WR-ZEN

SERIES

WR-ZEN TP-FL

WR-ZEN TP-FL 4PPS

WR-ZEN TP-FL 8PPS

User Manual

Revision: v5.0

Date: 27-July-2023

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.safran.com

Blank page.

CONTENTS

CHAPTER 4

CHAPTER 5

CHAPTER 8

APPENDIX

INDEX

BLANK PAGE.

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-ZEN Series. It describes the essential information about the WR- ZEN 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-ZEN.

1.2   About WR-ZEN

The White Rabbit ZEN Time Providers (referred to as WR-ZEN in this document) are high performance timing nodes family whose design is based on the Xilinx Serie-7 FPGAs that combines an Artix-FPGA with a Dual-Core ARM.

It is composed of the following products:

» WR-ZEN TP

» WR-ZEN TP-32BNC

» WR-ZEN TP-FL

They are the versatile standalone nodes that provide WR features (i.e., sub-nanosecond time synchronization in a network with thousands of nodes) controlled by a user-friendly Linux OS which runs on a dual ARM processor.

The WR-ZEN easily distributes time and frequency to other equipment by implementing standard timing protocols such as PTP, NTP and IRIG-B. The WR-ZEN combines ultra-stable clocks with low jitter and temperature compensated clock resources to enhance its synchronization accuracy. In addition, it allows integrating FMC cards.

Its robustness to failure is one of its key features thanks to redundant dual power supply and a fanless design.

It is also worth mentioning its ability to support daisy chain configurations in which each WR-ZEN device synchronizes with another WR-ZEN belonging to a higher order in the hierarchy level. At the same time, it provides synchronization to a lower level.

A typical intra-datacenter network topology is shown in the figure below, where the WR-ZEN 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.
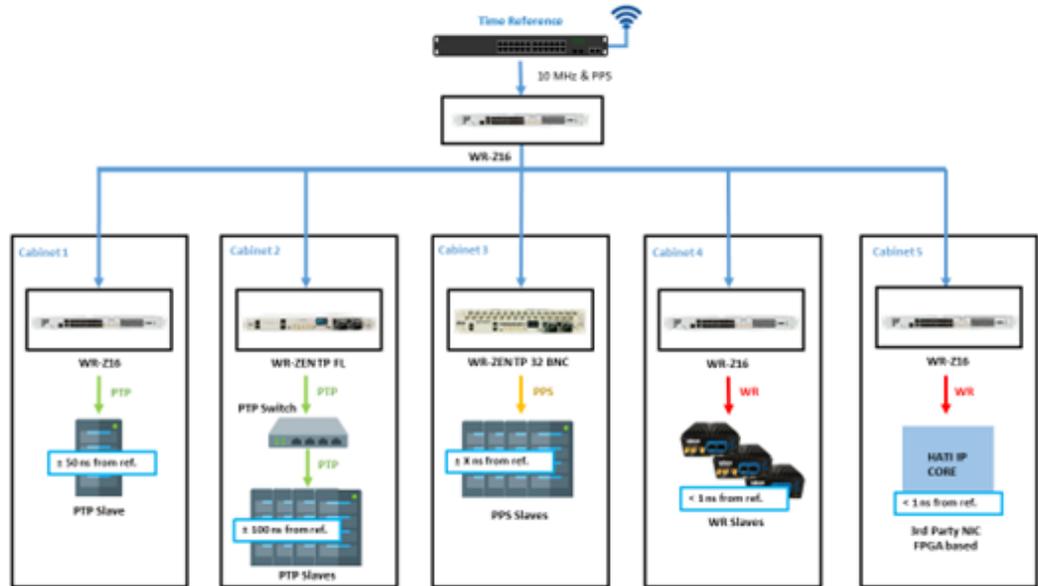
Figure 1-1: Intra-datacenter WR network topology

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

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

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-ZEN 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-ZEN is part of a full ecosystem of products which maintain sub-nanosecond 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, standard 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 standard 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, standard 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 standard 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-ZEN 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. Since the publication of the new IEEE 1588-2019 standard, White Rabbit is also known as "PTP High Accuracy" (HA) profile, 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 pico-seconds 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-ZEN Series, its capabilities, main technical features and specifications.

The following topics are included in this Chapter:

2.1 Front panel
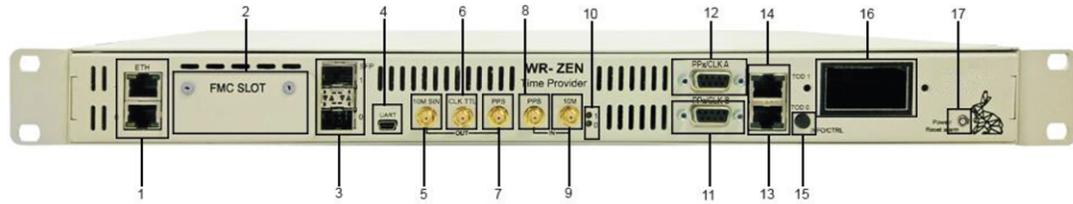
Figure 2-1: WR-ZEN TP front panel

Table 2-1: Front Panel of the WR-ZEN TP Legend

#	Name	Information	Ref.
#1	2x Management Ethernet (RJ45)	●10/100/1000 ethernet network interface (eth0 & eth1)	"WR-ZEN TP" on page 18
#2	FMC expansion port	●Slot placed to allow the use of FMC cards	"WR-ZEN TP" on page 18
#3	2x SFP Fiber Ports	●1GBps SFP compatible	
#4	1x ARM Mini-USB (B) UART	●Serial UART Mini-USB (B)	"White Rabbit" on page 75"IEEE 1588-2008 (PTPv2)" on page 80
Timing output			
#5	10 MHz output	● SMA connector (F) ● Output from PLL ● High level output: 2.5V (with 50 Ω termination)	"Virtual Clock Overview" on page 63
#6	CLK TTL output	● SMA connector (F) ● High level output: 2.5V +/- 0.2V (with 50 Ω termination) ● By default, it is configured to output 10MHz	"Virtual Clock Overview" on page 63
#7	PPS output	● SMA connector (F) ● Digital output ● High level output: 2.5V +/- 0.2V (with 50 Ω termination)	"Virtual Clock Overview" on page 63
Timing input			

#	Name	Information	Ref.
#8	PPS input	● SMA connector (F) ● 50 Ω termination ●TTL input (3.3V)	"External Reference (GM)" on page 91
#9	10 MHz input	● SMA connector (F) ● 50 Ω termination ● 1 Veff (+/- 30%) sine wave	"External Reference (GM)" on page 91
#10	LED	● Red: error LED ●Green: Status LED	"General Status" on page 16
#11	DB9 PPx/CLK A	● High level output: 1.7V +/- 0.2V (with 50 Ω termination)	
#12	DB9 PPx/CLK B	● Selectable 50 Ω termination ● High level output: 3.0V +/- 0.2V (with 50 Ω termination) ● High level output: 3.3V +/- 0.2V (without 50 Ω adaptation)	
#13	TOD0	● RJ45 RS422 ●Time of Day / NMEA	
#14	TOD1	● RJ45 RS422 ●Time of Day / NMEA	
#15	INFO/CTRL Button	●Button used to access to information a control settings	
#16	LCD Display	●Panel for alert & configuration information	"LCD screen menu" on page 141
#17	Power/Reset Alarm button ⏻	●Power On/Off the device and perform a factory reset	"Recovery Mode" on page 159
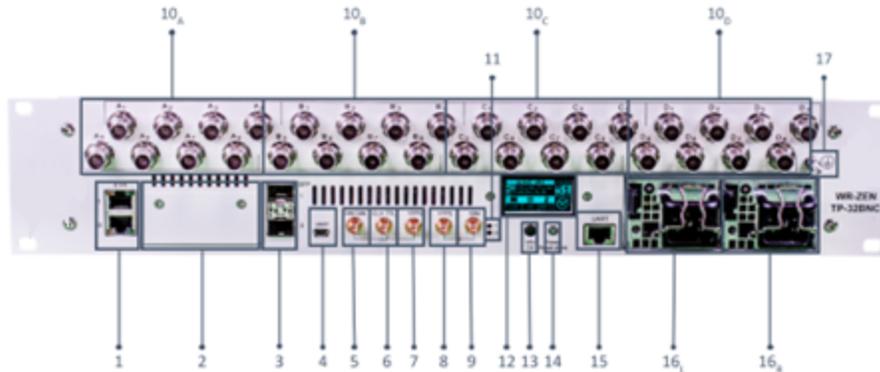
Figure 2-2: WR-ZEN TP 32-BNC front panel

Table 2-2: Front Panel of the WR-ZEN TP 32-BNC Legend

#	Name	Information	Ref.
#1	2x Management Ethernet (RJ45)	●10/100/1000 ethernet network interface (eth0 & eth1)	"WR-ZEN TP-32BNC" on page 20
#2	FMC expansion port	●Slot placed to allow the use of FMC cards	"WR-ZEN TP-32BNC" on page 20
#3	2x SFP Fiber Ports	●1GBps SFP compatible	"White Rabbit" on page 75"IEEE 1588-2008 (PTPv2)" on page 80
#4	1x ARM Mini-USB (B) UART	●Serial UART Mini-USB (B)	
Timing output			
#5	10 MHz output	● SMA connector (F) ● Output from PLL ● High level output: 2.5V (with 50 Ω termination)	"Virtual Clock Overview" on page 63
#6	CLK TTL output	● SMA connector (F) ● High level output: 2.5V +/- 0.2V (with 50 Ω termination) ● By default, it is configured to output 10MHz	"Virtual Clock Overview" on page 63

#	Name	Information	Ref.
#7	PPS output	● SMA connector (F) ● Digital output ● High level output: 2.5V +/- 0.2V (with 50 Ω termination)	"Virtual Clock Overview" on page 63
Timing input			
#8	PPS input	● SMA connector (F) ● 50 Ω termination ●TTL input (3.3V)	"External Reference (GM)" on page 91
#9	10 MHz input	● SMA connector (F) ● 50 Ω termination ● 1 Veff (+/- 30%) sine wave	"External Reference (GM)" on page 91
#10 A	8x Configurable PPS/10 MHz	● BNC connector (F) ● Digital output ● High level output: 1.7V +/- 0.2V (with 50Ω termination)	
#10 B	8x Configurable PPS/10 MHz	● BNC connector (F) ● Digital output ● High level output: 1.7V +/- 0.2V (with 50Ω termination)	
#10 C	8x Configurable PPS/10 MHz	● BNC connector (F) ● Digital output ●High level output (Selectable 50Ω termination): ●3V +/-0.2V (with 50Ω termination) ● 3.3V +/- 0.2V (without 50Ω adaptation)	

#	Name	Information	Ref.
#10 D	8x Con-figurable PPS/10 MHz	● BNC connector (F) ● Digital output ●High level output (Select-able 50Ω termination): ●3V +/-0.2V (with 50Ω ter-mination) ● 3.3V +/- 0.2V (without 50Ω adaptation)	
#11	LED	● Red: error LED ●Green: Status LED	"General Status" on page 16
#12	LCD Display	● Panel for alert & con-figuration information	"LCD screen menu" on page 141
#13	INFO/CTRL Button	●Button used to access to information a control set-tings	
#14	Power/Reset Alarm button ⏻	●Power On/Off the device and perform a factory reset	"Recovery Mode" on page 159
#15	1x Man-agement UART (RJ45)	Serial UART RS232 on a RJ45 connector with pinout (USB-RJ45/RS232 adaptor not included)	
		Pin #1: ✗ Pin #2: ✗ Pin #3 RXD Pin #4 GND Pin #5: ✗ Pin #6: TXD Pin #7: ✗ Pin #8: ✗	
Power Supply			
#16 L	Power Supply #1	●Swappable & monitorable module: - 100- 240VAC, 50- 60Hz, (80W Max)	
#16 R	Power Supply #2	● Swappable & mon-itorable module: - 100- 240VAC, 50- 60Hz, (80W Max)	

#	Name	Information	Ref.
#17	Ground	●Ground connector of the device	
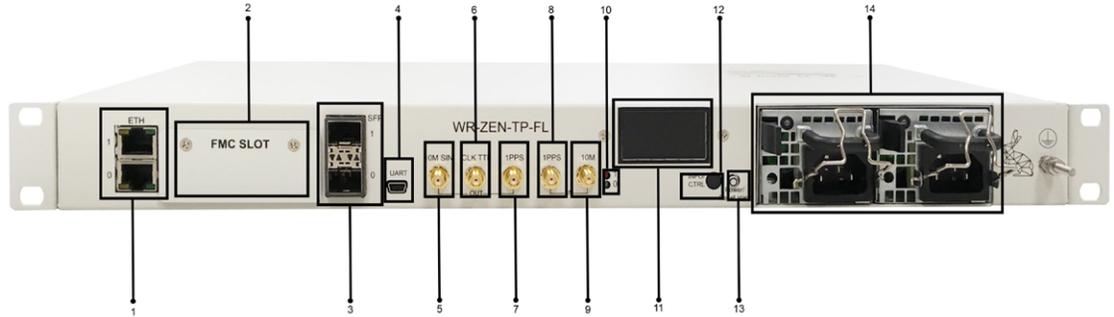
Figure 2-3: WR-ZEN TP-FL front panel

Table 2-3: Front Panel of the WR-ZEN TP-FL Legend

#	Name	Information	Ref.
#1	2x Management Ethernet (RJ45)	●10/100/1000 ethernet network interface (eth0 & eth1)	"WR-ZEN TP-FL" on page 19
#2	FMC expansion port	●Slot placed to allow the use of FMC cards	"WR-ZEN TP-FL" on page 19
#3	2x SFP Fiber Ports	●1GBps SFP compatible	"White Rabbit" on page 75"IEEE 1588-2008 (PTPv2)" on page 80
#4	1x ARM Mini-USB (B) UART	●Serial UART Mini-USB (B)	
Timing output			
#5	10 MHz output	● SMA connector (F) ● Output from PLL ● High level output: 2.5V (with 50 Ω termination)	"Virtual Clock Overview" on page 63
#6	CLK TTL output	● SMA connector (F) ● High level output: 2.5V +/- 0.2V (with 50 Ω termination) ● By default, it is configured to output 10MHz	"Virtual Clock Overview" on page 63

#	Name	Information	Ref.
#7	PPS output	● SMA connector (F) ● Digital output ● High level output: 2.5V +/- 0.2V (with 50 Ω termination)	"Virtual Clock Overview" on page 63
Timing input			
#8	PPS input	● SMA connector (F) ● 50 Ω termination ●TTL input (3.3V)	"External Reference (GM)" on page 91
#9	10 MHz input	● SMA connector (F) ● 50 Ω termination ● 1 Veff (+/- 30%) sine wave	"External Reference (GM)" on page 91
#10	LED	● Red: error LED ●Green: Status LED	"General Status" on page 16
#11	LED	● Red: error LED ●Green: Status LED	"General Status" on page 16
#12	LCD Display	● Panel for alert & configuration information	"LCD screen menu" on page 141
#13	Power/Reset Alarm button ⏻	●Power On/Off the device and perform a factory reset	"Recovery Mode" on page 159
#14	1x Management UART (RJ45)	Serial UART RS232 on a RJ45 connector with pinout (USB-RJ45/RS232 adaptor not included) Pin #1: ✗ Pin #2: ✗ Pin #3 RXD Pin #4 GND Pin #5: ✗ Pin #6: TXD Pin #7: ✗ Pin #8: ✗	
Power Supply			
#15 L	Power Supply #1	●Swappable & monitorable module: - 100- 240VAC, 50- 60Hz, (80W Max)	

#	Name	Information	Ref.
#15 R	Power Supply #2	● Swappable & monitorable module: - 100- 240VAC, 50- 60Hz, (80W Max) ⚠️	
#16	Ground	●Ground connector of the device	

2.2 Rear panel

Figure 2-4: WR-ZEN TP rear panel

Table 2-4: Back Panel of the WR-ZEN TP Legend

#	Name	Information	Ref.
Power Supply			
#1	Power Supply #1	● Swappable & monitorable module: -100-240VAC, 50-60Hz, (80W Max) ⚠️	
#2	Power Supply #2	● Swappable & monitorable module: -100-240VAC, 50-60Hz, (80W Max) ⚠️	
Plug-and-Play Modules			
#3	Plug-and-play module slot #J1	For swappable modules	
#4	Plug-and-play module slot #J2	For swappable modules	
#5	Plug-and-play module slot #J3	For swappable modules - By default, we provide the WR-ZEN-TP with one FAN on slot #J3	

#	Name	Information	Ref.
#6	Plug-and-play module slot #J4	For swappable modules	
#7	Ground	●Ground connector of the device	

2.3 Monitoring LEDs

The status of the WR-ZEN 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-ZEN.

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 General 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 2-5: Status LED behavior

Visual	Behavior	Description
1 ●●● 0 ●●●	Steady Green	Device system state is OK
1 ●●● 0 ●●●	Steady Red	Device system state is CRITICAL: user might login to verify the source of error. The device might reboot itself if the error is persistent.
1 ●●● 0 ●●●	Steady Red & Green	Device system state is WARNING
During booting procedure		
1 ●●● 0 ●●●	1x Blinking Green	Bootloader initialization OK
1 ● ●● 0 ●● ●	[1-15]x Blinking Red & Green	Reset button is held during the booting procedure: - If released: entering recovery mode - If hold > 15s: entering reset factory mode
1 ●●● 0 ●●●	Steady Red	Device is booting in recovery mode

Visual	Behavior	Description
1 ●●● 0 ●●●	Steady Red & Green	Device is booting in reset factory mode
1 ●●● 0 ●●●	2x Blinking Green	Booting in normal mode
1 ●●● 0 ●●●	Idle	Device is loading kernel and transitioning between modes
1 ●●● 0 ●●●	3x Blinking Green	FPGA initialization OK
1 ●●● 0 ●●●	Steady Red & Green	The hald daemon has been loaded and it is waiting till all daemons are properly loaded
1 ●●● 0 ●●●	Nx Blinking Red & Green	Device is loading in failsafe mode and for each module skipped during initialization the device blinks in orange

2.3.2 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	▲	Link down	When link is down led B is disabled
		▲	Link up	When link is up the led B stays in green
	D	▲	Activity	Blinks in orange each time a packet is received on this port
	1		**wr1**	**wr1 corresponds to lower SFP in the stack**
	A	▼	Activity	Blinks in orange each time a packet is received on this port
	C	▼	Link down	When link is down led C is disabled
		▼	Link up	When link is up the led C stays in green

2.4   Product Specifications

2.4.1   WR-ZEN TP

System On-Chip

» **SoC**: Xilinx Zynq series

» **CPU**: Dual ARM® CortexTM-A9 MP@ 866 MHz

» **Memory**:

   » 512 MB DDR3 (32-bit bus)

   » 256 Mbits Quad-SPI Flash

   » 16GB SD Card

Physical Dimension

» **Dimension**: 431 mm x 44 mm x 300 mm / (1 Rack Unit)

» **Color**: White (Metallic)

Environmental Conditions

» **Temperature**: -10ºC ~ +50ºC

» **Humidity**: 0% ~ 90% RH

Front Panel

» **Ethernet**: 2x 100/1000 Base-T RJ45 (Management, PTP, NTP)

» **FMC**: 1x Low-Pin count FMC slot (LPC)

   » Using HPC-FMC footprint to support extra GTP channel/clock on board

» **SFP Ports**: 2x 1GbE

» **Clocks I/O**: 5x SMA connectors (3.3V @50Ω):

   » 10MHz SIN OUT ref.clock (PLL)

   » CLK TTL OUT

   » PPS OUT (LVTTL)

   » PPS IN (LVTTL)

   » 10MHz IN (TTL/CMOS/ECL/clipped sine)

» **Timing**: Common outputs to provide Timing

   » 2xRJ45, ToD (NMEA) RS485

   » 2xDB9, 4x configurable* xPPS/CLK

» *xPPS/CLK outputs are divided in two groups of 4 customized PPS/CLK (1, 10, 1k-10M)Hz similar to the SMA Aux Clk.

» **LCD**: Information panel for alert & network

» **Debug**: USB Mini-B (ARM-UART)

Back-panel

» **Fan** 2 x (1-4) Swappable fan modules

2.4.2 WR-ZEN TP-FL

System On-Chip

» **SoC**: Xilinx Zynq series

» **CPU**: Dual ARM® CortexTM-A9 MP@ 866 MHz

» **Memory**:

» 512 MB DDR3 (32-bit bus)

» 256 Mbits Quad-SPI Flash

» 16GB SD Card

Physical Dimension

» **Dimension**: 444 mm x 43 mm x 221 mm / (1 Rack Unit)

» **Color**: White (Metallic)

Environmental Conditions

» **Temperature**: -10ºC ~ +50ºC

» **Humidity**: 0% ~ 90% RH

Front Panel

» **Ethernet**: 2x 100/1000 Base-T RJ45 (Management, PTP, NTP)

» **SFP Ports**: 2x 1GbE

» **Clocks I/O**: 5x SMA connectors (3.3V @50Ω):

» 10MHz SIN OUT ref.clock (PLL)

» CLK OUT (LVTTL)

» PPS OUT (LVTTL)

» PPS IN (LVTTL)

» 10MHz IN (TTL/CMOS/ECL/clipped sine)

» **LCD**: Information panel for alert & network

» **Expansion**: 4x SMA coaxial connectors (3.3V/50Ω):

  » PPS OUT (LVTTL)

2.4.3   WR-ZEN TP-32BNC

System On-Chip

» **SoC**: Xilinx Zynq series

» **CPU**: Dual ARM® A9 MP@ 866 MHz

» **Memory**:

  » 512 MB DDR3 (32-bit bus)

  » 256 Mbits Quad-SPI Flash

  » 16GB SD Card

Physical Dimension

» **Dimension**: 428 mm x 88 mm x 220 mm / (2 Rack Units)

» **Color**: White (Metallic)

Environmental Conditions

» **Temperature**: -10ºC ~ +50ºC

» **Humidity**: 0% ~ 90% RH

Front Panel

» **Ethernet**: 2x 100/1000 Base-T RJ45 (Management, PTP, NTP)

» **SFP Ports**: 2x 1GbE

» **Clocks I/O**: 5x SMA connectors (3.3V @50Ω):

  » 10MHz SIN OUT ref.clock (PLL)

  » CLK OUT (LVTTL)

  » PPS OUT/IRIG-B (LVTTL)

  » PPS IN (LVTTL)

  » 10MHz IN (TTL/CMOS/ECL/clipped sine)

» **Fanout**: 32 BNC configurable outputs divided in 2 groups (1.7V/50Ω):

  » A&B: 10MHz (xPPS under license) with 50Ω termination

  » C&D: xPPS, IRIG-B with selectable 50Ω termination

» **LCD**: Information panel for alert & network

» **Debug**: USB Mini-B (ARM-UART)

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 screwdriver 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 103.

3.2 Connecting to the Device

There are two ways to connect to the WR-ZEN 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-ZEN 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-ZEN 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 26).

>> Follow the steps described in network configuration from CLI (see "Network configuration from CLI" on page 53).

>> 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-ZEN as shown in the "Front panel" on page 8Hardware 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 tt*y 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 27.

3.2.5   Connecting on Windows

3.2.5.1   Logging in from UART

The connection to the UART in the WR-ZEN Series 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.
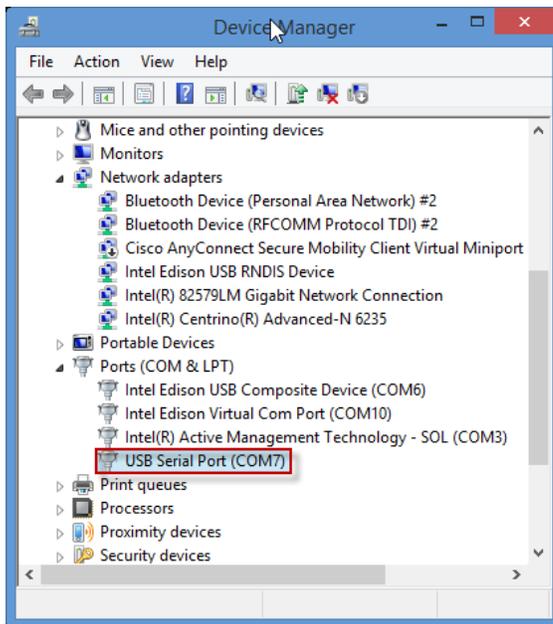
Figure 3-1: Device manager. New serial port detected.

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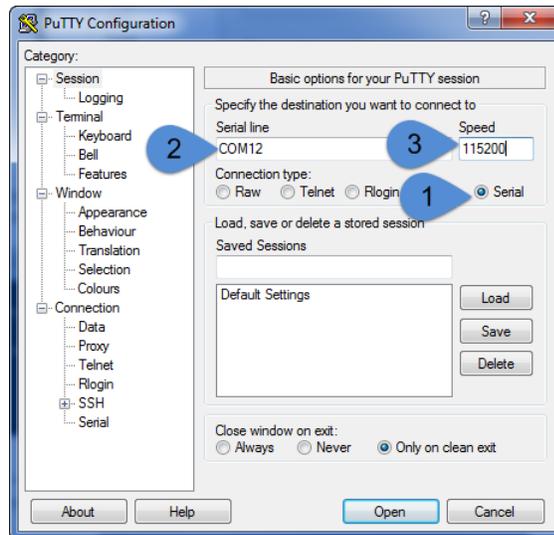
Figure 3-2: Putty configuration for serial port connection.

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-ZEN, the user must login by clicking on Login Page, and provide the corresponding password for the root user (see "Default Configuration" on page 26).
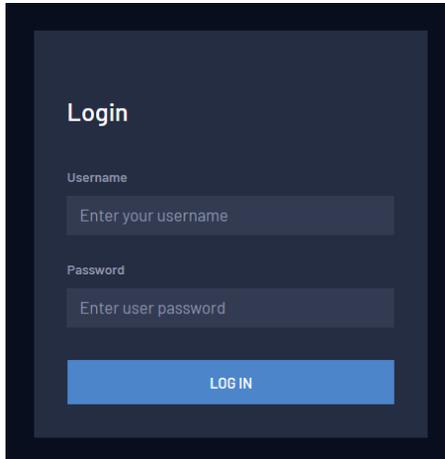
Figure 3-3: Login page of the web interface.

The main dashboard view will then be displayed.
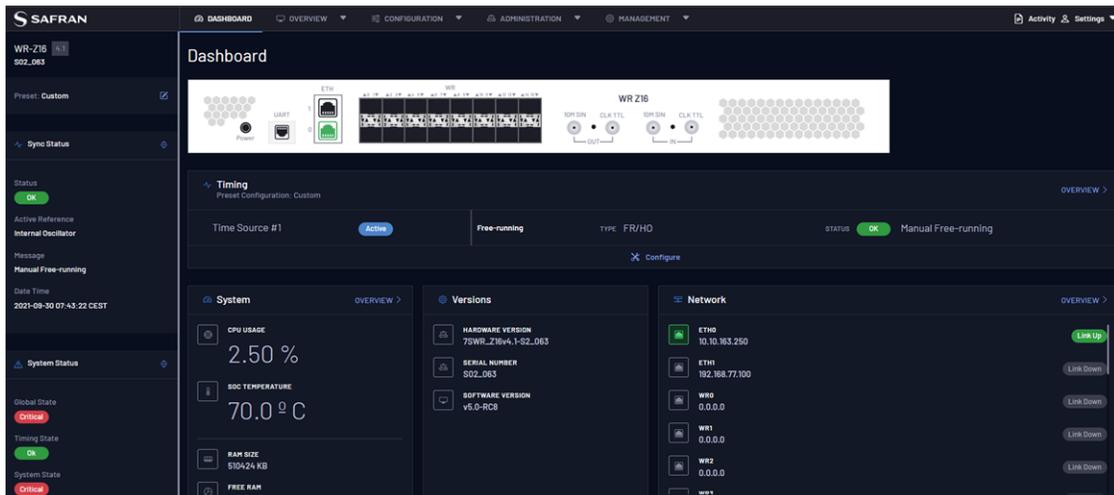
Figure 3-4: Dashboard page of the web interface

Further instructions on the available features and how to use them in the WR-ZEN's web interface are detailed in the "GUI & CGI Tools" on page 33 section.

CHAPTER 4

GUI & CGI 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]).

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>:8201/`, where `<ip>` is a valid IP address that reach the device (e.g. `http://192.168.77.100:8201/`). Once the user is logged in, a web page in shown where all the methods and its corresponding examples are described.
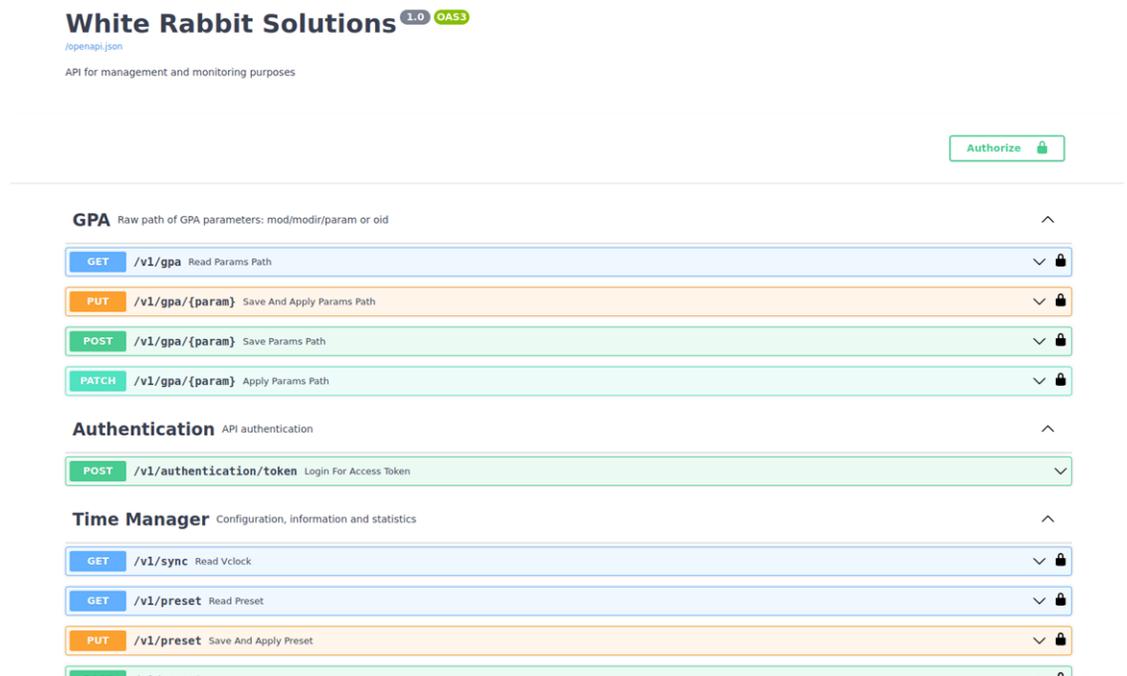
Figure 4-1: Main page of the REST-API documentation.

4.2.2 Interacting with the REST-API

The REST-API is accessible via `http://<ip>:8201/`. 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.

> **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 clicking the "Authorize :lock: " button on the top of the page and inserting the credentials:

   a. Open the Authorize dialog for obtaining a token:
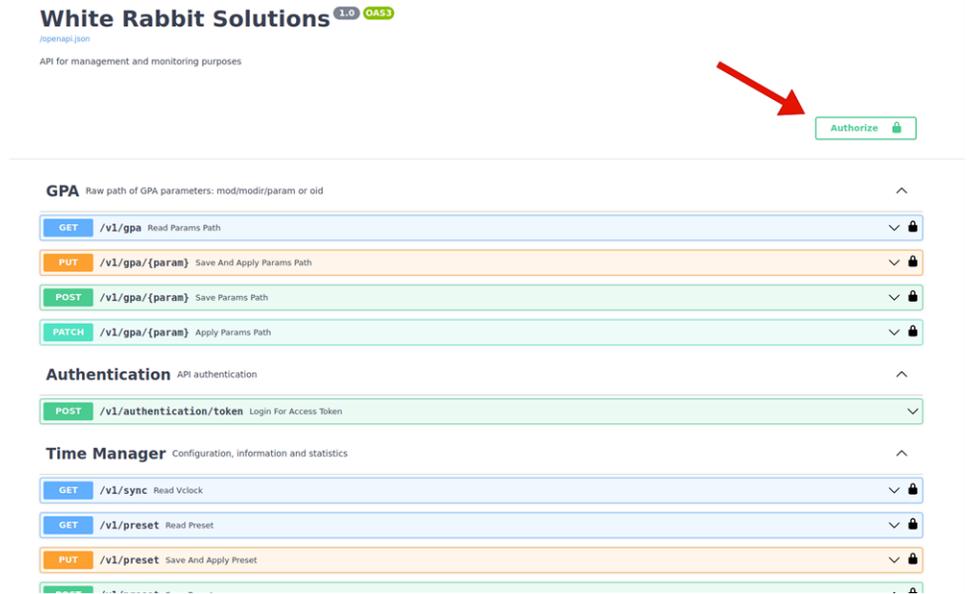
   b. Enter the credentials in the Authorize form.
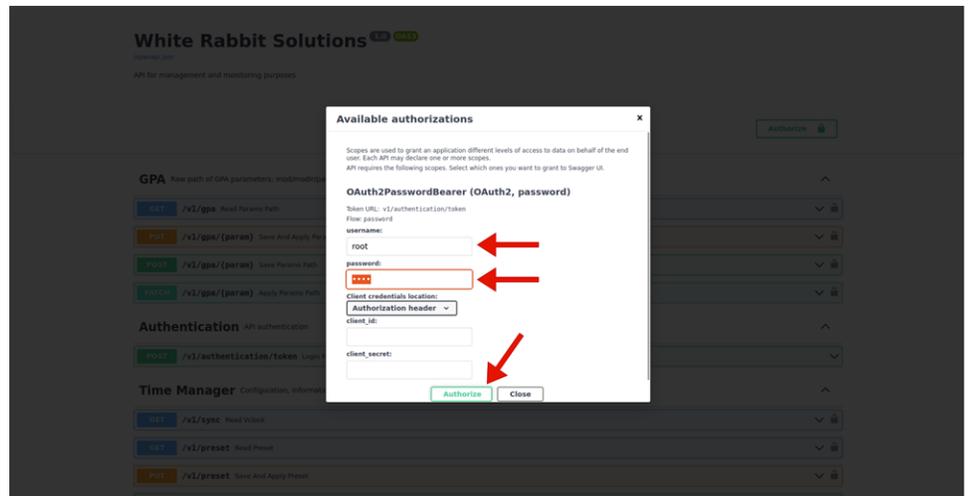
c. Once the token was correctly obtained, proceed to close the dialog and tests the REST-API.
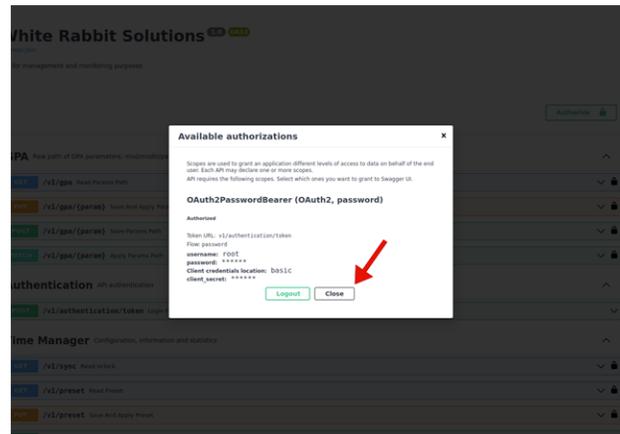
3. Choose a parameter and a method. If using the "GET" method, choose also a "format". For example:

a. Open the `GET /v1/preset` method and click in "Try it out".
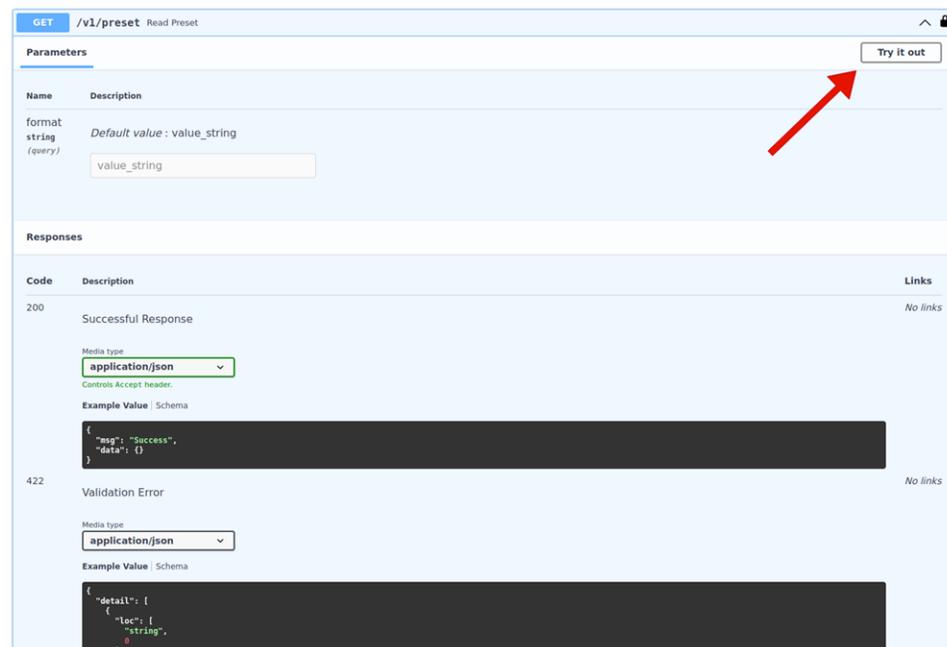
b. Choose a format string (for example: complete) and click in "Execute".

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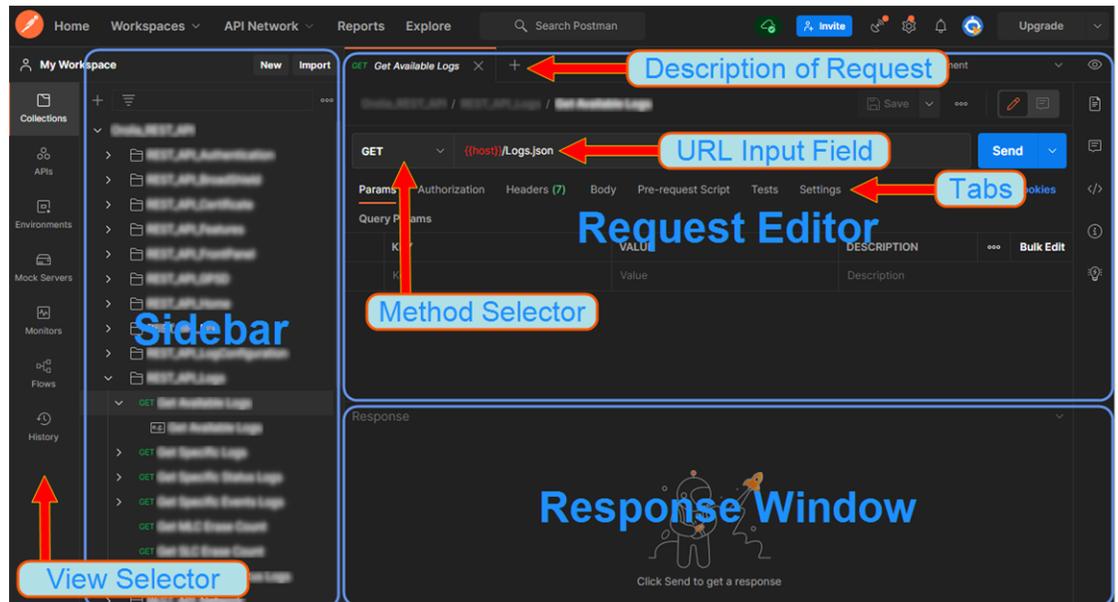
The **Sidebar** lists the requests stored in the loaded Collections tab, the parameters stored in the Environments tab, and – under the History tab – a list of recently submitted requests.

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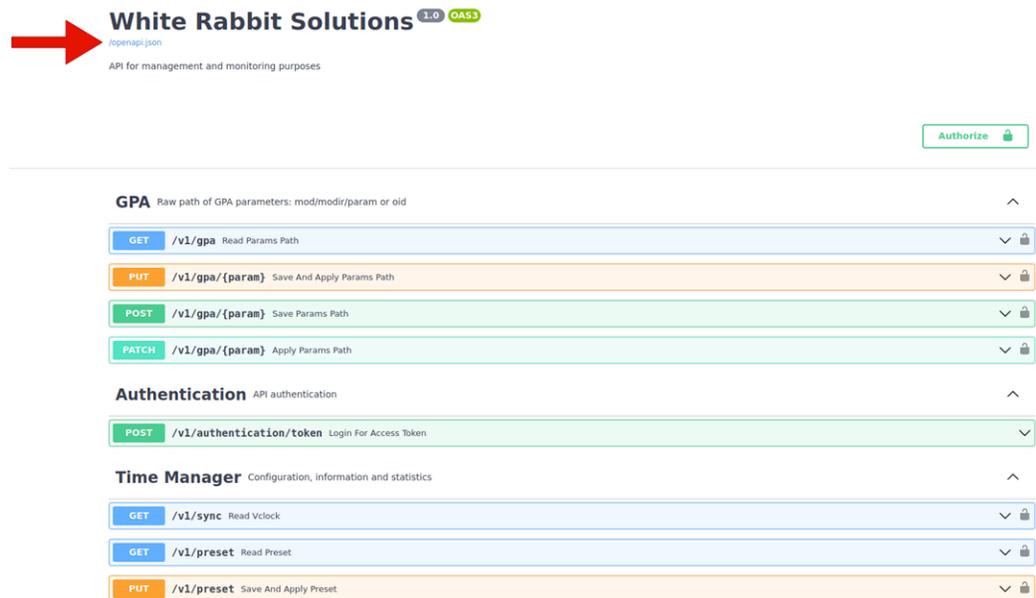
2. Open the Postman app, using the credentials of your previously created account.

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/`)
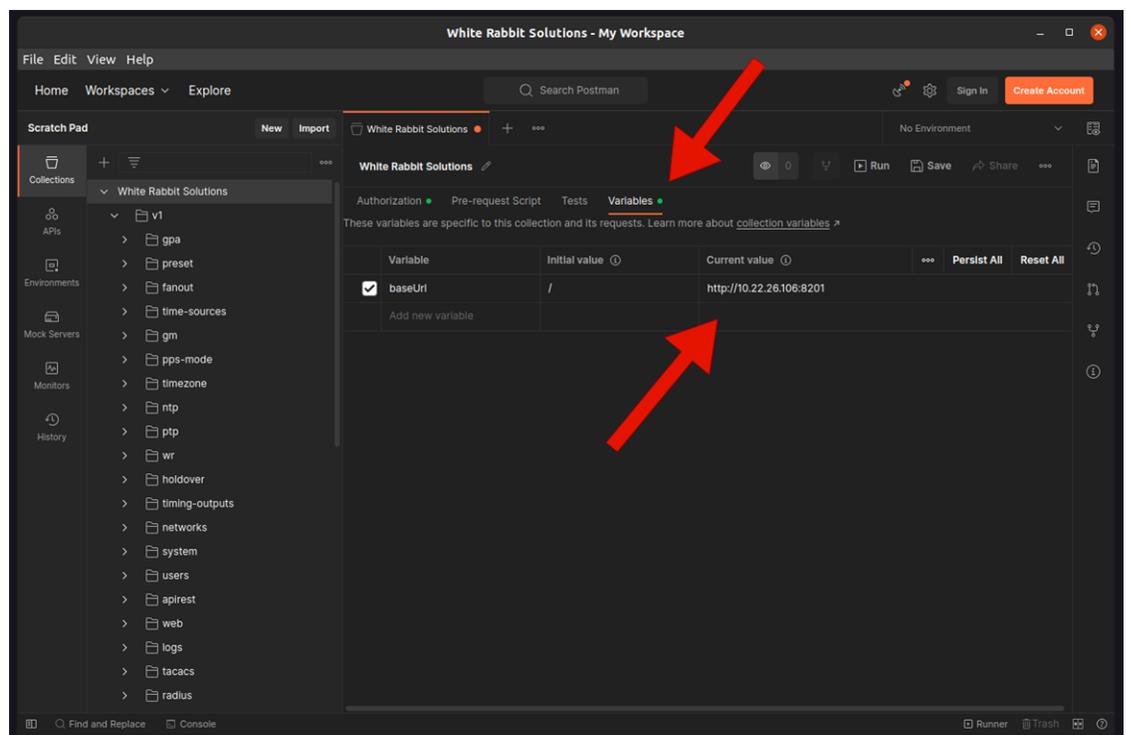
4.2.4.6 Obtaining the Access Token

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.
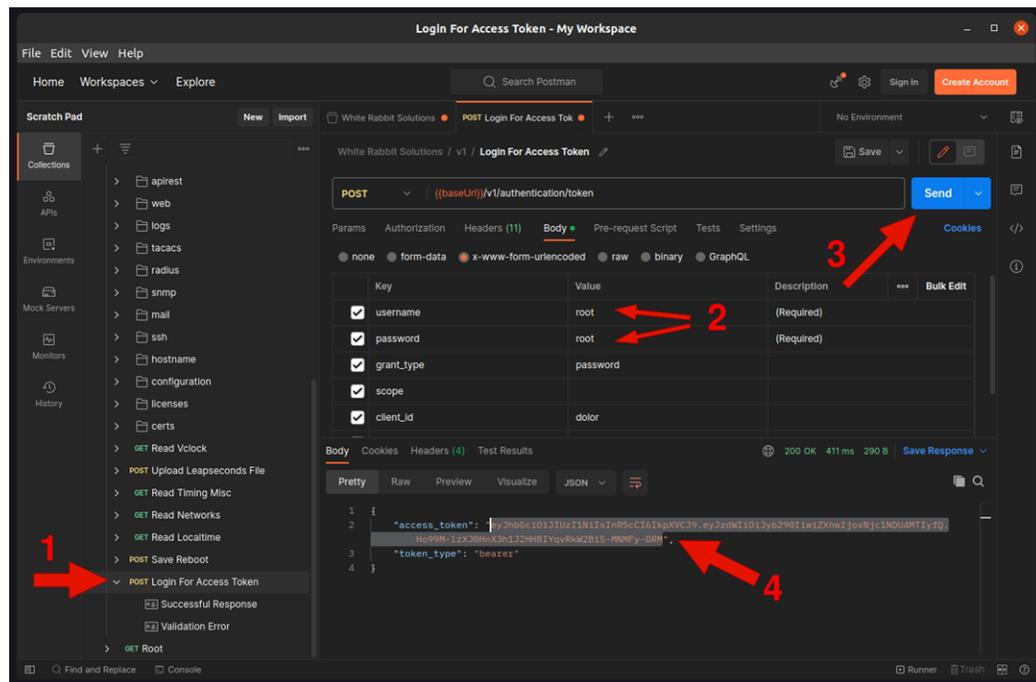
4.2.4.7 Making a Request

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 Pre-set).

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.
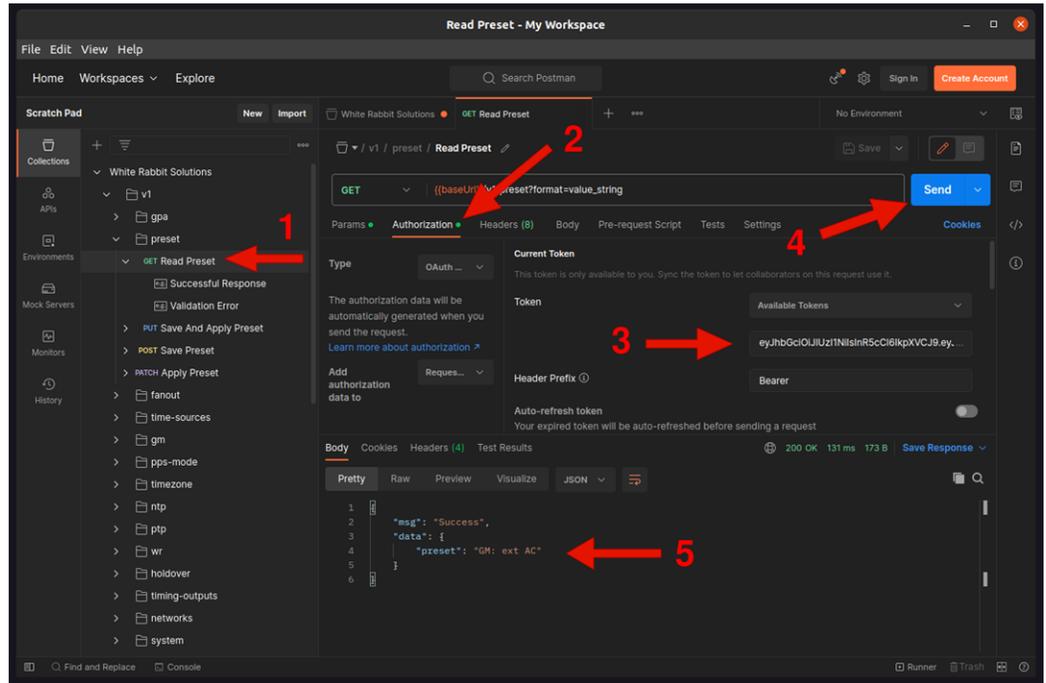
4.3 The Web GUI

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.
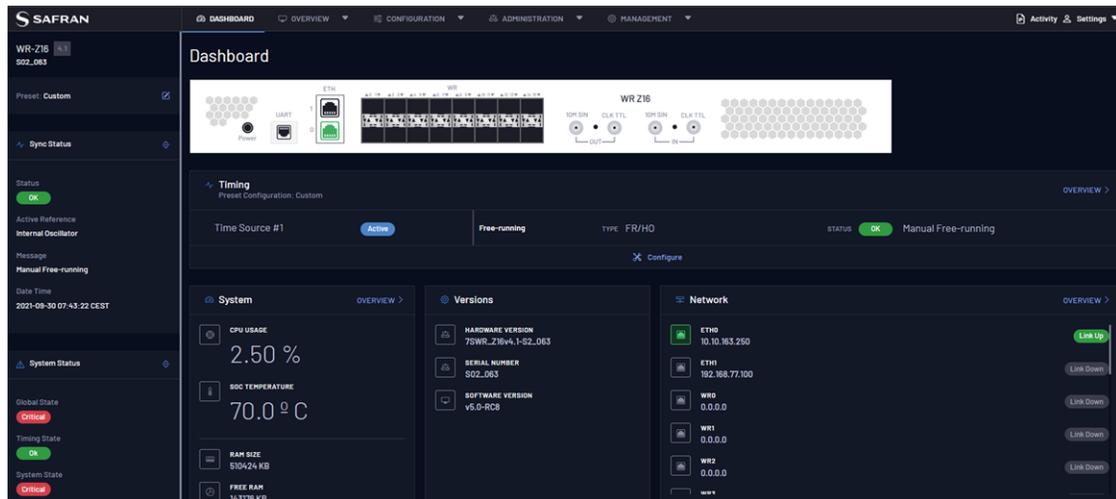
Figure 4-2: Dashboard in web interface

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

    » Healthing information: System, Power Supplies, Fan

  In the Overview pages, the breadcrumb bar will contain a link to the settings page, if applicable:
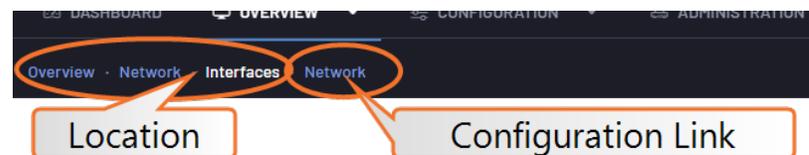
  » The **Configuration Menu** has links to the configuration of General Timing, White Rabbit, PTP, External Reference (GM), Holdover, NTP, and Misc.

  » 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 **Product Image** provides real-time depictions of the status of any con-nected 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 num-ber, and software version.

» The **Network Status Panel** details the IP address information for each port.
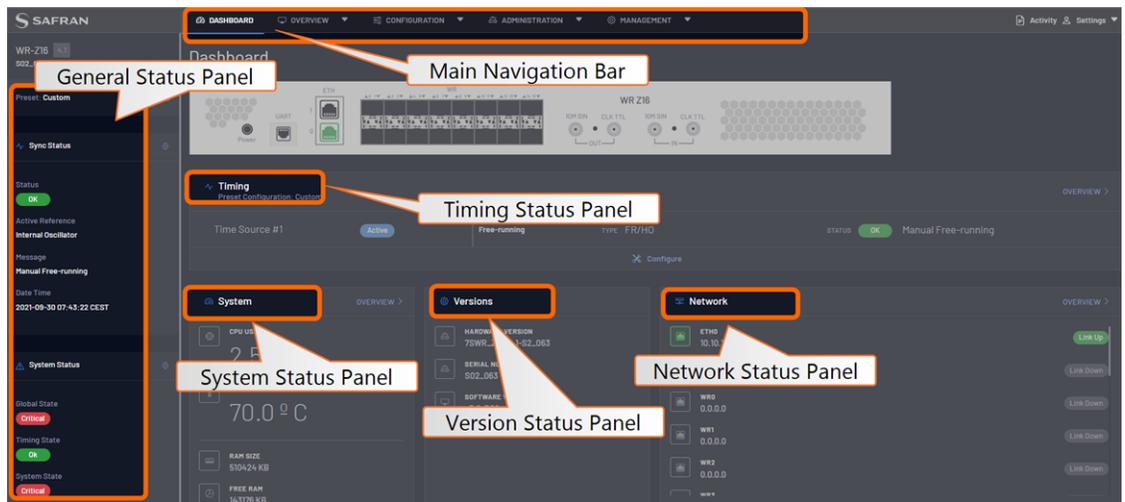
Figure 4-3: Web GUI Dashboard sections

The web session will automatically terminate after 15 minutes of inactivity. This setting can be disabled or configured in **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:
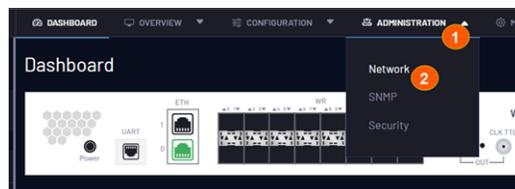
Figure 4-4: Web GUI Network Page Navigation

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.
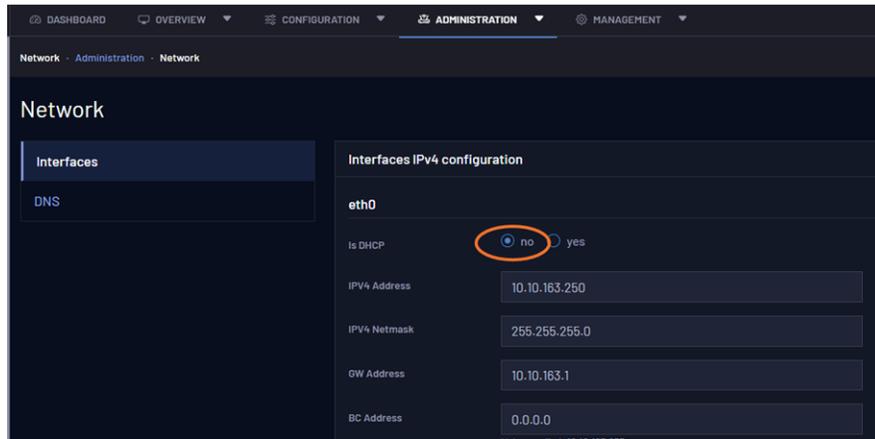
Figure 4-5: Disable DHCP via the Web GUI

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 save; you will also see a warning that the saved changes will not be applied to the functionality until the unit is rebooted:
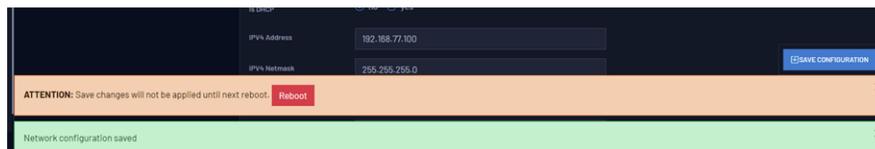
Figure 4-6: Network change banners at the bottom of the page

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.).

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 53).

» Healthing: Power Supplies & Fans related configuration ("Healthing " on page 135).

» Security: Configuration related to the security of the device ("Security & Authentication" on page 103).

» Management: Logging & Monitoring ("Monitoring & Logging" on page 115) configuration and aspects related to the maintenance of the device ("Device Maintenance" on page 149).
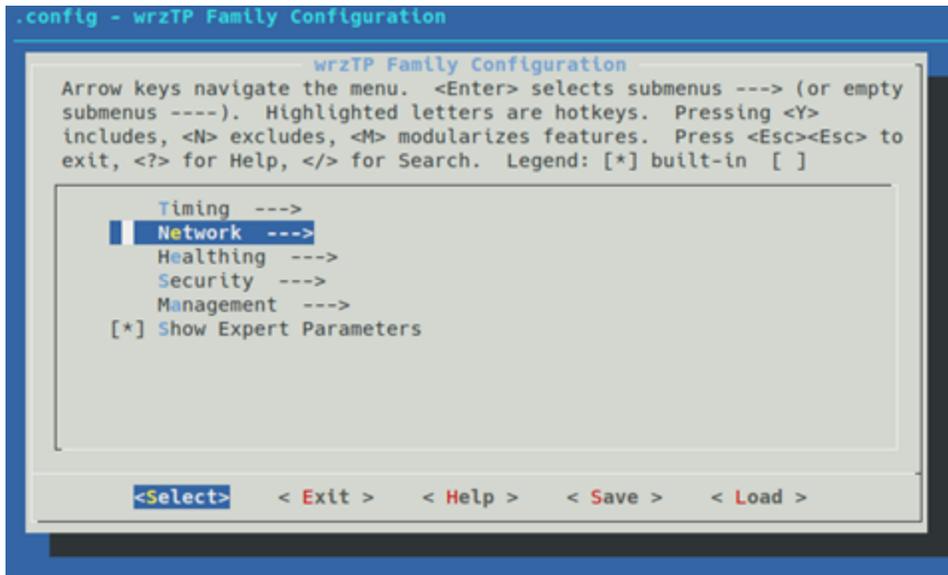
Figure 4-7: Main wrz_config interface. Modules to modify

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:
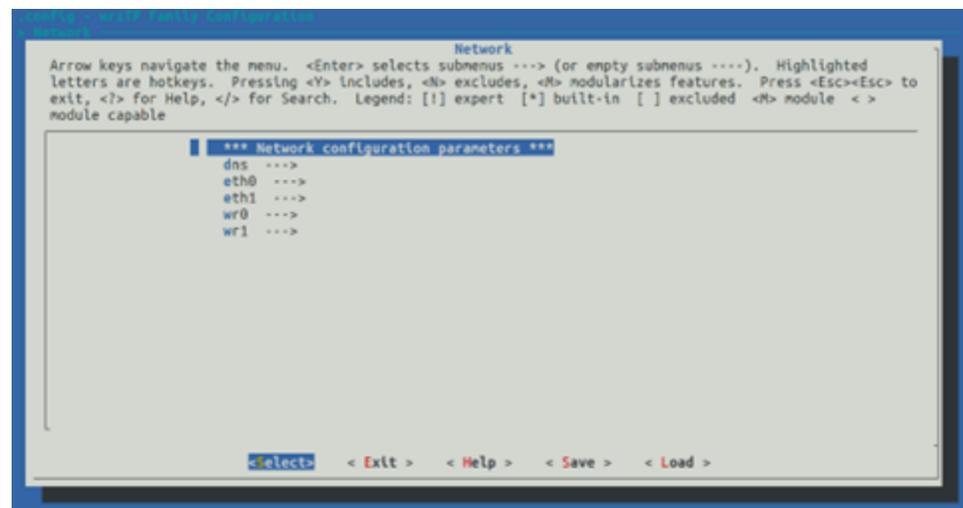
3. Then the corresponding network interface (eth0) to be modified must be selected:.

Figure 4-8: wrz_config interface. Network interfaces to change

4. If the static IPv4 settings must be loaded, disable DHCP
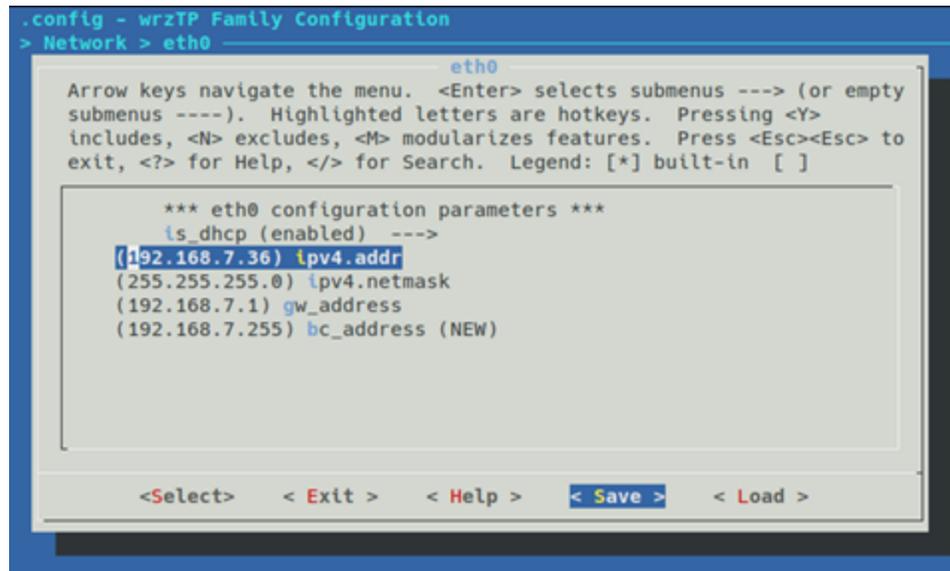
Figure 4-9: wrz_config interface. Interface parameters to change

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.
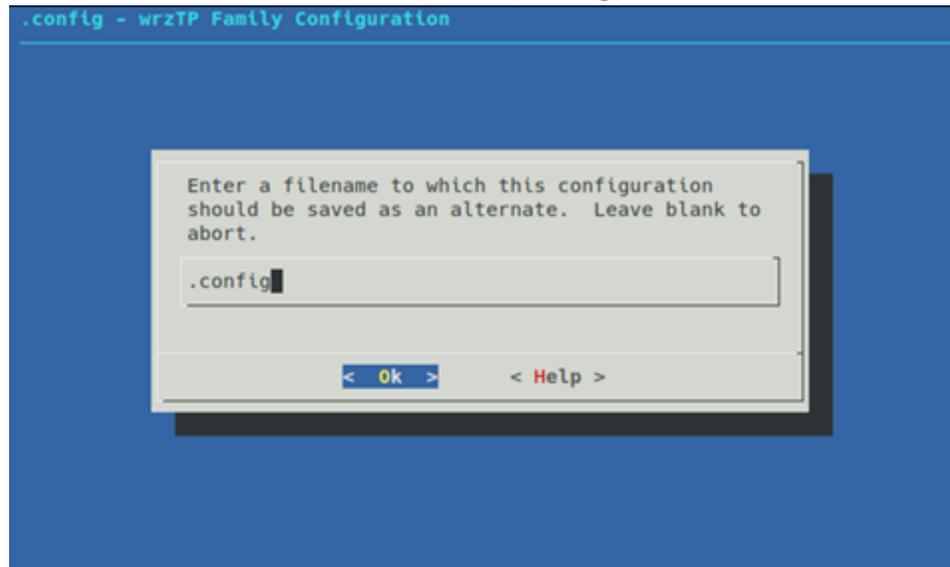
Figure 4-10: wrz_config interface. File in which to save the new applied configuration

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.

4.5 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@sup-tpf1-461:~# gpa_ctrl hald pws
---          hald --------------------------------------------------------|
C  0.9110.1   pws/pws1/status              : NOT DETECTED
   0.9110.2   pws/pws1/temperature         : 0                      C
   0.9110.3   pws/pws1/v_in                : 0.00                   V
   0.9110.4   pws/pws1/v_out               : 0.00                   V
   0.9110.5   pws/pws1/power_in            : 0.00                   W
   0.9110.6   pws/pws1/power_out           : 0.00                   W
   0.9110.7   pws/pws1/disable_alert       : No
   0.9120.1   pws/pwsr/status              : OK
   0.9120.2   pws/pwsr/temperature         : 38                     C
   0.9120.3   pws/pwsr/v_in                : 238.50                 V
   0.9120.4   pws/pwsr/v_out               : -5757.00               V
   0.9120.5   pws/pwsr/power_in            : 22.00                  W
   0.9120.6   pws/pwsr/power_out           : 16.00                  W
   0.9120.7   pws/pwsr/disable_alert       : No
```

Figure 4-11: 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 (0.9110.1) 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@zen- 425:~#   gpa_ ctrl   hald   pws/pwsl/status;   echo   $?
OK
0
root@zen- 425:~#   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@zen-425:~# 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)
```

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

```
root@zen-425:~#                 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@zen-425:~#        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 157).

» `wrz_logdump`: Tool used to report an error log for the support team (See " How to report an error" on page 170).

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 main purpose of the WR-ZEN is to distribute ultra-accurate synchronization through a timing network.

The device can be configured as a Grand-Master ("External Reference (GM)" on page 91) to retrieve time from an external source and then redistribute it using NTP, IEE1588-2008/PTPv2 or White Rabbit protocols.

Through its virtual clock ("Virtual Clock Overview" on page 63) centralization system, it allows to combine these protocols in multiple ways and enable resiliency by configuring multiple timing sources. Additionally, an optional Holdover ("Holdover" on page 96) oscillator can be included to maintain high accuracy (1.5us < 24h after learning 3 days) even if all timing references are down.

Finally, the WR-ZEN family can provide timing to third party devices using ultra precise 10MHz/PPS output signals or through NTP ("NTP" on page 93) if accuracy is not a strong necessity.

The following topics are included in this Chapter:

5.1 Multi-sources & Resiliency

To ensure continued operation over possible failures, the WR-ZEN 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) → Coming soon!

» 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. However, a combination of PTP+SyncE as timing source, allows to re-distribute timing using WR without significant penalties.

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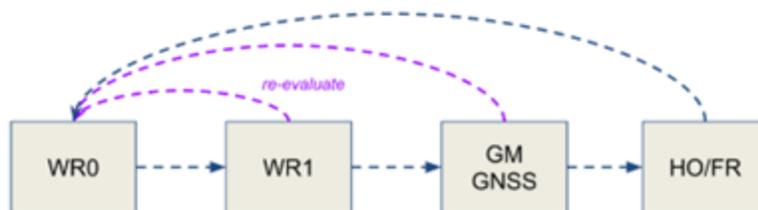
Figure 5-1: Multi-timing sources handle by FOCA policy with its two strategies: only fall-down (blue) & re-evaluation (purple)

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 173).

For a deeper understanding of the behavior of the FOCA algorithm it is recommend reading the section "Others" on page 180 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.
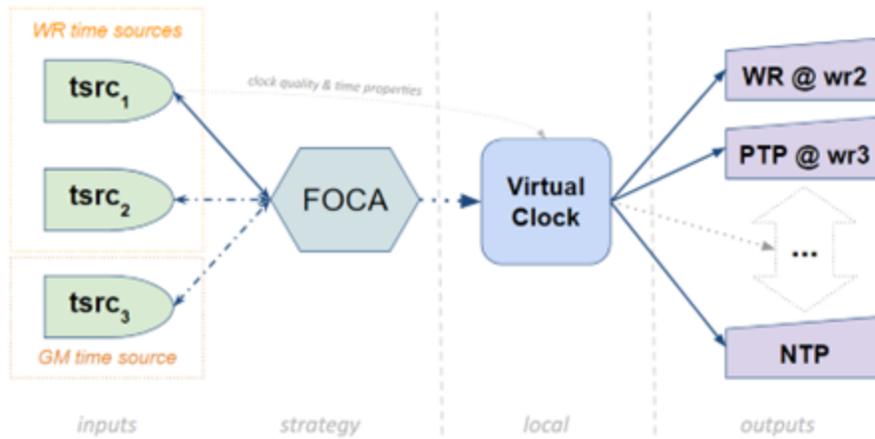
Figure 5-3: Data-flow between timing sources, virtual clock and outputs

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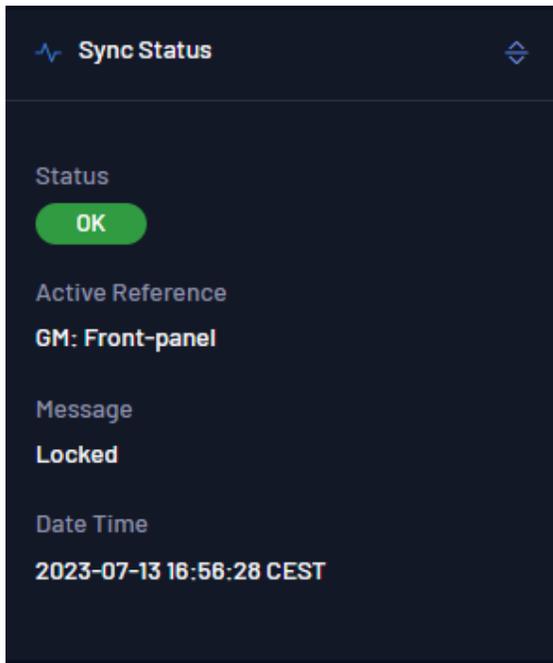
Figure 5-4: Virtual Clock Overview (Dashboard Web GUI View)

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 173.

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.

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-wr1) 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 trans-ition (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 meth-ods. 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 guar-antee 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 sig-nal. 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 ref-erence 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 ref-erence 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 out-put 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 173).

5.2.1.4   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.

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.

5.2.1.5   WR Slave @ wr0 / PTP Fan-Out

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

» The primary timing source is provided using WR protocol through interface wr0.

» The wr1 is configured as WR master and the ethernet ports (i.e., eth0 & eth1) are configured as PTP masters.

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.

> **Note:** PTP Master configuration: This preset only configures 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.

5.2.1.6   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.

» The ethernet ports (i.e., eth0 & eth1) are configured as PTP masters.

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.

> **Note:** PTP on copper ports (eth0 & eth1): In v3.x, PTP will be activated on copper ports only while using "PTP Fanout" presets or by manually enabling while forcing the "Custom" preset. In default mode it is disabled.

5.2.1.7   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.

5.2.1.8   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.
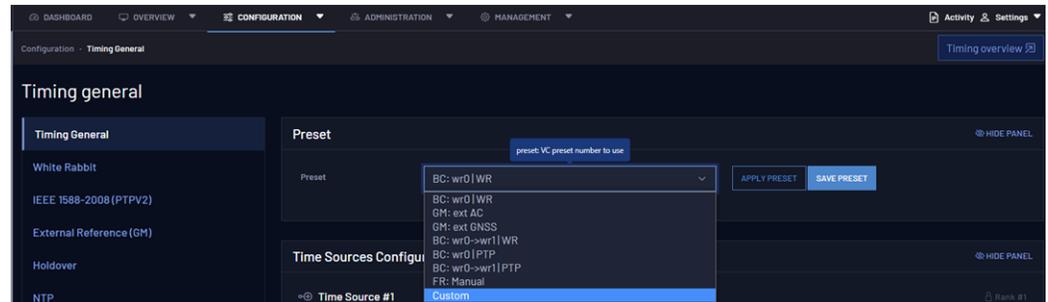
2. You will be given a warning prompt to acknowledge that your current preset settings (both Time Sources Configuration and Fanout Configuration) will be deleted from the form:
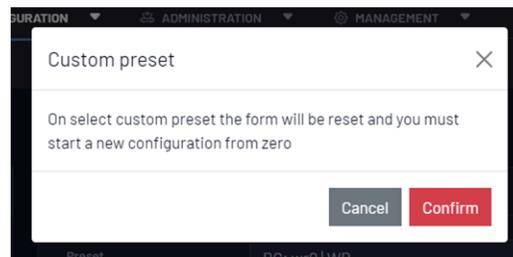
3. Modify each Timing Source in the **Timing Source Configuration** panel in the order they should be evaluated by the FOCA algorithm. If the Type field is left DISABLED, it will not be evaluated.

   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" below, the user first needs to select the preset=Custom to reveal the Ports Configuration submenu and other parameters.

Figure 5-6: Custom Preset with CLI tool

Then each port can be configured independently with:

» Protocol: WR, PTP, NTP, Disabled

» Role: Master, Slave, Auto , Survey

» 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 5-7: Port configuration (e.g., wr0) from CLI tool

> **Note:** PTP modes on Custom preset: The Custom preset allows to configure any desired interface as PTP master through Web and CLI. However, PTP slave mode in ethernet ports can only be configured through the CLI.

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:
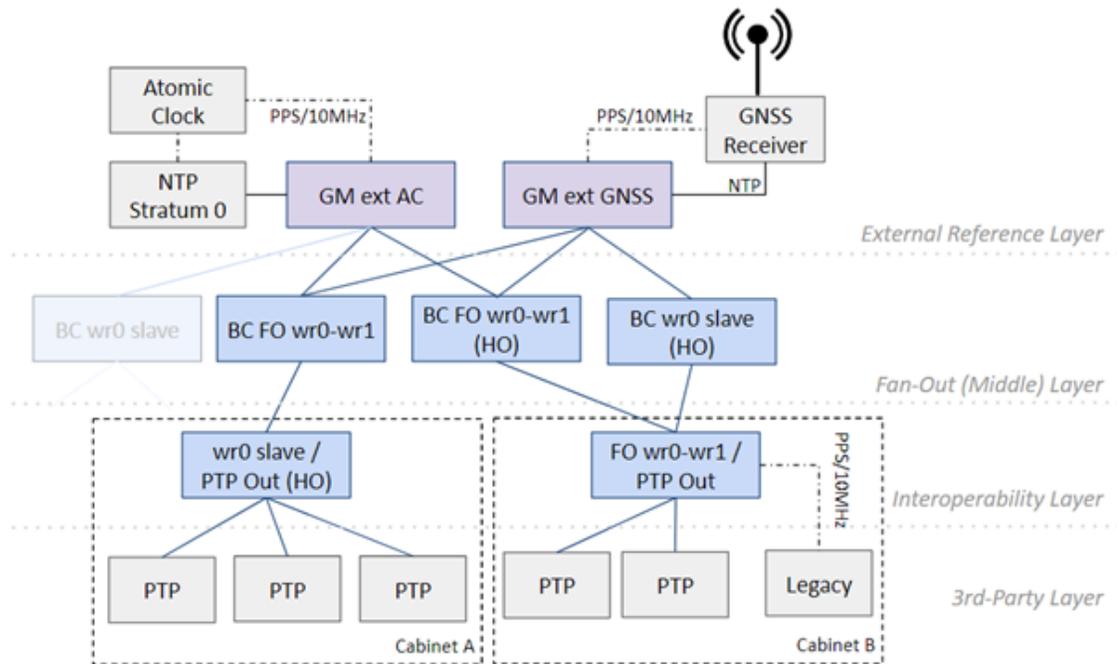
Figure 5-8: Reference topology with different presets.

» 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.
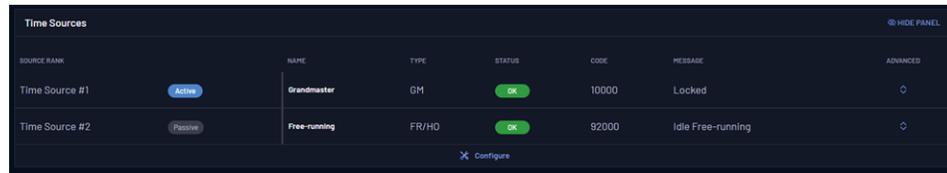
Figure 5-9: Timing Sources Overview panel

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 173) 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:
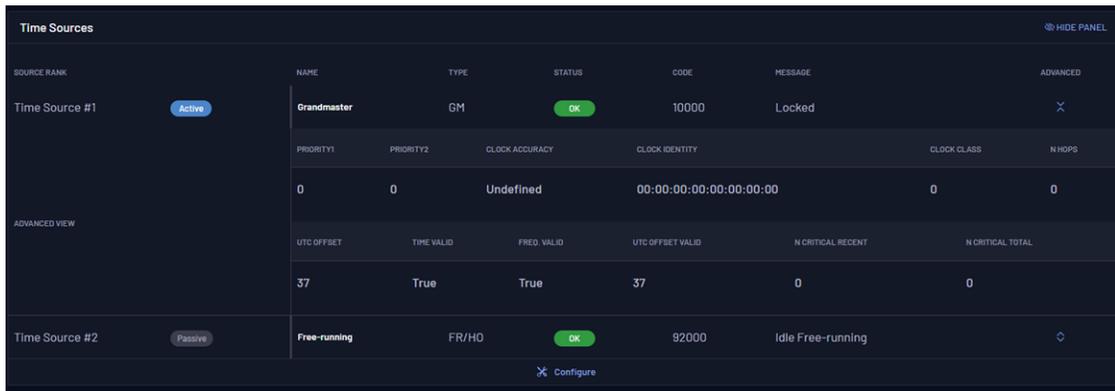
Figure 5-10: Advanced info Time Source #1

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.

5.3 White Rabbit

5.3.1 Configuration

After configuring if White Rabbit is a Slave (Timing Source) or a Master port, the only configuration that might be realized is which Transport Protocol should be used.

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.

To configure Transport Protocol, log in to the Web GUI and navigate to **Configuration** > **White Rabbit**.
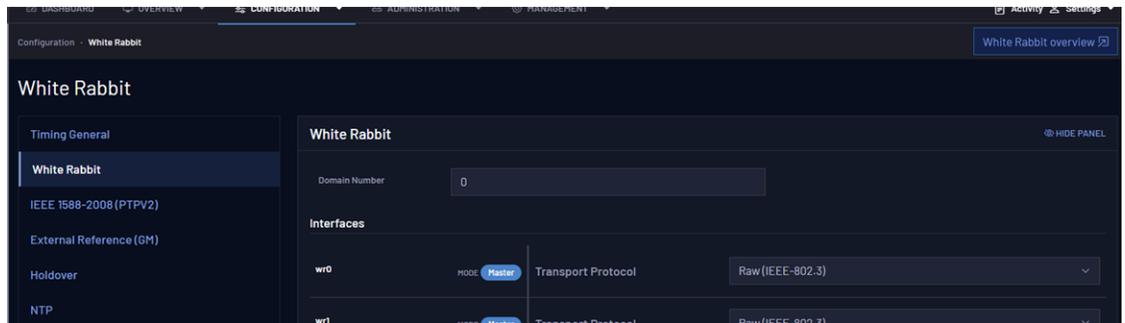
Figure 5-11: Configuration of WR instance.

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	Cable round trip time excluding fixed+semistatic (cRTT).
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'.
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.
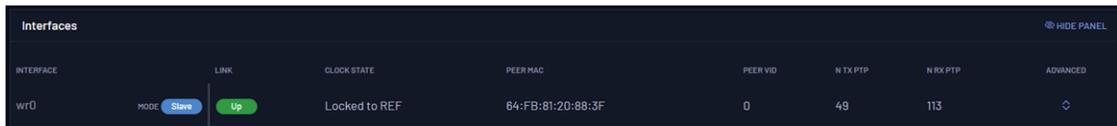
Figure 5-12: WR Interfaces overview (Only first interface captured (wr0)).

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.
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):
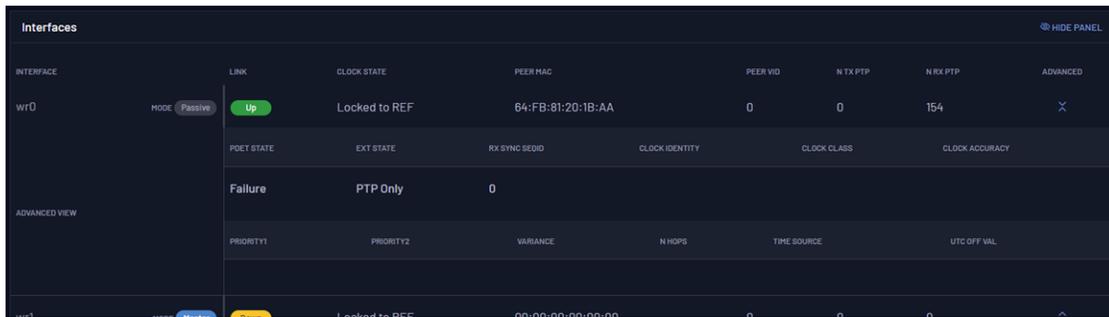
Figure 5-13: Advanced WR interface Overview (WR0 configured as slave)

> **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).**

OID	Name	Value Type	Description
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.
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   IEEE 1588-2008 (PTPv2)

The IEEE 1588-2008 (PTPv2) module offers interoperability with a wide range of 3rd party devices. It has been specifically designed to work as PTP master while supporting the following profiles:

» Default

» Telecom profiles:

   » G.8265.1

» Power profiles:

   » IEEE C37.238-2017

   » IEEE 61850-9-3

> **Caution:** Limitation on packet rate: It is not recommended to distribute PTP at full packets rate (128 pkt/s) on all its 16 network interfaces at the same time. In advanced configuration the user can modify CPU priorities to avoid dropping packets on most critical ports. Unicast negotiation can also be used to smartly decrease the packet rate depending on CPU load.

5.4.1   License

> **Note:** PTP License: A specific license must be purchased in order to get full access to the IEEE 1588-2008 (PTPv2) module. When no license is provided, the PTP instance will start with default profile and parameters. The user will only be allowed to configure the Role of the port (Master, Slave & Disabled).

The following table compares the configuration of IEEE-1588 (PTP) when using or not a valid license.

Table 5-2: IEEE 1588 configuration with/without license

PTP	With License	Without License
Mode	Auto, Master, Slave, Disabled	Master, Slave or Disabled
BMCA	Enabled or Disabled	Disabled
Transfer mode	Multicast, Unicast, or Hybrid	Multicast

PTP	With License	Without License
Unicast Negotiation	Enabled or Disabled	Disabled
Delay Mechanism	End to End (E2E) or Peer to Peer (P2P)	End to End (E2E)
Network mode	IPv4 or Ethernet (layer 2)	IPv4
Domain	0-255	0
Announce ratio	[1 msg/128s, 16 msg/s] ([-7,4])	1 msg/s
Sync ratio	[1 msg/128s,128 msg/s] ([-7,7])	1 msg/sec
Delay Req ratio	[1 msg/128s,128 msg/s] ([-7,7])	1 msg/sec
Advanced Settings		
Offset correction	Any number (in nanoseconds)	0
Servo Fit Algorithm	Normal, Soft, Hard, Hardx2	Normal

5.4.1.1 PTP license management

All the topics related to PTP license management such as purchasing, activating and checking are explained within the Licenses section ("Licenses" on page 150).

Once the license is activated (see image below) the web interface should allow the user to configure these settings as explained in "Configuration" on the next page.
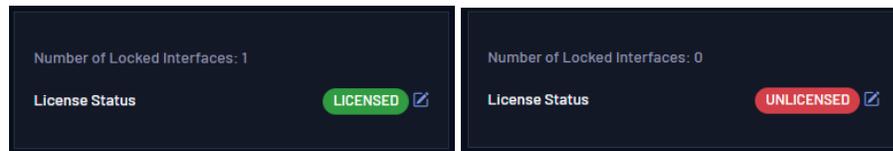
Figure 5-14: Unlicensed VS Licensed PTP daemon

> **Note:** The CLI tool allows to modify and save all the parameters listed in "IEEE 1588 configuration with/without license" on the previous page even without license. However, they will not be Applied/Loaded until the PTP license is validated.

5.4.2 Configuration

> **Note:** Master/Slave port configuration: The role of the port is given by the Preset (*"Presets" on page 65*) and cannot be modified directly from the PTPv2 configuration tab. The section *"Custom" on page 68* should be read in case the user needs to modify the role for a specific port.

> **Note:** PTP provides many options to support a wide range of applications. Then, the PTP profiles restrain some specific settings in order to limit compatibility to only a subset of PTP. The user should thus consider these restrictions when configuring PTP for a specific profile.

The parameters associated to PTP are listed below:

OID	Name	Value Type	Description
19.xx13.x	**net/<iface>/1/cfg**		**Configuration of the corresponding <iface> port instance**.
19.xx13.8	Profile	0. Default 1. Custom 2. Telecom 8265.1 3. C37.238-2017 Power 4. 61850-9-3 Power	Profile selected (Default, telecom...) with a set of options pre-configured.
19.xx13.9	Transport Protocol	5. Layer 2 6. UDP/IPv4	Define the network layer that delivers the PTP packets.
19.xx13.10	Delay Mechanism	0. E2E 1. P2P	Path delay measuring mechanism used by the PTP Port {E2E (End to End delay request-response mechanism), P2P (Peer to Peer delay mechanism) }.
19.xx13.14	Domain	<Integer> [0-255] Default: 0	Domain number associated to the PTP transactions. Several domains can work simultaneously. PTP profiles can restrict the value of domain number (i.e., 8275.1)

OID	Name	Value Type	Description
19.xx13.11	Transport Mode	0. Multicast 1. Unicast 2. Hybrid	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. The hybrid mode will send delay request and response packets via unicast and the others via multicast.
19.xx13.12	Unicast Negotiation	<Bool>	Enable unicast negotiation support. This option is needed by ITU-T G8265.1 When Unicast is enabled, this parameter will be activated automatically.
19.xx13.13	Unicast Destination	<String>	IPv4 address (or coma separated list of IPv4 addresses). Mandatory for slave unicast as they must request to a predefined GM(s). Mandatory for GMs with unicast negotiation disabled as they must deliver to a pre-configured group of slaves.
19.xx13.20	Announce Rate	<Enums> Default: 1 packet/s	Rate of announce messages transmitted to be used by BMCA/FOCA. From 1 packet each [2,4,8,16,32,128] seconds to [1,2,4,8,16] packets per second.
19.xx13.21	Sync Rate	<Enums> Default: 1 packet/s	Rate of sync (and follow-up) messages from master to slave (Figure "IEEE 1588-2008 (PTPv2)" on page 80). From 1 packet each [2,4,8,16,32,128] seconds to [1,2,4,8,16,32,61,128] packets per second.
19.xx13.22	Delay Req Rate	<Enums> Default: 1 packet/s	Rate of delay request from slave to master. From 1 packet each [2,4,8,16,32,128] seconds to [1,2,4,8,16,32,61,128] packets per second.
19.xx13.23	Peer Delayreq Rate	<Enums> Default: 1 packet/s	Rate of Peer delay request when operating in P2P mode. From 1 packet each [2,4,8,16,32,128] seconds to [1,2,4,8,16,32,61,128] packets per second.
19.xx13.18	User offset	<Integer> Default: 0 ns	User offset to compensate internal PTP delay.
19.xx13.24	PTP Timescale	<Enum> 1. PTP 2. ARB	Since we only support hardware timestamping, PTP Timescale will always be set to "PTP". ARB (arbitrary) timescale cannot be selected.

5.4.2.1 Configuration of PTP Profiles

A valid license is needed to perform the configuration of PTP profiles. If your presets allow a valid MASTER or SLAVE port state, then you will be able to configure your PTP profiles for each port by navigating to **Configuration** > **IEEE 1588-2008 (PTP)**.

The easiest way to configure a specific PTP profile is through the web interface. The user can quickly select its desired profile as illustrated in below and then all the related settings will be modified accordingly to fit the selected profile.
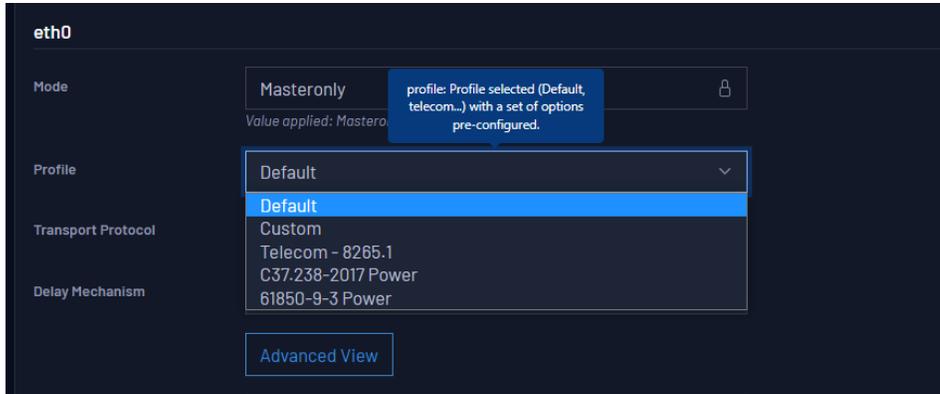
Figure 5-15: Selecting PTP Profiles

Then if some advanced settings need to be modified the user can click on the Advanced View button to expand the configuration of a given interface.

> **Note:** Custom Profile: In case some (advanced) settings are modified not accordingly to the specification of the selected profile the web interface will automatically change to use the Custom profile. It will not restrict the user to any configuration.

The tables below summarize the limitation of the configuration for the different PTP profiles supported by the WR-ZEN device. Worth mentioning that some of the settings have no default value but must fit with a specific range. Strikethrough settings are ignored by the profile and can be left empty or with previous value.

Telecom ITU-T 8265.1

The objective of this profile is to distribute frequency within 16 ppb over a network with some non-aware PTP nodes.

OID	Name	Default Value	Range		Note
			Min	Max	
19.xx13.x	**net/<iface>/1/cfg**				
19.xx13.9	Transport Protocol	IPv4/UDP	-	-	Only IPv4/UDP allowed. IP address must be set before selecting this profile.

OID	Name	Default Value	Range		Note
			Min	Max	
19.xx13.10	Delay Mech-anism	E2E	-	-	Only E2E allowed
19.xx13.14	Domain	4	4	23	
19.xx13.11	Transport Mode	Unicast	-	-	Only Unicast allowed
19.xx13.12	Unicast Nego-tiation	True			Always True for this profile
19.xx13.13	Unicast Destination	<IP>			
19.xx13.20	Announce Rate	1/2	1/16	8	
19.xx13.21	Sync Rate	1/16	1/16	128	
19.xx13.22	Delay Req Rate	1/16	1/16	128	
19.xx13.24	PTP Times-cale	PTP			

Power Profile: IEEE C37.238-2017

This profile has been created to target Power Systems Applications.

OID	Name	Default Value	Range		Note
			Min	Max	
19.xx13.x	**net/<iface>/1/cfg**				
19.xx13.9	Transport Protocol	Layer 2 (IEEE802.3)	-	-	Only Layer 2 is allowed
19.xx13.10	Delay Mech-anism	P2P	-	-	Only P2P allowed
19.xx13.14	Domain	254	0	127	The range also includes the single default value 254 ([0, 127] and 254).
19.xx13.11	Transport Mode	Multicast	-	-	Only Multicast allowed.
19.xx13.20	Announce Rate	1	-	-	Only 1 packet/s allowed
19.xx13.21	Sync Rate	1	-	-	Only 1 packet/s allowed
19.xx13.23	Peer Delay req Rate	1	-	-	Only 1 packet/s allowed
19.xx13.24	PTP Times-cale	PTP			

Power Utility Profile: IEEE 61850-9-3

This profile has been created to target Power Utility applications .

OID	Name	Default Value	Range Min	Range Max	Note
19.xx13.x	**net/<iface>/1/cfg**				
19.xx13.9	Transport Protocol	Layer 2 (IEEE802.3)	-	-	Only Layer 2 is allowed
19.xx13.10	Delay Mechanism	P2P	-	-	Only P2P allowed
19.xx13.14	Domain	0	0	255	
19.xx13.11	Transport Mode	Multicast	-	-	Only Multicast allowed.
19.xx13.20	Announce Rate	1	-	-	Only 1 packet/s allowed
19.xx13.21	Sync Rate	1	-	-	Only 1 packet/s allowed
19.xx13.23	Peer Delay req Rate	1	-	-	Only 1 packet/s allowed
19.xx13.24	PTP Timescale	PTP			

Default Profile

This profile has default settings.

OID	Name	Default Value	Range Min	Range Max	Note
19.xx13.x	**net/<iface>/1/cfg**				
19.xx13.9	Transport Protocol	UDP/IPv4	Layer 2	UDP/IPv4	IP address must be set before setting transport protocol to UDP/IPv4.
19.xx13.10	Delay Mechanism	E2E	E2E	P2P	
19.xx13.14	Domain	0	0	255	
19.xx13.11	Transport Mode	Multicast	Multicast	Hybrid	Multicast Unicast and Hybrid modes are allowed.
19.xx13.20	Announce Rate	1	1/128	16	
19.xx13.21	Sync Rate	8	1/128	128	

OID	Name	Default Value	Range		Note
			Min	Max	
19.xx13.22	Delay req Rate	8	1/128	128	
19.xx13.23	Peer Delay req Rate	8	1/128	128	
19.xx13.24	PTP Times-cale	PTP	-	-	Always PTP (only hardware timestamping supported)

5.4.3 Info/Overview

Each PTP port runs its own PTP instance independently on a specific process. The information about the instance can be obtained from command line, using for example the following command for the first port (wr0):

```
gpa_ctrl wptpd net/wr0/1/info/
```

Or through the web interface as shown below, where each port is displayed by a row (see the table "PTP information for a port interface" on the next page for more details on the provided parameters). The PTP Overview page is located at **Overview** > **Timing** > **IEEE-1588-2008 (PTPV2)**.
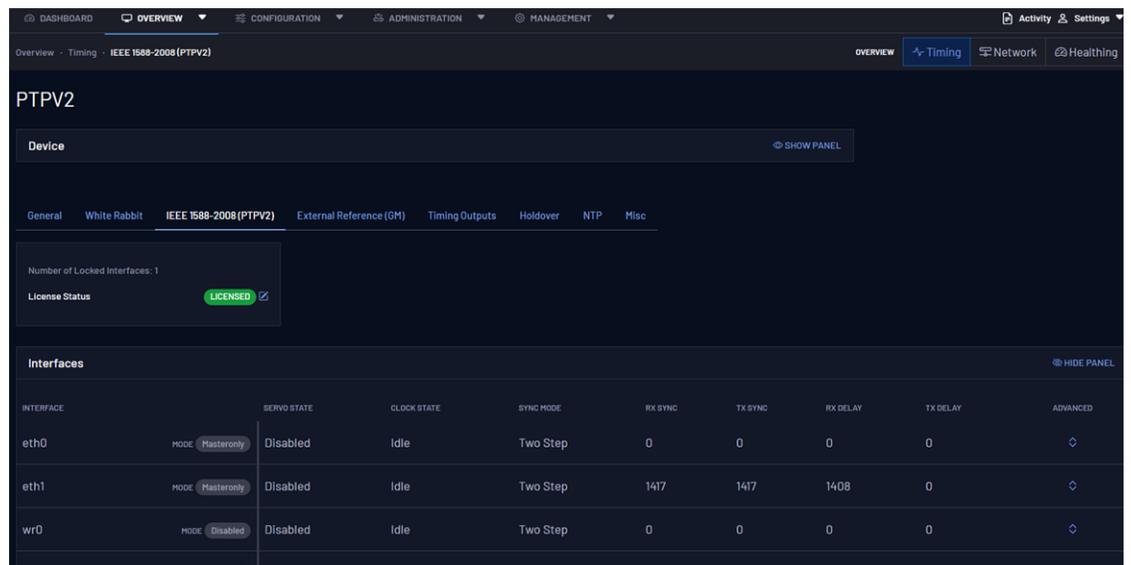
Figure 5-16: PTP Overview Menu

Table 5-3: PTP information for a port interface

OID	Name	Value Type	Description
19.xx12.x	**net/\<iface\>/1/info**		**Information on the corresponding \<iface\> port instance**.
19.xx12.2	Port State	\<Enum\> 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 WR instead of PTP. • The port state will be Master if it distributes PTP timing (Color: Purple). Finally, the other states are transition states (mainly used by BMCA) or error states.
19.xx12.12	Mode of Operation	\<Enum\> 0. Two Step 1. One Step 2. Disabled	Reports the current mode of operation received from the master. Two steps=Sync+FollowUp, One stepp=Sync
19.xx12.26	Rx Sync Packets	\<Integer\>	Number of received sync messages (Incrementing for Slave instances)
19.xx12.31	Tx Sync Packets	\<Integer\>	Number of transmitted sync messages (Incrementing for Master instances)
19.xx12.28	Rx DelayReq Packets	\<Integer\>	Number of received delay request message (Incrementing for Master instances)
19.xx12.33	Tx DelayReq Packets	\<Integer\>	Number of received delay request message (Incrementing for Slave instances)
19.xx12.64	Rx PeerDelayReq Packets	\<Integer\>	Number of received peer delay request messages (Incrementing for Slave instances when a PTP exchange is completed)
19.xx12.67	Tx PeerDelayReq Packets	\<Integer\>	Number of transmitted peer delay request messages (Incrementing for Slave instances when a PTP exchange is completed)

Then if a specific interface is not disabled (Port State ≠ None), by clicling on the (+) a user will expand the interface to its Advanced View to get more information about the PTP exchange. The bi-directional exchanges of PTP is resumed in "IEEE

1588-2008 (PTPv2)" on page 80, where the number of packets for each type of messages is provided by the parameters in advanced view (see the table below).

Table 5-4: Advanced PTP information for a port instance

19.xx12.x	net/<iface>/1/info		Information on the corresponding <iface> port instance.
19.xx12.2	Offset from Master	<Decimal> (f64)	The value represents the actual offset (in seconds) between master and slave. It is calculated from the retrieved timestamps with corrections from calibrations & asymmetries settings.
19.xx12.3	One-Way Delay	<Decimal> (f64)	One-way path delay in seconds (equivalent to delay_MS)
19.xx12.26	Rx Sync Packets	<Integer>	Number of received sync messages (Incrementing for Slave instances)
19.xx12.28	Rx DelayReq Packets	<Integer>	Number of received delay request message (Incrementing for Master instances)
19.xx12.25	Rx Announce Packets	<Integer>	Number of received announce messages (Incrementing for Slave/Passive instances)
19.xx12.27	Rx FollowUp Packets	<Integer>	Number of received Follow up messages (Incrementing for Slave instances with Two Steps)
19.xx12.29	Rx DelayResponse Packets	<Integer>	Number of received Delay Response messages (Incrementing for Slave instances when a PTP exchange is completed)
19.xx12.64	Rx PeerDelayReq Packets	<Integer>	Number of received peer delay request messages (Incrementing for Slave instances when a PTP exchange is completed)
19.xx12.65	Rx PeerDelayResp Packets	<Integer>	Number of received peer delay response messages (Incrementing for Slave instances when a PTP exchange is completed)

19.xx12.66	Rx PeerDelayRespFUp Packets	<Integer>	Number of received peer delay response follow up messages (Incrementing for Slave instances when a PTP exchange is completed)
19.xx12.31	Tx Sync Packets	<Integer>	Number of transmitted sync messages (Incrementing for Master instances)
19.xx12.33	Tx DelayReq Packets	<Integer>	Number of received delay request message (Incrementing for Slave instances)
19.xx12.30	Tx Announce Packets	<Integer>	Number of transmitted announce messages (Incrementing for Master instances)
19.xx12.32	Tx FollowUp Packets	<Integer>	Number of transmitted Follow up messages (Incrementing for Slave instances with Two Steps)
19.xx12.34	Rx DelayResp Packets	<Integer>	Number of transmitted Delay Response messages (Incrementing for Master instances each time a DelayReq is received)
19.xx12.67	Tx PeerDelayReq Packets	<Integer>	Number of transmitted peer delay request messages (Incrementing for Slave instances when a PTP exchange is completed)
19.xx12.68	Tx PeerDelayResp Packets	<Integer>	Number of transmitted peer delay response messages (Incrementing for Slave instances when a PTP exchange is completed)
19.xx12.69	Tx PeerDelayRespFUp Packets	<Integer>	Number of transmitted peer delay response follow up messages (Incrementing for Slave instances when a PTP exchange is completed)

> **Note:** PTP in management ports (Eth): In scenarios with a high load of traffic, a non-optimal amount of timestamps is lost through copper ports (eth management ports). However, this behavior does not appear in optical ports. Hence, even though this functionality has been saved to keep the compatibility with previous firmware architectures, it is not recommended to use PTP through management ports.

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.

If the user selects the Custom Preset it might be able to configure the following parameters when expanding (+) to the Advanced View for the GM source type:

OID	Name	Value Type	Description
3.7110.x	**gm/cfg/xxx**		**Configuration of the GM timing source (By Preset).**.
3.7110.2	Align PPS	<Boolean>	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.
3.7110.4	Source Type	• ATOMIC CLOCK • GNSS • PTP • OTHER	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).
3.7110.7	PPS Mandatory	• YES • NO • STARTUP_ONLY	Controls whether an PPS input signal is needed to enter/stay active for the GM source.

The user can also configure the following parameters independently from the chosen preset:

OID	Name	Value Type	Description
3.7110.x	**gm/cfg/xxx**		**Configuration of the GM timing source (By Preset).**.

OID	Name	Value Type	Description
3.7110.1	GM Offset	<Integer> Default: 0	Offset to compensate user cable delay for PPS input (in picoseconds). 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	Priority1	<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).
3.7110.3	Priority2	<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).
3.7110.11	Clock Accuracy	<Enum> Default: Unknown	It announces the expected accuracy provided by the external reference. It shall be conservatively estimated based on the type of time source (e.g., Atomic Clock <= 1ns, GNSS receiver <= 50ns).
3.7110.15	Leapsec file ignore	<Enum>Default: OFF	Leapsec_file_ignore disables the critical warning when the leap second file is expired.

> ⚠️ **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 General** > **External Reference (GM)**, where the user can easily audit the condition of its external reference (see figure below).
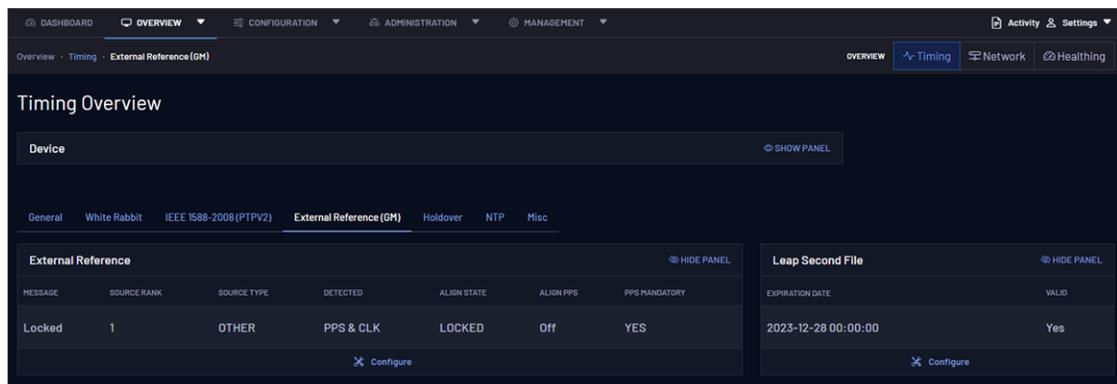
It basically offers a readback of the configuration value (Source Type, Source Rank, Align PPS and PPS Mandatory) as detailed *"Configuration" on page 91*, along with an user friendly Message that summarize the state of the GM internal state machine and a Detected value that reports the situation with 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>	User friendly message that summarizes the current state of the GM timing source.
3.7120.2	PPS Detected	• 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>	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> • No • Yes	"Yes" if the leap seconds file is valid. "No" if it is expired or missing. See *"Update Leap Seconds File" on page 101* for updating this file.

5.6   NTP

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

> ℹ **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.

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.
3.7005.4	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.
3.7005.7	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 96 for more information

> ⚠ **Caution:** Reboot to apply NTP provider parameter: Many of the settings to configure the NTP server are only loaded during initialization of the device and thus a reboot might be needed in order to apply them.

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 68 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.**.
3.7001.2	Refresh Rate	<Integer> • Default: 30	Time lapse between NTP server queries (in seconds).
3.7001.3	Retries	<Integer> • Default: 5	Number of retries for NTP server queries performed at the initialization of the device.

The second panel is a table where each column represents an NTP timing source:

OID	Name	Value Type	Description
3.70x0.x	**ntp/x/cfg/xxx**		**Configuration of the x NTP timing source (x= [1-5]).**
3.70x0.0	IPv4 Server	<String>	IP or URL of the reference NTP server.

> **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]).**
3.70x5.1	Server Status	• Disabled • OK • NTP sync error • NTP stopped replying	NTP server status. Warns if the NTP server cannot be reached.
3.70x5.0	Offset	<Integer>	Time offset between the device and the NTP reference server (in seconds).

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.

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.

5.7 Holdover

The WR-ZEN can be ordered with an optional holdover oscillator (OCXO) in order to ensure an accuracy of 1.5Qs 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.
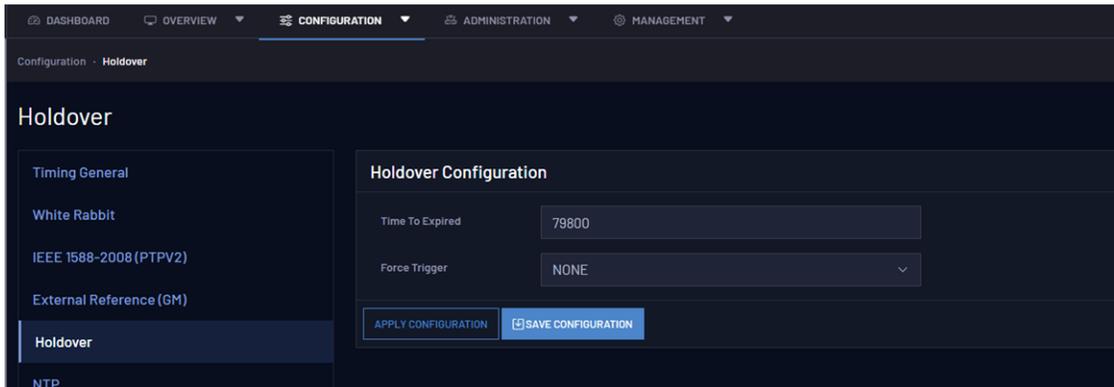
OID	Name	Value Type	Description
3.7210.x	**holdover/cfg/xxx**		**Configuration for the Holdover if available**
3.7210.0	Source Rank	<Integer> Default: 0	Source rank of the holdover as timing source. If leaved at zero it will be always placed as the last timing source. Then the user can allow to trigger 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" below.

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 hold-over option but the device does not detect it, please contact "Technical Support" on page 172.
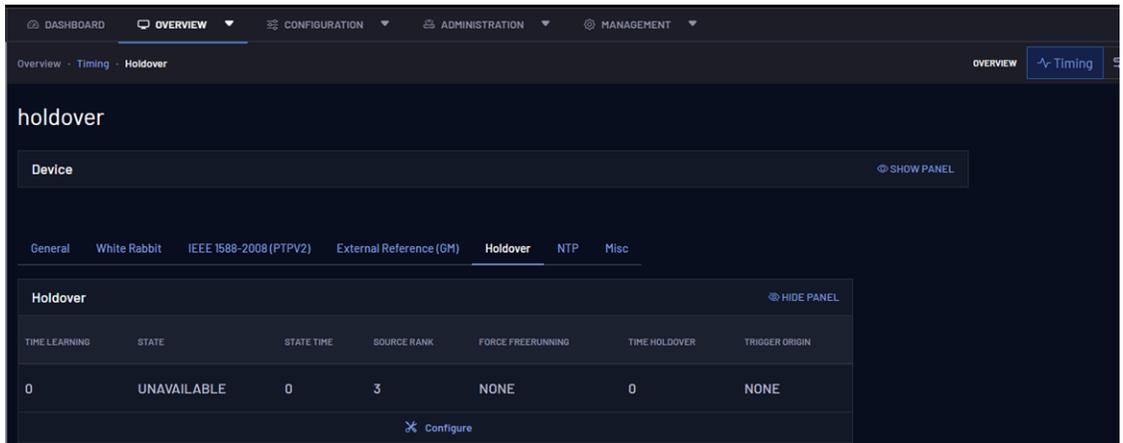
Figure 5-18: Holdover overview

OID	Name	Value Type	Description
3.7220.x	**holdover/info/xxx**		**Information about the Hol-dover 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 intro-duction 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 activ-ation

OID	Name	Value Type	Description
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.**
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 through the **Configuration** > **Misc** panel:
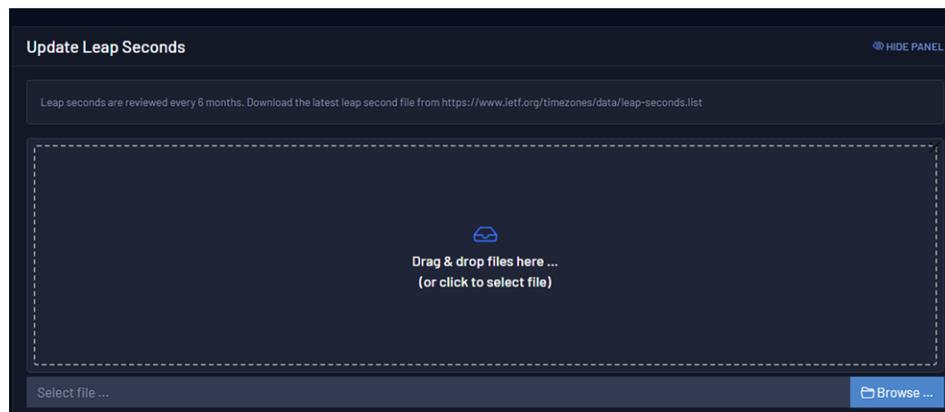

Figure 5-19: Manual Leap seconds update.

> **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-ZEN 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 26).

```
ssh root@192.168.7.25
```

```
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 '192.168.7.25' (ECDSA) to the list of
known hosts.
```

```
Welcome to WR-ZEN
```

```
wr-zynq-os version: v3.2-RC1
```

```
root@zen-425:~#
```

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

```
ssh root@192.168.7.25
```

```
Welcome to WR-ZEN
```

```
wr-zynq-os version: v3.2-RC1
```

```
root@zen-425:~#
```

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-ZEN 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-ZEN 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 186.

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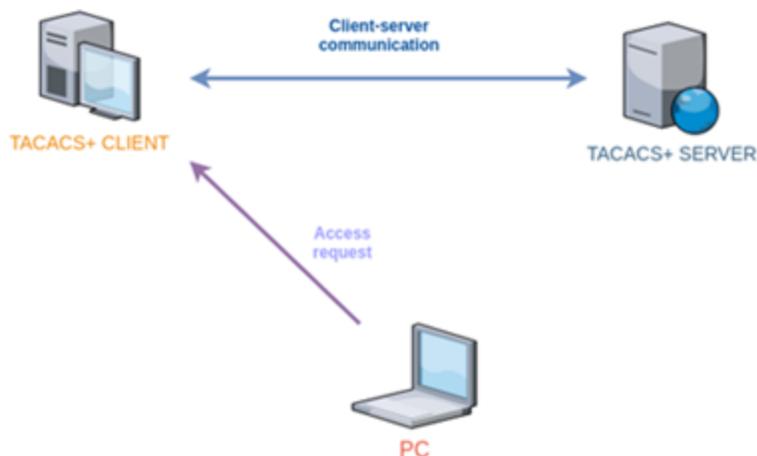
Figure 6-2: TACACS setup for verifying the installation.

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-ZEN 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-ZEN 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 con-
> figured 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 con-
> figuration 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-ZEN 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 186.

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.
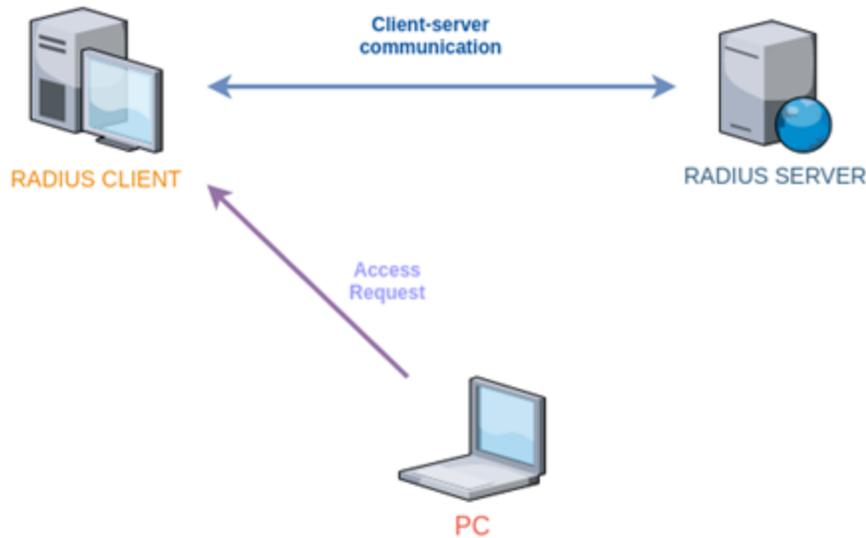

Figure 6-5: Set-up RADIUS for verifying the installation

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-ZEN.

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-ZEN 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:

```
→ ~ ssh raduser1@192.168.1.145
Password:
Password:
Last login: Fri Jun 10 04:47:54 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 6-6: SSH connection with the WR-ZEN board

Figure 6-7: Freeradius failed attempt with debug information

> **Caution:** When TACACS and RADIUS work and have been con-
> figured 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 con-
> figuration 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 182) 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-ZEN 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 under the Management > Misc > Dumplog tab or from the CLI executing the following command:

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

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.

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).
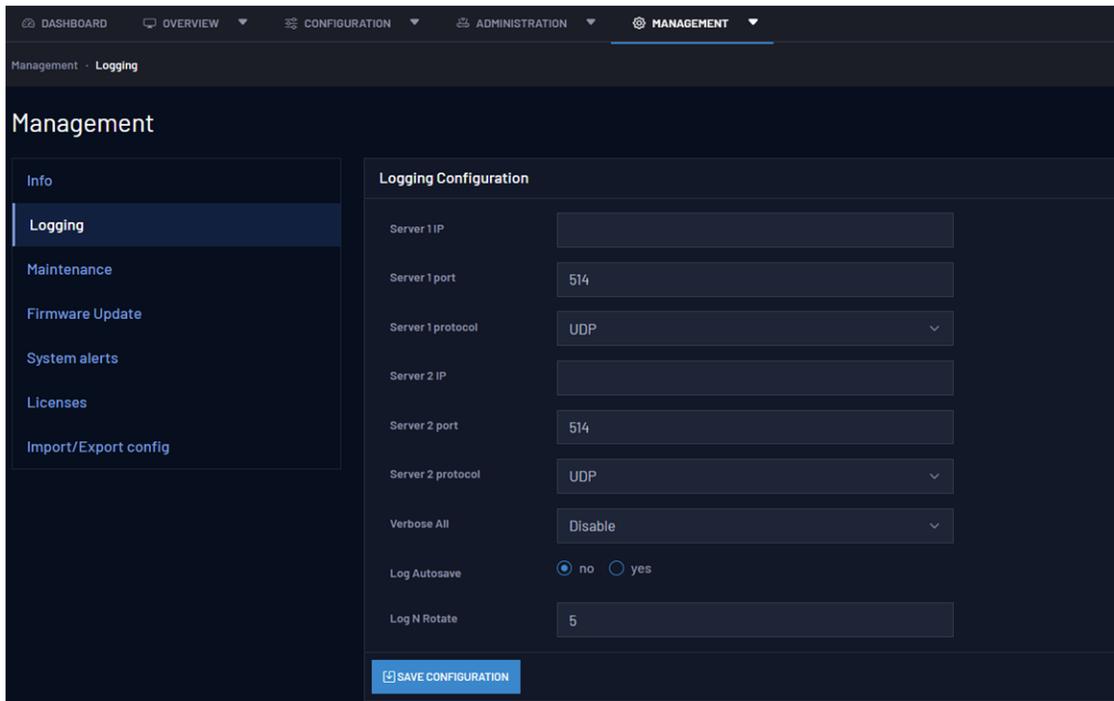
7.2 SNMP

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 /us-

r/share/snmp/snmpd.conf can be used as a reference to modify the /etc/s-nmp/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:
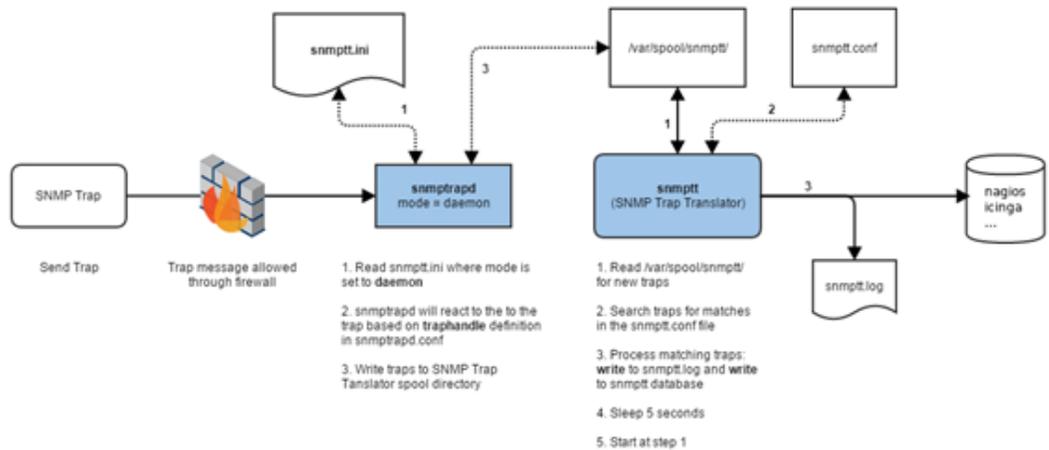
Figure 7-3: SNMP traps scheme.

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 pre-defined 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

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

OID	Name	Value Type	Description
13.3900.1	Community name	<String> (i.e., public)	Name of the community.
13.3900.2-5	NMS IP 1-4	<IP address> (i.e., 192.168.1.5 or 192.168.1.0/24)	Authentication encryption protocolDestination NMS IP address.
13.3900.6	Start/shutdown	<Enum> • Enabled • Disabled	Enable/disable startup and shutdown traps.

OID	Name	Value Type		Description
13.3900.7	Modules Start/Close	\<Enum\> • • Disabled	Enabled	Enable/disable module launch or close traps.
13.3900.8	Prms tracked	\<Enum\> • • Disabled	Enabled	Enable/disable tracked parameters traps.
13.3900.9	Prms alert	\<Enum\> • • Disabled	Enabled	Enable/disable traps when a parameter is in an alert status or back to a normal status.

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

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.

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

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-ZEN 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.

```
*** LLDP configuration parameters ***
enable (y)  --->
```

Figure 7-4: LLDP configuration from CLI

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 (v5.0), 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.

OID	Name	Value Type	Description
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.
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.

OID	Name	Value Type	Description
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.**
20.xx30.0	Address	<String> (i.e., 192.168.7.36)	Management address of the remote peer.

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.:
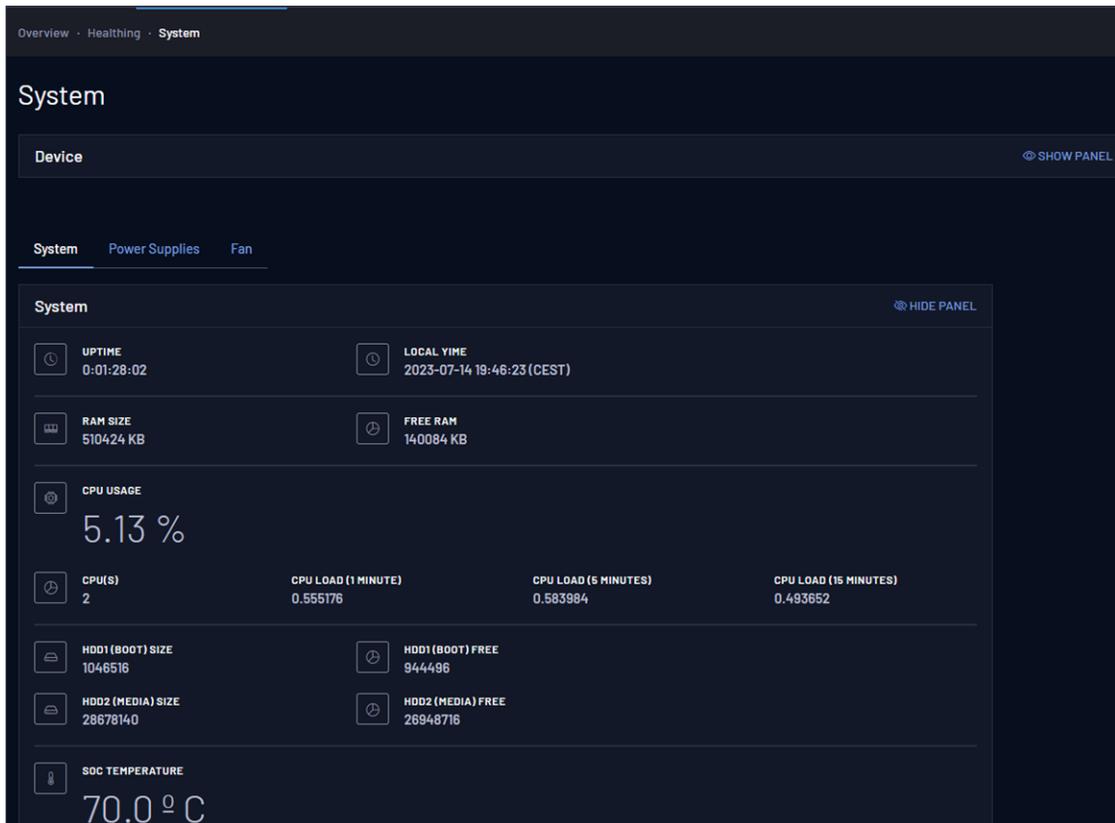
Figure 7-5: Healthing System Overview

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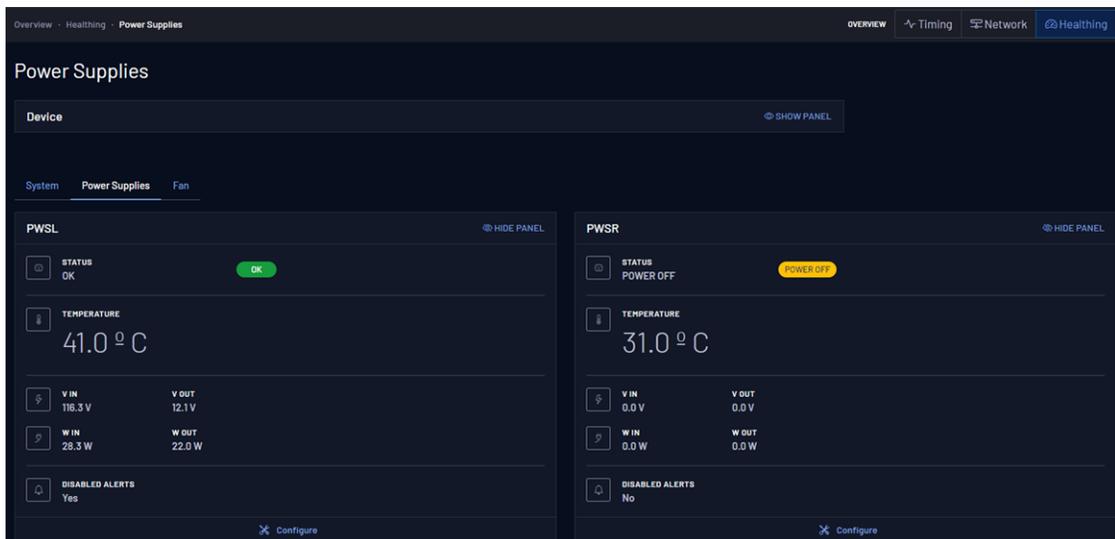

Figure 7-6: Healthing Power Overview

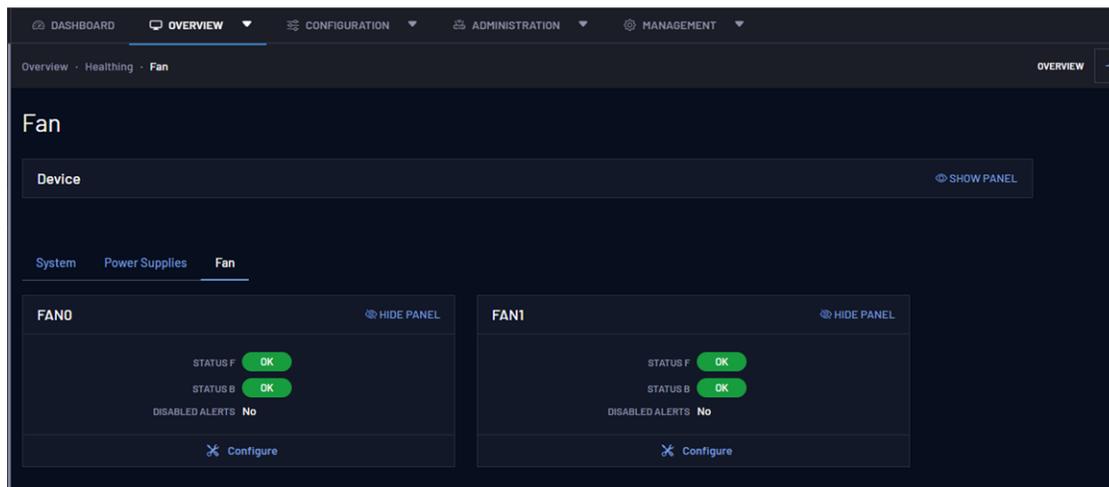And the Healthing Fan Overview details all necessary fan state information:

Figure 7-7: Healthing Fan Overview

Alternatively, the LLDP can be configured 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:

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**:
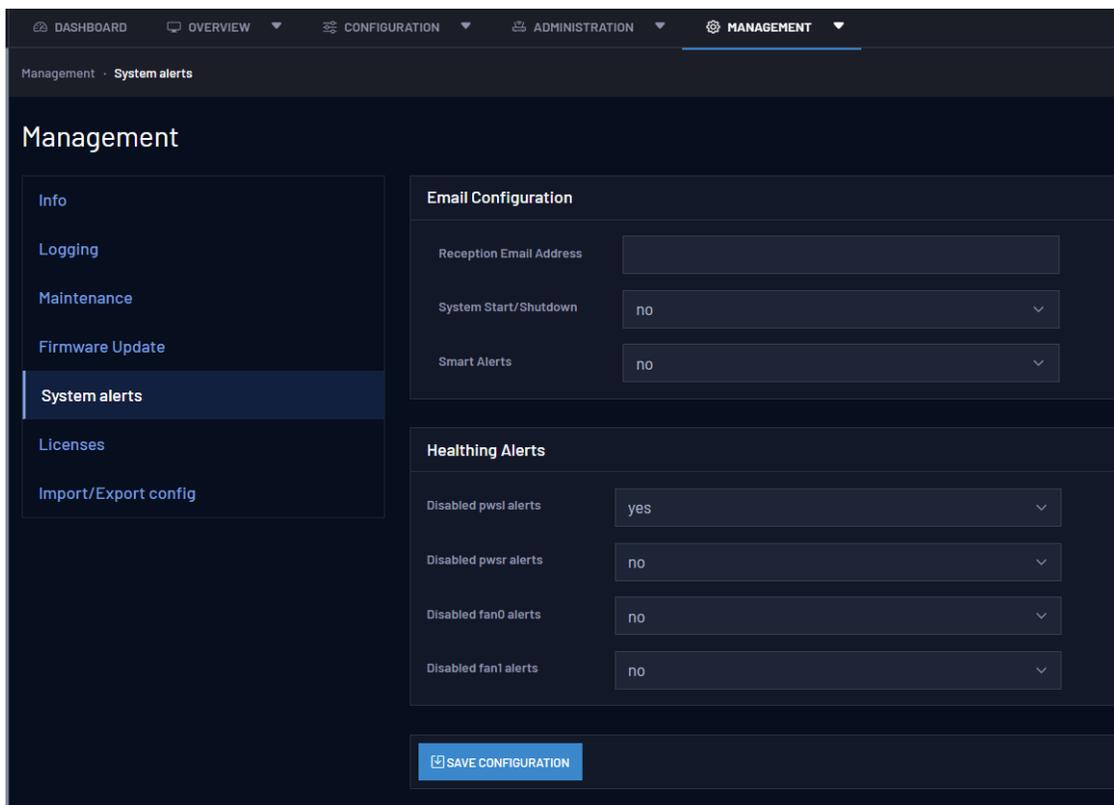

Figure 7-9: Healthing Web GUI settings

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-ZEN device.
2.1000.1	Screen saver delay	<Integer> (i.e., 60)	Not used in WR-ZEN device.
2.1000.2	Screen contrast	<Integer> (i.e., 255)	Not used in WR-ZEN 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.

7.5   LCD screen menu

The WR-ZEN family includes an LCD screen where you can see some configuration values and the state of the device.

7.5.1   Screen Saver

This is the screen you usually see when not interacting with the device. This is a screen saver screen, showing the IPs of each interface to connect to the device. When you press INFO/CTRL button the main menu will appear.

You can configure the screen saver options with these parameters:

» cfg/screen_saver:

  » 1: screen saver configured.

  » 0: no showing screen saver (the screen turns off when screen_saver_dealy is reached).

» cfg/screen_saver_delay: time without pressing INFO/CTRL button before active screen_saver.

You can change these values through gpa_ctrl or snmp.

7.5.2 Main Screen

When you press INFO/CTRL button, or if the screen saver is not active, you can see the main screen.

The main screen has the next structure:

This screen shows the general status of your device. Status can be:

» Ok: All is ok.

» Alert: There is at least a parameter with warning values.

» Critical: There is a problem with at least one parameter value or module.

If you press INFO/CTRL button, the main menu will show.

7.5.3   Main Menu

To select an option, long press over INFO/CTRL button. To go to the next option, short press over INFO/CTRL button.

The different options in this menu are:

» REVIEW PARAMETERS: Select this submenu to review the status and values of all modules and parameters in the device.

» NETWORK: This submenu allows you to review the eth0 and eth1 info and white rabbit network status.

» WR STATS: It shows info about white rabbit statistics.

» MANAGEMENT: It allows to execute simple predefined actions like rebooting the device or restore the default configuration.

REVIEW PARAMETERS
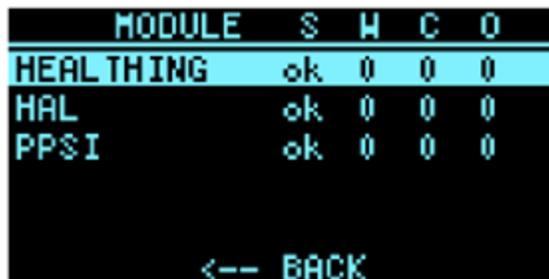
This screen shows the global status of the modules running in the device.

» S: Status:

  » OK: all is ok.

  » ! (warning): at least one parameter in warning range.

  » X (critical): at least one parameter in critical range, or there is a problem with the module.

» W: All parameters in warning range.

» C: All parameters in critical range.

» O: Out of sync.

You can view details of each module by pressing long INFO/CTRL button over the module you want to review.

The module status description is shown. The different options to select in this screen are:

» WARNINGS(n): select this option to review parameters in warning range.

» CRITICALS(n): select this option to review parameters in critical range.

» OUT OF SYNC(n): select this option to review parameters which are out of sync.

» SHOW ALL PARAMS: select this option to review all parameters in the module.

Where n is the number of parameters in that condition.

If you select any of the above options, the screen will show information about the parameters that caused the special status. If there are 0 parameters with a special status, it will not enter the sub-menu.

To exit this screen, long press over INFO/CTRL button. To go to next parameter, short press over INFO/CTRL button. When the last parameter was reached, a short press over INFO/CTRL exits this screen, returning to the previous menu.

NETWORK

In this section you can review the parameters regarding the network interfaces of the device.

The options are:

» ETH0: information about copper interface Ethernet 0.

» ETH1: information about copper interface Ethernet 1.

» WR0: information about fiber interface White Rabbit 0.

» WR1: information about fiber interface White Rabbit 1.

When you select ETH0 or ETH1 option the details of the Ethernet interface will appear:

When you select WR0 or WR1 option the details of the White Rabbit interface will appear.

WR STATS

In this sub-menu you can review the information about white rabbit:

MANAGEMENT

In this section you can perform simple predefined actions like rebooting the device or restore the default configuration.

The options are:

- » POWER OFF: Power off the device in a safe way.
- » REBOOT: Reboot the device.
- » REBOOT WR: Reboot white rabbit.
- » TOGGLE M/GM: Activate/Deactivate Grand-Master mode.

» FACTORY RESET: Reset the value of all parameters to factory configuration.

Once you select an option by long pressing INFO/CTRL button, a countdown is shown:

By press INFO/CTRL button you can cancel the execution of the option selected. When countdown reaches 0, the option will be executed.

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-ZEN are:

Group	Feature Name	Description
PTP	ptp_pro-file_cfg	Unlock the configuration of different options to enable other profiles than the default such as the telecom profile ITU-T G.8265.1 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:

> **Note:** Matching Device hardware: Safran distributes different WRZ-OS firmwares according to the hardware family and version of the device. The user must follow indications provided in "Firmware Update " on page 157 to get the corresponding firmware before proceeding to its update.
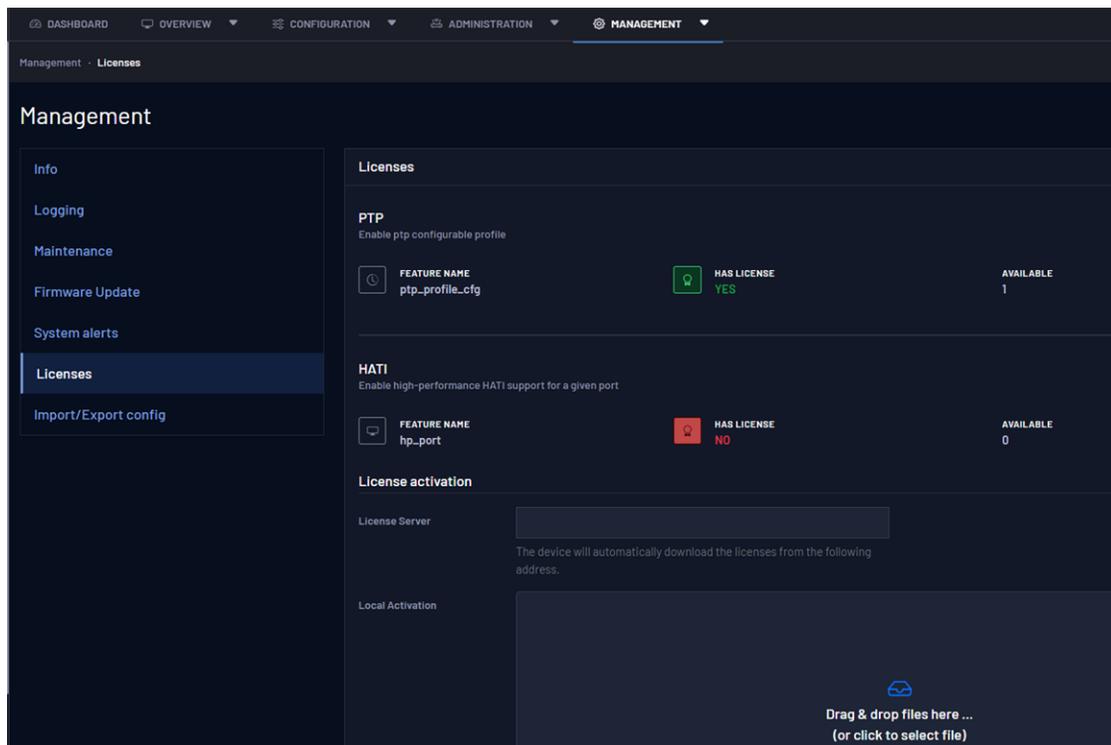
Figure 8-1: Checking available licenses.

Each possible license available for the WR-ZEN devices is represented by a single box where its status is summarized by the color of the box:

» Red: License is not available (See Figure 8:1 - Checking available licenses.).

» Yellow: License is in trial.

» 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.

OID	Name	Value Type	Description
12.2xyy.8	Available	<Integer>	Total of available corresponding licenses by device (i.e., features associated to port might need up to 16 licenses).
12.2xyy.15	Description	<String>	Description of the corresponding license feature.
12.2xyy.20	Has License	NO YES TRIAL ONLY	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@nav-timing.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 clicking on the following link:

https://flex1667.flexnetoperations.com/flexnet/operationsportal/logon.do

8.1.4.1 Map a feature to a device

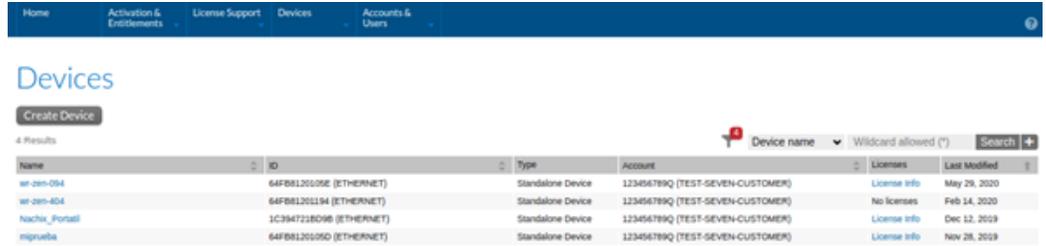» The user must first navigate to the tab Devices > Devices.

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.
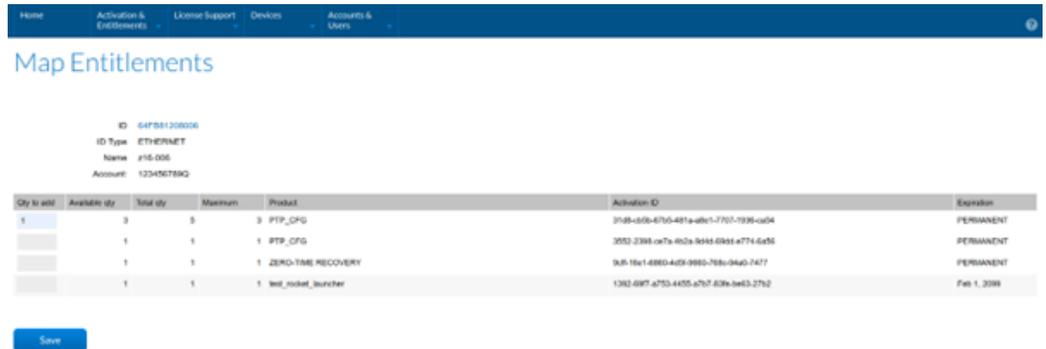
Mapping Purchased Licenses to Device.

» 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 click on Create Device Button.

» The 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 access to the Management > Licenses within the web interface of the device.
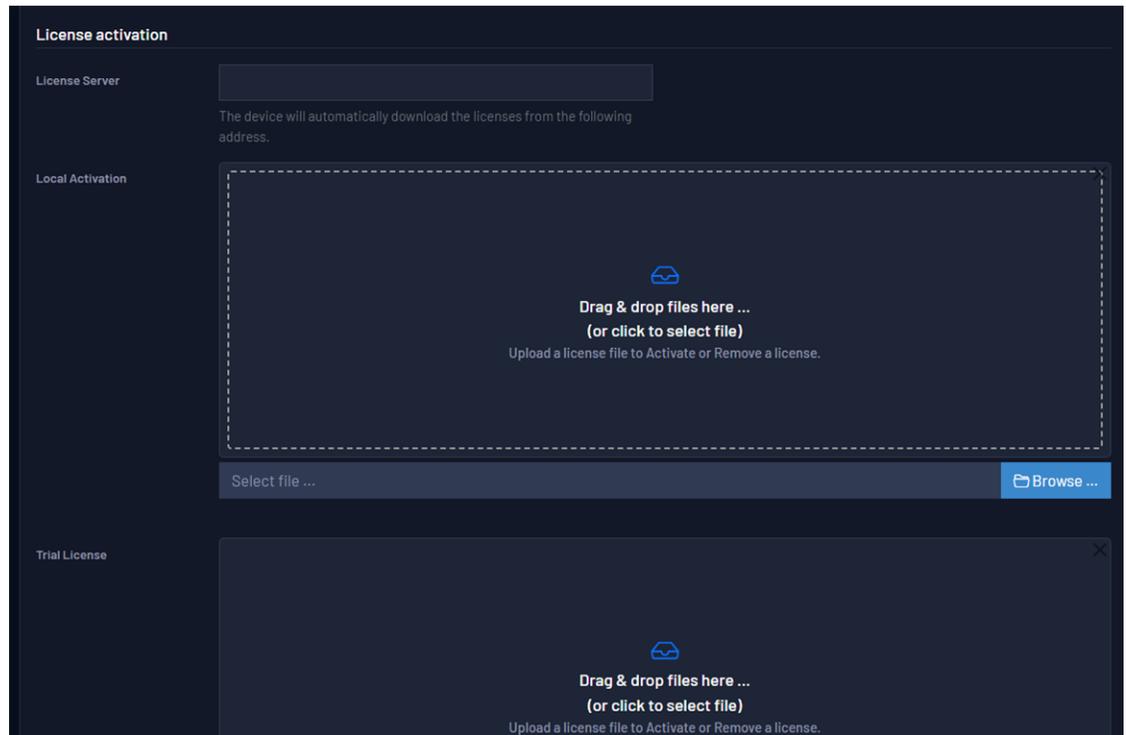
Figure 8-3: Licenses Configuration Panel

Then under Local Activation or Trial License, the user should Browse to the downloaded license file, or drag and drop the file into the labeled box. Upload it to device and finally Save & Apply the changes

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 page 154

> **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.

8.1.5   License Server

It is not recommended to use local license management when running multiple (>10) devices in the network. A license server can be setup in the management network so that all the devices can directly request an active license to enable a feature. This solution has the following advantages:

» Only license server must synchronize to license portal in order to get all the purchased license. This synchronization can be done online (seamless) or offline (using a file).

» Each device only needs to configure the License Server IP.

» The license server is viewed as a poll of license that will distribute licenses to the device only when they need them. This means that a license does not need to be associated to a specific device (e.g., an offline/backup device will not consume any licenses until it is connected to the network).

In order to get more information about this alternative, contact Safran for assistance on the solution that best fits the topology and help through its setup.

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" below 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 2.X to 3.X or vice-versa), the configuration on the device must be removed and configured again.

8.2.1 Hardware version and firmware

The HW version is displayed on the dashboard page in the Versions overview panel. The device below is a WR-ZEN TP that mounts a ZENv3.3 as main board.
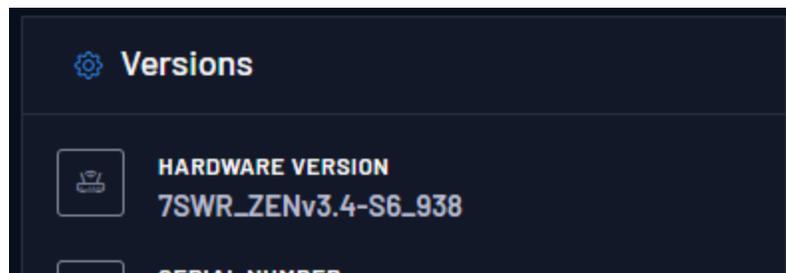
Figure 8-4: HW version displayed in dashboard.

The WR-ZEN TP 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>-ZENx.x_binaries.tar`

» `wr-zynq-os-v<XXX>-<YYYMMDD>-ZEN3.x_binaries.tar`

» `wr-zynq-os-v<XXX>-<YYYMMDD>-ZEN_binaries.tar`

But not the one that correspond to another device:

» ~~`wr-zynq-os-v<XXX>-<YYYMMDD>-Z16x.x_binaries.tar`~~

Or the one that corresponds to another hardware version

» ~~wr-zynq-os-v<XXX>-<YYYMMDD>-ZEN2.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 26), 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 auto-matically start the upgrade procedure and reboot (twice) the device. Please wait in this screen until the procedure completes.
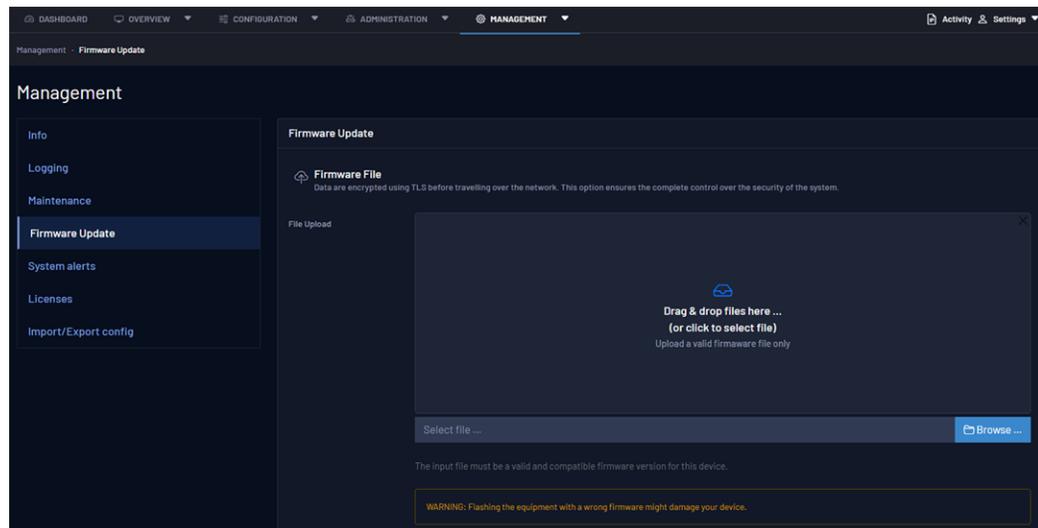
Figure 8-5: Update Procedure Waiting screen.

» If the uploaded version is lower than 5.0, 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 con-firmed that this is the way to fix a specific problem.

» If the hardware version is higher or equal to 5.0 and the software version is lower than 3.4, 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 5.0, 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 current 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 26) 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:
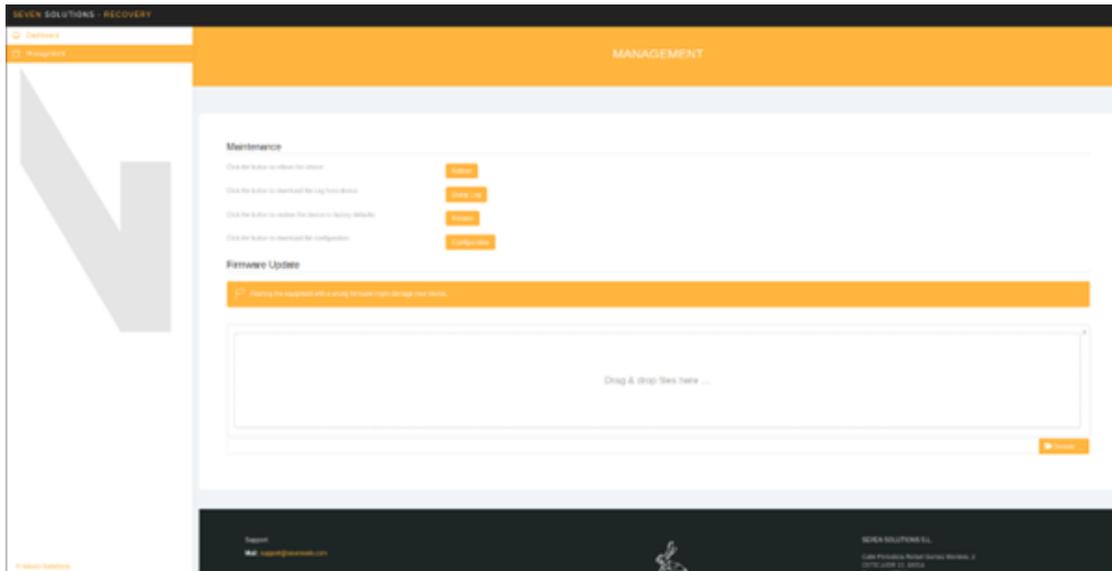

Figure 8-6: Management panel in recovery mode

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 172 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 16).

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).

8.4 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).

8.4.1 Reset via Front Panel LCD Screen

Using the LCD screen, press the info/ctrl button.

Access the following path: Management > Factory Reset.

8.4.2 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 a reset factory message appears on LCD.
4. Wait until the device reboot with default factory parameters.

8.4.3 Rest 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.

8.5 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-ZEN 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-ZEN 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-ZEN's console, type:

```
root@zen- 425:~#  gpa_ ctrl  hald  /spll/ext/fpanel/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-ZEN6 sync status by typing:

```
gpa_ctrl healthingd
gpa_ctrl tmgrd vclock/info/
```

The WR-ZEN web interface

You can access the WR-ZEN 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-ZEN with the SNMP commands.

After having installed SNMP on your host, all parameters on the WR-ZEN can be checked by typing:

```
$ snmpwalk -v2c -c public <WR-ZEN 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 () 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).
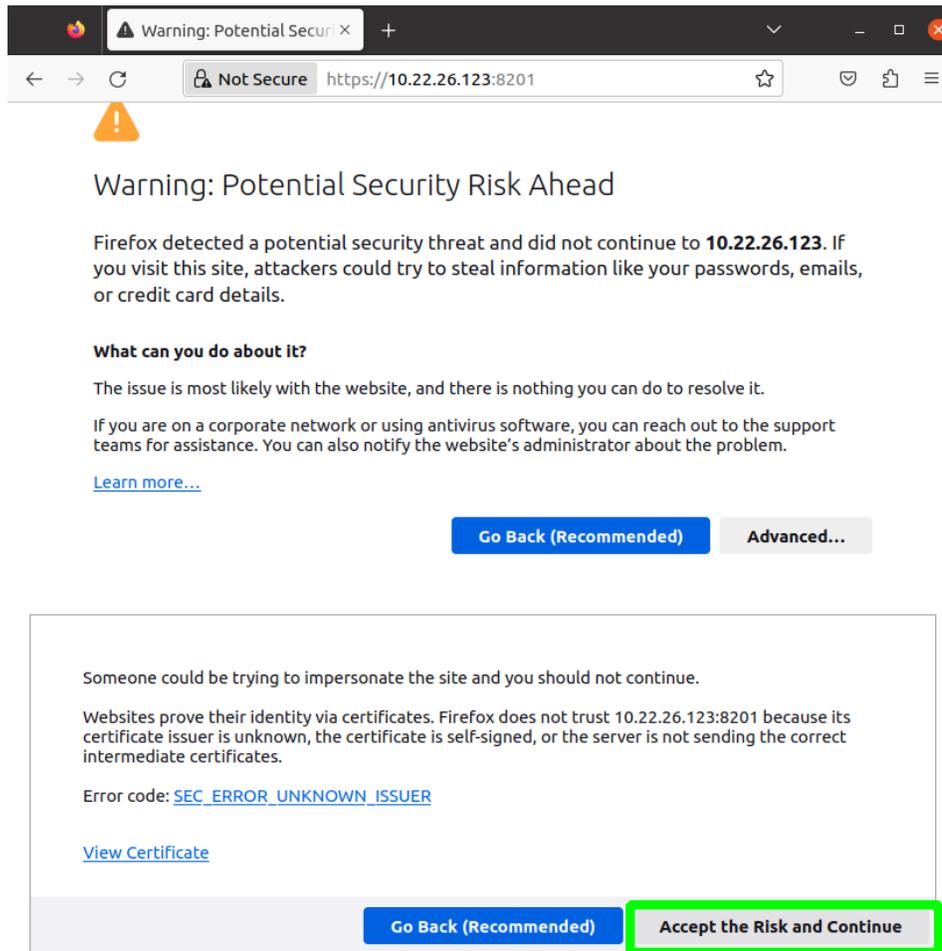
Figure 9-1: Accepting an untrusted certificate in Firefox.

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 172), 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-ZEN'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-ZEN 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-ZEN 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-ZEN 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-ZEN, 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 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-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-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-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-10110		WARNING	Locked: PPS not present	6	GM: Front-panel	In case we are using an Atomic Clock as the main reference and someone has unplugged the PPS cable we might have some problem on the next reboot. PPS defined as not man-datory by user
VSC-10201		WARNING	Locked: Time of Day was not set (NTP error)	6	GM: Front-panel	We have the same ref-erence on all the net-work but the given ToD is not be valid: Dur-ing boot we can not reach NTP server (timeout or IP not con-figured). The ToD used is provided by release ToD or last shutdown ToD.
VSC-10202		WARNING	Locked: Leap seconds file has expired	6	GM: Front-panel	Leapsec file in the GM device is expired or has reach 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	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-10203		WARNING	Locked: ToD offset bigger than 1s (NTP offset)	6	GM: Front-panel	We are seeing a drift with the current NTP offset (this is probably that NTP server has some server, but it could be that our external reference is in free-running)
VSC-10204		WARNING	Locked: NTP does not reply any-more	6	GM: Front-panel	As NTP does not reply we might have some problem with the net-work. This is not critical for operation but might be a problem at next reboot. Only the GM should send alert

9.4.2   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 provide all the inform-ation properly set
VSC-20001		OK	Locked	6	BC: PTP @ ifname	The BC clock is locked using PTP and the upstream device provide all the inform-ation properly set

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
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-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	HO	Device Status	Message	Clock Class	Active Reference	Description
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-20305		CRITICAL	Can not lock to reference	248	BC: WR/PTP @ ifname	We receive announce PTP message, 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-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-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-21320		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-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-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-21201		CHANGEOVER	Upstream GM in Free-Running	187	BC: Holdover	GM announce itself to be now in FR, if we have an active HO we should exit through this state and fail to the HO timing source

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 restart-ing/re-eval-uation of GM
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

9.4.3   Others

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-09000		OK	Timing source is ready	X	X	For ports, "Ready" means that the link is up and some announce messages sent by a master has been received. For GM, "Ready" means that 10M/PPS has been detected prop-erly

VSC	HO	Device Status	Message	Clock Class	Active Reference	Description
VSC-09100		CRITICAL	System Error	248	Internal Oscillator	In case we get an unexpect behavior because some of the src modules (ppsi,pt-pd,gnss,hald) has crashed, The tmgr will enforce this status in order to alert to the managment and also to other nodes (if possible) that our current situation is unexpected.
VSC-09110		WARNING	Initializing...	248	Internal Oscillator	The device will always initialize with the following configuration before using any policy/strategy/src in tmgr. The following value are only written in case the user run: /etc/init.d/tmgrd restart
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-92000		OK	Iddle Free-running	248	Passive	Passive state of the free-running timinig source when Iddle
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 but and enter 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	<previous message>	<prev>	<prev>	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-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 Timing outputs

In the following chart, the relevant electrical information about the timing outputs in several WR-ZEN family devices is shown. Note that some of these devices are equipped with expansion cards (e.g., FPO in the case of the WR-ZEN TP-FL).

Device	Channel	Termination at the receiver device	Mode	High voltage	Raise time
WR-ZEN	1PPS OUT	50 Ohms (optimal) 1 MOhm (over-shoot)	1PPS	~ 3.0 V	0.60 ns

Device	Channel	Termination at the receiver device	Mode	High voltage	Raise time
WR-ZEN	CLK TTL	50 Ohms (optimal) 1 MOhm (over-shoot)	10 MHz	~ 3.0 V	0.60 ns
WR-ZEN TP-FL FPO v1.0	A & B	50 Ohms (optimal) 1 MOhm (over-shoot)	xPPS	~ 3.0 V	1.10 ns
WR-ZEN TP-FL FPO v2.0	A & B	50 Ohms (optimal) 1 MOhm (over-shoot)	xPPS & 10MHz	~ 3.0 V	0.80 ns
WR-ZEN TP-32BNC EXP32BNC V2.0	A & B C & D	1 Ohm (optimal) 50 Ohm (low volt) 1 Ohm (optimal) 50 Ohm (optimal)	xPPS & 10MHz xPPS	~ 3.3 V ~ 1.7 V ~ 3.3 V ~ 3.0 V	2.06 ns 1.55 ns 2.06 ns 1.67 ns
WR-ZEN TP-32BNCEXP32BNC V2.2	A & B C & D	1 Ohm (optimal) 50 Ohm (optimal) 1 Ohm (optimal) 50 Ohm (optimal)	xPPS & 10MHz xPPS	~ 3.3 V ~ 3.0 V ~ 3.3 V ~ 3.0 V	2.06 ns 1.67 ns 2.06 ns 1.67 ns
WR-ZEN TP TP-EXP V2.2 or higher	A B	1 Ohm (optimal) 50 Ohm (low volt) 1 Ohm (optimal) 50 Ohm (optimal)	xPPS & 10MHz xPPS & 10MHz	~ 3.3 V ~ 1.7 V ~ 3.3 V ~ 3.0 V	1.80 ns 1.30 ns 1.80 ns 1.47 ns

9.7 Low Jitter Setup

Users interested in low-jitter applications of thier WR-ZEN 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-ZEN 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-ZEN 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-ZEN 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-ZEN 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
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
V3.2-a	Improving LLDP and NTP section Adding new section (Timing outputs) for ZEN family	05-Apr-2022
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. 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

BLANK PAGE.

INDEX