

PLAYER ZERO

AUDIO PLAYER

Local and Streaming Audio Player



USER MANUAL

INDEX

HARDWARE	7-17
1 IMPORTANT REMARK.....	7
2 IMPORTANT SAFETY INSTRUCTIONS	7
3 IMPORTANT NOTE.....	9
4 INTRODUCTION	9
4.1 Main features:.....	9
5 INSTALLATION AND CONNECTIONS	10
5.1 Location, set up and ventilation	10
5.2 AC connection and powering on	10
5.3 Audio output connections	10
5.4 Ethernet port for configuration and Internet connection.....	11
5.5 Wi-Fi interface for configuration and Internet connection.....	11
5.6 GPI remote control ports.....	12
6 FRONT PANEL	13
7 STARTUP	14
7.1 Restore default / Firmware Update.....	14
8 PLAYER ZERO FUNCTION LIST	15
9 PLAYER ZERO FUNCTION DIAGRAM.....	15
10 TECHNICAL SPECIFICATIONS.....	16
SOFTWARE	18-144
* PLAYER ZERO Web GUI (v1.00r0)	18-107
11 INTRODUCION Web GUI (v1.00r0)	19
12 FIRST STEPS	19
12.1 Quick Guide for Ethernet Connection	22
12.2 Quick Guide for WiFi Connection.....	22
13 DEVICE.....	23
13.1 Player	23

13.2	Presets	26
13.3	Events	30
13.3.1	<i>GPI Events</i>	32
13.3.2	<i>SILENCE detection event</i>	34
13.4	Calendar	35
13.4.1	<i>General</i>	37
13.4.2	<i>Source</i>	38
13.4.3	<i>Date and time interval</i>	38
13.4.4	<i>Target</i>	40
13.4.5	<i>Regarding priorities</i>	41
13.4.6	<i>Practical example of configuration of calendar events</i>	42
13.5	Cloud Disk Sync	50
13.5.1	<i>Google Drive</i>	51
13.5.2	<i>Store and Forward (rsync)</i>	60
13.6	CMS Management	64
13.7	Scripts/LUA	65
13.7.1	<i>Example Script01</i> :.....	69
13.7.2	<i>Example Script02</i> :.....	70
13.7.3	<i>Example Script03</i> :.....	71
13.7.4	<i>Example Script04</i> :.....	72
14	NETWORK	73
14.1	Connection via RJ45 cable	73
14.2	Point-to-point wireless connection	78
14.3	Connecting to a WiFi network	81
15	SYSTEM	86
15.1	Name and Time	86
15.1.1	<i>General settings</i>	86
15.1.2	<i>Time synchronization</i>	87

15.2	Security	88
15.2.1	Web password.....	88
15.2.2	Front panel functions.....	89
15.2.3	RePlayer.....	90
15.3	Backup, Restore and Firmware	91
15.3.1	Make backups	92
15.3.2	Restore backups and factory settings.....	93
15.3.3	Firmware Update	93
15.4	USB/MMC Settings	94
15.5	Register	96
15.6	Reboot	97
16	Annex I: SSH server configuration for Store & Forward	98
16.1	Content groups	99
16.2	Installing SSH on Linux	101
16.3	Creating users on Linux	101
16.4	Generating SSH keys	103
16.5	Adding audio content to the SSH server	105
16.6	Configurind the remote source in the player	106
* THIRD-PARTY CONTROL PROTOCOL	108-115	
17	PLAYER ONE JSON Commands PRODUCT OVERVIEW	109
18	GENERAL CONSIDERATIONS	109
19	PLAYER COMMANDS	109
20	CONFIGURATION COMMANDS	111
21	PRESET COMMANDS	112
22	EVENT COMMANDS	113
23	CALENDAR COMMANDS	113
24	STORE AND FORWARD COMMANDS	113
25	GOOGLE DRIVE COMMANDS	113
26	CONTENT MANAGEMENT SYSTEM (CMS) COMMANDS	114
27	SCRIPTS COMMANDS	114

28	REGISTER COMMANDS.....	114
29	DEVICE COMMANDS.....	115
30	PANEL COMMANDS.....	115
*	LUA SDK	116-135
31	ePLAYER1 LUA SDK v1.02.....	116
31.1	<i>Changelist</i>	116
32	ePLAYER1 LUA SDK Introduction.....	116
33	How to use ePLAYER1 module?.....	117
33.1	<i>Conventions</i>	117
33.2	<i>PLAYER FUNCTIONS</i>	118
33.3	<i>CONFIG (CFG) FUNCTIONS</i>	121
33.4	<i>PRESET FUNCTIONS</i>	122
33.4.1	<i>Preset variables</i>	123
33.4.2	<i>Preset examples</i>	124
33.5	<i>EVENT FUNCTIONS</i>	124
33.5.1	<i>Event variables</i>	125
33.6	<i>CALENDAR (CAL) FUNCTIONS</i>	125
33.6.1	<i>Calendar variables</i>	126
33.7	<i>STORE AND FORWARD (SAF) FUNCTIONS</i>	127
33.7.1	<i>Store and Forward variables</i>	127
33.8	<i>GOOGLE DRIVE (GDRIVE) FUNCTIONS</i>	128
33.8.1	<i>Store and Forward variables</i>	128
33.9	<i>CONTENT MANAGEMENT SYSTEM (CMS) FUNCTIONS</i>	129
33.9.1	<i>CMS variables</i>	129
33.10	<i>SCRIPTS FUNCTIONS</i>	129
33.10.1	<i>Script variables</i>	131
33.11	<i>REGISTER (LOG) FUNCTIONS</i>	131
33.12	<i>DEVICE (DEV) FUNCTIONS</i>	132
33.13	<i>PANEL FUNCTIONS</i>	134

33.14	FINDER FUNCTIONS.....	135
33.15	LCD FUNCTIONS	135
* INTERNET RADIO URL STREAMS		136-144
34	HOW TO IDENTIFY INTERNET RADIO URL STREAMS	136
34.1	How to discover streaming URLs included in websites:	136
34.2	Playing internet radios via AirPlay (ePLAYER1):	141

1 IMPORTANT REMARK




WARNING: SHOCK HAZARD - DO NOT OPEN
AVIS: RISQUE DE CHOC ÉLECTRIQUE - NE PAS OUVRIR



The lightning flash with arrowhead symbol, within an equilateral triangle, is intended to alert the user to the presence of uninsulated “dangerous voltage” within the product’s enclosure that may be of sufficient magnitude to constitute a risk of electric shock to persons.



The exclamation point within an equilateral triangle is intended to alert the user to the presence of important operating and maintenance (servicing) instructions in the literature accompanying the appliance.

WARNING (If applicable): The terminals marked with symbol of “” may be of sufficient magnitude to constitute a risk of electric shock. The external wiring connected to the terminals requires installation by an instructed person or the use of ready-made leads or cords.

WARNING: To prevent fire or shock hazard, do not expose this equipment to rain or moisture.

WARNING: An apparatus with Class I construction shall be connected to a mains socket-outlet with a protective earthing connection.

2 IMPORTANT SAFETY INSTRUCTIONS

1. Read these instructions.
2. Keep these instructions.
3. Heed all warnings.
4. Follow all instructions.
5. Do not use this apparatus near water.
6. Clean only with dry cloth.
7. Do not block any ventilation openings. Install in accordance with the manufacturer’s instructions.
8. Do not install near any heat sources such as radiators, heat registers, stoves, or other apparatus (including amplifiers) that produce heat.

9. Do not defeat the safety purpose of the polarized or grounding type plug. A polarized plug has two blades with one wider than the other. A grounding type plug has two blades and a third grounding prong. The wide blade or the third prong are provided for your safety. If the provided plug does not fit into your outlet, consult an electrician for replacement of the obsolete outlet.
10. Protect the power cord from being walked on or pinched particularly at the plugs, convenience receptacles, and at the point where they exit from the apparatus.
11. Only use attachments/accessories specified by the manufacturer.
12. Unplug the apparatus during lightening sorts or when unused for long periods of time.
13. Refer all servicing to qualified personnel. Servicing is required when the apparatus has been damaged in any way, such as power supply cord or plug is damaged, liquid has been spilled or objects have fallen into the apparatus, the apparatus has been exposed to rain or moisture, does not operate normally, or has been dropped.
14. Disconnecting from mains: Switching off the POWER switch all the functions and light indicators of the amplifier will be stopped, but fully disconnecting the device from mains is done unplugging the power cord from the mains input socket. For this reason, it always shall remain readily operable.
15. Equipment is connected to a socket-outlet with earthing connection by means of a power cord.
16. The marking information is located at the bottom of apparatus.
17. The apparatus shall not be exposed to dripping or splashing and that no objects filled with liquids, such as vases, shall be placed on apparatus.

NOTE: *This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.*



WARNING: This product must not be discarded, under any circumstance, as unsorted urban waste. Take to the nearest electrical and electronic waste treatment centre.

NEEC AUDIO BARCELONA, S.L. accepts no liability for any damage that may be caused to people, animal or objects due to failure to comply with the warnings above.

3 IMPORTANT NOTE

Thank you for choosing our Ecler **PLAYER ZERO multimedia player!**

It is **VERY IMPORTANT** to carefully read this manual and to fully understand its contents before any connection in order to maximize your use and get the best performance from this equipment.

To ensure optimal operation of this device, we strongly recommend that its maintenance be carried out by our authorised Technical Services.

Ecler **PLAYER ZERO** comes with a **3-year warranty**.

4 INTRODUCTION

PLAYER ZERO is a compact stereo audio player able to play music content from local storage devices (USB/MICRO SD), Internet streaming services (online radios, etc.) and digital media exchange (DLNA, AirPlay) with the following main features:

4.1 Main features:

- 1 unbalanced stereo audio output, RCA connector and mini-jack (with stereo/mono selection).
- Compatible with mp3, ogg, WAV, AIFF and FLAC audio formats.
- One USB port and one MICRO SD card slot for access to local media content.
- Ethernet interface with RJ45 connector for communication with web setup application as well as Internet streaming reception.
- Wi-Fi interface (client or master mode) for communication with web application configuration as well as Internet streaming reception.
- Fully configurable through web application (point-to-point or through the same LAN)
- 2 GPI (General Purpose Inputs) ports, to trigger the 2 available events.
- Silence detection event.
- Internal clock with up to 240h autonomy (without AC adapter) and automatic synchronization with NTP services.
- **Modular firmware:** PLAYER ZERO has a firmware with modular services that allow each user to personalize its operation and adapt it to the size of his project or to his business model. The firmware includes, among other functions and services:
 - Calendar event triggering
 - Synchronization of cloud content (Cloud Disk Sync): compatible with Google Drive service

- Execution of scripts (user-written instruction files in lua language - www.lua.org).
- Local file encryption (USB/MICRO SD)
- Activity Log

Note: PLAYER ZERO programming is done through the web application embedded in the unit. Refer to the [PLAYER ONE & PLAYER ZERO web GUI application manual for](#) more information.

5 INSTALLATION AND CONNECTIONS

5.1 Location, set up and ventilation

PLAYER ZERO has been specially designed to be used both as a desktop player and a 19" rackmount unit, occupying third a rack space (with VEO-RACK19, optional mounting kit for standard rack cabinets).

In professional installations, it should preferably be placed in the same rack as the audio sources.

Since its power consumption is very low, forced ventilation is not necessary. However, avoid exposing the device to extreme temperature and the environment in which it is placed should be as dry and clean (no dust) as possible.

5.2 AC connection and powering on

PLAYER ZERO is powered by alternating current (AC) through its external power supply: 100-240 VAC, 50-60 Hz. This external power supply has several interchangeable connectors: American, European, British and Chinese.

The operating environment should be dry and be totally free of dust. Do not expose the unit to dripping or splashing. Do not place incandescent objects like candles on top of it.

If the unit requires any intervention and/or connection/disconnection, it must be powered off first. There are no user-serviceable parts within the unit. To avoid buzzing, do not allow the power cable to intertwine with audio shielded cables.

5.3 Audio output connections

PLAYER ZERO has 1 unbalanced stereo output on its rear panel.
The signal output connectors are 2 x RCA and 1 x stereo mini-jack.

5.4 Ethernet port for configuration and Internet connection

An RJ45 connector allows the unit to be connected to an Ethernet network or even directly (point-to-point) to a computer.

This connection provides access to Internet contents, and to the configuration of the unit using a web browser on the computer, which points to the IP address of the PLAYER ZERO, showing the web application embedded in the unit. Refer to the [PLAYER ONE & PLAYER ZERO web GUI application manual](#) for more information.

5.5 Wi-Fi interface for configuration and Internet connection

A Wi-Fi interface allows the unit to be connected to a Wi-Fi network or even directly (point-to-point) to a computer by Wi-Fi.

This connection provides access to Internet contents, and to the configuration of the unit using a web browser on the computer, which points to the IP address of the PLAYER ZERO, showing the web application embedded in the unit. Refer to the [PLAYER ONE & PLAYER ZERO web GUI application manual](#) for more information.

5.6 GPI remote control ports

PLAYER ZERO has 2 GPI control inputs on its rear panel. These inputs can be connected to an external physical device (e.g. a contact closure) and associated with an PLAYER ZERO function:

- Load and play previously configured audio content
- Preset recall
- Audio playback with priority over the music program
- Transport bar control (PLAY/PAUSE, STOP, etc.)
- Internal triggering, to interact with other player services (scripts for example)

GPI connectors use 3-position screw terminal blocks (Euroblocks). The wiring is as follows:

GPI terminal	>	- 1, 2 terminals
Ground	>	⊥ terminal

Connecting cables can be up to 500 meters long with a minimum section of 0,5 mm².

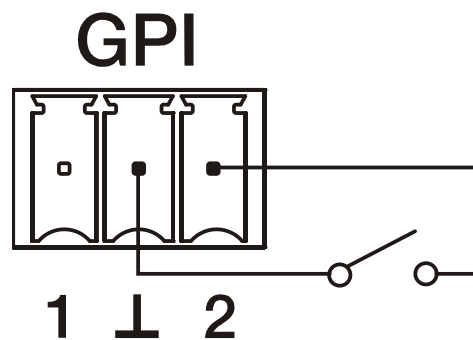


Figure 1: Connection example of GPI 2

6 FRONT PANEL

PLAYER ZERO front panel provides the following elements:

- **MICRO SD/MICRO SDHC slot:** for local audio content playback. Up to 2TB. FAT16/32 format.
- **USB 2.0 port:** for local audio content playback. Up to 2TB. FAT16/32 and NTFS format.
- **SP LED:** *signal presence* indicator.
- **NET LED:** network (Internet) data reception indicator.



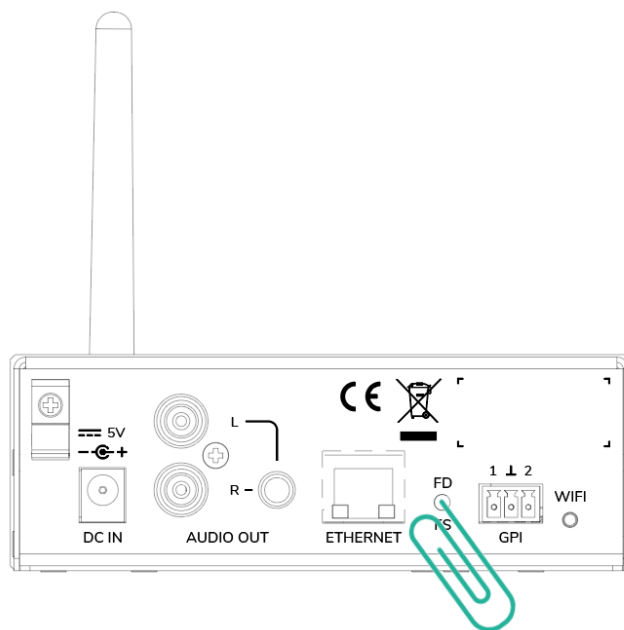
7 STARTUP

Ecler recommends configuring PLAYER ZERO with your web application to enjoy its full functionality. Please refer to the [PLAYER ONE & PLAYER ZERO web GUI application manual](#) to know all the functionality it offers.

Plug the power cord into the back of the device to turn it on.

7.1 Restore default / Firmware Update

PLAYER ZERO has two functionalities that can be done via the FD/FS button on the rear panel:



- **Restore Factory Defaults:** Restore factory settings. Press and hold the FD/FS button on the rear panel for 10 seconds with a small pin or paper clip while the unit is powered on.
- **Fail Safe (Restore firmware):** Install the latest firmware published on the Ecler website or a firmware file hosted on one of the local storage systems (USB/MICRO SD).

While the unit is unplugged, press and hold the FD/FS button with a small pin or paper clip and then plug in the unit. The front LEDs will start flashing rapidly for 3 seconds (at this point the FD/FS button can be released).

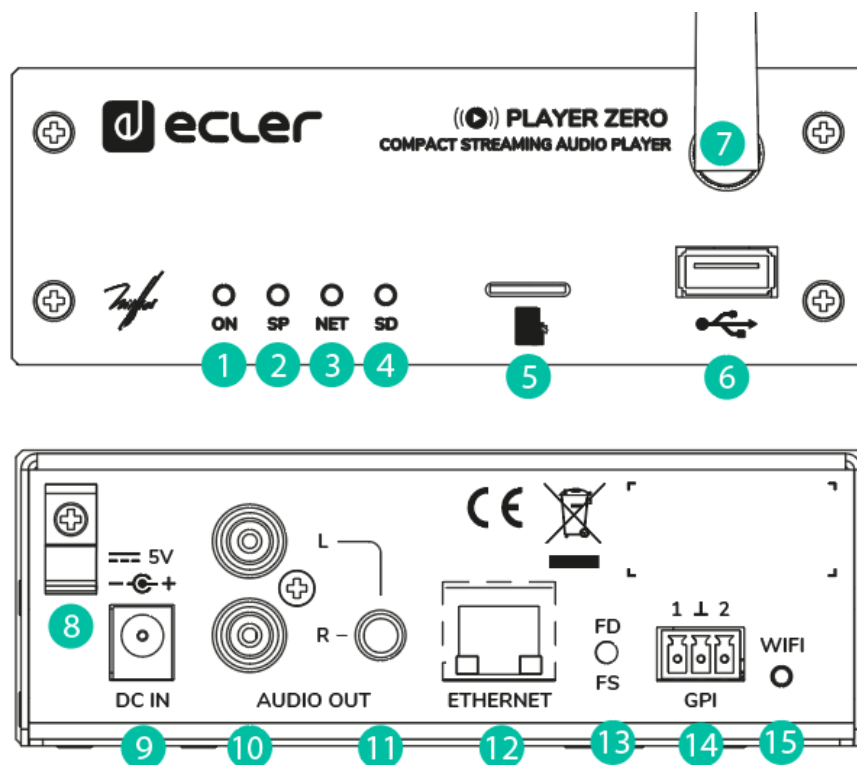
Notes:

- **To restore the latest released firmware, it is necessary to connect the device to a DHCP server with Internet access, to download the firmware.**
- **If this action is not performed correctly, you may lose all device settings and parameters. Be sure to save a backup copy of the device settings before executing this action.**

8 PLAYER ZERO FUNCTION LIST

1. ON LED power indicator
2. SP LED signal present indicator
3. NET LED network traffic indicator
4. SD LED Micro SD source operation indicator
5. MICRO SD slot
6. USB port
7. WiFi Antenna
8. Security flange for power supply cable
9. External power supply connector
10. Stereo output 2 x RCA
11. Stereo mini-jack output
12. RJ-45 connector
13. Factory Default / Fail Safe button
14. GPI port
15. Indicator light, WIFI

9 PLAYER ZERO FUNCTION DIAGRAM



10 TECHNICAL SPECIFICATIONS

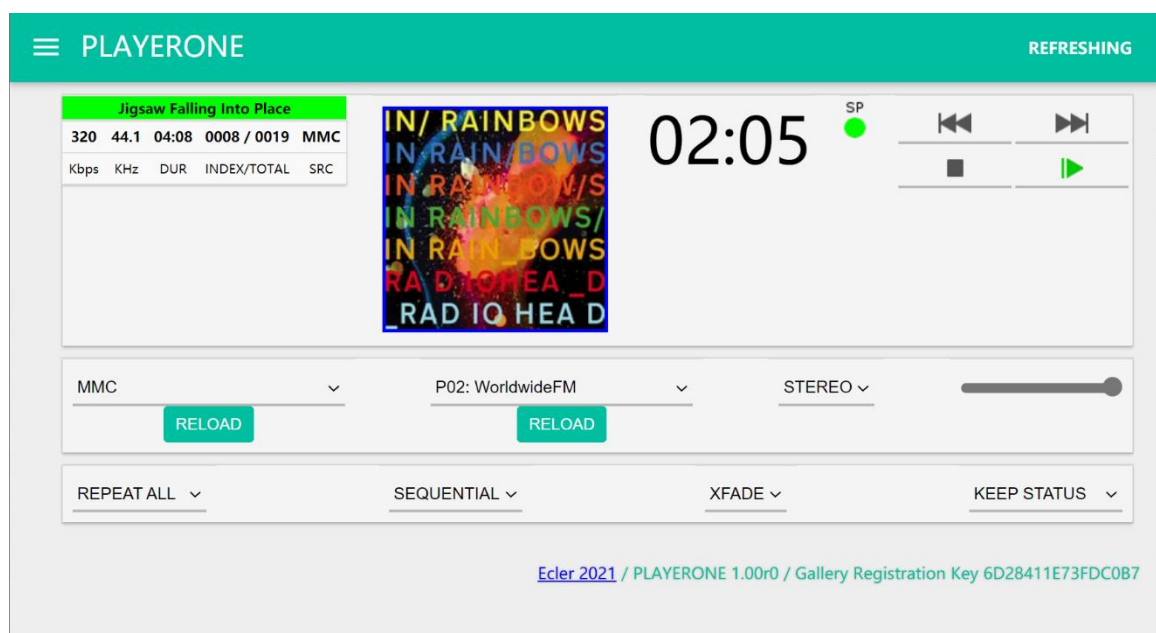
DIGITAL ENGINE		
Processor	MIPS Single Core 64bits 580MHz	
AUDIO CONVERTERS		
Sampling rate	48kHz	
Resolution	16bit	
Bit rate	32~320kbps	
Frequency response	5Hz - 24kHz (-3dB)	
MEDIA PLAYER		
Audio sources	Local storage (USB & microSD), Internet radio, AirPlay, DLNA	
Dynamic range	From -80 dB to 0 dB	
THD + Noise	< 0.008% (1kHz, 1Vrms)	
ANALOGUE AUDIO OUTPUTS		
Number of outputs	2 x Stereo output (unbalanced)	
Connection type	RCA estéreo, minijack 3,5mm	
Max output level	6dBV / 5k ohm	
Output impedance	460 ohm	
Dynamic range	TBC	
Crosstalk	TBC	
CONNECTIVITY		
Ethernet	RJ45 10/100Mbps	
Wi-Fi	2.4GHz Wi-Fi, 802.11 b/g/n	
Wi-Fi antenna	Front panel	
Programing and control	Web Application, RePLayer & Gallery. Third party integration: JSON	
REMOTE CONTROL CONNECTIONS		
GPIs	2 ports, Dry contact to ground, 3 pin Terminal block	
REAL-TIME CLOCK		
Retention time	240 hours aprox.	
Accuracy	±1 minute / month	
MONITORING AND CONTROL		
Display	No	
LED indicators	Front panel: NET, SP, SD, Power Rear panel: Wifi	
Buttons	Rear panel: Factory Defaults / Fail Safe (pin-hole)	
LOCAL STORAGE		
Micro SD	Front panel micro SD slot Supports micro SD SDXC	
USB	Front panel USB female connector Support USB 2.0 High Speed (480 Mbps)	
Capacity	Up to 2TB	
File system	Supports FAT16, FAT 32, VFAT and NTFS (read-only) Multi-partition up to 1	
Playable audio files	mp3, ogg, WAV, FLAC, AIFF, m3u, pls	

Files analysis	65354 playable folders 65354 playable folders within each folder 65354 playable files within each folder
Folder hierarchy	Up to 8 containing the root directory
Sorting	UNICODE Up to 100 folders, 100 files by folder (Folders/files over 100 sorted in the FAT order)
ELECTRICAL	
Power supply	External, universal, 5VDC (included)
AC mains requirement	90-264VCA 50-60Hz
AC mains connector	International AC plugs set
Power consumption	4.5 VA / 2.2 W
Ventilation	Passive
PHYSICAL	
Operating temperature	Min 0°C, +32°F Max +35°C, +95°F
Operating humidity	< 85% HR
Storage temperature	Min -10°C, +14°F Max +50°C, +122°F
Storage humidity	< 90% HR
Installation options	Desktop, rack (1/3 of 1UR)
Included accessories	AD/DC Power Supply, WiFi antenna, adhesive rubber feet, rack tray screws , GPI connector, warranty card and standard user guide
Optional accessories	VEO-RACK19 Compatible
Dimensions	126.5 x 44 x 132 mm / 4.98" x 1.73" x 5.19"
Weight	0.6 Kg / 1.32 lb
Shipping dimensions	450 x 220 x 60 mm / 17.71" x 8.66" x 2.36"
Shipping weight	0.8 Kg / 1.76 lb
CERTIFICATIONS	
CB Report	Pending approval
EMC	Pending approval
FCC	Pending approval

PLAYER ONE / PLAYER ZERO – Web GUI (v1.00r0)

AUDIO PLAYERS

Streaming and local media audio player



USER MANUAL

11 INTRODUCCION Web GUI (v1.00r0)

PLAYER ONE / PLAYER ZERO has a web application, embedded in the device itself, that allows you to configure it. It is not necessary to install any additional software. With this application, you can configure the device's advanced options, create playlists, schedule calendar events, create scripts or control basic functions remotely. The application can be accessed in a web browser from any device that is connected to the same local network, via Ethernet (wired) or WiFi.

12 FIRST STEPS

In order for the device to access the PLAYER ONE / PLAYER ZERO web application, it must be connected to the network, either via a cable (RJ-45 connector) or wirelessly (WiFi).

- **Wired (Ethernet connection):** By default, PLAYER ONE / PLAYER ZERO is set to DHCP mode. It will be assigned an IP address automatically.
 - Make sure that the static IP network parameters are compatible with your local network and the existing IP range of the system.

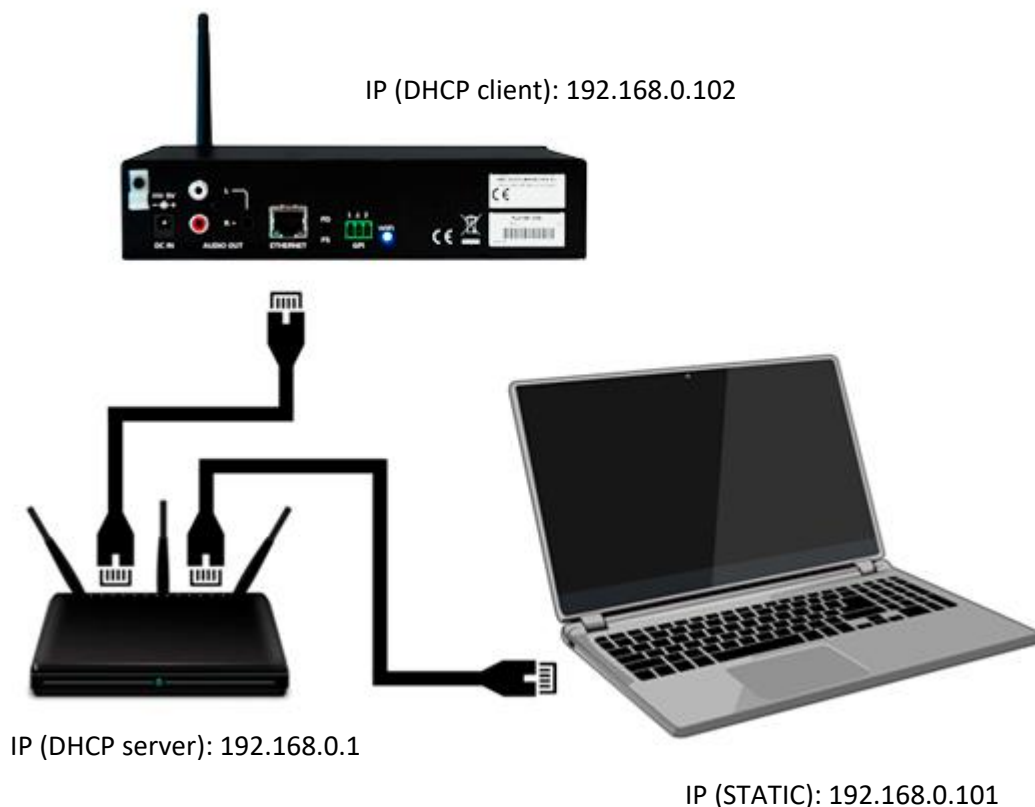


Figure 2: example of connection to a local network via the Ethernet network interface (wired)

- **WiFi:** PLAYER ONE / PLAYER ZERO has a WiFi network interface, which means that mobile devices can stream audio to the device. It can be configured wirelessly. There two modes of operation:
- **MASTER mode:** point-to-point connection. The device's WiFi network interface is set to this mode by default. Connect your WiFi device (computer, smartphone, etc.) as a client of the device via your WiFi network wizard (connect to the PLAYER-WIFI network, default SSID). The default password is: 0123456789.

Please Note: in this operating mode, no Internet connection is available. However, it will be useful when opening the web application for the first time and configuring the network parameters according to your needs.

- **CLIENT mode:** this connection mode allows the device to connect to your preferred WiFi network. Mobile devices must then be connected to the same network in order to be able to configure PLAYER ONE / PLAYER ZERO. If your WiFi network has an Internet connection, both PLAYER ONE / PLAYER ZERO and the mobile devices will have Internet access.

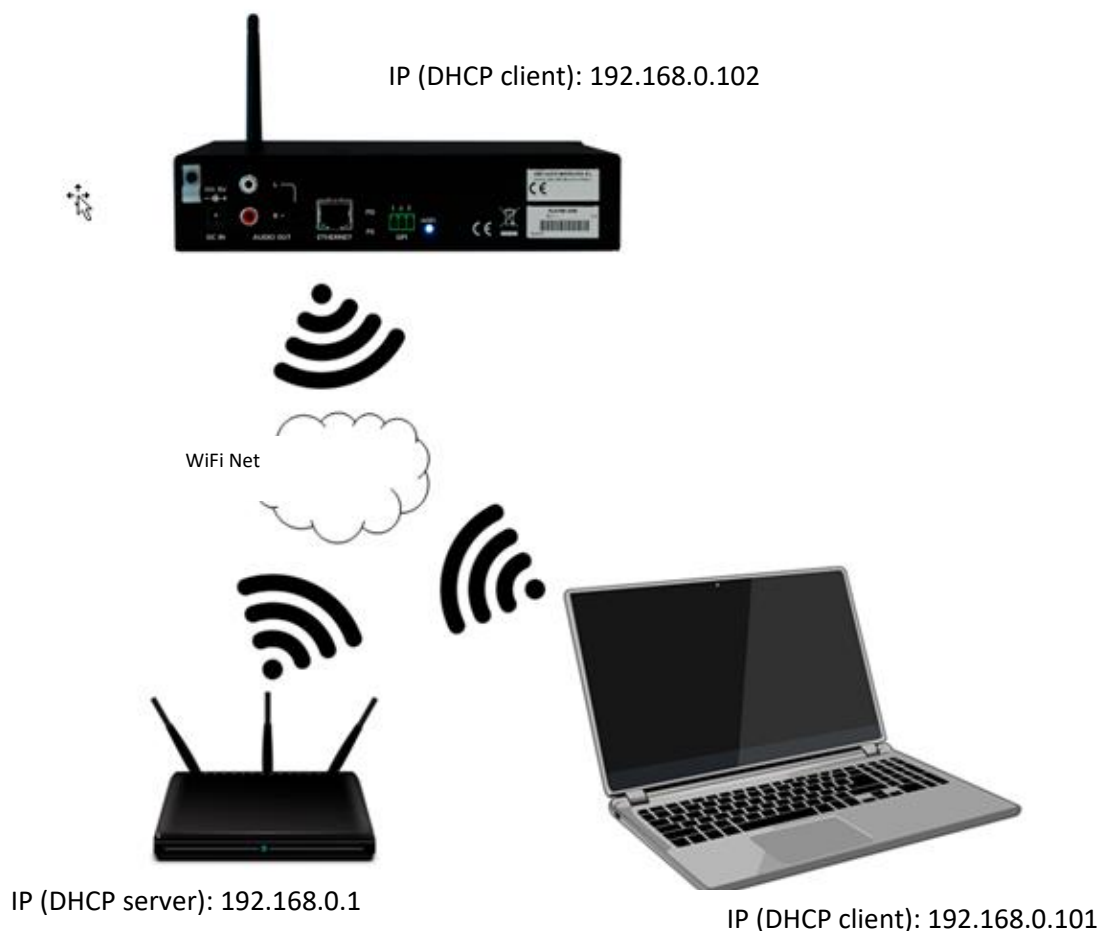


Figure 3: example of connection to a local network via WiFi network interface (wireless)

PLAYER ONE / PLAYER ZERO devices use the mDNS service so that they can be accessed intuitively through a web browser on the same local area network (LAN). To access the web application easily, enter “devicename.local/” into the search bar of your browser. By default this is, "**playerone.local/**" for PLAYER ONE, "**playerzero.local/**" for PLAYER ZERO.

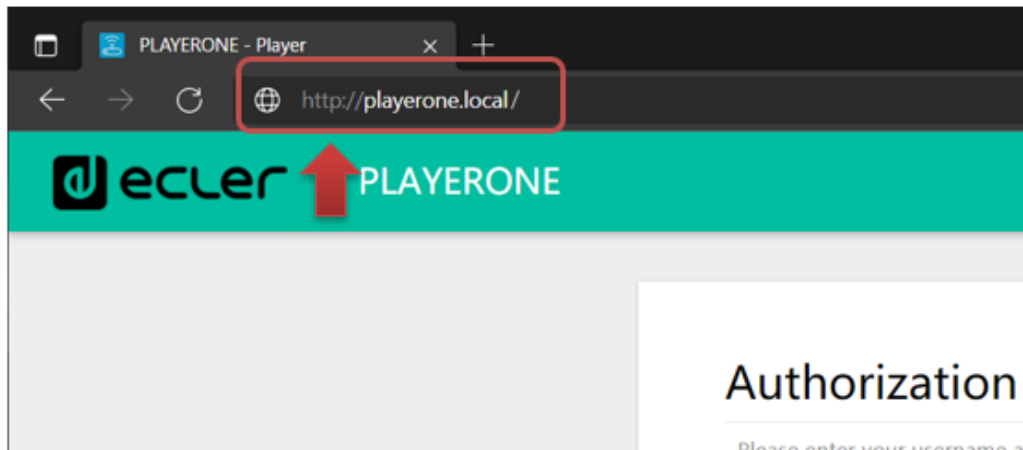


Figure 4: access via mDNS service

If you prefer (or if the mDNS service is not available), you can also access the device using the IP assigned to the PLAYER ONE / PLAYER ZERO:

- **PLAYER ONE:**

- Press and hold the MENU button for a few seconds until the configuration menu appears.
- Navigate by pressing the NEXT button, until the display shows WIFI or ETHERNET, depending on the chosen connection.
- Press the rotary encoder (SELECT)

- **PLAYER ZERO:**

As there is no screen to show you the IP, you need to find it on the web GUI of the DHCP server (Switch/Router). Refer to the manufacturer's documentation.





Icon	Device Name	IP Address	Protocol	MAC Address	Icon
	Player One	192.168.1.9	DHCP	0C:CF:89:21:E6:FD	
	Player Zero	192.168.1.146	DHCP	30:EB:1F:2B:CB:E0	

Figure 5: web GUI Router example

Type the IP address of the device into the navigation bar of your browser (the IP shown does not have to match the one assigned to your device).

You will see the welcome screen. To access the application, the *username* and *password* are as follows (preset):

- **Username:** root
- **Password:** ecler

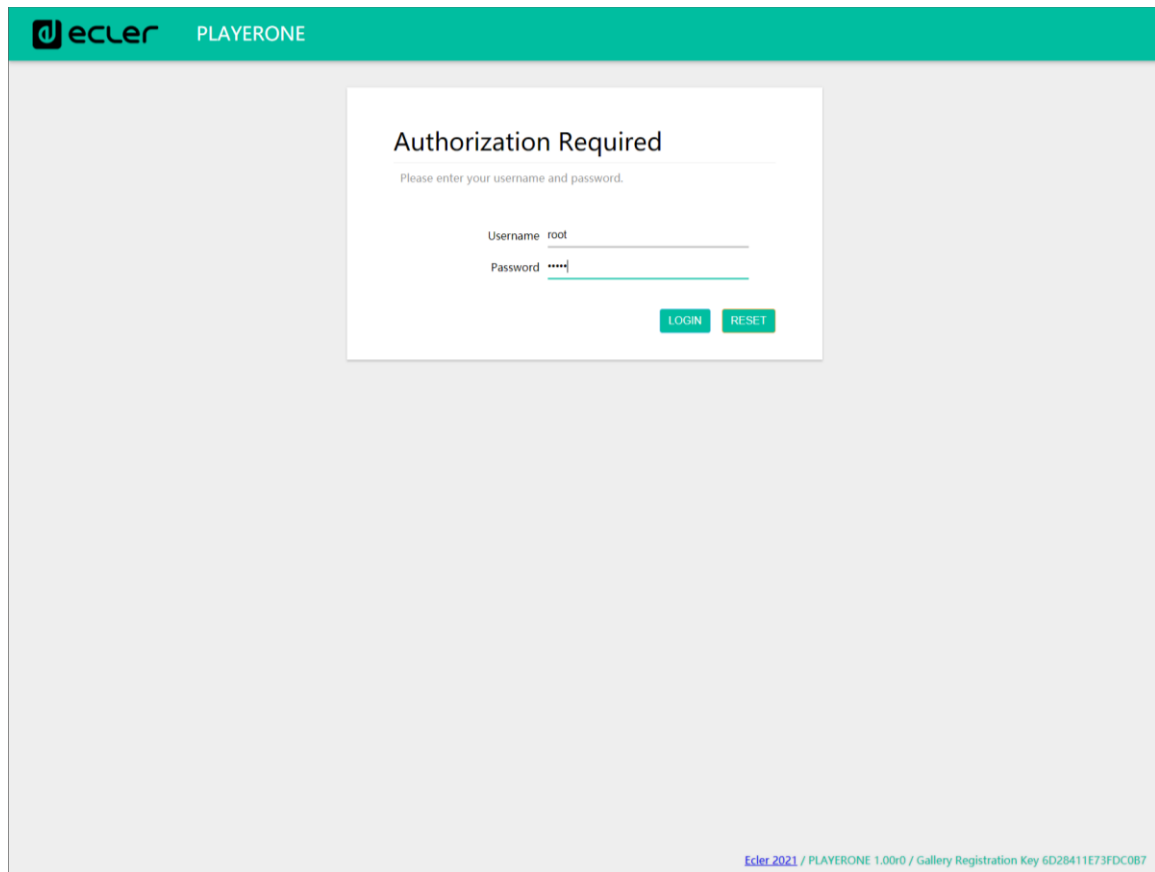


Figure 6: web application welcome page

12.1 Quick Guide for Ethernet Connection

- Connect the PLAYER ONE / PLAYER ZERO to a switch/router via the Ethernet interface (wired).
- Connect the computer/smart device to the same network.
- Enter "playerone.local/" for PLAYER ONE, or "playerzero.local/" for PLAYER ZERO, into your browser.

12.2 Quick Guide for WiFi Connection

- Connect the computer/smart device to the PLAYER-WIFI network. Password: 0123456789
- Enter "playerone.local/" for PLAYER ONE, or "playerzero.local/" for PLAYER ZERO, into your browser.

13 DEVICE

13.1 Player

This page in the PLAYER ONE / PLAYER ZERO menu displays information about playback, *streaming tags* and detailed information about the audio content, and in addition, it shows the album art. It also allows basic functions to be controlled remotely, such as PLAY/PAUSE, STOP, PREV and NEXT; source selection and user presets; repeat, play, and fade modes; channels (stereo/mono), volume control; and reset mode. There is useful information at the bottom of the page as well, such as the firmware version and the [Ecler Gallery](#) registration key.

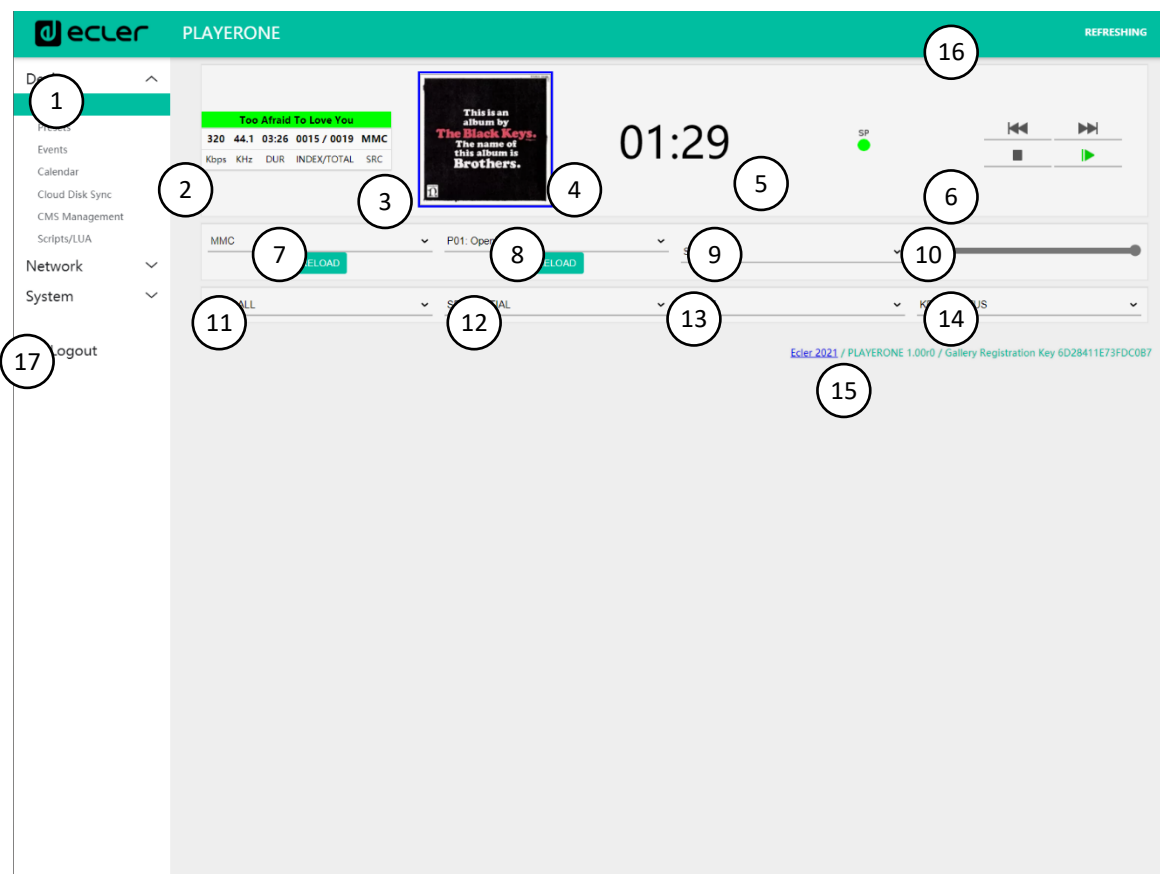
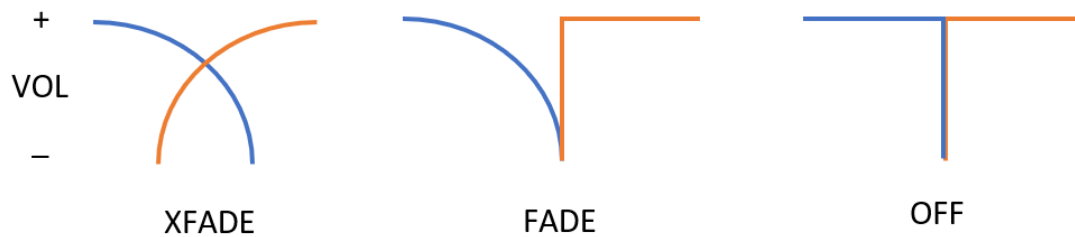


Figure 7: player page (Player)

1. **Navigation menu:** displays the web application's different navigation menus and submenus.
2. **Streaming information:** displays information about the stream or the audio file (depending on the way it is configured). If this information is not available, the default value is displayed, which is the URL.
 - o ID3 Tags: title, artist, album, etc.

- Kbps: bit rate/s
 - KHz: sampling frequency
 - DUR: duration
 - INDEX/TOTAL: index or position within all the files
 - SRC: source (USB, MMC, NET, etc.)
- 3. Album art:** displays the album art for the file. For the album art to be correctly displayed, the device needs to have an Internet connection. If the album art cannot be displayed, it will show a default image.
 - 4. Playback time:** time elapsed since the start of playback of the URL or audio file.
 - 5. Signal presence:** this will be green if the device has an audio signal at its output. If the device is not playing any audio, or if the volume level is too low, or the device is muted, it will be grey. This virtual LED will be the same colour as the SP LED on the front of the device. It is useful for dealing with issues when there is no audio signal.
 - 6. Playback controls:** allow the device's playback controls to be operated remotely: PREV, NEXT, STOP, PLAY/PAUSE. **Please Note:** PLAYER ZERO has no playback controls on the front panel.
 - 7. Sources:** allows one of the available sources to be selected. The RELOAD button allows the current source to be reloaded.
 - 8. Presets:** allows one of the available presets to be selected. The RELOAD button allows the current preset to be reloaded. If changes are made to the current preset, a reload is required in order for the changes to be applied.
 - 9. Stereo-mono:** selection of stereo output (left and right channels) or mono output (same signal on left and right channel).
 - 10. Volume:** allows the volume to be controlled remotely. **Please Note:** PLAYER ZERO does not have a volume control on the front panel.
 - 11. Repeat mode:**
 - PLAY ALL: Plays through the entire playlist once.
 - PLAY ONE: Only plays the first item in the playlist.
 - REPEAT ALL: Repeats the entire playlist on a loop.
 - REPEAT ONE: Only repeats the first item in the playlist.
 - 12. Play mode:**
 - SEQUENTIAL: Plays through the entire playlist in alphanumeric order.
 - RANDOM: plays through the entire playlist in pseudo-random order.
 - 13. Transition mode,** for transitions between audio files:
 - XFADE: The file that is playing fades out at the end of playback, as the next file increases in volume. There is a smooth transition between files (approx. 5 sec.). There is some overlap between tracks.
 - FADE: The file that is playing fades out at the end of playback (approx. 2.5 sec.). There is a smooth transition between files. There is no overlap between tracks.

- OFF: deactivated. There is an abrupt transition between files. There is no attenuation or overlap between audio files.



PLEASE NOTE: if the playback file is short (2-3 sec. tone, for example) and XFADE transition mode is used together with REPEAT ONE/ALL repeat mode, special attention must be paid to the playback times of the files, and the transitions. This may cause unexpected behaviour.

14. Reset mode:

- KEEP STATUS: when the device is reset, the playback status will be maintained: source, preset, playback (PLAY, STOP, etc.) repeat mode, etc.
- LOAD PRESET 1: when the device is reset, it automatically loads preset 1.

15. Information: this displays the following relevant information:

- Year of development
- Device firmware version
- Ecler Gallery platform registration key

16. Screen refresh: allows the screen refresh to be paused (SP, playback time, file information, etc.).

It also shows the number of changes made to a configuration before it is saved.

17. Logout: this logs you out of the web application. You will be redirected to the welcome screen.

13.2 Presets

PLAYER ONE/ PLAYER ZERO allows up to 20 presets or user configuration memories to be created. If you recall a *preset* stored in the device at a later date, it will recall all the settings that are stored in that preset.

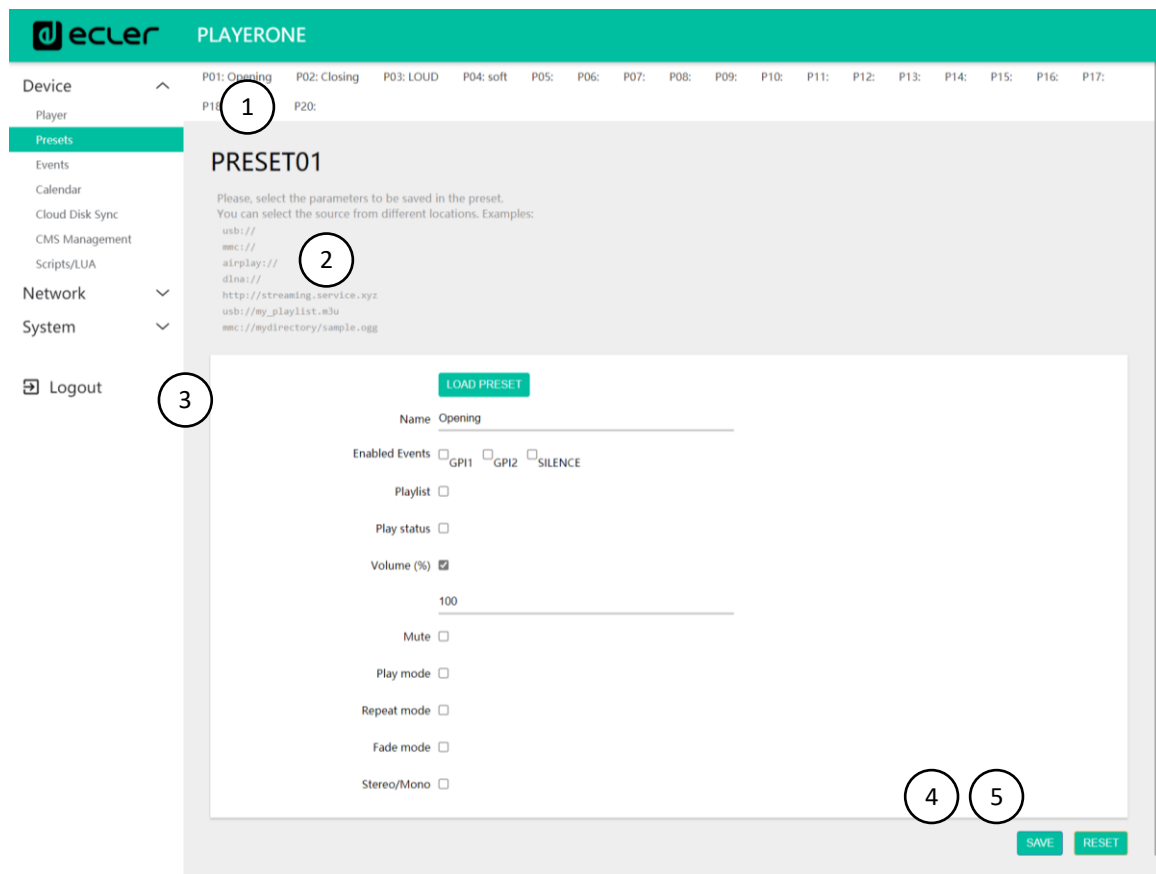
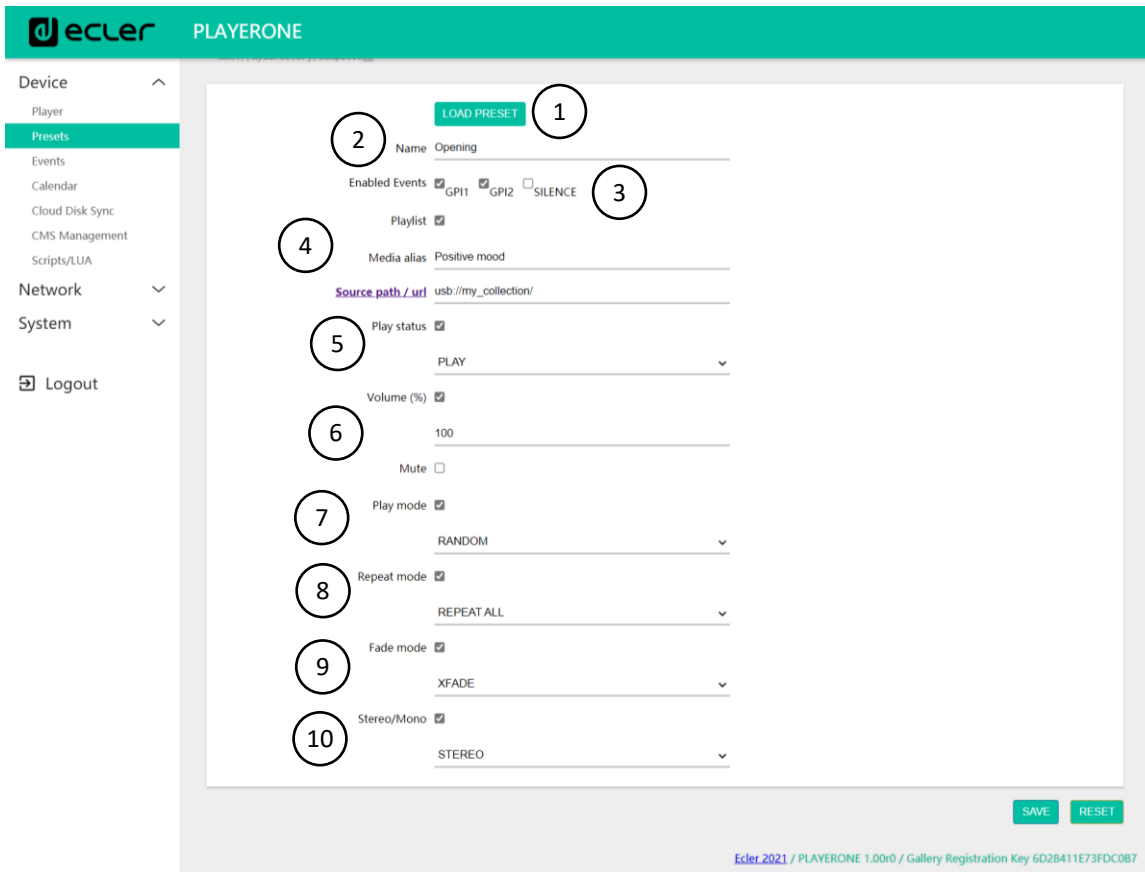


Figure 8: Preset configuration example

1. **Page header:** this will show the 20 *presets*. Their default names are: P01, P02...P20. When you click on the name, it will display the configuration for that *preset*. The names that appear for the *presets* can be edited within the preset configuration. After this setting is saved in the preset, you need to refresh the browser page (F5) in order to display the changes in the header.
2. **Information:** displays useful information for configuring the presets.
3. **Configuration options** for the selected preset.
4. **Save button:** saves the settings made to the *preset* that is being edited.
5. **Reset button:** resets the changes made to the last configuration that was saved in the *preset* that is being edited.



1. **Load Preset button:** loads the selected preset. This is useful as you can load the preset right after editing it, without having to change pages or handle the device.
2. **Name:** name of the preset. This is the name that will be displayed on the PLAYER ONE main menu/PRESETS (LCD screen), in the preset list on the Player page, in the Presets page header, and in the RePlayer application.

Please Note: PLAYER ZERO does not have an LCD screen.

3. **Enabled Events:** enables/disables events triggered by GPIs (General Purpose Input) and the silence detection event in the preset. The GPIs and the silence detection event must be configured on the Events page. See chapter [Events](#) for more information.

Please Note: for the GPI event to work correctly it must be configured and enabled in the preset, and the preset must be loaded. If a preset is loaded when the GPIs are disabled, the GPIs will not work.

4. **Playlist:** if this option is enabled, when the preset is recalled, the playlist that is currently playing will be replaced by the source that is entered in the Source path/url field.
 - o Media alias: alias of the source that is hosted in the preset (Source path/url). This alias will be displayed on the PLAYER ONE main menu/SOURCE (LCD screen) as an available source, in addition to the default sources (USB, SD, etc.). It will also allow you to directly access that media from any preset, from the Player page or the RePlayer application.

- **Source path/url:** this stores a network or local address in the preset. This address must be a valid address for the correct playback of audio content on the device. We recommend reading the document "how to identify audio stream URLs on the Internet". You can refer to the application prompts for entering local addresses (USB, SD, AirPlay, etc.). You can click on "Source path/url" (blue) to open the address that is entered in this field in a new browser tab. This option is available on several pages within the application. It is useful for checking that an audio source (Internet radio, for example) is functioning correctly, or for copying the address to create playlists (.m3u file, for example). See technical information (datasheet) for audio and playlist formats that are supported by the player.

Please Note: Internet radios saved via the front panel of PLAYER ONE are stored in the Playlist field.

- 5. Play status:** if this option is enabled, the status of the player will be overwritten when loading the preset.
- 6. Volume (%) /MUTE:** if this option is enabled, the volume/MUTE of the player will be overwritten when loading the preset (in %).
- 7. Play mode:** if this option is enabled, the play mode (sequential/random) will be overwritten.
- 8. Repeat mode:** if this option is enabled, the repeat mode (play all, play one, repeat all or repeat one) will be overwritten.
- 9. Fade mode:** if this option is enabled, the transition mode between tracks within a playlist (off/fade/cross-fade) will be overwritten.
- 10. Stereo/Mono:** if this option is enabled, the mono or stereo output selection will be overwritten.

13.2.1.1 Examples of audio sources

Please Note: the addresses shown are **examples**, these Internet radio or local file addresses may not work on your player.

Media Path	Media Location	Items included in the play queue (just valid audio media)
usb://	USB storage device, root folder	Media stored in the USB root folder and up to the third level of sub-folders in it
mmc://	SD card storage device, root folder	Media stored in the SD card root folder and up to the third level of sub-folders in it
usb://musicfolder/jazz/	USB storage device, \musicfolder\jazz folder	Media stored in the USB device \musicfolder\jazz folder and up to the third level of sub-folders in it
mmc://musicfolder/jazz/	SD card storage device, \musicfolder\jazz folder	Media stored in the SD card \musicfolder\jazz folder and up to the third level of sub-folders in it
mmc://evacuation_message.mp3	SD storage device, root folder	mp3 single file named evacuation_message.mp3
usb://evacuation_message.mp3	USB storage device, root folder	mp3 single file named evacuation_message.mp3
usb://...path.../my_collection.m3u mmc://...path.../my_collection.m3u	Defined by the m3u playlist file	Media pointed by my_collection.m3u playlist ...path ... is the folder path where the m3u file is located
usb://...path.../my_songs.m3u8 mmc://...path.../my_songs.m3u8	Defined by the m3u8 playlist file	Media pointed by my_songs.m3u8 playlist ...path ... is the folder path where the m3u8 file is located
usb://...path.../best_of_rock.pls mmc://...path.../best_of_rock.pls	Defined by the pls playlist file	Media pointed by best_of_rock.pls playlist ...path... is the folder path where the pls file is located

http://www.ecler.com/music/party_songs/party.m3u	Defined by the m3u playlist file	Media pointed by party.m3u playlist, and located in a Internet url (corporate web, etc.)
http://www.ecler.com/music/party_songs/party.pls	Defined by the pls playlist file	Media pointed by party.pls playlist, and located in a Internet url (corporate web, etc.)
http://65.60.19.42:8040/	Real-time streaming	Media served by an Internet audio streaming service with url http://nnn.nnn.nnn.nnn:port (in the example http://65.60.19.42:8040/)
http://stream1.megarockradio.net:8240	Real-time streaming	Media served by an Internet audio streaming service with url http://url:port
http://208.92.53.90:443/LOS40_SC	Real-time streaming	Media served by an Internet audio streaming service with url http://url/folder
http://media.musicradio.com/channel01.m3u	Real-time streaming	Media served by an Internet audio streaming service, a playlist with url http://url/playlist.m3u

13.3 Events

3 events are available: 2 of them are triggered by the GPI ports (by means of external, potential-free contact closures, which are connected to the GPI ports on the rear panel of the unit) and the other is triggered by silence detection. Both types of events can be configured on the Events page. By selecting the tab that corresponds to the event, you can access its configuration.

Please Note: bear in mind that, for the events to work correctly, they must be enabled in the *preset* currently in operation.

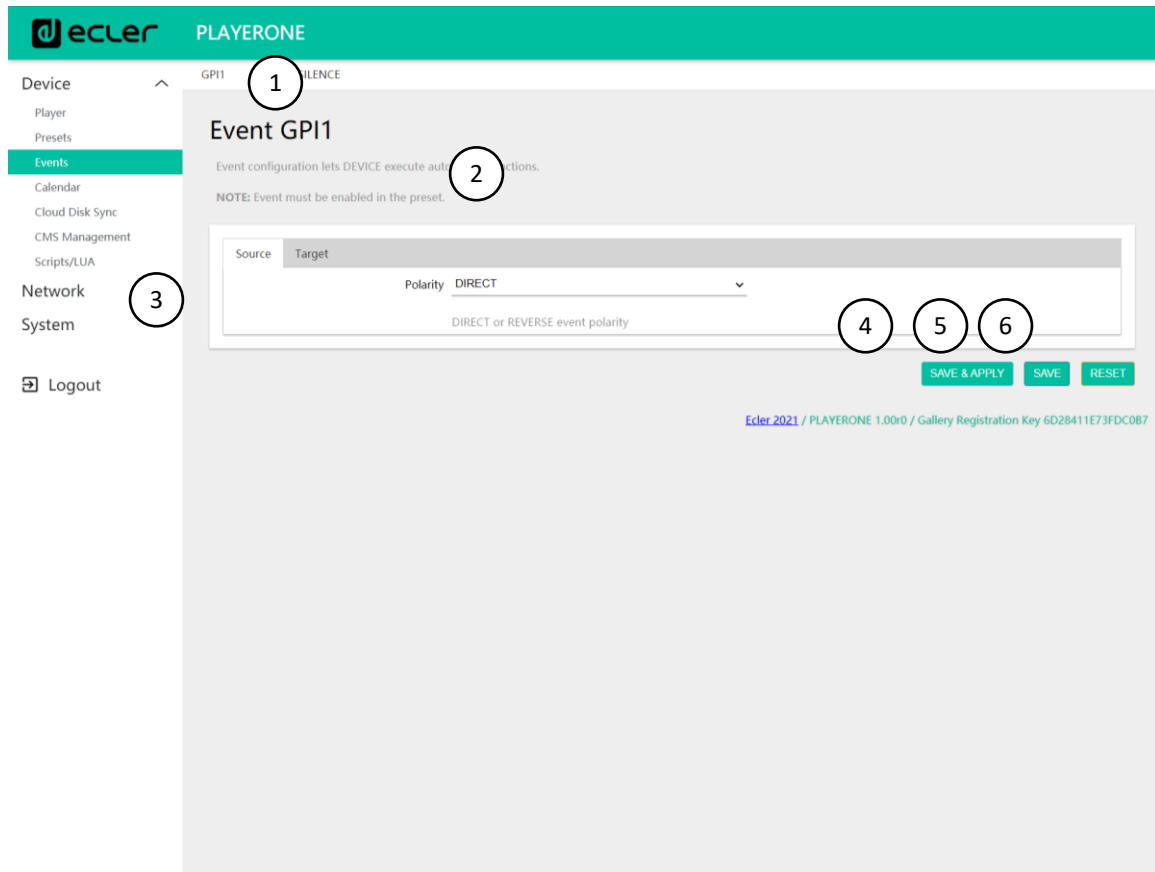


Figure 9: GPI event configuration example

1. **Page header:** displays GPI and silence events. When you click on the name, it will display the configuration for that event.
2. **Information:** displays useful information for configuring the events.
3. **Configuration options** for the selected event.
4. **Save & Apply button:** saves the changes made to the GPI that is being edited and applies them. If the *preset* that is currently in operation has such an event enabled, it is not necessary to reload the *preset*.
5. **Save button:** saves the changes made to the GPI that is being edited without applying them. In this way, if the *preset* that is currently in operation has such an event enabled, the changes will only be applied when the *preset* is reloaded.
6. **Reset button:** resets the changes made to the last saved configuration in the event that is being edited.

Please Note: you can configure the different tabs (Source, Target) before saving, and the changes are retained.

13.3.1 GPI Events

There are 2 GPI events: GPI1 and GPI2. They can each be configured to trigger in different ways and perform independent actions.

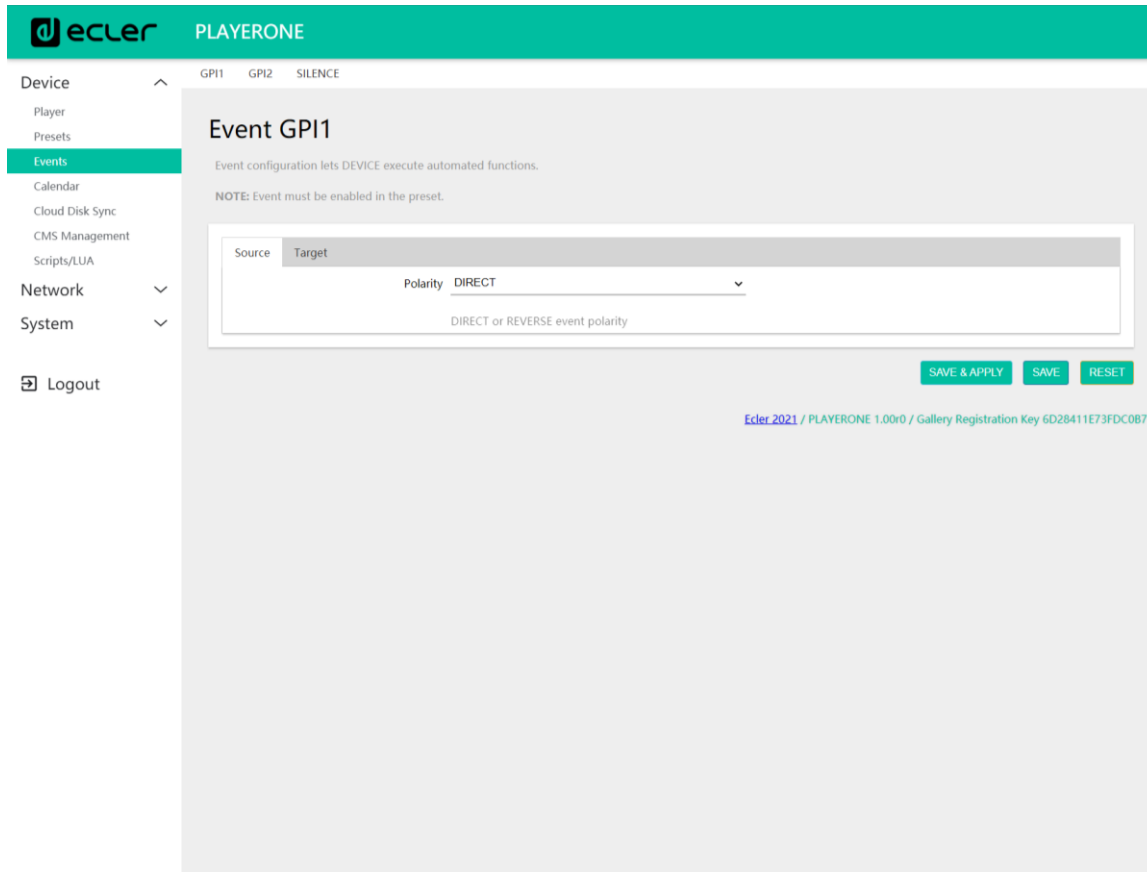


Figure 10: GPI configuration, Source

- **Source:** source
 - **Polarity:** polarity, direct or inverse; to define the trigger, by contact closure or contact release.

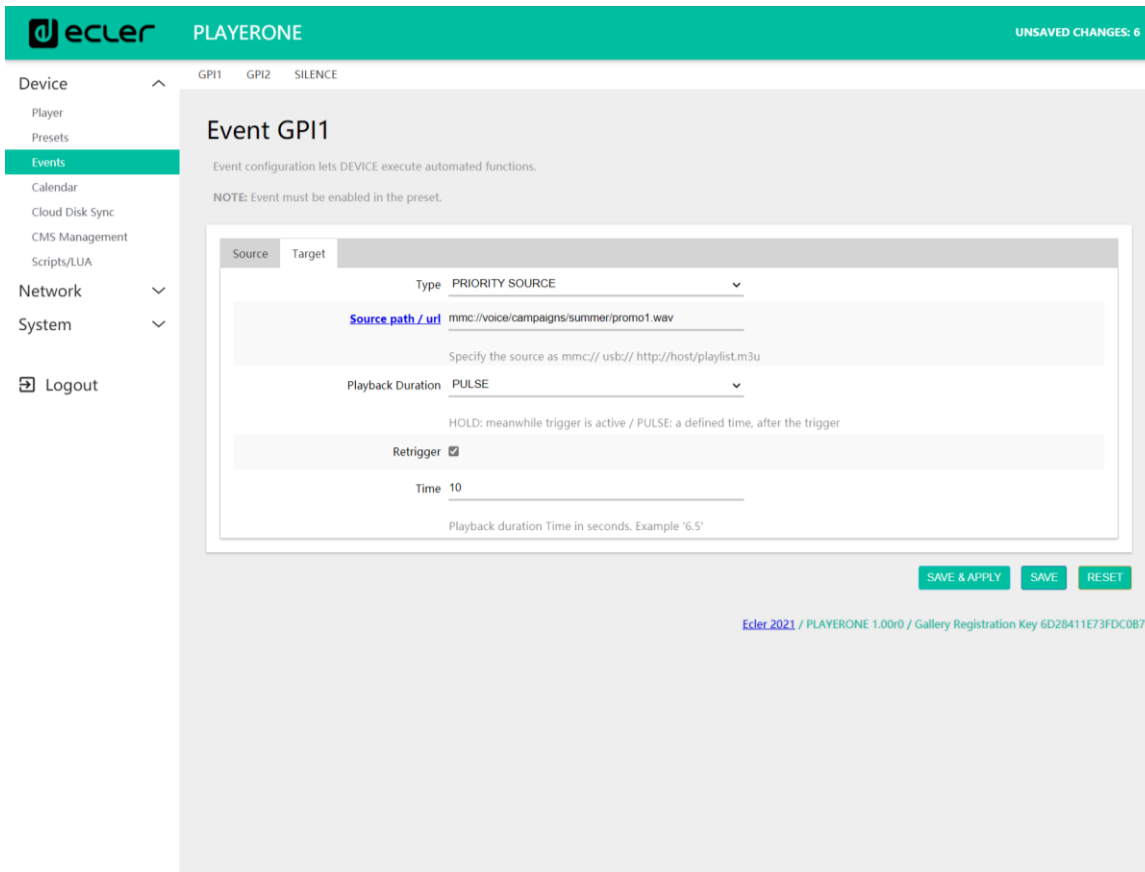


Figure 11: GPI configuration, Target

- **Target tab:** an action that the device will perform after GPI activation. There are several options or types of event (Type):
 - **Internal:** internal stimulus. This is useful for triggering actions within a Script.
 - **Preset recall:** loading a preset. Select the preset you want to load with the GPI activation.
 - **Transport control:** control of current playback, play/pause, play, stop, next, prev.
 - **Load & Play Source:** loading and playback of a source. The source must be specified in the "Source path/url" field.
 - **Priority Source:** playback of a source with priority over programmed audio. The source specified in the "Source path/url" field attenuates the source that is playing (programmed audio). When the priority audio ends, the programmed audio resumes playing, gradually returning to its previous volume.
 - If the **HOLD option** is selected, the source specified in the "Source path/url" field will be given priority WHILE the stimulus is held (GPI direct/reverse, as specified in the "Source" tab).
 - If the **PULSE option** is selected, the source specified in the "Source path/url" field will be prioritized for the time (in seconds) specified in the Time field. The Retrigger option allows the priority event to be re-triggered without waiting for it to finish, and restarting the timer.

Please Note: a priority event can be useful for playing announcements, pre-recorded announcements, emergency announcements, etc. For more information on priorities see chapter [Regarding Priorities](#).

13.3.2 SILENCE detection event

PLAYER ONE/ PLAYER ZERO has a special event, the SILENCE detection event: absence of a real analogue audio signal at the unit's outputs. This allows the player to continue playing the media when the programmed audio that is being played ends, or is interrupted for any reason, which may sometimes happen due to certain problems that may arise (loss of Internet, accidental disconnection of the network cable, bad files, etc.): "The show must go on".

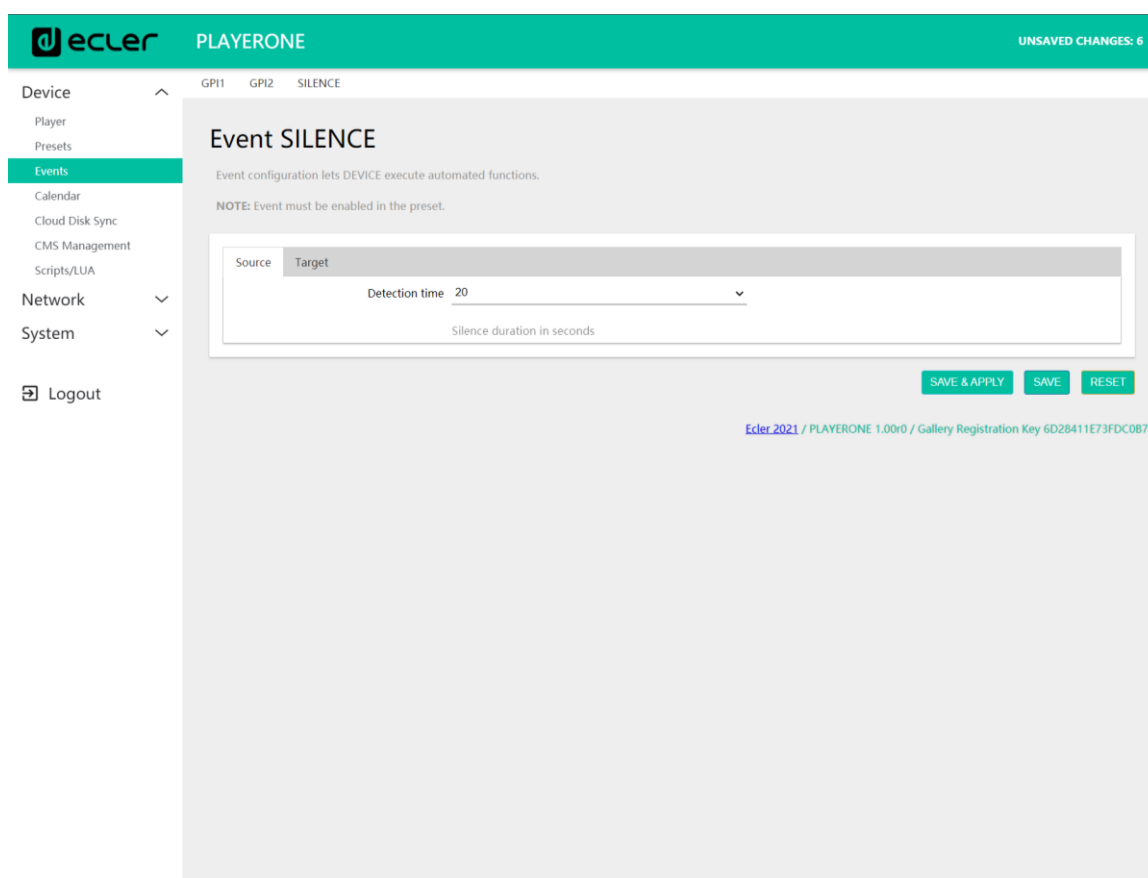


Figure 12: silence event configuration

- **Source tab:** waiting time, or trigger threshold (detection time). Select the time allowed for silence (no audio signal) before the event is triggered.
- **Target tab:** select the action that ePLAYER1 will perform when the specified waiting time has elapsed.
 - **Internal:** internal stimulus. This is useful for triggering actions within a Script.
 - **Preset recall:** loading a preset. Select the preset you want to load when the silence event is activated.

- **Load & Play Source:** loading and playback of a source. The source must be specified in the "Source path/url" field.

Tip: configure the uploading of local audio content (hosted on USB or uSD) as an action. This will ensure that you always have audio available, regardless of network connection issues that may occur outside of the player. Likewise, if you want to load a preset, ensure that the audio address stored in the selected preset is a local one (e.g. usb://) and that this preset has Play Status overwrite enabled, so that the PLAY option is selected. This will force the local medium to play, ensuring the continuity of the musical programme.

13.4 Calendar

The *Calendar* page allows calendar-triggered events to be configured.

A calendar event performs a certain action, such as loading a message with priority, in accordance with configurable parameters: date, time, repetitions, etc.

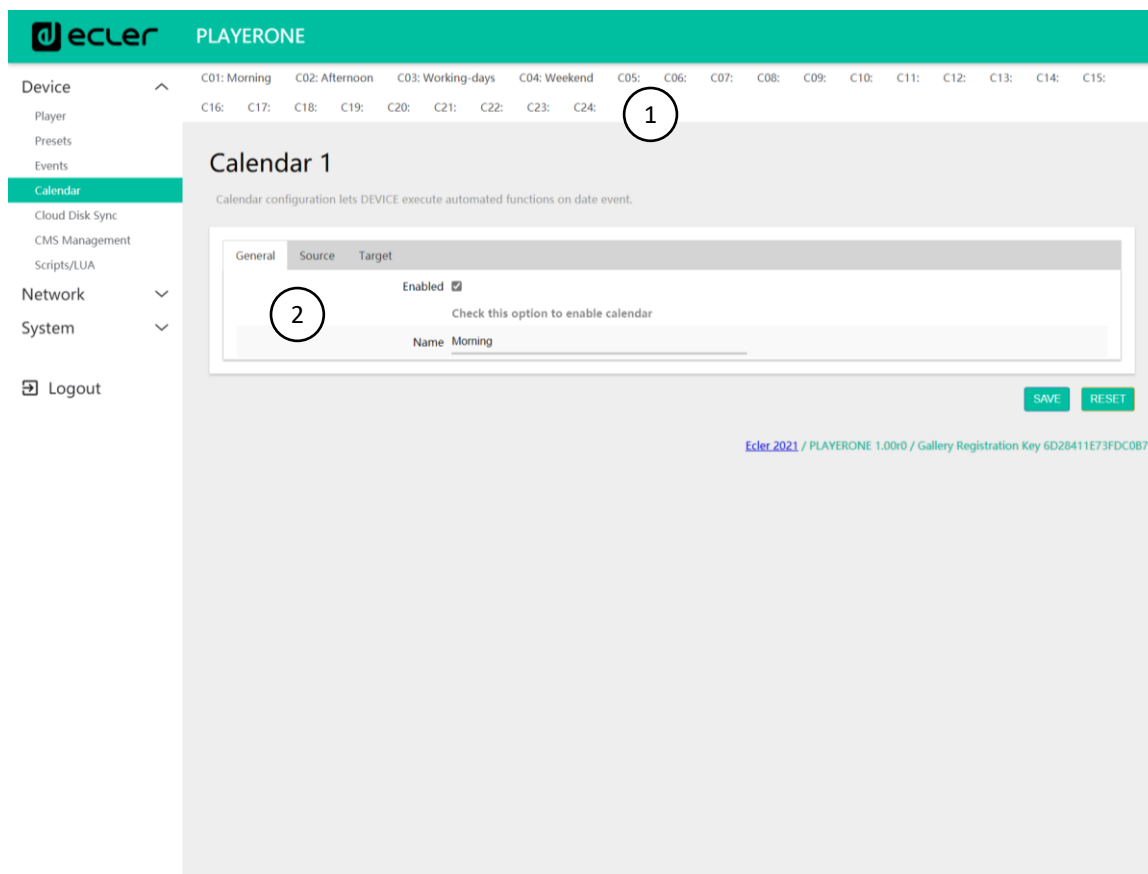


Figure 13: example of configuring a calendar event

- PLAYER ONE/ PLAYER ZERO has 24 calendar events available, each of which are fully configurable. The default names are: C01, C02...C24. When you click on the name, it will take you to the configuration for that calendar event. The names that appear for the calendar events can be edited within the preset configuration. After this setting is saved in the calendar event, you need to refresh the browser page (F5) in order to display the changes.
- The configurable parameters of each calendar event are grouped into 3 tabs:
 - **General:** activation/deactivation of calendar events and their name
 - **Source:** start/end date, start/end time, and event repeat conditions
 - **Target:** action to be performed when the event is triggered

Please Note: before configuring calendar events, make sure that the time zone is set correctly: System/Name and time.

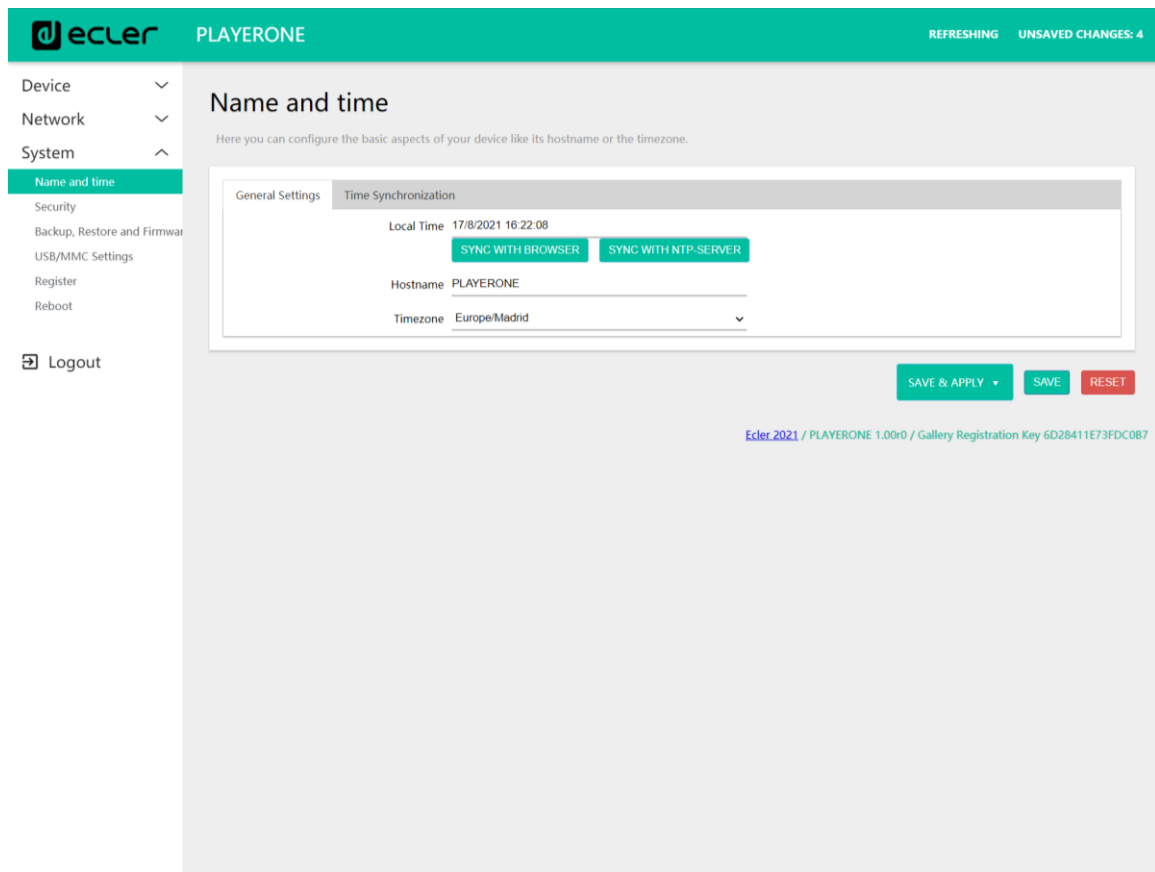


Figure 14: Name and Time configuration

13.4.1 General

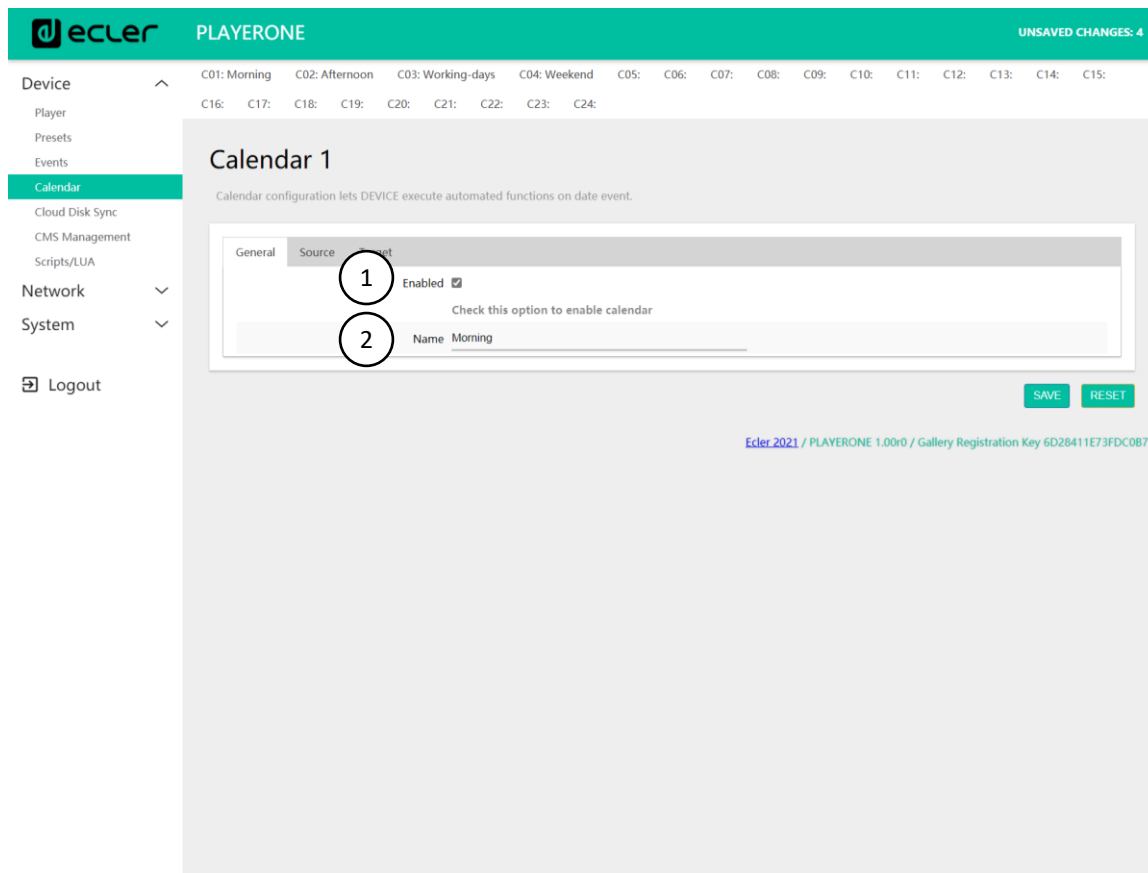


Figure 15: configuring a calendar, General

- **Enable:** allows you to enable or disable the calendar. If the calendar is enabled (ready for activation by dates/times), it will be enabled in all *presets*.
- **Name:** name of the calendar.

13.4.2 [Source](#)

In the SOURCE tab, the parameters for the triggering time/date and event replay conditions are configured.

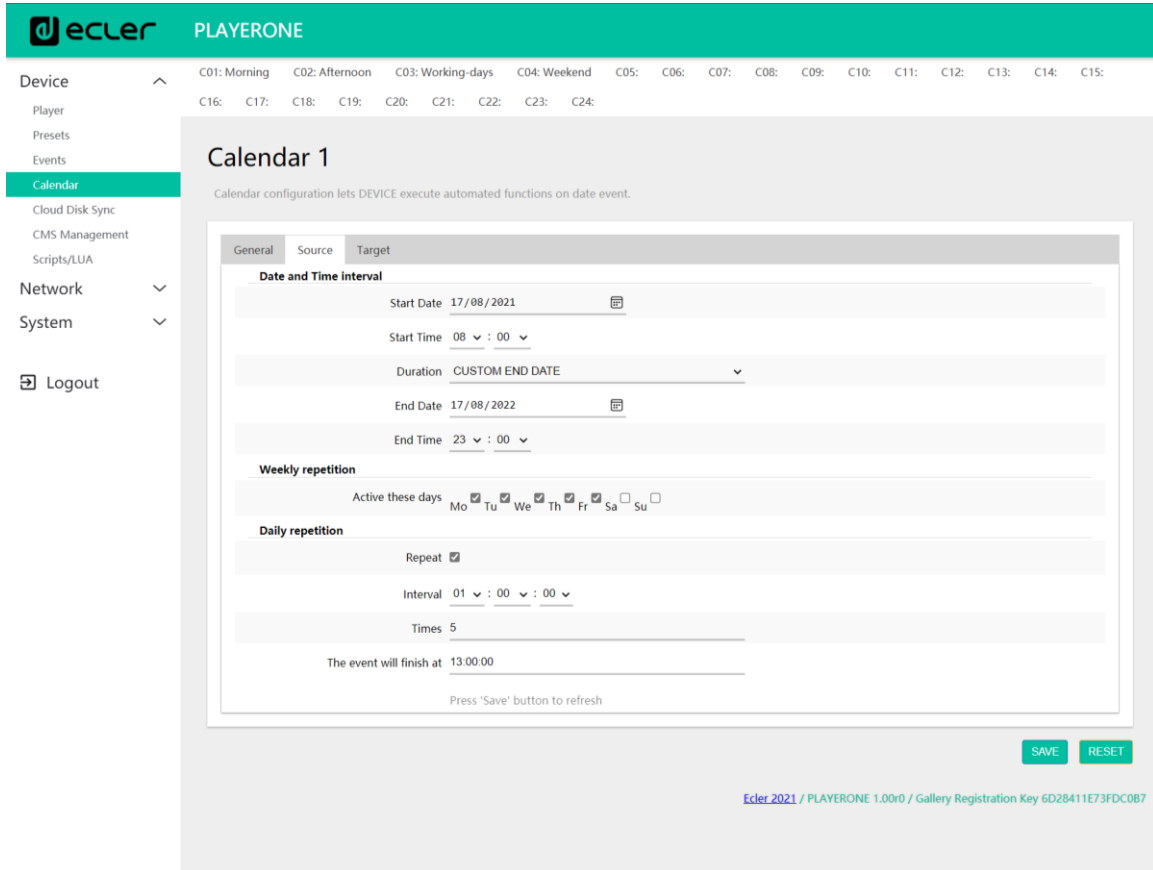


Figure 16: configuring a calendar, Source

13.4.3 [Date and time interval](#)

Defines the date and time when the event starts and the date and time when the event ends, if there is a specific end date and time.

- **Start date:** start date for the event. This field is not mandatory. If no specific date is selected ("dd/mm/yyyy"), the event will start on the same day as the changes are applied. If you select a date that is before the current date, the event will be valid as of the day the changes are applied.
- **Start time:** start time for the event. This field is mandatory. If there is a repeat condition, this will be the time at which the event is triggered for the first time each day.
- **Duration:** duration or validity of the calendar event. This allows calendar event duration settings to be selected.
 - **FOREVER:** This option is selected by default. The calendar event will not have an end date.

- **CUSTOM END DATE:** end of the calendar event. This allows you to set the date and time when the calendar event will be triggered for the last time, regardless of repeat conditions.



Date and Time interval

Start Date 17/08/2021

Start Time 08 : 00

Duration CUSTOM END DATE

End Date 17/08/2022


End Time 23 : 00

Figure 17: interval example

13.4.3.1 Weekly repetition

Weekly mask, or days of the week on which the calendar event will be repeated, at the time indicated in "Start Time".

For example, if you select Monday to Friday (weekdays), the configured event will not be triggered on Saturdays and Sundays (weekends).



Weekly repetition

Active these days Mo Tu We Th Fr Sa Su

Figure 18: example of a weekly mask

Please Note: if you do not select at least one day of the week, the calendar event will never be triggered. Also, if you set an event to be triggered on a day of the week which is not selected in the weekly mask, it will never be triggered.

13.4.3.2 Daily repetition

Daily repetitions. This option is set to be deselected by default. Selecting it will open a drop-down menu where you can configure the repetition conditions:

- **Interval:** repetition interval. This indicates how often the calendar event triggering is repeated, beginning at the Start Time.
- **Times:** number of repetitions. This indicates the number of times that the calendar event will be repeated, with time in between as indicated by the value in "interval". The first time the event is triggered each day does not count as a repetition, so if you want an event to be triggered twice a day, "times" should be set to 1 (the first time + 1 repetition). This value must be equal to or greater than 1.

Daily repetition

Repeat

Interval 01 : 00 : 00

Times 5

The event will finish at 13:00:00

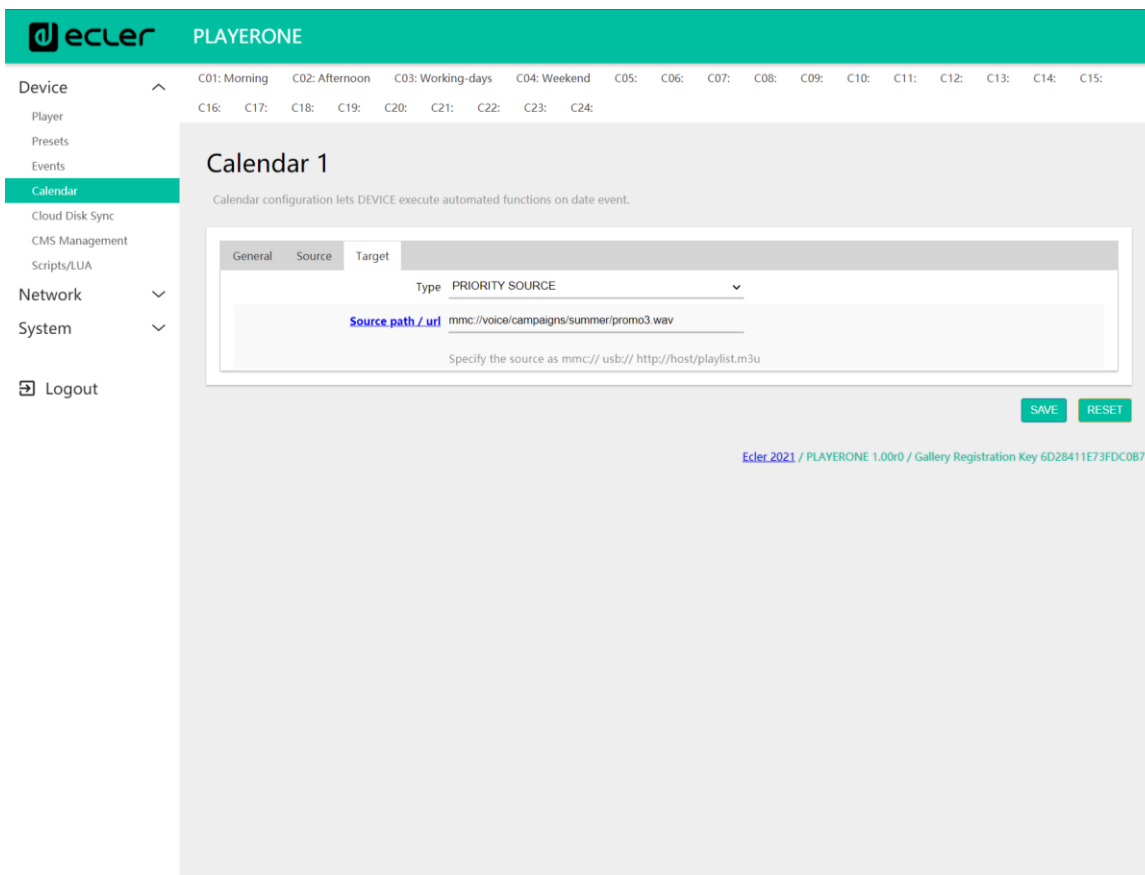
Press 'Save' button to refresh

Figure 19: daily repetition example

- **"The event will finish at"** indicates the time at which the calendar event will run for the last time each day. This parameter cannot be configured (read only). It is intended to help the user configure the "Interval" and "Times" parameters in a more efficient way.

13.4.4 Target

An action that is executed each time the calendar event is triggered.



The screenshot shows the 'Calendar 1' configuration page in the ecler PLAYERONE interface. The 'Target' tab is selected, displaying the following configuration:

- Type: PRIORITY SOURCE
- Source path / url: mmc://voice/campaigns/summer/promo3.wav
- Instruction: Specify the source as mmc:// usb:// http://host/playlist.m3u

Buttons for 'SAVE' and 'RESET' are visible at the bottom right of the configuration area.

Figure 20: calendar event example, Target

This action may be:

1. **Internal:** internal stimulus (does not perform an action directly, but allows actions to be triggered via scripts)
2. **Preset recall:** loading a preset
3. **Transport control:** pressing a button on the transport bar; STOP, PLAY, PREV, NEXT, PLAY/PAUSE
4. **Load & Play source:** loading and playback of a source
5. **Priority source:** playback of a source with priority. The source with priority will overwrite the source that is currently playing. When the priority announcement/message ends, the source that was playing will start to play again.

13.4.5 Regarding priorities

Calendar events have **lower** priority than events that are triggered by GPI. Therefore, different degrees of priority can be established. For example, in a shop, calendar-based events can be used to trigger voice messages indicating an offer, while GPI events can be used to trigger an evacuation message.

In addition, there is a higher level index that determines priority if it is the case that 2 calendar-based triggers occur at the same time. For example, CALENDAR02 is set to repeat a message every hour and CALENDAR03 is set to repeat a message every two hours (both events have the same start time). The messages will alternate every hour, as CALENDAR03 will take precedence over CALENDAR02.

If a calendar event is playing and then a second event comes in, the last event to arrive will overwrite the one that is playing, regardless of their indexes.

When a priority event is playing, it can be seen on the PLAYER ONE screen. The characters **PC01** indicate: **P**, event with Priority; **C**, calendar event; and **01**, calendar index, from 01 to 24.

If this is an event triggered by a GPI, PI01 or PI02 will be shown on the screen, indicating: **P**, event with Priority; **I**, GPI event; and **01**, GPI input index, from 01 to 02.



Figure 21

Please Note: PLAYER ZERO does not have an LCD screen.

13.4.6 Practical example of configuration of calendar events

An establishment whose business hours are 10:00 a.m. to 8:00 p.m., and which is open from Monday to Friday, requires background music, and will periodically make announcements to its customers.

- **Background music:** this will always be the same audio *stream*. It will start playing at 9:45 a.m., and it will play continuously until 8:00 p.m. At this time, there should no longer be audio playing in the establishment.
- **Announcements:** every day, the closing time of the establishment will be indicated (pre-recorded message) 15 minutes before closing. It will be recalled again 5 minutes before closing.
- **Campaigns:** there will be a special Christmas campaign from 15 December to 15 January indicating an offer (pre-recorded message). This message is to be repeated every day during this period, every 30 minutes.

Please Note: there may be several methods of meeting these requirements. This example attempts to illustrate the most relevant features of calendar events in a simple way.

13.4.6.1 Calendars for background music

2 calendar events will be created: one to load the *stream* and play it, and another to stop it.

The calendar is enabled, and it is given an identifying name.

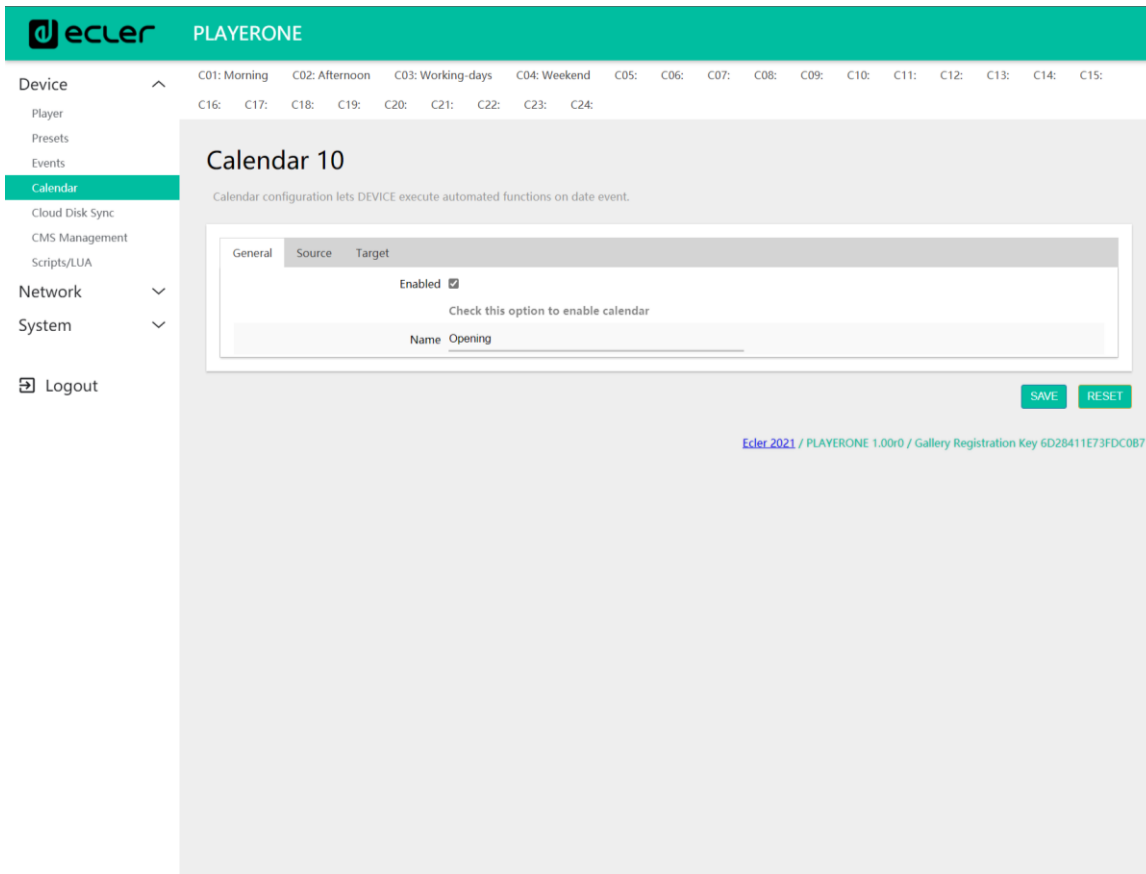


Figure 22

Since no start date is specified, "Start date" is left as the default value ("dd/mm/yyyy"). This means that the event is active from the day the changes are applied. The time is known (9:45 a.m.), and it is to be played every day, without an end date (FOREVER).

Since the event is to be repeated from Monday to Friday, the corresponding days are selected in the weekly mask.

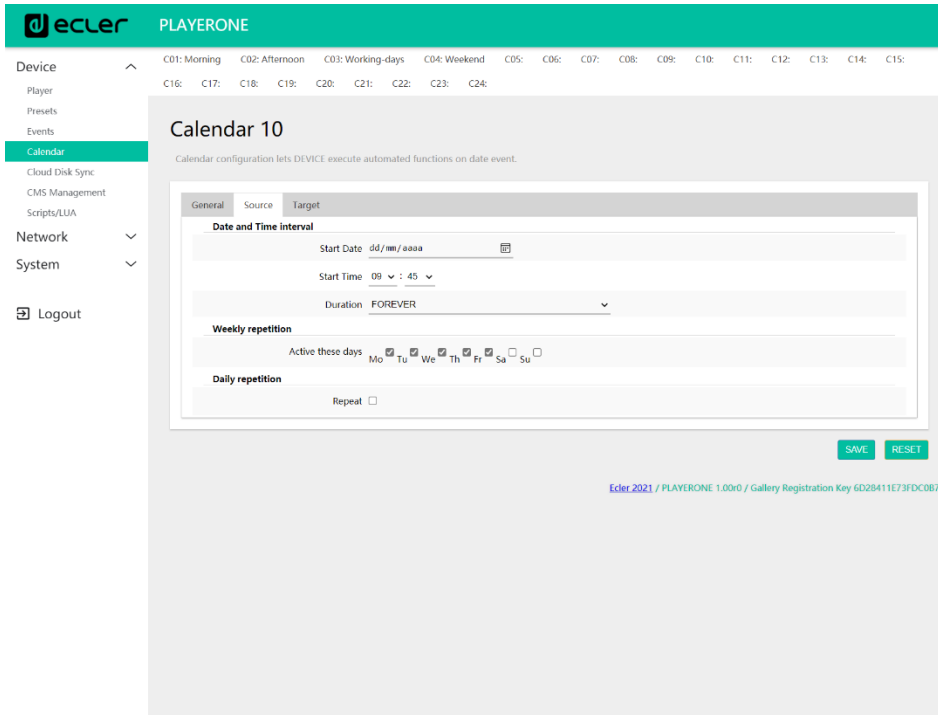


Figure 23

The action of loading and playing the specified *stream* is configured.

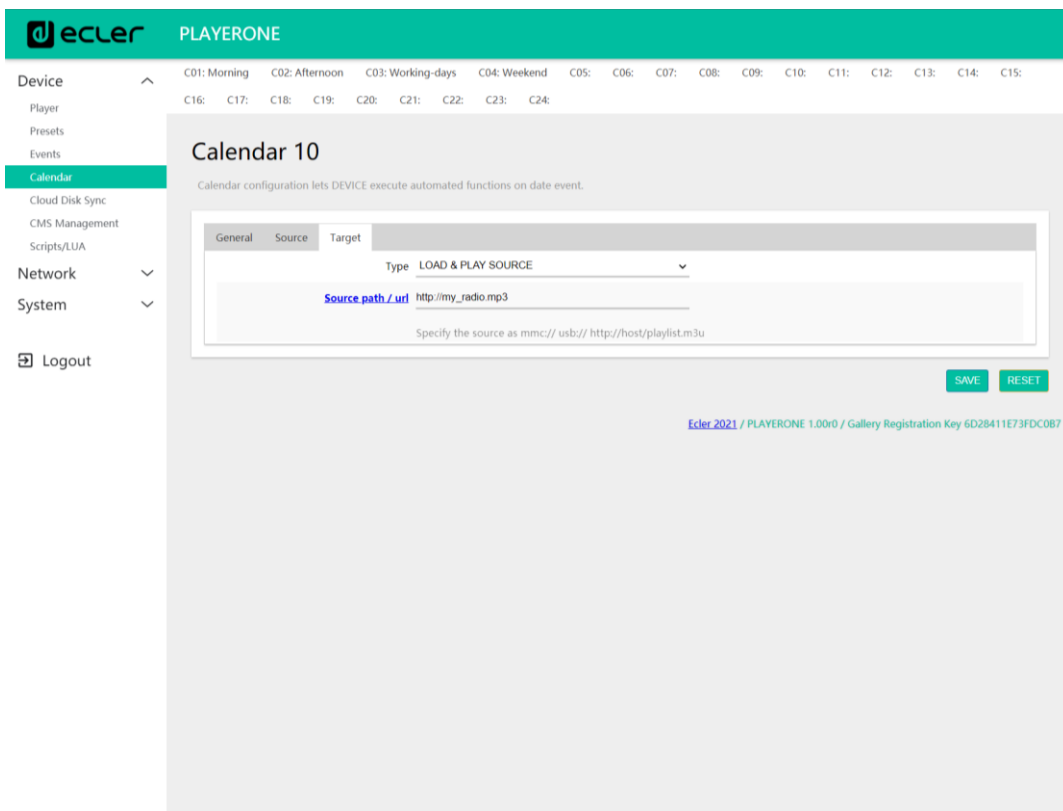


Figure 24

The calendar event that will stop the playback of the background music is configured in the same way as the last one. The only difference is that this time the action in question is to stop the playback; STOP. This calendar event will be given a different name and start time.

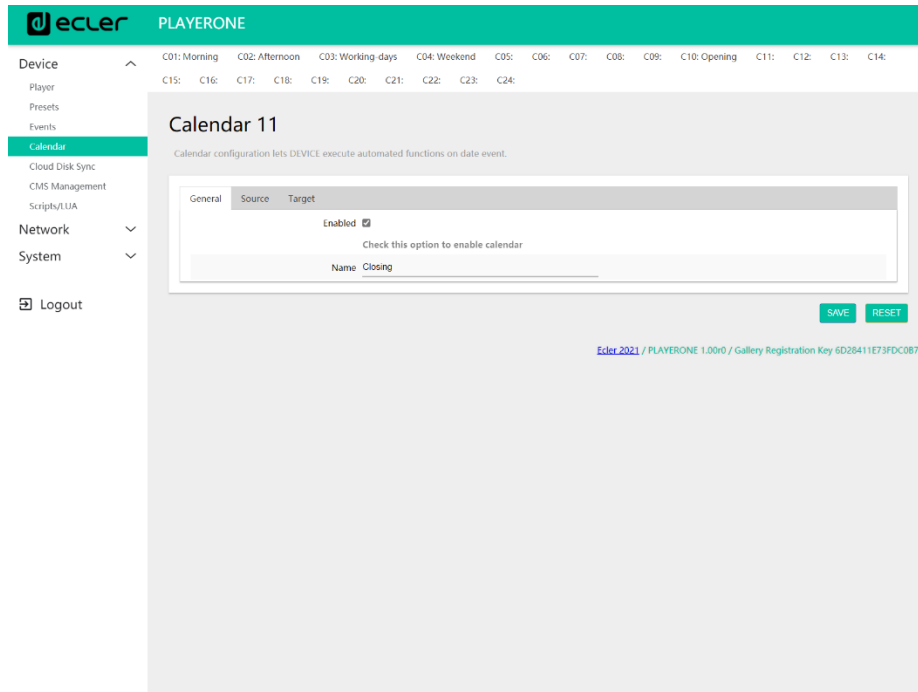


Figure 25

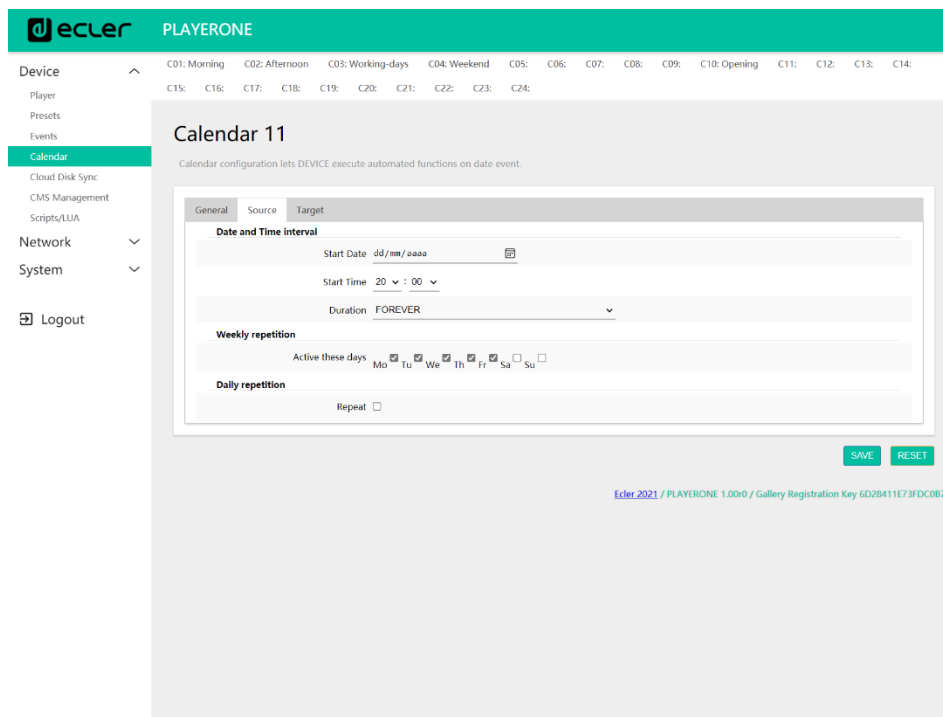


Figure 26

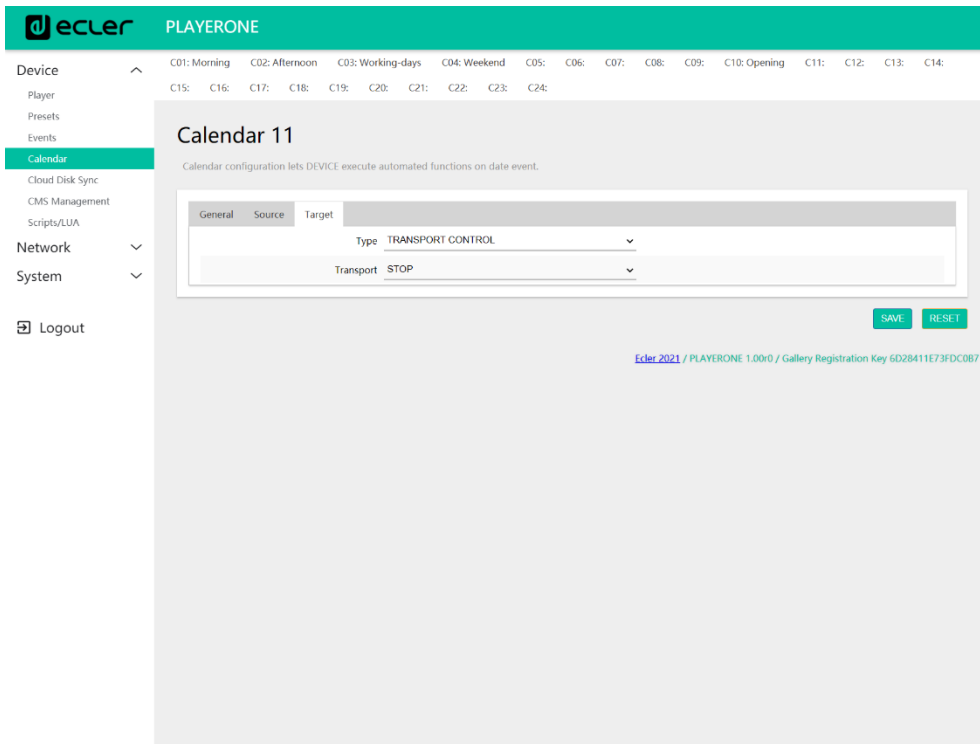


Figure 27

13.4.6.2 Calendar for closing announcements

Every day, the closing time of the establishment will be indicated (pre-recorded message) 15 minutes before closing. It will be recalled again 5 minutes before closing.

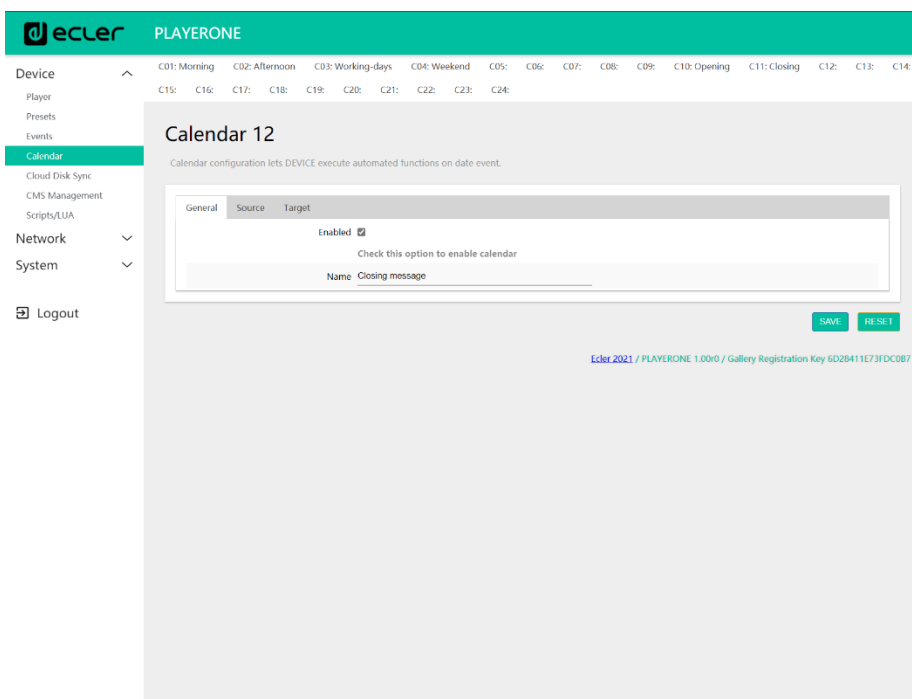


Figure 28

The message should be played twice a day: 15 minutes before closing (at 7:45 p.m.) and 5 minutes before closing (at 7:55 p.m.). Therefore, it will be necessary to configure a calendar event that repeats one time, 10 minutes after it is triggered for the first time. This is achieved by enabling *Repeat* and configuring the *Interval* and *Time* parameters correctly.

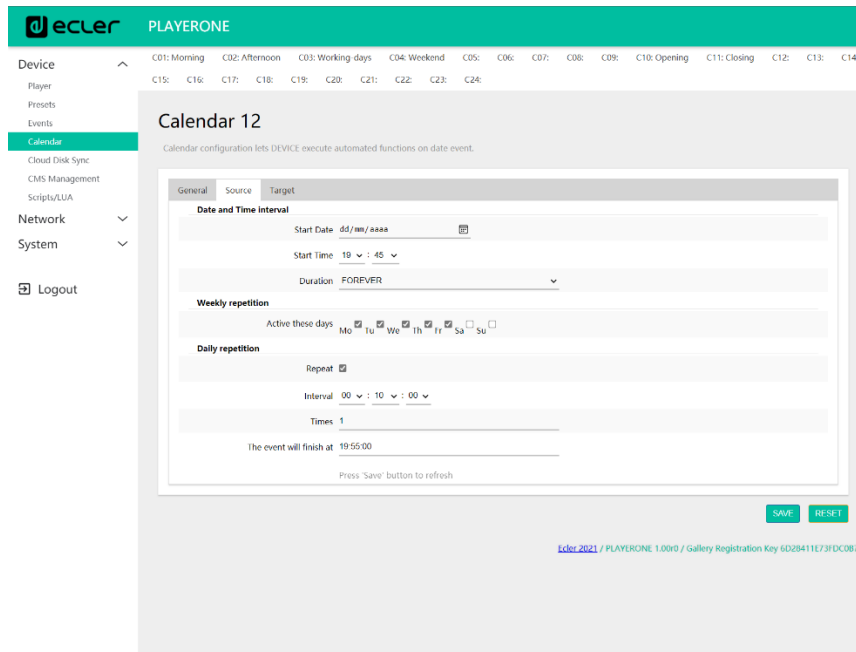


Figure 29

Finally, the message located on the SD card ("mmc://...") will be announced with priority.

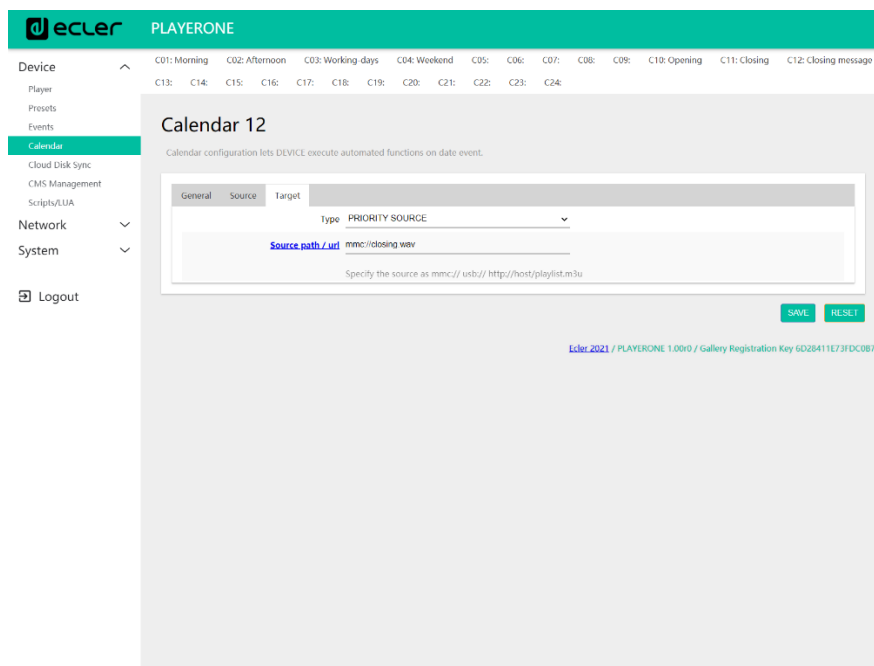


Figure 30

13.4.6.3 Calendar for Christmas campaign

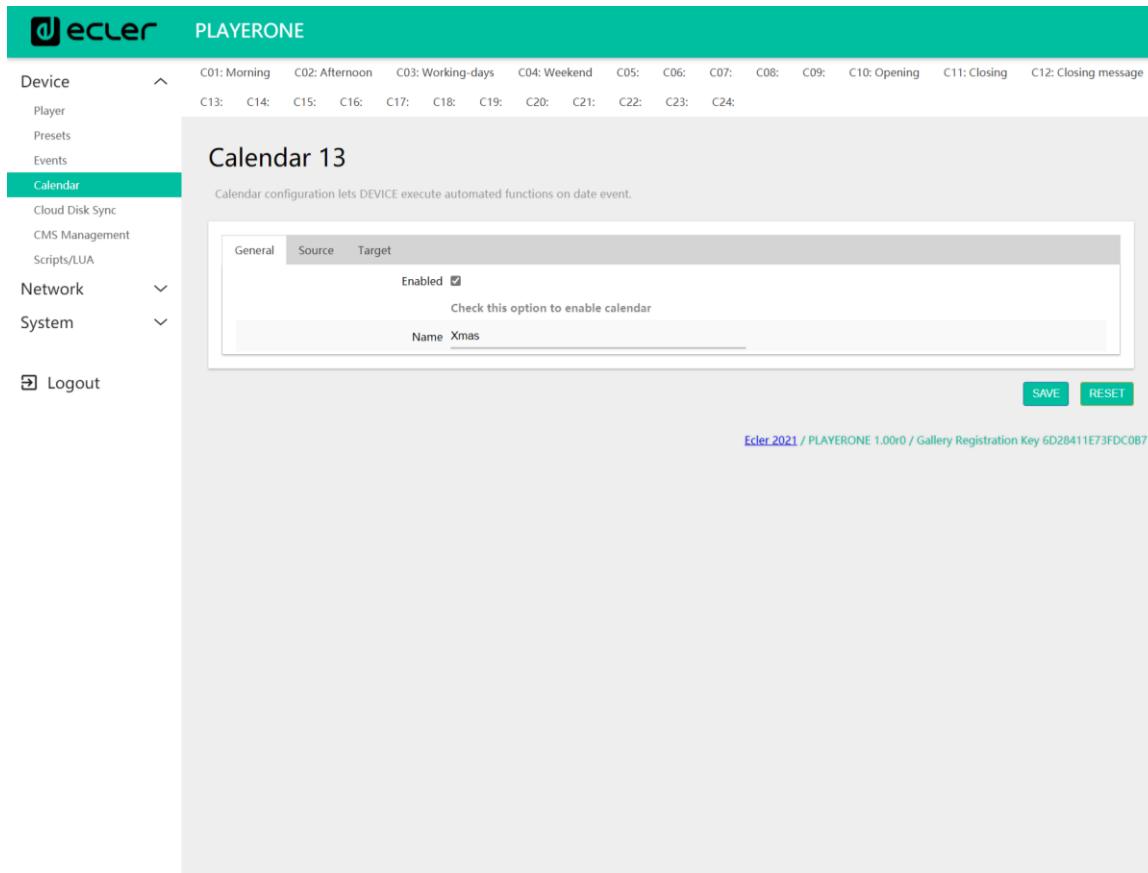


Figure 31

As this is for a specific calendar interval (15/12/21 to 15/01/22), an end date is set by choosing CUSTOM END DATE and setting the *End date* and *End time* parameters correctly.

Regarding repetitions, the message has to be played every 30 minutes, and it is known that the establishment closes at 8:00 p.m. As there will be no audio at 8:00 p.m., we will set the number of repetitions in such a way that the last time the event is triggered is 30 min before closing.

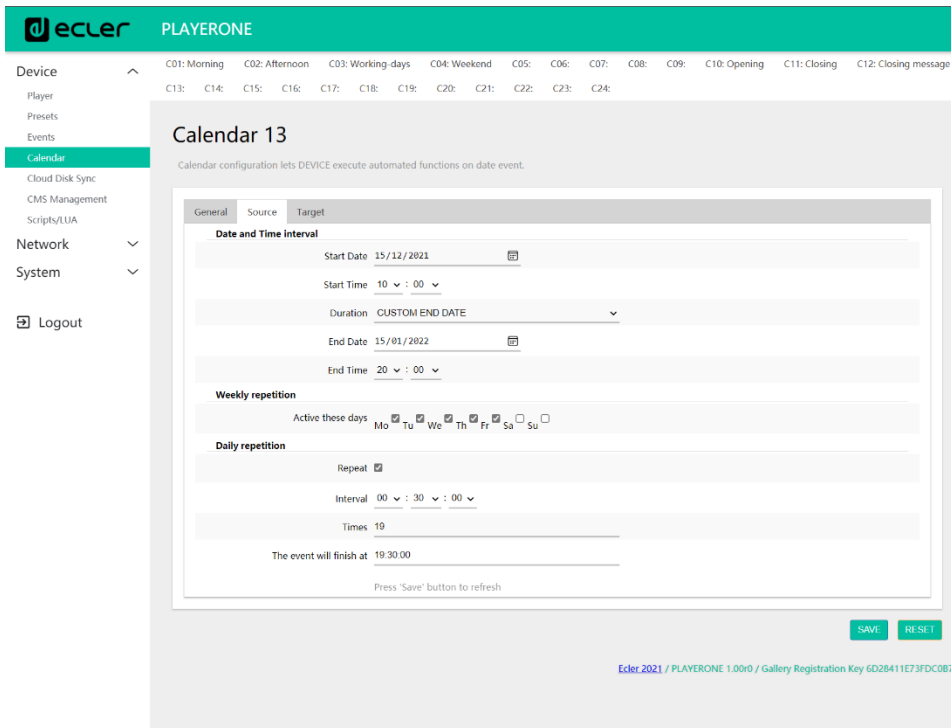


Figure 32

Finally, the message located on the USB device ("usb://...") will be announced with priority.

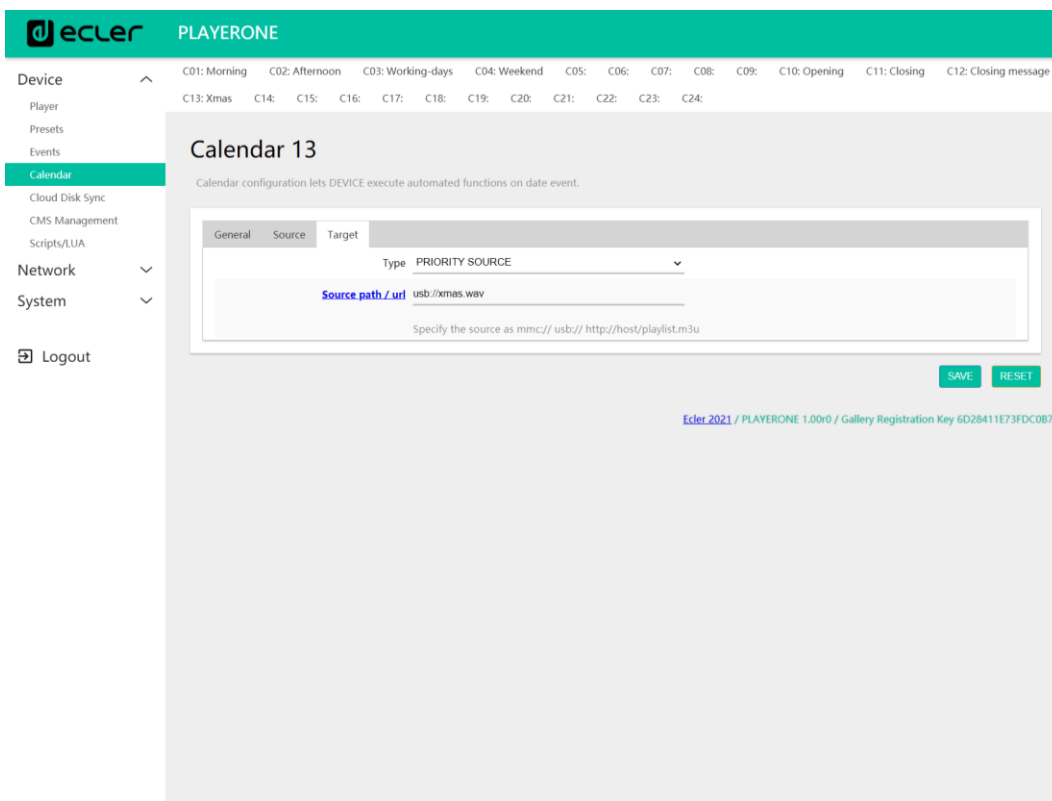


Figure 33

13.5 Cloud Disk Sync

The Cloud Disk Sync module allows the device to download remote audio content to local storage media (USB/uSD). When it is active, it checks a remote location that hosts audio content on a daily basis, compares it to the current content stored on local media (USB/uSD) and, if necessary (if differences are detected), syncs the local content to make it an exact copy of the remote content. This is a safe method of playing content during the device's working hours (during the day), as it is stored on local media, and does not carry the same risks as receiving real-time *streaming*.

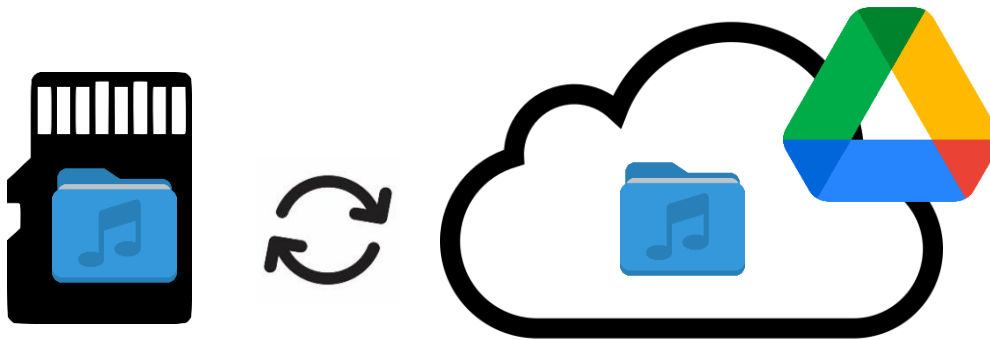


Figure 34: syncing folders with Google Drive

PLAYER ONE/PLAYER ZERO enables syncing through 2 services: **Google Drive** and Store and Forward (**rsync**). Both services are compatible, but we recommend only using one or the other, never both, because if there is an overlap between the services that are in operation, the sync with Google Drive may not run.

13.5.1 [Google Drive](#)

In the first tab on the Cloud Sync Storage page, Google Drive, you can configure the syncing with Google's cloud storage system.

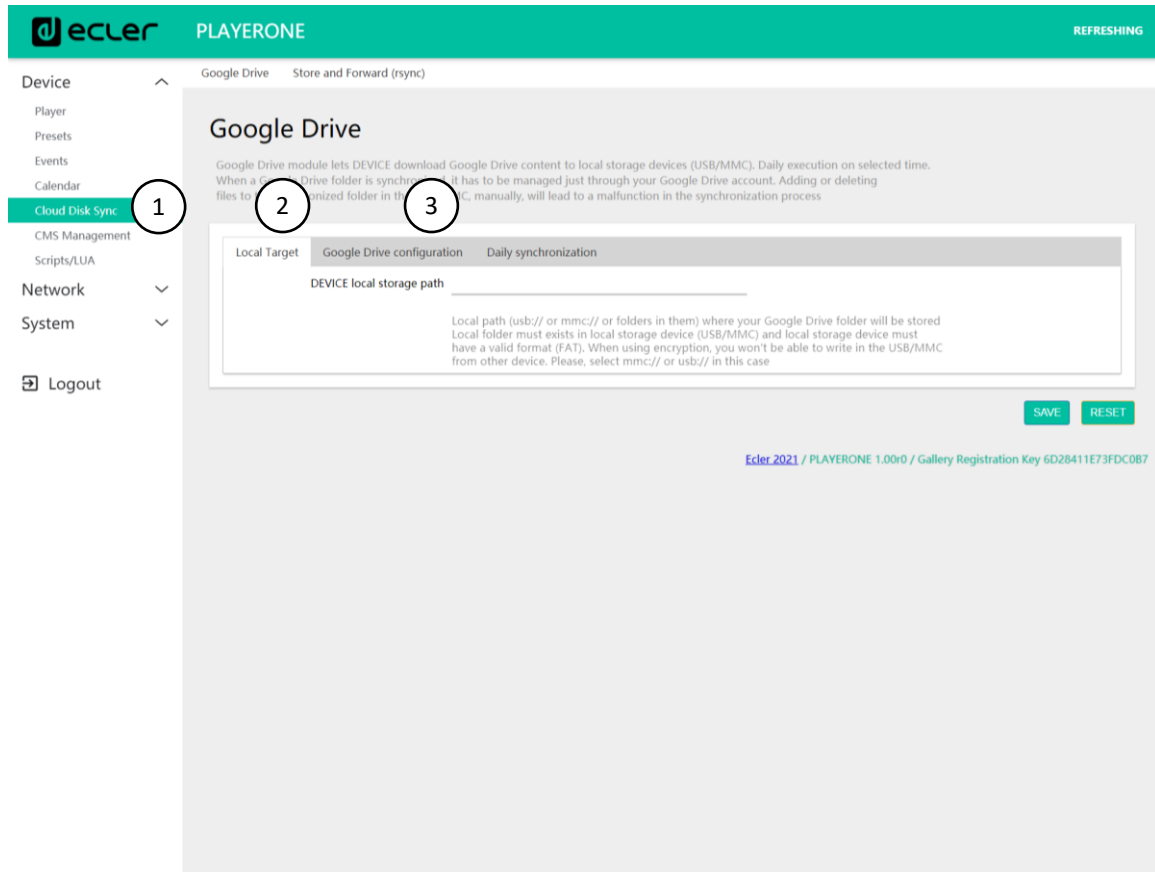


Figure 35: Google Drive configuration page

1. **Local Target:** local storage configuration, USB/uSD.
2. **Google Drive configuration:** Google account configuration and syncing.
3. **Daily synchronization:** activation/deactivation of the Google Drive syncing service, and time configuration.

Principle of operation: The Google Drive module allows the device to download content to local storage devices (USB/uSD) on a daily basis, at the selected time. When a Google Drive folder is synced to the device, it must **only be managed through the Google Drive account**. Manually adding or deleting files in the synced folder will cause the syncing process to malfunction.

Considerations:

- You will need to create a Google Account in advance in order to be able to use this service.
- Do not try to encrypt media after setting up syncing with remote content. The files will be lost and you will need to set up the service again. Encryption formats the media.

- For more information on encryption, see chapter **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia..**

13.5.1.1 Local target

Path that the contents will be downloaded to on the computer. This path must be a folder that was created in advance on your USB or uSD storage media, using a computer for example. The storage device (USB/uSD) must be in a valid format: **FAT**.

Please Note: the NTFS format supported by PLAYER ONE/PLAYER ZERO is "read-only". Write permissions are required to download and store content on the external USB or uSD media.

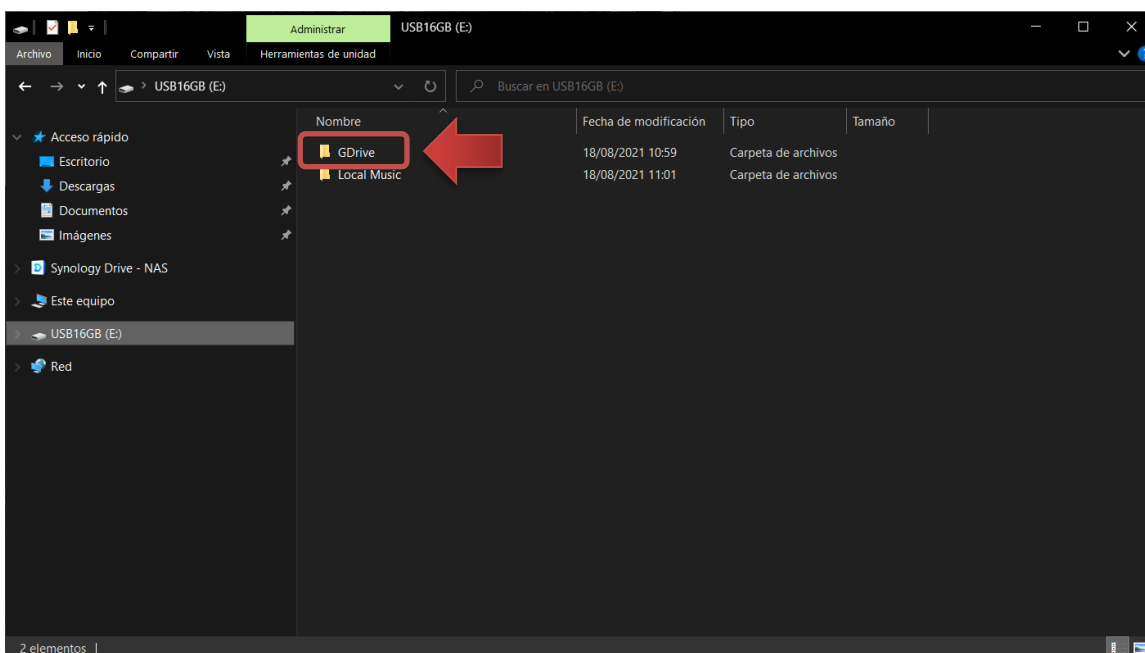


Figure 36: example of a local folder for content on Google Drive

Please Note: if you use storage media **encryption** alongside this, you will not be able to create a folder for the synced content, as the storage device will not be accessible to any device other than the device that has encrypted it. Use the root of the device as the Local Target: **usb://** or **mmc://**.

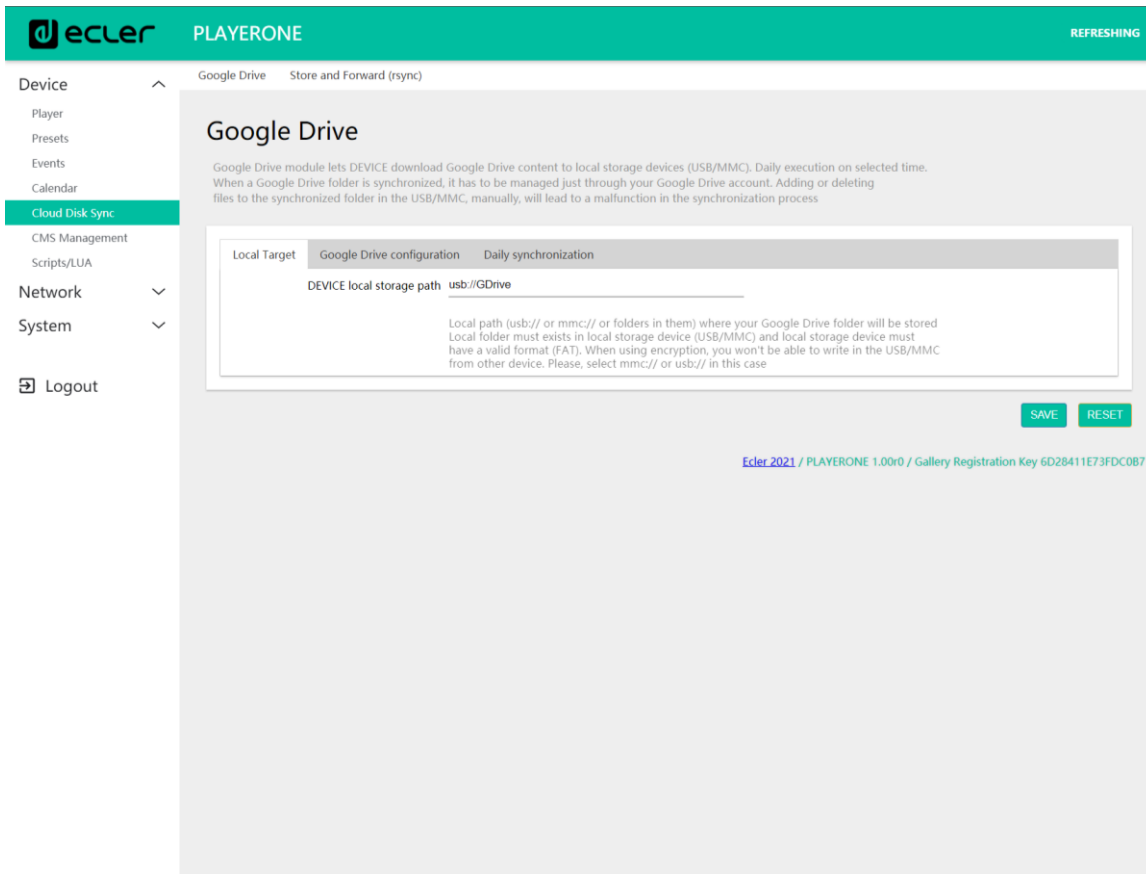


Figure 37: Local Target configuration

13.5.1.2 Google Drive configuration

Before configuring Google Drive on your device, make sure that the folder with the valid audio content that you want to sync is in your Google Drive account. Only one folder can be synced.

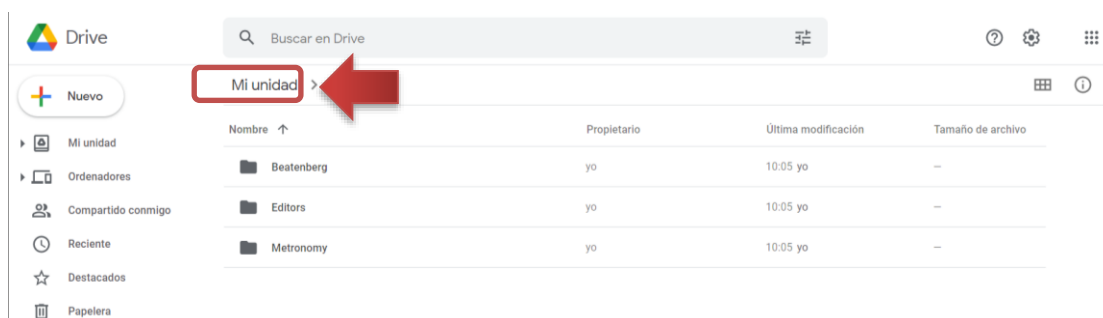


Figure 38: example of a folder in Google Drive

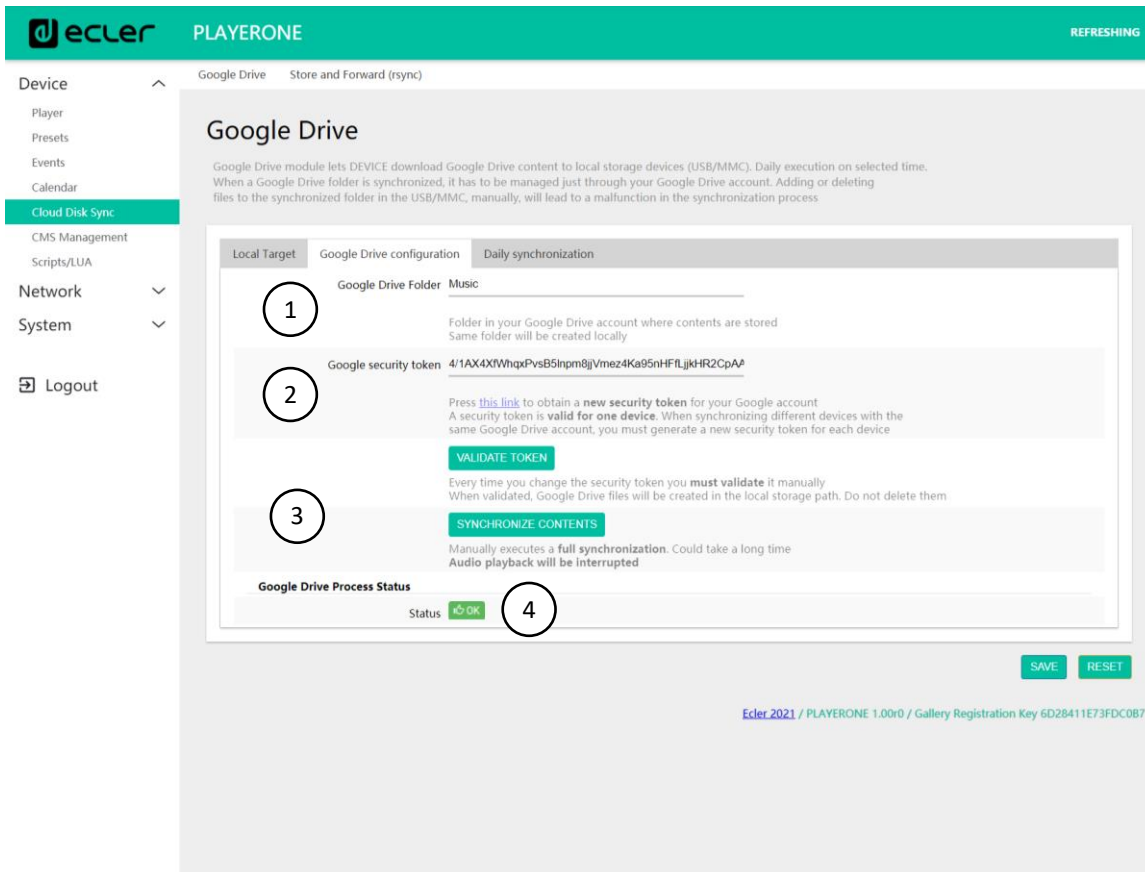


Figure 39: Google Drive configuration example

1. **Google Drive folder:** a folder that you want to sync with your PLAYER ONE/PLAYER ZERO device. This folder must contain valid audio files. Please bear in mind how much space that is available for content storage on your local storage device (USB/uSD). If the cloud folder you want to sync is larger than the space available on the storage media chosen on the device (USB/uSD), the files will be downloaded in alphabetical order until the memory (USB or uSD) is full. Figure 38 shows the "Music" folder in the root of the example Google Drive. Therefore, you would need to type "Music" in the Google Drive Folder to sync it to the PLAYER ONE/PLAYER ZERO device.
2. **Google security token:** enter the security token of your Google account in this field. To generate a new security token:
 - 1) Click on "this link"

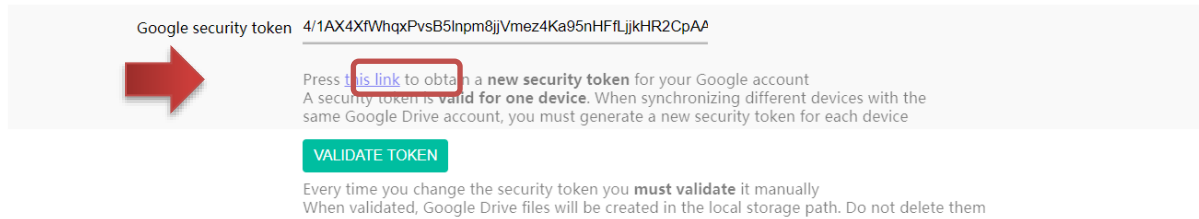


Figure 40

2) This will open a new tab in the Google application. Sign in with your Google Account

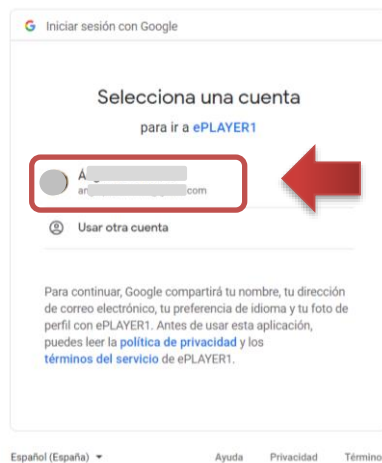


Figure 41

3) If the following warning message appears, ignore it. Click on "Advanced Settings" and then on "Go to ePLAYER1"

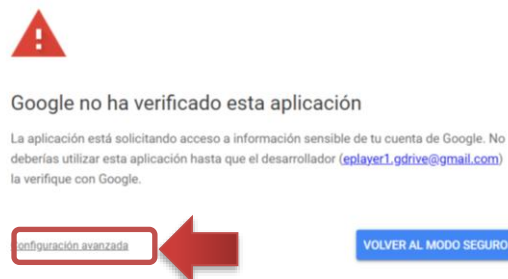


Figure 42

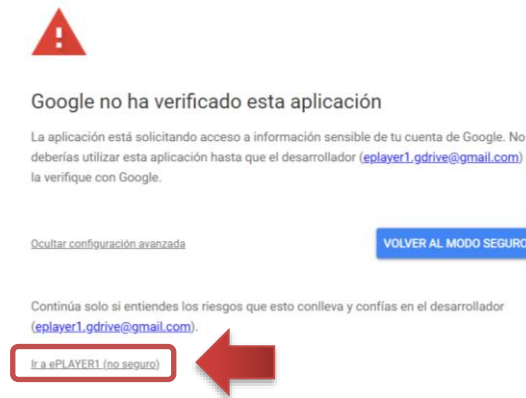


Figure 43

4) Accept the permissions required to view, modify, create and delete files, and click "Continue".

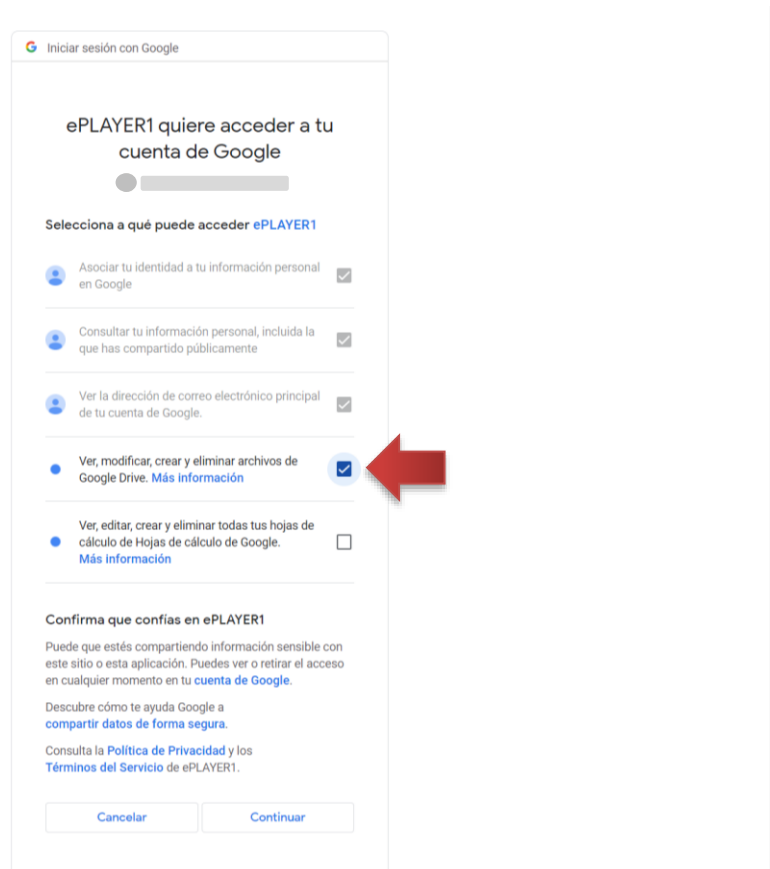


Figure 44

5) Copy the key that is generated.



Figure 45

6) Paste it into the "Google security token" field and click on the VALIDATE TOKEN button.

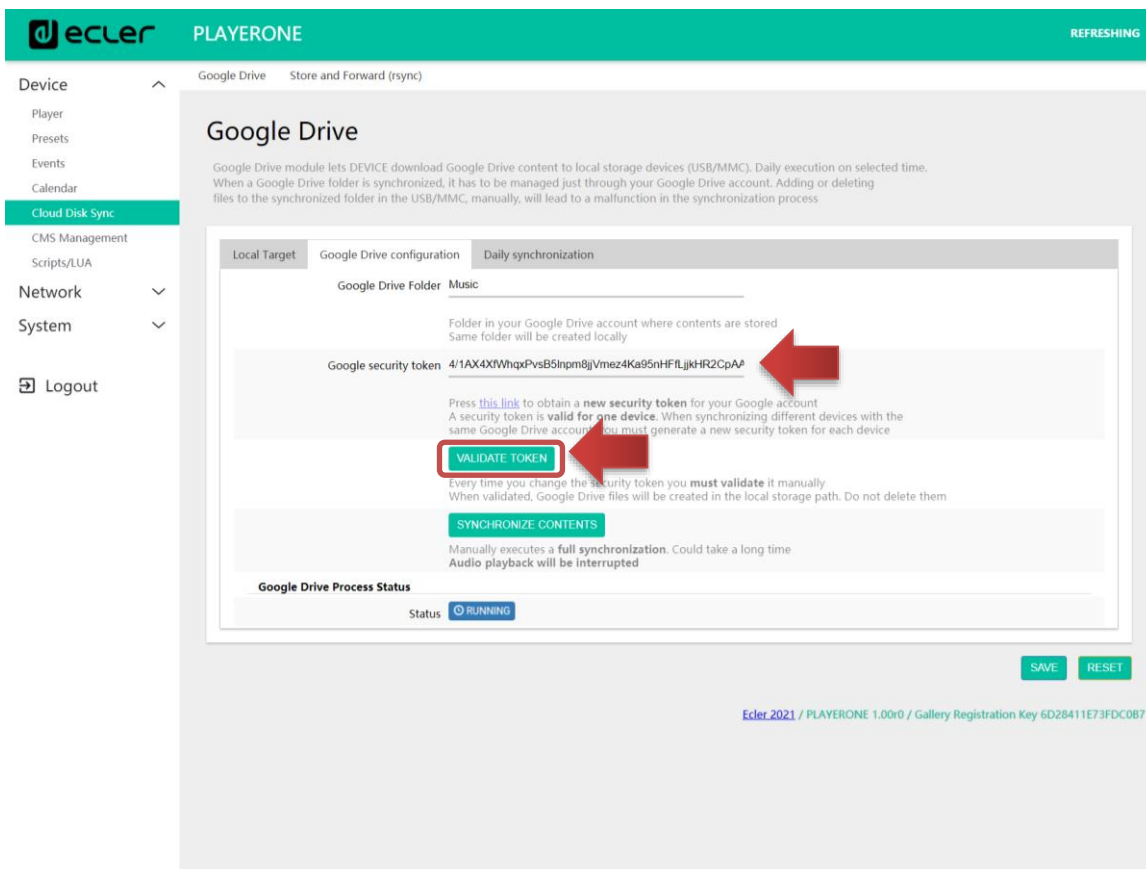


Figure 46

7) The process will start the validation of the service. The "Google Drive Process Status" field is displayed as RUNNING. This process takes a few seconds.

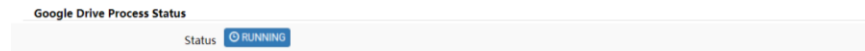


Figure 47

8) Once the validation process is finished, 2 possible messages may be displayed under "Google Drive Process Status":

- OK: validation successful. You can proceed to content syncing.

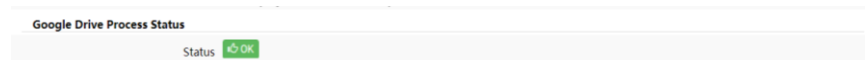


Figure 48

- FAILED: validation error.

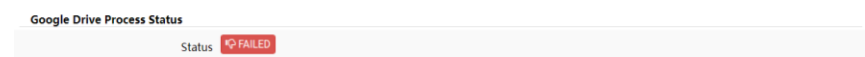


Figure 49

Perform the following validation troubleshooting checks before repeating the validation process:

- The device has an Internet connection
- Your network does not have any filters that prevent access to applications such as Google Drive
- The token is correct: it has been copied and pasted correctly
- The folder path entered in the "Google Drive folder" field matches the folder path in Google Drive.
- The USB/uSD folder has not been modified by adding or deleting files from a computer.
- The destination exists on the USB/uSD
- The USB/uSD storage media is in a valid format (FAT). NTFS is not a valid format for use with Google Drive (read-only support).
- The USB/uSD storage medium is not write protected

3. Synchronize contents: run the sync immediately (do not wait for the time set in the General tab). Recommended if it is the first time that the service is being run.

Considerations:

- This process can take a long time if it is the first time that it is being run, or if substantial changes are being made to your Google Drive folder. The time it takes depends on the number of audio files to be synced.
- Audio playback will be interrupted during syncing.
- Do not disconnect your device from the Internet or the mains during the syncing process.

4. Google Drive Process Status: indicates the status of content syncing.

- **OK:** validation/sync successfully completed.
- **RUNNING:** validation/sync process in progress.
- **FAILED:** validation/sync error.

13.5.1.3 3.5.1.3. Daily synchronization

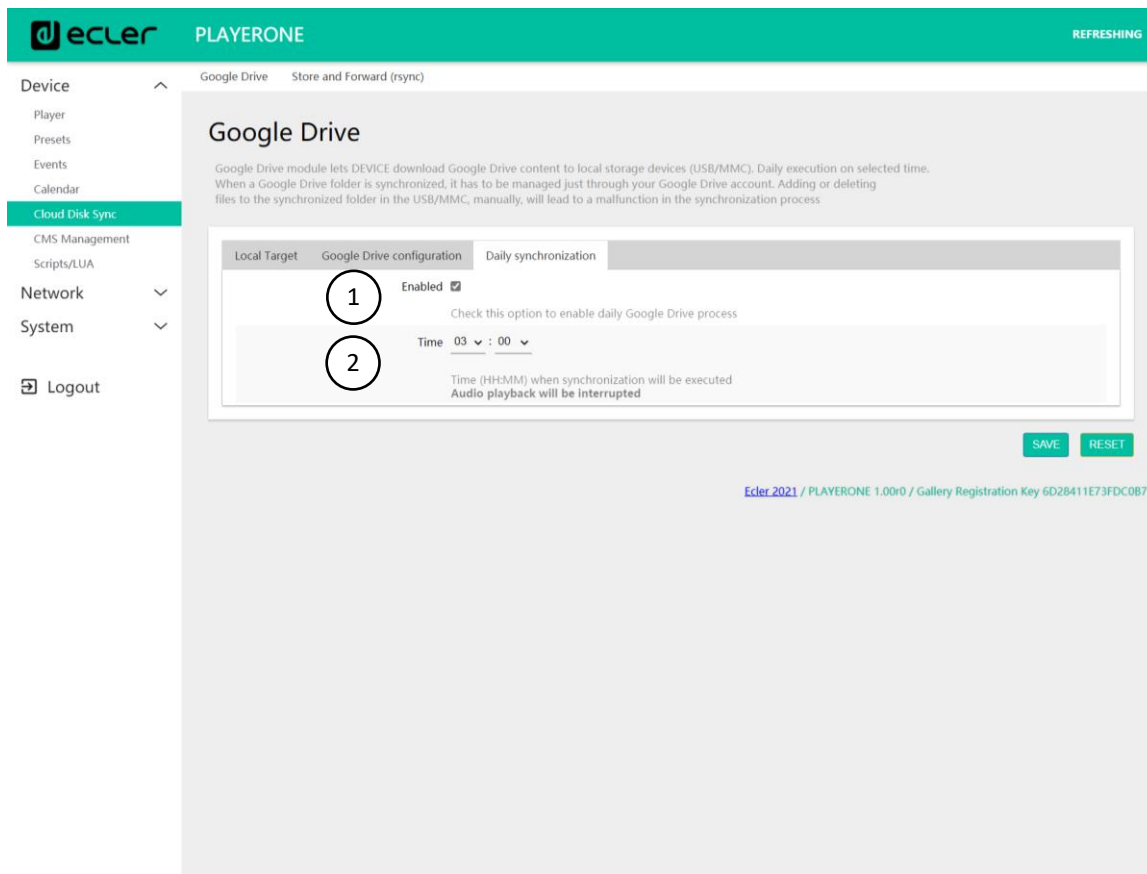


Figure 50. daily synchronization

1. **Enable:** enables/disables the sync service.
2. **Time:** daily sync time. Syncing with Google Drive will run every day at the specified time.

Tip: Select a sync time that is outside the operating hours of the player. Please note that the operation of downloading content from the cloud may take several minutes (depending on the size of the files, the number you need to download and the available Internet connection). Do not turn off the player during this operation or before it runs.

13.5.2 Store and Forward (rsync)

The Store & Forward module allows you to sync the content of the USB/uSD device with a folder that is hosted on a remote server on a daily basis. It also allows for automatic playback of this content, in combination with the *Load preset1* reboot mode. This service uses the *rsync (Remote Sync)* sync tool.

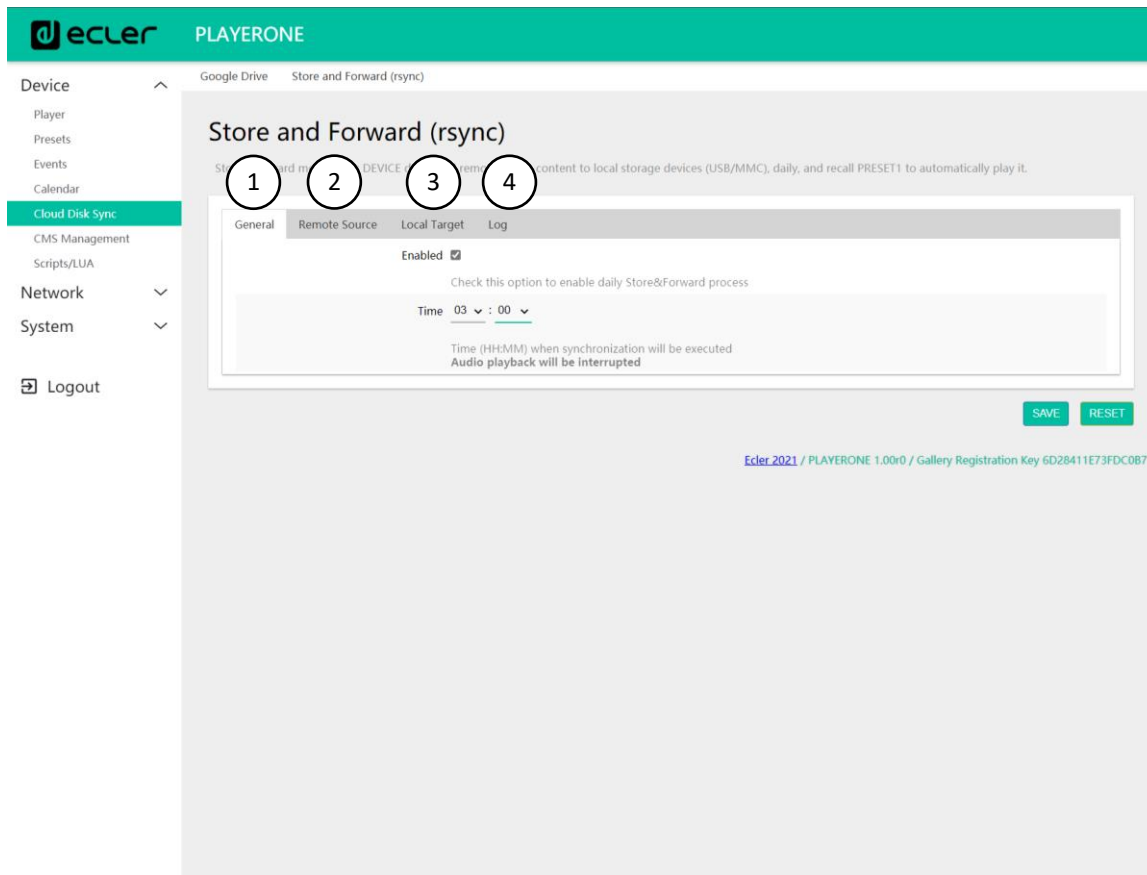


Figure 51: Store & Forward module

1. **General:** enables the S&F service and selects the sync time.
2. **Remote source:** remote server configuration.
3. **Local target:** configuration of the local folder where the content is stored.
4. **Log:** activity log for the S&F module

For more information on the Store & Forward service, please [refer to Annex1 Configuration of an SSH server for Store & Forward.](#)

13.5.2.1 General

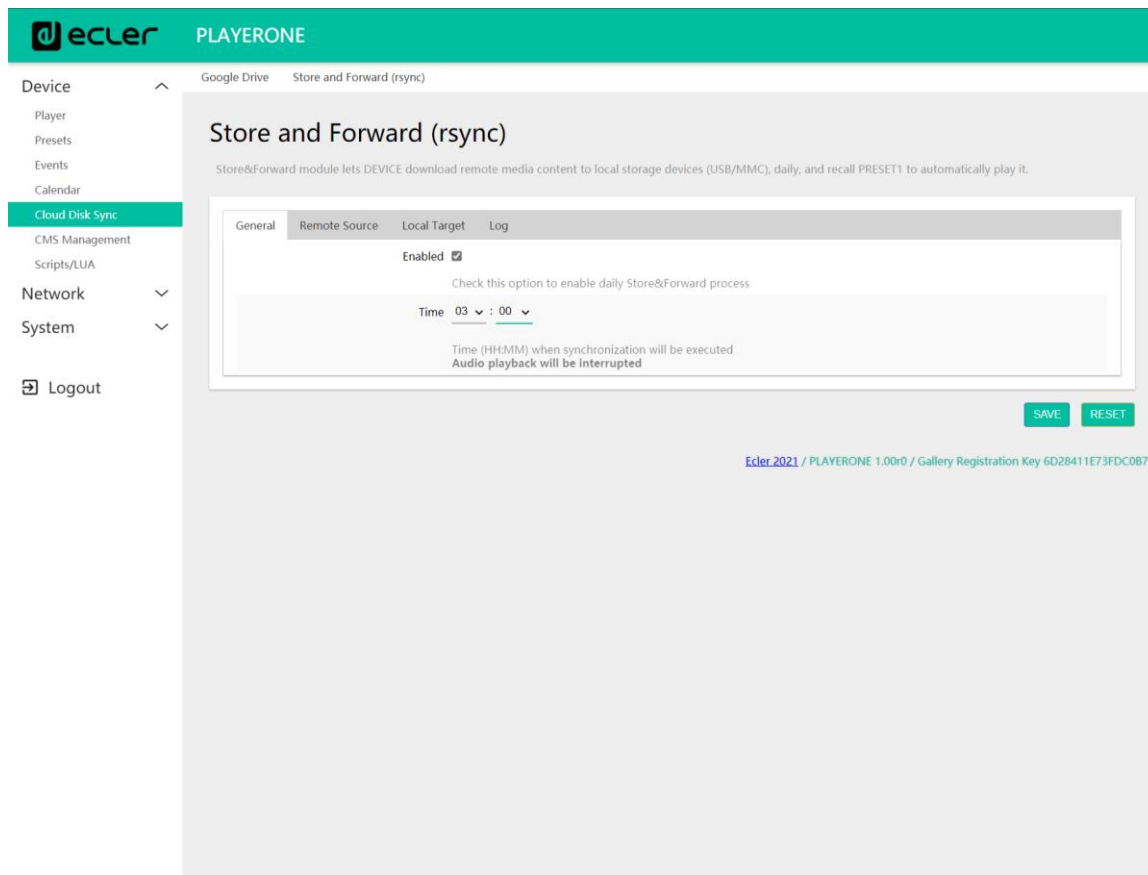


Figure 52: S&F, General

- **Enabled:** This enables/disables the running of the daily sync.
- **Time:** time at which the daily sync is run.

13.5.2.2 Remote source

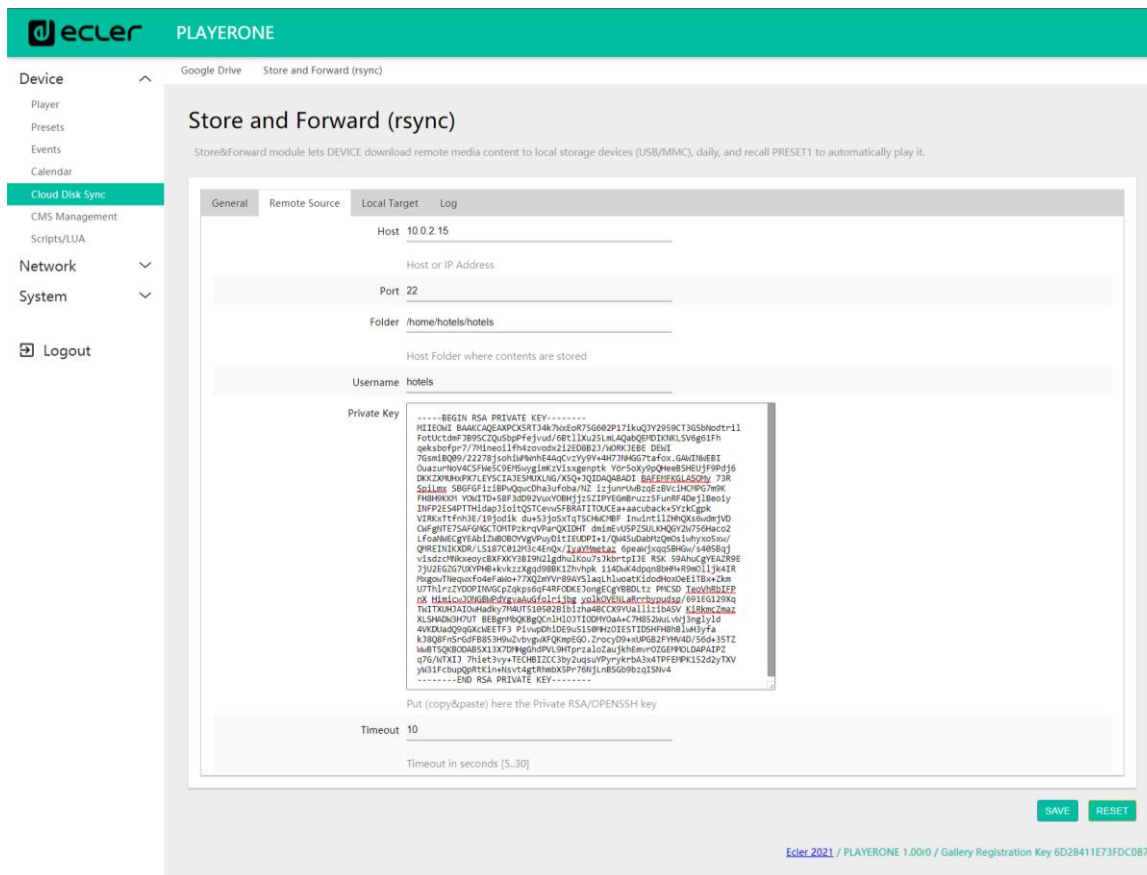


Figure 53: S&F, Remote source

- **Host:** host or IP address of the server.
- **Port:** server port, by default, 22
- **Folder:** directory within the server where the audio content that is to be synced is stored.
- **Username:** user or name of the content group
- **Private key:** private password generated for the indicated user or content group.

Please Note: for security and efficiency reasons, the remote server where the content is hosted must be **SSH**, and public and private passwords must be enabled and used.

13.5.2.3 Local target

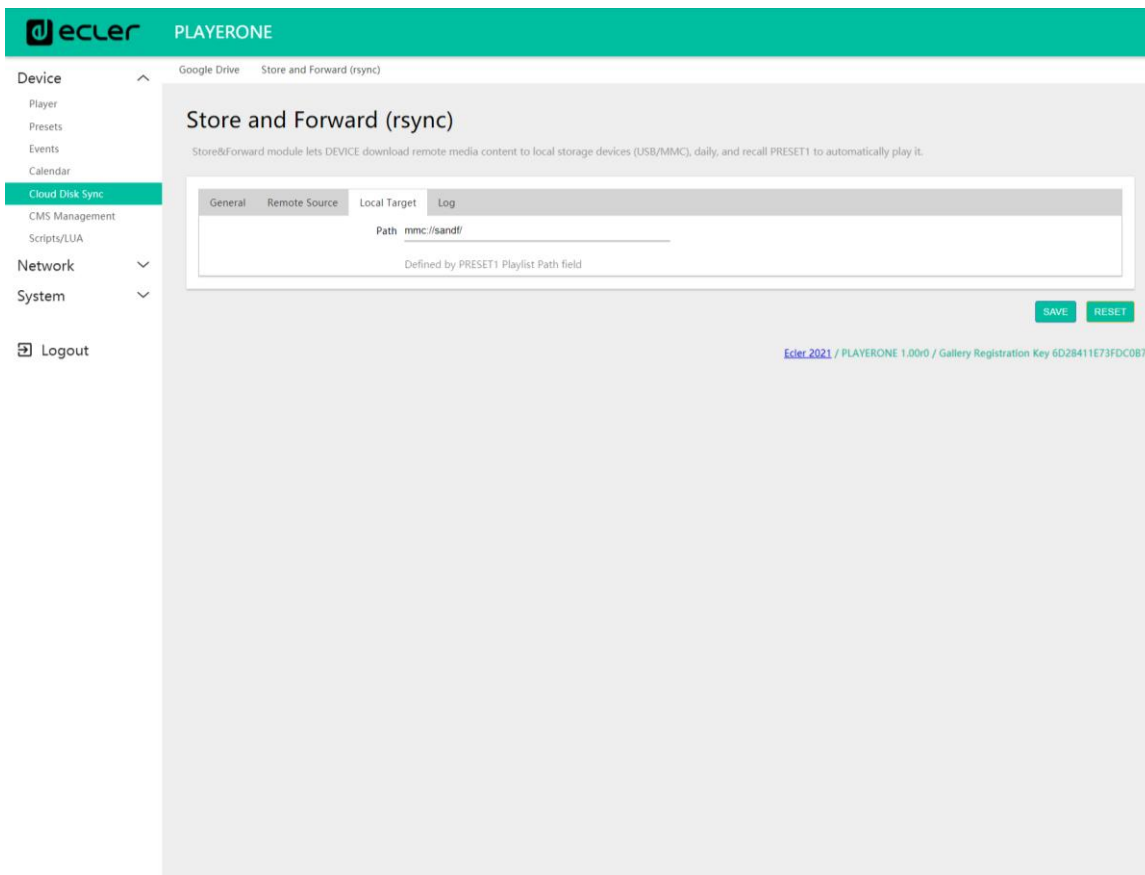


Figure 54: S&F, Local target

- **Path:** defined by the Playlist Path field for PRESET01. This can be changed in the *preset* settings.

13.5.2.4 Log

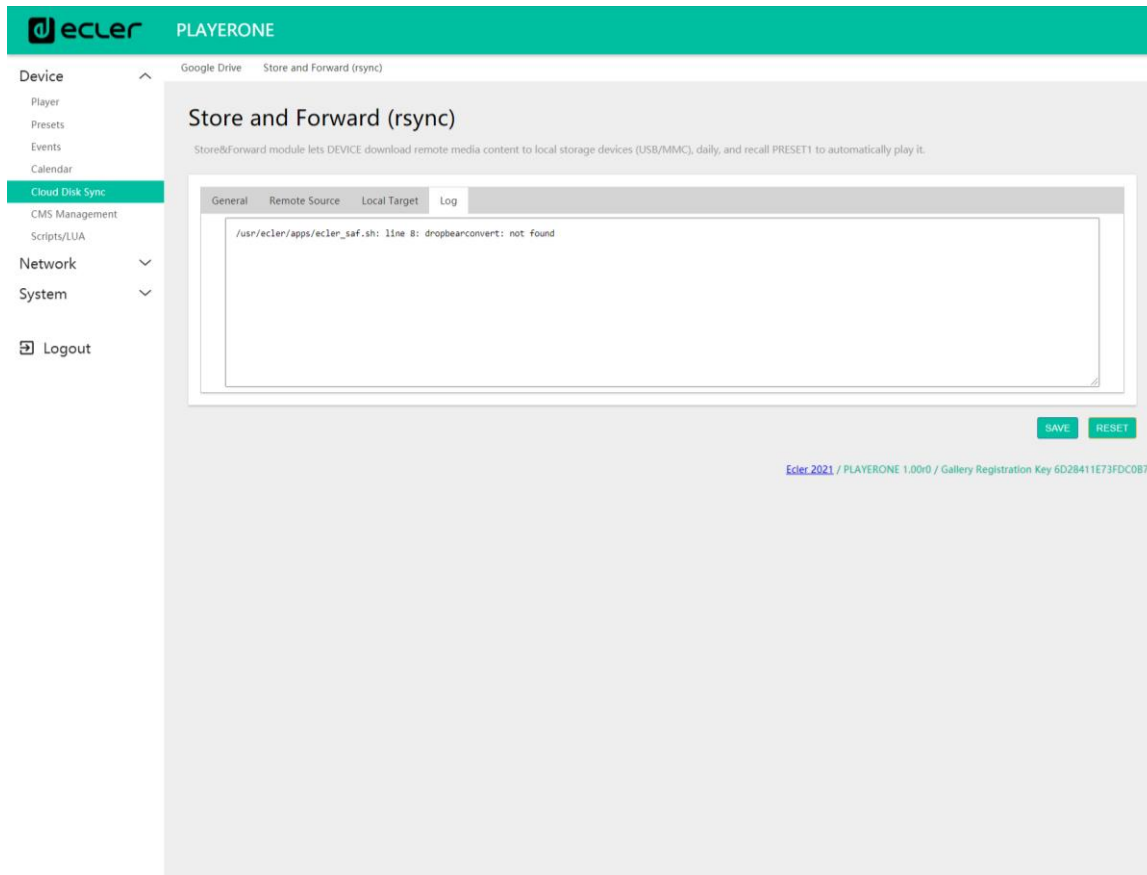


Figure 55: S&F, Log

- This displays information and activity in relation to the *rsync* syncing process. It is useful for debugging possible server or device configuration problems.

13.6 CMS Management

CMS stands for *Content Management System*.

In this mode, PLAYER ONE/PLAYER ZERO can be configured to behave as a client of a content management platform, obeying a schedule (calendar events, playlists, announcements, etc.) that is managed by the CMS company via the Internet.

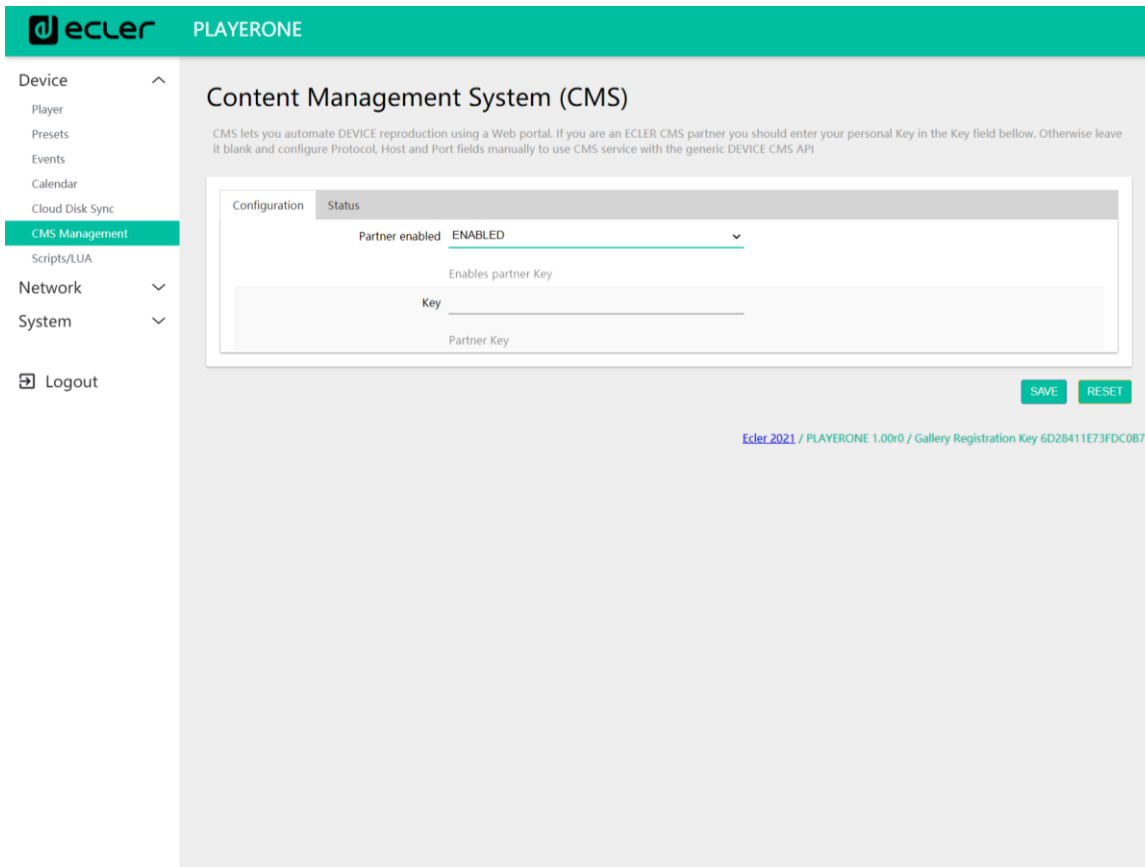


Figure 56: CMS

This page is available to provide support for CMS companies that used this functionality in the past. For further information please contact your supplier, distributor or fill in the contact form on our website in the Support/Technical Queries section.

13.7 Scripts/LUA

A *script* is a simple program; a file of instructions written by the user in LUA language (<https://www.lua.org/>).

Each *script* can be thought of as a script that the player runs; a series of pre-programmed tasks that are run when a certain stimulus is received.

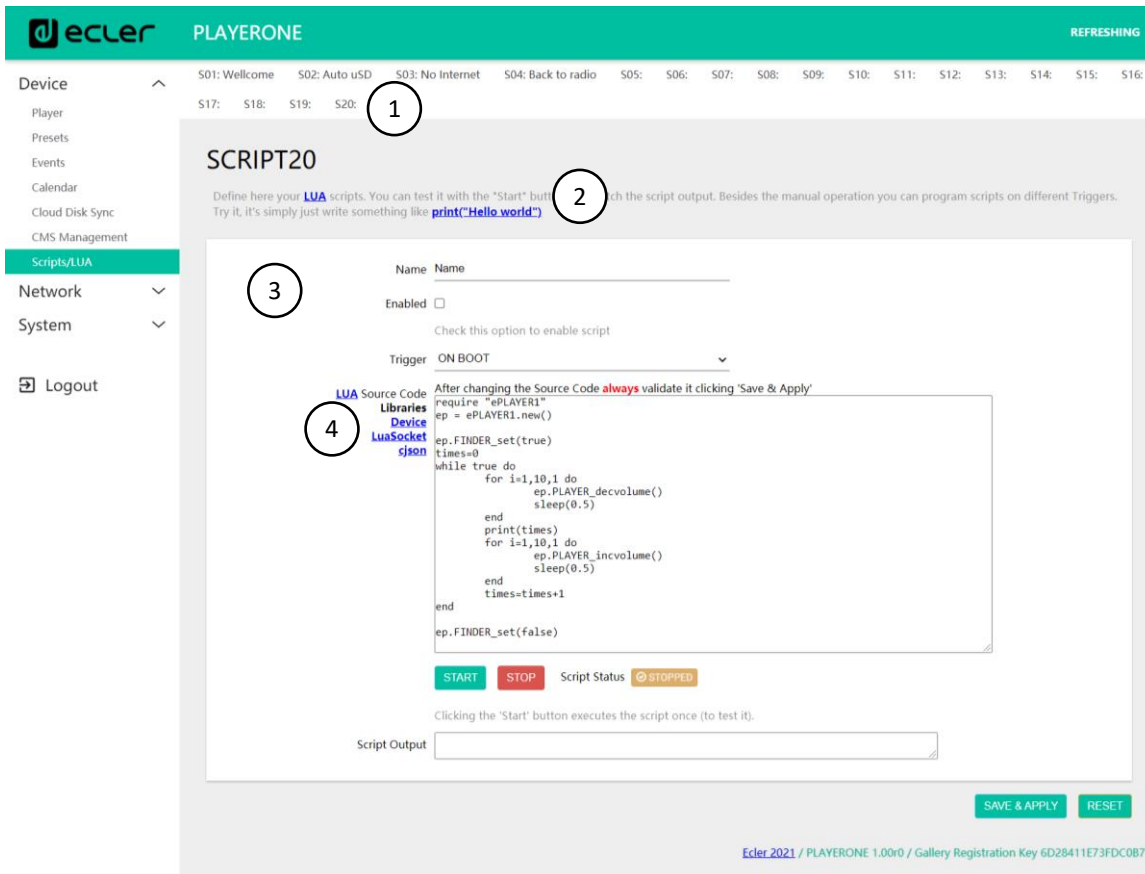


Figure 57: Script example

- PLAYER ONE/ PLAYER ZERO has **20 scripts** available, each of which is completely configurable. The default names are: S01, S02...S20. When you click on the name, it will take you to the configuration for that *script*. The name that appears here for the *script* can be edited within the *script* configuration. After this setting is saved in the *script*, you need to refresh the browser page (F5) in order to display the changes.
- Useful information for *script* configuration and Lua reference manual.
- Parameters of each *script* that can be configured:
 - **Name:** name given to the *script* by the user. This name will appear in the header of the Scripts/LUA page.
 - **Enable:** enables or disables the *script*.
 - **Trigger:** stimulus that triggers the execution of the *script*. There are different ways to trigger a *script* or automate tasks:

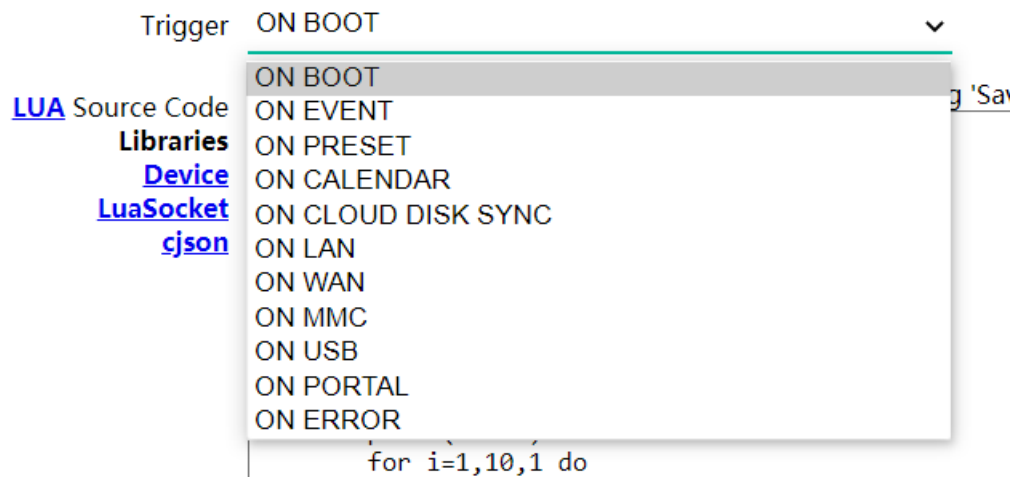
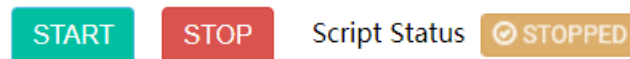


Figure 58: available stimuli

- **ON BOOT:** when the device boots up. Start-up tasks can be configured.
- **ON EVENT:** when a GPI event (GPI1 or GPI2), or the silence detection event (SILENCE), is triggered
- **ON PRESET:** after loading a *preset*. One of the 20 *presets* has to be selected, and when it is recalled the *script* will be triggered.
- **ON CALENDAR:** when a calendar event is triggered. One of the 24 calendar events has to be selected, which will, in turn, trigger the script.
- **ON CLOUD DISK:** when syncing with a remote server is successfully completed (with sync result "OK"). It must indicate which of the servers it synced with: Google Drive or *rsync*
- **ON LAN:** when detecting whether the LAN (local network) is available or not.
- **ON WAN:** when detecting whether the WAN (Internet access) is available or not.
- **ON MMC:** when a uSD card is connected or disconnected, and recognized correctly by the device.
- **ON USB:** when a USB device is connected or disconnected, and recognized correctly by the device.
- **ON PORTAL:** when syncing with the Ecler Gallery is successfully completed (with sync result "OK").
- **ON ERROR:** when the error indicated in "Error code" with a code occurs. A table of error codes can be found in the PLAYER ONE PLAYER/ZERO programming library (programmer's appendix).
- **LUA Source Code:** the text of the *script* must be entered in this field.

- **Execution buttons:** script execution and stop buttons. The START button runs the *script* immediately, without the need for the programmed stimulus to occur. The *script* must be saved before you click on this button in order for the changes to be applied. This is useful for testing. The STOP button stops the running of the *script*. These functions are especially useful during *script* programming, as they allow you to check the actions executed by each *script* and debug the code.



Clicking the 'Start' button executes the script once (to test it).

Figure 59: script execution buttons

- **Status:** shows the status of the *script*: RUNNING (flashing) the *script* is running, or STOPPED (steady), the *script* has finished or stopped.
- **Script output:** output/return value of the *script*. You can write output messages that will appear on this screen. This is useful for debugging scripts.

Script Output Returned value: 0

Figure 60: example of a script return value

- **Documentation for programmers:** links (blue) available on the device for consultation (Internet connection required):
 - **LUA:** LUA programming language manual
 - **Device:** PLAYER ONE/PLAYER ZERO LUA library (programmer's annex). Details the objects, functions, and parameters of the library. Interface between LUA and the firmware of the device using the JSON protocol.
 - **LuaSocket:** LuaSocket library documentation.
 - **cjson:** LUA CJSON module documentation. Provides JSON support for LUA.

The following chapters show a number of examples of simple *scripts*. Keep in mind that this PLAYER ONE/PLAYER ZERO programming and customization tool is very powerful, as the same *script* can execute several tasks which can be linked together, depending on various circumstances, and thus implement an entire logic and intelligence in the way it works.

The only limit is your imagination!

13.7.1 Example Script01:

Write messages on the LCD screen

The following is an example of how to write a message on the LCD. This message has been configured so that, when the player starts up, the message "Hello Ecler" is displayed on the screen for 10 seconds.

Please Note: this is only available for PLAYER ONE. PLAYER ZERO does not have an LCD screen.

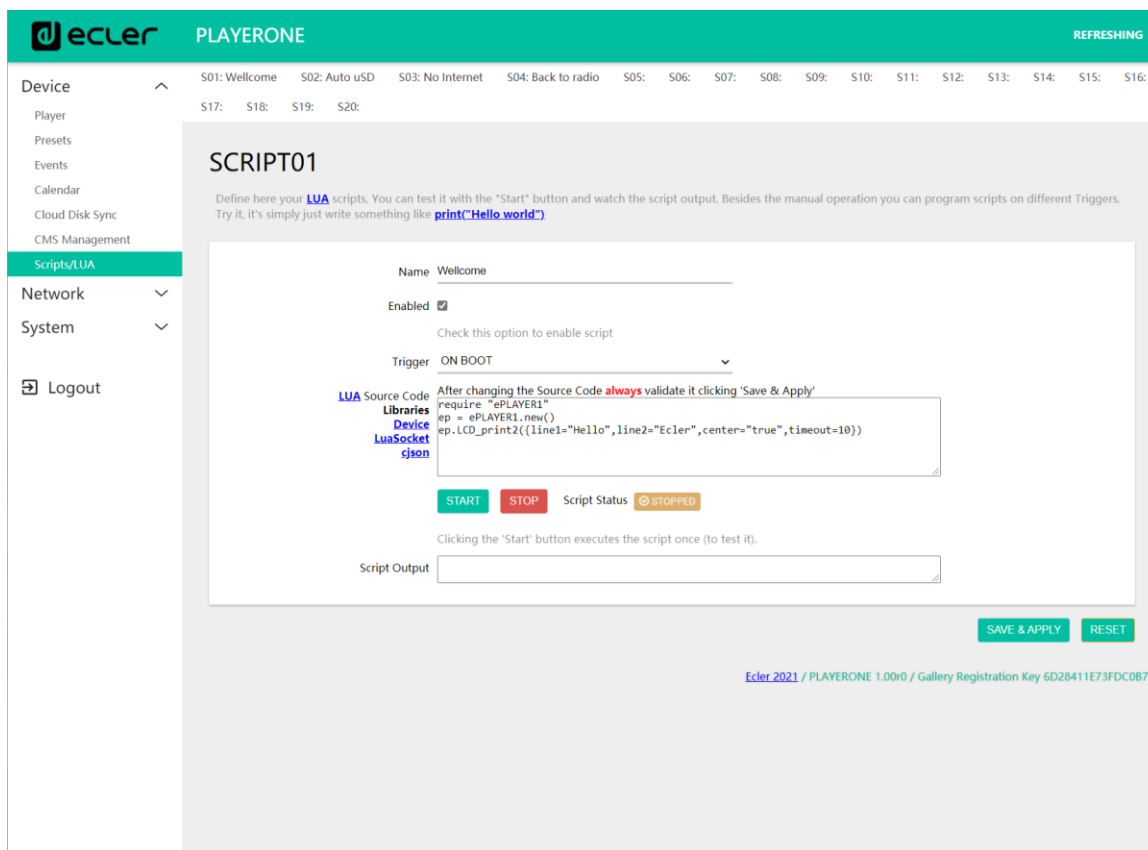


Figure 61

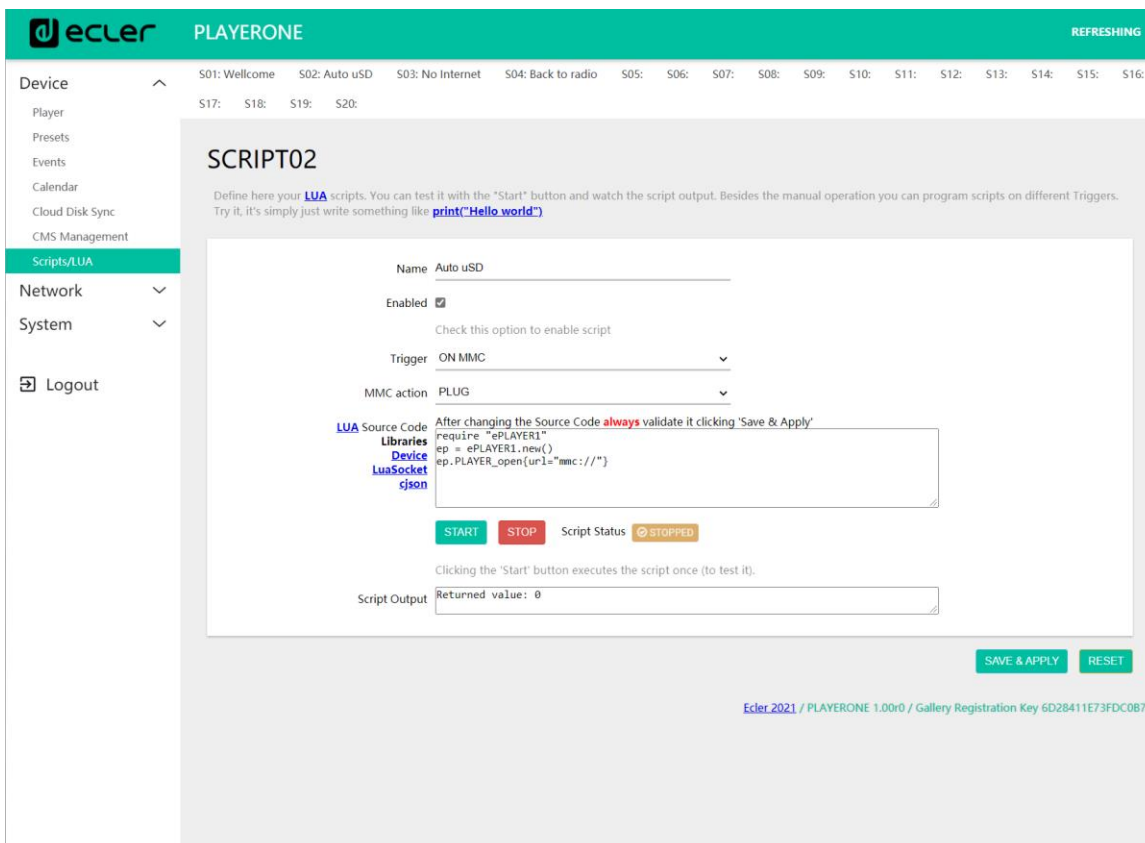
- require "ePLAYER1"
- ep = ePLAYER1.new()
- ep.LCD_print2({line1="Hello",line2="Ecler",center="true",timeout=10})

13.7.2 Example Script02:

Automatically play uSD cards when inserted

There are applications in which the local storage medium will change depending on the user. For example, an activity room in a gym where, for each session, different instructors will change the music. In other words, they will connect their USB device or uSD card to play their content. This can be automated so that users simply have to insert their device into the device, and playback starts automatically. This avoids unauthorised tampering and/or saves them from having to read the manual for the device.

So, the following example *script* automatically plays the uSD card as soon as it is inserted. You could create a *script* to do the same thing with the USB device, by simply changing the trigger and the URL.



The screenshot shows the ecler PLAYERONE interface. The main content area is titled 'SCRIPT02' and contains the following configuration:

- Name: Auto uSD
- Enabled:
- Trigger: ON MMC
- MMC action: PLUG
- LUA Source Code:


```
require "ePLAYER1"
ep = ePLAYER1.new()
ep.PLAYER_open{url='mmc://'}

```
- Script Status: STOPPED
- Script Output: Returned value: 0

Buttons for 'START', 'STOP', 'SAVE & APPLY', and 'RESET' are visible. A footer note reads: 'Ecler 2021 / PLAYERONE 1.00r0 / Gallery Registration Key 6D28411E73FDC087'.

Figure 62

- require "ePLAYER1"
- ep = ePLAYER1.new()
- ep.PLAYER_open{url="mmc://"}

13.7.3 Example Script03:

Messages can be configured for practical uses. The following *script* displays a "NO INTERNET" message that flashes when the Internet (WAN) connection is lost. It has also been written in such a way that we can use the "blink" function multiple times within the same *script*. In addition, it loads *preset1*, which has local audio content stored and set to play (PLAY) when loaded, in order to ensure continuity of the music program.

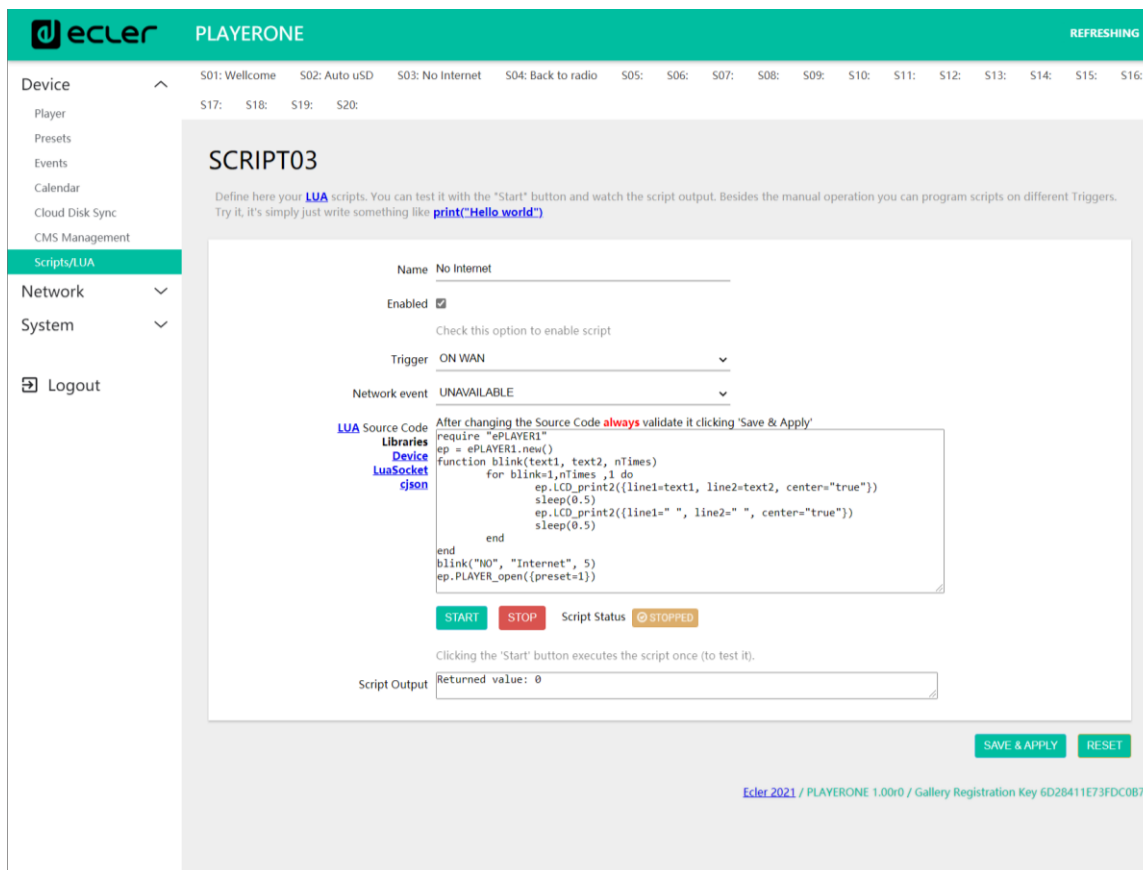


Figure 63

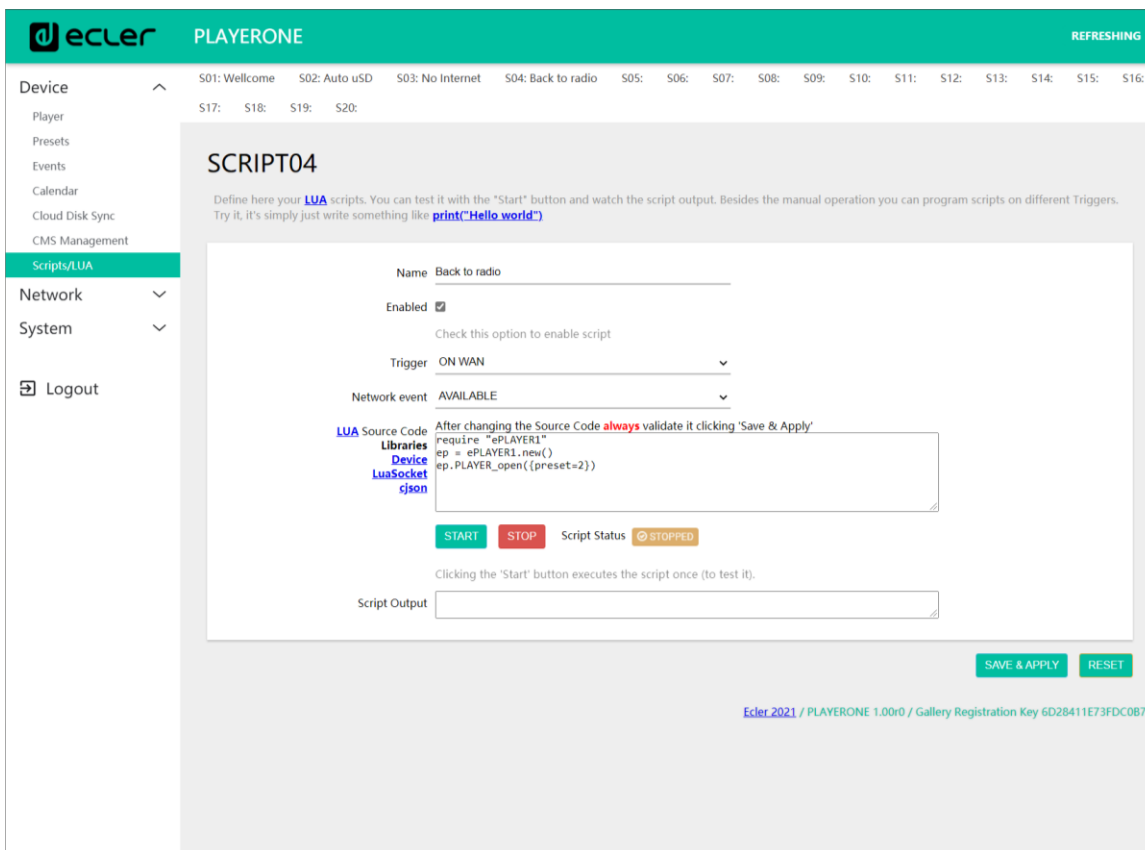
```
require "ePLAYER1"
ep = ePLAYER1.new()
function blink(text1, text2, nTimes)
  for blink=1,nTimes ,1 do
    ep.LCD_print2({line1=text1, line2=text2, center="true"})
    sleep(0.5)
    ep.LCD_print2({line1=" ", line2=" ", center="true"})
    sleep(0.5)
  end
end
blink("NO", "Internet", 5)
ep.PLAYER_open({preset=1})
```

13.7.4 Example Script04:

Play an Internet radio station when the Internet connection is restored

The following *script* loads a *preset* when an available Internet connection (WAN) is detected. It can be good to use this in combination with the silence detection event:

The device is playing an Internet radio station, however, due to a network problem, it no longer has access to the radio. After a few seconds without audio, the silence detection event is triggered, and the device starts playing content from the uSD card (back-up music). However, you want the radio that was playing to start up again automatically once the Internet connection is available again.



The screenshot shows the ecler PLAYERONE interface. The left sidebar contains navigation options: Device, Player, Presets, Events, Calendar, Cloud Disk Sync, CMS Management, Scripts/LUA (highlighted), Network, System, and Logout. The main content area is titled 'SCRIPT04' and contains the following configuration:

- Name: Back to radio
- Enabled:
- Check this option to enable script
- Trigger: ON WAN
- Network event: AVAILABLE
- LUA Source Code:


```
require "ePLAYER1"
ep = ePLAYER1.new()
ep.PLAYER_open({preset=2})
```
- Script Status: STOPPED
- Buttons: START, STOP, SAVE & APPLY, RESET

At the bottom right, there is a footer: Ecler 2021 / PLAYERONE 1.00r0 / Gallery Registration Key 6D28411E73FDC087

Figure 64

```
require "ePLAYER1"

ep = ePLAYER1.new()

ep.PLAYER_open({preset=2})
```


14 NETWORK

Allows you to configure the Ethernet and WiFi network interfaces.

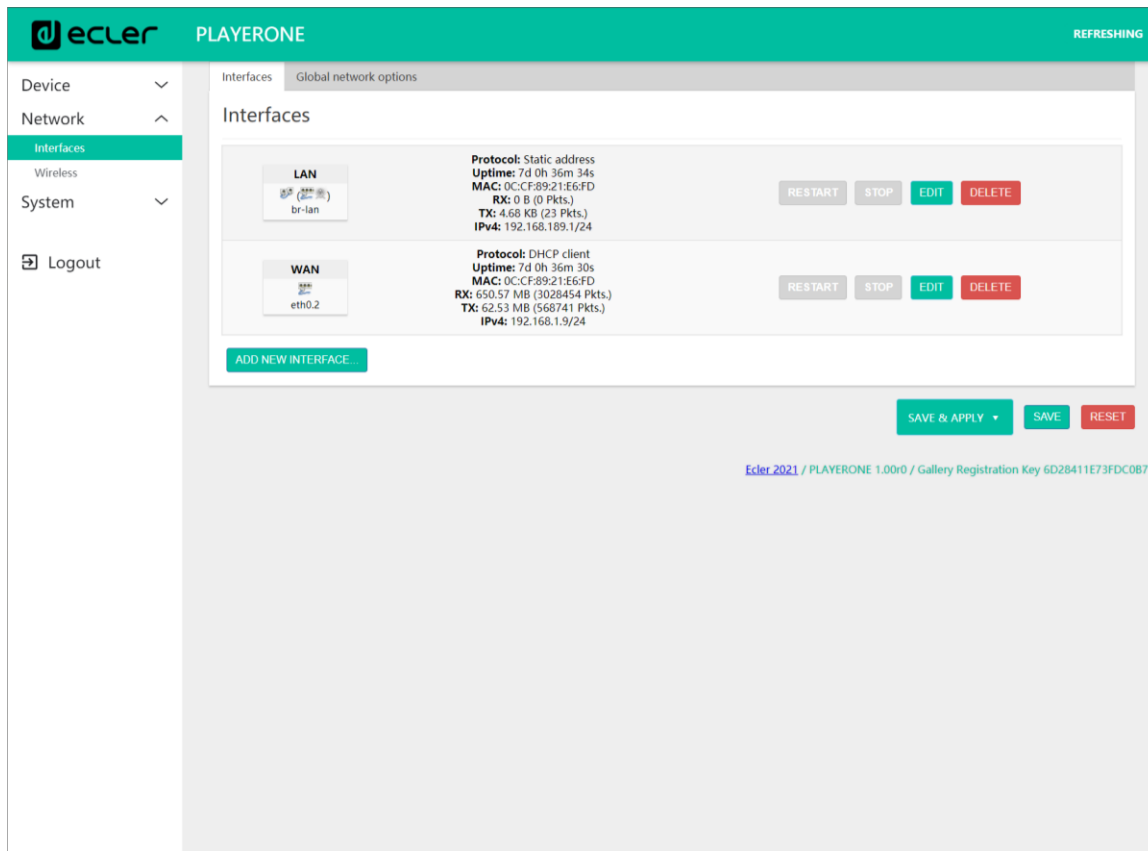


Figure 65: Network, Interfaces

- **Interfaces:** allows parameters to be edited for the wired connection, RJ45 Ethernet port (LAN/WAN).
- **Wireless:** allows parameters to be set for the wireless connection, WiFi interface.

Please Note: for detailed information on all network parameters, click on this [link](#).

14.1 Connection via RJ45 cable

By default, PLAYER ONE/PLAYER ZERO has automatic network addressing (DHCP). To change to manual addressing (editing network parameters), access the web application and go to the Network/Interfaces menu.

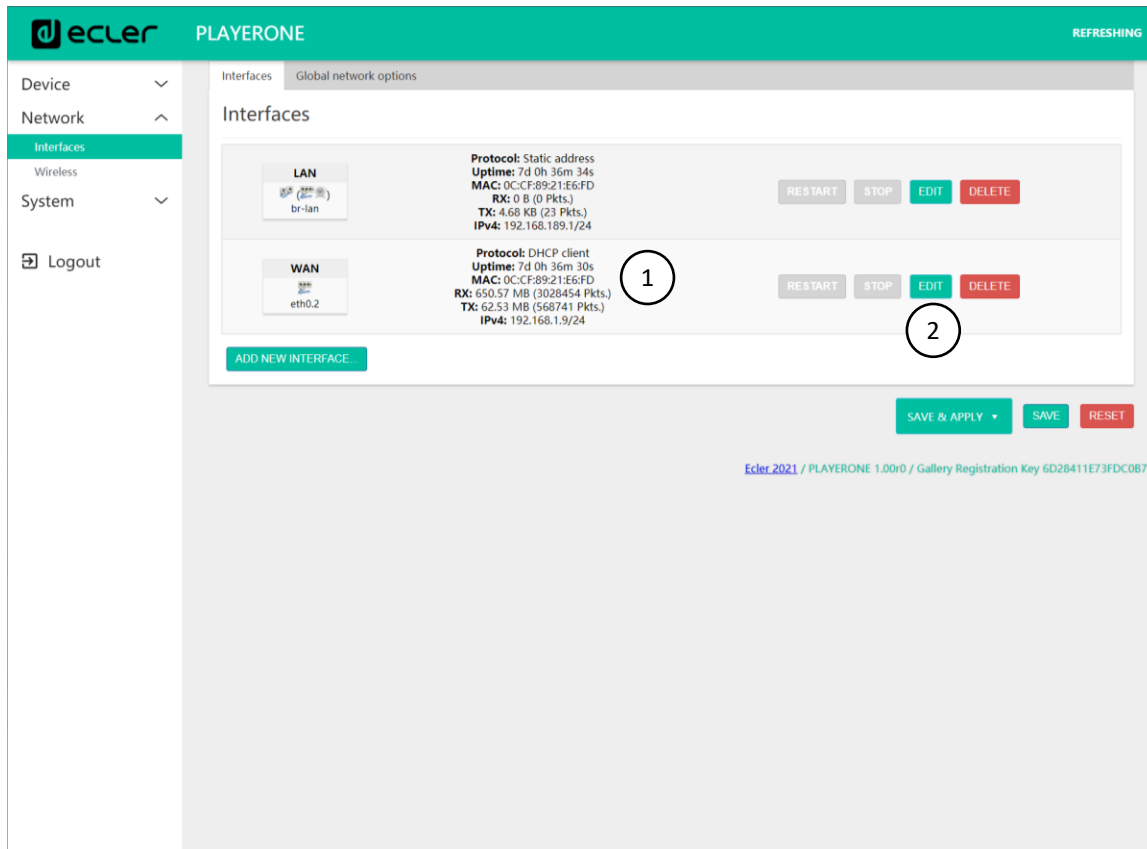


Figure 66

- **Network information:** checking network parameters and network information
 - **Protocol:** DHCP client/static address
 - **Uptime:** connection time
 - **MAC:** MAC address of the device
 - **RX:** amount of data received
 - **TX:** amount of data transmitted
 - **IPv4:** IP address of the device
- **Edit network parameters:** accesses the Ethernet interface configuration

Press the EDIT button to access the Ethernet interface configuration.

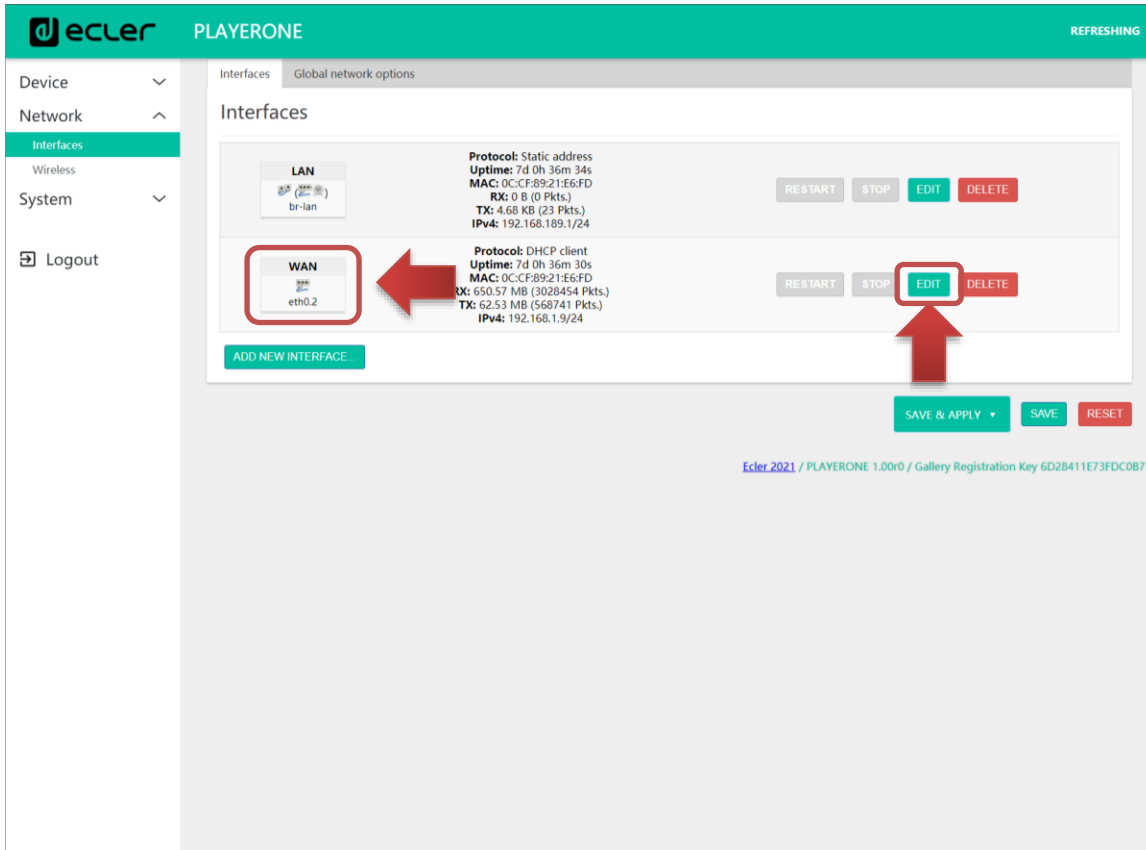


Figure 67: accessing the network configuration

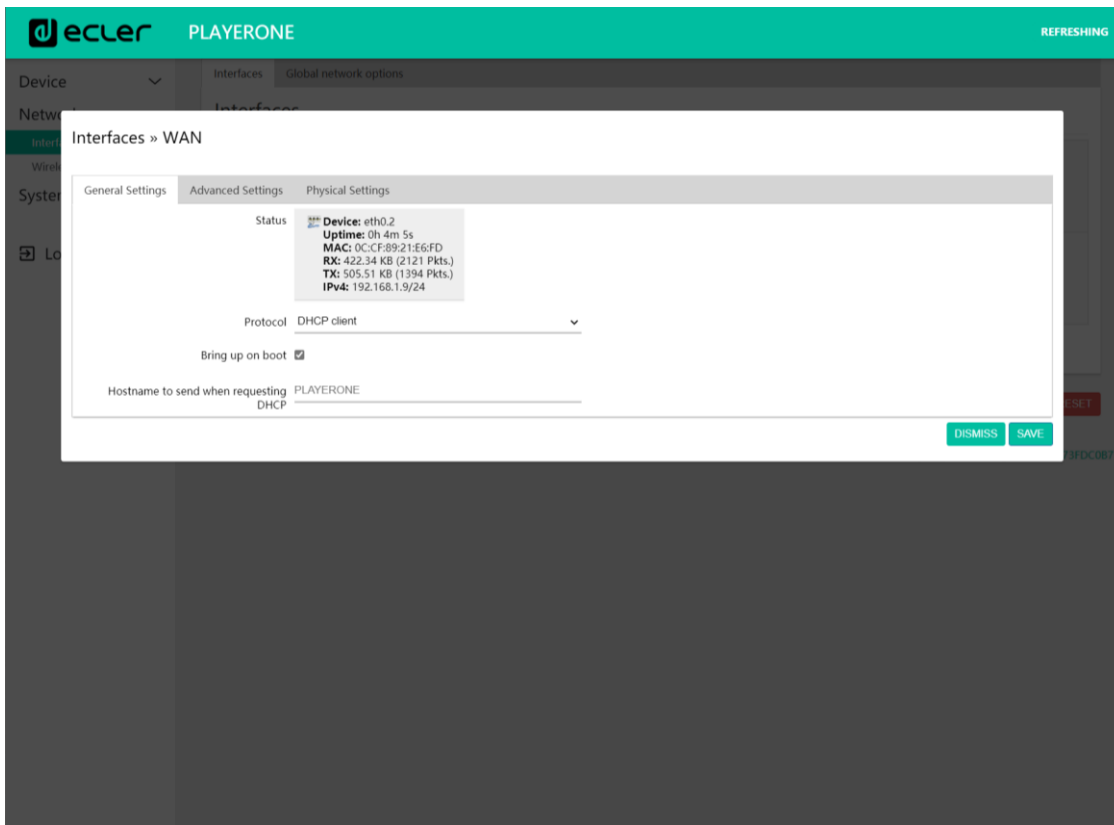


Figure 68: editing network parameters

Select *Static address* mode and press the SWITCH PROTOCOL button to edit the network addressing manually.

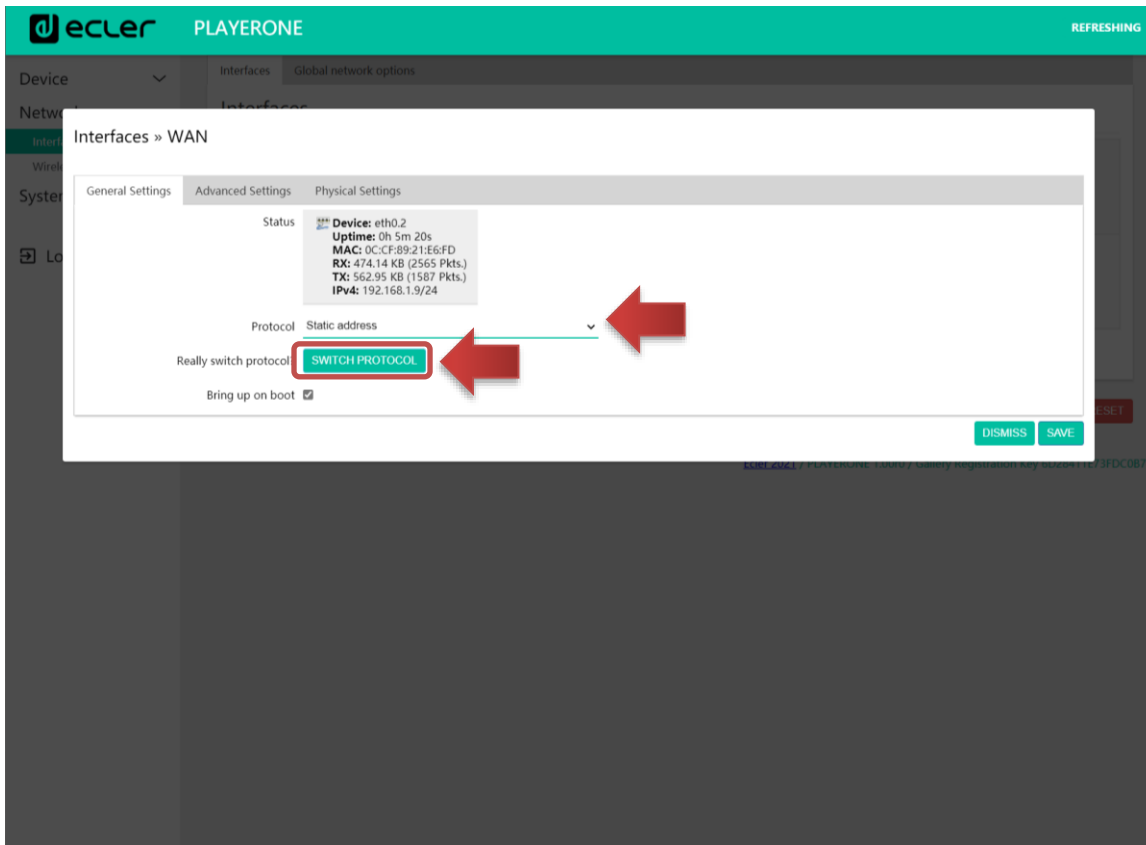


Figure 69

Configure the network parameters according to your infrastructure:

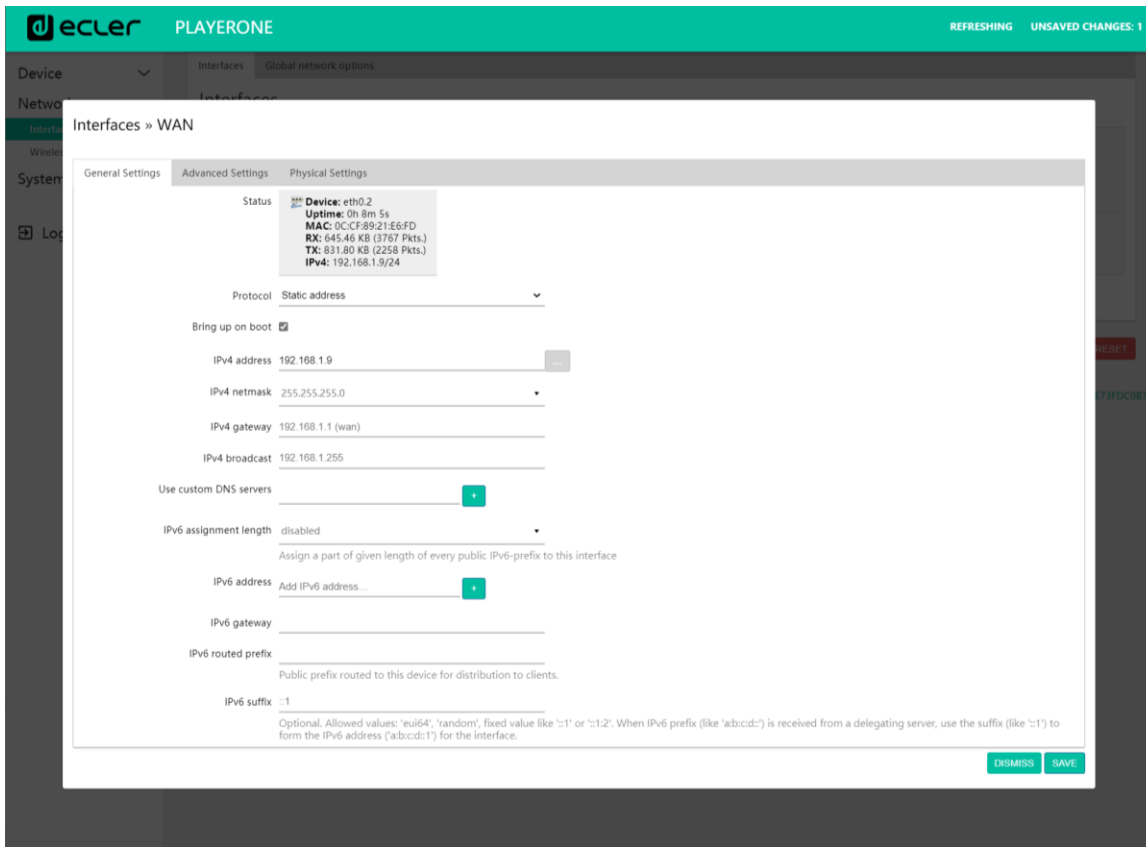


Figure 70

- **IPv4 address:** device network address
- **IPv4 netmask:** subnet mask
- **IPv4:** gateway (switch/router with Internet access)
- **DNS1:** Domain Name System 1 (optional)
- **DNS2:** Domain Name System 2 (optional)

Save the changes on the Edit page, and apply them on the Network/Interfaces page.

14.2 Point-to-point wireless connection

PLAYER ONE/PLAYER ZERO can establish a point-to-point wireless connection with a WiFi interfaced device (PC, *smartphone*, *tablet*, etc.), either for accessing the web application or for *streaming* via AirPlay/DLNA.

Make sure that the wireless connection is enabled and the device is set to MASTER (Access Point) mode.

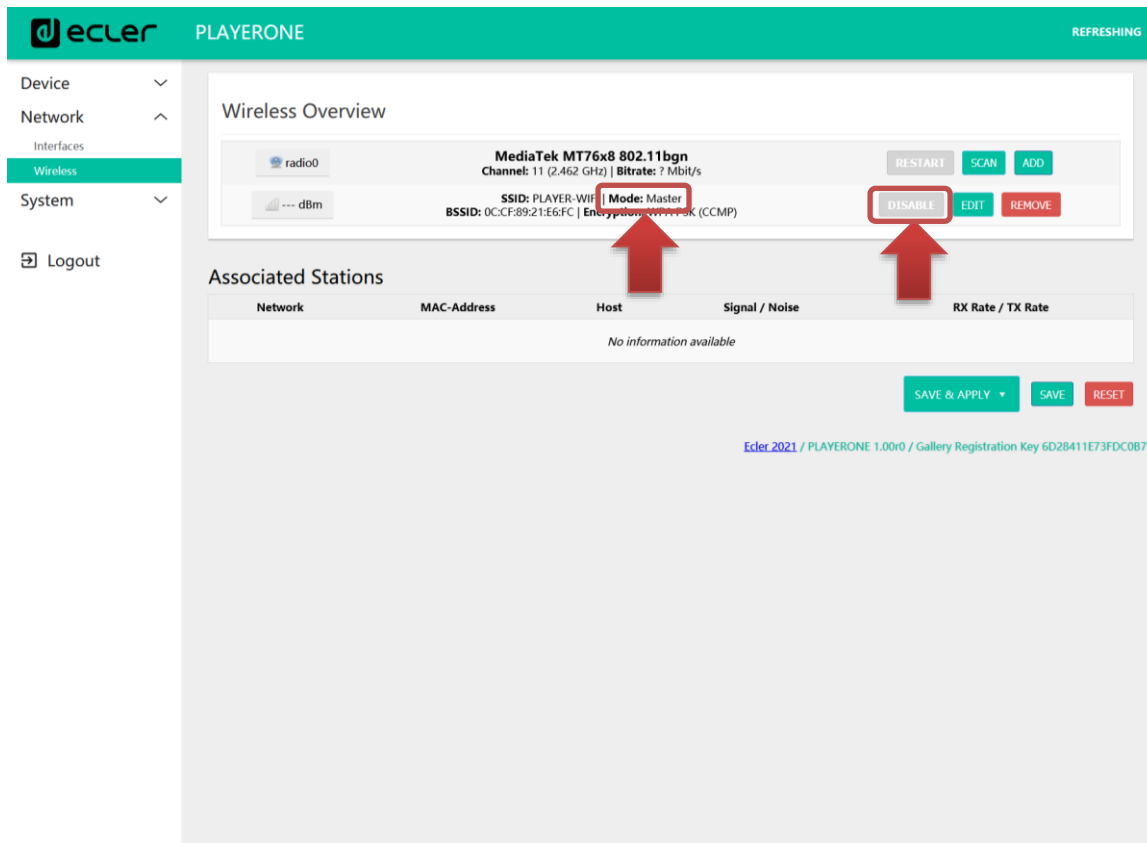


Figure 71: WiFi interface configuration

In your device's WiFi settings, select your device's WiFi network (factory default: PLAYER-WIFI), and enter the password (0123456789). A point-to-point wireless connection will be established.



Figure 72: WiFi settings on a smartphone

If you have more than one Ecler player in the same system, or you simply want to customize the parameters with which the network is displayed, we recommend changing the SSID and password of the WiFi network of your device(s). To do this, click on the EDIT button and adjust the WiFi network parameters according to your needs. When you have made the changes, save and apply them.

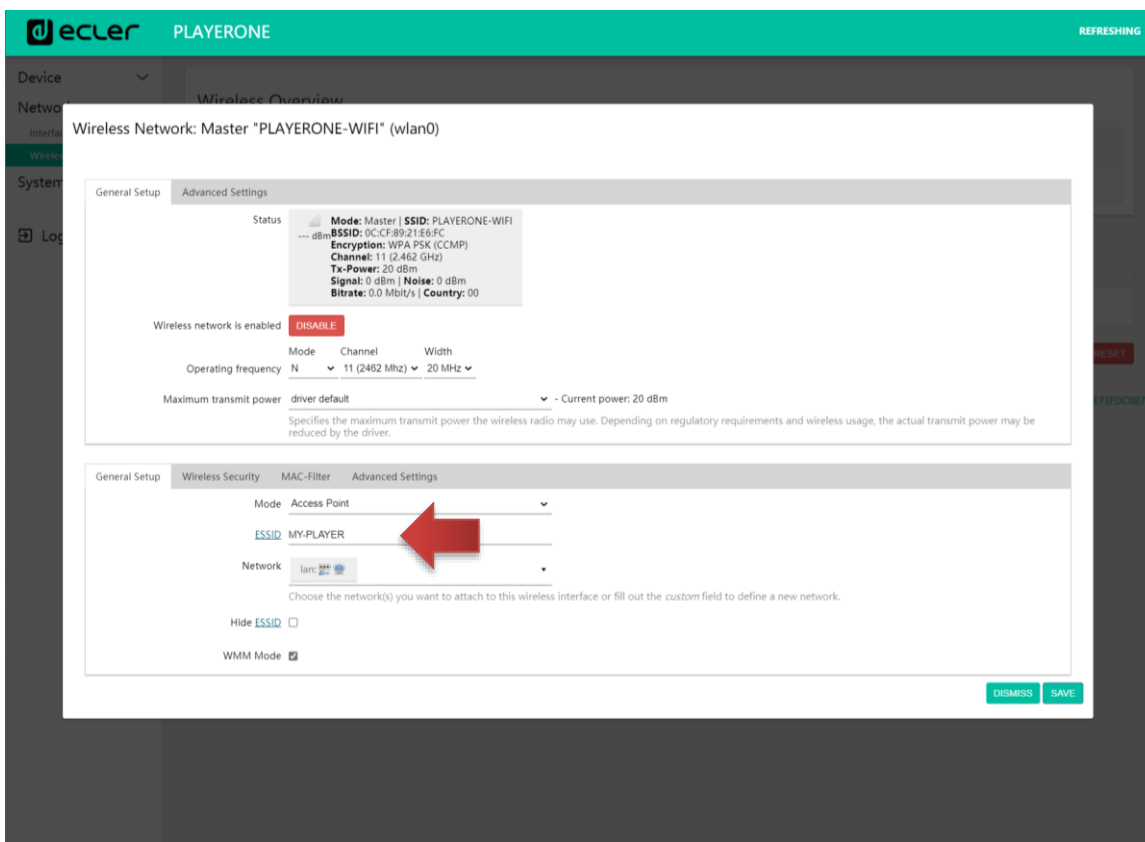


Figure 73: editing WiFi network SSID

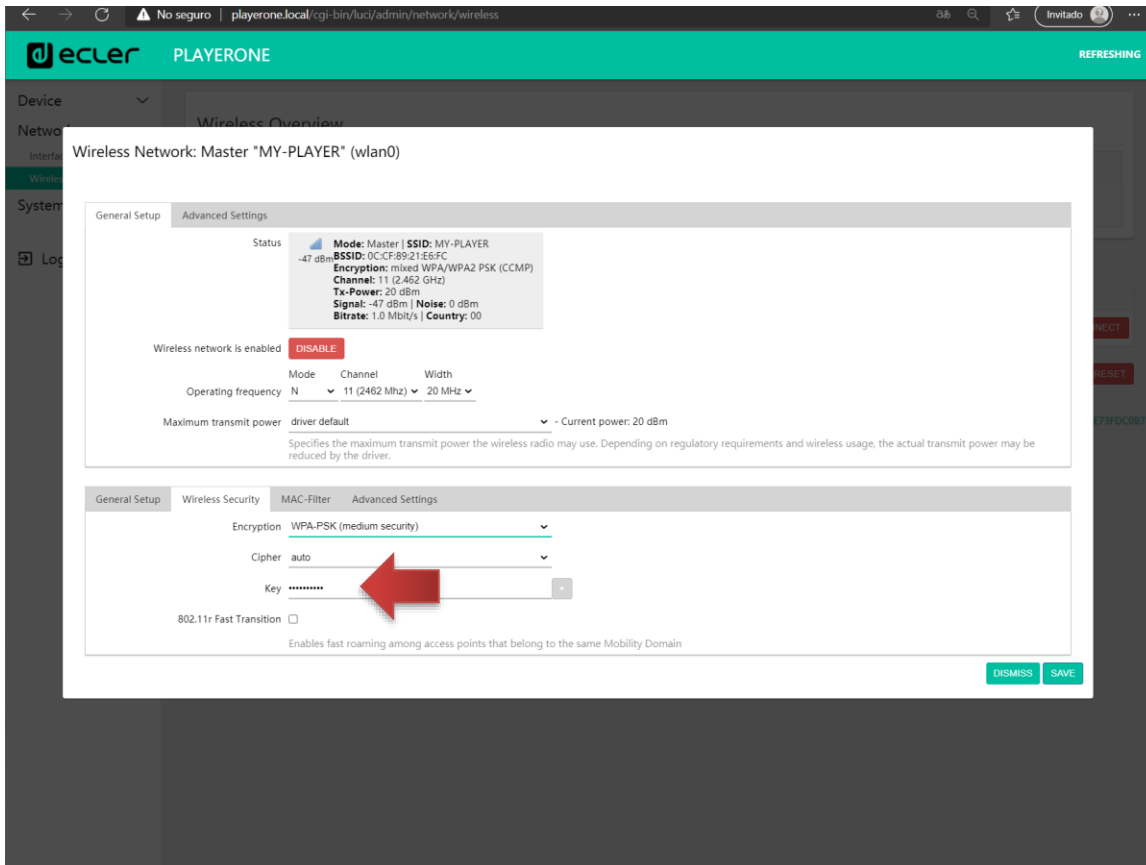


Figure 74: editing WiFi password

14.3 Connecting to a WiFi network

PLAYER ONE/PLAYER ZERO can connect to the Internet via a home WiFi network to access network addresses, such as Internet radios or remote file syncing services, such as Google Drive.

On the WiFi interface configuration page, click SCAN.

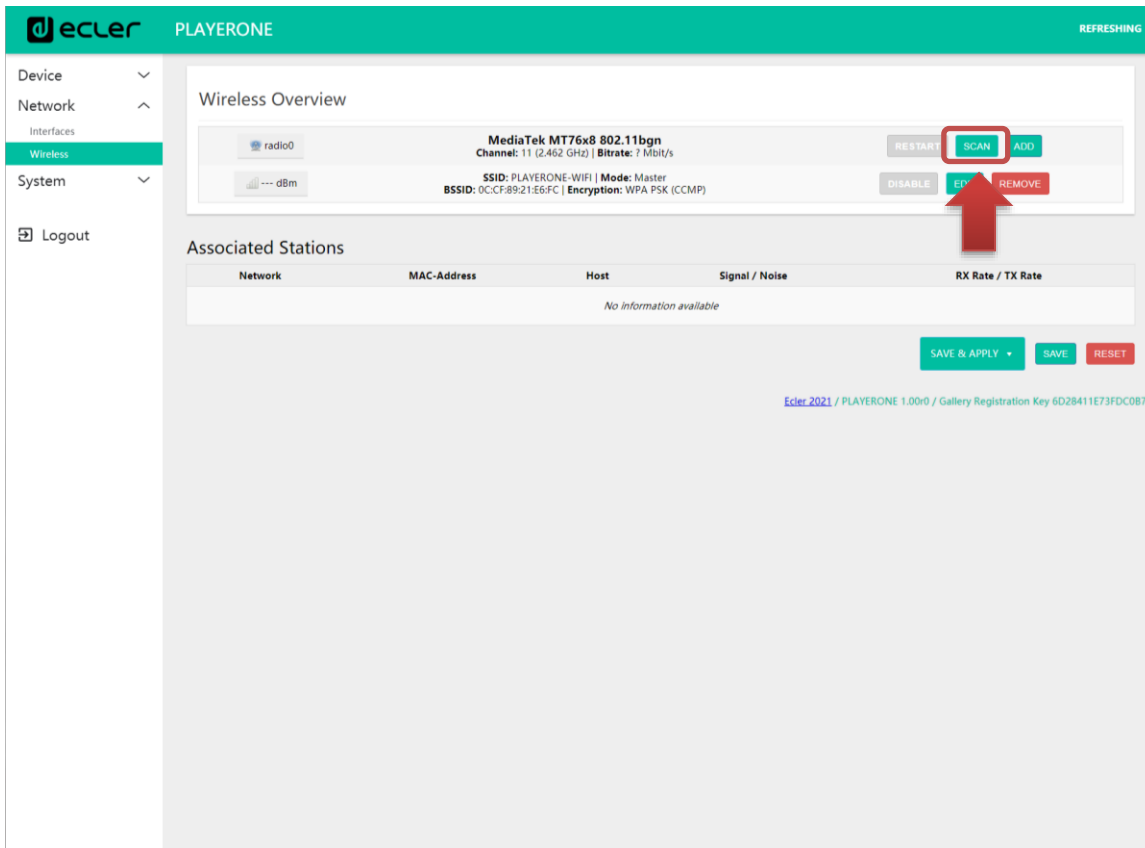


Figure 75

Select your home WiFi network.

Please Note: PLAYER ONE/PLAYER ZERO is only compatible with 2.4GHz wireless networks.

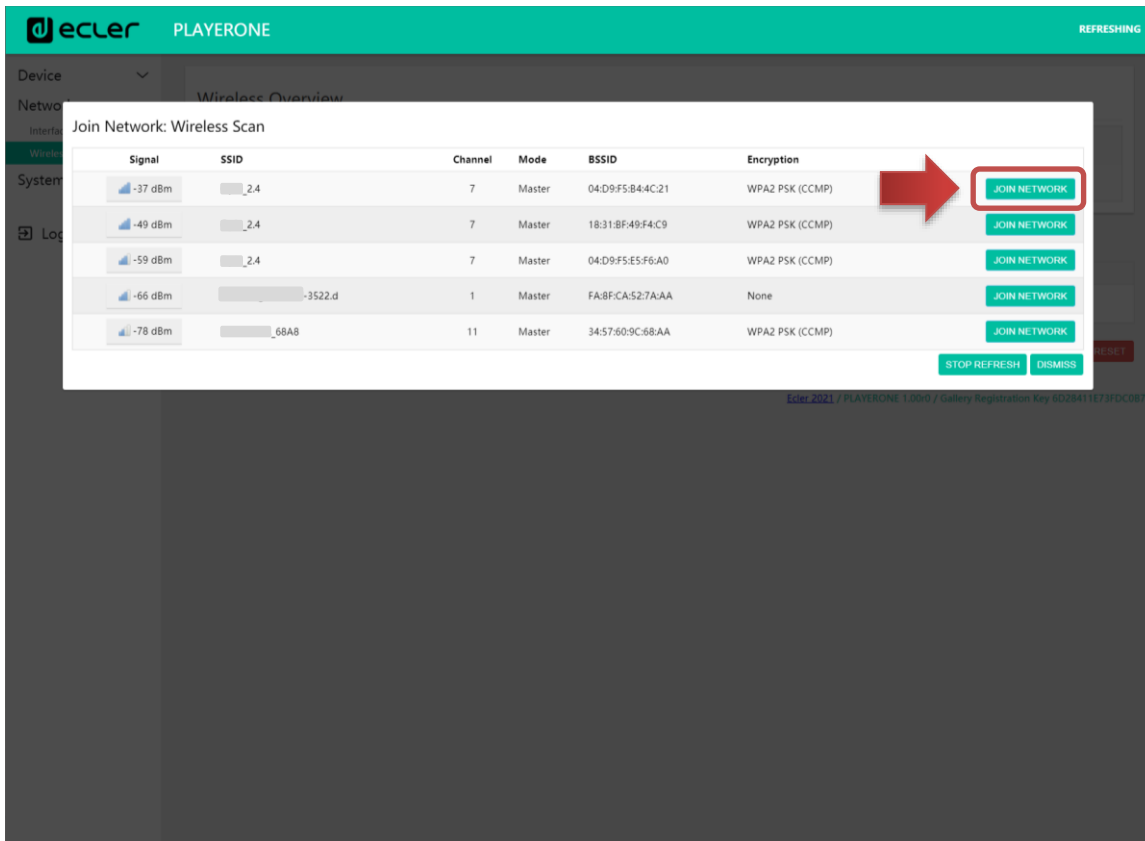


Figure 76

Enter the password of the WiFi network you want to connect to and press the SUBMIT button.

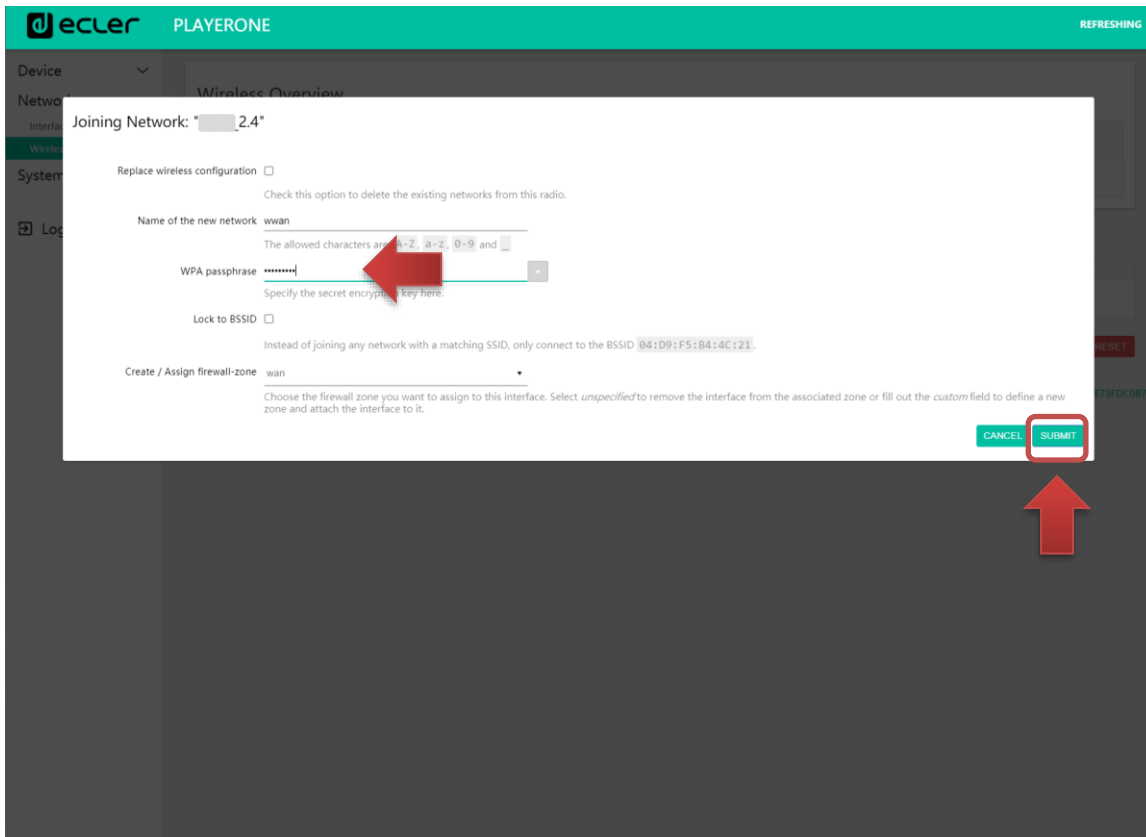


Figure 77

The WiFi network settings are shown below. Press the SAVE button if you do not need to make any additional changes.

Please Note: the operating mode has changed to Client.

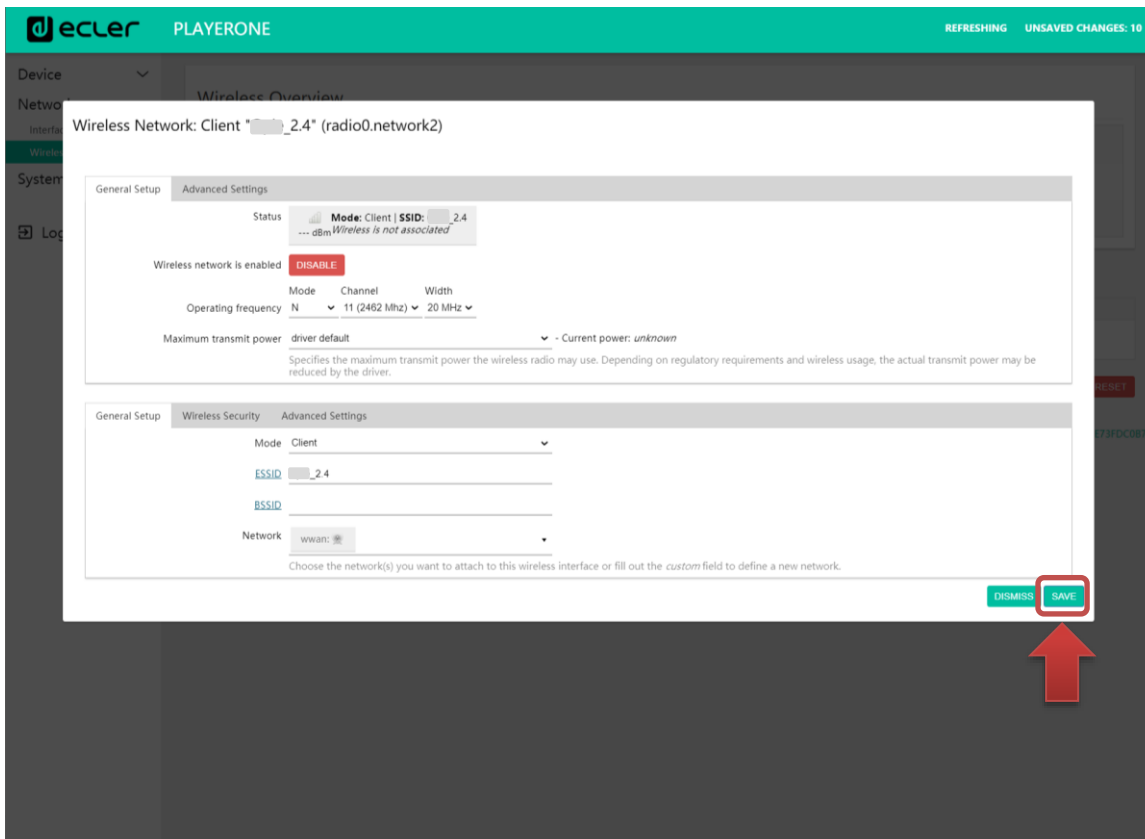
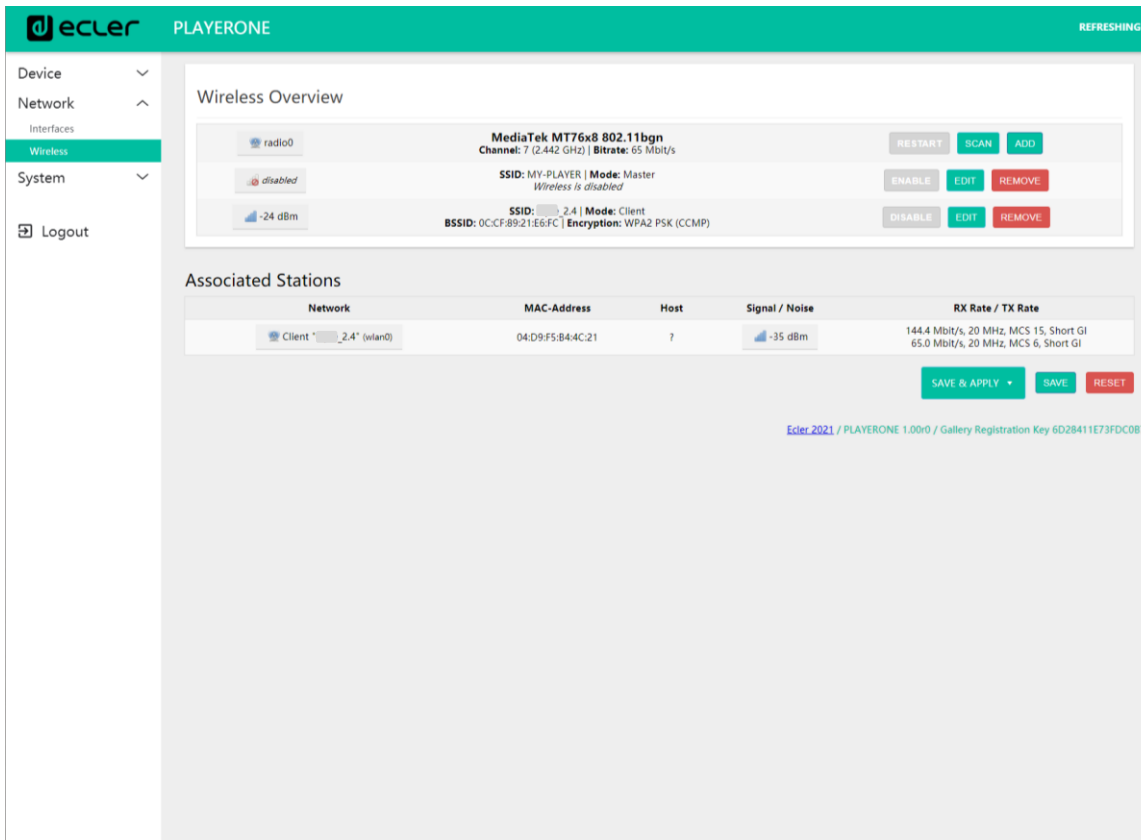


Figure 78

A wireless connection to your WiFi network will be established.



The screenshot displays the ecler PLAYERONE web interface. The top navigation bar includes the ecler logo, the text 'PLAYERONE', and a 'REFRESHING' indicator. A left sidebar contains a menu with 'Device', 'Network', 'Interfaces', 'Wireless', 'System', and 'Logout'. The 'Wireless' section is active, showing a 'Wireless Overview' panel with three entries: 'radio0' (MediaTek MT76x8 802.11bgn, Channel: 7, Bitrate: 65 Mbit/s), 'disabled' (SSID: MY-PLAYER, Mode: Master, Wireless is disabled), and '-24 dBm' (SSID: *_2.4, Mode: Client, BSSID: 0CCF8921E6FC, Encryption: WPA2 PSK (CCMP)). Below this is an 'Associated Stations' table with columns for Network, MAC-Address, Host, Signal / Noise, and RX Rate / TX Rate. One station is listed: Client *_2.4* (wlan0) with MAC-Address 04:D9:F5:B4:4C:21, Host ?, and Signal / Noise -35 dBm. The RX Rate / TX Rate is 144.4 Mbit/s, 20 MHz, MCS 15, Short GI / 65.0 Mbit/s, 20 MHz, MCS 6, Short GI. At the bottom right, there are 'SAVE & APPLY', 'SAVE', and 'RESET' buttons, and a footer with the text 'Ecler 2021 / PLAYERONE 1.00r0 / Gallery Registration Key 6D28411E73FDC087'.

Figure 79

15 SYSTEM

In the Settings menu you can make changes as an administrator, such as changing the device name, or the password for accessing the web application, or local media encryption, resetting the device to factory settings, saving backups, updating firmware, etc.

15.1 Name and Time

Device name and time sync settings.

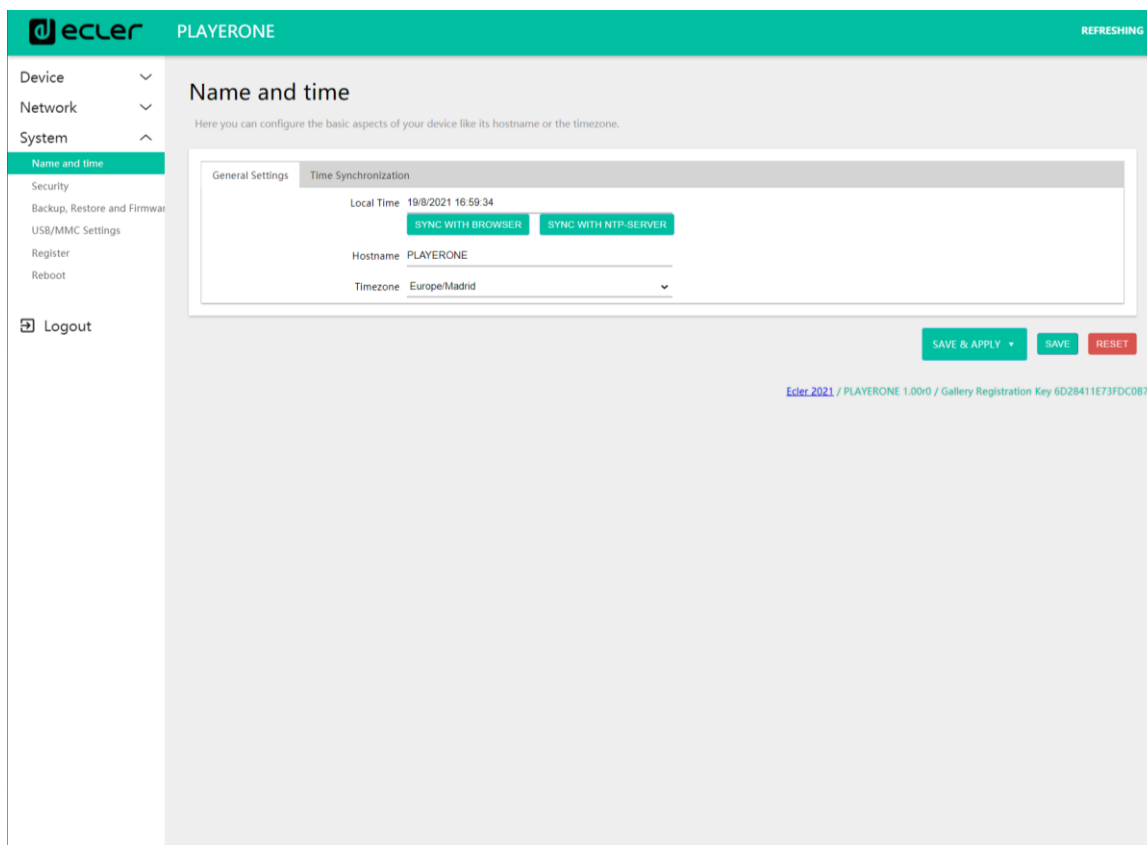


Figure 80: Name and Time, General settings

15.1.1 General settings

- **Local Time:** setting the time on the device. You can sync it with the browser time (Sync with Browser), and/or you can sync it via NTP server: recommended if you have an Internet connection and you are using calendar-triggered events.

Please Note: syncing with an NTP server means you don't have to worry about summer and winter time changes.

- **Hostname:** device name. Name that will be displayed on other services such as AirPlay, RePLAYER control application or mDNS. By default, this is "playerone" for PLAYER ONE and "playerzero" for PLAYER ZERO. Enter "devicename.local/" in your browser's search bar to access the device's web GUI: playerone.local/ or playerzero.local/ by default.
- **Timezone:** setting the time zone. It is important to set this correctly if you are managing calendar events.

15.1.2 Time synchronization

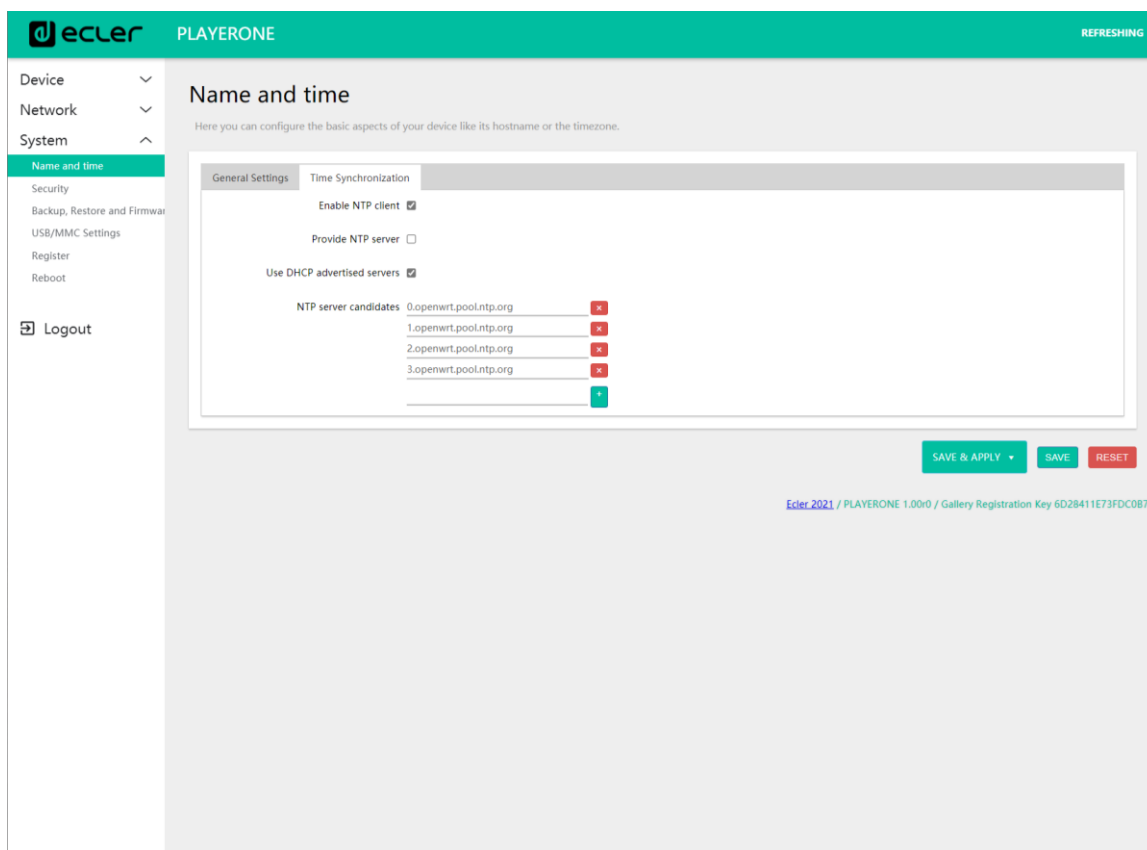


Figure 81: Name and time, Time synchronization

Enable/disable time syncing with an NTP server. This also allows you to manage the candidate servers for syncing.

Please Note: PLAYER ONE/PLAYER ZERO has an internal clock that means that you won't lose the time setting if it is not connected to the mains or not synced with an NTP service. However, please note that this clock has an accuracy of ± 1 minute/month.

15.2 Security

On this page, changes can be made to prevent unauthorized personnel from tampering with the player.

15.2.1 Web password

Password for accessing the web application. By default, **ecler**.

Please Note: the user is always **root**. This cannot be changed.

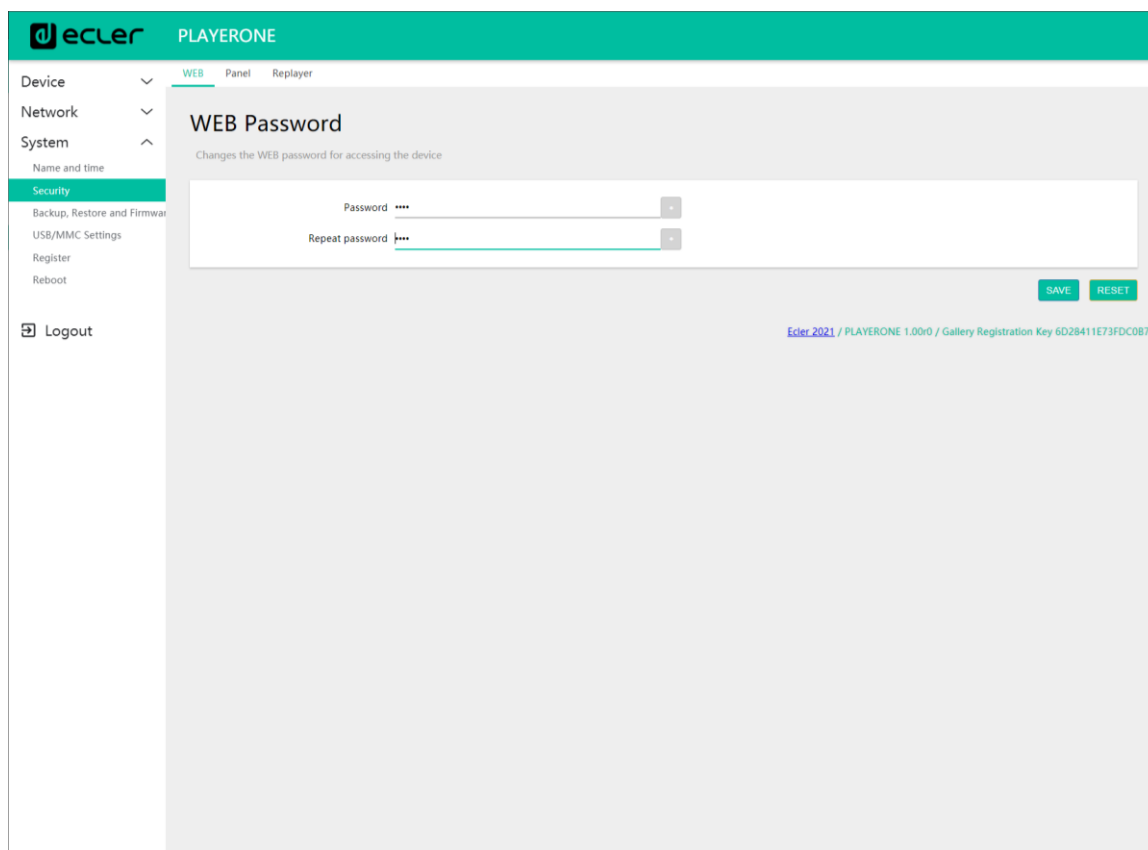


Figure 82

15.2.2 Front panel functions

Setting the front panel button lock to prevent unauthorised tampering or changing of administrator settings. **Only available on PLAYER ONE.**

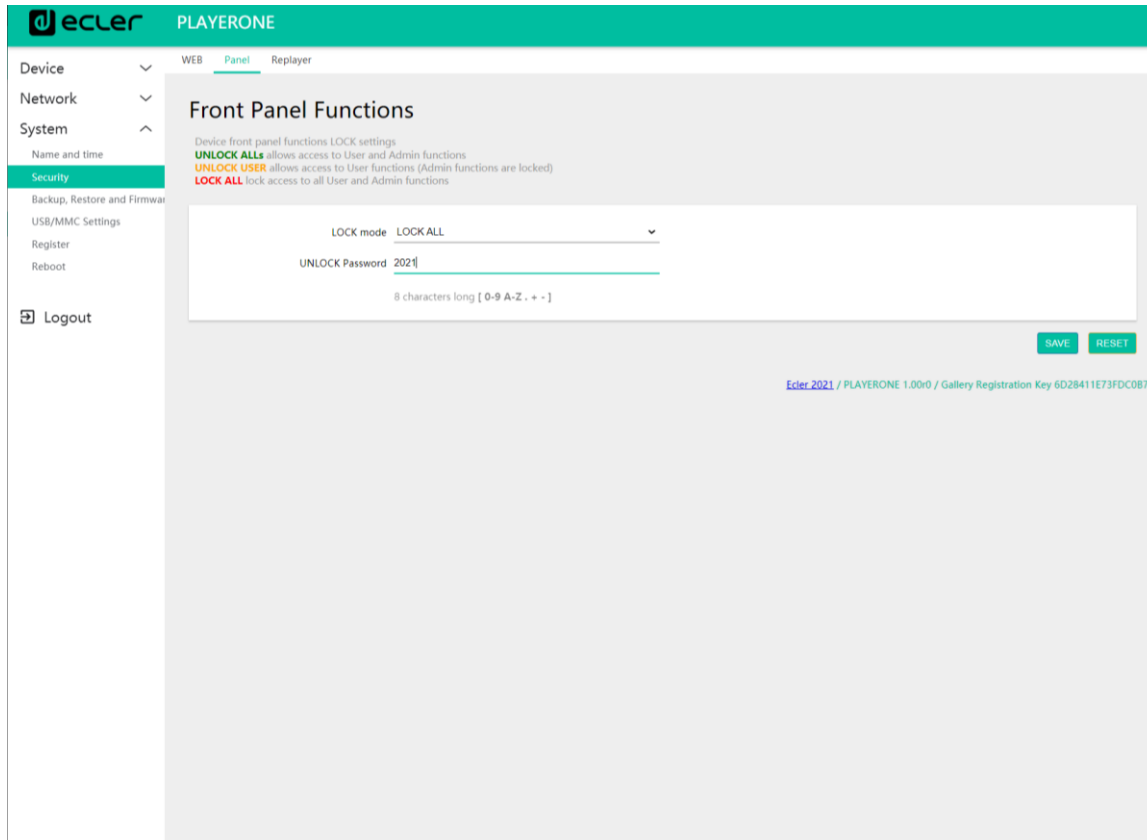


Figure 83

- **LOCK mode:** selection of the lock mode
 - **UNLOCK ALL:** unlocks all buttons on the front panel.
 - **UNLOCK USER:** locks access to the device configuration menu (administrator functions), allowing the use of all other functionalities.
 - **LOCK ALL:** locks all buttons on the front panel.
- **UNLOCK Password:** Password for unlocking. If the device is in lock mode, once you have entered the password, you will have a window of time in which to use the device. After you stop using it, the device will automatically lock the controls again.

15.2.3 RePlayer

4-digit PIN for accessing the device via the RePLAYER control application. Click on this [link](#) to find all the information and documentation related to the control application.

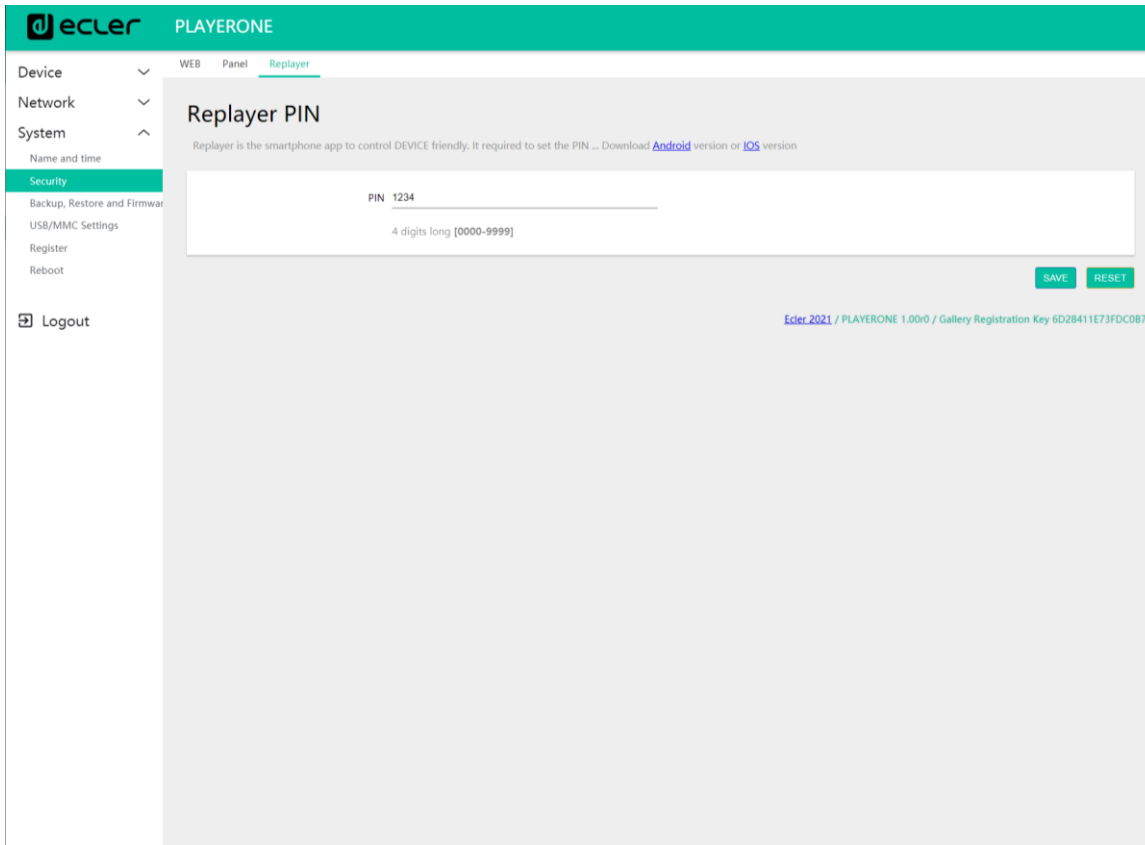
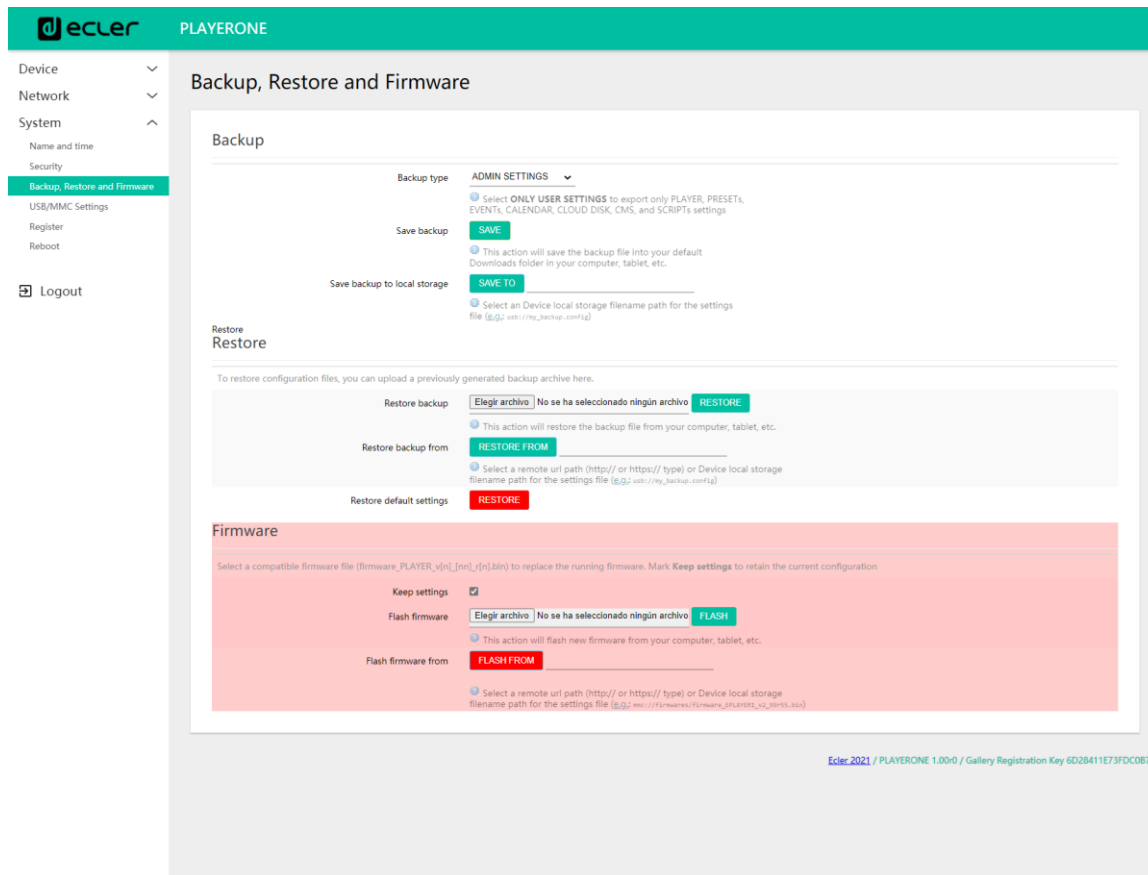


Figure 84

15.3 Backup, Restore and Firmware

Manage backups of your device and retrieve configuration files, and also update the firmware version.



Backup, Restore and Firmware

Backup

Backup type: ADMIN SETTINGS

Save backup: SAVE

Save backup to local storage: SAVE TO

Restore

Restore backup: Elegir archivo No se ha seleccionado ningún archivo RESTORE

Restore backup from: RESTORE FROM

Restore default settings: RESTORE

Firmware

Keep settings:

Flash firmware: Elegir archivo No se ha seleccionado ningún archivo FLASH

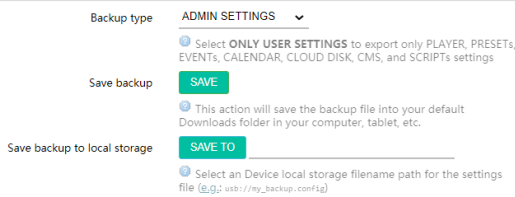
Flash firmware from: FLASH FROM

Ecler_2021 / PLAYERONE 1.00+0 / Gallery Registration Key 6D28411E73FDC087

Figure 85: System. Backup, restore and firmware

15.3.1 Make backups

Backup



Backup type: ADMIN SETTINGS

Select ONLY USER SETTINGS to export only PLAYER, PRESETS, EVENTS, CALENDAR, CLOUD DISK, CMS, and SCRIPTS settings

Save backup: SAVE

This action will save the backup file into your default Downloads folder in your computer, tablet, etc.

Save backup to local storage: SAVE TO

Select an Device local storage filename path for the settings file (e.g.: usb://my_backup.config)

Figure 86: Backup

- **Backup type:** type of backup.
 - **ADMIN SETTINGS:** saves all settings (both administrator and user settings).
 - **USER SETTINGS:** only saves the user settings.
 - **GALLERY SETTINGS:** saves the settings that are compatible with the Ecler Gallery platform.

	ADMIN SETTINGS	USER SETTINGS	GALLERY SETTINGS
Player	✓	✓	✓
Presets	✓	✓	✓*
Calendars	✓	✓	✗
Events	✓	✓	✓
Scripts	✓	✓	✓
Google Drive	✓	✗	✓
Store and Forward	✓	✗	✓
Front panel lock settings	✓	✗	✗
Network settings	✓	✗	✗
System: Name, Time & NTP	✓	✗	✗
Player Profile	✓	✗	✗
Encryption	✓	✗	✗
Web password	✓	✗	✗

*Partial: no playlist

Figure 87: types of backup or configuration files

- **Save Backup:** generates a backup that will be stored in the download folder which is configured in your browser.
- **Save backup to local storage:** generates a backup that will be saved to the specified local storage address, with the specified name. For example, "mmc://backups/copy1.config" (example of a folder on a uSD card inserted in the player)

15.3.2 Restore backups and factory settings

Restore

To restore configuration files, you can upload a previously generated backup archive here.

Restore backup No se ha seleccionado ningún archivo

This action will restore the backup file from your computer, tablet, etc.

Restore backup from

Select a remote url path (http:// or https:// type) or Device local storage filename path for the settings file (e.g.: usb://my_backup.config)

Restore default settings

Figure 88: Restore backup

- **Restore backup:** restores a configuration file (or backup) hosted on your computer, *tablet*, server, etc.
- **Restore backup from:** restores a configuration file hosted on one of the device's storage media, USB or uSD. You can also restore a file hosted in a remote location, URL address.
- **Restore default settings:** restores the device to factory defaults. Administrator and user settings will be lost.
- **Please Note:** if encryption is being used and you restore it to factory settings or restore an administrator file, the content on the storage media will not be able to be recovered.

15.3.3 Firmware Update

Firmware

Select a compatible firmware file (firmware_PLAYER_v[n]_[nn]_[r][n].bin) to replace the running firmware. Mark **Keep settings** to retain the current configuration

Keep settings

Flash firmware No se ha seleccionado ningún archivo

This action will flash new firmware from your computer, tablet, etc.

Flash firmware from

Select a remote url path (http:// or https:// type) or Device local storage filename path for the settings file (e.g.: mmc://firmwares/firmware_EPLAYERS1_v2_09f55.bin)

Figure 89: Firmware

- **Keep Settings:** keeps the device's current settings. If you want to return to factory settings after updating the firmware, deselect this box. It is selected by default.
- **Flash Firmware:** updates via a firmware file hosted on your computer, *tablet*, server, etc.
- **Flash Firmware from:** updates via a firmware file hosted on one of the device's storage media, USB or uSD. You can also use a file hosted in a remote location with a URL address.

15.4 USB/MMC Settings

This allows you to view the space used on the local storage media and manage the encryption function, if desired: for reasons of **security**, data **protection**, to protect the music content stored on the uSD card or USB device, etc. In this way, if the local storage media is removed without authorisation, the data will still be protected, as it will only be readable by the device that encrypted it.

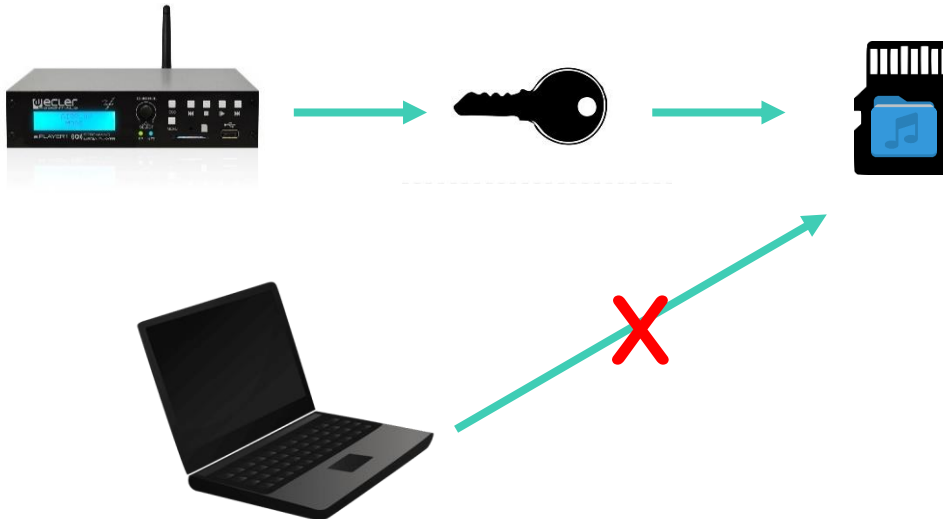


Figure 90: encryption concept

It **operates** as follows:

1. The device encrypts the media, formatting the external drive. This process **deletes all content** from the memory of the uSD or USB device.
2. This process **may take several minutes**. "FORMATTING..." is displayed on the LCD screen for the duration of the process (PLAYER ONE only).
3. At the end of the process, "NO FILES" will be displayed on the LCD screen, indicating that it has been **formatted correctly** and there are no audio files on the external media (PLAYER ONE only).
4. The external drive is now ready to download content using the "**Cloud disk sync**" tools - **Store and Forward (rsync)** and **Google Drive** -.

Please Note: once this feature is disabled, if you re-enable it, you will need to restart the process described above.

Some **considerations** with regards to the encryption function:

- The encryption process is **destructive**, as formatting the external drive deletes all the files on it.
- Once this function is enabled, any **other** external **device** (uSD or USB) that is inserted into the corresponding port, and which **does not contain the** encryption keys, **will be encrypted** and therefore all files will be deleted. As a result, it will not be able to play content other than that which is hosted on the media that the device itself has encrypted.
- The external storage media **cannot be read by any other device** (including other PLAYER ONE/PLAYER ZERO) that does not have the encryption keys.
- The content can be read by the device that encrypted it, **as long as** the encryption option is **not changed**. That is, if the encryption function is disabled, the device will stop using this function and therefore will be able to read any external storage media. However, it loses the keys to the previously encrypted external media, and you will therefore not be able to play the content stored on them.

Please Note: encryption is a tool that is **designed to be used in conjunction with other** content syncing tools: "Store and Forward" (rsync) and "Google Drive". The external storage media, as it is encrypted, cannot be read or written to (copy contents) by any device, such as a computer. Therefore, the only device capable of copying content onto the media is the player that encrypted it. The "Store and Forward" (rsync) and "Google Drive" tools make this possible.

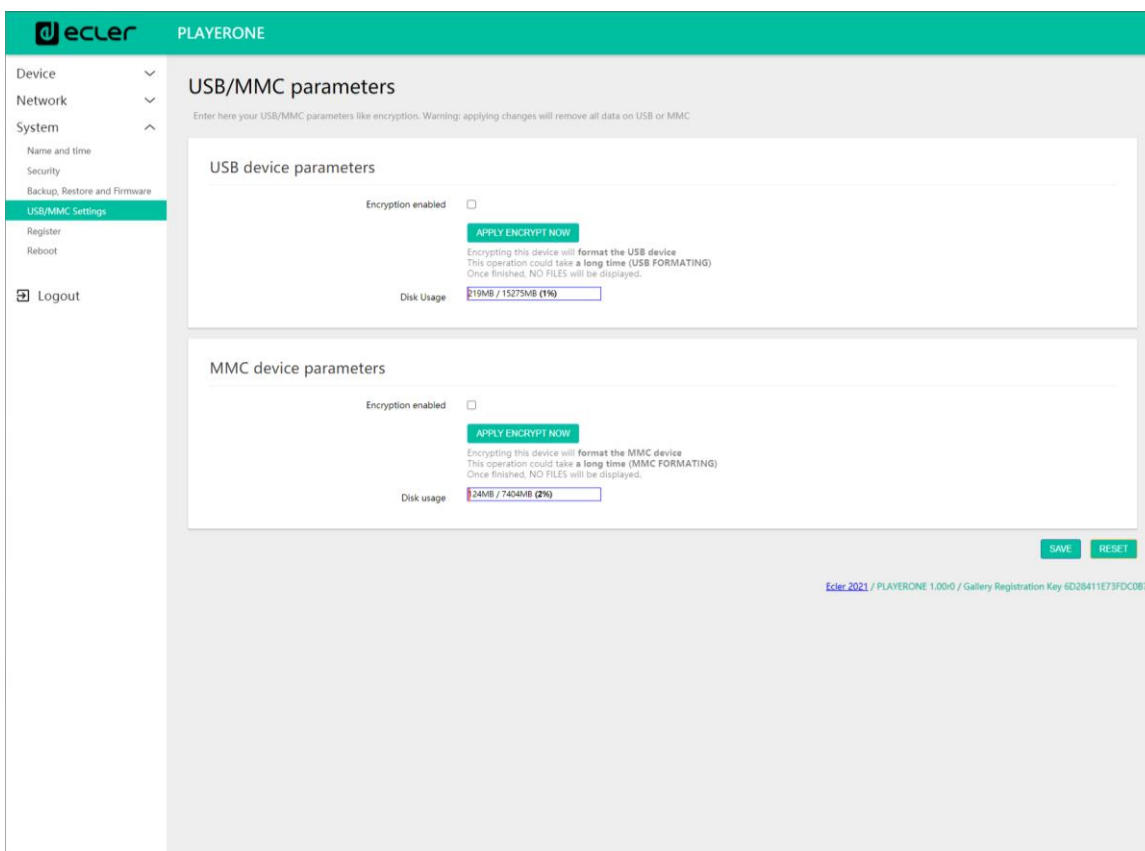


Figure 91: USB/MMC Settings

- **Encryption enabled:** enables/disables media encryption, USB or uSD. If you save this setting, the device will encrypt the disk the next time the drive is mounted, either when the media is inserted or when the device is rebooted.
- **Apply Encrypt now:** immediately encrypts the media.
- **Disk Usage:** capacity of the storage media in MB, and percentage of the storage media that is in use (%), uSD card and USB device respectively. If no media is found, "NO DISK" will be displayed on the LCD screen (PLAYER ONE only)

15.5 Register

The **register** allows you to keep a **detailed record of the device's activity**. This information can be useful for troubleshooting, monitoring device activity, checking for correct programming, etc.

The **lines in the register** contain information about actions performed by the device, errors and/or announcements, along with time information (when different events occur). The register page displays a list of the register lines reported by the device.

The device register is updated daily, and also when it is rebooted. The previous register lines are deleted. However, a copy of each day's register can be kept on local storage media. This feature allows you to view the register for several days.

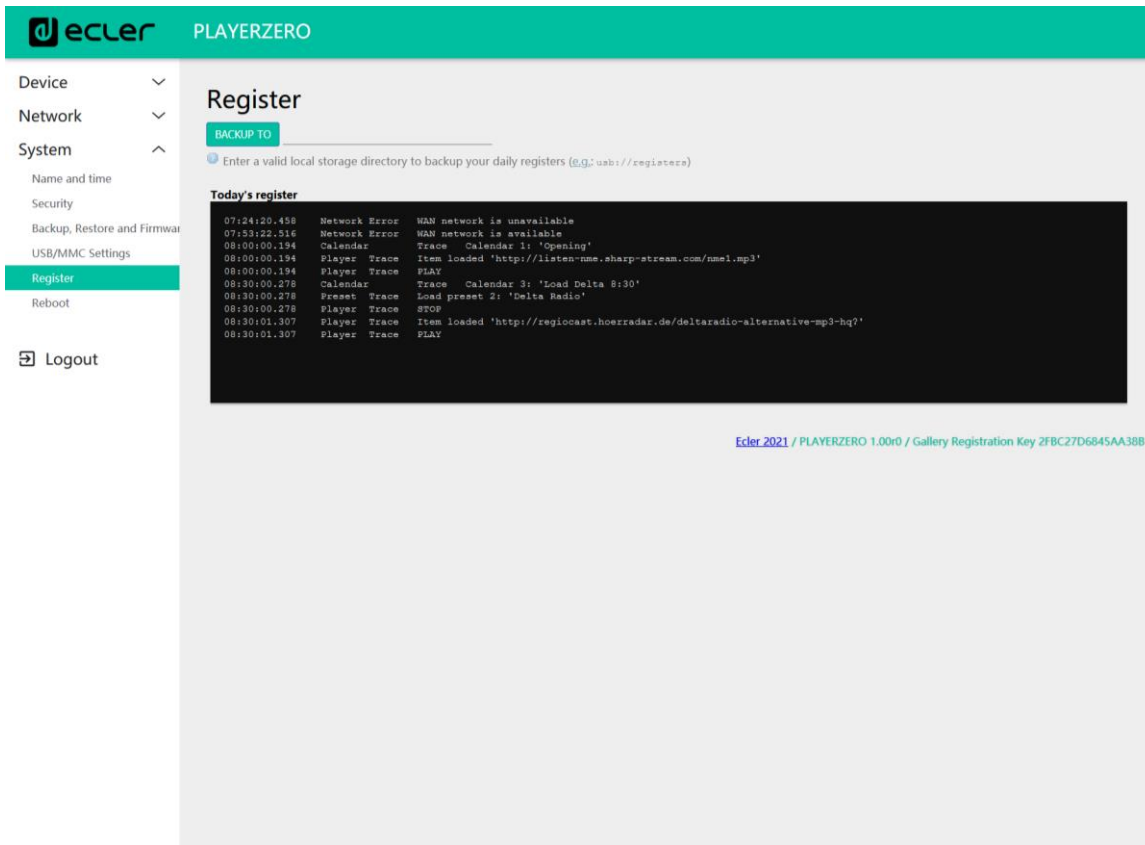


Figure 92: Register

- **Backup to:** allows you to save a copy of the register file for each day to a local storage medium. The path must be specified ("usb://registers", for example).

15.6 Reboot

Allows you to reboot the device from the web application. Click the PERFORM REBOOT button to force the player to reboot.

16 Annex I: SSH server configuration for Store & Forward

The Cloud Disk Sync module, Store and Forward, on the PLAYER ONE/PLAYER ZERO allows the device to download remote audio content to local storage media (USB/uSD). When active, it daily checks a remote location that hosts audio content, compares this content with the current content stored on its local media (USB/uSD) and if necessary (if differences are detected), synchronises the local content so that it is an exact copy of the remote content. This is a safe method of playing content with the device during working hours (during the day), storing it on a local media without the risks associated with real-time streaming.

Store and Forward utility for synchronizing music content remotely uses rsync (Remote Sync) tool for it.

In this guide, it is shown an example of how to configure a **SSH server using Linux** (Ubuntu Desktop 18.04.2 LTS). It is important, that all machines, server and clients, are connected to the same network (LAN/Intranet).

Important note: for configuring Store and Forward cloud-based, would be necessary to hire a VPS (Virtual Private Server) service in order to get a Public IP and access to the SSH server over the Internet.

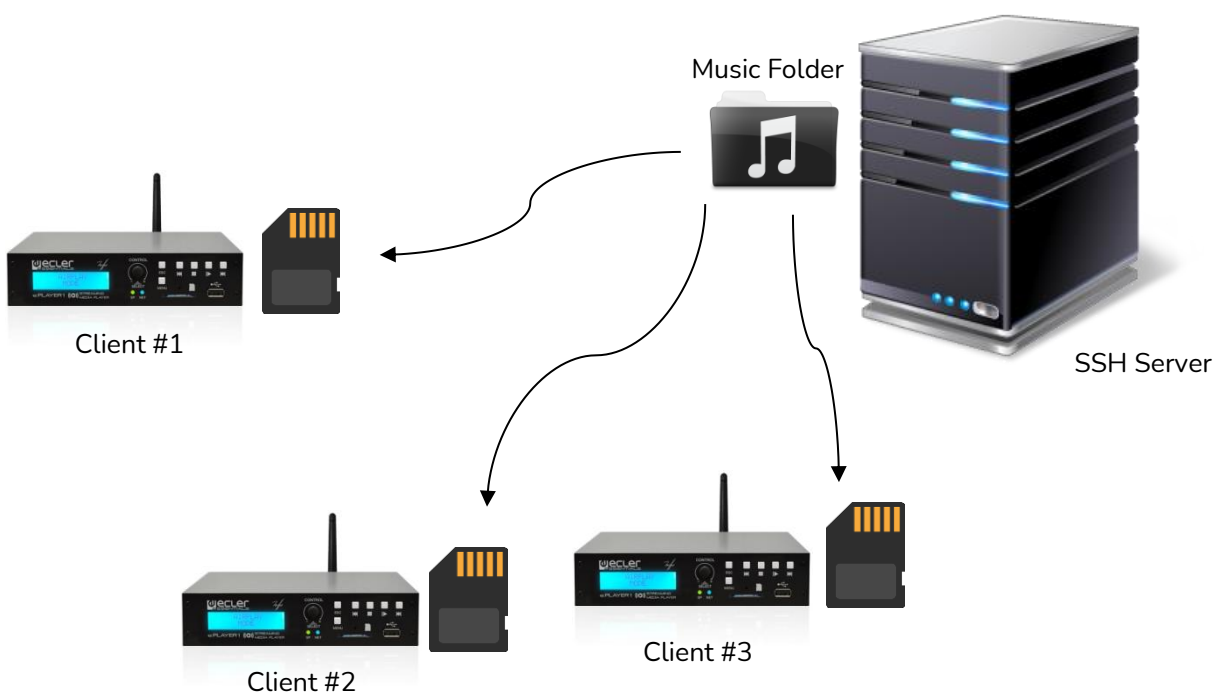


Figura 93: S&F synchronization

16.1 Content groups

A content group is a group of devices that synchronize the same audio content using the Store and Forward utility. It is necessary to create a different user for each content group. Thus, a device assigned to a content group can access only to the content assigned to that group, and not to other content. This procedure states for security. Each content group will manage its private key to access assigned content in the server that hosts all the music, advertisements, voice messages, etc.

Each content group, or user, can have associated multiple connections at the same time. The maximum number of simultaneous connections will depend on the power of the hardware (server).

So, we will be creating as many content groups, or users on Linux, as contents (music folder e.g.) we want to manage.

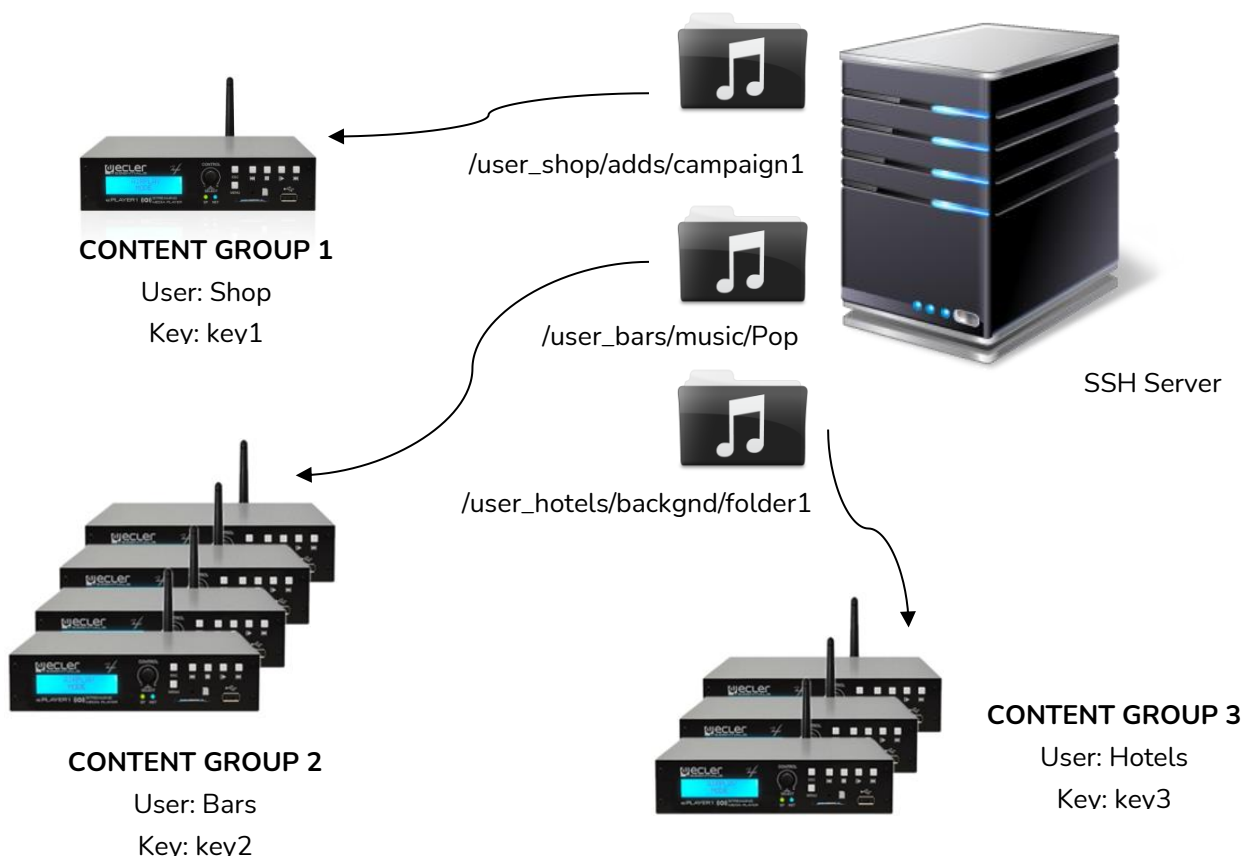


Figura 94: content groups

For a simple configuration, it is possible to create a single user, so, every device can access to its content using the same user and key. **It is skipped security level in this configuration.** A skilled user could configure the Store and Forward using the embedded web server in the player (changing the folder assigned to it), in order to access to any content in the SSH server, since the key is known.

It is recommended to create a user per content group for professional applications in which the same SSH server hosts content for different companies. However, if the final user is generating its own content, a single user would be enough.

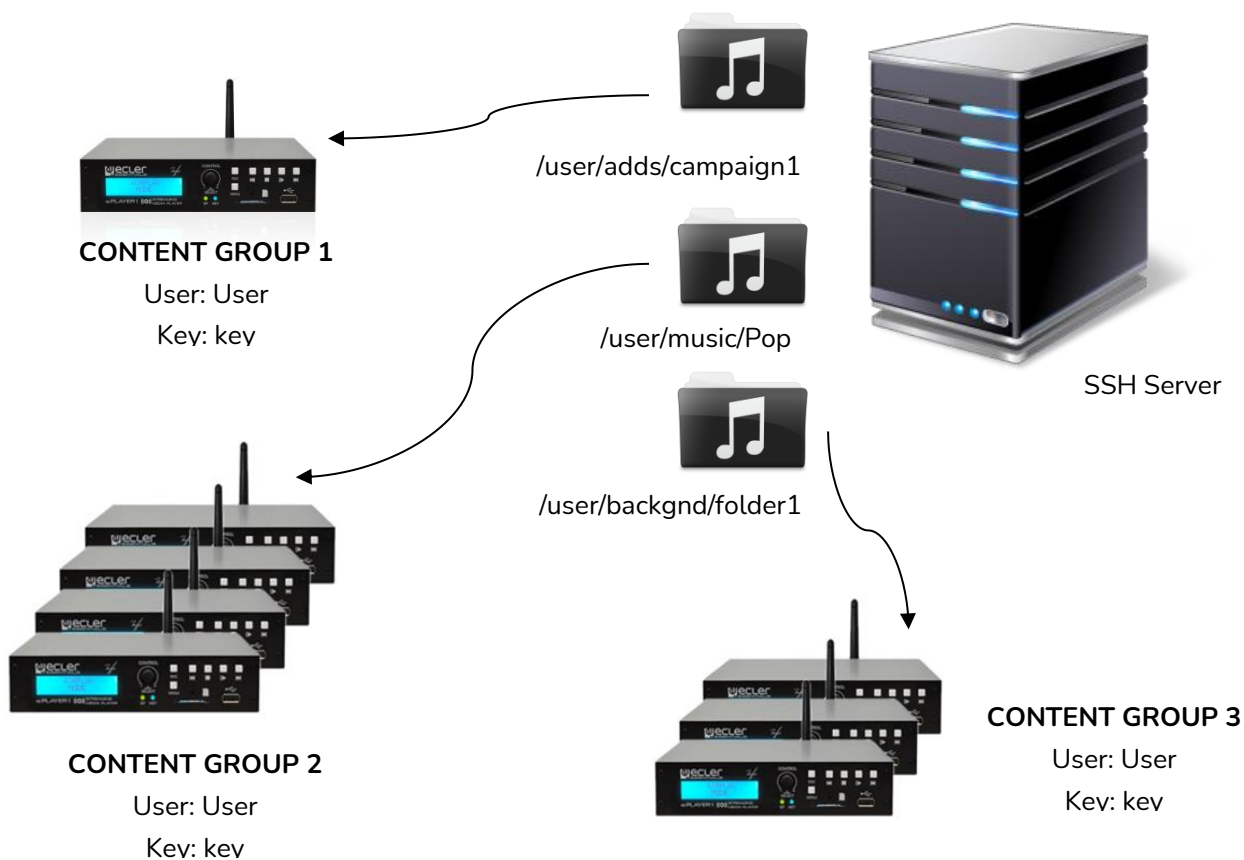


Figura 95: single user

16.2 Installing SSH on Linux

First of all it is necessary to install the SSH package on Linux. Open a terminal and type:

```
sudo apt-get install ssh
```

NOTE: you can open a terminal by using the shortcut **{ctrl + alt + T}**.

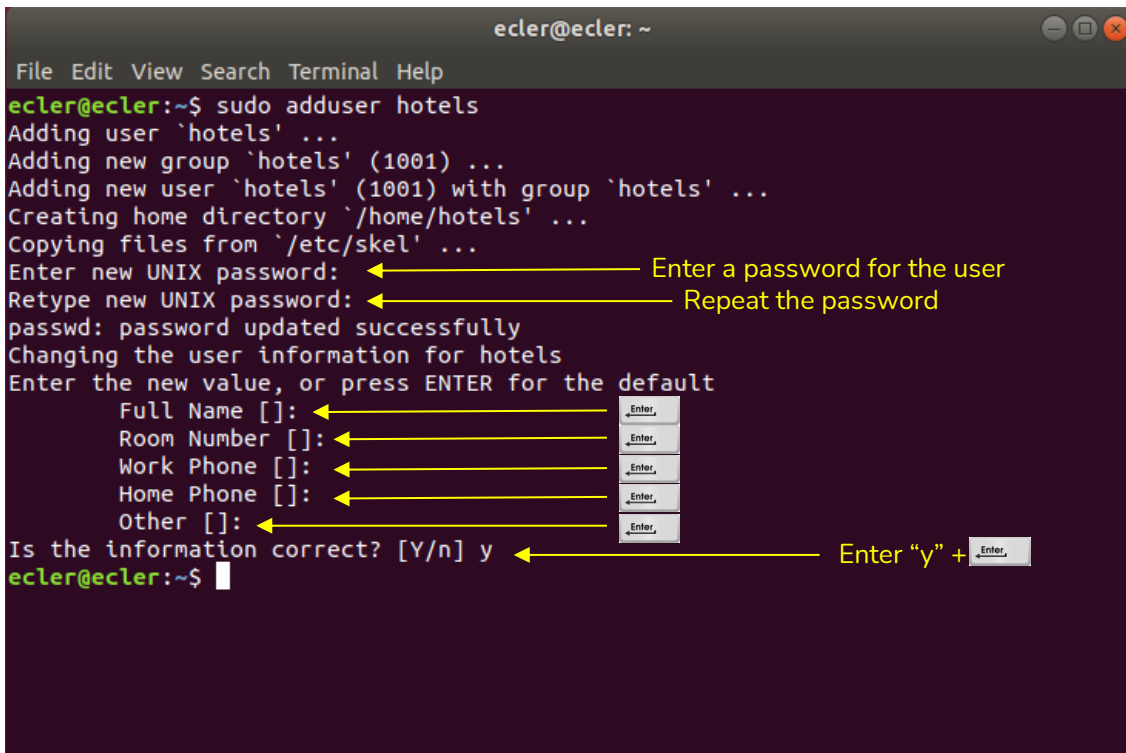
16.3 Creating users on Linux

There will be created as many users as content groups have to be managed. To add a new user, type:

```
sudo adduser <new_user_name>
```

Where **<new_user_name>** stands for the name you want to give to the content group, for example:

```
sudo adduser hotels
```



```
ecler@ecler: ~
File Edit View Search Terminal Help
ecler@ecler:~$ sudo adduser hotels
Adding user `hotels' ...
Adding new group `hotels' (1001) ...
Adding new user `hotels' (1001) with group `hotels' ...
Creating home directory `/home/hotels' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for hotels
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
ecler@ecler:~$
```

Annotations in the image:

- Yellow arrows point from the text "Enter a password for the user" to the "Enter new UNIX password:" prompt.
- Yellow arrows point from the text "Repeat the password" to the "Retype new UNIX password:" prompt.
- Yellow arrows point from the text "Enter 'y' + [Enter]" to the "Is the information correct? [Y/n] y" prompt.
- Small boxes containing "Enter," are placed next to the prompts for Full Name, Room Number, Work Phone, Home Phone, and Other.

Figura 96

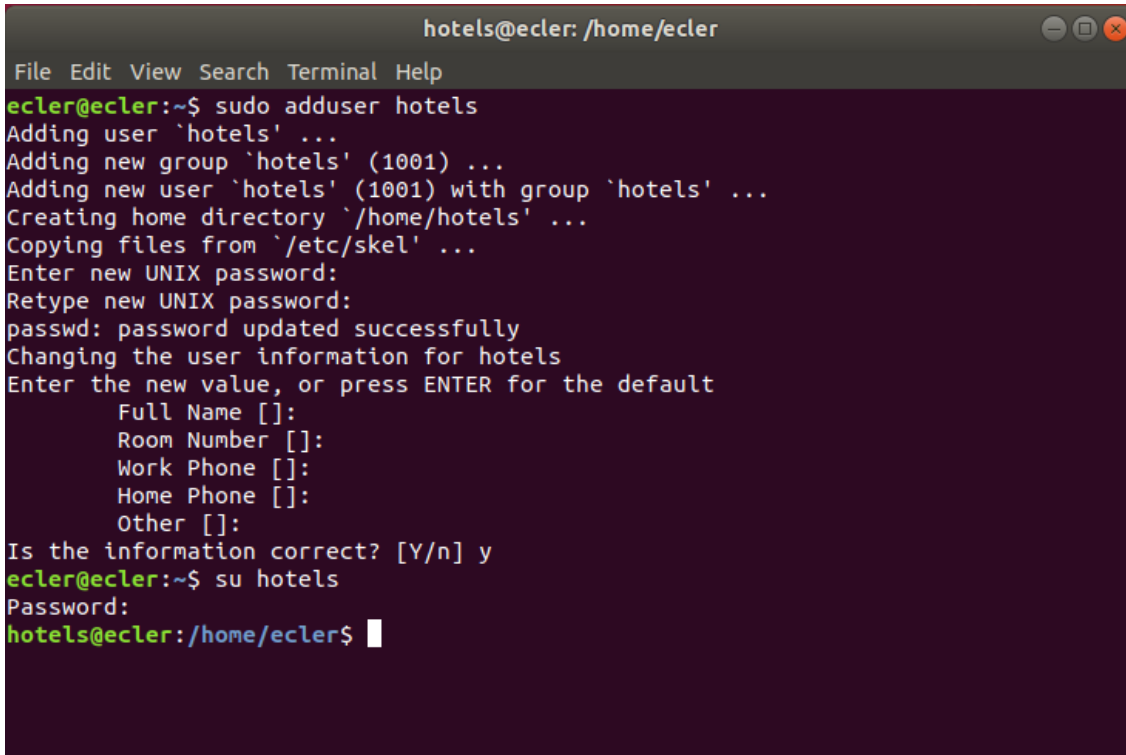
Now, login as the new user:

```
su <user_name>
```

And enter the password introduced in the step before.

In this example:

`su hotels`



```
hotels@ecler: /home/ecler
File Edit View Search Terminal Help
ecler@ecler:~$ sudo adduser hotels
Adding user `hotels' ...
Adding new group `hotels' (1001) ...
Adding new user `hotels' (1001) with group `hotels' ...
Creating home directory `/home/hotels' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for hotels
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] y
ecler@ecler:~$ su hotels
Password:
hotels@ecler:~/home/ecler$
```

Figura 97

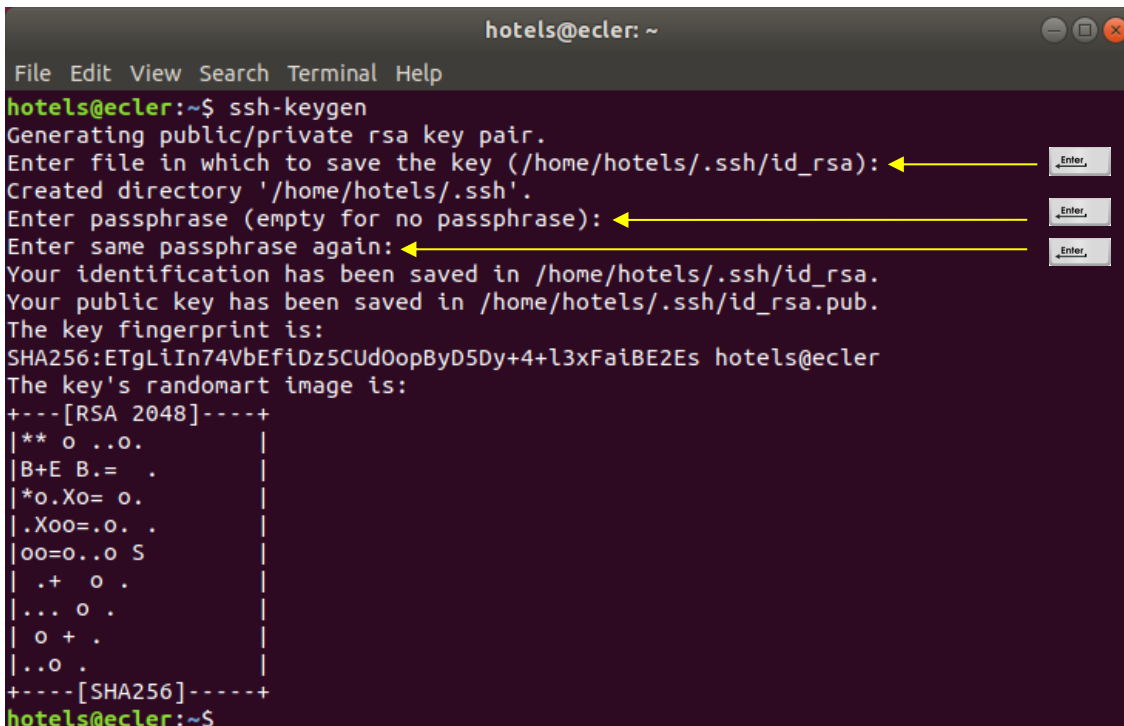
16.4 Generating SSH keys

Before generating the SSH keys, it is necessary to go first to the home folder of the new user. To do this:

```
cd
```

Then use the following command to generate the keys and press enter to each question showed up:

```
ssh-keygen
```



```
hotels@ecler: ~
File Edit View Search Terminal Help
hotels@ecler:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hotels/.ssh/id_rsa):
Created directory '/home/hotels/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/hotels/.ssh/id_rsa.
Your public key has been saved in /home/hotels/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:ETgLiIn74VbEfiDz5CUD0opByD5Dy+4+l3xFaiBE2Es hotels@ecler
The key's randomart image is:
+---[RSA 2048]-----+
|** o ..O. |
|B+E B.= . |
|*o.Xo= o. |
|.Xoo=.o. . |
|oo=0..o S |
|. + o . |
|... o . |
| o + . |
|..o . |
+-----[SHA256]-----+
hotels@ecler:~$
```

Figura 98

Add the public keys to the authorized keys. Use the following command:

```
cat .ssh/id_rsa.pub >> .ssh/authorized_keys
```

Display the private key you will need to introduce in the ePLAYER1's Store and Forward configuration page:

```
cat .ssh/id_rsa
```

Copy from
here →

```

hotels@ecler: ~
File Edit View Search Terminal Help
hotels@ecler:~$ cat .ssh/id_rsa.pub >> .ssh/authorized_keys
hotels@ecler:~$ cat .ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAp9ZG6QXw2QlkwvaE0gFBCC6Dw7ScqG5yARq0bg4ntmzhHm1t
VyD+l/gnaapM2MTSmXwk1Cl0q2fkync8z8daIf3Edv5cm1jRpOCrx69p0KowK4r6
QTrC0vUiZiTE/YS1Q+qA7IhSNCfruh2sStg3KyYbk58ul10HVMVtSLVZHzeSByeL
5kLqAp2Ye8Y4N3iW5LYN0hT9IiEoDZ5gpKkozBU13iY7oob+EaFhKs+Z+POu2Ww4
xeA4JlOI9JnImVq1tRBD1aEdQkoApBBVUW4IK2Czbfbu0VAQS2WsREqGi/BKSC1o
bYOrG08q1F2njDvtXEaT1CSPc2vdg2VaKocWzQIDAQABAoIBAFsANaZ3ECW9kCKd
o1pyT55uCGd80EwKxkm0/zGE/0dMNatXzacmPqSLamNCcTQ/U6jgxN88MwQU6tpU
WGaYQmiXvYR/Whrox5LMNpeopoj7He8eBZ6Ei8nyHBMFoMGMyPOwHIMjTt70nj1l
hVqV3mPI377k2lA2e/Qcr6tUJ2RizjAUm+DgaxgpRIInUycJBzzi9mZkJvov1Tj+
iF2jKwvoVm/l35Mo3bwiJR69aOdfLF1XpKXbntt8+CCSr4puOPhWMPkjd4C7otr4
W6U1jSyQKnjY+fH83tilYTQ/wkSmUJsQBcPa2VMs6+N+53HZLLqTm4Ko14gEkfBl
hN13v0kCgYEA10zJK1lC1r3MLHA2/JSP9FRW2v1ZkXFzS9z6+9JRIUPz4WL2shCT
dgfAgnNlDuqceTE8NTsAWFicNX40gRd9Ehv8Hp/1EOxfOyLRpsyqPFQo6t+uRnyh
VOXL4Eb4RJCYzIsmRTfkWgtZki/TrzeGuSmbnJkqp0Q2R+b4b8mjk+sCgYEAycps
WpqK2JPvDlW8o88VqQ4KS82QktnEWz2BzBosvZtxPQWmwLGVLIcAgZGwt18k8H3I
V9KgyaKMfDs1KvDuFW/2UA4jG0dp5xoGvhj6lAU30bzjDU1GE0ssa9QMOKFftnMF
J2o0kaHEddBV9k7iusyFmm+0mZQc6r7LT/1lKicCgYEA9GBFyXcB6TkcEcYePky
FaLac1LwuJ76QwBNyivVLmXH6f184+aR0mU7lC/GEfHa0ZLeCQpaAUJcMeTyjG78
xjVyF575BSDKUBPYwKgpwKdGowziJ09TJHKpwjA1bRpTarLxUBWjd0nZrFbNzDT1
hxx3eansZLZ7rP+HokjE558CgYB0yU4NAhH8BaZsvmTswEoAu46hkh1EeOoFhbpL
6lxVm39XCAHYF0qGFz/HVw7cSbwugC3cvEwy7dhUu2XC4lB53+gX0a07VWkv7KV9
S1tqt67XnpKF+wnFNrCdWSl9BVFNDxMoFht580M37KPji2FS9c+kQME0xZmUvfG7
xX/e0QKBgBYw83m4k8HEdmgNCdt9nBuVMDL4Vk/IGfYwbFxC0l1DakFarQWEX8eU
h3vaI+rTs++jne0Dex7xwmC8Luh05lFgOCud82cMwZTmHNptLqYDhrfwdxQSCqIH
X0m34GVnrqqiGv14WuzY8bs6y4saG9KF3/EkMEVivPc4lCeCjElh
-----END RSA PRIVATE KEY-----
hotels@ecler:~$

```

→ To here

Figura 99

16.5 Adding audio content to the SSH server

The next step is to place the correct audio content in the SSH server for each specific user or content group. Thus, it will be created a folder inside the home directory of each content group:

```
sudo cp -r <path_with_content> <path_to_place_it>
```

In this example, it will be copied the audio content in directory

```
sudo cp -r /home/ecler/sandf/hotels /home/hotels
```

Notice that in directory `/home/ecler/sandf/` it is storage all the different audio content, for each content group or user, organized in folders.

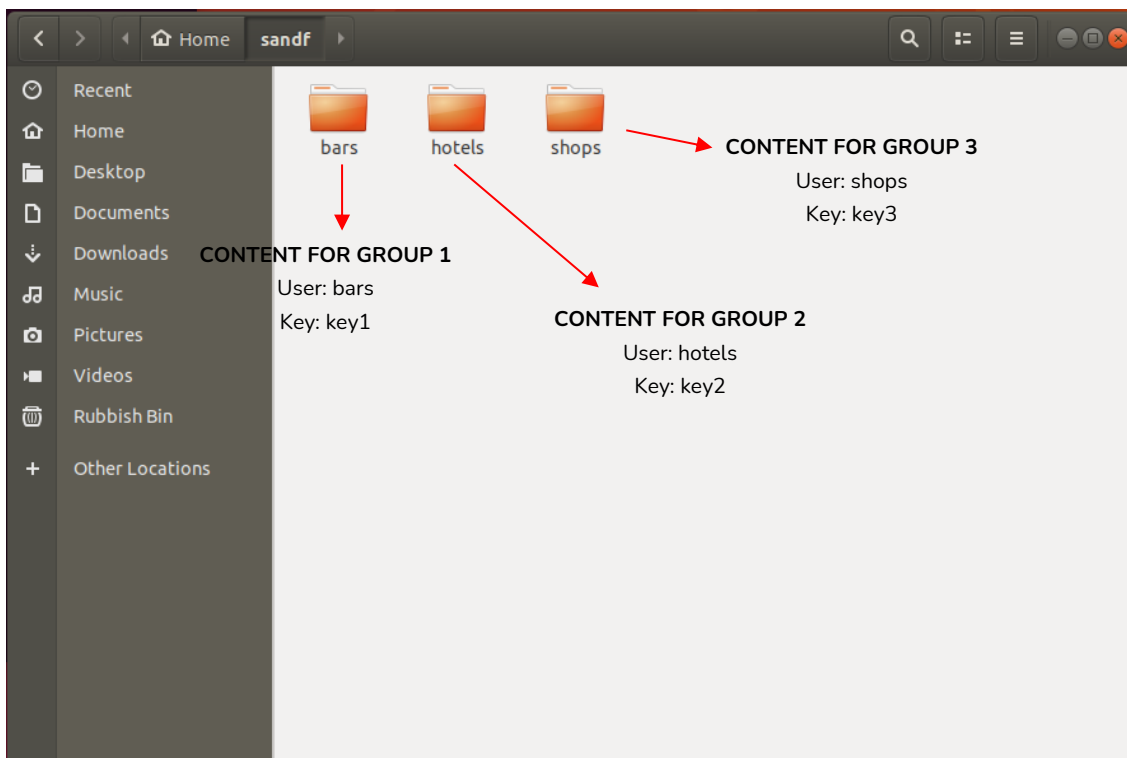


Figura 100

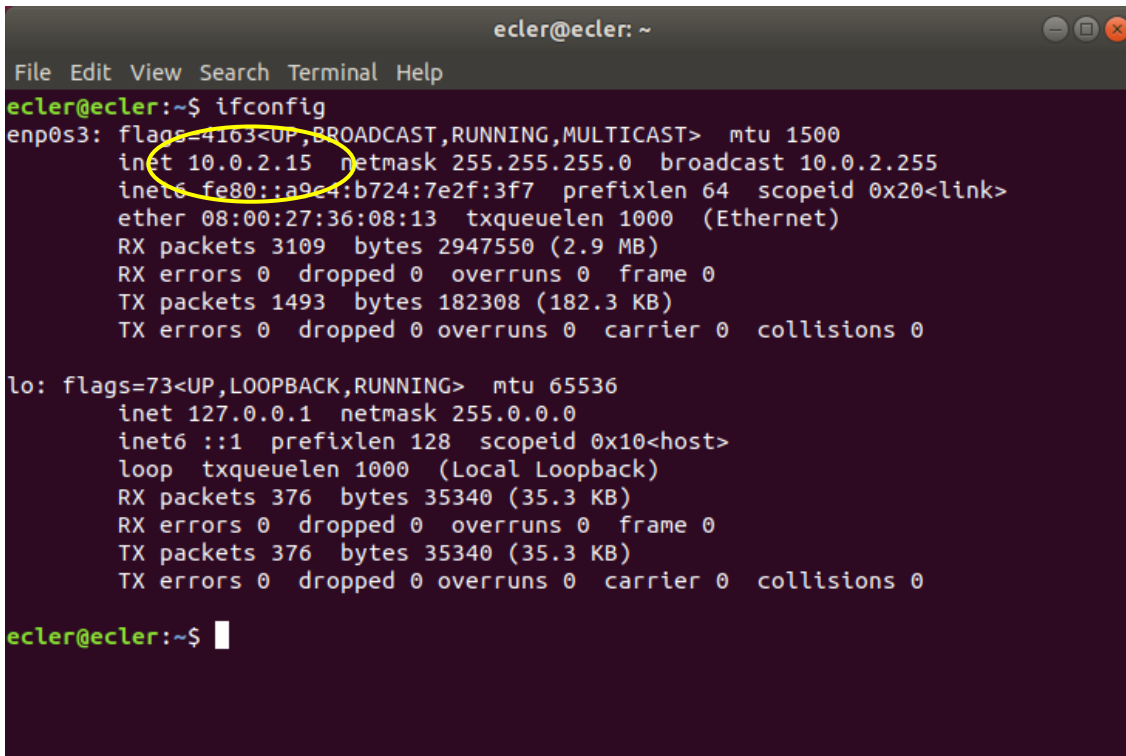
Copy the new directory created `/home/hotels/hotels`. It is the folder where audio contents are storage in the host and need to be copied into PLAYER ONE/PLAYER ZERO S&F's configuration page.

16.6 Configurind the remote source in the player

Finally, it can be configured the remote source (SSH server) in the S&F application.

- Host: is the IP of the SSH server. To check it you can type on the Terminal:

`ifconfig`



```
ecler@ecler: ~
File Edit View Search Terminal Help
ecler@ecler:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
  inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
  inet6 fe80::a9c4:b724:7e2f:3f7 prefixlen 64 scopeid 0x20<link>
  ether 08:00:27:36:08:13 txqueuelen 1000 (Ethernet)
  RX packets 3109 bytes 2947550 (2.9 MB)
  RX errors 0 dropped 0 overruns 0 frame 0
  TX packets 1493 bytes 182308 (182.3 KB)
  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
  inet 127.0.0.1 netmask 255.0.0.0
  inet6 ::1 prefixlen 128 scopeid 0x10<host>
  loop txqueuelen 1000 (Local Loopback)
  RX packets 376 bytes 35340 (35.3 KB)
  RX errors 0 dropped 0 overruns 0 frame 0
  TX packets 376 bytes 35340 (35.3 KB)
  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ecler@ecler:~$
```

Figura 101

- Port: Port of the SSH server. 22 (by default)
- Folder: the directory inside the SSH server with the audio content to sync
- Username: user or name of content group
- Private key: key generated for the user or content group

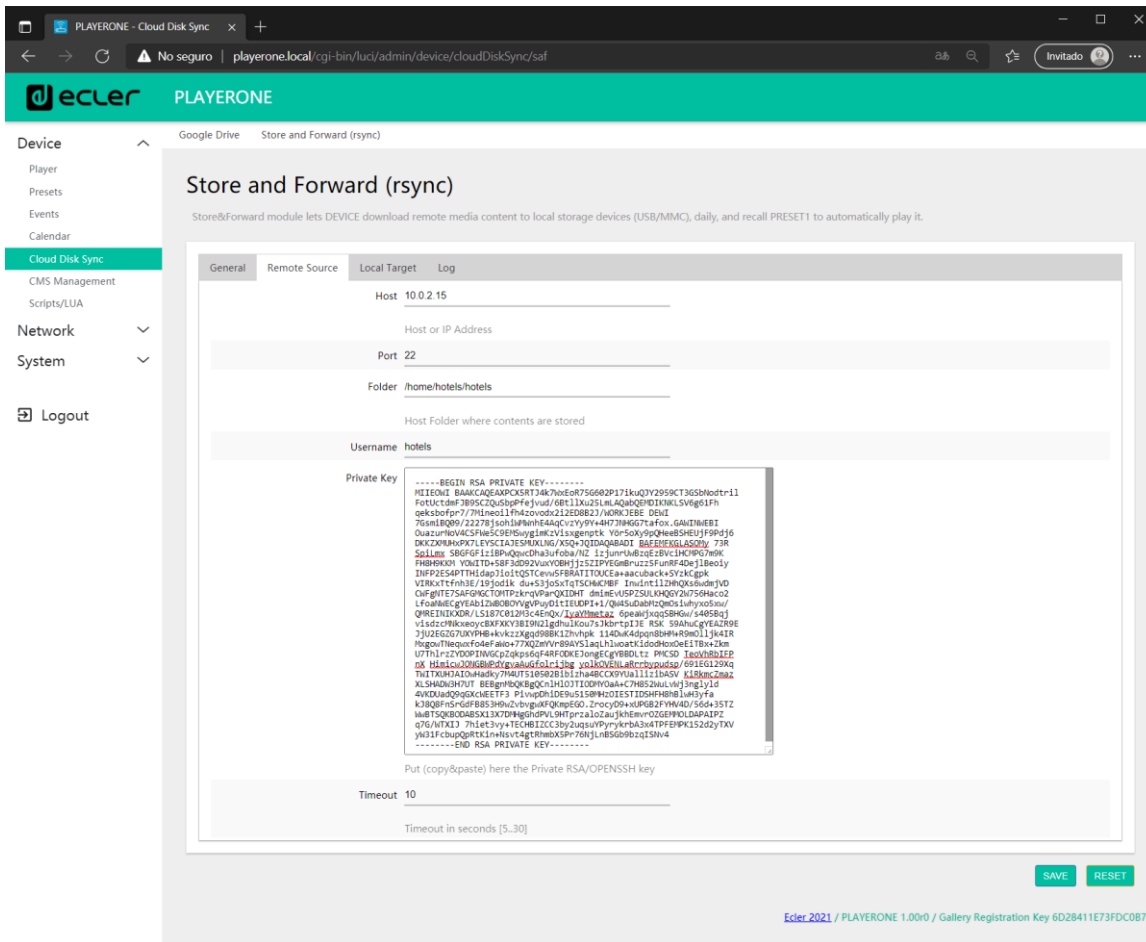


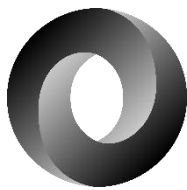
Figura 102

This configuration is the same for every device in the content group.

Repeat steps 3 to 6 for every content group you want to configure for Store & Forward.

THIRD-PARTY CONTROL PROTOCOL

PLAYER ONE JSON Commands



USER MANUAL

17 PLAYER ONE JSON Commands PRODUCT OVERVIEW

JSON (JavaScript Object Notation) is a lightweight data-interchange format that allows PLAYER ONE to communicate with third-party devices and platforms, such as EclerNet Manager. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages. Visit the official website for more information: <https://www.json.org>

18 GENERAL CONSIDERATIONS

- The communication with can be established using Ethernet or WiFi and the TCP/IP transport protocol, always by means of the **2003 TCP port**.
- To check IP address, hold ENTER button for 2-3 seconds to enter admin menu.
- To let some control systems (like CRESTRON®, EXTRON®, AMX®, RTI®, VITY®, MEDIALON®, etc.) process the messages more easily, PLAYER ONE allows to the end of each message with a CR (**\n**) -line feed, character 10-.
- All commands answer {"result":true} (success) or {"result":false} (something failed)

19 PLAYER COMMANDS

SET PLAYER STEREO/MONO

Mono mode

```
{"jsonrpc":"2.0","method":"Player.Stereo","Stereo":false}
```

Stereo mode

```
{"jsonrpc":"2.0","method":"Player.Stereo","Stereo":true}
```

SET PLAYER FADE

No fade

```
{"jsonrpc":"2.0","method":"Player.Fade","Fade":0}
```

Cross Fade

```
{"jsonrpc":"2.0","method":"Player.Fade","Fade":1}
```

Fade

```
{"jsonrpc":"2.0","method":"Player.Fade","Fade":2}
```

SET PLAYER MODE

Player mode Sequential

```
{"jsonrpc":"2.0","method":"Player.Mode","PlayMode":0}
```

Player mode Random

```
{"jsonrpc":"2.0","method":"Player.Mode","PlayMode":1}
```

SET PLAYER REPEAT

Play all

```
{"jsonrpc":"2.0","method":"Player.Repeat","Repeat":0}
```

Play one

```
{"jsonrpc": "2.0", "method": "Player.Repeat", "Repeat": 1}
```

Repeat all

```
{"jsonrpc": "2.0", "method": "Player.Repeat", "Repeat": 2}
```

Repeat one

```
{"jsonrpc": "2.0", "method": "Player.Repeat", "Repeat": 3}
```

GET SHORT PLAYER INFORMATION

```
{"jsonrpc": "2.0", "method": "Player.GetStats"}
```

```
{"title": "Brian Hyland - Sealed With a  
Kiss", "counter": "19:30", "txtSource": "NET", "status": 1}
```

GET FULL PLAYER INFORMATION

```
{"jsonrpc": "2.0", "method": "Player.GetStatsEx"}
```

```
{"title": "Elvis Presley -  
Judy", "counter": "07:02", "txtSource": "NET", "status": 1, "SourceList":  
["", "MMC", "USB UNAVAILABLE", "DLNA", "AIRPLAY", "JVL  
PLAYLIST", "MUSICUP"], "source": 6, "preset": 1, "volume": 100, "txtVolume": "0dB", "stereo": 1, "r  
repeat":  
2, "playmode": 0, "fade": 1, "bootpreset1": 0, "sp": 1, "bitrate": "128", "duration": "--:--  
", "freq": "44.1", "playlist_index": " 0006 / 0056"}
```

PLAYER ADD NEXT PLAYLIST ITEM

With this function user can manage device playlist simply inserting next item before the end of current item.

set next item to "next_item.mp3"

```
{"jsonrpc": "2.0", "method": "Player.QueueNextElem", "url": "mmc://next_item.mp3"}
```

PLAYER INSERT PRORITY ITEM

With this function user can insert a priority item that will be played "over" the actual playing item. Current playing item will be fade.

set next item to "priority_item.mp3"

```
{"jsonrpc": "2.0", "method": "Player.PrioritySetElem", "url": "usb://priority_item.mp3"}
```

PLAYER PLAY

If the player is paused or stopped use this function to start current loaded item reproduction, otherwise the player is paused.

```
{"jsonrpc": "2.0", "method": "Player.Play"}
```

PLAYER STOP

```
{"jsonrpc": "2.0", "method": "Player.Stop"}
```

PLAYER NEXT

```
{"jsonrpc": "2.0", "method": "Player.Next"}
```

PLAYER PREVIOUS

```
{"jsonrpc": "2.0", "method": "Player.Prev"}
```

INCREMENT VOLUME

Increment volume just one dB

```
{"jsonrpc": "2.0", "method": "Player.Volume", "Action": "inc"}
```

DECREMENT VOLUME

Decrement volume just one dB

```
{"jsonrpc": "2.0", "method": "Player.Volume", "Action": "dec"}
```

SET VOLUME

Param volume is expressed in %. To set volume to 50% use next command.

```
{"jsonrpc": "2.0", "method": "Player.Volume", "Volume": 50}
```

-

OPEN PLAYLIST URL

Url param must be any valid device url.

```
{"jsonrpc": "2.0", "method": "Player.Open", "Url": "http://50.7.181.186:8060"}
```

OPEN PRESET INDEX 10

Preset param must be a valid preset index 1 to 20

```
{"jsonrpc": "2.0", "method": "Player.Open", "Preset": 10}
```

OPEN SOURCE INDEX 4 AIRPLAY (FROM AVAILABLE SOURCES LIST)

Source must be a valid player source index. Please check "Get of list available sources" to know all valid sources.

```
{"jsonrpc": "2.0", "method": "Player.Open", "Source": 4}
```

GET LIST OF AVAIAABLE SOURCES

This command returns the list of available sources.

```
{"jsonrpc": "2.0", "method": "Source.GetList"}
```

```
{"SourceList": [ "", "MMC", "USB UNAVAILABLE", "DLNA", "AIRPLAY", "ROCK 80s", "DISCO 80s" ] }
```

20 CONFIGURATION COMMANDS

RESET DEVICE SETTINGS

Restore device factory default settings. All your settings will be lost.

```
{"jsonrpc": "2.0", "method": "Settings.Reset"}
```

RESTORE DEVICE SETTINGS FROM URL

Restore device settings to values in url file.

```
{"jsonrpc": "2.0", "method": "Settings.Restore", "url": "http://ecler.com/my\_player\_config.conf"}
```

BACKUP CURRENT DEVICE CONFIGURATION

Backup device settings to url. Available configurations: user, admin or gallery.

```
{"jsonrpc": "2.0", "method": "Settings.Backup", "url": "mmc://backups/gim.config", "user": "admin"}
```

-

GETTING DEVICE VARIABLE VALUE

This function returns a device variable value. Please check the Player LUA manual in order to check all the interface.settings.variable values.

In order to retrieve preset01.settings.bname (preset name) user should send next command to the PLAYER ONE:

```
{"jsonrpc": "2.0", "method": "CFG.get", "interface": "preset01", "section": "settings", "variable": "bname"}  
{"value": "AFTERNOON PRESET"}
```

SETTING DEVICE VARIABLE VALUE

This function set a device variable value. Please check the Player LUA manual in order to check all the interface.settings.variable values.

In order to set preset01.settings.bname (preset name) user should send next command to the PLAYER ONE:

```
{"jsonrpc": "2.0", "method": "CFG.set", "interface": "preset01", "section": "settings", "variable": "bname", "value": "MIDNIGHT PRESET"}
```

STORE CHANGES IN DEVICE INTERNAL MEMORY

This function stores all interface variables to the internal device memory. Should be called after set all the changes. PLAYER ONE must reload the data using [reload functions](#).

```
{"jsonrpc": "2.0", "method": "CFG.commit", "interface": "preset01"}
```

21 PRESET COMMANDS

Preset variables for GFG.get and CFG.set. Please check Player LUA manual. [Preset variables](#).

RELOAD PRESET

Reload indicated preset index. Index should be a valid preset index 1..20. Must be called after modifying preset variables and call commit command

```
{"jsonrpc": "2.0", "method": "Preset.Reload", "Index": 1}
```


22 EVENT COMMANDS

Event variables for GFG.get and CFG.set. Please check Player LUA manual. Event variables.

RELOAD EVENT

Reload indicated event. Name should be: GPI1, GPI2 or SILENCE. Must be called after modifying event variables and call commit command.

```
{"jsonrpc": "2.0", "method": "Event.Reload", "Name": "GPI1"}
```

23 CALENDAR COMMANDS

Calendar variables for GFG.get and CFG.set. Please check Player LUA manual. Preset variables

RELOAD CALENDAR

Reload indicated calendar. Calendar index should be a number 1..24. Must be called after modifying calendar variables and call commit command. Reload calendar 24 example:

```
{"jsonrpc": "2.0", "method": "Calendar.Reload", "Index": 24}
```

24 STORE AND FORWARD COMMANDS

SAF variables for GFG.get and CFG.set. Please check Player LUA manual. SAF variables

RELOAD SAF

Reload SAF configuration. Must be called after modifying SAF variables and call commit command.

```
{"jsonrpc": "2.0", "method": "SAF.Reload"}
```

25 GOOGLE DRIVE COMMANDS

SAF variables for GFG.get and CFG.set. Please check Player LUA manual. SAF variables

RELOAD GOOGLE DRIVE

Reload GDRIVE configuration. Must be called after modifying Google Drive variables and call commit command.

```
{"jsonrpc": "2.0", "method": "GDRIVE.Reload"}
```

AUTHENTICATE GOOGLE DRIVE

Call this command to validate Google Drive configuration with Google servers.

```
{"jsonrpc": "2.0", "method": "GDRIVE.Authenticate"}
```

SYNCHRONIZE GOOGLE DRIVE

Call this function to synchronize now Google Drive content
{ "jsonrpc": "2.0", "method": "GDRIVE.Synchronize" }

26 CONTENT MANAGEMENT SYSTEM (CMS) COMMANDS

CMS variables for GFG.get and CFG.set. Please check Player LUA manual. CMS variables.

RELOAD CMS

Reload CMS configuration. Must be called after modifying CMS variables and call commit command.
{ "jsonrpc": "2.0", "method": "CMS.Reload" }

27 SCRIPTS COMMANDS

Script variables for GFG.get and CFG.set. Please check Player LUA manual. Script variables.

RELOAD SCRIPT

Reload script configuration. Index should be script index 1 to 20. Must be called after modifying Script variables and call commit command. Reload script 7 example:
{ "jsonrpc": "2.0", "method": "Script.Reload", "Index": 7 }

EXECUTE SCRIPT 6

{ "jsonrpc": "2.0", "method": "Script.Command", "Index": 6, "Command": "Start" }

KILL SCRIPT 3

{ "jsonrpc": "2.0", "method": "Script.Command", "Index": 3, "Command": "Stop" }

QUERY SCRIPT 11 STATUS

{ "jsonrpc": "2.0", "method": "Script.Status", "Index": 11 }
{ "status": "Idle" }

28 REGISTER COMMANDS

ADD REGISTER LINE

Add line to device LOG. Possible line values are: Trace, Warning, Error.

Add a warning line example:

{ "jsonrpc": "2.0", "method": "Device.Log", "Severity": "Trace", "Message": "This is a warning message" }

29 DEVICE COMMANDS

DEVICE REBOOT

```
{"jsonrpc":"2.0","method":"Device.Reboot"}
```

GET DEVICE VERSION

```
{"jsonrpc":"2.0","method":"Device.GetVersion"}  
{"version":"3.04r0"}
```

DEVICE UPDATE FIRMWARE

With this function user could update device firmware to an specific version. User must provide firmware url. Device setting will be saved.

```
{"jsonrpc":"2.0","method":"Device.Update","url":"https://www.ecler.com/new_firmware.bin"}  
}
```

DEVICE BOOT CONFIG COMMAND

Available BootPreset1 options are: 1 – PRESET1, 2 – keep status

```
{"jsonrpc":"2.0","method":"Device.BootPreset1","BootPreset1":2}
```

DEVICE GET MAC

```
{"jsonrpc":"2.0","method":"Device.GetMac"}  
{"mac":"32 41 41 20 40 42"}
```

DEVICE GET GALLERY REGISTRATION KEY

```
{"jsonrpc":"2.0","method":"Device.GetRegkey"}  
{"regkey":"2E1BB146B2DB2WA1"}
```

30 PANEL COMMANDS

PANEL SET LOCK SETTINGS

Set panel Lock to "UNLOCK ALL" "UNLOCK USER" "LOCK ALL". Set panel password to Pass.

```
{"jsonrpc":"2.0","method":"Device.Panel","Lock":"LOCK ALL","Pass":"1234"}
```

PANEL GET LOCK SETTINGS

```
{"jsonrpc":"2.0","method":"Device.GetPanel"}  
{"Lock":"UNLOCK ALL","Pass":""}
```

FINDER COMMANDS

Start/stop finder operation

```
{"jsonrpc":"2.0","method":"Device.Finder","Finder":true}
```

LCD FUNCTIONS

Print text on device frontal display. Two lines are available and aligned centre if Centre variable is true. Is possible to specify the display timeout in seconds.

```
{"jsonrpc":"2.0","method":"Device.Print","Line1":"Hi","Line2":"Bye","Center":true,"Timeout":3}
```

31 ePLAYER1 LUA SDK v1.02

31.1 Changelist

- **v1.02 – xxxx/xx/xx**
 - ✓ Added DEV_reboot function
 - ✓ Added CFG_backup2 function with type parameter: user, admin and gallery
 - ✓ Bug fix CFG_backup was calling to Device.Backup instead Settings.Backup
 - ✓ Typo, Calendar general_enabled changed to bEnabled
 - ✓ Typo, Scripts enabled change to bEnabled
- **v1.01 – 2018/03/14**
 - ✓ Added DEV_mac function
 - ✓ Added DEV_regkey function
- **v1.00 – 2018/02/08**
 - ✓ First version of the document also named “Preliminary version”

32 ePLAYER1 LUA SDK Introduction

ePLAYER1 is a Lua extension library that acts as interface between LUA and ePLAYER1 firmware using the well known ePLAYER1 JSON protocol . Basically is composed by different kind of objects:

- PLAYER – access player functions
- CFG – access configuration. With this object you can access all the ePLAYER1 configuration.
- PRESET – PRESET settings. There are 20 presets that act as memories where you can store all the ePLAYER1 player settings like url, play mode, repeat mode, fade mode, etc.
- EVENT – Event configuration lets ePLAYER1 execute automated functions. There are 3 possibilities GPI1, GPI2 and SILENCE settings
- CAL – Calendar configuration lets ePLAYER1 execute automated functions on date event. There are up to 24 different calendar.
- SAF – Store and Forward settings
- GDRIVE - lets ePLAYER1 download Google Drive content to local storage devices (USB/MMC). Daily execution on selected time

- CMS – CMS settings
- SCRIPT – Manage ePLAYER1 scripts. Is possible to execute scripts based on different kind of triggers like BOOT, EVENT, LOAD PRESET, CALENDAR, NETWORK, etc.
- LOG – Register functions that allows the user to write to the ePLAYER1 internal REGISTER
- DEV – Device status and other setups (like install new firmwares)
- PANEL – Panel lock mode settings and status
- FINDER – use FINDER settings to discover your ePLAYER1.
- LCD – Allow access to ePLAYER1 LCD display

33 How to use ePLAYER1 module?

In order to use ePLAYER1 module you must include it in your source code using require statement. EPLAYER1 automates this function for you, but you can also include ePLAYER1 module manually and create your ePLAYER1 object as you need using the next two lines of code:

```
require "ePLAYER1"  
mydev = ePLAYER1.new()
```

In this case you can program actions directly to your ePLAYER1 device, but also it is possible to access ePLAYER1 directly from your PC (in this case it is recommended to use some LUA development tool like Eclipse/LUA)

If you want to use ePLAYER1 directly from your PC you can call “new” function with IP and PORT parameters as shown next:

```
require "ePLAYER1"  
mydev = ePLAYER1.new("10.120.120.4", 2003)
```

Where 10.120.120.4 is the ePLAYER1 IP address, that you can check by pressing ENTER button for 10 seconds and going to WIFI or ETHERNET menu. 2003 is the JSON port.

One time ePLAYER1 object is created is possible to access all functions in it.

Let see what you can do.

33.1 Conventions

- **n** indicates number
- **b** indicates Boolean
- **txt** indicates text

33.2 PLAYER FUNCTIONS

Access player functions like repeat, fade, etc.

Functions

Function:

[PLAYER_stereo\(bStereo\)](#)

Description:

Set the player output mode to stereo or mono

Parameters:

bStereo – *boolean*, set to true for stereo output, otherwise mono output is selected

Function:

[PLAYER_fade\(nFade\)](#)

Description:

Set the player fade mode

Arguments:

nFade – *integer* 0-OFF, 1-XFADE, 2-FADE

Function:

[PLAYER_mode\(nPlayMode\)](#)

Description:

Set the player playlist sequence mode

Arguments:

nPlayMode – *integer* 0-SEQUENTIAL, 1-RANDOM

Function:

[PLAYER_repeat\(nRepeat\)](#)

Description:

Set the player repeat mode

Arguments:

nRepeat – *integer* 0-PLAY ALL, 1-PLAY ONE, 2-REPEAT ALL, 3-REPEAT ONE

Function:

[PLAYER_getStats\(\)](#)

Description:

Get the player statistics and status

Return:

String – xml string containing all the player status

In the next example you could see how to obtain the player stats and how to parse it using cJSON library.

```
require "ePLAYER1"
```

```
ep=ePLAYER1.new()
json_text=ep.PLAYER_getStats()
print(json_text)

cjson=require "cjson"
value=cjson.decode(json_text)
print("SourceList[3]=" .. value.SourceList[3])
```

output

```
{"title":"The White Stripes - Seven Nation Army","counter":"52:52","txtSource":"NET","status":1,"SourceList":
[["","MMC","USB","DLNA","AIRPLAY","rock
alternativo"],"source":5,"preset":1,"volume":100,"txtVolume":"0dB","stereo":1,"repeat":2,"playmode":0,"fade":1,"
bootpreset1":0,"sp":1,"bitrate":"192","duration":"--:--","freq":"44.1","playlist_index":" 0001 /
0001","playlist_url":"mmc://radiobob-alternativerock-mp3-hq?
sABC=5n6s2sr8%230%23no8617362n29q2o435p17n54928n16s5%23gharva&amspams=playerid:tunein;skey:15
17236200","priority":""}]
SourceList[3]=USB
```

Function:

```
PLAYER_queue(urlNextElem)
```

Description:

Add next playlist item. Using this function you can compose your own playlist. In order to do a continuous play it is necessary to queue next item prior to the end of the current one.

Arguments:

urlNextElem – url of the item to add

Function:

```
PLAYER_priority(urlElemPriority)
```

Description:

Use this function to play priority sounds over the normal playlist. It apply

Return:

String – url of the priority element

Function:

```
PLAYER_play()
```

Description:

If the player is paused or stopped use this function to start current loaded item reproduction, otherwise the player is paused.

Function:

```
PLAYER_stop()
```

Description:

Use this function to stop the current player reproduction.

Function:

```
PLAYER_next()
```

Description:

Use this function to advance to the next item. It retains the play state after advance is done.

Function:

```
PLAYER_previous()
```

Description:

Use this function to move back to the previous item. It retains the play state.

Function:

```
PLAYER_incvol = function()
```

Description:

Use this function to increase the player volume 3dB.

Function:

```
PLAYER_decvol = function()
```

Description:

Use this function to decrease the player volume 3dB.

Function:

```
PLAYER_setvol = function(perc)
```

Description:

Use this function to set the current player volume

Arguments:

nVolume – *integer* 0 to 100

Function:

```
PLAYER_setvol = function(perc)
```

Description:

Call this function to set the mute player parameter.

Arguments:

bMute – *boolean* set to true to mute the player, otherwise unmute the player.

Function:

```
PLAYER_open = function( options )
```

Description:

Call this function to open/load a preset or source or url

Arguments:

url (optional) -- String of the url you want to open e.g. mmc://my-music/ http://my.server/music-mp3
preset (optional) – integer from 1 to 20. Indicates the preset you want to open
source (optional) – integer from 1 to 24 representing MMC, USB, DLNA, AIRPLAY and presets from PRESET1 to PRESET 20 (preset must be configured properly)

How to load "PRESET 12"?

```
require "ePLAYER1"  
ep = ePLAYER1.new()  
ep.PLAYER_open({preset=12})
```

How to load url "http://my.server/music-mp3"?

```
require "ePLAYER1"  
ep = ePLAYER1.new()  
ep.PLAYER_open({url="http://my.server/music-mp3"})
```


How to load source "AIRPLAY"?

```
require "ePLAYER1"  
ep = ePLAYER1.new()  
ep.PLAYER_open({source=4})
```

*note the { } around the parameter which indicates optional arguments

33.3 CONFIG (CFG) FUNCTIONS

Set and get ePLAYER1 parameters: PRESET, CMS, SAF, CAL, EVENT, GDRIVE, SCRIPT and LOG. Please refer to the corresponding section to check how to set/get each parameter.

Functions

Function:

```
CFG_reset()
```

Description:

Restore ePLAYER1 configuration to its initial state (factory defaults). Please note that using this function you will lost all your changes.

Function:

```
CFG_restore(urlRestore)
```

Description:

Restore ePLAYER1 settings from the urlRestore file e.g. mmc://good-settings.config
http://my.server/ePLAYER1/mycompany.settings

Arguments:

urlRestore – String indicating the url where settings you want to restore

Note:

If you wan to apply all settings you must call [DEV_reboot](#) or call every XXX_reload function to apply changes one per one.

Function:

```
CFG_backup(urlBackup, bUser)
```

Description:

Backups ePLAYER1 settings to urlBackup.

Arguments:

urlBackup – String indicating the url where settings are stored e.g. mmc://good-settings.config
http://my.server/ePLAYER1/mycompany.settings

bUser – Boolean set to true for user settings only, otherwise all the settings are backedup to the target destination

Function:

```
CFG_backup2(urlBackup, type)
```

Description:

Backups ePLAYER1 settings to urlBackup.

Arguments:

urlBackup – String indicating the url where settings are stored e.g. mmc://good-settings.config
http://my.server/ePLAYER1/mycompany.settings
type – String available possibilities are: user, admin and gallery. Select gallery for Player, Events, CMS, Scripts, Player Profile and Network settings. Select user for Player, Presets, Events, Calendar, Cloud Disk, CMS and Scripts. Select Admin for all the available settings.

Function:

```
CFG_get = function(interface, section, variable)
```

Description:

Return the value of the selected variable “interface.section.variable”

Arguments:

interface – must be a valid interface: preset, cms, saf, calendar(01..24), event (gpi1, gpi2, silence), gdrive, script and log.
section – variable's section. Most of the cases is “settings”
variable – variable's name.

Return:

String with the variable value.

Function:

```
CFG_set = function(interface, section, variable, value)
```

Description:

Set the value of the selected variable “interface.section.variable” to “value”

Arguments:

interface – must be a valid interface: preset, cms, saf, calendar(01..24), event (gpi1, gpi2, silence), gdrive, script and log.
section – variable's section. Most of the cases is “settings”
variable – variable's name.
value – variable's value.

Function:

```
CFG_commit = function(interface)
```

Description:

Commit interface changes. After changes all variables in one interface it's necessary to dump changes to ePLAYER1 internal memory. Do it one time for each modified interface.

Arguments:

interface – must be a valid interface: network, wireless, preset, cms, saf, calendar(01..24), event (gpi1, gpi2, silence), gdrive, script and log.

33.4 PRESET FUNCTIONS

Set and get ePLAYER1 PRESET parameters. There are 20 presets available. Each preset can configure: name, events enabled, playlist, media alias, play status, volume, mute, play mode, repeat mode, fade mode, stereo/mono.

Functions

Function:

[PRESET_reload\(index\)](#)

Description:

Reload preset configuration. Call it after modify the preset configuration using CFG_set and CFG_commit

Arguments:

index – number indicating the preset number (from 1 to 20)

33.4.1 [Preset variables](#)

presetNN.settings.bname	Preset name
presetNN.settings.eventList	List of enabled events separated by % (GPI1%GPI2%SILENCE)
presetNN.settings.bPlaylist	1 to overwrite playlist
presetNN.settings.loadPlaylist	url of the playlist. Must be enabled using presetNN.settings.bPlaylist
presetNN.settings.mediaAlias	Alias of the enabled playlist. It appears as source
presetNN.settings.bStatus	1 to overwrite status
presetNN.settings.status	Preset status must be PLAY or STOP
presetNN.settings.bVolume	1 to overwrite preset volume
presetNN.settings.volume	Preset volume in % (from 0 to 100)
presetNN.settings.mute	1 indicates mute, 0 to unmute
presetNN.settings.bPlayMode	1 to overwrite play mode
presetNN.settings.playMode	RANDOM or SEQUENTIAL
presetNN.settings.bRepeatMode	1 to overwrite repeat mode
presetNN.settings.repeatMode	PLAY ALL, PLAY ONE, REPEAT ALL or REPEAT ONE
presetNN.settings.bFadeMode	1 to overwrite fade mode
presetNN.settings.fadeMode	OFF, XFADE or FADE
presetNN.settings.bStereo	1 to overwrite stereo/mono mode
presetNN.settings.stereo	MONO or STEREO

Note: all variables are optional.

NN indicates a number 01 to 20

33.4.2 Preset examples

```
require "ePLAYER1"
ep = ePLAYER1.new()

ep.CFG_set("preset03", "settings", "bname", "My first preset") ep.CFG_set("preset03",
"settings", "bPlaylist", 1)

ep.CFG_set("preset03", "settings", "mediaAlias", "Alias of my first preset")
ep.CFG_set("preset03", "settings", "bVolume", 1)

ep.CFG_set("preset03", "settings", "volume", 100)
ep.CFG_set("preset03", "settings", "loadPlaylist", "mmc://")
ep.CFG_set("preset03", "settings", "eventList", "GPI1%SILENCE")
ep.CFG_commit("preset03")

ep.PRESET_reload(3)
```

At this moment you can load the configured preset with this instruction:

```
ep.PLAYER_open({preset=3})
```

33.5 EVENT FUNCTIONS

Event functions lets ePLAYER1 execute automated functions. There are 3 possibilities GPI1, GPI2 and SILENCE settings.

Functions

Function:

```
EVENT_reload(txtEvent)
```

Description:

Reload event configuration. Call it after modify the preset configuration using CFG_set and CFG_commit

Arguments:

txtEvent – String indicating the event to reload (GPI1, GPI2 or SILENCE)

33.5.1 Event variables

When programming GPI1 or GPI2 these variables are available

gpiN.settings.source_polarity	Event polarity DIRECT or REVERSE
gpiN.settings.target_type	INTERNAL, PRESET RECALL, TRANSPORT CONTROL, LOAD & PLAY SOURCE or PRIORITY SOURCE
gpiN.settings.target_preset	If target_type=PRESET RECALL put here the preset number 1 to 20
gpiN.settings.target_transport	If target_type=TRANSPORT CONTROL enter here one of these options: STOP, PLAY, PREV/RW, NEXT/FW
gpiN.settings.target_loadsource	if target_type=LOAD & PLAY SOURCE or target=PRIORITY SOURCE put here the url you want to load and play
gpiN.settings.target_priomode	If target_type=PRIORITY SOURCE enter here the priority mode. Choose between HOLD or PULSE
gpiN.settings.target_prio_retrigger	If target_priomode=PULSE set to 1 to active retrigger option. 0 to disable it
gpiN.settings.target_prio_pulse_time	If target_prio_retrigger=1 put here the playback duration

Note: N should be 1 or 2 (for GPI1 or GPI2)

While programming SILENCE these variables are available

silence.settings.source_detecttime	Put here the detection time in seconds
silence.settings.target_type	INTERNAL, PRESET RECALL, LOAD & PLAY SOURCE
silence.settings.target_preset	If target_type=PRESET RECALL put here the preset number 1 to 20
silence.settings.target_loadsource	if target_type=LOAD & PLAY SOURCE or target=PRIORITY SOURCE put here the url you want to load and play

33.6 CALENDAR (CAL) FUNCTIONS

Calendar functions let ePLAYER1 execute automated functions on a date/time. There are 24 different calendar events available.

Functions

Function:

[CAL_reload\(nIndex\)](#)

Description:

Reload calendar configuration. Call it after modifying the calendar configuration using [CFG_set](#) and [CFG_commit](#)

Arguments:

nIndex – number indicating the calendar to reload (1 to 24)

33.6.1 Calendar variables

calendarNN.settings.general_description	Calendar name or description
calendar01.settings.bEnabled	Set to 1 to enable this calendar
calendarNN.settings.source_start_date	Start date (YYYY/MM/DD). Could be blank
calendarNN.settings.source_start_time	Start time (HH:MM). Could be blank.
calendar.settings.source_end_enable	Could be FOREVER or CUSTOM END DATE
calendarNN.settings.source_end_date	If CUSTOM END DATE is selected put here the calendar end date (YYYY/MM/DD)
calendarNN.settings.source_end_time	If CUSTOM END DATE is selected put here the calendar end time (HH:MM)
calendarNN.settings.source_week_nonactive	Week mask where calendar is not active. Should be a string of days. Each day is represented by his 2 first letters (MoTuWeThFrSaSu)
calendarNN.settings.source_repeat_enable	1 indicates that the calendar must repeat the selected interval time up to the number indicated in times variable. 0 no repetition
calendarNN.settings.source_repeat_interval	If repeat is enabled this parameter indicates the repetition time
calendarNN.settings.source_repeat_times	If repeat is enabled this parameter indicates the repetition's number
calendarNN.settings.target_type	INTERNAL, PRESET RECALL, TRANSPORT CONTROL, LOAD & PLAY SOURCE or PRIORITY SOURCE
calendarNN.settings.target_preset	If target_type=PRESET RECALL put here the preset number 1 to 20
calendarNN.settings.target_transport	If target_type=TRANSPORT CONTROL enter here one of this options: STOP, PLAY, PREV/RW, NEXT/FW
calendarNN.settings.target_loadsource	if target_type=LOAD & PLAY SOURCE or target=PRIORITY SOURCE put here the url you want to load and play

Note: NN indicates a number 01 to 24

33.7 STORE AND FORWARD (SAF) FUNCTIONS

Store and Forward module lets ePLAYER1 download remote media content from a rsync/ssh server to local storage devices (sme as PRESET1), daily, and recall PRESET1 to automatically play it.

Functions

Function:

[SAF_reload\(\)](#)

Description:

Reload Store and Forward configuration. Call it after modify the Store and Forward configuration using CFG_set and CFG_commit

33.7.1 Store and Forward variables

saf.settings.bEnabled	Set to 1 to enable daily Store and Forware synchronization
saf.settings.time_param	Enter here the synchronization hour (HH:MM)
saf.settings.host	Put here the rsync/ssh server address
saf.settings.port	Enter here the rync/ssh port (default 22)
saf.settings.source_path	Enter here the server folder where the contents are stored
saf.settings.source_user	SSH/rsync username parameter
saf.settings.source_key	Enter here your private SSH/rsync key
saf.settings.source_timeout	Enter here the SSH/rsync operation timeout (in seconds)

33.8 GOOGLE DRIVE (GDRIVE) FUNCTIONS

GoogleDrive module lets ePLAYER1 download remote media content from a Google Drive account to local storage devices (USB/MMC), daily, and recall PRESET1 to automatically play it.

Functions

Function:

[GDRIVE_reload\(\)](#)

Description:

Reload Google Drive configuration. Call it after modify the Store and Forward configuration using [CFG_set](#) and [CFG_commit](#)

Function:

[GDRIVE_authenticate\(\)](#)

Description:

Run the authentication process. Is mandatory to run the authenticate process one time after change the key with [CFG_set](#)

Function:

[GDRIVE_synchronize\(\)](#)

Description:

Run the synchronization process. It's mandatory to run authenticate process before to call [GDRIVE_synchronize](#) (only fist time)

33.8.1 Store and Forward variables

gdrive.settings.bEnabled	Set to 1 to enable daily Google Drive synchronization
gdrive.settings.time_param	Enter here the synchronization hour (HH:MM)
gdrive.settings.source_path	Put here the rsync/ssh server address
gdrive.settings.source_token	Enter here the generated token. Please use this url to generate a token for your Google account
gdrive.settings.target_path	Enter here the server folder where Google Drive contents are stored

33.9 CONTENT MANAGEMENT SYSTEM (CMS) FUNCTIONS

CMS lets you automate ePLAYER1 reproduction using a CMS web portal. If you are an ECLER CMS partner you should enter your personal Key in the Key field bellow. Otherwise leave it blank and configure Protocol, Host and Port fields manually to use CMS service with the generic ePLAYER1 CMS API

Functions

Function:

[CMS_reload\(\)](#)

Description:

Reload CMS configuration. Call it after modify the Content Management System configuration using [CFG_set](#) and [CFG_commit](#)

33.9.1 CMS variables

cms.settings.cms_partner_enabled	Set to 1 to enable partner key
cms.settings.cms_key	If cms_partner_enabled is enabled put here your partner key (please contact you ECLER sales rep)
cms.settings.cms_http	If cms_partner_enabled is disabled put here your server protocol (HTTPS or HTTP)
cms.settings.cms_host	Enter here your CMS host address
cms.settings.cms_port	Enter here your CMS host port. If using default port leave it blank

33.10 SCRIPTS FUNCTIONS

CMS lets you automate ePLAYER1 reproduction using a CMS web portal. If you are an ECLER CMS partner you should enter your personal Key in the Key field bellow. Otherwise leave it blank and configure Protocol, Host and Port fields manually to use CMS service with the generic ePLAYER1 CMS API

Functions

Function:

[SCRIPT_reload\(nIndex\)](#)

Description:

Reload selected Script configuration. Call it after modify the Script configuration using [CFG_set](#) and [CFG_commit](#)

Arguments:

nIndex – number indicating the script index to reload. Must be a number between 1 and 20

Function:

[SCRIPT_run\(nIndex\)](#)

Description:

Executes the selected script.

Arguments:

nIndex – number indicating the script index to reload. Must be a number between 1 and 20

Function:

[SCRIPT_kill\(nIndex\)](#)

Description:

Terminates the selected script.

Arguments:

nIndex – number indicating the script index to reload. Must be a number between 1 and 20

Function:

[SCRIPT_status\(nIndex\)](#)

Description:

Retrieves the selected script status

Arguments:

nIndex – number indicating the script index. Must be a number between 1 and 20

Return:

A XML string with script status {"status":"Idle"} or {"status":"Running"} or {"status":"Success"} or {"status":"Failed"}

33.10.1 Script variables

scriptNN.settings.name	Script name or description
scriptNN.settings.bEnabled	Set to 1 to enable this event trigger
scriptNN.settings.trigger	Script trigger type. Could be one of the next values: ON BOOT, ON EVENT, ON PRESET, ON CALENDAR, ON CLOUD DISK SYNC, ON LAN, ON WAN, ON MMC or ON USB
scriptNN.settings.trigger_event	If trigger is ON EVENT enter here the event that triggers the script. You could put here one of the next values: GPI1, GPI2 or SILENCE
scriptNN.settings.trigger_preset_index	If trigger is ON PRESET enter here the preset number that triggers the script
scriptNN.settings.trigger_calendar	If trigger is ON CALENDAR enter here the calendar number that triggers the script
scriptNN.settings.trigger_cloud	If trigger is ON CLOUD DISK SYNC you could select here RSYNC or GOOGLE DRIVE
scriptNN.settings.trigger_network	If trigger is ON LAN or ON WAN enter here the action that triggers the script. Could be AVAILABLE or UNAVAILABLE
scriptNN.settings.trigger_mmc	if trigger is ON MMC enter here the action that triggers the script. Could be PLUG or UNPLUG
scriptNN.settings.trigger_usb	if trigger is ON USB enter here the action that triggers the script. Could be PLUG or UNPLUG

Note: NN indicates a number 01 to 20

33.11 REGISTER (LOG) FUNCTIONS

Register functions that allows the user to write to the ePLAYER1 internal REGISTER

Functions

Function:

[LOG_trace\(txtLog\)](#)

Description:

Add to ePLAYER1 LOG register a trace line

Arguments:

txtLog – String containing the text you want to add. The source of the register line will be equal to “ScriptNN” where NN represents the script calling LOG_trace

Function:

[LOG_warning\(txtLog\)](#)

Description:

Add to ePLAYER1 LOG register a warning line

Arguments:

txtLog – String containing the text you want to add. The source of the register line will be equal to “ScriptNN” where NN represents the script calling LOG_warning

Function:

[LOG_error\(txtLog\)](#)

Description:

Add to ePLAYER1 LOG register a error line

Arguments:

txtLog – String containing the text you want to add. The source of the register line will be equal to “ScriptNN” where NN represents the script calling LOG_error

33.12 DEVICE (DEV) FUNCTIONS

Device functions allows the user to setup firmware and general ePLAYER1 configurations.

Functions

Function:

[DEV_reboot\(\)](#)

Description:

Reboot the ePLAYER1 inmediately.

Function:

[txtVersion = DEV_version\(\)](#)

Description:

Get the ePLAYER1 firmware version

Return:

txtVersion – string containing ePLAYER1 firmware version formatted

Function:

[DEV_update\(urlFirmware\)](#)

Description:

Installs a new ePLAYER1 firmware version. After installation device is rebooted automatically

Arguments:

urlFirmware – Url containing the path where ePLAYER1 firmware resides. Must be a local storage device or http/https url.

Function:

[total, used, percent = DEV_get\(devUrl\)](#)

Description:

Get the ePLAYER1 firmware version

Arguments:

devUrl – url of local storage device. Must be mmc:// or usb://

Return:

total – number representing total number of bytes of external storage device

used – number of bytes representing the used size

percent – number parameter representing the used percentadge

Example:

```
require    "ePLAYER1"
ep=ePLAYER1.new()

print("usb", ep.DEV_get("usb://"))

total,user,percent=ep.DEV_get("mmc://")
print("mmc", total, user, percent)
```

```
usb    7823420    4223365
mmc    15629312    68912    0
```

Function:

```
DEV_boot(nBoot)
```

Description:

Set device boot mode to nBoot

Arguments:

nBoot – number indicating load PRESET1 (bBoot=2) or keep settings (nBoot=1)

Function:

```
bEncrypt = DEV_isEncrypted(devUrl)
```

Description:

Get local storage encrypted flag

Arguments:

devUrl – string must be mmc:// or usb://

Return:

bEncrypt – boolean, true, if local storage is encrypted

Function:

```
bEncrypt = DEV_isEncrypted(devUrl)
```

Description:

Get local storage encrypted flag

Arguments:

devUrl – string must be mmc:// or usb://

Return:

bEncrypt – boolean, true, if local storage is encrypted

Function:

```
bEncrypt = DEV_mac()
```

Description:

Get device MAC address

Return:

jsonMAC – json string representing MAC address value

```
require    "ePLAYER1"
device=ePLAYER1.new()
print(device.DEV_mac())
```

```
{"mac":"A8 40 41 16 47 D6"}
```

Function:

```
bEncrypt = DEV_regkey()
```

Description:

Get device Registration Key code

Return:

jsonMAC – json string representing Registration Key code

```
require "ePLAYER1"
device=ePLAYER1.new()
print(device.DEV_regkey())

{"regkey":"A269FCEAB4F1C20B"}
```

33.13 PANEL FUNCTIONS

Panel functions allows the user to setup firmware and general ePLAYER1 configurations.

Functions

Function:

```
PANEL_set=function(lockMode, password)
```

Description:

Set the panel lock mode and password

Arguments:

lockMode – string indicating lock mode. Possible options are UNLOCK ALL, UNLOCK USER, LOCK ALL
password – string with the LOCK/UNLOCK password. Max length is 8. Valid characters are: 0-9 A-Z .
+ -

Function:

```
lockMode, password = PANEL_get()
```

Description:

Set the panel lock mode and password

Return:

lockMode – string representing lock mode. Possible values are UNLOCK ALL, UNLOCK USER, LOCK ALL
password – string with the LOCK/UNLOCK password

33.14 FINDER FUNCTIONS

Finder functions allows the user to run finder operation on ePLAYER1.

Functions

Function:

`FINDER_set(bLigth)`

Description:

Start or Stop finder operation. If finder is active, ePLAYER1 display blinks.

Arguments:

bLigth – boolean, set to true to start finder operation (display blink). Set to 0 to stop finder operation (normal state)

33.15 LCD FUNCTIONS

LCD functions allows the user to show messages on ePLAYER1 LCD display. The texts are limited to LCD physical dimensions: 2 lines per 16 columns

Functions

Function:

`LCD_print(txtLine1, txtLine2, center, timeout)`

Description:

This functions shows a message on the ePLAYER1 LCD display

Arguments:

txtLine1 – String corresponding to the text on the first LCD line
txtLine2 – String corresponding to the text on the second LCD line
center – Boolean, set to true to center the text on the LCD display
timeout – Number of seconds that message is showed in ePLAYER1 LCD display

Function:

`LCD_print2({txtLine1, txtLine2, bCenter, nTimeout})`

Description:

This functions shows a message on the ePLAYER1 LCD display. Parameters are optional.

Arguments:

txtLine1 (optional) – String corresponding to the text on the first LCD line
txtLine2 (optional) – String corresponding to the text on the second LCD line
center (optional) – Boolean, set to true to center the text on the LCD display
nTimeout (optional) – Number of seconds that message is showed in ePLAYER1 LCD display

34 HOW TO IDENTIFY INTERNET RADIO URL STREAMS

This document shows how to identify internet radio streams in order to play them in audio streaming players such as Ecler ePLAYER1 or Ecler DUO-NET PLAYER. Few examples are given, but there are a lot of different options. Feel free to find your favourite internet radio websites!

IMPORTANT NOTE:

A valid audio stream looks like this:

http://www.my_favourite_radio.mp3

<http://111.111.11.1:8080>

<http://listen.radio/rock.m3u>

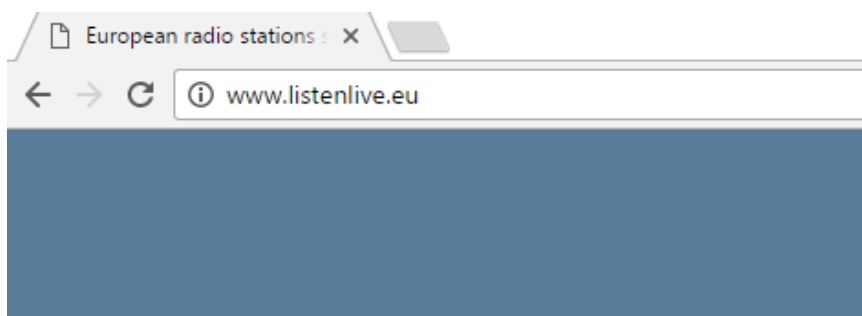
etc.

An address such as “http://www.my_radio.com” is not directly an audio streaming, but a generic website address. This website could include a real audio streaming service, which will have its own URL.

34.1 How to discover streaming URLs included in websites:

1. EXAMPLE 1:

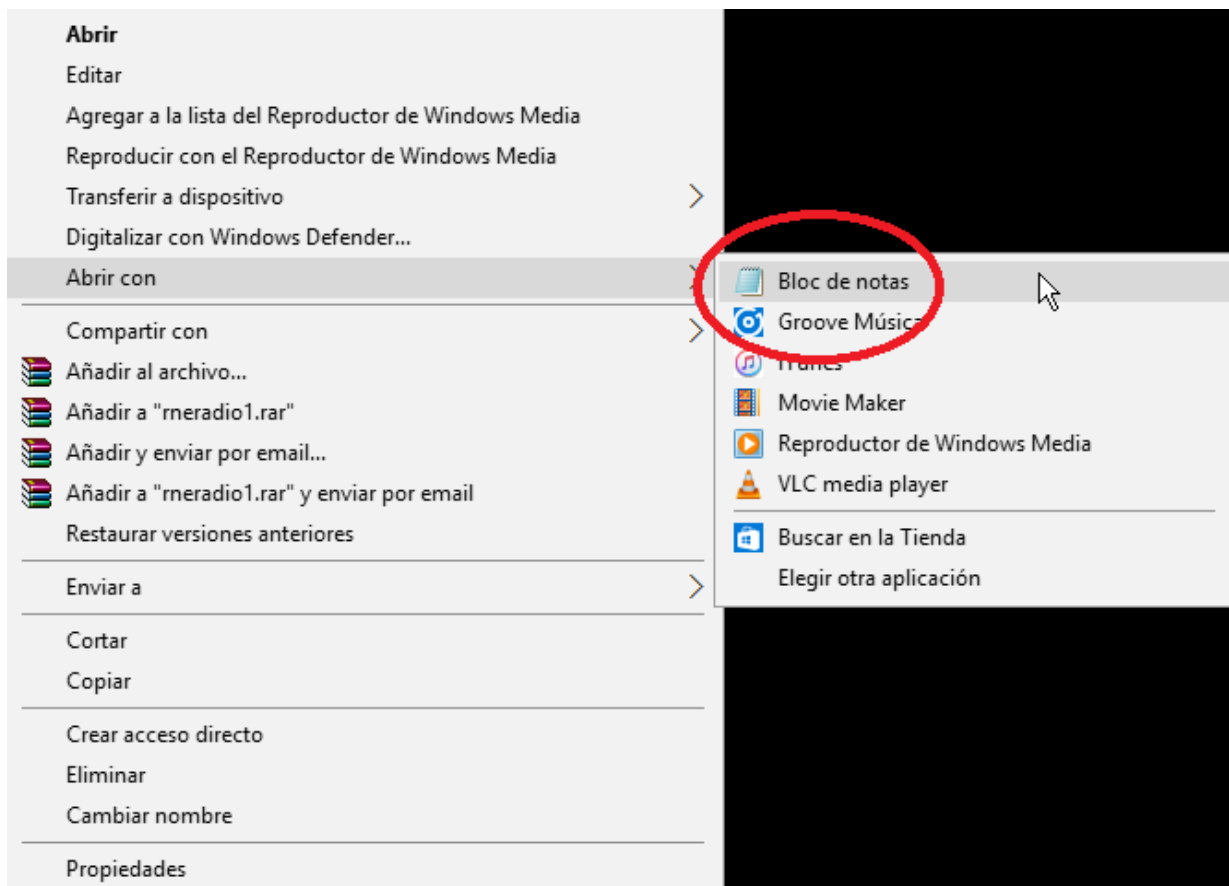
1. Open your favourite web browser.
2. Type (or search for) a live radio manager website. Listenlive.eu is used in this example.



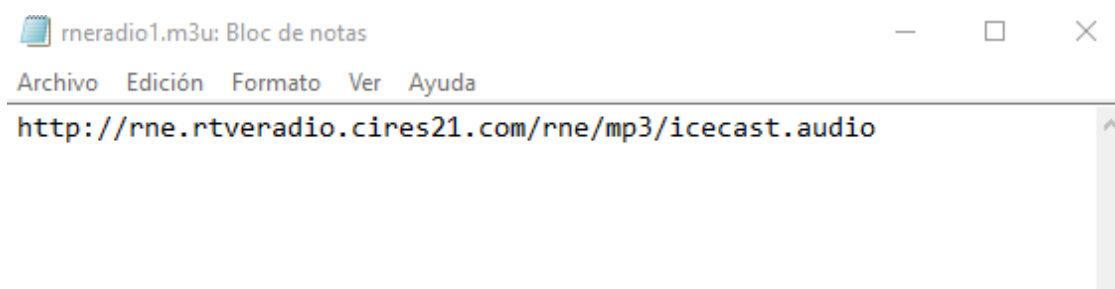
3. In this website, look for your radio and click on the bitrate. An “.m3u” will be downloaded.

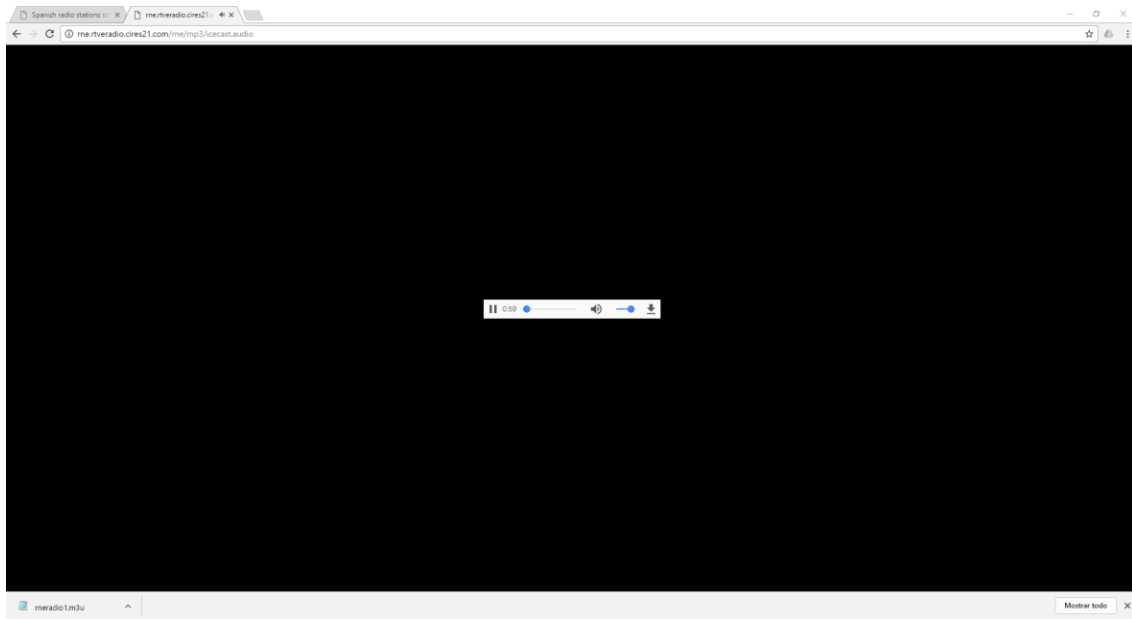
listenlive.eu			
European radio stations streaming live on the internet			
Radio station	Location	Listen Live	Format/Comments
RNE Radio Nacional	Madrid	64 Kbps	News/information/features
RNE Radio Clásica	Madrid	128 Kbps	Classical/cultural programming
RNE Radio 3	Madrid	128 Kbps	News/music for young people
RNE Ràdio 4	Barcelona	64 Kbps	News/music/features (Catalan)

4. Open this “.m3u” with a text editor. You will see the audio streaming URL.



5. To check it, open a new tab in your web browser and copy & paste this URL. If it is valid, it will start to play.





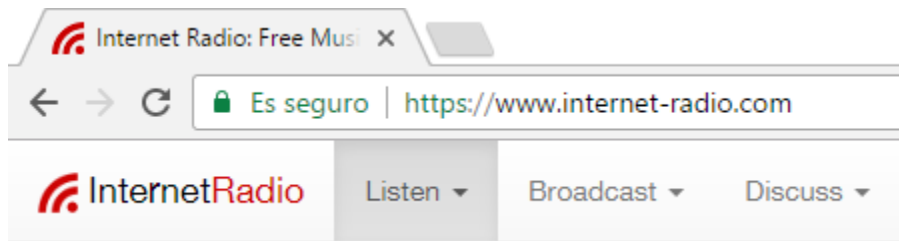
6. Finally, save this address in your audio streaming player (ePLAYER1/DUO-NET PLAYER).

PARAMETERS

Name	<input type="text" value="News"/>
Enabled Events	<input type="checkbox"/> GPI1 <input type="checkbox"/> GPI2 <input type="checkbox"/> SILENCE
Playlist	<input checked="" type="checkbox"/>
Path	<input type="text" value="http://me.rtveradio.cires21.com/m"/>
Media alias	<input type="text" value="RNE"/>

EXAMPLE 2:

1. Open your favourite web browser.
2. Type (or search for) a live radio manager website. Internet-radio.com is used in this example.

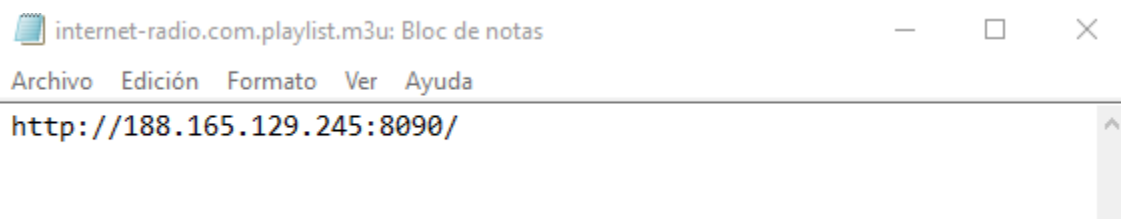
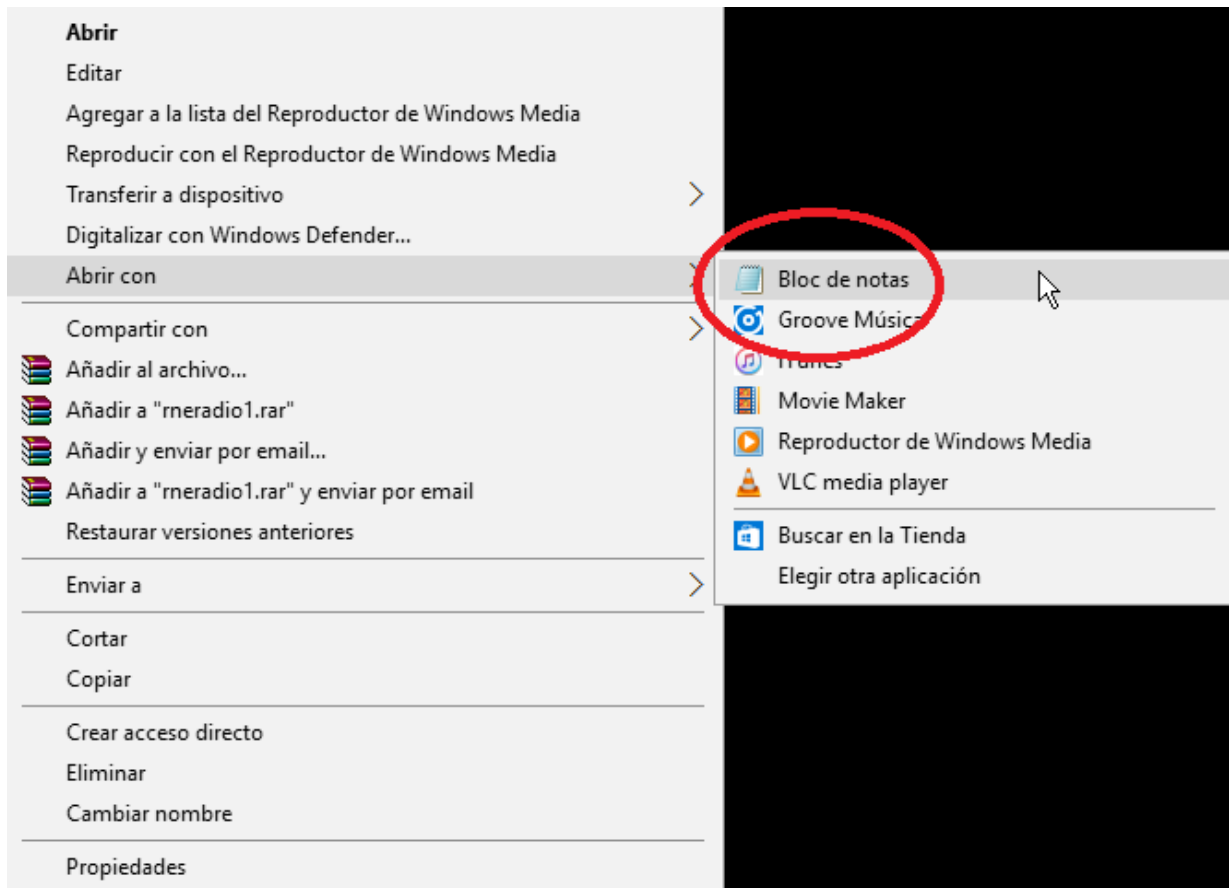


3. In this website, search your radio and click on “.m3u” or “.pls” link. An “.m3u” or “.pls” file will be downloaded.



4. Open this “.m3u” with a text editor. You will see the audio streaming URL.





5. To check it, open a new tab in your web browser and copy & paste this URL. If it is valid, it will start to play.



6. Finally, save this address in your audio streaming player (ePLAYER1/DUO-NET PLAYER).

PARAMETERS

Name

Enabled Events GPI1 GPI2 SILENCE

Playlist

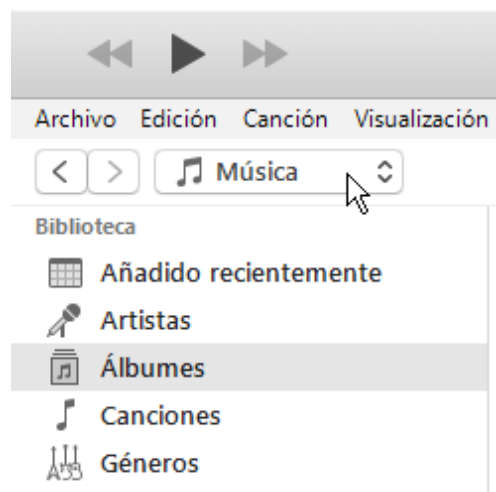
Path

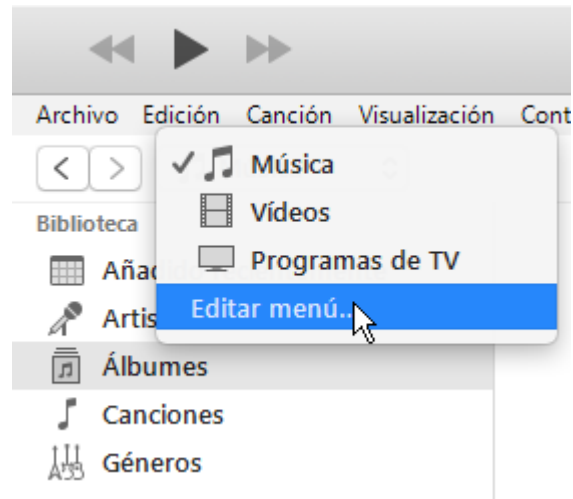
Media alias

34.2 Playing internet radios via AirPlay (ePLAYER1):

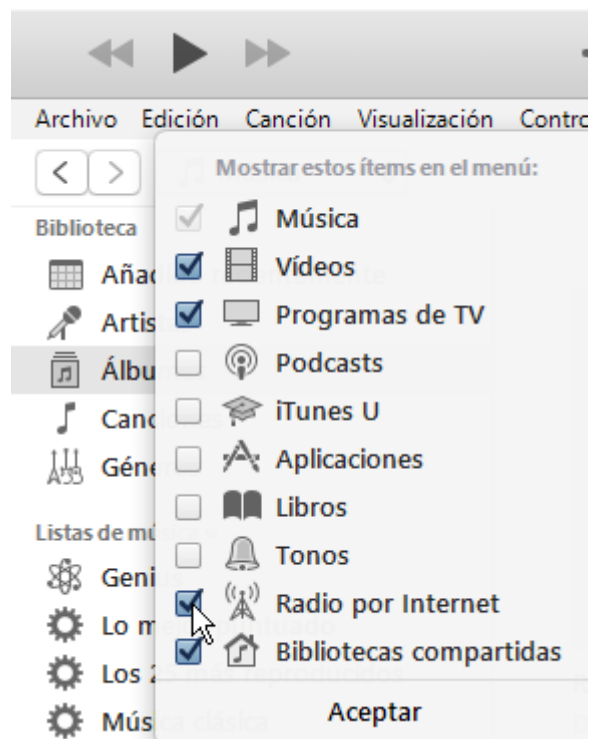
ePLAYER1 must be in AirPlay Mode. Please, consult the user manual to know more about it.

1. Download and install iTunes. Once is installed in your PC, open it and click on Music/Edit Menu.

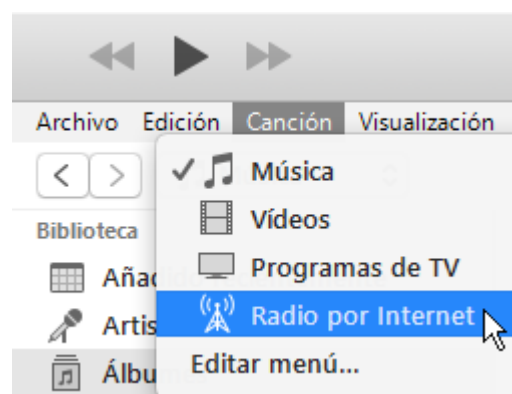









2. Check “Internet Radios”.





3. Now, select “Internet Radios”. You will see a wide catalogue of internet radio streamings.



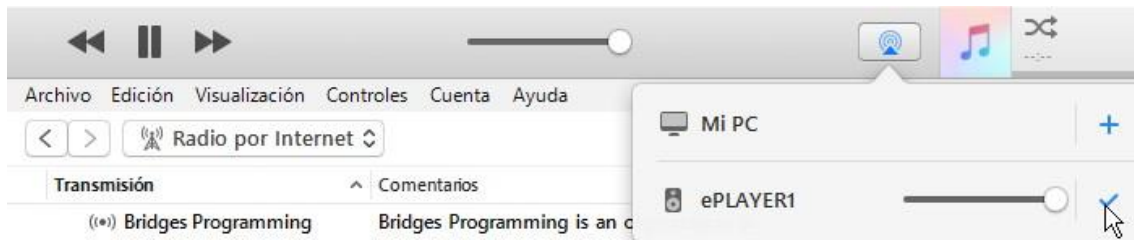






[Archivo](#)
[Edición](#)
[Visualización](#)
[Controles](#)
[Cuenta](#)
[Ayuda](#)

Transmisión	Comentarios
▶ Adult Contemporary	
▶ Alternative Rock	
▶ Ambient	
▶ Blues	
▶ Classic Rock	
▶ Classical	
▶ College/University	
▶ Comedy	
▶ Country	
▶ Eclectic	
▶ Electronica	
▶ Golden Oldies	
▶ Hard Rock / Metal	
▶ Hip Hop / Rap	
▶ International / World	
▶ Jazz	
▶ News / Talk Radio	
▶ Reggae / Island	
▶ Religious	
▶ RnB / Soul	
▶ Sports Radio	
▶ Top 40 / Pop	
▶ '70s Retro	
▶ '80s Flashback	
▶ '90s Hits	

4. Select your favourite radio and click on it. Check on iTunes that ePLAYER1 is selected as the AirPlay device.



5. ePLAYER1 will start to play.



All product characteristics are subject to variation due to production tolerances. **NEEC AUDIO BARCELONA S.L.** reserves the right to make changes or improvements in the design or manufacturing that may affect these product specifications.

For technical queries contact your supplier, distributor or complete the contact form on our website, in [Support / Technical requests](#).

Motors, 166-168 08038 Barcelona - Spain - (+34) 932238403 | information@ecler.com | www.ecler.com