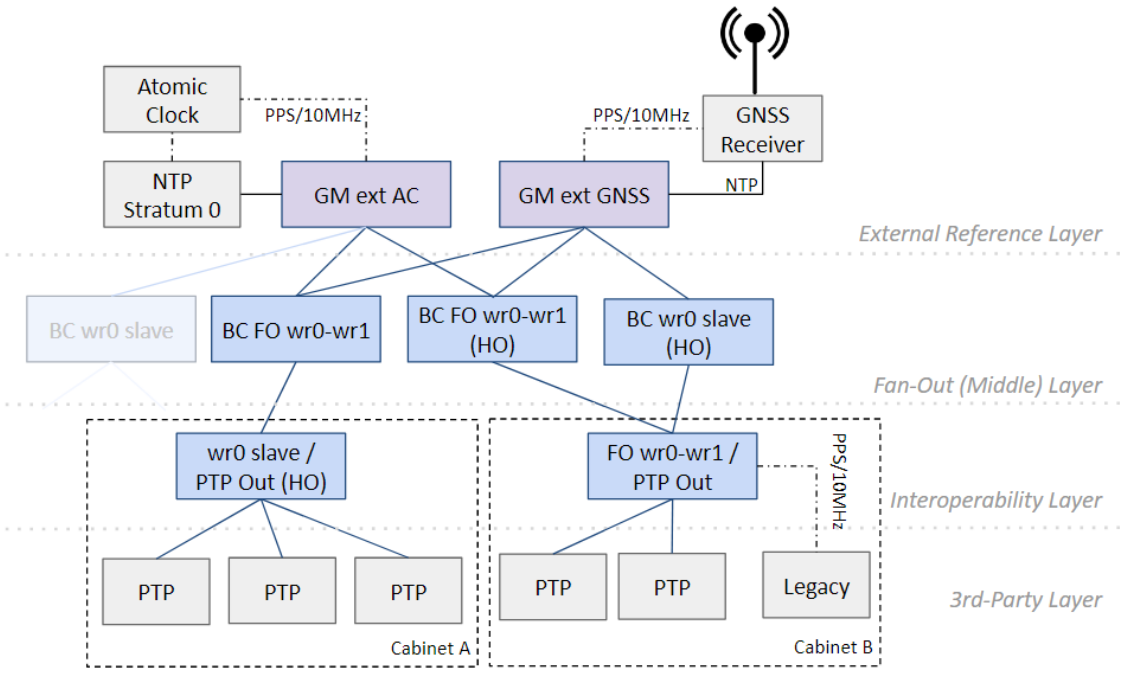


Figure 5-27: Reference topology with different presets.

- External Reference Layer: It includes the devices that will be fed by several external references (in grey), such as an Atomic Clock or a GNSS receiver, and will receive ToD (Time of Day) from an NTP server (external or embedded). These devices will act as Grand-Master (GM) in the timing network and their timing information will be forwarded to all the timing nodes.
- The Fan-Out Layer or Middle Layer: The devices in this layer are mainly dedicated to spread (fan-out) the timing synchronization to more devices on the down layers. In order to ensure continuous operation, they can be configured with redundant timing sources (e.g., BC FO wr0-wr1) or could incorporate the Holdover option (e.g., BC wr0 slave HO).
- The interoperability Layer: The devices that belong to this layer are also known as last-hop devices. Typically, one of these devices is placed per rack cabinet and is in charge of distributing the ultra-accurate timing provided by the White Rabbit network to other 3rd party devices in the cabinet via PTP, via 10MHz/PPS (legacy devices), etc.

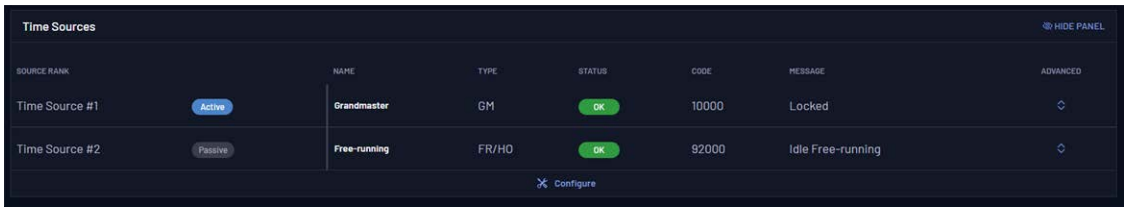
Note: This reference topology is a simplified version of a real timing network and the proposed structure in layers might not be respected: A last-hop device could be connected directly to the GM or an external GNSS reference could be used as backup in the fan-out-/interoperability layer.

Note: Some devices in the reference topology strategically include the holdover option (HO) to ensure continuous operation even if not locked to any timing sources. This option is automatically enabled if detected and the provided presets can be used without any modifications.

5.2.3 Timing source info

Each timing source shares a common set of values processed by the strategy in order to decide how to discipline the virtual clock of the device.

By navigating to Overview > Timing Overview the user will be able to quickly understand the state of all timing sources.The figure below shows the parameters related to the primary (#1) timing source.



[Above screenshot to be updated]

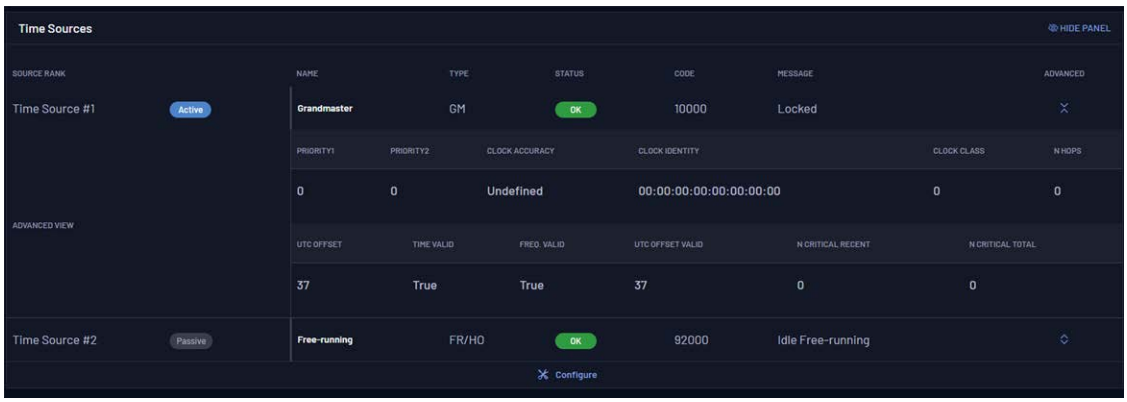
Figure 5-28: Timing Sources Overview panel

The parameters contained in the previous table are described as follows:

OID	Name	Value	Description
3.13x0.x	**tsrc_info/x/xxx**		**Information about the x timing source.**
3.13x0.1	Name	<String> (i.e., wr0, front-panel, eth1, etc.)	Name of the corresponding timing source.
3.13x0.2	Type	GM WR PTP HO/FR	Type of timing source, each type can have slightly different state machines to properly handle its timing source.

OID	Name	Value	Description
3.13x0.3	VCS Code	<Integer>	Code defined in the VSC table ("VCS Code" on page 233) that corresponds to a given condition for this timing source.
3.13x0.4	Status	Disabled OK Warning Critical	Status that corresponds to the code defined in the VSC table.
3.13x0.5	Message	<String>	Message that corresponds to the code defined in the VCS table.
3.13x0.6	Is Active	<Boolean>	Flag that indicates if this timing source has been selected by the policy to actively discipline the virtual clock of our device.

The timing source can be expanded to show its advanced view by clicking on the up-down icon under Advanced:



[Above Screenshot to be updated]

Figure 5-29: Advanced info Time Source #1

The advanced view parameters are described below:

Table 5-1: Timing source info description

OID	Name	Value	Description
3.13x1.xx	**tsrc_info/x/Q**		**Clock Quality of the x timing source.**
3.13x1.1	Clock Identity	<String>	Unique identity of PTP instance in the network.
3.13x1.2	Priority1	<Integer> Default: 128	Force BMCA decision using 1st priority (Lower values take precedence).
3.13x1.3	Priority2	<Integer> Default: 128	Manually force BMCA to select a clockID when clock quality is the same (Lower values take precedence).

OID	Name	Value	Description
3.13x1.10	Clock Class	<Integer> Default: 248	The Clock Class is one of the attributes that characterizes the timing source.
3.13x1.11	Clock Accuracy	<Enum>	It indicates the expected accuracy of the timing source. It shall be conservatively estimated based on the time source.

5.2.3.1 Timing Alerts

The Offset Alert parameter displays a warning or critical alert based on specific time error thresholds configured in **MANAGEMENT** > **Timing alerts**.

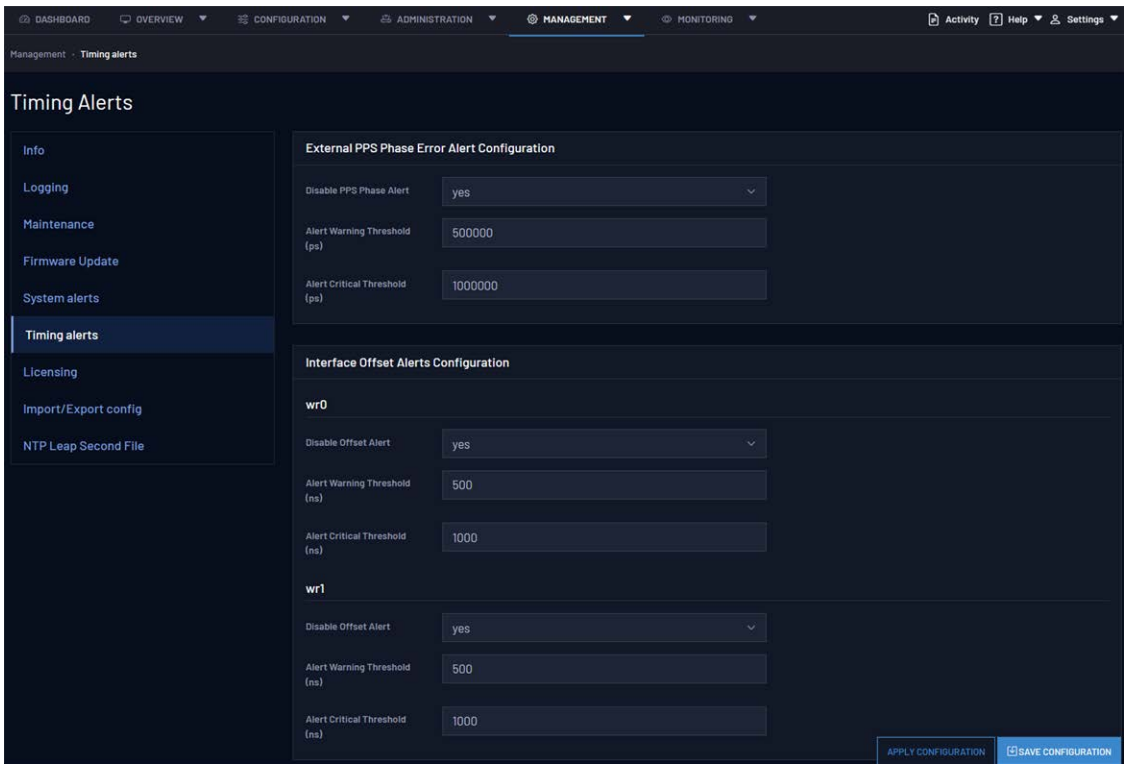
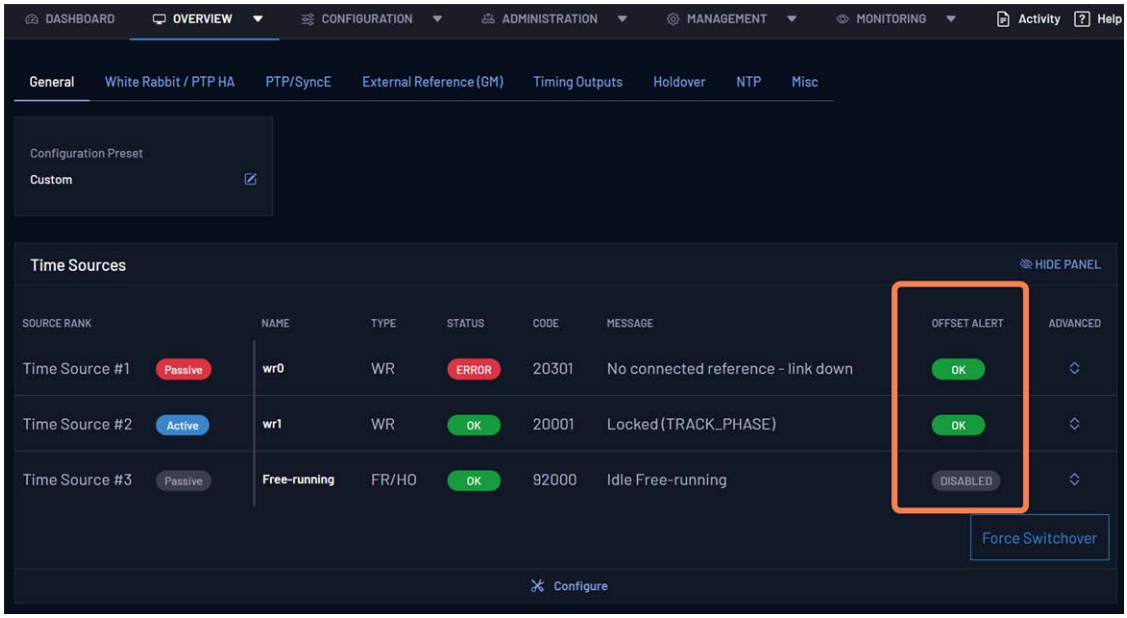


Figure 5-30: Timing alerts page

Depending on the configured thresholds, parameters will display as OK, WARNING, or CRITICAL.

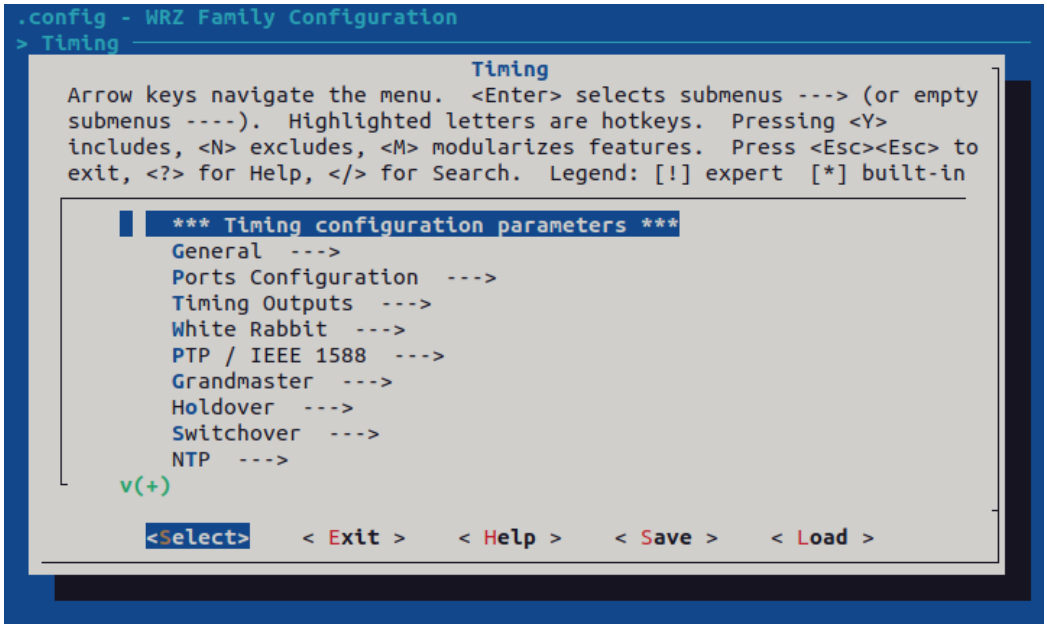


Alerts are disabled by default and may be enabled by selecting "no" under "Disable Offset Alert." Alerts for each interface may be individually configured.

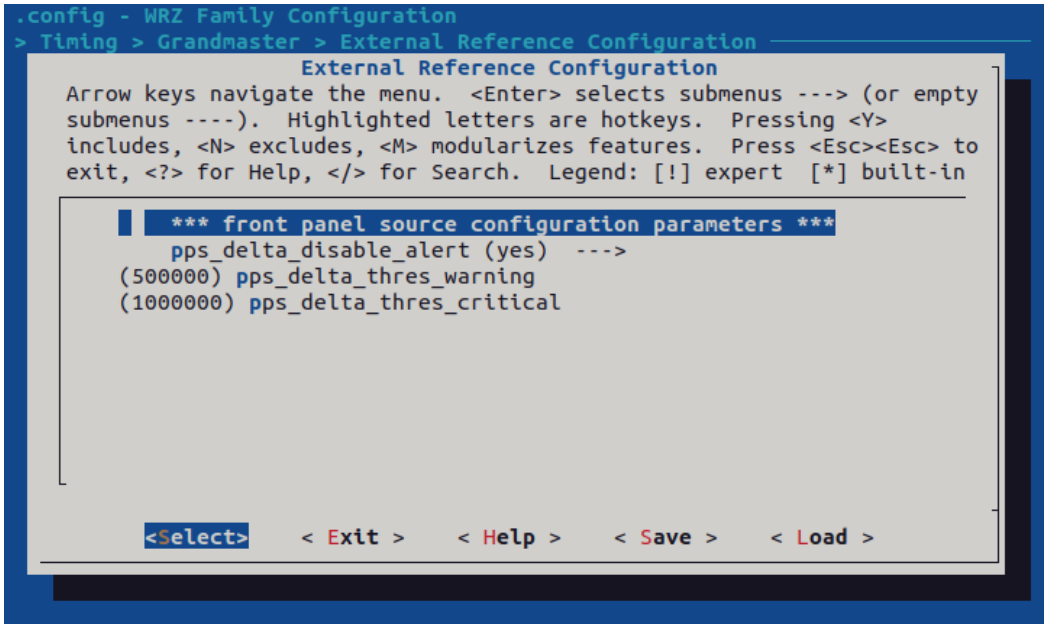
After configuring alerts, select the "SAVE CONFIGURATION" button. To apply the saved configuration settings, select "APPLY CONFIGURATION."

To configure Timing Alerts through the CLI:

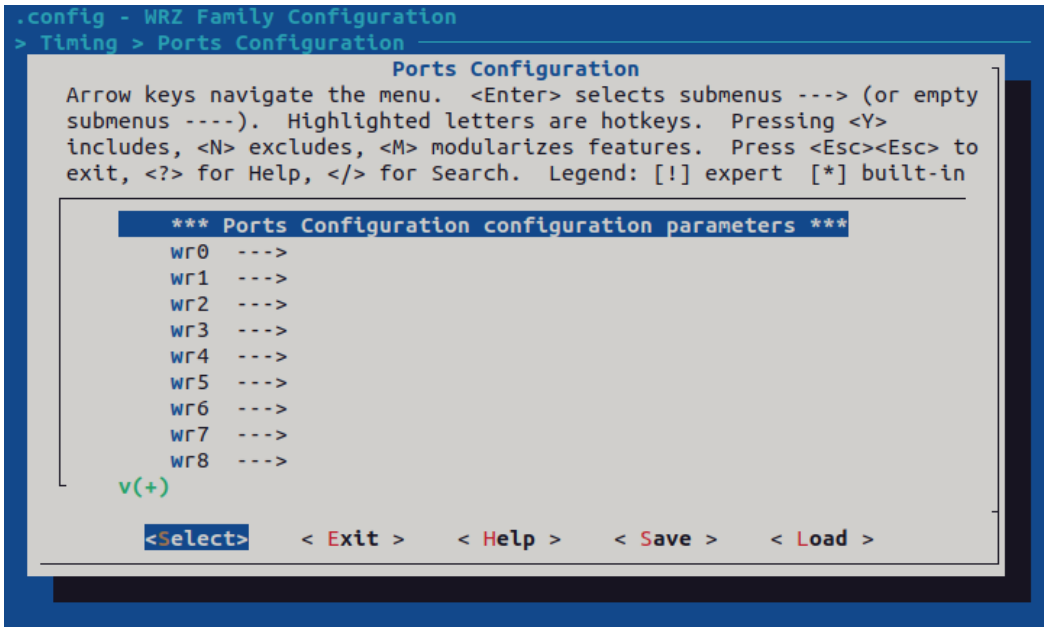
1. The first step is to execute the wrz_config command from a terminal.
2. Select the Timing section from the main menu.



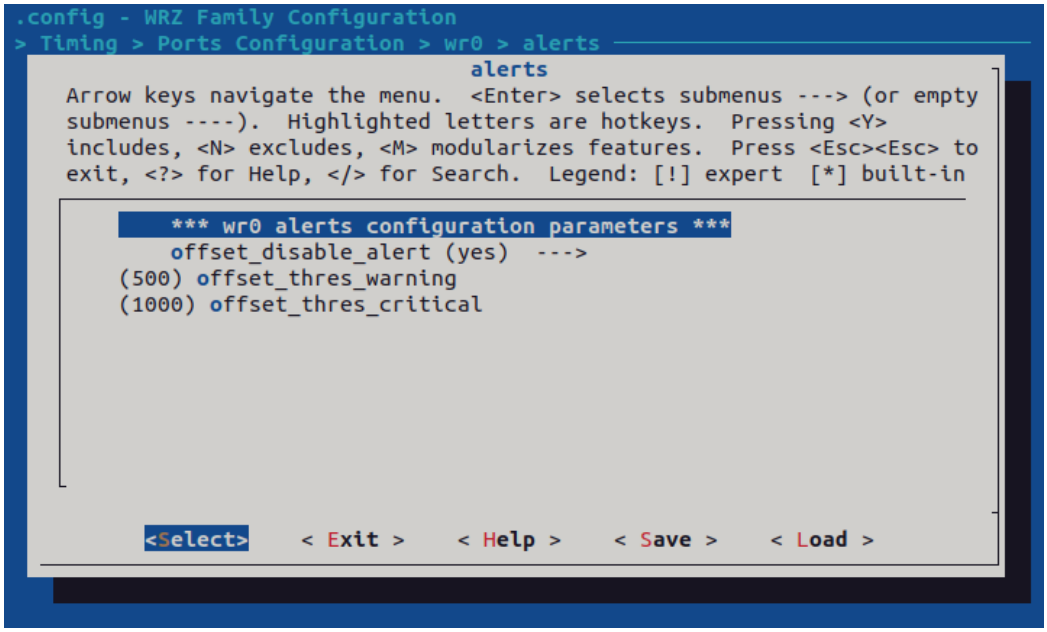
3. To access PPS phase alert configuration, select the Grandmaster section and then select External Reference Configuration. Proceed to make your changes.



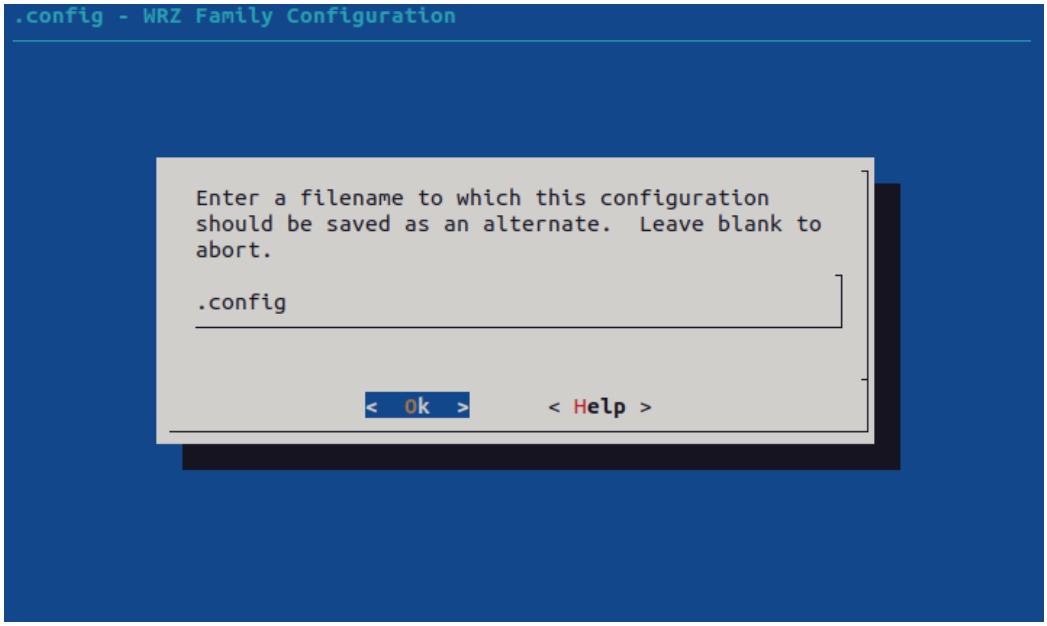
4. To access interface offset alert configuration, select the Ports Configuration section and then select the interface you would like to configure.



5. Select alerts and proceed to make your changes.



6. Do not forget to <Save> the changes once the configuration is done. The following message will prompt. To load this configuration at next reboot the default filename (.config) must be used.



5.3 White Rabbit / IEEE 1588-2019 HA

5.3.1 Configuration

After configuring if White Rabbit is a Slave (Timing Source) or a Master port, desired interfaces may be configured.

To configure desired interfaces, log in to the Web GUI and navigate to **Configuration** > **White Rabbit**.

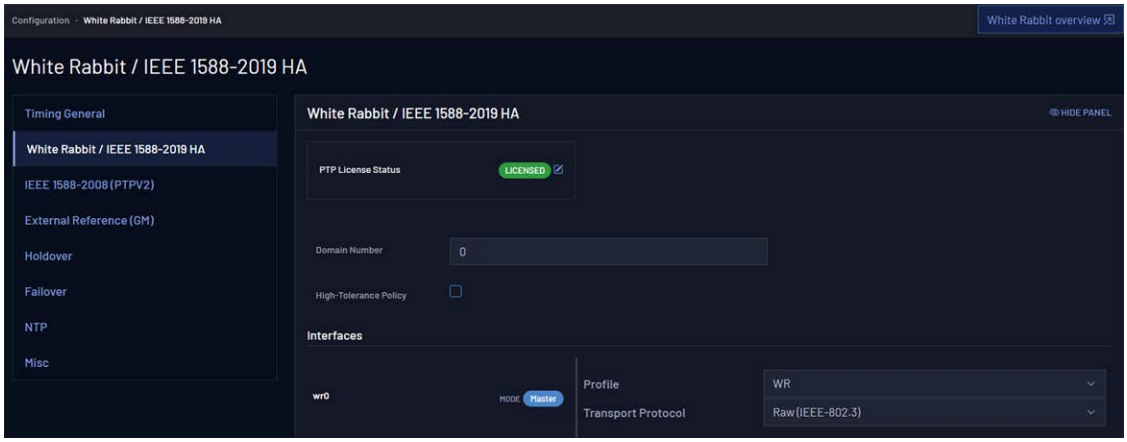


Figure 5-31: Configuration of WR instance.

5.3.1.1 WR Profile

By default, WR uses the data layer (RAW Ethernet packets - IEEE 802.3) to communicate to other WR devices but it can be configured to also use the UDP/IPv4 packets.

The table below summarizes the configuration limitations for the WR profile.

OID	Name	Default Value	Possible Values	Note
1.xx11.101	Transport Protocol	Raw (IEEE-802.3)	Raw (IEEE-802.3) and UDP/IPv4	

5.3.1.2 HA Profile

A PTP license is required for full access to HA profile configuration.

For all topics related to PTP license management such as purchasing, activating, and verifying see" Licenses" on page 202.

Once the license is activated (see image below) the web interface should allow the user to configure these settings.

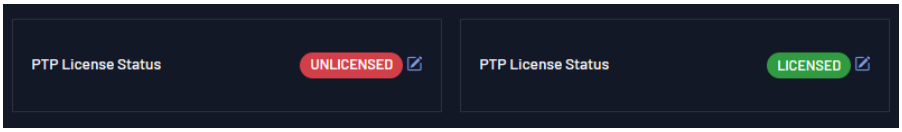


Figure 5-32: Unlicensed and Licensed PTP status

The table below summarizes the configuration limitations for the HA profile.

OID	Name	Default Value	Possible Values	Note
1.xx11.108	Delay Mechanism	E2E	E2E and P2P	
1.xx11.101	Transport Protocol	Raw (IEEE-802.3)	Raw (IEEE-802.3) and UDP/IPv4	
1.xx11.111	Announce Interval	1	0 to 4	The value is $\log_2(x)$ where x is the user inputted value.
1.xx11.106	Announce Receipt Timeout	3	2 to 10	The value is $\log_2(x)$ where x is the user inputted value.
1.xx11.113	Sync Interval	0	-1 to 1	The value is $\log_2(x)$ where x is the user inputted value.
1.xx11.112	Minimum Delay Request Interval	0	0 to 5	The value is $\log_2(x)$ where x is the user inputted value.
1.1100.103	Domain	0	0 to 127	
1.xx11.132	Asymmetry Correction	yes	yes and no	
1.xx11.104	Extension	l1sync	l1sync	
1.xx11.116	L1sync Interval	0	-4 to 4	The value is $\log_2(x)$ where x is the user inputted value.
1.xx11.117	L1sync Receipt Timeout	3	2 to 10	The value is $\log_2(x)$ where x is the user inputted value.
1.xx11.120	Egress Latency	0	-140737488355328000 to 140737488355327000	Measured in picoseconds.
1.xx11.121	Ingress Latency	0	-140737488355328000 to 140737488355327000	Measured in picoseconds.

5.3.2 Info/Overview

5.3.2.1 Active servo

When the device is running as a WR Boundary Clock, this means that one of the fiber network interfaces is an active slave. The data related to how the servo disciplines the internal oscillator can thus be visualized under the active servo panel.

OID	Name	Value Type	Description
1.1220.x	**act/servo**		**Information about the active servo instance**
1.1220.1	Interface Name	<String>	Name of the network interface on which the servo is running.
1.1220.6	State	0. Disabled 1. Adjusting Time 2. Adjusting Time 3. Adjusting Phase 4. Locked 5. Wait Stable Phase 6. Invalid 7. Undefined 8. Not Updated 9. Wait Time Adjust 10. Wait Phase Adjust 11. Initializing	Servo State: where 'Locked' corresponds to the legacy TRACK_PHASE state and means that the corresponding interface is actively disciplining the device. Disabled is used when the port is setup as Master or does not receive any valid PTP/WR exchange. **Note:** The 1st state corresponds to adjustments in seconds order, and the 2nd one to adjustments in nanoseconds order.
1.1220.5	Up Count	<Integer> (u32)	Number of updates for the servo. It is typically increased by 1 each second.
1.1220.10	Mean Delay	<Decimal> (f64) Unit: s	Half of the cable round trip time excluding fixed+semistatic (cRTT/2).
1.1220.11	Delay MS	<Decimal> (f64) Unit: s	Calculated Delay between Master and Slave considering asymmetry and fixed delays.
1.1220.21	Egress Latency	<Decimal> (f64) Unit: ns	Fixed latency between the moment when a PTP packet is timestamped to its exit on the physical layer (i.e., optical fiber). Legacy: 'WR Slave ΔTx'.

OID	Name	Value Type	Description
1.1220.20	Ingress Latency	<Decimal> (f64) Unit: ns	Fixed latency between the moment when a PTP/WR packet ingresses from the physical layer to its timestamp. Legacy: 'WR Slave ΔRx'.

5.3.2.2 Port Instance

A WR port instance is then associated to each network interface. The table displayed in the following image provides a quick overview of the state of each interface.

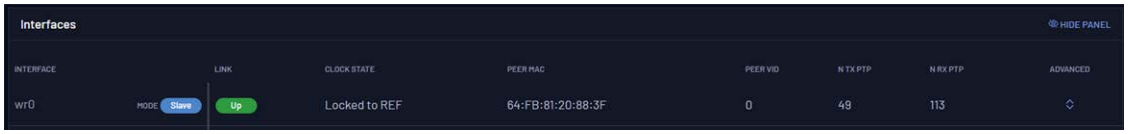


Figure 5-33: WR Interfaces overview (Only first interface captured (wr0)).

The parameters shown are explained in the following table:

OID	Name	Value Type	Description
1.xx10.x	**net/wrX/1/**		**Information about WR for the wrX network interface. (Where OIDs follow the given pattern: wr0 → 20xx, wr1 → 21xx, ..., wr15 → 35xx)**
1.xx10.5	Link	Down Up	Specify if the link is up or down.
1.xx10.10	Port State	0. None 1. Initializing 2. Faulty 3. Disabled 4. Listening 5. Pre-Master 6. Master 7. Passive 8. Uncalibrated 9. Slave	Current state of the port that changes according to the PTP protocol events. • If this port is configured as a timing source, it can be Slave (active) or Passive (only handle announce messages) →Color: Blue. • The port state is Disabled when the link is down or when the port has been configured with PTP instead of WR. • The port state will be Master if it distributes WR timing (Color: Purple). • Finally, the other states are transition states (mainly used by BMCA) or error states.

OID	Name	Value Type	Description
1.xx10.11	Clock State	0. Idle 1. Locking 2. Locked to REF 3. Holdover 4. Error 5. Free-Running	State of the clock (internal oscillator) shared by all PTP instances. “Locked to Ref” is the desired stated.
1.xx10.20	Peer MAC	<Data Array> (6 x u8)	MAC address of the latest peer.
1.xx10.23	Peer VID	<Integer> (u16)	VLAN ID of the connected peer.
1.xx10.25	Peer N Tx PTP	<Integer> (u32)	Number of transmitted PTP packet on this port.
1.xx10.26	Peer N Rx PTP	<Integer> (u32)	Number of received PTP packet on this port.

If the interface is currently running WR (“Port State” not “Disabled”), the user can expand (+) a specific interface to display an Advanced Overview (See the figure below):

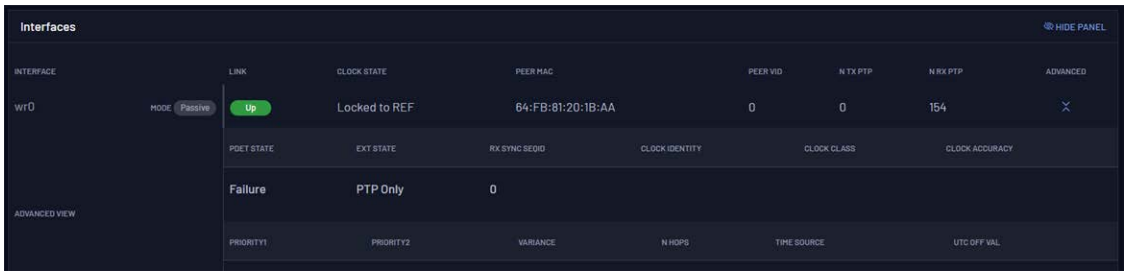


Figure 5-34: Advanced WR interface Overview (WR0 configured as slave)

Note: The clock information (clock quality & time properties) displayed in the expanded view above corresponds to the announced messages received on this specific interface and not the transmitted ones. This information is irrelevant (disabled) if the link is down or when the connected peer is not sending any announce messages (e.g., slave role).

The information shown in the advanced overview menu is explained in the following table:

OID	Name	Value Type	Description
1.xx10.x	**net/<wrX>/1/**		**Information of the corresponding WR port instance (wrX). (Where OIDs follow the given pattern: wr0 → 20xx, wr1 → 21xx, ..., wr15 → 35xx)**
1.xx10.6	PDet State	0. None 1. Waiting 1st Msg 2. Checking 3. Detected 4. Failure	State of the Protocol Detection.
1.xx10.5	Ext State	0. Disabled 1. Active 2. PTP Only	State of the extension. If PTP Only this means that the WR extension has not been detected.
1.xx10.5	Rx Sync ID	<Integer> (u16)	Receive Sync Sequence ID.
1.xx10.5	Peer VID	<Integer> (u16)	VLAN ID of the connected peer.
1.xx31.xx	**net/<wrX>/1/clk/Q/**		**Clock Quality of the corresponding WR port instance (wrX).**
3.13x1.1	Clock Identity	<Data Array> (8 x u8)	Unique identity of PTP instance in the network.
1.xx31.2	Priority1	<Integer>	Force BMCA decision using 1st priority (Lower values take precedence).
1.xx31.3	Priority2	<Integer>	Used by BMCA to force selection between two clocks when their clock qualities are the same (Lower values take precedence).
1.xx31.10	Clock Class	<Integer>	The Clock Class is one of the attributes that characterizes the port instance.
1.xx31.11	Clock Accuracy	<Enum>	It indicates the expected accuracy of the timing source. It shall be conservatively estimated based on the time source.
1.xx31.12	Variance	<Integer> (u16)	Estimation of the variations of the Local PTP Clock as measured by comparison to a suitable reference clock.

OID	Name	Value Type	Description
1.xx31.20	N Hops	<Integer> (u32)	Number of PTP communication paths traversed between this PTP instance to the GrandMaster PTP Instance (aka stepsRemoved).
1.xx32.xx	**net/<wrX>/1/clk/tprop/**		**Time Properties of the corresponding WR port instance (wrX).**
1.xx32.1	Time Source	<Enum>	This information-only attribute indicates the immediate source of time used by the Grandmaster.
1.xx32.12	UTC Offset Valid	<Bool>	True, if the current UTC offset is known to be valid (It will handle the next leap second jump).

Note: To obtain more details about time properties & clock quality of a given WR port instance, the user should use gpa_ctrl tool with the -a (expert) flag or with the –A (expert & disabled) flag.

5.4 PTPv2.1 (Precision Time Protocol)

The Precision Time Protocol (PTP) and Synchronous Ethernet (SyncE) are technologies used for time and frequency synchronization. **PTP** distributes timing information throughout a network, which is necessary for applications that require precise timing, such as telecommunications, financial systems, and industrial automation. **SyncE** provides frequency alignment over Ethernet networks by transmitting clock signals alongside data traffic to maintain a stable and accurate frequency.

To address the specific needs of various industries and applications, PTP defines multiple profiles. Each profile outlines a set of configuration parameters and behaviors tailored for particular environments. SyncE can be activated alongside any of these **PTP profiles** to enhance frequency synchronization across the network. The supported profiles are:

- Default PTPv2.1 (IEEE 1588-2019)
- Telecom profiles:
  - G.8265.1
  - G.8275.1

- Power profiles:
  - IEEE C37.238-2017
  - IEEE 61850-9-3
- Enterprise
- IEEE 1588-2019 HA

The following sections will explain how to configure PTP and SyncE, provide examples to help you set up your network for synchronization, and describe how to check the status and obtain information about PTP and SyncE modules.

5.4.1 License

For full access to the PTP and SyncE modules configuration, a specific license must be purchased. Without a valid license, the PTP instance will be configured with default profiles and parameters, and SyncE will be disabled. However, the user will be able to configure the port role (Master, Slave, Survey, or Disabled), network mode (Layer 2/Ethernet or Layer 3/IPv4), and offset correction. The following table compares PTP and SyncE configuration options available with and without a valid license:

PTP	With License	Without License
Mode	Master, Slave, Disabled	Master, Slave or Disabled
Profile	Default (1588-2019), 8265.1, 8275.1, C37.238, 61850-9-3	Default 1588-2019
Transport Protocol	Ethernet (Layer 2) or IPv4 (Layer 3)	Ethernet (Layer 2) or IPv4 (Layer 3)
Transport Mode	Multicast, Unicast, or Hybrid	Multicast
Unicast Negotiation	Enabled or Disabled	Disabled
Enable SyncE	Off or On	Off
Delay Mechanism	End to End (E2E) or Peer to Peer (P2P)	End to End (E2E)
Domain	0-127	0
Announce ratio	[1 msg/128s, 16 msg/s]	1 msg/2 sec
Sync ratio	[1 msg/128s,128 msg/s]	1 msg/sec
Delay Req ratio	[1 msg/128s,128 msg/s]	1 msg/sec
Offset correction	Any number (in nanoseconds)	Any number (in nanoseconds)

5.4.1.1 PTP Web UI License Management

For all topics related to PTP license management such as purchasing, activating and verifying, see " Licenses" on page 202.

Once the license is activated (see image below) the web interface should allow the user to configure these settings as explained in "Configuration Parameters" below.

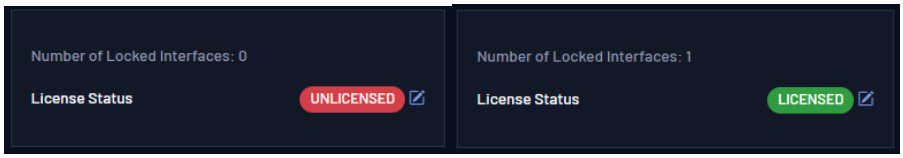


Figure 5-35: Unlicensed and Licensed PTP status as seen in the PTP configuration section of the WebUI.

5.4.2 Configuration Parameters

PTP and SyncE both have complex configuration parameters that influence synchronization mechanisms. Connected masters and slaves must have the same configuration settings to ensure proper synchronization.

Master/Slave Port Configuration: The port's role is determined by the selected Preset (see " Presets" on page 79) and cannot be modified directly in the PTPv2 configuration tab.

> **Note: Note:** While PTP offers numerous options to support a wide range of applications, PTP profiles restrict certain settings to limit compatibility to a specific subset of PTP. Therefore, users should consider these restrictions when configuring PTP for a particular profile.

The configuration parameters associated with PTP and SyncE are listed in the two tables below. Examples of how to configure these parameters using the Command Line Interface (CLI) and the Web User Interface (Web UI) are provided later in "Command Line Interface for PTP and SyncE " on page 120 and "PTP Web UI Configuration and Overview" on page 122. Additionally, you can configure PTP and SyncE through SNMP ("SNMP " on page 162), REST API ("The REST-API" on page 31), and wrz_config ("CLI Configuration" on page 47).

OID	Name	Value Type	Description
19.xx13.8	Profile <profile>	<Enum> 0. Default 1. Custom 2. Telecom 8265.1 3. C37.238-2017 Power 4. 61850-9-3 Power 5. Telecom 8275.1 6. Enterprise Default: Default profile.	Profile selected (Default, Telecom, etc.) with a set of options pre-configured. The PTP profiles and their configuration are explained in "Profiles" on page 108.
19.xx13.9	Transport Protocol <transport_protocol>	<Bool> 0. Layer 2 1. UDP/IPv4 Default: Layer 2.	Define the network layer that delivers the PTP packets.
19.xx13.10	Delay Mechanism <delay_mech>	<Bool> 0. E2E 1. P2P Default: E2E.	Specifies the path delay measurement mechanism used by the PTP port. It can be configured as E2E (End-to-End) or P2P (Peer-to-Peer).
19.xx13.11	Transport Mode <transport_mode>	<Enum> 0. Multicast 1. Unicast 2. Hybrid Default: Multicast Default: Multicast.	By default, multicast is used to automatically discover the PTP peers. When unicast mode is selected, the PTP topology must be pre-defined by filling Unicast Destination. In hybrid mode, delay request and response packets are sent via unicast, while other packets are sent via multicast.
19.xx13.56	Enable SyncE <enable_synce>	<Bool> 0. No 1. Yes Default: No.	Enables SyncE (Synchronous Ethernet), a syntonization protocol that locks the clock over Ethernet networks, improving PTP frequency stabilization.
19.xx13.12	Unicast Negotiation <unicast_neg>	<Bool> 0. Off 1. On Default: Off.	This option is necessary for ITU-T G8265.1 profile compliance. It will be automatically activated when Unicast mode is enabled.
19.xx13.13	Unicast Destination <unicast_dest>	<String> Default: empty.	For unicast slaves, specifying the IPv4 addresses or MAC addresses is mandatory because they must request synchronization from predefined Grandmasters (GMs). You can enter multiple IPv4 and MAC addresses, separated by commas, to configure communication with multiple devices.

OID	Name	Value Type	Description
19.xx13.14	Domain <domain>	<Integer> [0-255] Default: 0.	Specifies the domain number for PTP packets. Multiple PTP domains can operate simultaneously in the same network; however, some PTP profiles may restrict the domain number value (e.g., G.8265.1).
19.xx13.20	Announce Rate <announce_ rate>	<Enum> From 1 packet every [128,64,32,16,8,4,2] seconds to [1,2,4,8,16] packets per second. Default: 1 packet each 2 seconds.	Sets the rate at which Announce messages are sent. The Announce Rate determines how frequently clock hierarchy information is updated.
19.xx13.21	Sync Rate <sync_rate>	From 1 packet every [128,64,32,16,8,4,2] seconds to [1,2,4,8,16,32,64,128] packets per second. Default: 1 packet per second.	Rate of the Sync (and follow-ups) packets.
19.xx13.22	Delay Requests Rate <delayreq_ rate>	<Enum> From 1 packet every [128,64,32,16,8,4,2] seconds to [1,2,4,8,16,32,64,128] packets per second. Default: 1 packet per second.	Sets the rate at which Delay Request messages are sent. The Delay Request Rate determines how frequently the device measures network delay to its master clock, affecting synchronization accuracy. It is used in E2E PTP networks.
19.xx13.23	Peer Delay Request Rate <peer_ delayreq_rate>	<Enum> From 1 packet every [128,64,32,16,8,4,2] seconds to [1,2,4,8,16,32,64,128] packets per second. Default: 1 packet per second.	Sets the rate at which Peer Delay Request messages are sent. The Peer Delay Request Rate determines how frequently the device measures network delay to its master clock, affecting synchronization accuracy. It is used in P2P PTP networks.
19.xx13.18	User Offset <user_offset>	<Integer> Default: The default value depends on the device. A typical value is beteween -180 and -250.	Allows the user to set an offset to compensate for internal PTP delays. However, devices are calibrated for each release, so changing the default value is not recommended.

OID	Parameter Name <Command Line Interface Name>	Value Type	Description
9.xx08.2	SSM QL Mode <ssm_ql_mode>	<Enum> Auto Manual Default: Auto	Defines the method for selecting the Quality Level (QL) codes transmitted in Synchronization Status Messages (SSM). In Auto mode, the device forwards the SSM QL code received from its slave SyncE port. In Manual mode, the user defines the transmitted SSM QL code using the "SSM QL TX" parameter. If no port is configured as a Slave PTP, the transmitted SSM QL is derived from the clock class, as specified by the SyncE standard. In the WebUI, this parameter and the one below are combined into a single configuration dropdown menu for convenience.
9.xx08.3	SSM QL TX <ssm_ql_tx>	<Enum> Not Initialized QL PRC QL SSU A QL SSU B QL SEC QL DNU Default: PRC if the "SSM QL Mode" parameter is configured as "Manual".	Specifies the Quality Level (QL) code transmitted by the device in Synchronization Status Messages (SSM) when operating in Manual mode. This parameter enables you to define a specific SSM QL code to indicate the quality of the clock source to other devices in the network. Note that in the WebUI, this parameter and the one above are combined into a single configuration dropdown menu for simplicity.

5.4.3 Profiles

A valid license is needed to perform the configuration of PTP profiles (see" License" on page 104). You will only be able to configure the PTPv2 section if you have configured a valid PTP configuration in **Configuration** > **Timing General**. If your presets allow a valid MASTER or SLAVE port state, you can configure your

PTP profiles for each port by navigating to **Configuration** > **IEEE 1588-2008 (PTP)**.

Once you have set up a valid PTP configuration in **Timing General**, you can select a profile by navigating to **Configuration** > **IEEE 1588-2008 (PTPV2)**.

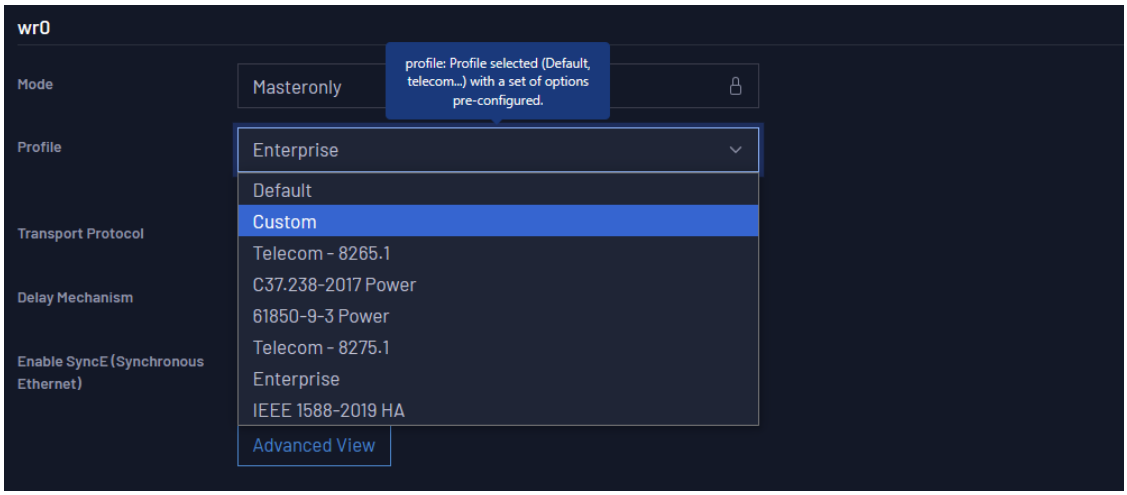


Figure 5-36: Selecting PTP Profiles

Selecting a profile in the dropdown menu will automatically configure the PTP parameters of that port with the default values of the chosen profile.

Advanced settings can be accessed by selecting the **Advanced View** button to expand the configuration of a given interface.

Note: Custom Profile: If settings are not modified accordingly to the specification of the selected profile, the web interface will automatically switch to the Custom profile. This will not restrict the user to any configuration.

The tables below summarize the configuration limitations for each PTP profile supported by the WR-Z16 device. Some settings have no default value but must fit within a specific range.

5.4.3.1 Default Profile

The Default PTP profile, defined in IEEE 1588-2019, provides a general-purpose configuration for clock synchronization. It ensures interoperability across devices and suits environments requiring basic timing without specific industry constraints.

OID	Name	Default Value	Possible Values	Note
19.xx13.9	Transport Protocol	Layer2	Layer2 and UDP/IPv4	Before configuring UDP/IPv4, an IP address must be set in the "Administration" > "Network" Section.
19.xx13.10	Delay Mechanism	E2E	E2E and P2P	E2E and P2P are supported.
19.xx13.56	Enable SyncE (Synchronous Ethernet)	No	No and Yes	
9.xx08.3	SyncE TX Quality Level	Auto	Auto and Manual	Available if the "Enable SyncE" parameter is Configured as "Yes".
19.xx13.11	Transport Mode	Multicast	Multicast, Unicast and Hybrid	
19.xx13.12	Unicast Negotiation	Off	Off and On	It is configured as "On" Automatically when the "Transport Mode" is configured as "Unicast".
19.xx13.13	Unicast Destination	(Empty)	Any IPv4 address	It can only be configured if "Transport Mode" is configured as "Unicast".
19.xx13.14	Domain	0	0 to 127	
19.xx13.20	Announce Rate	1/2	1/128 to 16	Measured in packets per second.
19.xx13.21	Sync Rate	1	1/128 to 128	Measured in packets per second.
19.xx13.22	Delay Requests Rate	1	1/128 to 128	Available if the Delay Mechanism is configured as E2E. Measured in packets per second.
19.xx13.23	Peer Delay Requests Rate	1	1/128 to 128	Available if the Delay Mechanism is configured as P2P. Measured in packets per second.
19.xx13.18	User Offset	Between -250 and -150, depending on the hardware of the device being used.	Between -(2^{31}) and (2^{31})	We advise not to change this parameter since i ts value has been calculated and tested for each device. Measured in nanoseconds.

5.4.3.2 Telecom ITU-T 8265.1

The Telecom ITU-T G.8265.1 profile supports frequency synchronization in telecommunications infrastructures, offering partial timing support for applications needing precise frequency alignment without full timing capabilities.

OID	Name	Default Value	Possible Values	Note
19.xx13.9	Transport Protocol	UDP/IPv4	UDP/IPv4	Only IPv4/UDP allowed. Anj IPv4 address must be set before selecting this profile.
19.xx13.10	Delay Mechanism	E2E	E2E	Only E2E is allowed.
19.xx13.56	Enable SyncE (Synchronous Ethernet)	No	No and Yes	
9.xx08.3	SyncE TX Quality Level	Auto	Auto and Manual	Available if the "Enable SyncE" parameter is Configured as "Yes".
19.xx13.11	Transport Mode	Unicast	Unicast and Hybrid	
19.xx13.12	Unicast Negotiation	On	On	When the "Transport Mode" is set to "Unicast," the "Unicast Negotiation" must always be configured as "True." However, this is not required when the mode is set to "Hybrid."
19.xx13.13	Unicast Destination	(Empty)	Any IPv4 address	In Slave PTP ports configured as "Unicast", the user must fill the IP of the master PTP port.
19.xx13.14	Domain	4	4 to 23	
19.xx13.20	Announce Rate	1/2	1/16 to 8	Measured in packets per second.
19.xx13.21	Sync Rate	1	1/16 to 128	Measured in packets per second.
19.xx13.22	Delay req Rate	1	1/16 to 128	Available if the Delay Mechanism is configured as E2E. Measured in packets per second.
19.xx13.23	Peer Delay Requests Rate	None	None	It cannot be configured since this profile only accepts "E2E" as "Delay Mechanism".

OID	Name	Default Value	Possible Values	Note
19.xx13.18	User Offset	Between -250 and -150, depending on the hardware of the device being used.	Between -(2^{31}) and (2^{31})	We advise not changing this parameter as its value has already been calculated and tested for each device. Measured in nanoseconds.

5.4.3.3 Power Profile: IEEE C37.238-2017

The IEEE C37.238-2017 profile is designed for precise time synchronization in power systems, typically used for timing in the operation of electrical grids.

OID	Name	Default Value	Possible Values	Note
19.xx13.9	Transport Protocol	Layer 2 (IEEE802.3)	Layer 2 (IEEE802.3)	Only Layer 2 is allowed.
19.xx13.10	Delay Mechanism	P2P	P2P	Only P2P is allowed.
19.xx13.56	Enable SyncE (Synchronous Ethernet)	No	No and Yes	
9.xx08.3	SyncE TX Quality Level	Auto	Auto and Manual	Available if the "Enable SyncE" parameter is Configured as "Yes".
19.xx13.11	Transport Mode	Multicast	Multicast	Only Multicast is allowed.
19.xx13.12	Unicast Negotiation	Off	Off	It cannot be configured since "Transport Mode" is always "Multicast" for this profile.
19.xx13.13	Unicast Destination	(Empty)	None	It cannot be configured since "Transport Mode" is always "Multicast" for this profile.
19.xx13.14	Domain	254	0 to 127, and 254	
19.xx13.20	Announce Rate	1	1	Only one packet per second can be selected. Measured in packets per second.
19.xx13.21	Sync Rate	1	1	Only one packet per second can be selected. Measured in packets per second.

OID	Name	Default Value	Possible Values	Note
19.xx13.22	Delay Requests Rate			It cannot be configured since "Delay Mechanism" is always "P2P" for this profile.
19.xx13.23	Peer Delay Requests Rate	1	1	Measured in packets per second.
19.xx13.18	User Offset	Between -250 and -150, depending on the hardware of the device being used.	Between -(2^{31}) and (2^{31})	We advise not changing this parameter as its value has already been calculated and tested for each device. Measured in nanoseconds.

5.4.3.4 Power Utility Profile: IEEE 61850-9-3

Like the IEEE C37.238-2017 profile, the IEEE 61850-9-3 is used for precise time synchronization in power systems, typically for timing transmission in electrical grids.

OID	Name	Default Value	Possible Values	Note
19.xx13.9	Transport Protocol	Layer 2 (IEEE802.3)	Layer 2 (IEEE802.3)	Only Layer 2 is allowed.
19.xx13.10	Delay Mechanism	P2P	P2P	Only P2P is allowed.
19.xx13.56	Enable SyncE (Synchronous Ethernet)	No	No and Yes	
9.xx08.3	SyncE TX Quality Level	Auto	Auto and Manual	Available if the "Enable SyncE" parameter is Configured as "Yes".
19.xx13.11	Transport Mode	Multicast	Multicast	Only Multicast is allowed.
19.xx13.12	Unicast Negotiation	Off	Off	It cannot be configured since "Transport Mode" is always "Multicast" for this profile.
19.xx13.13	Unicast Destination	(Empty)	None	It cannot be configured since "Transport Mode" is always "Multicast" for this profile.
19.xx13.14	Domain	0	0 to 127	

OID	Name	Default Value	Possible Values	Note
19.xx13.20	Announce Rate	1	1	Only one packet per second can be selected. Measured in packets per second.
19.xx13.21	Sync Rate	1	1	Only one packet per second can be selected. Measured in packets per second.
19.xx13.22	Delay Requests Rate	None	None	It cannot be configured since "Delay Mechanism" is always "P2P" for this profile.
19.xx13.23	Peer Delay Requests Rate	1	1	Measured in packets per second.
19.xx13.18	User Offset	Between -250 and -150, depending on the hardware of the device being used.	Between -(2^{31}) and (2^{31})	We advise not changing this parameter as its value has already been calculated and tested for each device. Measured in nanoseconds.

5.4.3.5 Telecom ITU-T 8275.1

The Telecom ITU-T G.8275.1 profile is designed for time and phase synchronization in telecommunications infrastructures, providing full timing support over packet networks through the use of Synchronous Ethernet.

OID	Name	Default Value	Possible Values	Note
19.xx13.9	Transport Protocol	Layer 2 (IEEE802.3)	Layer 2 (IEEE802.3)	Only Layer 2 is allowed. VLAN tags are not supported.
19.xx13.10	Delay Mechanism	E2E	E2E	Only E2E is allowed.
19.xx13.56	Enable SyncE (Synchronous Ethernet)	Yes	Yes	SyncE is required for this profile.
9.xx08.3	SyncE TX Quality Level	Auto	Auto and Manual	Available if the "Enable SyncE" parameter is Configured as "Yes".
19.xx13.11	Transport Mode	Multicast	Multicast	Only Multicast is allowed.

OID	Name	Default Value	Possible Values	Note
19.xx13.12	Unicast Negotiation	Off	Off	It cannot be configured since "Transport Mode" is always "Multicast" for this profile.
19.xx13.13	Unicast Destination	(Empty)	None	It cannot be configured since "Transport Mode" is always "Multicast" for this profile.
19.xx13.14	Domain	24	24 to 43	
19.xx13.20	Announce Rate	8	8	Measured in packets per second.
19.xx13.21	Sync Rate	16	16	Measured in packets per second.
19.xx13.22	Delay Requests Rate	16	16	Available if the Delay Mechanism is configured as E2E. Measured in packets per second.
19.xx13.23	Peer Delay Requests Rate	None	None	It cannot be configured since "Delay Mechanism" is always "E2E" for this profile.
19.xx13.18	User Offset	Between -250 and -150, depending on the hardware of the device being used.	Between -(2^{31}) and (2^{31})	We advise not changing this parameter as its value has already been calculated and tested for each device. Measured in nanoseconds.

5.4.3.6 Enterprise

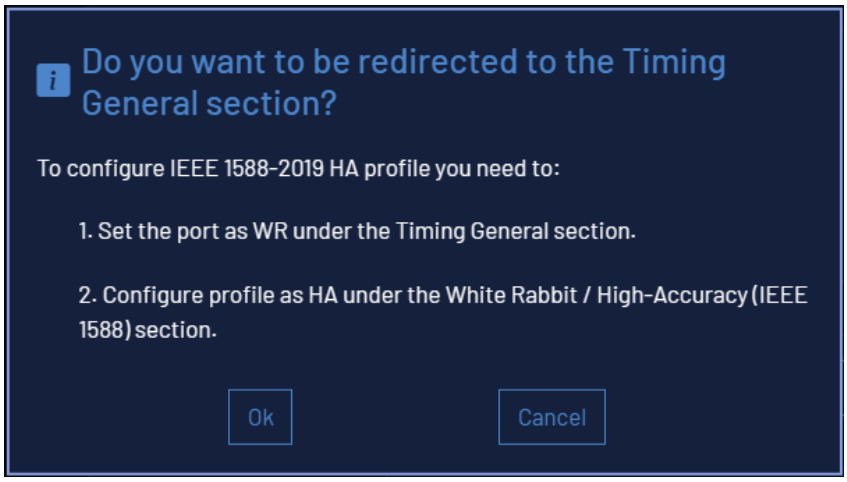
The Enterprise profile is designed for large enterprise networks to achieve interoperability between equipment manufactured by different vendors.

OID	Name	Default Value	Possible Values	Note
19.xx13.9	Transport Protocol	UDP/IPV4	UDP/IPV4	Only UDP/IPV4 is allowed.
19.xx13.10	Delay Mechanism	E2E	E2E	Only E2E is allowed.
19.xx13.56	Enable SyncE (Synchronous Ethernet)	No	No and Yes	

OID	Name	Default Value	Possible Values	Note
9.xx08.3	SyncE TX Quality Level	Auto	Auto and Manual	Available if the "Enable SyncE" parameter is Configured as "Yes".
19.xx13.11	Transport Mode	Hybrid	Multicast and Hybrid	
19.xx13.12	Unicast Negotiation	Off	Off	It cannot be configured since "Transport Mode" is always "Multicast" or "Hybrid" for this profile.
19.xx13.13	Unicast Destination	(Empty)	None	It cannot be configured since "Transport Mode" is always "Multicast" or "Hybrid" for this profile.
19.xx13.14	Domain	0	0 to 127	
19.xx13.20	Announce Rate	1	1	Only one packet per second can be selected. Measured in packets per second.
19.xx13.21	Sync Rate	1	1/128 to 128	Measured in packets per second.
19.xx13.22	Delay Requests Rate	1	1/128 to 128	Measured in packets per second.
19.xx13.23	Peer Delay Requests Rate	None	None	It cannot be configured since "Delay Mechanism" is always "E2E" for this profile.
19.xx13.18	User Offset	Between -250 and -150, depending on the hardware of the device being used.	Between -(2^{31}) and (2^{31})	We advise not changing this parameter as its value has already been calculated and tested for each device. Measured in nanoseconds.

5.4.3.7 IEEE 1588-2019 HA

The IEEE 1588-2019 HA or High Accuracy profile implements extensions to PTPv2 that allow for enhanced precision in time synchronization. It requires a valid PTP license for configuration, and upon selection a notification will be displayed with the following setup instructions.



Selecting "Ok" will redirect to **CONFIGURATION** > **Timing General**. Selecting "Cancel" will deselect the profile and close the notification.

More information on this profile can be found in "White Rabbit / IEEE 1588-2019 HA" on page 97.

5.4.4 Information Parameters

This section introduces the informational parameters for PTP and SyncE, designed to monitor the status and performance of connected master and slave ports. The information is organized into two tables: the first covers PTP parameters, and the second focuses on SyncE parameters.

OID	Parameter Name <Command Line Interface name>	Value Type	Description
19.xx12.2	Port State <port_state>	<Enum> 0. None 1. Initializing 2. Faulty 3. Disabled 4. Listening 5. Pre-Master 6. Master 7. Passive 8. Uncalibrated 9. Slave 10. Survey 11. SyncE Locking 12. SyncE Locked 13. SyncE Error 14. Disabled - Invalid Leap Secs	Indicates the current state of the port based on its role and synchronization status, providing real time insight into the operational and synchronization state of the port.

OID	Parameter Name <Command Line Interface name>	Value Type	Description
19.xx12.4	Servo State <servo_state>	<Enum> 0. Disabled 1. "----" 2. Waiting Sync Sec (+) 3. Waiting Sync Sec (-) 4. Error 5. Freq Estimation 6. Tracking 7. Not Updated 8. Locked 9. Master 10. Clock Adjust	Indicates the state of the PTP servo. For a port configured as a slave, the servo must reach the "Locked" state to ensure proper synchronization.
19.xx12.57	SyncE State <synce_state>	<Enum> 0. Disabled 1. Master Mode 2. Unlocked 3. Locking 4. Locked	Indicates the current state of the SyncE operation. The "Locked" state signifies successful frequency synchronization.
19.xx12.25	Rx Announce Packets <rx_announce>	<Integer>	Number of received announce messages (Incrementing for Slave/Passive instances)
19.xx12.30	Tx Announce Packets <tx_announce>	<Integer>	Number of transmitted announce messages (Incrementing for Master instances)
19.xx12.26	Rx Sync Packets <rx_sync>	<Integer>	Number of received sync messages (Incrementing for Slave instances)
19.xx12.31	Tx Sync Packets <tx_sync>	<Integer>	Number of transmitted sync messages (Incrementing for Master instances)
19.xx12.32	Rx Follow Up Packets <rx_follow_up>	<Integer>	Number of received Follow up messages (Incrementing for Slave instances)
19.xx12.32	Tx Follow Up Packets <tx_follow_up>	<Integer>	Number of transmitted Follow up messages (Incrementing for Master instances)
19.xx12.28	Rx Delay Request Packets<rx_ delayreq>	<Integer>	Number of received delay request message (Incrementing for Master instances). It only applies to E2E connections.
19.xx12.33	Tx Delay Request Packets <tx_delayreq>	<Integer>	Number of transmitted delay request message (Incrementing for Slave instances). It only applies to E2E connections.

OID	Parameter Name <Command Line Interface name>	Value Type	Description
19.xx12.64	Rx Peer Delay Request Packets <rx_pdelayreq>	<Integer>	Number of received peer delay request messages (Incrementing for Master instances). It only applies to P2P connections.
19.xx12.67	Tx Peer Delay Request Packets <tx_pdelayreq>	<Integer>	Number of transmitted peer delay request messages (Incrementing for Slave instances). It only applies to P2P connections.
19.xx12.29	Rx Delay Response Packets <rx_delayresp>	<Integer>	Number of received peer delay response messages (Incrementing for Slave instances). It only applies to E2E connections.
19.xx12.34	Tx Delay Response Packets <tx_delayresp>	<Integer>	Number of transmitted delay response messages (Incrementing for Master instances). It only applies to E2E connections.
19.xx12.65	Rx Peer Delay Response Packets <rx_pdelayresp>	<Integer>	Number of received peer delay response messages (Incrementing for Slave instances). It only applies to P2P connections.
19.xx12.68	Tx Peer Delay Response Pack-ets<rx_pdelayresp>	<Integer>	Number of transmitted peer delay response messages (Incrementing for Master instances). It only applies to P2P connections.
19.xx12.66	Rx Peer Delay Response Follow Up Packets <rx_pdelayrespfup>	<Integer>	Number of received peer delay response follow-up messages (Incre-menting for Slave instances). It only applies to P2P connections.
19.xx12.69	Tx Peer Delay Response Follow Up Packets <rx_pdelayrespfup>	<Integer>	Number of transmitted peer delay response follow-up messages (Incre-menting for Master instances). It only applies to P2P connections.
19.xx12.35	Offset from Master <offset_from_mas-ter>	<Decimal> (f64)	The value represents the actual off-set (in seconds) between master and slave. It is calculated from the retrieved timestamps with cor-rections from the PTP calibration.
19.xx12.36	One-Way Delay <one_way_delay>	<Decimal> (f64)	Average one-way path delay in seconds of the PTP packets.
19.xx12.55	Mean Delay <mean_delay>	<Decimal> (f64)	Average Round Trip Time (RTT) of the PTP packets.

OID	Parameter Name <Command Line Interface name>	Value Type	Description
9.xx09.0	SyncE State <synce_state>	<Enum> 0: Disabled 1: Master Mode 2: Unlocked 3: Locking 4: Locked	Indicates the current state of the SyncE (Synchronous Ethernet) operation for the port.
9.xx09.1	EEC State <eec_state>	0: Unknown 1: Invalid 2: Free Running 3: Locked 4: Locked. Holdover is ready 5: Holdover	Represents the current status of the Ethernet Equipment Clock (EEC) for the device.
9.xx09.2	SSM QL RX <ssm_ql_rx>	<Enum> 0: Not initialized 2: PRC 4: SSU A 8: SSU B 11: SEC 15: DNU	Displays the Synchronization Status Message (SSM) Quality Level (QL) code received by the port.
9.xx09.3	RX Packets <rx_packets>	<Integer>	Indicates the count of Synchronization Status Message (SSM) Quality Level (QL) packets received by the port
9.xx09.4	TX Packets <tx_packets>	<Integer>	Indicates the count of Synchronization Status Message (SSM) Quality Level (QL) packets transmitted by the port.

5.4.4.1 Command Line Interface for PTP and SyncE

PTP and SyncE can be managed via the Command Line Interface (CLI). This section includes commands and examples for configuring PTP profiles, adjusting SyncE parameters, and retrieving status and informational data for both protocols.

A few commands allow management and monitoring of PTP and SyncE through the CLI for each port. To configure or access PTP parameters for a specific port number <X>, use the command `gpa_ctrl wptpd` followed by `net/wr<X>/1/cfg/` for configuration parameters or `net/wr<X>/1/info/` for informational parameters. Similarly, for SyncE, use the command `gpa_ctrl esmcd` followed by

`net/wr<X>/cfg/` for configuration parameters or `net/wr<X>/info/` for informational parameters.

In the tables of configuration and informational parameters (see "Configuration Parameters" on page 105), you will find the CLI name for each parameter enclosed in < >. For example, to configure the profile parameter, the table lists it as:

Profile
`<profile>`

The image below displays the complete WR0 list of parameters (both configuration and informational).

```
root@z16-216:~# gpa_ctrl wptpd net/wr0
---            wptpd ------------------------------------------------------|
   19.5011.1  net/wr0/1/wproc/state               : Running
   19.5011.5  net/wr0/1/wproc/run_stop            : 1
   19.5012.1  net/wr0/1/info/ifname               : wr0
   19.5012.2  net/wr0/1/info/port_state           : Master
   19.5012.4  net/wr0/1/info/servo_state          : Master
   19.5012.57 net/wr0/1/info/synce_state          : Disabled
   19.5012.25 net/wr0/1/info/rx_announce          : 1346
   19.5012.26 net/wr0/1/info/rx_sync              : 2686
   19.5012.27 net/wr0/1/info/rx_follow_up         : 2686
   19.5012.28 net/wr0/1/info/rx_delayreq          : 0
   19.5012.29 net/wr0/1/info/rx_delayresp         : 0
   19.5012.30 net/wr0/1/info/tx_announce          : 0
   19.5012.31 net/wr0/1/info/tx_sync              : 0
   19.5012.32 net/wr0/1/info/tx_follow_up         : 0
   19.5012.33 net/wr0/1/info/tx_delayreq          : 0
   19.5012.34 net/wr0/1/info/tx_delayresp         : 0
   19.5012.35 net/wr0/1/info/offset_from_master   : 0.000000000          s
   19.5012.36 net/wr0/1/info/one_way_delay        : 0.000000000          s
   19.5012.55 net/wr0/1/info/mean_delay           : 0.000000000          s
   19.5013.7  net/wr0/1/cfg/mode                  : Masteronly
   19.5013.8  net/wr0/1/cfg/profile               : Default
   19.5013.9  net/wr0/1/cfg/transport_protocol    : Layer2
   19.5013.10 net/wr0/1/cfg/delay_mech            : E2E
   19.5013.11 net/wr0/1/cfg/transport_mode        : unicast
   19.5013.56 net/wr0/1/cfg/enable_synce          : No
   19.5013.12 net/wr0/1/cfg/unicast_neg           : On
   19.5013.13 net/wr0/1/cfg/unicast_dest          :
   19.5013.14 net/wr0/1/cfg/domain                : 0
   19.5013.18 net/wr0/1/cfg/user_offset           : -244                 ns
   19.5013.20 net/wr0/1/cfg/announce_rate         : 1 packet/2s
   19.5013.21 net/wr0/1/cfg/sync_rate             : 1 packet/s
   19.5013.22 net/wr0/1/cfg/delayreq_rate         : 1 packet/s
   19.5015.20 net/wr0/1/clkinfo/Q/n_hops          : 0
```

If you want to access information (ranges, type, description) of any parameter, you can type "-i." at the end of any of them, as shown in the image below:

```
root@z16-216:~# gpa_ctrl wptpd net/wr0/1/info/rx_announce -i.
oid: 19.5012.25
path: net/wr0/1/info/rx_announce
value: 0
status:
type: u64
access: R
unit:
enum: {}
description: Number of received announce messages (Incrementing for Slave
Passive instances)
```

Examples

To read and write these parameters, you can follow the examples below.

To read the profile configuration of port number 0, enter:

```
gpa_ctrl wptpd net/wr0/1/cfg/profile
```

To configure port number 1 with the Telecom 8275.1 profile, enter:

```
gpa_ctrl -s wptpd net/wr1/1/cfg/profile 5
```

To modify the two configuration parameters of SyncE, follow the same process as for PTP, but replace wptpd with esmcd. For example, to set the SSM QL Mode to Manual on port wr0, enter:

```
gpa_ctrl esmcd net/wr0/cfg/ssm_ql_mode Manual
```

The same applies to the information parameters. You can access the information parameters from the tables in "Information Parameters" on page 117 as in the following examples.

For reading how many announces packets we received in the port WR1, enter:

```
gpa_ctrl wptpd net/wr1/1/info/rx_announce
```

To read the received SyncE state of the port WR0, enter:

```
gpa_ctrl esmcd net/wr0/info/synce_state
```

5.4.4.2 PTP Web UI Configuration and Overview

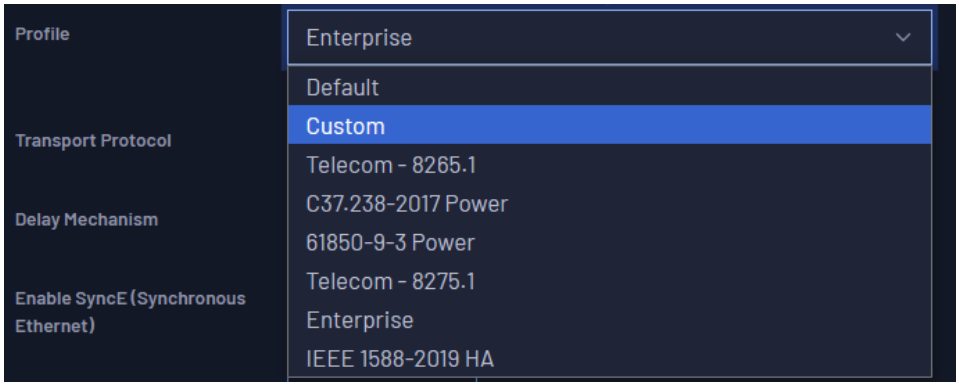
The Web UI (Web User Interface) provides a user-friendly platform for managing PTP and SyncE settings. The **Configuration** dropdown menu contains a list of pages that allow you to adjust parameters and set up profiles for both PTP and SyncE, enabling easy customization of synchronization settings. The **Overview** dropdown menu contains a list of pages that can display the current status and

detailed information about PTP and SyncE, helping you monitor and verify synchronization performance.

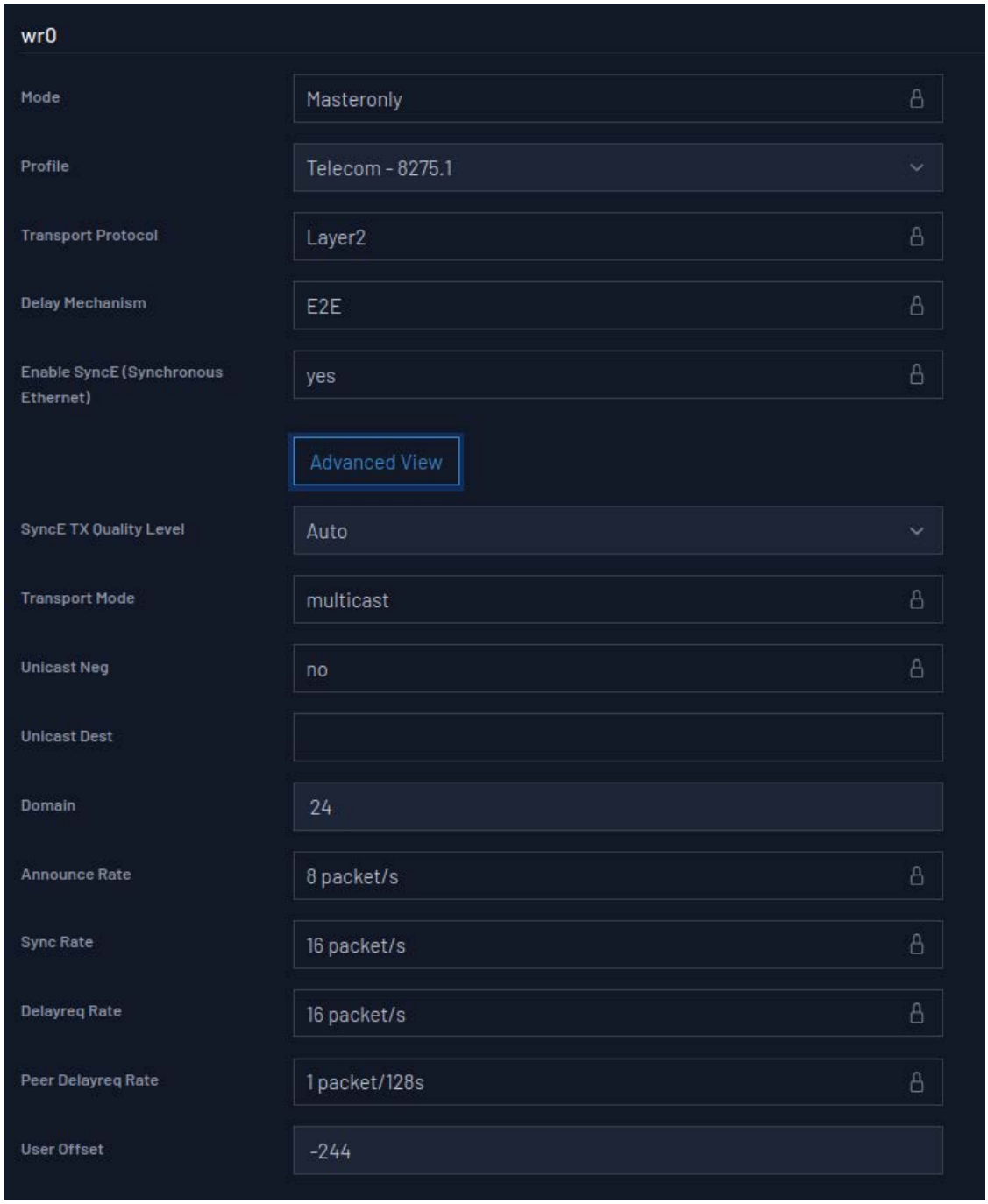
Configuration

To configure PTP, navigate to **CONFIGURATION** > **IEEE 1588-2008 (PTPv2).**

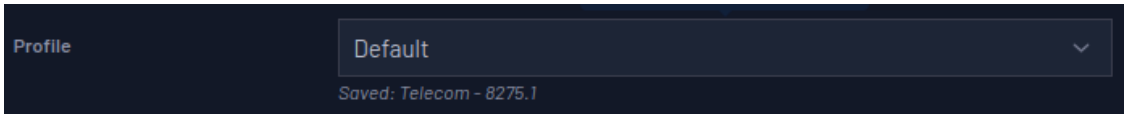
On this page, you can apply and save all PTP and SyncE configurations for all device ports. For instance, the Profile dropdown menu allows you to select from the profiles detailed in "Profiles" on page 108, as illustrated in the Image below.



On this page you can configure the PTP parameters for each port. For example, in the image below you can view the configuration section of the WR0 port.

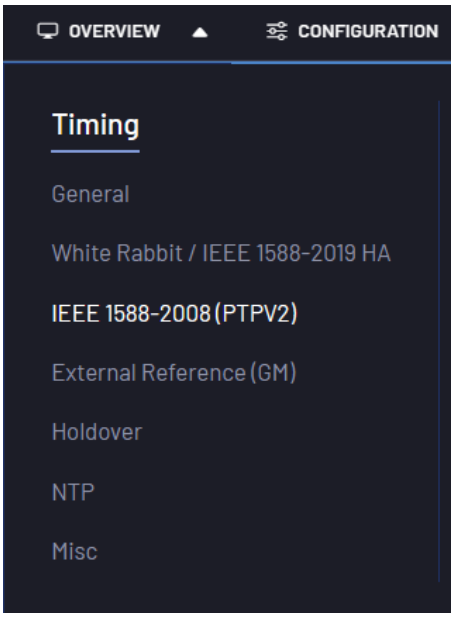


If a configuration does not match its saved value, the Web UI will display a notification similar to the following:



Overview

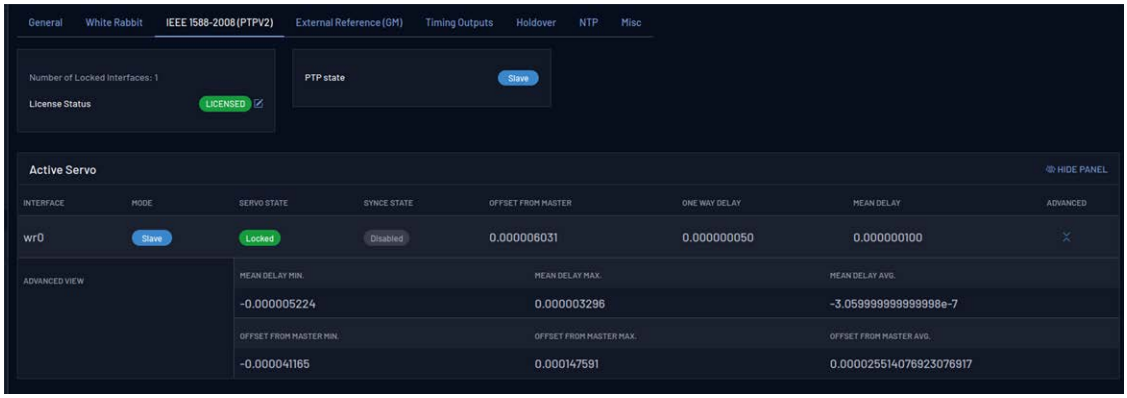
To access the PTP and SyncE Overview page, navigate to **OVERVIEW** > **IEEE 1588-2008 (PTPv2).**



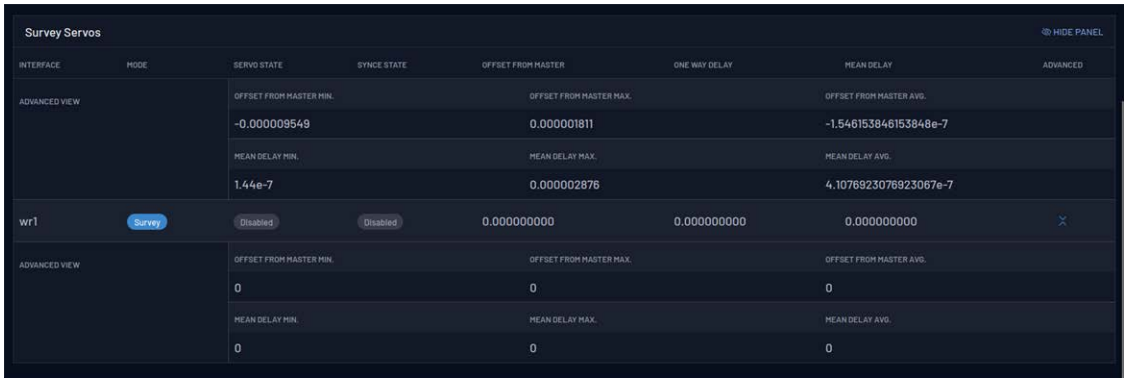
The information available to the user on this page, listed from top to bottom, is as follows:

The **device drawing** shown below displays the connections currently established within the device.

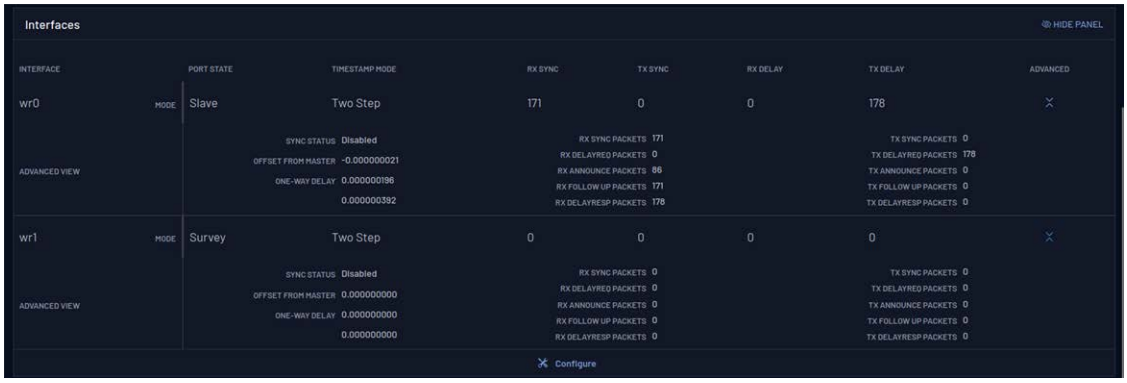
The **IEEE 1588-2008(PTPv2)** tab displays the PTP License Status, the device's PTP configuration (Slave in the example below), and a summary of the device's Active Servo. The Active Servo refers to the port configured as a Slave on the device. This panel provides details such as the Servo State, SyncE State, offset from the master, and additional relevant information.



The **Survey Servo** panel provides details about ports with the Survey configuration. Survey ports exchange PTP packets but do not synchronize the device's clock. Instead, they allow the device to gather information about other PTP master clocks without actively locking to them.



The **Interfaces panel** provides information about all PTP interfaces. It includes a summary of details such as the number of packets exchanged for each type, the offset from the master, and the configuration of each port (Slave, Master, Survey, or Disabled), among other parameters.



5.5 External Reference (GM)

5.5.1 Configuration

The Configuration of the GM is partially done by the Preset and then by the Configuration Tab under the Timing > External Reference (GM) section.

The user will be able to configure the External Reference (GM) parameters by accessing the External Reference (GM) section. A table containing the configurable parameters is shown below: :

OID	Name	Value Type	Description
3.7110.x	**gm/cfg/xxx**		**Configuration of the GM timing source (By Preset).**
3.7110.0	Source Rank	<Integer> Default: 0	GM priority as Timing Source
3.7110.1	GM Offset	<Integer> Default: 0	Offset to compensate user cable delay for PPS input (in pico-seconds). When Align PPS is enabled the PPS output should be aligned to the PPS input but the user might want to compensate this delay.
3.7110.2	Align PPS	<Boolean> Default: OFF	Enable this to align the PPS output to the PPS input during the locking procedure. It should be enabled when using a GNSS receiver as external reference as PPS might be shifted from 10MHz after each GNSS relock.
3.7110.3	Clock Accuracy	<Enum> Default: UNKNOWN	Announces the expected accuracy provided by the external reference. Conservatively estimated based on the type of the timing source (e.g., Atomic clock <= 1ns, GNSS receiver <= 50ns)
3.7110.4	Time Source	<Enum> Default: OTHER • ATOMIC CLOCK • GNSS • TERRESTRIAL RADIO • PTP • NTP • HAND SET • OTHER • INTERNAL OSCILLATOR	Type of timing source announced by the GM. It should correspond to the type of external reference that provides 10MHz/PPS to the front-panel of the device. (This field is informative and not used for decision making).

OID	Name	Value Type	Description
3.7110.5	Priority 1	<Integer> Default: 128	PTP priority1 announced when the GM is active. It is mainly used by BMCA to force the best-clock selection using 1st priority (Lower values take precedence)
3.7110.6	Priority 2	<Integer> Default: 128	PTP priority2 announced when the GM is active. It is mainly used by BMCA to force the choice between two references when their clock qualities are the same (Lower values take precedence)
3.7110.7	PPS Mandatory	<Enum> Default: YES • YES • NO • STARTUP ONLY	Controls whether an PPS input signal is needed to enter/stay active for the GM source.
3.7110.15	Leap Second File Ignore	<Enum> Default: OFF • OFF • ON	Leapsec_file_ignore disables the critical warning when the leap second file is expired.
3.7110.16	Mode	<Enum> Default: SLAVE • SLAVE • SURVEY	Determine the GM mode: Slave (port can be used as timing source) or Survey (only report the time offset of the source port)

Caution: If GM is used as a timing source, it should always be associated to the configuration of at least one NTP server to properly recover the time of day (ToD).

5.5.2 Info/Overview

The GM timing source provides its own overview panel under **Overview** > **Timing** > **External Reference (GM)**, where the user can easily audit the condition of its external reference (see figure below).

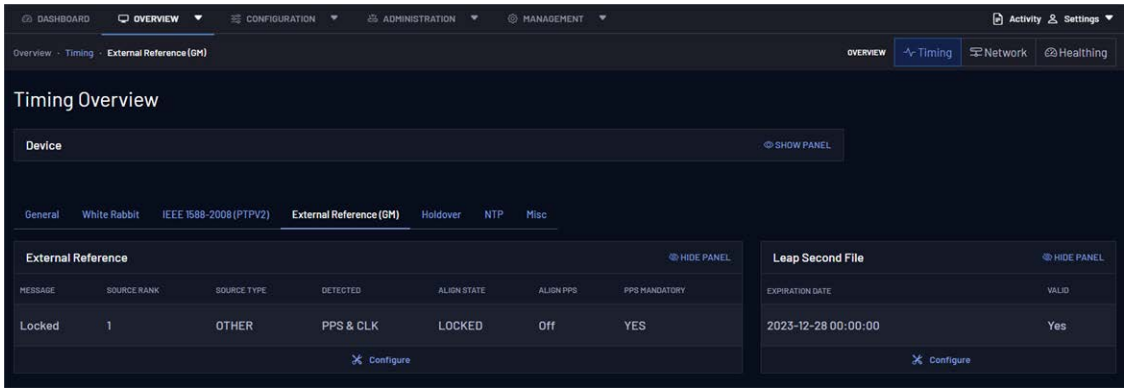


Figure 5-37: Overview tab for GM timing source.

It basically offers a readback of the configuration value (Source Type, Source Rank, Align PPS and PPS Mandatory) as detailed "Configuration" on page 128, along with a user friendly message that summarize the state of the GM internal state machine and a Detected value that reports the situation with the external reference inputs signals. Finally, the validity of the leap second file needed to perform the conversion from UTC (NTP timescale) to TAI (PTP timescale) is detailed in the panel. A detailed explanation of the parameters is provided below:

OID	Name	Value Type	Description
3.7120.x	**gm/info/xxx**		**Specific information about the state of GM timing source.**.
3.7120.0	Message	<String> Default: UNINITIALIZED.	User friendly message that summarizes the current state of the GM timing source.
3.7120.2	PPS Detected	<Enum> Default: NONE • NONE • PPS (Only) • CLK (Only) • PPS & CLK	Report the detection of external reference input on the front panel.
3.7120.4	Leap Second File Expiration Date	<String> Default: Depends on file.	Expiration date of the leap seconds files. If there is more than one file, this will show the date that is further in the future. The date format is YYYY-MM-DD HH:MM:SS.
3.7120.5	Leap Second File Validity	<Bool> Default: YES • No • Yes	“Yes” if the leap seconds file is valid. “No” if it is expired or missing. See "Update Leap Seconds File" on page 142 for updating this file.

5.6 NTP

This section is about the configuration and monitoring related to the NTP protocol.

Note: Periodic pooling of NTP offset: In the current version of WRZ-OS it is recommended to set the NTP server in every node of the topology, either GM or BC, to check the coherence of the timing reference.

5.6.1 Configuration

The WRZ-OS supports NTP over management interfaces (ethX) and fiber optics ports (wrX). NTP configuration is described in the following sections.

In the CLI, NTP is configured using the `gpa_ctrl` command. Instructions for Web UI configuration are described in the sections below.

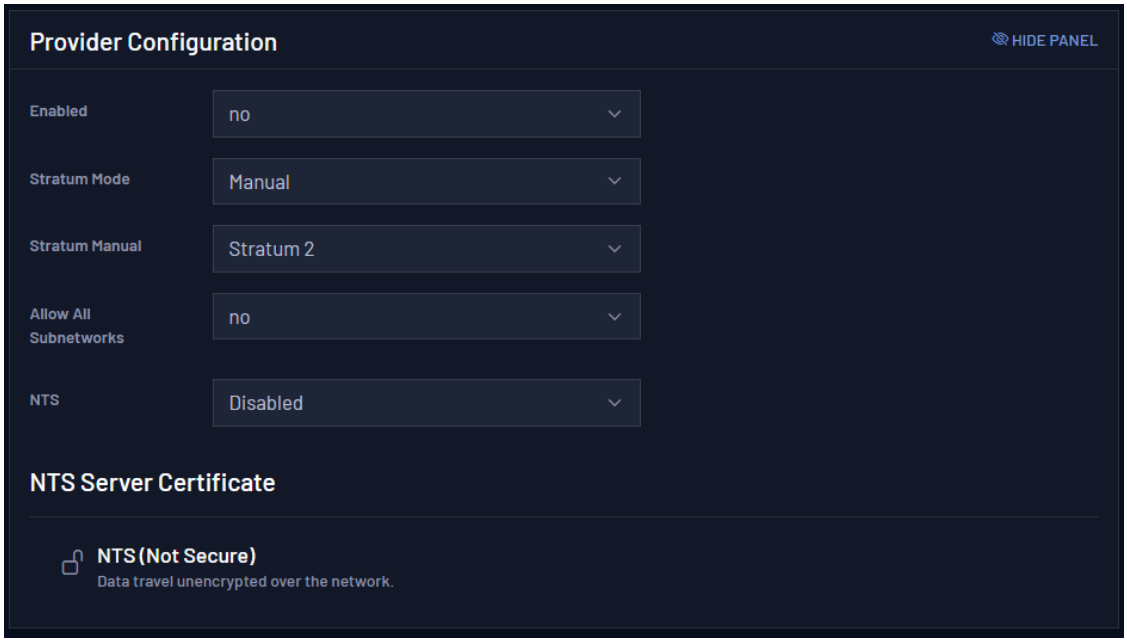
5.6.1.1 NTP Provider

The WRZ-OS allow the device to provide its time through NTP on the management interfaces using the following parameters:

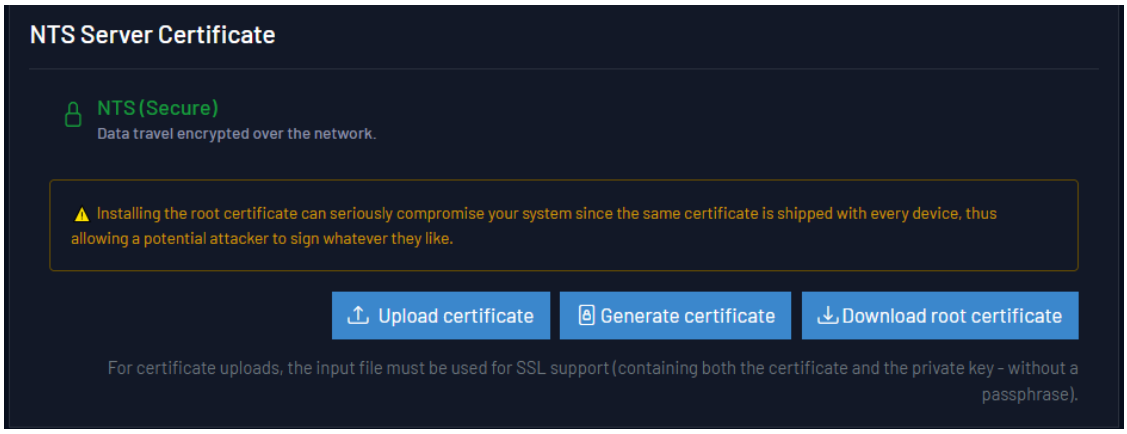
OID	Name	Value Type	Description
3.7005.x	**ntp/cfg/provider/xxx**		**Configuration on how to provides NTP to 3rd party devices.**
3.7005.1	Enabled	<Boolean> • YES • NO	If 'Yes' the device will act as an NTP server on its management interfaces to distribute its own Time of Day to other devices.
3.7005.2	Stratum Mode	• Manual	Mode to provide the NTP stratum. If Manual it will directly set the value from 'Manual Stratum', otherwise it will take into account the virtual clock quality (timing source, clock accuracy, etc.) to modify this value.
3.7005.3	Manual Stratum	• Stratum 1 • Stratum 2 • Stratum 3 • Stratum 4 • Stratum 15	Manually force the Stratum announced by the NTP server. See "Stratum Levels" on page 136 for more information
3.7005.4	Allow All Subnets	<Boolean> • NO • YES	If 'Yes' the device will accept NTP requests from any subnet; otherwise it will only accept NTP requests from explicitly allowed subnets.

OID	Name	Value Type	Description
3.7005.5	NTS Enabled	<Boolean> • Disabled • Enabled	If 'Enabled' NTP will use NTS (Network Time Security).

The Web UI settings for NTP Provider are located at**Configuration**>**NTP**>**Provider Configuration**.



If NTS is Enabled on this panel, an additional section will display with options to install an NTS Server Certificate:



> **Caution:** Reboot to apply NTP provider parameters, as many of the settings to configure the NTP server require a Web UI reboot (from the CLI, the command `gpa_ctrl tmgrd cfg/update_config on` will also restart the timing service and apply NTP changes).

5.6.1.2 NTP over Fiber Optics (wrX ports)

To configure an NTP server over a fiber optics port, set the particular wrX port protocol as NTP (see "Custom" on page 84 for information on the Custom Preset).

> **Caution:** NTP time is distributed over UDP and thus it requires the interface to have an IP and subnet configured on the subnet where NTP is going to be served.

The NTP Stratum can be configured using the table in the previous section.

5.6.1.3 NTP Timing Source Configuration

The configuration of an NTP timing source is divided in two parts. The first panel provides the configuration shared by all the NTP timing sources:

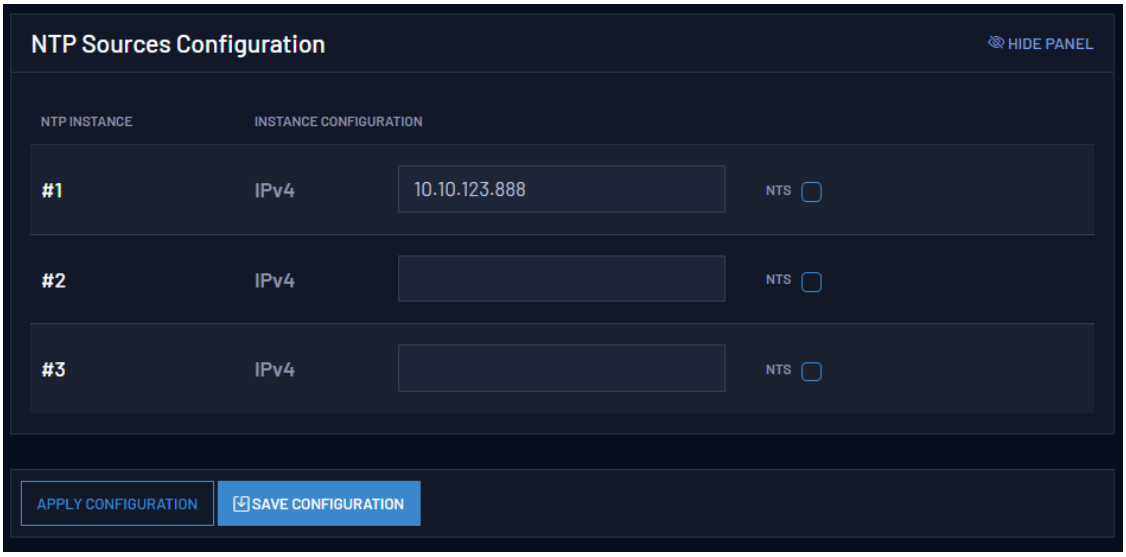
OID	Name	Value Type	Description
3.7001.x	**ntp/cfg/xxx**		**Configuration on how to provides NTP to 3rd party devices.**.
3.7001.1	Poll Rate	<Integer> • Default: 32 s/query • Positive exponent of 2	Time between NTP server queries is 2^poll_rate. E.g.: if the poll rate is 5, the interval will be 32 seconds.
3.7001.3	Retries	<Integer> • Default: 5	Number of retries for NTP server queries performed at the initialization of the device.

In the Web UI, these settings are located in **Configuration**>**NTP**>**General Configuration**.

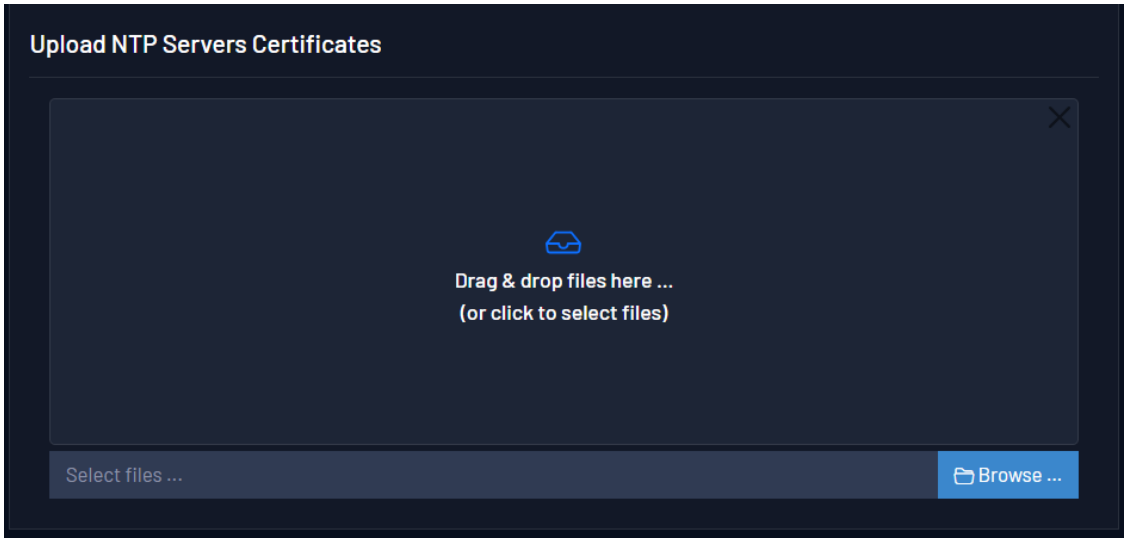
The second panel is a table representing up to 3 NTP timing sources:

OID	Name	Value Type	Description
3.70x0.x	ntp/x/cfg/xxx		Configuration of the xNTP timing source (x= [1-3]).
3.70x0.0	IPv4 Server	<String>	IP or URL of the reference NTP server.
3.70x0.1	NTS Server	<Boolean> • YES • NO	If 'Yes' NTS shall be used for this server.

In the Web UI, these settings are located in**Configuration**>**NTP**>**NTP Sources Configuration:**



If the NTS checkbox is selected for any of the 3 NTP Servers, an additional panel will display to allow certificate upload:



Once NTS certificates are entered, a Delete All Trusted Certificates button will also be displayed.

Note: NTP Passive Timing Source: Due to its poor performance NTP timing sources are always forced to be "Passive Only". However, adding them to the configuration will provide a more robust solution as they can be used to cross-validate the active timing source.

5.6.2 Info/Overview

This panel provides an overview of the status for each NTP instance:

OID	Name	Value Type	Description
3.70x5.x	**ntp/x/info/xxx**		**Information of the xth NTP timing source (x=[1-5]).**
3.70x5.0	Offset	<Integer>	Time offset between the device and the NTP reference server (in seconds).
3.70x5.1	Server Status	• Disabled • OK • NTP sync error • NTP stopped replying	NTP server status. Warns if the NTP server cannot be reached.

OID	Name	Value Type	Description
3.70x5.2	Stratum	• Stratum 0 • Stratum 1 • Stratum 2 • Stratum 3 • Stratum 4 • Stratum 15 • Undefined	Stratum announced by the corresponding NTP server.
3.70x5.3	Standard Deviation	<Integer>	Standard deviation of the time offset between the device and the NTP server (in seconds)

5.6.3 Stratum Levels

The NTP stratum is a measure for synchronization distance from the reference clock which might not always reflect the timing performance such as jitter or delay. In other words, a server synchronized to a stratum (n) server will be running at stratum (n+1) where the upper limit for stratum is 15.

- Stratum 0: Corresponds to the reference clock sources that relays Coordinated Universal Time (UTC). Stratum 0 servers should only be deployed within a metrology institute and must not be available on the internet.
- Stratum 1: Corresponds to the servers that are directly synchronized to stratum 0. They can also be considered a Primary Reference Source (PRS) such as calibrated GNSS receiver or Atomic Clocks. The Grand-Master node is typically connected to an external reference that provides NTP with Stratum 1.
- Stratum 2: They are synchronized by a stratum 1 clock. It is the default stratum level when NTP provider is set in manual mode.
- Stratum 3: They are synchronized by a stratum 2 clock.
- Stratum 4 and below: Devices that announce this level should only be used for cross-validation or backup but not as the primary NTP reference to synchronize a 3rd party device.
- Stratum 15: It is the last valid stratum level defined by the NTP protocol.
- Stratum 16: It is commonly used to indicate that the device is not synchronized and thus does not provide any valid NTP time.

5.6.4 Customizing Chrony Configuration

To customize the configuration of Chrony and make your changes persistent, please follow these steps:

1. Create a folder in the persistent storage by executing the following command:

   ```
   mkdir -p /media/data/usr/local/etc/chrony
   ```

2. Copy the standard Chrony configuration file into the newly created folder:

   ```
   cp /etc/chrony/chrony.conf /media/data/usr/local/etc/
   ```

3. Open the copied file with your preferred text editor (e.g., vim):

   ```
   vim /media/data/usr/local/etc/chrony.conf
   ```

   Upon the next reboot, and as long as it is not removed, the file you just created will overwrite the default configuration file.

4. Once you have made the necessary changes to the configuration file, reboot the machine to ensure that Chrony loads the new configuration.

5.6.4.1 Example Directives

Here are a couple of directives that can be used as examples for customization. For more information about these and other directives, please refer to the Chrony documentation.

Allowing Access to NTP from Arbitrary Subnets

By default, when an interface is configured as an NTP server, all requests originating from the subnet of that interface are allowed. However, it is possible to allow other subnets or even allow all subnets.

To allow all subnets, add the following "allow" directive to the configuration file:

```
allow all
```

To allow an arbitrary subnet, such as 1.2.0.0/16, add the following "allow" directive:

```
allow 1.2.3.0/24
```

Please note that Chrony operates over IP/Layer3, so IP routing rules apply. Remember to reboot the machine after making changes to the configuration file.

After rebooting, you can perform a server-side check to test if a particular IP would be allowed with the current configuration. To do this, follow these steps:

1. Access the Chrony management console by executing the following command:

   ```
   chronyc
   ```

2. From the chronyc console, run the access check command, replacing <IP> with the IP address you want to test:

   ```
   accheck <IP>
   ```

For example:

```
chronyc> accheck 192.168.100.58
208 Access allowed
chronyc> accheck 192.180.1.24
209 Access denied
```

Limiting Server Response Rate to Clients

Another useful directive is "ratelimit," which helps reduce network traffic from misconfigured or broken NTP clients that poll the server too frequently. The "ratelimit" directive limits the server's response rate to individual IP addresses. Here's an example of using this directive:

```
ratelimit interval 2
```

This configuration sets the minimum interval between responses, defined as a power of 2 in seconds.

5.7 Holdover

The WR-Z16 can be ordered with an optional holdover oscillator (OCXO) in order to ensure an accuracy of 1.5µs even after 24 hours.

If this holdover oscillator is detected, it will be automatically enabled as a timing source. Otherwise, when UNAVAILABLE, the device will announce itself directly as Free-running with UNKNOWN accuracy.

When the holdover is detected and enabled it can go through the following states:

1. **Locking:** During a minimum amount of time the holdover needs to perform a rough and quick learning on a stable clock reference before using it.
2. **Learning:** In order to maintains the best accuracy during enough time the holdover is learning about its environment using adaptative algorithms. This learning period has been set to 3 days in order to ensure to fulfill the accuracy specifications.
   - » If the holdover is triggered before this learning time, it will directly enter the expired state.
3. **Ready:** Once the HO has learned enough time to ensure good performance, the HO will be ready to be triggered at any moment (it will continue learning to slightly improve its performance).

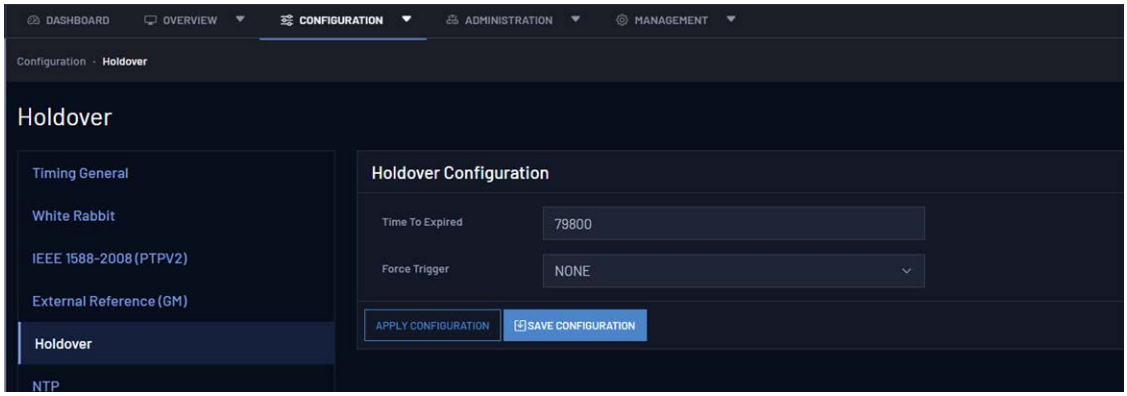
4. **Activated:** The holdover has been triggered (by trigger_origin) and it is actually being the active timing source of the device. The clock info will be modified accordingly and announced to the timing network.
5. **Expired:** Reaching the holdover expired state means that the device announce itself with a Free-Running clock_class and a clock_accuracy to UNKNOWN. This also means that the corresponding VSC code is CRITCAL and thus if a better timing source is detected it will switch to this one. Worth mentioning that during the expired state, the holdover timing source is using the OCXO oscillator that provides better performance than the internal onboard oscillator.

Note: Holdover and FOCA: As mentioned above FOCA only switches between timing source when a failure is detected. This mean that if the active timing source of a device is HO, it will stick to it until reaching the expired states (Failure state of HO).

Note: GNSS reference to discipline the HO: The holdover adaptative algorithms have been optimized to learn from a GNSS reference clock (GPS L1 signals). A better clock can be used as reference (e.g., Atomic Clock, ePRTC, multi bands/constellations GNSS receiver) but using a clock reference with worth performance might not fulfil the provided specifications.

5.7.1 Configuration

The configuration of the holdover is easy and can be leaved untouched. However, depending on the user needs and the boundaries for timing accuracy, the Time to expired value should be adjusted to meet its specifications. In the Web GUI, this can be adjusted by navigating to Configuration > Holdover.



OID	Name	Value Type	Description
3.7210.x	**holdover/cfg/xxx**		**Configuration for the Holdover if available**
3.7210.0	Source Rank	<Integer> Default: 0	Source rank of the holdover as timing source. If leaved at zero it will be always placed as the last timing source. Then the user can allow to trigger the holdover between W
3.7210.1	Time to expire	<Integer> Default: 79800	Time until the holdover is considered out of specification and expired (default ~24h)
3.7210.2	Force Trigger	<Enum> 0. STOP 1. START 2. NONE	Force to manually trigger the holdover (START) or to expire it (STOP) without waiting the expiration timer. NONE, does nothing.

Note: HO in between timing source: Using the source rank the HO can be placed between two timing sources. For example, the user can use WR as primary timing source, HO as secondary and PTP 8265.1 as third one. This means that if WR fails, the device will enter in HO until expiration and finally switch to PTP that might provide better accuracy than an expired HO.

5.7.2 Info/Overview

An overview of the holdover timing source is provided to monitorize its state at any time as shown in "Holdover overview" on the facing page.

If its state is UNAVAILABLE this means that the holdover oscillator has not been detected and its related information in irrelevant. If you have ordered the holdover option but the device does not detect it, please contact "Technical Support" on page 232.

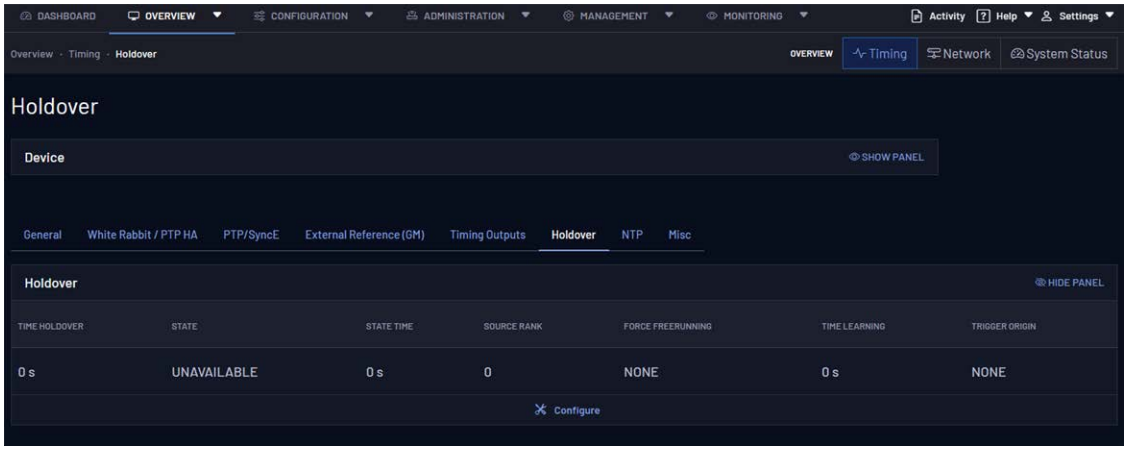


Figure 5-38: Holdover overview

OID	Name	Value Type	Description
3.7220.x	**holdover/info/xxx**		**Information about the Holdover timing source**
3.7220.2	State	<Enum> 0. UNAVAILABLE 1. DISABLED 2. LOCKING 3. LEARNING 4. READY 5. ACTIVATED 6. EXPIRED	Current state of the holdover timing source as explained in the introduction of the section. If the holdover is not detected the corresponding state if UNAVAILABLE. The user can also manually force it as DISABLED in case to avoid triggered it.
3.7220.0	Time Learning	<Integer>	Time the holdover has been in LEARNING state (in seconds)
3.7220.1	Time Holdover	<Integer>	Seconds elapsed since holdover activation
3.7220.2	Trigger Origin	<Enum> • NONE • MANUAL • PPS_DRIFT • TRACK_LOST • LINKDOWN • EXTCLK_DOWN • EXTPSS_DOWN • CLK_DRIFT	Trigger origin of last one launched

Note: Holdover and FOCA: As mentioned above FOCA only switches between timing source when a failure is detected. This means that if the active timing source of a device is HO, it will stick to it until reaching the expired states (Failure state of HO).

5.8 Miscellaneous

This section, found by navigating to **Configuration** > **Misc** in the Web GUI, allows to configure various settings that do not fit in any previous categories. The parameters can be seen below:

OID	Name	Value Type	Description
3.8010.x	**misc/cfg/xxx**		**Miscellaneous Timing configuration.**

OID	Name	Value Type	Description
3.8010.0	Time Zone	<String>	Configure the device time zone such that local time is properly displayed (web interface, LCD screen).
3.8010.0	PPS Mode	• Always ON • Only Locked • Legacy	Configurable mode to control the PPS output where: - PPS is always output even if CRITICAL (Always ON). - PPS is only output if the active reference is locked. - PPS follows the same behavior as in the legacy release (wr-zynq-os-v2.x).
3.8020.x	**misc/info/xxx**		**Miscellaneous Timing information.**
3.8020.1	Uptime	<Integer> (u64)	Time Manager uptime in seconds.

5.8.1 Update Leap Seconds File

Besides using 10MHz & PPS signals from the front-panel, the GM time source needs to obtain the Time of Day (ToD) from an external reference. NTP is commonly used because of its easy configuration. However, the leap seconds must be properly handled. Indeed, NTP is based on UTC timescale whereas PTP is based on TAI and thus the non-fixed offset between UTC-TAI is provided by the leap second file which varies according to earth rotation.

This file (also known as Bulletin C) is published by the International Earth Rotation and Reference Systems Service (IERS) every six months to tell if a leap second jump is scheduled for the end of next June or December, or not. This also means that the file shipped within the release has an expiration date and does not guarantee a valid UTC-TAI conversion after this date.

In order to always ensure a correct UTC-TAI correction, the device that can act as Grand-Master on the network, can also manually update this file by navigating to **Management** > **NTP Leap Second File**:

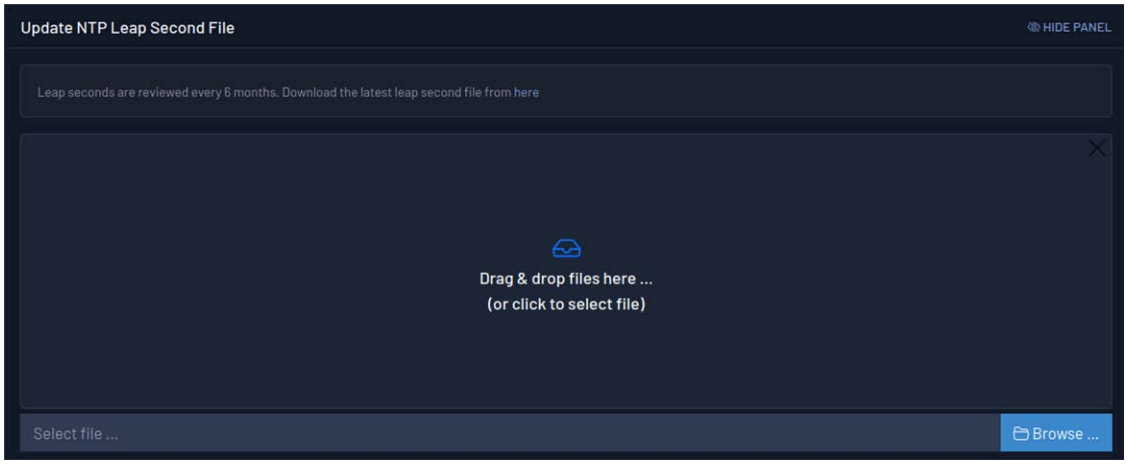


Figure 5-39: Manual Leap seconds update.

Note: Updating the leap seconds file can be performed hot by applying the configuration, so there is no need to reboot.

BLANK PAGE.

CHAPTER 6

Security & Authentication

The WR-Z16 incorporates several mechanisms in order to provide enhanced security to the system. TACACS+ and RADIUS are integrated to enable remote authentication for network access control through a centralized server. Additionally, the secure version of the network protocols used in the system are implemented, i.e. SCP, HTTPS, SNMPv3, and a firewall is included to provide a robust system against malicious users.

The following topics are included in this Chapter:

6.1 Upload SSH keys

The first time a device is accessed via SSH by a host, its IP should be added to the known hosts list as illustrated below. Then the password corresponding to the root user will be asked (default password is 'root' as detailed in "Default Configuration" on page 20).

```
ssh root@192.168.7.35
```

```
The authenticity of host '192.168.7.35 (192.168.7.35)'
can't be established. ECDSA key fingerprint is
SHA256:YgGTNfRPHYH4ekrJxDSHK7D7PiD+llHUy7dv+7460dSs.
```

```
Are you sure you want to continue connecting (yes/no)? Yes
```

```
Warning: Permanently added '192.168.7.35' (ECDSA) to the
list of known hosts.
```

```
Welcome to WR-Z16 board
```

```
Password:
```

```
root@z16-005:~#
```

This authentication procedure will only need to be confirmed the first time and will not be asked in the later connections.

```
ssh root@192.168.7.35
```

```
Welcome to WR-Z16 board
```

```
Password:
```

```
root@z16-005:~#
```

In order to improve security, it is strongly recommended to upload your public key to the device instead of using a password. This can easily be done by running the command:

```
ssh-copy-id root@<device_ip>
```

This setting is also available in the Expert mode of the GUI under Security > Authentication > SSH public key only (disable password). Choose Yes in the field, Save, and Reboot your device to activate changes.

6.2 HTTPS

Hypertext Transfer Protocol over TLS (HTTPS) is the encapsulation of HTTP over a Transport Layer Security (TLS) secured channel, which is the primary protocol used to send data between a web browser and a website.

The WR-Z16 includes the possibility of activating HTTPS. This can be done from the web interface by following the next steps:

The options about HTTPS can be accessed under **Administration** > **Security** > **HTTP/HTTPS Configuration** as shown in the figure below:

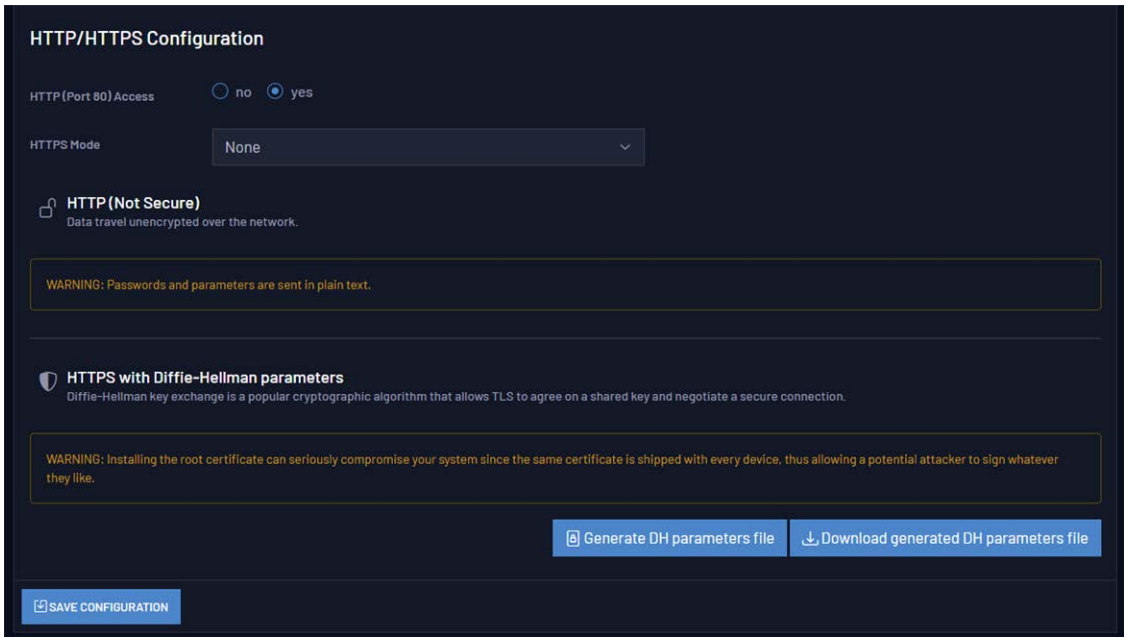


Figure 6-1: Security-HTTP/HTTPS menu of Web Interface.

The option selected by default is HTTP. While this option is active, the contents are transmitted in plain text.

In order to use the secure mode, it is necessary to either use an already-existing certificate or generate a new one. In the **HTTPS Mode** drop-down menu, select Uploaded to upload a certificate. Select Generated to generate a certificate, download a generated certificate, and to download a root certificate.

> **Caution:** Once the secure mode has been activated, an info message will be shown advising that the next connection will be done on HTTPS. After rebooting the device, the HTTP port will be redirected to HTTPS. There is a possibility to completely disable port 80, but be careful because if HTTPS is not configured, the web access will be lost and the only way to enable it again will be using CLI.

Finally, the last section of the HTTP/HTTPS menu enables Diffie-Hellman parameters in the TLS key exchange. This is optional but recommended. There are two buttons to generate and download the DH parameters file.

> **Caution:** Diffie-Hellman generation time: To generate the Diffie-Hellman parameters file, it is required to reboot the device and wait up to 20 minutes, or even more in some particular cases. In this period, the device MUST NOT be powered off, rebooted or any similar action. The device will not be accessible until this process finishes.

6.3 TACACS+

TACACS+ (Terminal Access Controller Access Control Server) is a security protocol for AAA (Authorization, authentication and accounting), which is used to provide centralised authentication for users who want to gain access to the network.

This section explains how to install and configure a TACACS+ on up to two servers on a Linux environment where the client is a WR-Z16 device.

The instructions to install and configure a TACACS+ server on an Ubuntu machine are explained in the Appendix " TACACS+ and RADIUS server configuration" on page 249.

In order to configure the TACACS+ protocol, it is necessary to modify the configuration file usually located at:

```
/etc/tacacs+/tac_plus.conf
```

Alternatively, the TACACS+ settings are located in the Web GUI under **Administration** > **Security** > **Authentication**.

6.3.1 Verification of TACACS+ installation

In order to verify the installation, it is possible to use the following set-up (see figure below). The TACACS+ client will ask for authentication to the server, which will answer if the user passed. Then the device will ask for credentials, which will be validated by the TACACS+ server and grant access to the user if the authentication was successful.

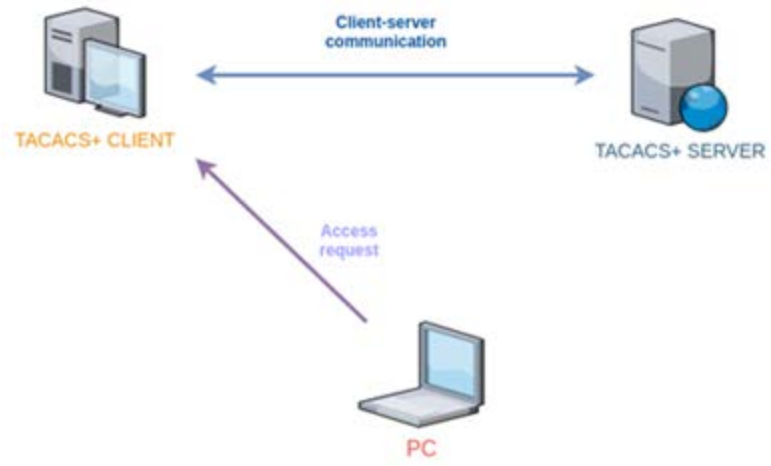


Figure 6-2: TACACS setup for verifying the installation.

6.3.2 TACACS+ Client configuration

Once the server is configured, it is necessary to configure the client. In this section, the client will be configured on the WR-Z16 device. For that purpose, gpa_ctrl is used to configure IP and secret. These parameters can be found in the security module:

```
root@zen-305:~# gpa_ctrl -s security auth/tacacs/server1_ip
172.17.5.39
```

```
root@zen-305:~# gpa_ctrl -s security auth/tacacs/server1_secret
sevensecret
```

And reboot to apply the changes.

Then the client can be accessed by using the configured user and password. In order to get debug messages from TACACS+, the service can be launched with the command tac_plus-g, always indicating the configuration file. For example, in the screenshots of figure a successful access with the TACACS+ password the first time, failed the second time and succeeded the third. Below you can see the verbose tac_plus output.

```
root@wrztpfl-417:~# Connection to 192.168.1.145 closed.
➜ ~ ssh tacuser1@192.168.1.145
Password:
Last login: Fri Jun 10 04:47:03 CEST 2022 from 192.168.1.181 on ssh
Welcome to WR ZEN TP-FL
wr-zynq-os version: v3.3
*****************************************************************
*  WARNING: We recommend you change the default password for the  *
*           'root' user immediately.                             *
*  You can do it by executing passwd (with option -a sha512)     *
*           or using the Web interface.                          *
*****************************************************************
root@wrztpfl-417:~# Connection to 192.168.1.145 closed.
➜ ~ ssh tacuser1@192.168.1.145
Password:
Password:
Last login: Fri Jun 10 04:47:16 CEST 2022 from 192.168.1.181 on ssh
Welcome to WR ZEN TP-FL
wr-zynq-os version: v3.3
*****************************************************************
*  WARNING: We recommend you change the default password for the  *
*           'root' user immediately.                             *
*  You can do it by executing passwd (with option -a sha512)     *
*           or using the Web interface.                          *
*****************************************************************
root@wrztpfl-417:~#
```

Figure 6-3: SSH connection with the WR-Z16 board

```
root@fc1081a5dc0f:/# /usr/sbin/tac_plus -C /etc/tacacs+/tac_plus.conf -g
Reading config
Version F4.0.4.27a Initialized 1
socket FD 3 AF 2
uid=0 euid=0 gid=0 egid=0 s=531977040
connect from 192.168.1.145 [192.168.1.145]
connect from 192.168.1.145 [192.168.1.145]
login failure: tacuser1 192.168.1.145 (192.168.1.145) ssh
connect from 192.168.1.145 [192.168.1.145]
```

Figure 6-4: tac_plus output with debug information

Caution: When TACACS and RADIUS work and have been configured on the same client device, be careful with the order of the configuration lines in /etc/pam.d/sshd. The TACACS configuration line must be added always in first place and after it, the RADIUS configuration line. This is because when the RADIUS configuration is the first line, authentication of the first password always goes to the RADIUS server and, if is the password of TACACS, the

authentication will fail. With TACACS configuration in first line, the first password is verified with both TACACS and RADIUS.

6.4 RADIUS

RADIUS (Remote Authentication Dial In User Service) is a security protocol for AAA (Authorization, authentication and accounting), which is used to provide centralized authentication for users who want to gain access to the network.

This section will define the processes necessary to install and configure the RADIUS client on up to two servers on the WR-Z16 device.

The steps to install and configure a RADIUS server on an Ubuntu machine are explained in Appendix" TACACS+ and RADIUS server configuration" on page 249.

Alternatively, the RADIUS settings are also located in the Web GUI under **Administration** > **Security** > **Authentication**.

6.4.1 RADIUS configuration files

The different existing configuration files to modify the operation of the protocol are:

- **radiusd.conf**: Contains protocol configuration parameters.
- **users**: Contains users and access passwords.
- **clients.conf**: Contains the list of clients that are allowed to make requests to the RADIUS server.
- **templates.conf**: The goal is to have a common configuration located in this file and list only the differences in the individual sections. This feature is more useful for sections such as "customers."
- **trigger.conf**: Used to set triggers for snmptrap.
- **proxy.conf**: RADIUS proxy and configuration directives.
- **policy.d**: Configuration files for policies of acceptance, rejection, filter, etc. of requests

6.4.2 Verification of RADIUS installation

In order to verify the installation, the following set-up is configured (Figure). When a user authenticates a device, this device will send a message to the

RADIUS server, which will accept or reject the user depending on if this device is taken as a client for this server.

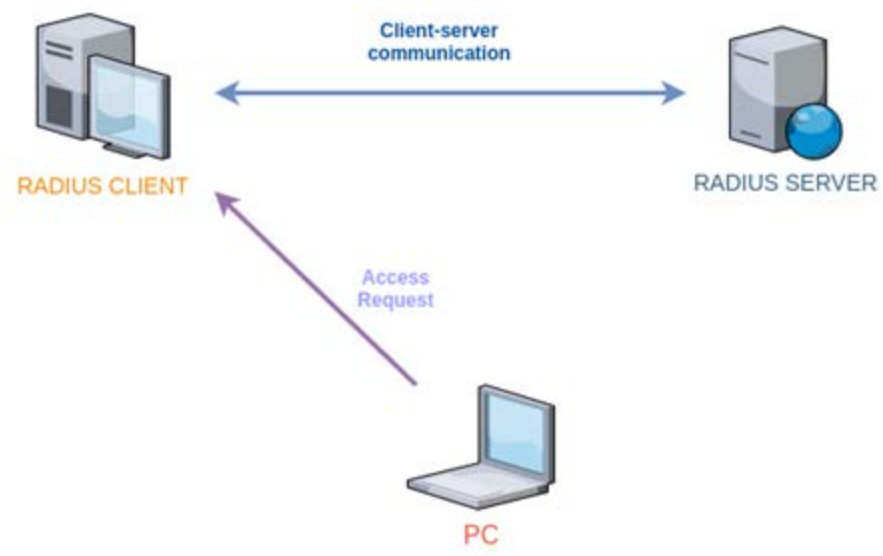


Figure 6-5: Set-up RADIUS for verifying the installation

6.4.3 RADIUS client configuration

Once the server is configured, the client must be configured as well. This section explains how to do it on the WR-Z16.

The use of gpa_ctrl allows to configure ip and secret. These parameters can be found in security module:

```
root@zen-305:~# gpa_ctrl -s security auth/radius/server1_ip
172.17.5.39
```

```
root@zen-305:~# gpa_ctrl -s security auth/radius/server1_secret
sevensecret
```

And reboot to apply the changes.

Now that everything has been configured correctly, it is possible to access the WR-Z16 board with these new passwords which have been set in the users file. In addition, the command freeradius-X can be used in order to verbose the RADIUS access.

The following figure shows an access using the password that was configured in the users file, but failing the first try. Looking at the output of freeradius at the host, it is possible to get the information from the first failed attempt:

```
mylaptop:~$ ssh test-radius@10.22.26.104
(test-radius@10.22.26.104) Password:
(test-radius@10.22.26.104) Password:
Welcome to WR ZEN TP-FL
wr-zynq-os version: v5.2
*** WARNING *** The current password is unsecure, please change it.
root@wrztpfl-1324:~#
```

Figure 6-6: SSH connection with the WR-Z16 board

```
(0)     [digest] = noop
(0) suffix: Checking for suffix after "@"
(0) suffix: No '@' in User-Name = "test-radius", looking up realm NULL
(0) suffix: No such realm "NULL"
(0)     [suffix] = noop
(0) eap: No EAP-Message, not doing EAP
(0)     [eap] = noop
(0) files: users: Matched entry test-radius at line 185
(0)     [files] = ok
(0)     [expiration] = noop
(0)     [logintime] = noop
(0)     [pap] = updated
(0)   } # authorize = updated
(0) Found Auth-Type = PAP
(0) # Executing group from file /etc/freeradius/3.0/sites-enabled/default
(0)   Auth-Type PAP {
(0) pap: Login attempt with password
(0) pap: Comparing with "known good" Cleartext-Password
(0) pap: ERROR: Cleartext password does not match "known good" password
(0) pap: Passwords don't match
(0)     [pap] = reject
(0)   } # Auth-Type PAP = reject
(0) Failed to authenticate the user
(0) Using Post-Auth-Type Reject
(0) # Executing group from file /etc/freeradius/3.0/sites-enabled/default
(0)   Post-Auth-Type REJECT {
```

Figure 6-7: Freeradius failed attempt with debug information

Caution: When TACACS and RADIUS work and have been configured on the same client device, be careful with the order of the configuration lines in /etc/pam.d/sshd. The TACACS configuration line must be added always in first place and after it, the RADIUS configuration line. This is because when the RADIUS configuration is the first line, authentication of the first password always goes to the RADIUS server and, if is the password of TACACS, the

authentication will fail. With TACACS configuration in first line, the first password is verified with both TACACS and RADIUS.

6.5 Firewall

The WRZ-OS is shipped with the standard iptable firewall that came in most of the Linux distribution.

The default rules applied is to forbid everything in the timing network (the optical fiber interface named wrX) so that only the necessary services can be accessed. The table below resume the port that can be accessed:

Table 6-1: Default firewall configuration

Timing (wrX)	
Service	**Port**
DNS	53
DHCP/BootP	67-68
NTP	123
NTS-KE	4460
PTP/WR	319-320

If an advanced user needs to customize the access to meet a specific security policy, he can use the persistent custom files ("Persistent Custom Files" on page 246) to overwrite the default rules with its own configuration.

6.5.1 Example to only allow a specific IP for management

This is a typical use case where only a single IP (or a subnetwork) should be allowed to access to the management port of the device.

```
##First append the current rule to existing rule (overwise
flush)
iptables -A INPUT -i eth0 -s 192.168.7.1 -j ACCEPT
iptables -A INPUT -i eth0 -j DROP
iptables -A INPUT -i eth1 -s 192.168.7.1 -j ACCEPT
iptables -A INPUT -i eth1 -j DROP
```

```
## Then save to local file so that this configuration is
applied at next reboot
```

```
iptables-save > /usr/local/etc/iptables.rules
```

Note: It is not recommended to edit the iptable files without any local access (UART) to the device as it is easy to make an error and fully block the network access to this device. To revert the changes, the user should perform a factory reset or delete the /usr/local/etc/iptables.rules files.

BLANK PAGE.

CHAPTER 7

Monitoring & Logging

The WR-Z16 device includes enhanced monitoring and logging tools to ease its deployment and manageability during operation.

The following topics are included in this Chapter:

7.1 Syslog

Syslog is a standard for message logging. It allows separating the software that generates messages, the system that stores them, and the software that reports and analyzes them. The aim of logging is to collect all the system information and make it easily accessible for the user. Kernel events, changes in the state of the device or user actions are sometimes useful information in terms of debugging or monitoring. Information about the state of the device in the past or a value of a given parameter in a certain time could be critical to find out the reason of a specific behavior of the device.

There are three different types of logging depending on the persistence:

- **Session logs**: These logs are initialized at boot time and are lost when the device is powered off. They are usually saved in a reserved directory /var/-log.
- **Permanent logs**: These logs are kept between reboots, giving information about the state of the device before it was restarted. These kinds of logs help to find out the reasons of the last reboot or if there is something preventing the device from start.
- **Remote logs**: They are saved remotely via rsyslog. It is necessary to set-up at least one external server for this purpose (max 2).

7.1.1 Session logs

During the operation of the device a log recording is performed, saving the information in different local files. These files are normally saved at /var/log, and have the following content:

- **auth.log**: It contains all the accesses or connections to the device through SSH (Secure Shell), serial port, web interface,...
- **boot.log**: It contains the boot information from a userspace perspective.
- **boot-procedure.log**: It contains the boot information from a kernel perspective.
- **wproc-child-xxx.log(*)**: These files contain the log of the module with the corresponding ID (100 -> wr0, 101 -> wr1, ..., 148 -> eth0, 149 -> eth1).
- **secure**: It contains the security logging.
- **systemlog**: It contains the kernel/user event logging.

Systemlog

In the same way as a normal Linux device, the kernel and the userspace processes send information to a central logger. Its contents can be found at

/var/log/systemlog and it centralizes all logs in a unique file via syslog.

The log entries have the following format:

```
May 28 06:19:06 zen-425 root: healthingd#W:
_gpa_prm_call_trigger_on_warning:852:
'(2.1002.2)alert/timing_state' changes to a warning value: Warning
```

As can be seen, the log event is divided in the following parts:

- **Timestamp**: It shows the date and time of the event information
- **DevID**: It shows the device identification (hostname)
- **Facility/Level**: It shows the type of program which is logging
- **Module**: It shows the name of the internal module that generated the logging
- **Message**: It shows information about the event

7.1.2 Permanent logs

The devices keep a permanent log to maintain the system information in case of unexpected reboots. This information is saved during the reboot process and can be found at /root/.log/reboot/.

- .last_reboot: It contains the timestamp of the last reboot
- wrz-xxx-xxx-xxxx-xxx.logdump: It contains the output of the wrz_logdump at the moment of the reboot

7.1.3 Remote logs

The devices can be configured to forward the system log information to a remote centralized server. This server needs to be configured by the user so it supports rsyslog. Saving information into the device normally is not practical for huge deployments, so it is recommended to set-up a rsyslog server and store the logging in a different machine, centralizing the logging for all devices in the deployment. The device can connect to up to 2 servers for this purpose, listed as Server 1 and Server 2.

7.1.4 Logging tools

Logdump

The wrz_logdump tool is responsible for generating the logdump. The logdump is a set of compressed files that can be easily shareable, which provides all the information about the current state and log of the device.

The logdump can be generated and downloaded from the web GUI by navigating to **Management** > **Maintenance** and selecting the Dump Log button or from the CLI executing the following command:

```
wrz_logdump -o /root/
```

To protect sensitive data, the logdump may be redacted and/or encrypted prior to being generated.

To redact sensitive data from the web GUI, select "Yes" under the "Redact sensitive data" setting. To do so from the CLI, execute the following command:

```
wrz_logdump -o /root/ -redacted
```

To encrypt the logdump from the web GUI, select "Yes" under the "Encrypt logs" setting and input your desired password in the "Encryption password" field. To do so from the CLI, execute the following command:

```
wrz_logdump -o /root/ -encrypted <password>
```

where <password> corresponds to your desired encryption password.

Caution: Sensitive information will remain un-redacted if "Verbose all" is enabled. For more information on "Verbose all," see "Configuration" on the facing page

The wrz_logdump contains different files that are useful to debug problems including the following information:

- The content of /boot and /media partitions including information about the software
- The main configuration from the /root/.config file
- Information about the interfaces, IP addresses, netmask, packets, status, etc
- Information about interrupts from the HW
- Information related with memory status
- The systemlog file under /var/log/systemlog
- Information about uboot and versions

Note: The Safran support service will require the wrz_logdump information in order to debug any issues. Please, download and attach this information when opening a support ticket.

All the logging information under /var/log/ is rotated for security reasons. This prevents to use all the available memory in the device in case the log files suddenly increase and is performed automatically when the file size exceeds 5 MB.

Note: Log rotation only affects the files with extension .log. Other files contained in /var/log/ folder are not affected.

7.1.5 Configuration

The logging configuration can be performed through the wrz_config tool in the CLI. Once the tool is launched, the logging configuration parameters can be found under Management > Logging.

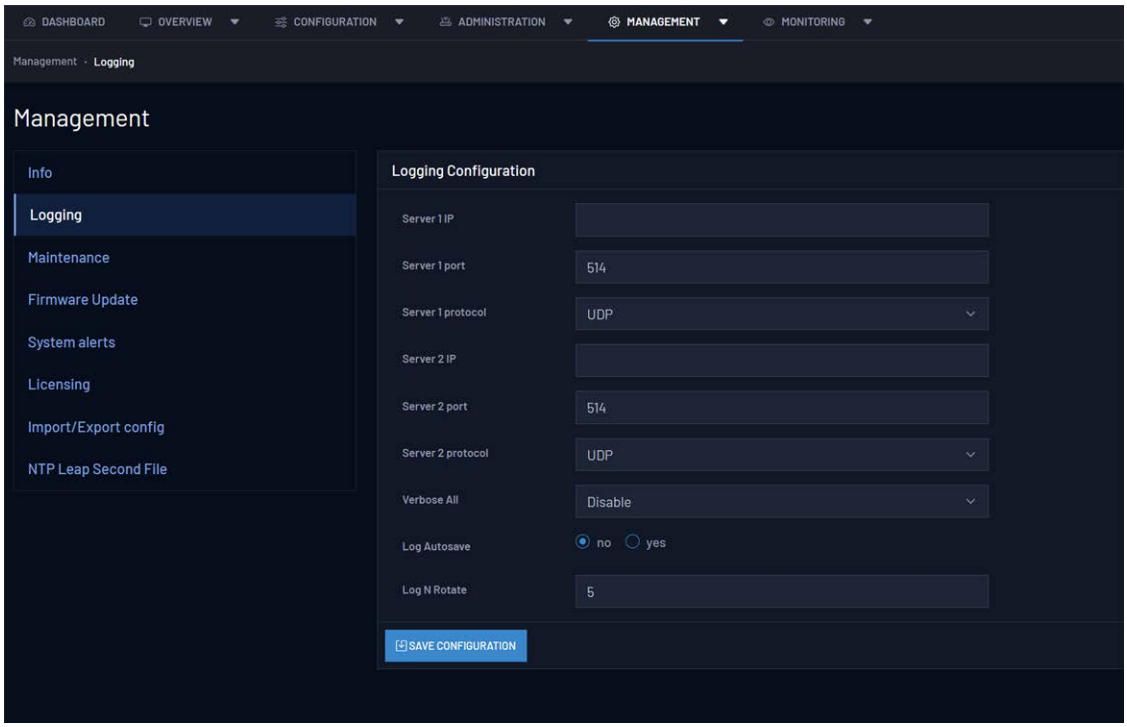
```
*** Logging configuration parameters ***
(192.168.7.1) server_ip
(514) server_port
    protocol (UDP)  --->
    verbose_all (Disable)  --->
    log_autosave (disabled)  --->
(5) log_n_rotate (NEW)
```

Figure 7-1: Logging configuration parameters through CLI.

The logging sub-tree is located under Misc. section and contains the following parameters:

OID	Name	Value Type	Description
13.2000.1	Server IP	<IP address> (i.e., 192.168.1.5)	IP address from the remote logging server.
13.2000.2	Server port	<Integer> (i.e., 514)	Port information from the remote logging server.
13.2000.3	Protocol	<Enum> • UDP • TCP	Communication protocol for remote logging between the device and the server.
13.2000.4	Verbose all	<Enum> • Disabled • Enabled	High verbosity logging configuration for modules and log information.
13.2000.5	Log autosave	<Enum> • Disabled • Enabled	Automatic permanent logging backup in the directory /root/.log with a periodicity of 6 hours in case of power cuts.
13.2000.6	Log N rotate	<Integer> (i.e., 5)	Number of logdumps rotations stored in the device.

Logging can also be configured via the Web GUI. Navigate to **Management** > **Logging** (see image below).



7.2 SNMP

Simple Network Management Protocol (SNMP) is an application–layer protocol defined by the Internet Architecture Board (IAB) in RFC1157 for exchanging management information between network devices. It is a part of Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite.

SNMP is one of the widely accepted protocols to manage and monitor network elements. Most of the professional–grade network elements come with bundled SNMP agent. These agents must be enabled and configured to communicate with the network management system (NMS).

7.2.1 Configuration

In order to understand the SNMP configuration, it is important to enumerate the configuration files in the device:

- /wr/etc/snmp/SEVEN-PRODUCT-MIB.txt: It contains the MIB file with all the SNMP OIDs in the device.

- /etc/snmp/snmpd.conf: It contains the global SNMP configuration. This file can be modified to customize the configuration.
- /usr/share/snmp/snmpd.conf: It contains the SNMP configuration managed by Seven Solutions software.
- /var/lib/snmp/snmpd.conf: It contains the SNMPv3 persistent data.
- /var/log/snmp/gpa_passwd.log: It contains the SNMPv3 user passwords change.

Note: SNMP file route: These files are located into /media/data/usr/local/... to make it persistent between reboots and firmware updates.

Caution: /usr/share/snmp/snmpd.conf modification: This file is automatically generated. Any manual changes will be lost when relaunching the SNMP daemon.

The SNMP parameters sub-tree is located under misc and is divided in three main parts. **SNMP**, **v1/v2**, and **v3**:

1. **SNMP**: General SNMP information.

OID	Name	Value Type	Description
13.3000.1	Location	<String> (i.e.,My location)	Name of the corresponding location.
13.3000.2	Contact	<String> (i.e.,user-@dom.com)	Contact information.

2. **v1/v2**: Parameters for SNMP v1 and v2.

OID	Name	Value Type	Description
13.3002.1	Community name	<String> (i.e., public)	Name of the community.

OID	Name	Value Type	Description
13.3002.2	Access view	<Enum> • none. • basic • extended • all	Access view options. • none: Disable SNMP v1 and v2. • basic: Show basic SNMP information. • extended: Show extended SNMP information. • all: Show all the SNMP information
13.3002.3	Access mode	<Enum> • ro: Read only. • rw: Read/write	Access mode options.
13.3002.4	Source mask (1-5)	<IP address> (i.e., 192.168.1.5 or 192.168.1.0/24)	IP addresses from hosts allowed to retrieve information from the device using SNMP v1 and v2 queries.

Note: In order to disable SNMP v1 and SNMP v2, the value none must be chosen for access_view.

Note: Source mask value: By default, the localhost source is added. If no other source mask is added the device will only accept local queries.

Caution: The default community name is public. For security reasons, it is recommended to change this parameter.

> **Caution:** Access mode configuration: For security reasons, it is not recommended to provide read/write permissions while using SNMP v1 or v2.

3. **v3**: Parameters for SNMP v3 users and passwords (see table below).

OID	Name	Value Type	Description
13.3X10.1	User name	<String> (i.e., userSNMP)	Name of the SNMPv3 user.
13.3X10.2	Access view	<Enum> • none. • basic • extended • all	Access view options. • none: Disable SNMP v1 and v2. • basic: Show basic SNMP information. • extended: Show extended SNMP information. • all: Show all the SNMP information
13.3X10.3	Access mode	<Enum> • ro: Read only. • rw: Read/write	Access mode options.
13.3X10.4	Auth	<Enum> • MD5 • SHA	Authentication encryption protocol.
13.3X10.5	Priv	<Enum> • DES • AES	Privacy encryption protocol.

7.2.1.1 General configuration

To configure SNMP via the Web GUI, log in to the device and navigate to **ADMINISTRATION** > **SNMP**.

In order to configure SNMP from the CLI, the wrz_config tool must be accessed. Once it has been launched, the SNMP configuration is under Management > SNMP.

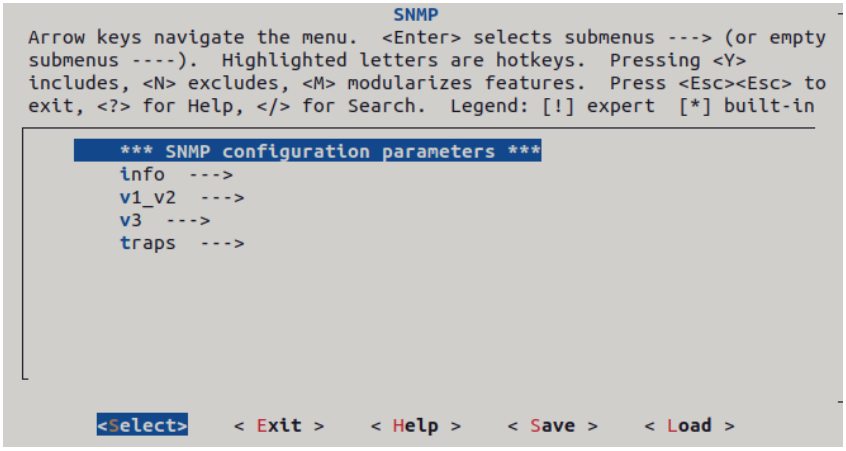


Figure 7-2: SNMP configuration.

Under the different tabs, all the described information in the previous tables or in the following sections can be accessed, modified, and saved.

On a different thread, the general commands to manage SNMP from a command line, e.g. **snmpget**, **snmpset**, **snmpwalk** or **snmpusm**, are available in the device. A detailed explanation about how to use them is out of the scope of this user guide, but multiple SNMP tutorial guides can be found online. Additionally, the **gpa_ctrl** tool allows to visualize all the parameters in a quick view.

```
gpa_ctrl -A misc snmp
```

The access mode parameter defines if it is allowed to execute a query (snmpset) for remote configuration using SNMP. If the parameter is set to read/write, the device will accept the query. However, if it is configured as read only, an error message will appear claiming that there is no access granted.

The view mode parameter defines the visible SNMP parameters in the device. In all cases, a minimum configuration that allows to configure the user passwords is set. The definition of available information in each category is defined in the SNMP configuration files. By default, none disables the SNMP version while the other three categories (basic, extended, all) provide increasing insights from the device information.

In order to create custom SNMP groups, the configuration file /usr/share/snmp/snmpd.conf can be modified to define or modify them.

Caution: SNMP configuration files customization: Safran is not responsible for any damage caused by the user while manually modifying the SNMP configuration files.

If it is needed to restore the default credentials in /etc/snmp/snmpd.conf and /var/lib/snmp/snmpd.conf the following command can be used:

```
/etc/init.d/snmpd force-reset
Are you sure you want to remove all persistent snmp files?
[y/N] y
```

Note: SNMP credentials reset: The snmpd force-reset command recreates persistent files but maintains the SNMP parameters sub-tree configuration.
The information in /usr/share/shmp/snmpd.conf is recreated following the saved SNMP sub-tree configuration.

7.2.1.2 Specific SNMP v1/v2 configuration

Additional SNMP v1 and v2 communities can be created editing the /etc/snmp/snmpd.conf file adding the following line:

```
rocommunity <community_name>
```

After modifying the file, restart the SNMP daemon to load the changes.

Alternatively, the groups definition can be modified to create and add communities. For that purpose, the information from the mapping section in /usr/share/snmp/snmpd.conf can be used as a reference to modify the /etc/snmp/snmpd.conf file.

7.2.1.3 Specific SNMPv3 configuration

By default, there are four created users in the device:

snmpv3 User	Auth Protocol	Priv Protoco	Auth Password (default)	Priv Password (default)
userSNMP	MD5	DES	userSNMPpass	userSNMPpass
adminSNMP	MD5	DES	adminSNMPpass	adminSNMPpass
secuserSNMP	SHA	AES	secuserSNMPpass	secuserSNMPpass
secadminSNMP	SHA	AES	secadminSNMPpass	secadminSNMPpass

Note: SNMP v3 user parameters: Only access_view and access_ mode parameters can be directly changed. Encryption protocols are changed through change_password.

Note: SNMP v3 users: UserSNMP and adminSNMP are included for retro-compatibility purposes. It is recommended to use users relying on SHA and AES encryption.

In the case of SNMP v3 the default passwords can be modified using the parameters under the change password sub-tree:

OID	Name	Value Type	Description
13.3800.0	User name	<String> (i.e., userSNMP)	Name of the SNMPv3 user.
13.3800.1	Auth	<Enum> • MD5 • SHA	Authentication encryption protocol.
13.3800.2	Priv	<Enum> • DES • AES	Privacy encryption protocol.
13.3800.3	Old Auth Password	<String> (i.e.,oldPassword)	Old Authentication password.
13.3800.4	Old Priv Password	<String> (i.e.,oldPassword)	Old Privacy password.
13.3800.5	New Auth Password	<String> (i.e.,newPassword)	New Auth Password
13.3800.6	New Priv Password	<String> (i.e.,newPassword)	New Privacy password.
13.3800.7	Change now	<Enum> • N: No • Y: Yes	Force the password change.

In order to create additional users in SNMPv3, the /etc/snmp/snmpd.conf can be modified using the createUser instruction:

```
createUser <user_name> <auth> <new_auth_password> <priv>
```

The access_mode for these users can be assigned as following:

```
<access_mode>user <user_name>
```

After saving the changes, the SNMP daemon must be restarted to apply them.

Alternatively, `net-snmp-create-v3-user` tool can be used after stopping the SNMP daemon or the groups definition can be modified to create users. For that purpose, the information from the mapping section in /usr/share/snmp/snmpd.conf can be used as a reference to modify the /etc/snmp/snmpd.conf file.

7.2.2 SNMP Traps

The SNMP traps are synchronous notifications generated by the agent which are sent to the manager. While in other SNMP communications, the manager actively requests information from the agent, the traps are sent from the agent to the manager without being explicitly requested. SNMP traps enable an agent to notify the management station of significant events by way of an unsolicited SNMP message. The SNMP traps include current sysUpTime value, an OID identifying the type of trap and optional variable bindings. A reference scenario where SNMP traps are used is shown below:

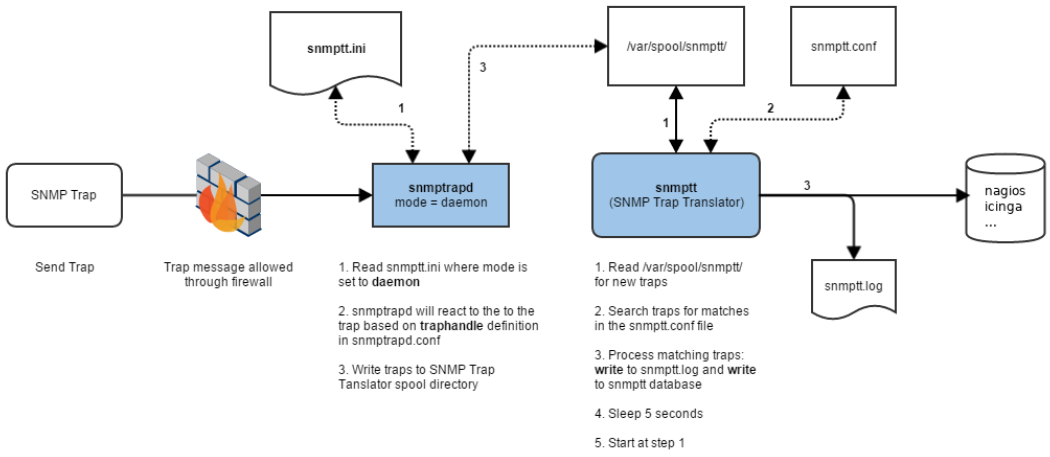


Figure 7-3: SNMP traps scheme.

Main parts:

- **SNMP Trap**: The device sends a trap to a monitoring device; the traps are defined in a MIB file.
- **snmptrad**: It receives the SNMP trap and manages it.
- **snmptt**: It translates the SNMP trap, classify the information with predefined clauses and acts accordingly to the rules defined for each SNMP trap type.
- **Monitoring center**: Monitoring software, e.g. Nagios, Icinga, etc, that handles the traps using snmptt.

> **Note:** The SNMP traps in this device are sent using SNMP v2, due to efficiency, security and simplicity reasons. All traps are sent with snmpinform that tries to confirm the reception of the trap by the Network Management System (NMS), and resends the trap until a timeout expires if it receives no confirmation.

7.2.2.1 Trap objects

The generated traps in the device contain different objects in order to provide general and specific information about the trigger of the notification.

- wrzTrapTime: Contains the time information when the trap was sent.
- wrzTrapPrmOID: Contains the OID information of the parameter associated to the generated trap.
- wrzTrapPrmKey: Contains the full path parameter key associated to the generated trap.
- wrzTrapPrmVal: Contains the value of the parameter that generated the trap as a number.
- wrzTrapPrmValStr: Contains the value of the parameter that generated the trap as a string if existing.
- wrzTrapModOID: Contains the OID information of the module associated to the generated trap.
- wrzTrapModKey: Contains the module name associated to the generated trap.

The best way to review our traps definitions is read trap section in SEVEN-PRODUCT-MIB.txt file. You can find this file inside your device at /wr/etc/snmp/SEVEN-PRODUCT-MIB.txt.

7.2.2.2 Trap notifications

There are different events that trigger the generation of a trap in the device including startup, shutdown, module open or close and parameter status.

- wrzInit: Trap generated when the system completely starts, and all services are initialized.
- wrzShutdown: Trap generated when all services are closed before a system shut down or reboot.
- modOpen: Trap generated when a module or service is launched.

- ModClose: Trap generated when a module or service is closed.
- okagainParam: Trap generated when a parameter comes back to a correct status after an alert condition.
- warningParam: Trap generated when a parameter changes to a warning status.
- criticalParam: Trap generated when a parameter changes to a critical status.
- outofrangeParam: Trap generated when a parameter goes to an out of range value.
- TrackedParam: Trap generated when a tracked parameter changes its value.

Note: Trap definition: An extended definition of the traps can be found in the MIB file in /wr/etc/snmp/SEVEN-PRODUCT-MIB.txt

Note: Tracked parameters: A selection of parameters in the devices have been performed and flagged like tracked parameters. These parameters create an alert each time that their value is changed.

7.2.2.3 Trap configuration

To configure SNMP traps via the Web GUI, log in to the device and navigate to the**Traps**tab in**ADMINISTRATION**>**SNMP**.

The SNMP traps generated by the device can be configured using the traps sub-tree under snmp.

OID	Name	Value Type	Description
13.3900.2-5	NMS IP 1-4	<IP address> (i.e., 192.168.1.5 or 192.168.1.0/24)	Authentication encryption pro-tocolDestination NMS IP address.
13.3900.6	Start/shutdown	<Enum> • Enabled • Disabled	Enable/disable star-tup and shutdown traps.
13.3900.7	Modules Start/Close	<Enum> • Enabled • Disabled	Enable/disable mod-ule launch or close traps.

OID	Name	Value Type	Description
13.3900.8	Prms tracked	<Enum> • Enabled • Disabled	Enable/disable tracked parameters traps.
13.3900.9	Prms alert	<Enum> • Enabled • Disabled	Enable/disable traps when a parameter is in an alert status or back to a normal status.
13.3900.10	Version	<Enum> • v2 • v3	The SNMP version number to send traps.
13.3910.1	Community name	<String> (i.e., public)	Name of the community for SNMP v2 traps.
13.3920.1	Username	<String> (i.e., secuserSNMP)	The SNMP username for SNMP v3 traps.
13.3920.2	Auth	<Enum> • MD5 • SHA	User auth protocol for SNMP v3 traps.
13.3920.3	Priv	<Enum> • DES • AES	User priv protocol for SNMP v3 traps.
13.3920.4	Auth password	<String> (i.e., secuserSNMP)	User auth password for SNMP v3 traps.
13.3920.5	Priv password	<String> (i.e., secuserSNMP)	User priv password for SNMP v3 traps.

Note: Default traps configuration: By default, all traps are enabled using the public SNMPv2 community for informative purposes.

Note: As of version 5.4, the Community name parameter with OID 13.3900.1 has been deprecated. If a configuration is written or loaded with this parameter, the information will be written to the new Community name parameter with OID 13.3910.1.

7.2.2.4 Basic trap receptor NMS configuration

Install snmptrapd in the server receiving the SNMP traps:

```
sudo apt-get install snmptrapd
```

After installing `snmptrapd`, the configuration in `/etc/snmp/snmptrapd.conf` needs to be modified to authorize the reception of traps.

For SNMP v2 traps, you will need to include:

```
disableAuthorization                                                yes
traphandle default /<example>/snmp_trap_test_handle.sh
```

For SNMP v3 traps, you will need to create the user with credentials defined in the device. For example:

```
createUser secuserSNMP SHA secuserSNMPpass AES secuserSNMPpass
authUser log,execute secuserSNMP
```

Once the configuration file has been modified, it is needed to edit the handle file for the received SNMP traps. For that purpose, it is important grant execution permissions to snmp_trap_test_handle.sh

```
#!/bin/sh

read host

read ip

vars=""

while read oid val; do

if [ "x$vars" = "x" ]; then

vars="$oid = $val"

else

vars="$vars, $oid = $val"

fi

done

echo trap: $1 $host $ip $vars
```

After this step, it is important to copy the MIB file from the device into the NMS.

```
sudo scp root@deviceip:/wr/etc/snmp/SEVEN- PRODUCT- MIB.txt /us-
r/share/snmp/mibs
```

Finally, the snmptrapd service must be stopped and re-run to view all the received traps.

```
sudo service snmptrapd stop

sudo snmptrapd -f -m all
```

7.3 LLDP

The WR-Z16 devices support the Link Layer Discovery Protocol (LLDP), which functions at the link layer (Layer 2 of OSI model) to discover neighboring devices and their capabilities.

7.3.1 Standard (IEEE 802.1AB-2005) TLVs

The WRZ-OS supports the mandatory and standard TLVs defined by the LLDP (IEEE 802.1AB-2005) protocol as listed below:

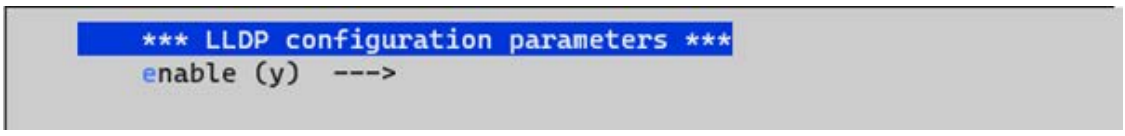
- Chassis ID
- Port ID
- Time-to-live
- Port Description
- System Name
- System Description
- System Capabilities
- Management Address

Therefore, when a neighbor supports LLDP, the mentioned TLVs will be recollected even if this neighbor does not run the WRZ-OS. The same apply in the over way, and the standard TLVs shared by the WRZ-OS device should be properly retrieved by any LLDP compatible device.

7.3.2 Configuration

In order to stop sharing device information to neighbors, the user must disable the LLDP protocol. By doing this, the device will also stop collecting information from its peers. (A configuration per ports will be coming soon).

Disabling LLDP can be performed through the wrz_config tool in the CLI. Once the tool is launched, the related parameter can be found under Management > LLDP as shown below.



Alternatively, the LLDP can be configured using the following parameters:

OID	Name	Value Type	Description
20.1100.0	Enable	<Boolean>	Enable sharing/collecting information between direct neighbors using LLDP.

7.3.3 Info/Overview

Note: In the current release (v5.5), LLDP information is not displayed by the web interface and can be only visualized from SNMP or using the CLI.

For each active network interface, LLDP will send its own information to the corresponding peer and recollect the information from the same peer if compatible with LLDP.

Note: Only active interface with a compatible LLDP neighbor are displayed. Other interfaces are leaved disabled. This means that if no neighbors are running a compatible LLDP agent, the LLDP daemons of this device will be empty.

The information gathered by each port running LLDP is then structured into three categories:

- Device: Information related to the system run by the neighbor.
- Port: Information related to the neighbor port.
- Management: Information about how the corresponding neighbor is managed.

OID	Name	Value Type	Description
20.xx10.x	**net/wrX/peer/**		**Information about LLDP for the wrX network interface. (Where OIDs follow the given pattern: wr0 → 20xx, wr1 → 21xx, ..., wr15 → 35xx)**
20.xx11.x	**net/wrX/peer/1/dev**		**Information related to the system run by the neighbor**
20.xx11.0	ID	<String> (i.e., 64:fb:81:20:80:06)	Unique identifier of the peer device (a.k.a Chassis ID): On WRZ-OS the MAC address of eth0 is used to ensure uniqueness
20.xx11.2	System Name	<String> (i.e., be-dist32-090)	Name of the system running on the peer device. By default, the hostname of the device is used.
20.xx11.3	System Description	<String> (i.e., WRZ-OS v3.2.1 for WR-ZEN TP-32BNC)	Description of the system running on the peer device
20.xx11.10	Firmware Version	<String> (i.e., v3.2.1-RC5)	Firmware version of the corresponding WRZ-OS (only) peer.
20.xx11.11	Hardware Version	<String> (i.e., WR_ZEN-v3.1)	Hardware version of the corresponding WRZ-OS (only) peer.
20.xx12.x	**net/wrX/peer/1/dev/timing**		**Information about the timing configuration of the neighbor**
20.xx12.1	Status	<String> (i.e., Ok)	General status of the peer device.
20.xx12.2	VCS Code	<Integer> (i.e., 20001)	Virtual clock use case code.

OID	Name	Value Type	Description
20.xx12.3	Message	<String> (i.e., Locked (TRACK_ PHASE))	Extra information for the vcs_code (Locked state, warning condition, etc).
20.xx12.4	Active Reference	<String> (i.e., BC: WR @ wr1)	Massage that contains the Active reference.
20.xx20.x	**net/wrX/peer/1/port**		**Information related to the neighbor port**
20.xx20.0	ID	<String> (i.e., 64:fb:81:20:88:06)	Unique identifier of the remote port (a.k.a port ID). For WRZ-OS peers, the MAC address of the corresponding port is used.
20.xx20.3	Description	<String> (i.e.,wr0)	Description of the remote port. For WRZ-OS peers, it corresponds to its interface name.
20.xx22.x	**net/wrX/peer/1/port/sfp**		**Information related to the neighbor SFP**
20.xx22.1	Vendor Name	<String> (i.e., Axcen Photonics)	SFP vendor name.
20.xx22.2	Part Number	<String> (i.e., AXGE-3454-0531)	SFP part number.
20.xx22.3	Serial Number	<String> (i.e., AX17460000223)	SFP serial number.
20.xx22.4	Transmission Wavelength	<Decimal> (i.e., 1490.000000)	SFP transmission wavelength.
20.xx22.5	DOM Availability	<Boolean> (i.e., Yes)	SFP DOM present flag.
20.xx22.6	Temperature	<Decimal> (i.e., 0.000000)	SFP temperature.
20.xx22.7	Reception Path Power	<Decimal> (i.e., 0.000000)	SFP power measurement for Rx path.
20.xx30.x	**net/wrX/peer/1/mgmt**		**Information about how the corresponding neighbor is managed.**

OID	Name	Value Type	Description
20.xx30.0	Address	<String> (i.e., 192.168.7.36)	Management address of the remote peer.
20.xx40.x	**net/wrX/peer/1/hati**		**Information about HATI Parameters.**
20.xx40.1	Version	<String> (i.e., HTI-v2.2.1-3-ga7a89b1)	HATI FW Version.
20.xx40.2	Offset	<Integer> (i.e., -3)	Current applied clock offset in ps.
20.xx40.3	MS Delay	<Integer> (i.e., 417980)	Master/Slave propagation delay in ps.
20.xx40.4	Round Trip Time	<Integer> (i.e., 839160)	Computed Round Trip Time.
20.xx40.5	Update Count	<Integer> (i.e., 20189)	Count of PTP exchanges.
20.xx40.6	TX Package Counter	<Integer> (i.e., 34943)	Number of transmitted packages by HATI.
20.xx40.7	RX Package Counter	<Integer> (i.e., 105083)	Number of packages received by HATI.
20.xx40.8	Manual Timeshift	<Integer> (i.e., 0)	Manual shift applied in ps to the PPS.
20.xx40.9	TAI Epoch	<Integer> (i.e., 1700087763)	TAI epoch reference value.
20.xx40.10	TAI Date	<Date> (i.e., 2023-11-15 10:23:55)	TAI date.
20.xx40.11	Sevo State	<Integer> (i.e., 4)	Synchronization Status

7.3.4 LLDP Locked Logging

The synchronization state (or servo state in HATI) change from Locked to Not locked or vice versa, is stored into /var/log/systemlog, here there is an example:

```
Oct 18 13:05:24 z16-303 root: /wr/bin/lldp-collect 2>&1 | logger
&
Oct 18 13:05:25 z16-303 root: /usr/sbin/lldpd -x 2>&1 | logger &
Oct 18 13:05:25 z16-303 lldpd[4007]: no privilege separation
available
```

```
Oct 18 13:05:25 z16-303 lldpd[4007]: protocol LLDP enabled
Oct 18 13:05:25 z16-303 lldpd[4007]: protocol CDPv1 disabled
Oct 18 13:05:25 z16-303 lldpd[40s07]: protocol CDPv2 disabled
Oct 18 13:05:25 z16-303 lldpd[4007]: protocol SONMP disabled
Oct 18 13:05:25 z16-303 lldpd[4007]: protocol EDP disabled
Oct 18 13:05:25 z16-303 lldpd[4007]: protocol FDP disabled
Oct 18 13:05:25 z16-303 lldpd[4007]: libevent 2.1.12-stable ini-
tialized with epoll method
Oct 18 13:05:25 z16-303 lldpd[4007]: enable SNMP subagent
Oct 18 13:05:25 z16-303 lldpd[4007]: NET-SNMP version 5.9 AgentX
subagent connected
Oct 18 13:05:26 z16-303 lldpcli[4008]: lldpd should resume
operations
Oct 18 13:05:26 z16-303 root: /wr/bin/lldp-agent 2>&1 | logger &
Oct 18 13:05:35 z16-303 root: lldp#W: lldp_log_event_action:43:
Synchronization status change -> status Locked, MAC 64:f-
b:81:20:1d:ee, Device wrztpB-494
```

7.4 Healthing

The healthing module provides general information about the system health for monitoring purposes. This includes information about the fans, power supplies, memories, or temperature between others.

7.4.1 Information/Overview

The associated parameters can be accessed through the Web GUI or the command line. The Web GUI has three Healthing pages, located under **Overview** > **Healthing**.

The Healthing System Overview page provides all necessary information on temperature, computer usage, etc.:

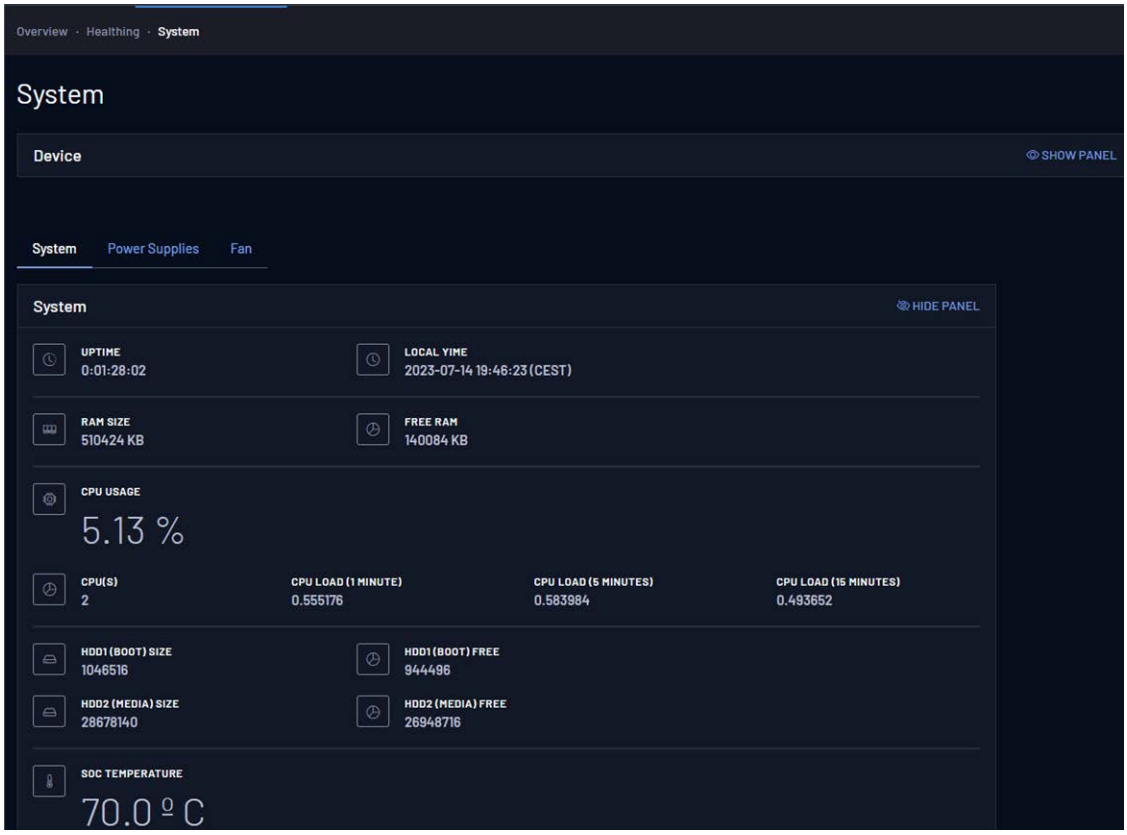


Figure 7-5: Healthing System Overview

The Healthing Power Overview page contains all necessary information related to the status and functionality of the power supplies:

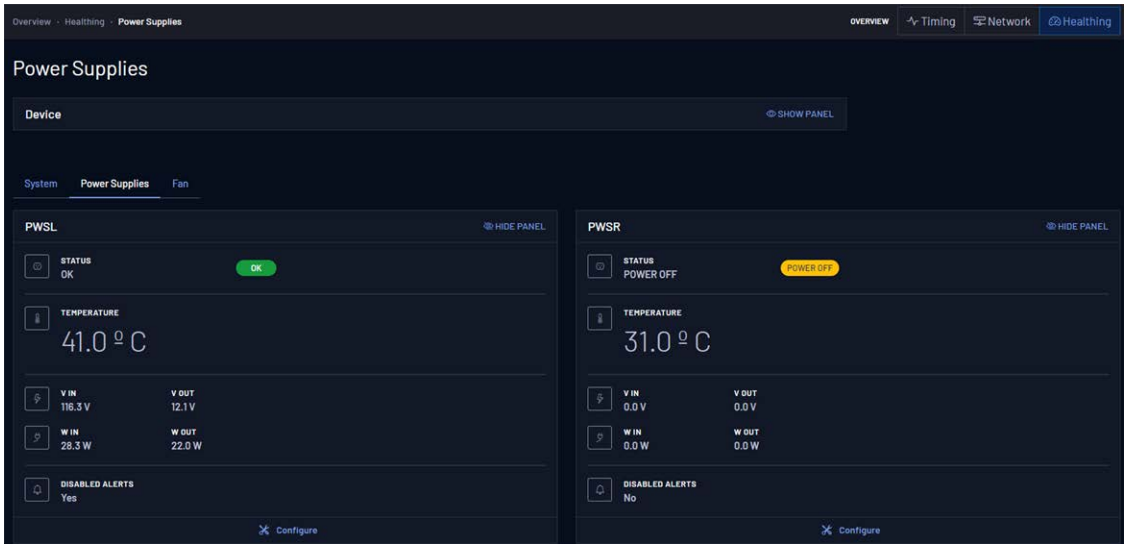


Figure 7-6: Healthing Power Overview

And the Healthing Fan Overview details all necessary fan state information:

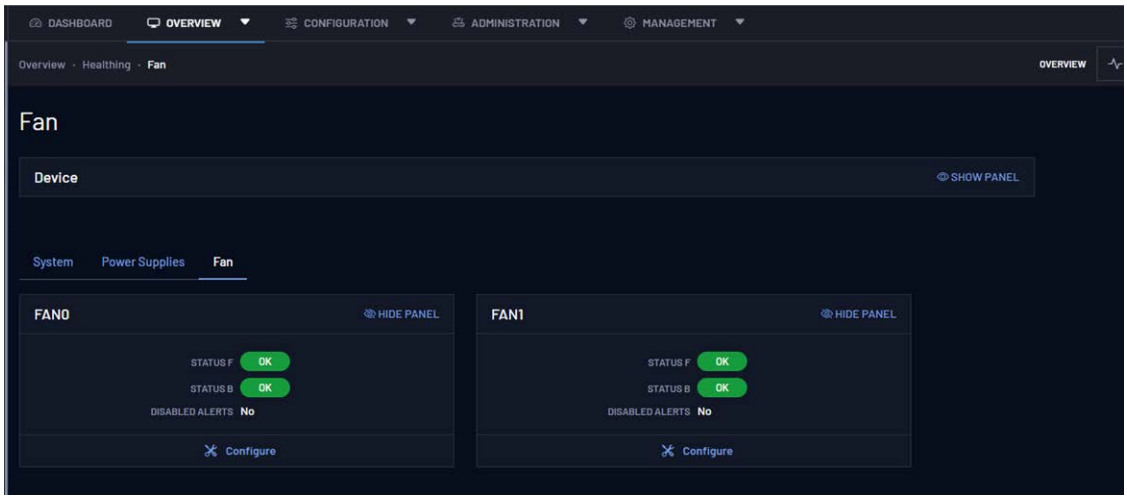


Figure 7-7: Healthing Fan Overview

Alternatively, the Healthing can be obtained using the following parameters:

OID	Name	Value Type	Description
2.1001.10	Uptime	<Time> (DD:HH:MM:SS)	Up time since the last reboot our power cycle.
2.1001.11	Local time	<Date and Time> (YYYY-MM-DD HH:MM:SS (UTC))	System date and hour in UTC format.
2.1001.20	RAM total	<Integer> (i.e., 511348)	Total available RAM.
2.1001.21	RAM free	<Integer> (i.e., 93884)	Remaining free RAM.
2.1001.31	CPUs	<Integer> (i.e., 2)	Total available CPUs.
2.1001.31	CPU load 1	<Decimal> (i.e., 0.054199)	Average CPU load during the last minute.
2.1001.32	CPU load 5	<Decimal> (i.e., 0.054199)	Average CPU load during the last 5 minutes.
2.1001.33	CPU load 15	<Decimal> (i.e., 0.054199)	Average CPU load during the last 15 minutes.

OID	Name	Value Type	Description
2.1001.34	CPU usage	<Decimal> (i.e., 0.054199)	Average CPU usage percentage in all cores.
2.1001.40	HDD1 size	<Integer> (i.e., 1046516 kB)	BOOT partition hard disk memory size.
2.1001.41	HDD1 free	<Integer> (i.e., 966132 kB)	BOOT partition free hard disk memory size.
2.1001.50	HDD2 size	<Integer> (i.e., 13785168 kB)	DATA/MEDIA partition hard disk memory size.
2.1001.51	HDD2 free	<Integer> (i.e., 12906276 kB)	DATA/MEDIA partition free hard disk memory size.
2.1001.60	FPGA temp	<Decimal> (i.e., 60 ºC)	Measured temperature in the FPGA.

Additionally, to these parameters, the system defines several smart alerts that comprise the information from several parameters to ease the monitoring, providing a quick overview of the general status:

OID	Name	Value Type	Description
2.1002.1	Global state	<Enum> 0. Ok 1. Warning 2. Critical	Global status including timing and system parameters.
2.1002.2	Timing state	<Enum> 0. Ok 1. Warning 2. Critical	Timing status extracted from the virtual active clock.
2.1002.3	System	<Enum> 0. Ok 1. Warning 2. Critical	System status extracted from the healthing parameters.

The devices incorporate redundant power supplies and fans. In order to ensure their proper behavior, their information can be checked too in the web GUI under Healthing or through the command line:

OID	Name	Value Type	Description
0.91x0	**pws/pwsX/**		**Information related to pwsX where pwsl (0.9100) corresponds to the left power supply and pwsr (0.9120) corresponds to the right power supply.**
0.9110.1	Status	<Enum> (i.e., OK)	Power supply status.
0.9110.1	Temperature	<Decimal> (i.e., 41 ºC)	Power supply temperature.
0.9110.1	Voltage In	<Decimal> (i.e., 233.250000 V)	Power supply input voltage.
0.9110.1	Voltage Out	<Decimal> (i.e., 11.949219 V)	Power supply output voltage.
0.9110.1	Power In	<Decimal> (i.e., 30.000000 W)	Consumption of the power supply input power.
0.9110.1	Power Out	<Decimal> (i.e., 24.000000 W)	Consumption of the supply output power.
0.91x0	**fan/fanX/**		**Information related to the module fanX. (Where OID 9210→fan0, 9220→fan1)**
0.92x0.1	Status Front	<Enum> 0. OK 1. Unplugged 2. Stopped 3. I2C Error	Status of the front ventilator of fanX module.
0.92x0.2	Status Back	<Enum> 0. OK 1. Unplugged 2. Stopped I2C Error	Status of the back ventilator of fanX module.

7.4.2 Configuration

There are a few parameters that can be configured. In the CLI, these parameters are configured via the Healthing module:

```
*** Healthing configuration parameters ***
(1) screen_saver
(60) screen_saver_delay
(255) screen_contrast
(60) temp_target (NEW)
    fan0  --->
    fan1  --->
    pwsl  --->
    pwsr  --->
```

Figure 7-8: Healthing configuration through CLI.

In the Web GUI, the healthing settings for fans and power supply modules are found under **Management** > **System Alerts**:

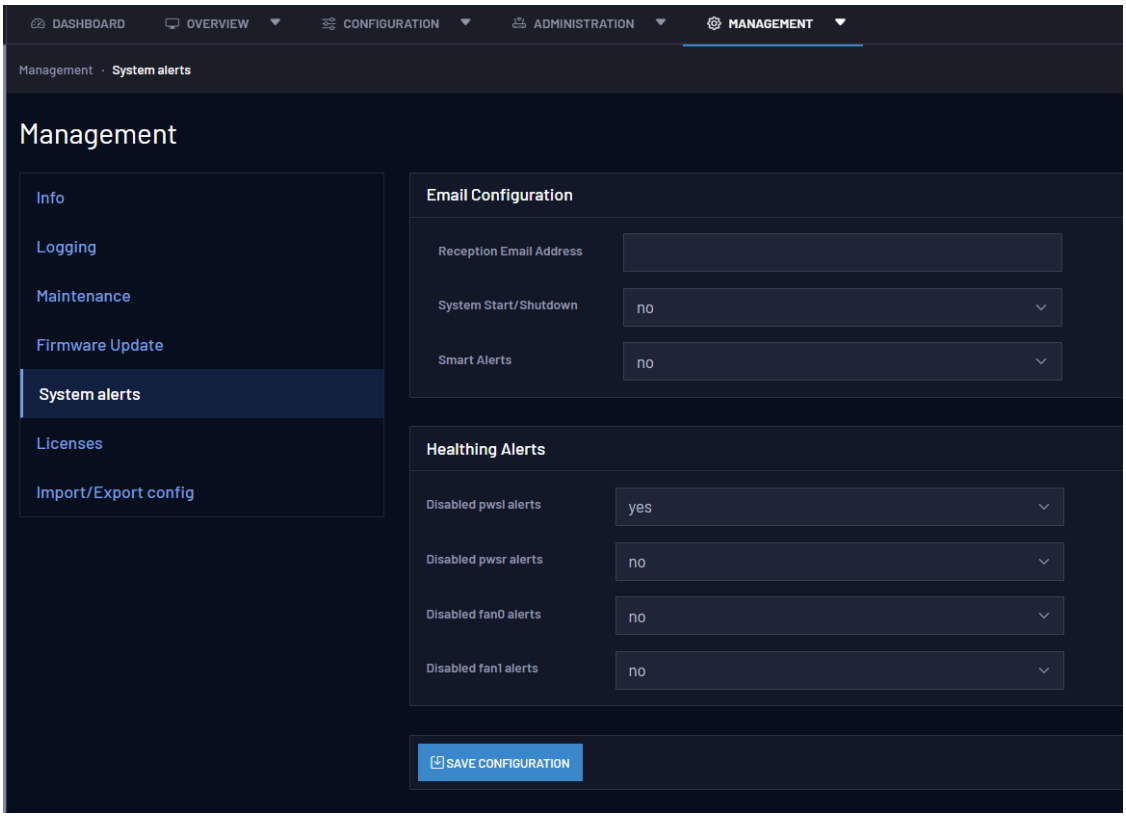


Figure 7-9: Healthing Web GUI settings

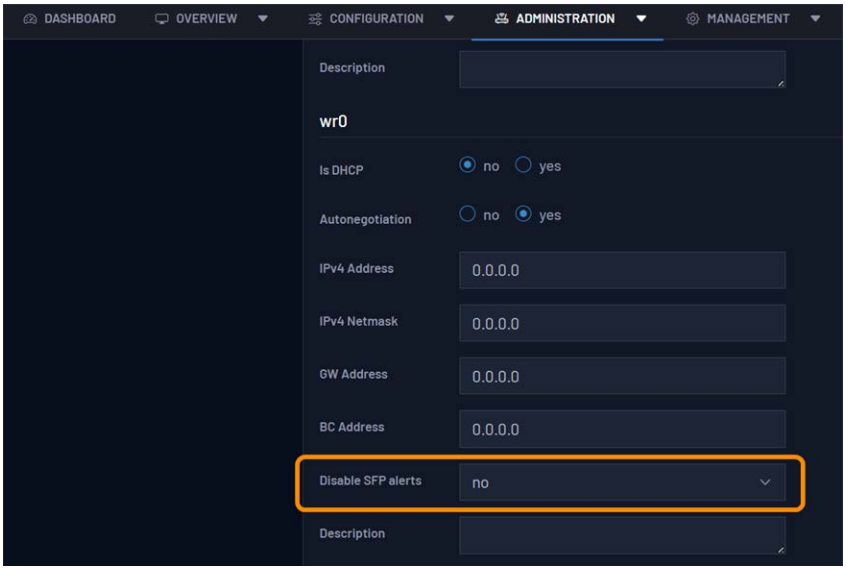
The healthing, fans and power supply configuration parameters can be found in the following table:

OID	Name	Value Type	Description
2.1000.0	Screen saver	<Integer> (i.e., 1)	Not used in WR-Z16 device.
2.1000.1	Screen saver delay	<Integer> (i.e., 60)	Not used in WR-Z16 device.

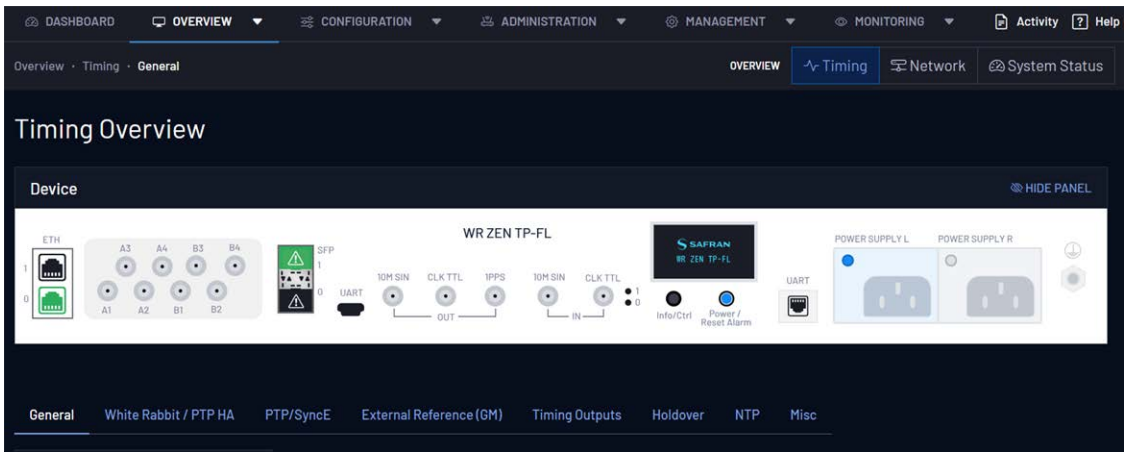
OID	Name	Value Type	Description
2.1000.2	Screen contrast	<Integer> (i.e., 255)	Not used in WR-Z16 device.
2.1000.3	Temp target	<Integer> (i.e., 60 ºC)	Target temperature for the fans PWM controller
0.9110.7	PWSL disable alert	<Enum> - No - Yes	Disable left power supply alerts.
0.9120.7	PWSR power OUT	<Enum> - No - Yes	Disable right power supply alerts.
0.9210.5	Fan 0 disable alert	<Enum> - No - Yes	Disable fan 0 alerts.
0.9220.5	Fan 1 disable alert	<Enum> - No - Yes	Disable fan 1 alerts.

7.4.3 SFP Alerts

SFP Alerts can be configured to monitor DOM parameters and assist in detecting defective SFPs. Alerts are disabled by default and can be enabled for individual interfaces by navigating to **ADMINISTRATION** > **Network** > **Interfaces**.



When enabled, if any SFP DOM parameter is outside its warning range an alert icon will appear in the corresponding interface within the Device panel of **OVERVIEW** > **Network** > **Interfaces**.



Note: SFP alerts can be enabled for both linked and unlinked ports.

An alert icon will also appear next to the interface's name in the Interfaces panel. Expanding the advanced settings of the corresponding interface in the Interfaces panel will display each parameter in its warning range as red text.

1. SFP alert icon indicating one ore more parameters in warning range.
2. Red highlighted parameter indicating it is in its warning range.

The monitored DOM parameters and their warning ranges can be found in the following table:

OID	Name	Value Type	Warning Range
0.2X11.6	Temperature	<Decimal> (i.e., 39.933594 ºC)	Outside of 0 °C to 70 °C
0.2X11.7	Voltage	<Decimal> (i.e., 3.316100 V)	Outside of 3.13 V to 3.47 V
0.2X11.8	TX bias	<Decimal> (i.e., 18.320000 mA)	Outside of 10 mA to 80 mA

OID	Name	Value Type	Warning Range
0.2X11.9	TX power	<Decimal> (i.e., -5.497509 dBm)	Outside of -8 dBm to -1 dBm
0.2X11.10	RX power	<Decimal> (i.e., -4.837291 dBm)	Outside of -12 dBm to +0.5 dBm

7.5 Service Persistent Raw Data and Runtime Statistics

The Service Persistent Raw Data and Runtime Statistics collection increases the observability of the system by storing the values of certain parameters, and by keeping track of some statistics computed at runtime based on the stored values.

The persistent storage records historical data of the monitored parameters for some time, allowing users to retrieve and perform analysis over time.

Additionally, parameter runtime statistics provide the current values for the min, max, and mean statistics, computed from the sampled date starting from the last time their statistics were reset.

For instance, the service `ppsi` keeps record of the following parameters for every WR interface, and for the active servo.

Parameter	Description
act/servo/state	State of the servo
act/servo/offset_from_master	The time error between a Slave Clock and a Master Clock
act/servo/delay_MS	Delay between Master and Slave
act/servo/delay_MM	Measured round trip time including fixed+semistatic delays
act/servo/mean_delay	Half of the cable round trip time excluding fixed+semistatic (cRTT/2).

For the sake of simplicity, this section will use the `ppsi` service as example, but every example applies to any other service as they share the same structure. For a complete list of parameters with statistics and persistent storage enabled see "List of Parameters with Statistics Enabled" on page 247.

7.5.1 Persistent Raw Data

Raw data from services is stored in a single SQLite3 database. The datafiles are stored in the home of the root user, located at `/root/.db/metrics.db`. When it

comes to the database schema, each service creates two different tables. The first one maps parameters to IDs, while the second one contains the actual data. The number of rows in the data table depends on the number of parameters with statistics activated.

The aforementioned pair of tables are prefixed by the service name that owns them and suffixed by `_info` and `_raw`. So, for instance, for the ppsi service, the tables are called `ppsi_info` and `ppsi_raw`. There are also two tables for the PTP service; they are called `wptpd_info` and `wptpd_raw`.

The following sections describe the schema of the tables, and how to access them.

7.5.1.1 Database Schema

As mentioned above, each service creates its _info and _raw. For `ppsi`, the schema could be:

```
sqlite> .schema

CREATE TABLE ppsi_info (
METRIC_ID TEXT PRIMARY KEY,
METRIC_NAME TEXT NOT NULL,
PARAM_NAME TEXT NOT NULL,
METRIC_TYPE TEXT NOT NULL
);

REATE TABLE ppsi_raw (
TIMESTAMP INTEGER PRIMARY KEY,
M0000 TEXT, M0001 REAL, M0002 REAL, M0003 REAL, M0004 REAL,
M0005 TEXT, M0006 REAL, M0007 REAL, M0008 REAL, M0009 REAL,
M0010 TEXT, M0011 REAL, M0012 REAL, M0013 REAL, M0014 REAL
);
```

Column names of the _raw table correspond with the data of the METRIC_ID column stored in the _info table.

For this particular `ppsi` example:

METRIC_ID	METRIC_NAME	PARAM_NAME	METRIC_TYPE
M0000	raw	act/servo/state	enu
M0001	raw	act/servo/offset_from_master	f64
M0002	raw	act/servo/delay_MS	f64

METRIC_ID	METRIC_NAME	PARAM_NAME	METRIC_TYPE
M0003	raw	act/servo/delay_MM	f64
M0004	raw	act/servo/mean_delay	f64
M0005	raw	net/wr0/1/servo/state	enu
M0006	raw	net/wr0/1/servo/offset_from_master	f64
M0007	raw	net/wr0/1/servo/delay_MS	f64
M0008	raw	net/wr0/1/servo/delay_MM	f64
M0009	raw	net/wr0/1/servo/mean_delay	f64
M0010	raw	net/wr1/1/servo/state	enu
M0011	raw	net/wr1/1/servo/offset_from_master	f64
M0012	raw	net/wr1/1/servo/delay_MS	f64
M0013	raw	net/wr1/1/servo/delay_MM	f64
M0014	raw	net/wr1/1/servo/mean_delay	f64

So, for example, the column M0000 corresponds to the metric `act/servo/state`, and the column M0012 corresponds to the metric `net/wr1/1/servo/delay_MS`.

Please note that the METRIC_ID is subject to change and should not be used directly to access the _raw table. Instead, always query the _info table to retrieve the PARAM_NAME to METRIC_ID mapping.

Here is an excerpt of the data from the corresponding _raw table, specifically the ppsi_raw table:

```
sqlite> select TIMESTAMP, M0000, M0001, M0002, M0003 from
ppsi_raw;
```

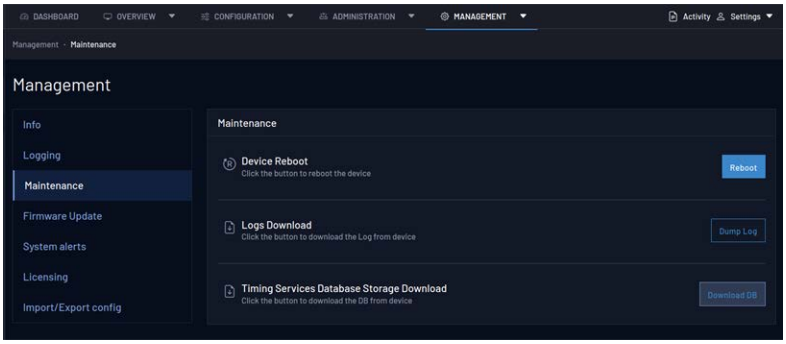
TIMESTAMP	M0000	M0001	M0002	M0003
1699978917	Adjusting Phase	8239778.69	-2.36115e-07	-4.72166e-07
1699978918	Adjusting Phase	8239778.69	-2.36115e-07	-4.72166e-07
1699978919	Adjusting Phase	8246688.591	-3.14177e-07	-6.2827e-07
1699978920	Adjusting Phase	8250518.773	6.07921e-07	1.215681e-06
1699978921	Adjusting Phase	8254038.727	9.1847e-08	1.83671e-07
1699978922	Adjusting Phase	8257666.677	9.00075e-07	1.799908e-06

TIMESTAMP	M0000	M0001	M0002	M0003
1699978923	Adjusting Phase	8261280.761	3.05927e-07	6.11772e-07
1699978924	Adjusting Phase	8264008.612	1.3921e-08	2.7838e-08
1699978925	Adjusting Phase	8269110.874	8.32007e-07	1.663792e-06
1699978926	Adjusting Phase	8269110.874	8.32007e-07	1.663792e-06

Dumping the Database

There are four ways to dump the database:

- via the Web GUI. Navigate to the Management -> Maintenance Tab to download the database containing the parameters data of timing services:



- calling the tool raw_data_db_dump from the CLI. The output file will be /root/raw_data_dump.csv
- using endpoint rawdata/db_dump of the REST-API, a file download will start; and
- manually, using the SQLite3 console. See the example in "Dumping the Database Manually" below.

Both API and the toll will output a CSV file, whereas the manual mode allows more flexibility.

Dumping the Database Manually

SQLite allows dumping the database in several formats such as CSV, SQL inserts, or JSON. For example, to export the database as a CSV file called metrics.csv follow these steps:

1. Open the SQLite command line by running `sqlite3 /root/.db/metrics.db`
2. Enable headers by entering `.headers on`
3. Set the output mode to CSV by entering .mode `csv`.
4. Specify the output file by entering `.output metrics.csv`
5. Select the desired data from the table `ppsi_raw` by performing the query select `TIMESTAMP,M0000,M0001,M0002,M0003 from ppsi_raw;`
6. Exit the SQLite command line by entering `.quit`

The resulting metrics.csv file will be stored in the current working directory.

For further information about database management please refer to the SQLite3 help and documentation.

Database Rotation

Raw data is kept in the database for seven days. Older data is removed automatically in a circular buffer fashion.

7.5.1.2 Accessing the RAW Data using the API

The endpoints to perform queries to the database are listed under the RAW DATA category in the Swagger API interface. The particular endpoints for performing queries are the following:

ENDPOINT	DESCRIPTION
rawdata/wr/{wr_interface}	Retrieve raw data of a single WR interface
rawdata/wr/servo/active	Retrieve raw data of the active servo
rawdata/ptp/{ptp_interface}	Retrieve raw data of a single PTP interface
rawdata/ptp/servo/active	Retrieve raw data of the active PTP servo
rawdata/gm/	Retrieve raw data of GrandMaster Clock

Along with the data values for a specific time frame, raw data endpoints can return statistics (min, max, mean, stdev) if requested.

Tip: Please keep in mind that raw data retrieving and statistics calculations are computationally expensive operations.

Please refer to the API definition for further information about the usage of these endpoints.

7.5.2 Runtime Statistics

Runtime statistics are internally handled can be retrieved using the CLI, SNMP, and the REST-API, the following sections describe how to access them.

7.5.2.1 Accessing Runtime Statistics from the CLI

To access runtime statistics from the command line interface (CLI), you can use the `gpa_ctrl` tool. The tool allows you to read and reset statistics.

Here are some examples of **reading the WR statistics** by specifying the `-M` modifier:

- To show every metric for a given parameter:

```
# gpa_ctrl ppsi net/wr0/1/servo/delay_MM -M
id:0:min: 0.000000007383
id:1:max: 0.000000291743
id:2:mean: 0.000000097378
```

- To request only the statistis with ID 2:

```
# gpa_ctrl ppsi net/wr0/1/servo/delay_MM -M2
id:2:mean: 0.000000123229
```

You can **reset metrics** by specifying the -Mr modifier. Here are some examples:

- To reset every statistic for a given parameter:

```
# gpa_ctrl ppsi net/wr0/1/servo/delay_MM -Mr
```

Result:

```
# gpa_ctrl ppsi net/wr0/1/servo/delay_MM -M
id:0:min: 0.000000000000
id:1:max: 0.000000000000
id:2:mean: 0.000000000000
```

- To reset only the statistic with ID 2:

```
# gpa_ctrl ppsi net/wr0/1/servo/delay_MM -Mr2
```

Result:

```
# gpa_ctrl ppsi net/wr0/1/servo/delay_MM -M2
id:2:mean: 0.000000000000
```

Here are some examples of **reading the PTP or GM statistics** by specifying the `-M` modifier:

- To reset the offset from master metrics of PTP:

```
# gpa_ctrl wptpd net/wr0/1/info/offset_from_master -Mr
id:0:min: 0.000000000000
```

```
      id:1:max: 0.000000000000
      id:2:mean: 0.000000000000
```

» To request only the statistis with ID 2 of the same previous metrics:

```
      # gpa_ctrl wptpd net/wr0/1/info/offset_from_master -M2
      id:2:mean: 0.000000000961
```

7.5.2.2 Accessing Runtime Statistics through SNMP

Runtime parameter statistics can also be retrieved and reset through SNMP by requesting the `wrzParamMetrics` and `wrzParamXResetMetrics` tables.

Here is an example of retrieving statistics through SNMP:

```
      # snmpget -v 3 -u adminSNMP -l authPriv -a MD5 -x DES -A
      adminSNMPpass -X adminSNMPpass 10.22.26.213 SEVEN-PRODUCT-
      MIB::wrzParamMetrics.1.2020.25
```

```
      SEVEN-PRODUCT-MIB::wrzParamMetrics.1.2020.25 = STRING: min:
      -0.003000000000 max: 0.001000000000 mean: -0.000526315789
```

To reset runtime statistics through SNMP, you can use the snmpset command. Here is an example:

```
      # snmpset -v 3 -u adminSNMP -l authPriv -a MD5 -x DES -A
      adminSNMPpass -X adminSNMPpass 10.22.26.108 SEVEN-PRODUCT-
      MIB::wrzParamXResetMetrics.1.2020.25 i 1
```

7.5.2.3 Accessing Runtime Statistics through the REST-API

Runtime metrics are retrieved using the same endpoint that is used to retrieve the actual value of a parameter, i.e, the gpa endpoint. They are requested by specifying the complete format. They are stored under the `metrics_dic` field of the response.

```
      {
        "msg": "Success",
        "data": {
          "servo": {
            "offset_from_master": {
              "value": "  0.000000007383",
              "value_str": "0.000000007383",
              "status_str": "",
              "conf_value": "",
              "oid": "1.2020.25",
```

```
            "access_str": "RX",
            "vtype_str": "f64",
            "unit": "ns",
            "desc": "The time error between a Slave Clock and a
    Master Clock (Legacy WR: clock offset)",
            "enum_dic": {},
            "metrics_dic": {
              "min": 0.000000007383,
              "max": 0.000000291743,
              "mean": 0.000000097378
            }
          }
        }
      }
    }
```


To reset the runtime metrics using the REST-API, use the `gpa/reset-param-metrics` endpoint, specifying the parameter to reset.

Please refer to "The REST-API" on page 31 section of this manual, and to the Swagger page of the API, for more information.

7.6 Synchronization Services Monitoring

The Monitoring section of the Web UI uses quantitative values (metrics) from measurable properties (parameters) to create computed statistics from raw data. This menu aims to provide visual information about how different timing interfaces perform over the time, through elements that define which parameters should be displayed as well as a time/date range selector. This is achieved by plotting different metrics regarding an interface (to be chosen by the user) into a user-friendly graph.

Figure 7-10: Monitoring section of the Web UI

This plot is created thanks to the device's database, which contains real time metrics about each interface sorted by timestamp. When a set of metrics (for a specific interface) have been captured from each service, these are redirected directly into the device's database. The plot retrieves the data from the database by using an REST API-based server running locally on the device. Please refer to "Database Schema" on page 188 for detailed information about how device's database perform.

There are two subsections available for different plotting purposes based on Timing Source and Timing Source Comparison.

7.6.1 By Timing Source

This graphing option will show the plot of a chosen metric for a given interface. The plot consists of a two-dimensional representation where the **X-axis** represents the **time dimension** (represented by the timestamps attached to each different metric) when the metric was calculated and the **Y-axis** represents the **scalar value of the metric** itself with its own unit. A custom interface and date range can be set to meet the user visualization needs.

Figure 7-11: Timing Source Plot

Legend for the Timing Source Plot image:

1. **Hide data from unsynchronized periods** switch: By default, the plot shows data from both locked and unlocked device/interface states. By clicking this switch, data shown is only from when the device was locked.
2. **Custom date range** switch: By default, the plot shows the full range available for the interface. By clicking this switch, a custom range selector appears to provide the user the possibility to define a new data range for the plot itself.
3. **Time Source** selector: By clicking this selector, a complete list of available interfaces is displayed in order to allow the user to select one of them.
4. **Date from & Date to** time date selector: When "Custom date range" switch is enabled[1] two time date selectors are available for a customized range set by the user.
5. **Refresh** button: By clicking "Refresh", The current values for Time Source and Date From & Date to selectors are used to extract again the information from the Rest-API updating the plot consequently. Therefore, the plot visualization will be updated.

6. **Plot Minimap (zoom)**: The minimap shows the complete range for the plot and allows to use the zoom feature by the user. The user can click and drag over the minimap to zoom over the desired area of the plot. The main plot will automatically adapt to the selected range in the minimap.
7. **Main plot view**: Contains the plot with the specific metric selected by "Metric selector" component in the specified time date range (latest or custom using the "Date from & Date to" selectors)
8. **Metric selector**: For each interface, the available metrics will be shown to provide the user the ability to select one of them to be depicted in the main plot view. One metric can be plotted at a time. Examples of this metrics for White Rabbit are: Offset from master, Delay MS, Delay MM, Mean Delay. For PTP we can choose over offset from master and mean delay.
9. **Resume table**: This table shows an overview of the current interface. This table contain stats about the interface for each of the metrics, including the max, min and mean value.

7.6.2 Timing Source Comparison

This subsection strengthens the previous one by allowing comparison of multiple timing sources. For such a purpose, additional selectors named Time source A, Time source B and Time source C. are provided.

Once an interface has been selected for each time source selector, the graph will show them overlayed. This warranties the correct visualization for both interfaces by using a evenly scaled X- and Y-axis.

As in the previous plot, the X-axis will represent the time dimension (represented by the timestamps attached to each different metric) when the metric was calculated. The Y-axis will contain the scalar value.

Figure 7-12: Timing Source Comparison plot

Legend for the Timing Source Comparison plot image:

1. **Time Source A / B / C**: A selector input that allow the user to choose one of the time sources to be compared.
2. **Time Source A / B / C**: A selector input that allow the user to choose one of the time sources to be compared.
3. **Custom date range switch**: By default, the plot shows the full range available for the interface. By clicking this switch, a custom range selector appears for a customized date range.
4. **Date from & Date to** time-date selector: When "Custom date range" switch is enabled [3] two time-date selectors are available for a customized range set by the user.
5. **Refresh** button: By clicking "Refresh", the above mentioned options selected by the user are applied. Therefore, the plot visualization will be updated.
6. **Plot Minimap (zoom)**: The minimap shows a simplified view of the current plot. The user can click and drag over the minimap to zoom over the desired area of the plot. The main plot will automatically adapt to the selected range in the minimap.
7. **Main plot view**: Contains the overlayed plot of each of the metrics selected for each interface.

8. **Resume table**: This table shows an overview of the current interface. This table contain stats about the interface for each of the metrics, including the max, min and mean value.
9. **Hide data from unsynchronized periods** switch: By default, the plot shows data from both locked and unlocked device/interface states. By clicking this switch, data shown is only from when the device was locked.

BLANK PAGE.

CHAPTER 8

Device Maintenance

The following topics are included in this Chapter:

8.1 Licenses

Some additional features require a specific license in order to benefit from their full potential. This section will provide a quick guide on how to:

- Buy a new license.
- Install a new license.
- Check if license has been activated.
- Perform maintenance on a license.

8.1.1 List of related Licenses

The available feature licenses related to WR-Z16 are:

Group	Feature Name	Description
PTP	ptp_profile_cfg	Unlock the configuration of different options to enable other profiles than the default such as the telecom profile ITU-T G.8265.1, ITU-T G.8275.1, and power profiles IEEE C37.238-2017 and IEEE 61850-9-3.
HATI	hp_port	Enable High-Performance HATI support for a given port. The HATI (High Accuracy Timing IP) is a FPGA core designed to easily integrate high-accuracy timing into Xilinx FPGA. Please, contact with info.spain@nav-timing.safrangroup.com for more information.

8.1.2 Check Licenses

The status of the licenses on the device can be retrieved under: **Management** > **Licenses** as shown in the figure below:

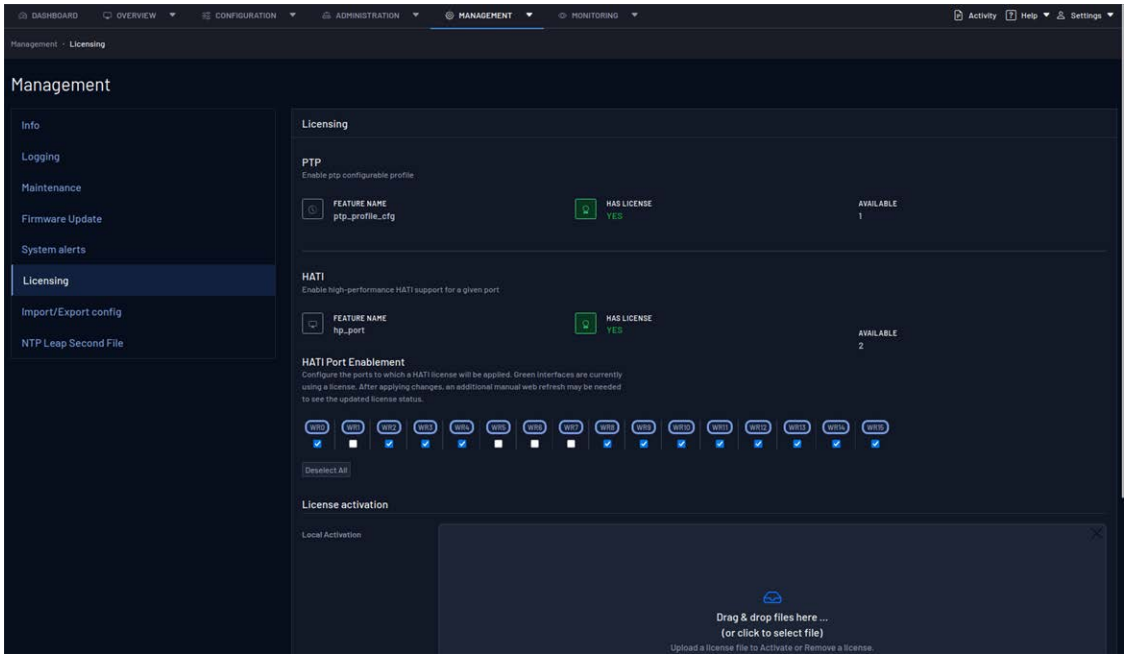


Figure 8-1: Checking available licenses.

Each possible license available for the WR-Z16 devices is represented by a single box where its status is summarized by the color of the box:

- Red: License is not available.
- Green: License is properly activated.

OID	Name	Value Type	Description
12.2xyy.	**licenses/xxx/yyy/**		**Information about license feature <yyy> in group <xxx>.**
12.2xyy.1	Feature Name	<String>	License feature name.
12.2xyy.8	Available	<Integer>	Total of available corresponding licenses by device (i.e., features associated to port might need up to 16 licenses).

OID	Name	Value Type	Description
12.2xyy.15	Description	<String>	Description of the corresponding license feature.
12.2xyy.16	Used	<Integer>	Total of used licenses of this type by device.
12.2xyy.20	Has License	NO YES	Status of the license for the corresponding feature.

8.1.3 Order Licenses

Usually, the licenses are purchased together with the devices. This eases the ordering and installation procedure.

In some cases, additional licenses must be purchased afterward: The recommendation is to contact the corresponding FAE in order to receive assistance during this procedure. Alternatively, contact info.spain@navtiming.safrangroup.com for a quotation.

Once the purchase has been confirmed, an email will be sent providing the credentials to access the Seven Solutions Licenses Portal.

Note: For security reasons, the generated temporary password expires quickly. Click on “Forget password” in case it was already expired.

8.1.4 Local Licenses Management

In order to perform local licenses management by directly uploading licenses files to the device the user first needs to login to the license portal by navigating to the following link:

https://flex1667.flexnetoperations.com/flexnet/operationsportal/logon.do

8.1.4.1 Map a feature to a device

» The user must first navigate to the tab **Devices** > **Devices**.

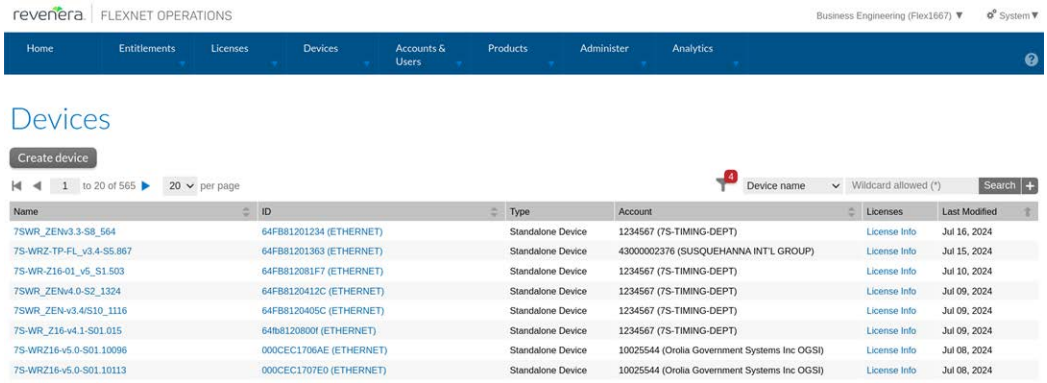


Figure 8-2: Devices Management in License Portal.

» Then, click on the corresponding device to map the new feature. If this device does not appear in the list, first create it (See the following section, Create a New Device).

» Once inside the corresponding device, click on Action > Map Entitlements to perform the association.

» This panel (Figure 8:3) allows to map any purchased licenses to this specific device. The user only needs to specify the quantity of a given feature license to associate to the device and save it.

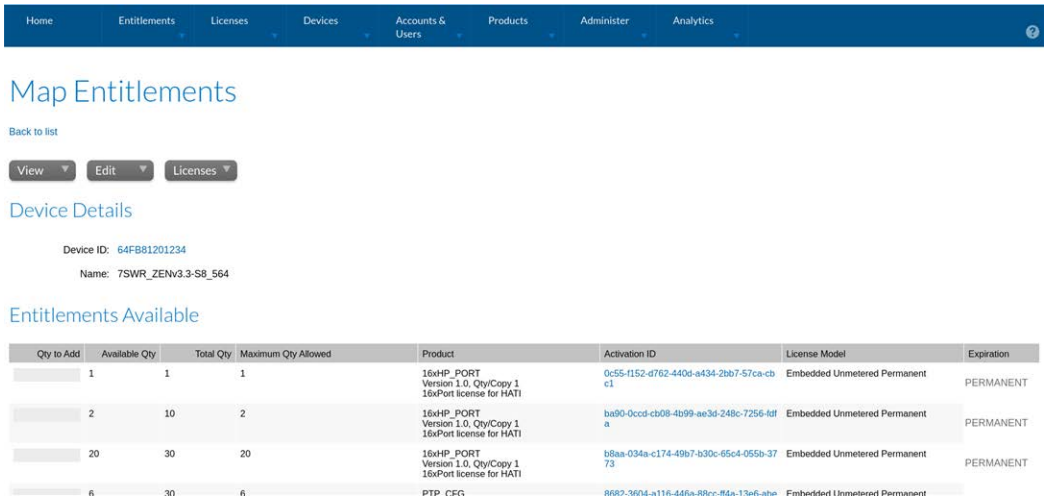


Figure 8-3: Mapping Purchased Licenses to Device.

» At this point the license(s) is(are) associated but still not generated yet. By clicking on the Action > Download Capability Response a license file (.bin)

will be automatically generated and downloaded. After refreshing the screen, the corresponding Status should be updated to License generated.

> **Note:** A license file will be generated using a <DEVICE_ID>.bin file-name pattern. Do not rename this file otherwise it will not be properly recognized when loading it to the device. If a <DEVICE_ID>.bin file is already present in your Download folder, the new generated file will be automatically renamed with a prefix (i.e., <DEVICE_ID> (1).bin). Please remove this prefix before uploading the file.

8.1.4.2 Create A New Device

- First select the **Create Device** button.
- Then fill the following parameters:
  - Name: unique device name on the network. It is recommended to use the same name as the hostname.
  - Run Licenses Server: Disabled
  - ID Type: ETHERNET
  - ID: It corresponds to the eth0 physical address (MAC) of the device.

> **Note:** The device ID format is based on the eth0 MAC address, but without the doble-dot (:) and with only upper-case characters. It can also be obtained from a terminal by executing:
> `gpa_ctrl hald net/eth0/ethaddr | sed 's/://g'`

8.1.4.3 Load local license file in the device

In order to load the generated license file, the user first needs to navigate to **Management** > **Licenses** within the web interface of the device.

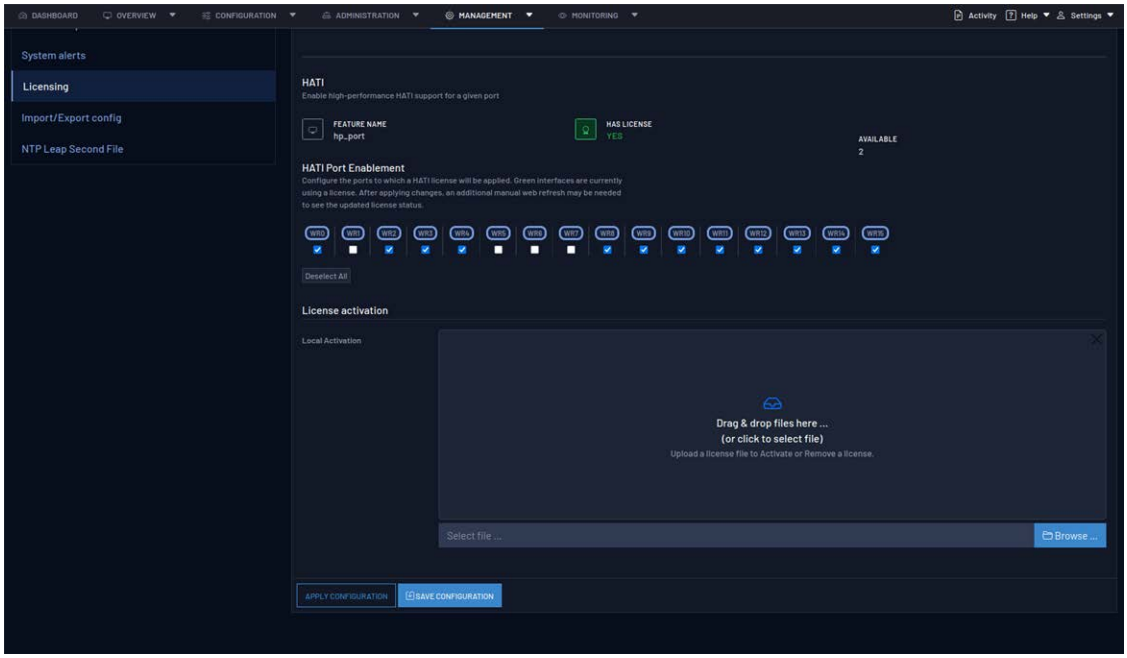


Figure 8-4: Licenses Configuration Panel

Then under Local Activation, the user should Browse to the downloaded license file, or drag and drop the file into the labeled box. Once dropped, the user can select Upload.

Once the operation is done, the user can review if the license has been properly activated (Green) by returning to the Management > Licenses screen.

8.1.4.4 Remove local license from device

In case a local license needs to be used in another device the user should first remote it from the previous device before associating it to the new one.

- Access Seven Solutions Flexera Portal.
- Go to the “previous" device.
- Click on Action > Remove Licenses.
- Select the quantity of the corresponding license to remove and Click Save.
- Review that the Status of the licenses to remove is Waiting for confirmation.
- Then click on Action > Download Capability Response.
- Upload previously downloaded file as described in the previous section, "Load local license file in the device" on the previous page

> **Note:** If <DEVICE_ID>.bin file is already present in your Download folder, the newly generated file will be automatically renamed with a suffix (i.e., <DEVICE_ID> (1).bin). Please remove this prefix before uploading the file.

- If everything works as expected, a <DEVICE_ID>.bin.confirmation should be generated back.
- Return to Seven Solutions Flexera Portal.
- Click on Devices > Offline Device Management.
- Select the Generated License option and then select the previously downloaded <DEVICE_ID>.bin.confirmation to Upload it.
- Finally review if the license has been properly unlinked from your device. If it is the case, this means that the corresponding purchased license can now be mapped again to any other device.

8.1.4.5 HATI License Manager

The HATI License Manager (under **Management** > **Licensing**) is a graphical user interface tool that provides control for the interfaces that will be assigned a HATI License. This way, the user can conveniently select the specific interfaces they need to assign a HATI license and see a visual representation of which interfaces are currently using licenses. A quick overview of the HATI License Manager interface is shown below.

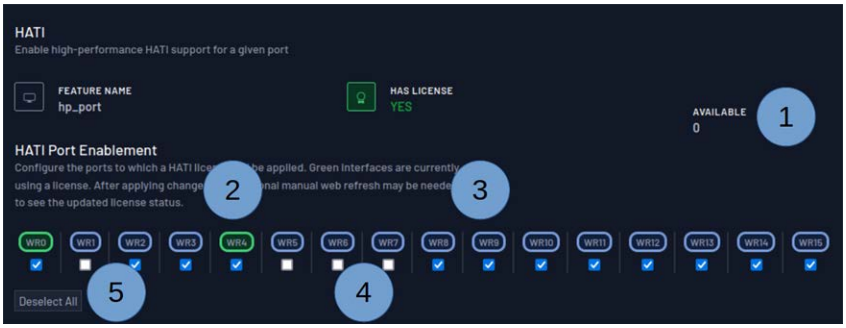


Figure 8-5: HATI License Manager interface

1. Number of port licenses available (Not to be confused with the main HATI license. One main HATI license can contain multiple port licenses).

2. Green indicator. The green indicator shows which interfaces are currently assigned a HATI license.

3. Blue indicator. The blue indicator shows which interfaces are not currently assigned a HATI license. A port without a license will not be able to communicate with a HATI.

4. License assignment(or checkbox array). Each of the interfaces will be paired with a checkbox. This checkbox allows the user to decide which interfaces will be able to be assigned a HATI license. If a checkbox is not enabled for a certain interface, this interface will not be assigned a HATI license.

5. "Select/Deselect All" button. This button allows the user to select/deselect all checkboxes. This is especially convenient when having a device with an elevated number of interfaces.

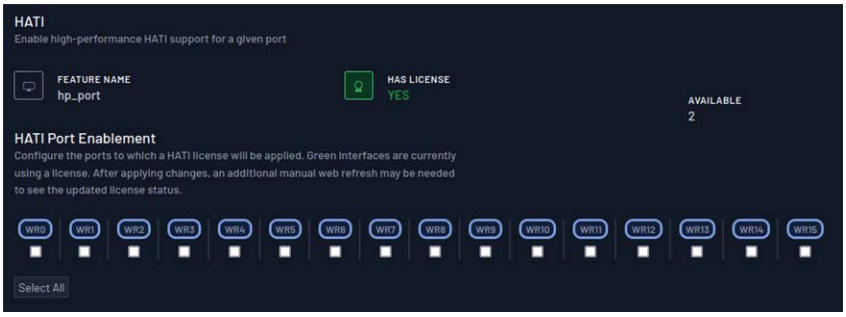
How to use HATI License Manager

Note: A checked box does not fully determine whether an interface is assigned a HATI license. This happens because there are a limited number of port licenses available.

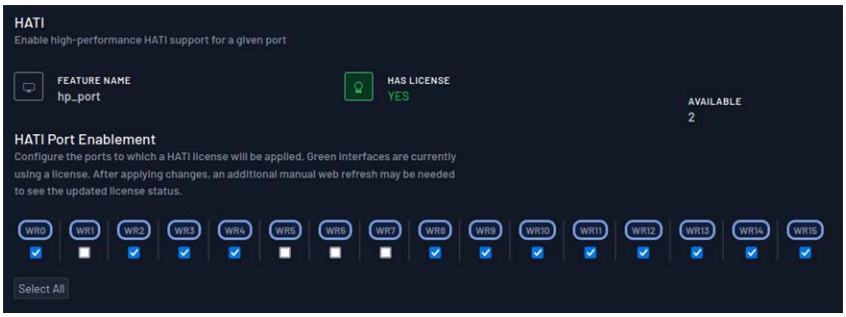
The following section will go over a typical configuration process where we have the following:

- A main HATI license that provides 2 port licenses.
- A HATI connected to each interface's end and each interface has an "up" state (these are requirements for an interface to accept a license)

In this example we start by unchecking all checkboxes and selecting the "Apply Configuration" button. After refreshing the page, it should look similar to the page shown below.

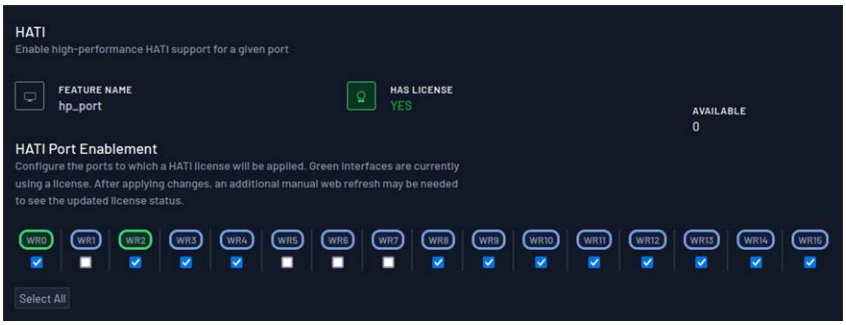


We can now select any number of interfaces we would like to be capable of being assigned a HATI license. In this example we have the following scenario:

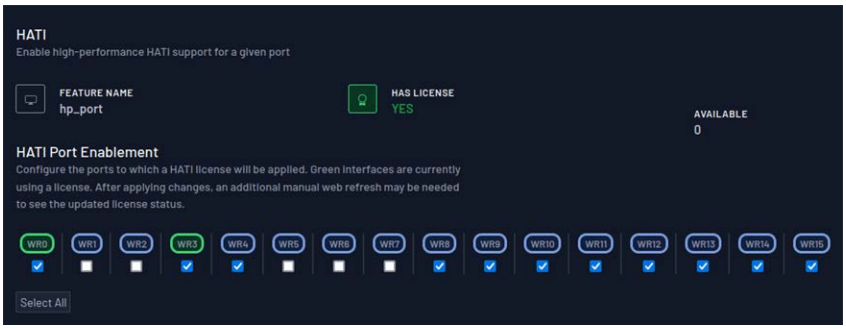


After selecting the "Apply Configuration" button, the page will automatically refresh. It can be observed that the number of available licenses has decreased to 0 and the two lowest-numbered interfaces that were enabled (checkbox status: on) have been assigned a license.

Note: If the above has not occurred, please manually refresh the page again (sometimes the license engine takes some time to apply changes) and make sure a HATI is connected on each interface's end.



If we uncheck interface number 2 (WR2), select the "Apply Configuration" button and refresh the page, the next lowest-numbered enabled interface will be assigned a HATI license.



If interface number one (WR1) was enabled, it would be assigned the license instead of interface number three (WR3). This happens because the licenses are assigned to the lowest-numbered enabled interface.

When the configuration is applied with the "Apply Configuration" button, the current checkboxes statuses are retained within page refreshes. When the configuration is saved with the "Save Configuration" button, the current checkboxes statuses are retained within device reboots.

Note: After selecting the "Apply Configuration" button, an automatic page refresh is performed. A manual page refresh may be required if the licenses' status have not been completely updated.

Note: The "Save Configuration" button will not actively apply any changes. If the configuration is saved, a complete device reboot will cause the changes to apply.

8.2 Firmware Update

There are two different ways to update the software and firmware of the device: through the web interface or by using SSH/SCP.

Note: Matching device hardware: Safrans distributes different WRZ-OS firmwares according to the hardware family of the device. The user must follow indications provided in "Hardware version and firmware" on the next page to get the corresponding firmware before proceeding to its update.

> **Caution:** The configuration is NOT compatible between major versions. If the major version changes (for example from 3.X to 5.X or vice-versa), the configuration on the device must be removed and configured again.

> **Caution:** HW/SW compatability: For hardware versions higher or equal to 5.0, only software versions higher than 3.4 will be supported.

8.2.1 Hardware version and firmware

The HW version is displayed on the dashboard page in the Versions overview panel. The device shown below is a WR-Z16 that mounts a Z16v4.0 as main board.

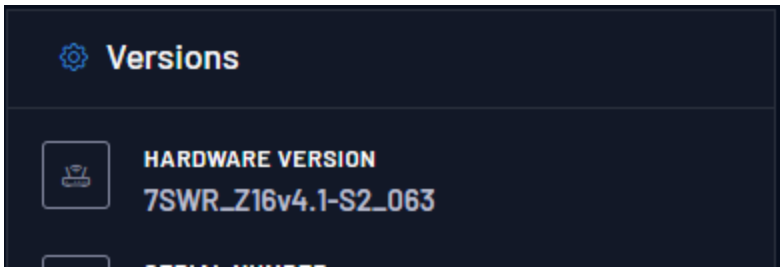


Figure 8-6: HW version displayed in dashboard.

The WR-Z16 can be updated using the firmware that matches the version of the main board, or the new generic family firmware developed from the 3.4 software version:

- `wr-zynq-os-v<XXX>-<YYYMMDD>-Z16x.x_binaries.tar`
- `wr-zynq-os-v<XXX>-<YYYMMDD>-Z164.x_binaries.tar`
- `wr-zynq-os-v<XXX> -<YYYMMDD>-Z16_binaries.tar`

But not the one that correspond to another device:

- ~~`wr-zynq-os-v<XXX>-<YYYMMDD>-ZEN3.x_binaries.tar`~~

Or the one that corresponds to another hardware version

- ~~`wr-zynq-os-v<XXX>-<YYYMMDD>-Z162.x_binaries.tar`~~

8.2.2 Using Web interface

Once the web GUI of the device has been properly opened (See "Connecting to the Device" on page 20), navigate to the Management > Firmware Update panel.

- The corresponding firmware tar ball can be drag-n-dropped, or you can Browse from the PC.
- Then, press on Upload button and wait until checking the compatibility of the given firmware. If the firmware is detected to be compatible it will automatically start the upgrade procedure and reboot (twice) the device. Please wait in this screen until the procedure completes.

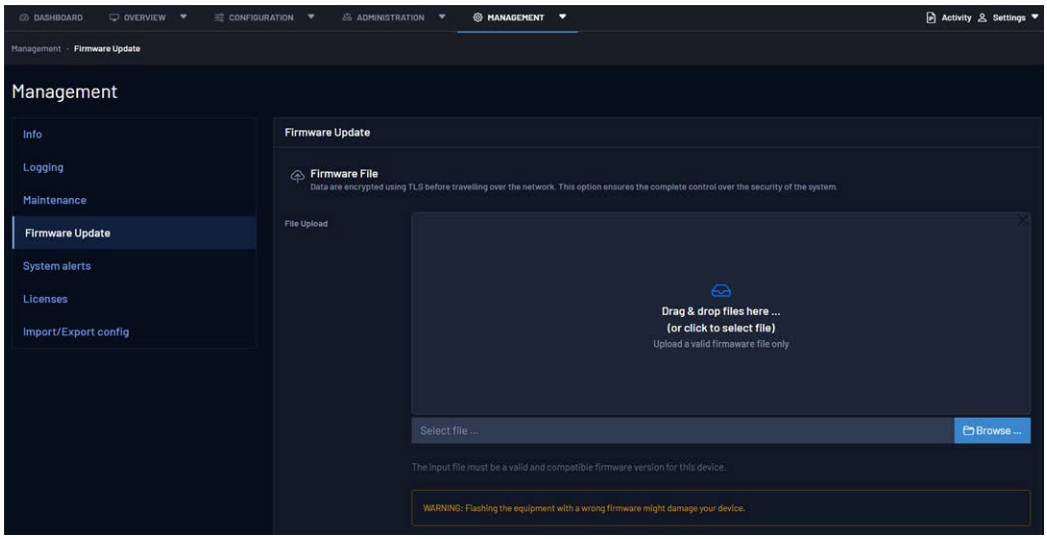


Figure 8-7: Update Procedure Waiting screen.

- If the uploaded version is lower than 5.5, a factory reset is mandatory. If this is the case, a message will appear indicating this.
- If an incompatibility (figure below) has been detected, the user should NOT continue with the flashing procedure except if the support team has confirmed that this is the way to fix a specific problem.
- If the hardware version is higher or equal to 4.0 and the software version is lower than 4.0, flashing will not be possible and a warning will be given.

8.2.3 Using SSH/SCP

A new firmware can be updated using SSH and SCP protocols. This method allows a secure way to perform a batch firmware update to many devices at the same time.

The first step of this method is to upload the corresponding firmware to the root folder of the device using SCP:

```
scp wr-zynq-os-v3.2-RC1-20210325-ZENv3.x_binaries.tar
root@<deviceip>:~
```

Then login to the device with SSH:

```
ssh root@<deviceip>
```

And finally run the wrz_flashfw tool to handle updates with the reboot flag if no errors were detected:

```
root@be-dist8-684:~# wrz_flashfw -r ~/ wr-zynq-os-v3.2-RC1-
20210325-ZENv3.x_binaries.tar
```

If the uploaded version is lower than 5.5, a factory reset is mandatory. If this is the case, the following message will prompt after executing the previous command:

```
Warning: The version to be flashed is older than the cur-
rent one, therefore the default password and user are going
to be set. The device will be automatically rebooted...
```

```
This will reset the device in its factory version and all
your modifications will be lost
```

```
Do you want to continue? [y/N] ?
```

Note: Please check wrz_flashfw -h to get more information about the various arguments accepted by this tool.

8.3 Recovery Mode

If an error has occurred (e.g., power down, wrong firmware) during a firmware update procedure the device might not be able to boot from the SD card and will enter itself into a recovery mode.

This recovery mode consists of a minimal Linux stored into internal memory of the equipment that allows to:

- Reflash the device with another firmware.
- Recover configuration (if possible).
- Clean/format SD remotely.

Once the device has been booted in recovery mode, it should apply the network configuration previously saved in the .config file. However, it might occasionally be impossible to recover the network configuration. In those cases, the device will be accessible using the default network parameters ("Default Configuration" on page 20) or through front USB-UART serial connection.

The following actions might be considered to try to repair the device. It is recommended to try them in the given order:

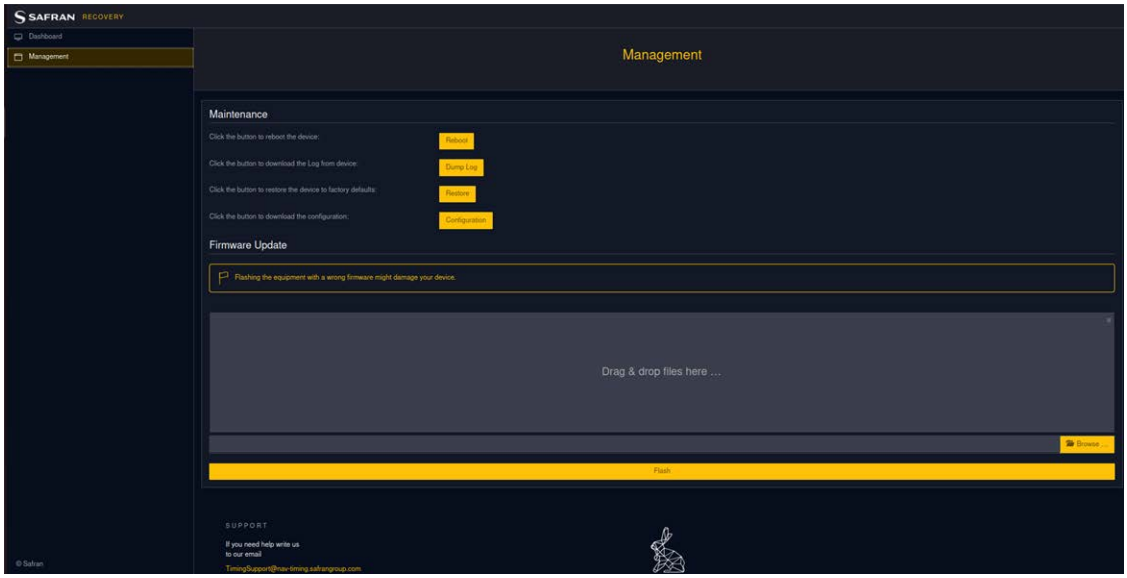


Figure 8-8: Management panel in recovery mode

1. **Configuration**: One of the first things to do when a device is in recovery mode is to try to back-up its configuration, so it is easy to import it back or load it to another device.
2. **Reboot**: Then, try to reboot the device as the recovery mode has already performed an automatic filesystem check and cleaning. If the device reboots in normal mode, this means that the error in SD partition has been automatically fixed, otherwise another recovery action might be executed.
3. **Firmware Update**: Try to flash the firmware again (this is the most frequent action to perform when an error has occurred during the flashing procedure).
4. **Restore**: Remove any customization and restore the device to its default values (WARNING: Any specific network settings will be removed).

If none of these actions can return the device to a normal booting mode, contact "Technical Support" on page 232 to get more help.

8.3.1 Manual recovery mode

8.3.1.1 Using reset button

In case the recovery mode must be entered manually, the following steps need to be performed:

1. Reboot the WRZ device.
2. Press the reset button 2.1 Front panel0) around 5s while the device is booting and release the button when the status led is blinking.
3. The status LED should light red (See "Monitoring LEDs" on page 10).
4. Wait until the recovery image is loaded from QSPI dataflash (This can take more than 1 minute).

8.3.1.2 From Serial UART

The recovery mode can also be started from Uboot console (Connected to the serial RJ45-UART) when it is not possible to access the reset button:

1. Press any key when seeing:

```
Loading wr7shw preboot...
U-Boot 201X.xx-wr7s-vX.X (Jun 25 2018 - 16:07:12) ZENv3
WR_ZEN-vx.x-Sxx_xxx
Hit any key to stop autoboot: 0
```

2. Execute:

```
wr7s-uboot> env run recoveryboot
```

3. Wait until the recovery image is loaded from QSPI dataflash (This can take more than 1 minute).

8.3.2 Recovery Upgrade Process

The following procedures will upgrade your unit's recovery partition binaries.

Caution: Do not follow the next steps if the device is still running in recovery mode. Updating the recovery mode without exiting first can corrupt the device's memory. Make sure the system rebooted correctly and that the new firmware version was correctly flashed.

Software version lower than v3.4

1. Download the release tar file and extract the content of the recovery tar file located in the recovery folder.
2. Copy the content of the recovery image to the device and execute the following commands:

```
root@XXX:~# flashcp -v BOOT-v3.x.bin /dev/mtd0 && flashcp -v
uImage /dev/mtd1 && flashcp -v devicetree-v3.x.dtb /dev/mtd2
&& flashcp -v uramdisk.image.gz /dev/mtd3
```

Note: the name of BOOT-vx.bin may change depending on the version used.

Software version v3.4 and higher

1. Download the release tar file and check that the recovery folder only contains a file called "uramdisk-recovery.image.gz".
2. Once the device has been flashed with this new release, you can flash the recovery with the following commands, where the BOOT.bin, uImage and devicetree.dtb files are included in the boot partition instead of the recovery folder:

```
root@XXX:~# flashcp -v /media/boot/BOOT.bin /dev/mtd0 &&
flashcp -v /media/boot/uImage /dev/mtd1 && flashcp -v /me-
dia/boot/devicetree.dtb /dev/mtd2 && flashcp -v /me-
dia/boot/recovery/uramdisk-recovery.image.gz /dev/mtd3
```

8.4 SD Recovery Tool

If a situation occurs where the SD card is corrupted or cannot be accessed and the system is unusable, the device will boot in a minimal version (recovery).

This tool is intended to fix any format errors in SD partitions. WR-Z16 products provide a solution for doing such tasks using either the CLI or the WebUI (recommended) in recovery mode without extracting the SD card.

Everything will be lost. Make sure to backup your data in the `/media/data` and `/root` folders before performing the recovery operation.

8.4.1 Formatting the SD thought the CLI

While in the device recovery mode prompt, as shown in the figure, you can invoke the `recover_sd` script that will automate the SD format process.

```
root@zen-305:~# cd /
root@zen-305:/# recover_sd
SUCCESS: SD card found in /dev/mmcblk0
    ---------------------------------------------
    ------------------ CAUTION ------------------
    ---------------------------------------------

    If you proceed you will:
    - Erase all the data in the /dev/mmcblk0 SD card
    - Create 1GiB FAT32 partition as BOOT
    - Create an EXT4 partition as DATA, taking the remaining space.

    THIS WILL ERASE ALL THE DATA IN THE DISK

    Do you want to continue? (Yes/No)
```

Figure 8-9: Recover_sd script

The script will run when the user is located in the / path

After running the script, users should see something as follows to check that the recovery process was completed after running the script.

```
    Do you want to continue? (Yes/No) yes
Unmounting all partitions on the device...
Creating new MSDOS partition table and partitions...
 mmcblk0: p1 p2
SUCCESS: SD partition table and partition created successfully
Waiting for the system to recognize the new partitions...
Formatting the first partition as FAT32...
Cannot initialize conversion from codepage 850 to ANSI_X3.4-1968: Invalid argument
Cannot initialize conversion from ANSI_X3.4-1968 to codepage 850: Invalid argument
Using internal CP850 conversion table
SUCCESS: Partition BOOT created successfully
Formatting the second partition as EXT4...
mke2fs 1.46.5 (30-Dec-2021)
SUCCESS: Partition DATA created successfully
EXT4-fs (mmcblk0p2): mounted filesystem with ordered data mode. Opts: (null)
SUCCESS: Paritions mounted in filesystem.
All operations completed successfully.

Now you can load the new firmware using the command line or the WebUI.
```

Figure 8-10: Recover_sd script confirmation

After seeing this prompt, users can load the firmware through the WebUI.

8.4.2 Formatting the SD though the WebUI

This is the recommended way to do the recovery since it is straightforward and is easy to load the firmware afterward.

To format the SD thought the WebUI go to the **Management** tab and click the **Format SD** button.

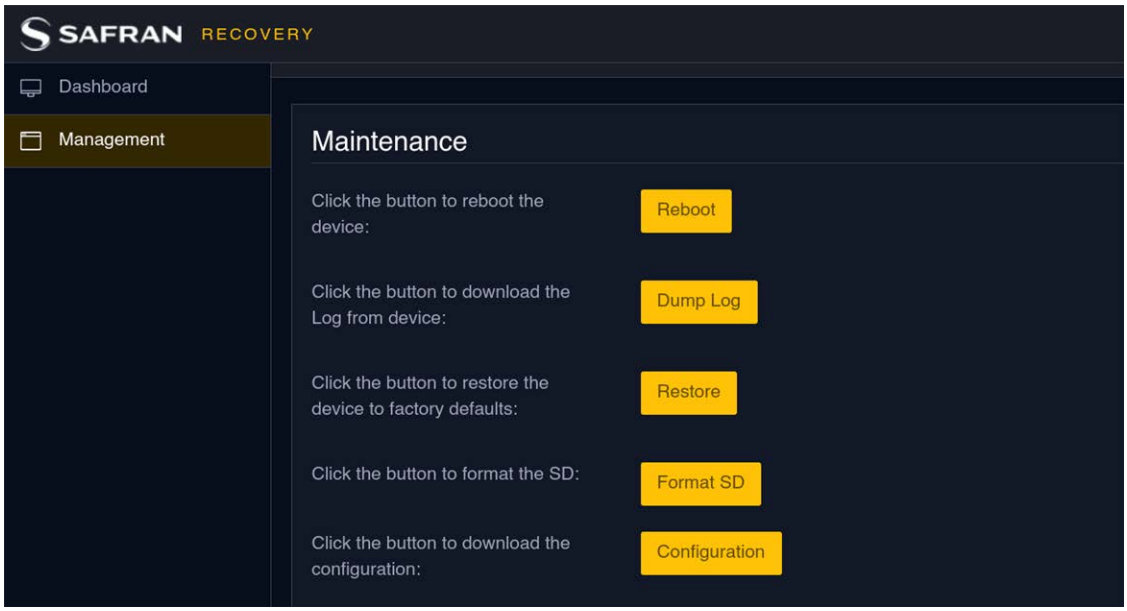


Figure 8-11: Recovery mode management tab

The WebUI will show a confirmation pop-up as shown:

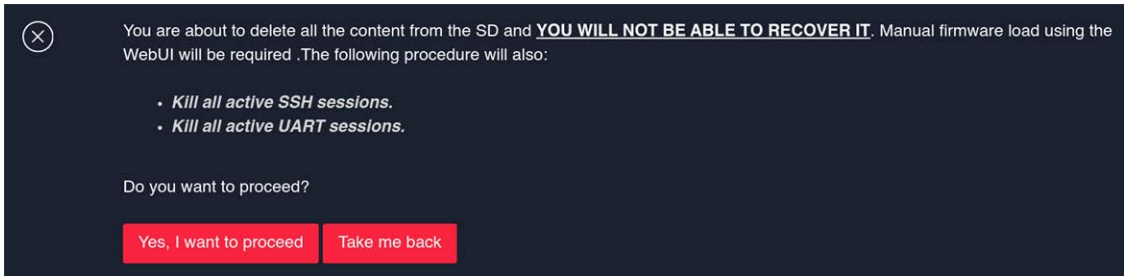


Figure 8-12: Recovery mode management confirmation dialog

After completion, users should see a message confirming that the process completed successfully, device is ready to load a new firmware:

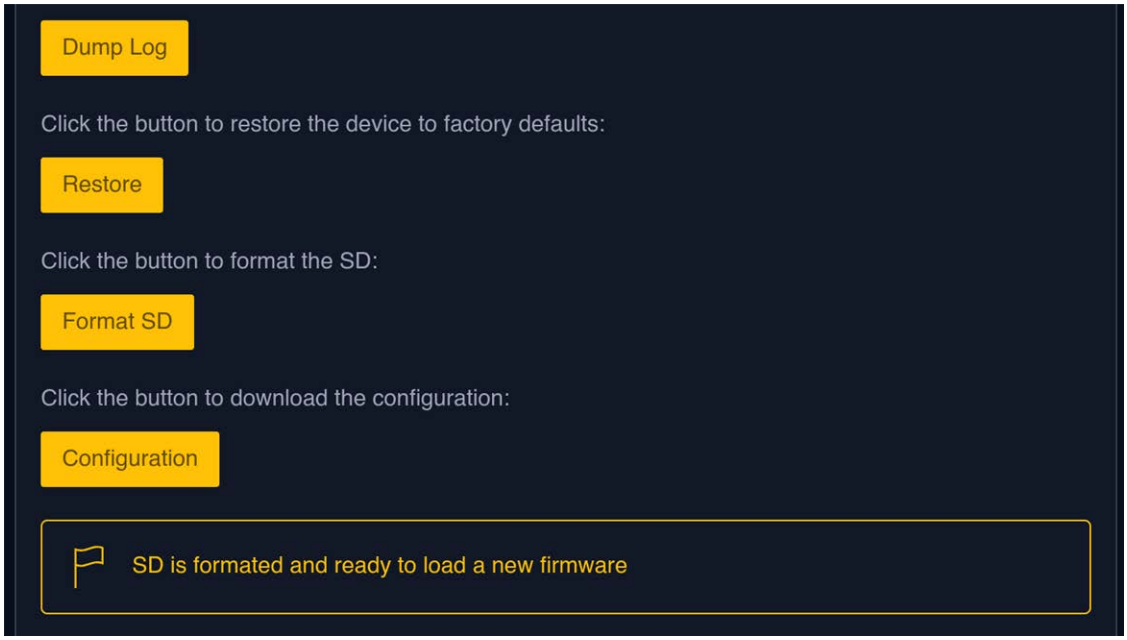


Figure 8-13: Recovery tool success message

8.5 Factory Config Mode

In case a miss configuration of the device invalids its correct login, one can manually reset the configuration to default factory value by using one of the following methods.

Note: The factory config mode does not revert the device to its factory firmware. It only removes all the configurations and customizations stored by the user and will reboot the device using a clean version of the last firmware flashed.

Note: Performing a factory reset erases IP address configuration, so serial access is mandatory to reconfigure remote access (web and SSH communication).

8.5.1 Reset via Front Panel Controls

1. Reboot the WRZ device.
2. Press the CTRL/Info button more than 15s while Uboot is loading.
3. Hold until the status LED lights are yellow:  (See "Monitoring LEDs" on page 10).
4. Wait until the device reboot with default factory parameters.

8.5.2 Reset via the CLI

To perform a CLI Factory Reset, serial access is required.

Execute a power cycle. Once the unit is just powered on, press the Reset button on the front panel (use a thin tool) until the following message displays in the CLI:

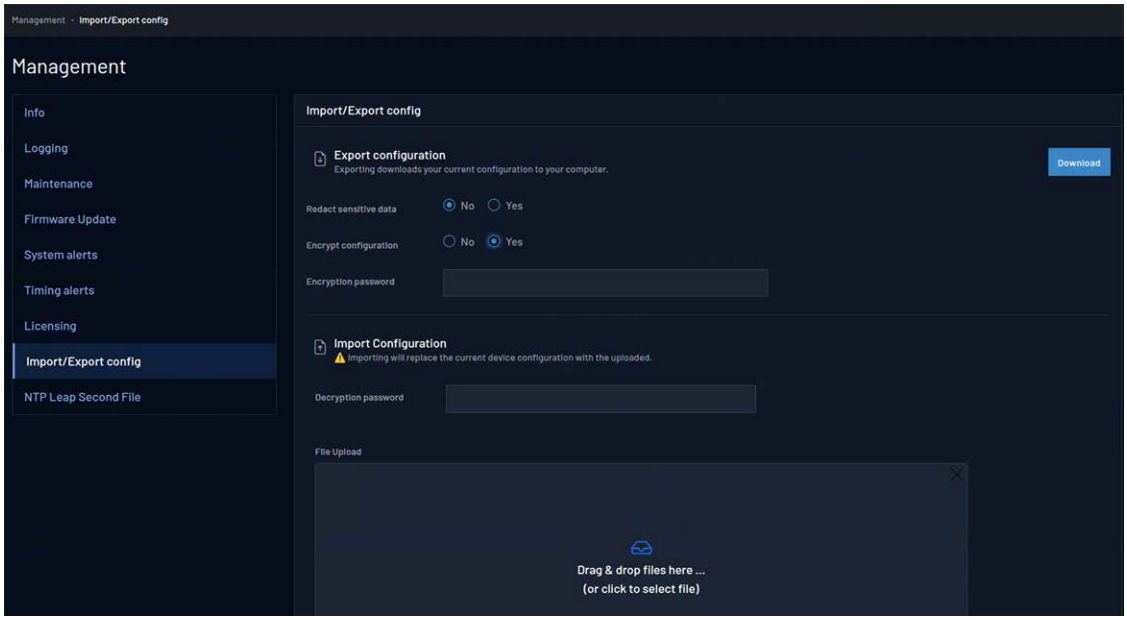
In the CLI:

```
U-Boot 2015.04-wr7s-v2.10-z16 (Apr 26 2022 - 10:51:49) Z16v4
WR_Z16-v4.1-S01_183
Environment found in flash
CTRL button is pressed!
.....
Setting reset factory boot mode
```

To complete the factory reset, hold the Reset button for 20s.

8.6 Importing/Exporting Configuration

The WR-Z16's device configuration can be imported, exported, and restored to default settings. From the web GUI, navigate to **MANAGEMENT** > **Import/Export config**.



8.6.1 Exporting

To export device configuration through the web GUI, select the "Download" button. To do so from the CLI, execute the following command:

```
wrz_loadconfig -e -o /root/
```

To protect sensitive data, the configuration file may be redacted and/or encrypted prior to being downloaded.

To redact sensitive data from the web GUI, select "Yes" under the "Redact sensitive data" setting. To do so from the CLI, execute the following command:

```
wrz_loadconfig -e -o /root/ -redacted
```

To encrypt the configuration file from the web GUI, select "Yes" under the "Encrypt configuration" setting and input your desired password in the "Encryption password" field. To do so from the CLI, execute the following command:

```
wrz_loadconfig -e -o /root/ -encrypted <password>
```

where <password> corresponds to your desired encryption password.

8.6.2 Importing

To import device configuration through the web GUI, upload a configuration file by dragging and dropping into the "File Upload" box or by selecting the "Browse" button and locating your desired file. With a file selected, select the "Upload" button. If the file is encrypted, input the encryption password into the "Decryption password" field before uploading.

8.6.3 Restore Configuration

Default configuration settings can be restored by selecting the "Restore" button. This will delete your current configuration and restore default settings.

8.7 Failsafe Mode

The Failsafe mode allows to only load the minimal Linux services (i.e., logging, network, ssh, web) but using the normal firmware stored in SD card. It has been mainly designed for advanced users that might have blocked the startup of the device through a bad customization of init.d services.

So, if after a failed customization, a device does not provide a usable access to its console (ssh or UART), the failsafe mode can be entered by following the procedure:

1. Power cycle the device.
2. Wait 30 second until the kernel starts loading.
3. Press Reset Button (#2) for more than 30s until Status LED (#3) starts blinking several times in yellow. This mean that the failsafe mode has been triggered.
4. Remove/fix the custom scripts that were blocking the OS initialization.

Note: Factory reset vs. Failsafe mode: If the device initialization is blocked due to a custom script, it might be easier to directly perform a factory reset even if this means that the device will lose all its configuration.

BLANK PAGE.

APPENDIX

Appendix

The following topics are included in this Chapter:

9.1 Acronyms

Acronyms	Description
BC	Boundary Clock (Disciplined by a master and discipling slaves)
BMCA	Best Master Clock Algorithm
FR	Free Running (Undisciplined local oscillator)
GLONASS	Globalnaya Navigatsionnaya Sputnikovaya Sistema
GM	Grand Master
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HA	High Accuracy
HO	Hold-Over
HTTP	Hypertext Transfer Protocol
NMEA	National Marine Electronics Association
NMS	Network Management System
NTP	Network Time Protocol
PPS	Pulse Per Second
PTP	Precision Time Protocol
PPS	Pulse Per Second
PWS	Power Supply
RTT	Round Trip Time
SFP	Small Form-factor Pluggable Transceiver for fiber link
SSH	Secure Shell
SNMP	Simple Network Management Protocol
SyncE	Synchronous Ethernet
TAI	International Atomic Time (Temps Atomique International)
ToD	Time of Day
UTC	Coordinated Universal Time
WR	White Rabbit

Acronyms	Description
WR-ZEN	White Rabbit Zynq Embedded Node
WR-LEN+	White Rabbit Lite Embedded Node Plus
WRZ-OS	White Rabbit Zynq based (Z16, ZEN, LEN+) Operative System

9.2 Troubleshooting

This section intends to help the user understand how to identify an issue in your WR-Z16 device, as well as giving some guidance to figure out the cause of the problem.

9.2.1 Frequently asked questions (FAQ)

A list of the most commonly asked questions will be described here, as well as the solutions that can be applied for each of the situations.

- **Why does the WR-Z16 report link down even if the SFPs are connected to the WR interfaces?**

  One of the most common cause of this issue is related to not using matching blue-violet SFPs.

  As in White Rabbit it is of uttermost importance to have equal cable lengths in both directions, a single fiber should be used for sending data both directions.

  Additionally, White Rabbit should follow the 1000BASE-BX10 standard and use 1310/1490 pairs with a single LC connector. More specifically, the Switch ports transmitting downstream (to endpoints) should use 1490nm on the transmitter and 1310nm on the receiver.

  The 1310 nm module corresponds to the blue color and the 1490 nm to the purple one.

- **What does the Error 500 mean while uploading the firmware? What is recommended to fix this issue?**

  In this case, the best modus operandi is to reboot before flashing the device. Also check the file is not corrupted.

» **How can you confirm that external PPS/10 MHz signals are being detected?**

At the WR-Z16's console, type:

```
root@z16-006:~# gpa_ctrl hald /spll/ext/fpanel/sig_detected
PPS & CLK
```

The expected output is PPS &CLK which means that both signals are detected.

9.2.2 Health general status

In order to check the device general status, there are multiple alternatives:

1. SSH/UART
2. Web interface
3. SNMP queries

SSH / mini-USB UART

If you connect to the device via SSH/UART you will be able to check the WR-Z166 sync status by typing:

```
gpa_ctrl healthingd
```

```
gpa_ctrl tmgrd vclock/info/
```

The WR-Z16 web interface

You can access the WR-Z16 graphical interface by setting the device an IP and copying its address into the browser's URL bar.

General ports, mode and other configuration can be consulted or changed in the web.

SNMP.

This is the recommended alternative for monitoring purposes. Follow the steps on the attached Monitoring Tools User Guide to get SNMP working.

You will be able to both consulting or changing configurations on the WR-Z16 with the SNMP commands.

After having installed SNMP on your host, all parameters on the WR-Z16 can be checked by typing:

```
$ snmpwalk -v2c -c public <WR-Z16 IP>
```

9.2.3 Virtual State Clock Code Error

Virtual State Clock (VSC) Code Errors are codes that refer to different virtual clock states for debugging purposes.

The format for these codes is the following one:

`VSC-XXXXX`

These codes provide information on the synchronization status in the device. For further details, please read the VSC Code Error table ("VCS Code" on page 233) containing all the possible values, each of them referring to a different condition.

9.2.4 HTTPS Firefox Error

The HTTPS website with self-signed certificates does not work with Firefox.

By design, Firefox requires accepting SSL certificates for every URL:PORT pair, even if the certificate has already been accepted for a different port. This behavior prevents the website from working unless the certificate is accepted for both ports. To do so, first access the API Swagger website on https://<MACHINE-IP>:8201 and accept its certificate, then access the administration website and accepts its offered certificate as well (see figure below).

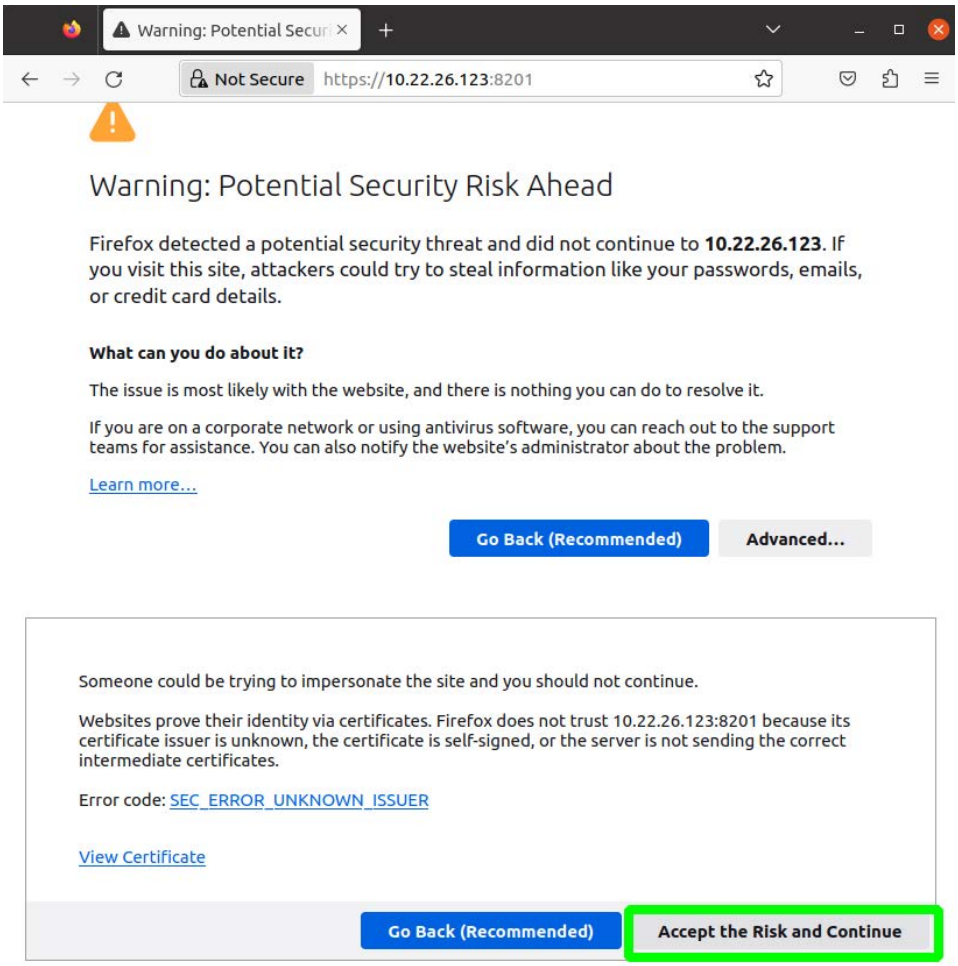


Figure 9-1: Accepting an untrusted certificate in Firefox.

9.2.5 How to report an error

1. If one of the devices experiments any technical issues, it would be recommendable to contact the Safran Support Team (see "Technical Support" on page 232), which will be in charge of addressing the problem. These are the steps that should be followed in case a problem happens.
2. If the device is alive and accessible, please go to the WR-Z16's web interface -> Management -> Download device's log dump.

3. Write to our Support Team at TimingSupport@nav- tim- ing.safrangroup.com. Describe the issue found going into details.
    a. What was the device's main activity before the error occurred? (e.g. the device was acting as a GM taking PPS/10MHz references from another one and running as a PTP master on interface wr0).
    b. Were any relevant actions previously performed on the device before the issue happened? (e.g. upgrading firmware or applying any specific configuration)
    c. Is the issue reproducible? Does it happen after specific actions are applied to the device or when a series of particular events happen in it?
    d. Attach the device's log dump if it was possible to retrieve on step 1.
4. Our Support Team will open a case (find it on the replies' email subject) in order to find out the possible causes on the issue and will give guidance so it can be solved.

9.2.6 Rsyslog template to improve remote login

Logging in the WRZ-OS is managed by the rsyslog daemon, which is in charge of both storing the events happened in the WRZ-OS in the internal /var/log/xxx files and sending them to a centralized rsyslog server.

In order to get rsyslog daemon sending all logs to a server, /etc/rsyslog.conf should be configured with the following lines:

```
module(load="imfile")

input                                                    (type="imfile"
File="/var/log/systemlog"
Tag="custom"
Facility="local0"
Severity="info")

local0.* @<rsyslog server1 IP>

local0.* @<rsyslog server2 IP>
```

9.2.7 Warranty

The WR-Z16 device is fully factory tested and warranted against manufacturing defects for a period of one year. Failure of the WRZ device due to installation

problems caused by the circumstances under the WRZ device is installed cannot be warranted. This includes misuse, miswiring, overheating, operation under loads beyond the design range of the WR-Z16 device.

An example of misuse that may void your warranty is customers opening the device, including removing warranty stickers designed to indicate that the product should not be opened by customers.

For warranty or non-warranty replacement please write to our "Technical Support" below team at TimingSupport@nav-timing.safrangroup.com.

9.3 Technical Support

To request technical support for your WR-Z16 unit, please go to the "Timing Support" page of the Safran website, where you can not only submit a support request, but also find additional technical documentation.

Phone support is available during regular office hours under the telephone numbers listed below.

To speed up the diagnosis of your WR-Z16, please send us:

- the current **product configuration**, and
- the **log files**, if possible. Log on to the web interface and navigate to **Management** -> **Download device's log dump**.

Thank you for your cooperation.

9.3.1 Regional Contact

Safran operates globally and has offices in several locations around the world. Our main offices are listed below:

Country	Location	Phone
France	Les Ulis	+33 (0)1 6453 3980
Spain	Granada	+34 958 285 024
USA	West Henrietta, NY	+1.585.321.5800

Table 9-1: Safran contact information

Additional regional contact information can be found on the Contact page of the Safran Trusted 4D website.

9.4 VCS Code

The Virtual Clock Status Code has been created to easily identify the timing status of a device and easily troubleshoot in case it has failed. The VCS code are also used by the FOCA algorithm to detect a failure within a timing source and switch to the next available one.

9.4.1 General Codes

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-9000		OK	Timing source is ready	<>	<>	The device is ready to receive a signal.
VSC-9100		CRITICAL	System Error	248	Internal Oscillator	System error.
VSC-9110		WARNING	Initializing...	248	Internal Oscillator	The system is initializing.
VSC-9200		WARNING	Switchover in transient	<>	<>	A switchover process is in progress, and the system is presently in a transitional state.
VSC-9210		CRITICAL	Switchover not ready	<>	Internal Oscillator	Switchover process not ready.
VSC-9220		CRITICAL	Switchover activated due to source preemption	248	Internal Oscillator	Switchover has been triggered as a result of source preemption, initiating a transition.
VSC-9230		WARNING	Switchover started	<>	<>	Switchover started.
VSC-9300	HO	CRITICAL	Holdover activated due to source preemption	7	Holdover	Holdover has been triggered as a result of source preemption, initiating a transition.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-9400		Critical	Disable source due to source preemption	248	Internal Oscillator	Disable source due to source preemption, initiating a transition.

9.4.2 Grand Master (GM VCS Code)

A device in Grand Master is when the timing source is external (i.e., 10MHz/PPS) or from a non-PTP source such as GNSS.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-10000		OK	Locked	6	GM: Front-panel	Everything is OK for GM device.
VSC-10101		CRITICAL	Unlocked: 10MHz not present	187	GM: Internal Oscillator	The 10MHz signal is not properly connected to the GM (or provide too low voltage, bad frequency, etc...).
VSC-11101	HO	WARNING	Holdover: 10MHz lost	7	GM: Holdover	The 10MHz signal is not properly connected to the GM (or provide too low voltage, bad frequency, etc...). Holdover as GM.
VSC-10102		CRITICAL	Unlocked: PPS not present	187	GM: Internal Oscillator	The PPS signal is not properly connected to the GM (or provide too low voltage).
VSC-11102	HO	WARNING	Holdover: PPS lost	7	GM: Holdover	The PPS signal is not properly connected to the GM (or provide too low voltage).
VSC-10103		CRITICAL	Unlocked: 10MHz+PPS not present	187	GM: Internal Oscillator	10MHz and PPS signals are not properly connected to the GM (or provide too low voltage).

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-11103	HO	WARNING	Holdover: 10Mhz+PPS lost	7	GM: Holdover	10MHz and PPS signals are not properly connected to the GM (or provide too low voltage). System in Holdover as GM.
VSC-10104		CRITICAL	Unlocked: 10MHz not stable	187	GM: Internal Oscillator	Timeout in locking or DAC blocked to the limit. ~99% of the case because 10MHz are not stable or correct and thus tmgr reach a timeout count.
VSC-10501		CRITICAL	PLL delocked: 10MHz not stable.	187	GM: Internal Oscillator	PLL delock detected: 10 MHz reference unstable, GM operating with Internal Oscilator.
VSC-11501	HO	CRITICAL	PLL delocked: 10MHz not stable.	7	GM: Holdover	PLL delock detected: 10 MHz reference unstable, GM operating in Holdover mode.
VSC-10110		WARNING	Locked: PPS not present	6	GM: Front-panel	If an Atomic Clock is being used as the main reference and the PPS cable is unplugged, there may be problems on next reboot. PPS defined as not mandatory by user.
VSC-10201		WARNING	Locked: Time of Day was not set (NTP error)	6	GM: Front-panel	The same reference is being used for the entire network but the given ToD is not valid: During boot the NTP server cannot be reached (timeout or IP not configured). The ToD used is provided by release ToD or last shutdown ToD.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-10202		WARNING	Locked: Leap seconds file has expired	6	GM: Front-panel	Leapsec file in the GM device is expired or has reached expiration while the GM is running. This means that the GM can not guarantee the UTC-TAI convertion even if the currently leap seconds used is still valid. If gm/cfg/leapsec_file_ ignore option is active, this status will never occur and will be noted as VSC-10000.
VSC-10203		WARNING	Locked: ToD offset bigger than 1s (NTP offset)	6	GM: Front-panel	A drift with the current NTP offset is observed (it is likely that the NTP server has some server, but our external reference could be in free-running).
VSC-10204		WARNING	Locked: NTP does not reply anymore	6	GM: Front-panel	As NTP does not reply there might be a problem with the network. This is not critical for operation but might be a problem at next reboot. Only the GM should send alert.

9.4.3 Boundary Clock (BC VCS Code)

A device in Boundary Clock mode is receiving its timing from a PTP/WR master and redistribute to other PTP/WR slave devices.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-20001		OK	Locked (TRACK_PHASE)	6	BC: WR @ ifname	The BC clock is locked using WR and the upstream device that is providing all information is properly set.
VSC-20002		OK	Locked	6	BC: PTP @ ifname	The BC clock is locked using PTP and the upstream device that is providing all information is properly set.
VSC-20003		OK	Locked (TRACK_PHASE) - Legacy GM	6	BC: WR @ ifname	The BC clock is locked using WR but the upstream is a Legacy GM.
VSC-20004		OK	Locked - Upstream in manual Free-running	193	BC: WR @ ifname	The BC clock is locked using WR but the upstream device has been configured in free-running.
VSC-20301		CRITICAL	No connected reference - link down	248	Internal Oscillator	The BC clock has no link with upstream device.
VSC-21301	HO	CHANGEOVER	Lost connected reference - link down	187	BC: Holdover	The Link has been lost due to a link down (VSC-20301), but holdover was learnt (READY) and was quickly and automatically triggered (ACTIVATED).

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-20303		CRITICAL	No WR/PTP connected reference	248	Internal Oscillator	The BC clock has link with upstream device but does not properly receive any PTP announce message (or any other messages). This include the servo_state-e=NOT_UPDATED.
VSC-21303	HO	CHANGEOVER	No WR servo update	187	BC: Holdover	Servo was locked but not receiving any PTP packets anymore. Tmgr detect servo_state=NOT_UPDATED, and exit to HO if this was READY.
VSC-20304		CRITICAL	Invalid WR/PTP exchange	248	BC: WR/PTP @ ifname	Critical PTP error , invalid message exchange detected.
VSC-20305		CRITICAL	Can not lock to reference	248	BC: WR/PTP @ ifname	Announce PTP message is recieved, start locking with slave but can not reach the Locked state after a timeout (Wait Stable). This state is enforced by tmgr when it FSM is blocked in WAIT_LOCK until a timeout.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-20307		CRITICAL	SyncE SSM QL error - QL-DNU received	248	Internal Oscillator	Unable to connect with the GM because bad QL received.
VSC-21307	HO	CRITICAL	SyncE SSM QL error - QL-DNU received	187	BC: Holdover	Unable to connect with the GM because bad QL received, but holdover was learnt (READY) and was quickly and automatically triggered (ACTIVATED).
VSC-20320		CRITICAL	Locked: Upstream device in Free-running	248	BC: WR/PTP @ ifname	The GM is not available in the network so we are locked to an upstream device.
VSC-21320	HO	CHANGEOVER	Upstream device in Free-running	187	BC: Holdover	The GM is not available anymore in the network. Instead of staying locked to a FR upstream device we fail to our HO timing source if it was READY.
VSC-22320		CRITICAL	Upstream device in Free-Running	248	Passive WR/PTP @ ifname	The GM is not available in this network. This passive timing source can become active only if no better time source is available.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-20201		CRITICAL	Locked:Upstream GM in Free-running	187	BC: WR/PTP @ ifname	The BC clock is locked but the upstream GM has fallen in free-running for different reasons.
VSC-21201		CHANGEOVER	Upstream GM in Free-Running	187	BC: Holdover	GM announce itself to now be in FR, if we have an active HO we should exit through this state and fail to the HO timing source.
VSC-22201		CRITICAL	Upstream GM in Free-Running	187	Passive: WR/PTP @ ifname	The GM is in this network is in Free-Running. This passive timing source can become active only if no better time source is available.
VSC-20210		WARNING	Locked: Upstream GM in holdover	7	BC: WR/PTP @ ifname	The upstream GM is using its Holdover. Accuracy will decrease over time and when the HO expiration time has passed, the upper BC will be announced as FR (the timing will then be placed into a critical state).

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VCS-20501		CRITICAL	PLL delocked: L1-Sync (Sync-E) error	248	BC: WR/PTP @ ifname	For some unexpected reason the MPLL is not able to follow the received frequency from L1-Sync (Sync-E) and it has been delocked.
VCS-21501		CHANGEOVER	PLL delocked: L1-Sync (Sync-E) error	187	BC: Holdover	For some unexpected reason the MPLL is not able to follow the received frequency from L1-Sync (Sync-E) and it has been delocked. If ready, the fast delock trigger launch the Holdover.
VSC-20211		WARNING	Locked: Upstream device in holdover	187	BC: WR/PTP @ ifname	The upstream BC is using its Holdover. Accuracy will decrease over time and when the HO expiration time has passed, the upper BC will be announced as FR (the timing will then be placed into a critical state).

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-20110		WARNING	Locked: Time of Day not available on GM	6	BC: WR/PTP @ ifname	The GM announce a problem that its ToD has not been properly set since start. Probably due to an NTP error, the GM will stay there until restarting/re-evaluation of GM.
VSC-22110		WARNING	Upstream GM in holdover	7	Passive: WR/PTP @ ifname	The GM in this network is in holdover. This passive timing source can become active only if no better time source is available.
VSC-22111		WARNING	Upstream device in holdover	187	Passive: WR/PTP @ ifname	The upstream device in this network is in holdover. This passive timing source can become active only if no better time source is available.
VSC-20111		WARNING	Locked: Leap seconds file on GM has expired	6	BC: WR/PTP @ ifname	The GM time is announced has valid but the utc_offset is not valid. This means that leapsec file has expired at GM.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-20112		WARNING	Locked: GM ToD integrity compromised	6	BC: WR/PTP @ ifname	Locked, but the signal is compromised due to Grandmaster (GM) TOD (Time of day) integrity issues.

9.4.4 Others

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-90000		OK	Manual Free-running	193	Internal Oscillator	The device has been manually set as FR master and thus will distribute time according to its own reference.
VSC-90201		WARNING	Manual Free-running: Time of Day was not set (NTP error)	193	Internal Oscillator	The device is in FR but the given ToD is not valid. During boot the NTP server cannot be reached (timeout or IP not configured). The ToD used is provided by release ToD or last shutdown ToD.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-90202		WARNING	Manual Free-running: Leap seconds file has expired	193	Internal Oscillator	The device is in FR but the leap seconds file has expired or reached expiration during FR. The UTC-TAI conversion is not guaranteed even if the leap seconds currently used are still valid.
VSC-92000		OK	Idle Free-running	248	Passive	Passive state of the free-running timing source when Idle.
VSC-92400	HO	OK	Holdover Ready	248	Passive	The Holdover timing source has been learning in background from the active timing source. It is now ready to being triggered.
VSC-92401	HO	OK	Holdover Ready	248	Passive	Transitional state.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-91101	HO	CRITICAL	Holdover Expired	7>187 187>248	Internal Oscillator	The device was previously in a HO exit state and entered the holdover time source until it has finally expired. Even if we will still be connected internally to the HO clock, we announce ourself exactly like FR and we allow to reset the algorithm.
VSC-91111	HO	WARNING	Free-running/Holdover Expired	7>187 187>248	Internal Oscillator	The last mode that was "ready" has been exited in HO mode. The only thing that we perform here is increasing our clock accuracy and stay in this mode until the timer expired.

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-92411	HO	WARNING	Holdover Learning	248	Passive	The Holdover timing source is learning in background from the active timing source. If triggered, it will directly reach its expired state.

9.5 Persistent Custom Files

When an expert user needs to modify some configuration with custom settings (e.g., complex firewall rules) or wants to add new tools to the "official" firmware, he can use the custom mount directories mechanisms: This allow to store persistent files by placing them into the second ext4 partition on the SD drive mounted as `/media/data` which will be then mounted at next boot into the operating system directories:

Directory in SD drive	Mount points	Comments
/media/data/update	/media/data/update	Always created, used to update the FW
/media/data/root	/media/data/root	Root files where we can store the configuration
/media/data/usr/local/bin	/usr/local/bin	For custom binaries tools
/media/data/usr/local/sbin	/usr/local/sbin	For custom script
/media/data/usr/local/lib	/usr/local/lib	For custom libraries
/media/data/usr/local/etc	/etc/	Create symbolic links into the /etc dir

Caution: When updating with custom scripts to a new release, the expert user needs to check that its custom scripts do not interferes

with the booting procedure of the new release. In case of doubt, please contact the support team to get advices on how to proceed.

Note: These directories are mounted/linked only at the early stage of WRZ-OS initialization. A reboot might be needed to make these custom files appears at the correct place.

9.6 List of Parameters with Statistics Enabled

Currently, three services, `ppsi`, `ptp`, and `hald`, perform raw data collection and keep runtime statistics.

PPSI

The service `ppsi` keeps record of the following parameters for every WR interface, and for the active servo.

Parameter	Description
act/servo/state	State of the servo
act/servo/offset_from_master	The time error between a Slave Clock and a Master Clock
act/servo/delay_MS	Delay between Master and Slave
act/servo/delay_MM	Measured round trip time including fixed+semistatic delays
act/servo/mean_delay	Half of the cable round trip time excluding fixed+semistatic (cRTT/2).
net/wrX/1/servo/state	State of the wrX servo
net/wrX/1/servo/offset_from_master	The time error between a Slave Clock and a Master Clock
net/wrX/1/servo/delay_MS	Delay between Master and Slave

Parameter	Description
net/wrX/1/servo/delay_MM	Measured round trip time including fixed+semistatic delays
net/wrX/1/servo/mean_delay	Half of the cable round trip time excluding fixed+semistatic (cRTT/2).

PTP

The service `ptp` keeps record of the following parameters for every WR interface, and for the active servo.

Parameter	Description
net/wrX/1/info/offset_from_master	The calculated time error between a Slave Clock and a Master Clock.
net/wrX/1/info/mean_delay	Equivalent to the Round Trip Time (RTT), it is delay between Master and Slave plus the the delay between Slave to Master.

HALD

For `hald`, the list of collected parameters is the following:

Parameter	Description
spll/survey/gm_phase	GM phase in picoseconds
spll/ext/fpanel/pps_delta	Delay between this external PPS and PPS generated internally (ps)
spll/ext/fpanel/clk_cfreq	Computed frequency by counting the number of input cycles between two internal PPS (Hz)

9.7 Low Jitter Setup

Users interested in low-jitter applications of thier WR-Z16 can refer to the following list of recommendations to reduce jitter:

- Phase noise results may be impacted by environmental factors such as temperature variations, uncontrolled airflows, and mechanical vibrations.
- The user must comply with the warmup time of the external reference that is using. (For the OCXO that was used for the WR-Z16 performance data, a 24 hour warmup period was kept).
- Oscillators inside WRZ LJ devices (Zen v4 and Z16 v5) need a minimum amount of warmup time before they can reliably lock to an external reference. Typically, minimum warmup time is 10 minutes in office conditions. This could take longer in colder environments.
- Unused fiber network interfaces that have not been explicitly turned off may add a certain amount of noise. If a user is concerned with achieving the lowest possible phase noise, it is desirable to keep unused interfaces completely shut down (writing the command 'ifconfig wrX down' in the device shell for each X unused interface). This can be reverted by simply writing the command 'ifconfig wrX up'. This is not the most user-friendly approach because it requires access to the device shell, and because this configuration is lost upon reboot.
- For the same reason stated right before, a user should not mix LJ and non-LJ devices in the same setup (or, if he/she wants to do so, he/she should be aware that the devices derived from it will have higher phase noise).

The magnitude that was measured in the unit's testing setup is additive phase noise. We have not measured the noise of the external reference, but only the noise that is linearly added by the WR-Z16 as the 10 MHz time signal is consumed and regenerated. For more information on your unit's documented jitter performance, see the product data sheet.

9.8 TACACS+ and RADIUS server configuration

9.8.1 TACACS+ server installation and configuration

In order to install TACACS+ on a server with Ubuntu 18.04, it is possible to use APT to install version 4.0.4 of the package tacacs+ by using the following command:

```
apt-get install tacacs+
```

After this, it can be verified if the service is running by using the command:

```
service tacacs_plus status
```

The first step to configure the server will be opening the port 49 with TCP:

```
#ufw allow 49/tcp

Rules updated

Rules updated (v6)
```

The users are configured in the file `/etc/tacacs+/tac_plus.conf`. To do this, it is possible to modify the key by replacing it by the one we want to define:

```
key = sevensecret
```

The following simple structure can be used to define a user:

```
user = test-tacacs {

pap = cleartext password

}
```

It is possible to encrypt the password with the "tac_pwd" terminal command and enter the password to the settings as follows:

```
pap = des yD0g3Qn/0ZDsg
```

Being `yD0g3Qn/0ZDsg` the encrypted password.

There are more sophisticated configurations that add complexity, such as using groups (which serve to put common characteristics to a group of users) or acl (which serves to accept or reject clients depending on their IP address).

Note: If your WR-Z16 unit has been used as the client, the password must be configured for the root user. Registration of new users is not allowed in this device so root is the only existing user.

After finishing with the settings, it is necessary to restart the protocol by using the following command:

```
service tacacs_plus restart
```

9.8.2 RADIUS server installation and configuration

In order to install RADIUS on a server with Ubuntu 18.04, it is possible to use APT to install the v3.0.16 of the package radius by using the following command:

```
apt-get install freeradius
```

It will also be necessary to install the certificates (version 20180409):

```
apt-get install ca-certificates
```

After this, the service status can be verified by using the command:

```
service freeradius status
```

The first step to configure the server is opening the UDP ports 1812 and 1813:

```
#ufw allow 1812/udp
Rules updated
Rules updated (v6)
```

The clients must be configured in `/etc/freeradius/3.0/clients.conf` by adding their IPs with the "shared secret". For example, this can be done as follows:

```
client nashostname {
ipaddr = 172.17.5.13
secret = ourchosensecret
}
```

A subnet can be used as IP address too:

```
client mynasnetwork {
ipaddr = 172.17.5.0/24
secret = sevensecret
}
```

The configuration of the users can be done in the file `/etc/freeradius/3.0/users` by using the following lines:

```
username Cleartext-Password := "userpassword"
[other-configs]
```

An example can be:

```
test-radius Cleartext-Password := "password"
```

Note: If your WR-Z16 unit has been used as the client, the password must be configured for the root user. Registration of new users is not allowed in this device so root is the only existing user.

9.9 List of supported SFPs

Information on the supported SFPs is shown in the following table. Although our devices are compatible with other SFPs, the use of any SFP outside this list may cause synchronization errors for which we are not responsible.

Model	Wavelength (nm)	Media	Power (dBm)	Sensitivity (dBm)	Distance
AXGD-1254-0531	T1310/R1490	SMF	-9 ~ -3	-20	10km
AXGD-3454-0531	T1490/R1310	SMF	-9 ~ -3	-20	10km
AXGE-1254-0531	T1310/R1490	SMF	-9 ~ -3	-20	10km
AXGE-3454-0531	T1490/R1310	SMF	-9 ~ -3	-20	10km
1000BASE-BX BiDi SFP 1550nm-TX/1490nm-RX 120km	T1550/R1490	SMF	-1 ~ 4	<-31	120km
1000BASE-BX BiDi SFP 1490nm-TX/1550nm-RX 120km	T1490/R1550	SMF	-1 ~ 4	<-31	120km

9.10 List of Tables

9.11 List of Images

9.12 Document Revision History

Rev	Description	Date
V3.0-a	Fully updated documentation of wr-zynq-os-3.0 for WR-Z16 family	24-Jul-2020
V3.1-a	Improve PTP profiles + sync-E configuration, add new VCS code and fix some OID errors	30-Oct-2020
V3.1-b	Fixing missing references in the document	26-Jan-2021

Rev	Description	Date
V3.3-a	System reliability improved, security and authentication (including web GUI) update and new features, time of day (ToD) daemon support, bug-fixes	12-Jul-2022
V3.4	Changes to firmware update section, added a new support list. Corrected LEDs section, and added explanation of new warning message due to firmware/hardware upgrade incompatibility. Switch to Orolia branding.	14-Oct-2022
V4.0	Updated firmware update information	Feb 2023
V5.0	Documented new REST-API and updated Web GUI functionality. Updated PTP profile information. Switch to Safran branding.	17-July 2023
V5.1	New Survey Mode information, Metrics & Raw Data, updated REST-API, LLDP, and PTP4L information.	22-Feb 2024
V5.2	Added new sections for SD card recovery, PTP survey mode, updated Web UI images, and timing sources plotting.	18-Jul 2024
V5.3	Added new sections for seamless failover feature and HATI license manager. Changes to PTP profiles in PTPv2.1 section.	Mar-2025
V5.4	New information on SNMPv3 traps, PTP Enterprise profile, PTP High Accuracy profile, Seamless Failover configuration, NTP configuration, and VCS codes.	14-Jul 2025
V5.5	Added new information for SFP alerts, syslog encryption/redacting, network bonding, timing alerts, network configuration description field, and auto-negotiation.	11-Dec-2025

INDEX

V

W