
RadioKit Electron Documentation

Release 1.0

RadioKit Ltd

May 24, 2017

Contents

1	Contents	3
1.1	Getting Started	3
1.2	Supported software and hardware	4
1.3	Installation	5
1.4	Restarting client application	5
1.5	Resetting configuration	6
1.6	Usage	6
1.7	Troubleshooting	8
1.8	Support	9
1.9	ChangeLog	9

RadioKit Electron is a cloud computing platform for routing audio between multiple devices while keeping quality required for broadcasting applications.

Getting Started

RadioKit Electron takes different approach than most Audio Over IP systems present on the market. It was designed with focus on easing both journalists' and sound engineers' lives, so there's a clear distinction between roles of *user* (who is usually a journalist but it may also mean sound device attached to the network) of the system and its *administrator*.

The idea is to keep functionality of users as minimum as it is necessary to operate and provide engineers centralized, web-based control panel so they can supervise and control behaviour of all connected devices from one place.

Device Pool

In this documentation we will refer to the *client* when talking about device that is connected to the network and is capable of playing or capturing audio.

All these devices form your Device Pool, and you can route sound between their audio interfaces. If they contain more than one audio interface, each is treated separately. One client can do multiple transmissions in parallel.

Workflow

The workflow essentially looks like the following:

- Administrator installs client application on all devices (clients) that are supposed to route the sound over the internet.
- Administrator adds all clients to the Device Pool in the web-based Console (control panel). For each device system generates unique one-off key that is valid for 15 minutes.
- User or administrator types appropriate one-off key in all clients, they synchronize with the system.
- Administrator configures desired routing between clients' audio interfaces.
- Administrator remotely turns on or off transmission from/to individual audio interfaces when necessary.

Supported software and hardware

Computers and miniaturized computers

Electron is supposed to work on every PC-like platform. Any computer with Intel CPU will work.

Regarding miniaturized computers, we suggest using [Intel Compute Stick](#). We have been successfully running Electron on Intel Compute Stick model [STCK1A8LFC](#) (with Atom processor and Ubuntu 14.04) and external USB Sound Card for making bidirectional transmission.

Smartphones and tablets

We are supporting Android-based devices, both if they are using ARM and Intel processors.

Operating systems

RadioKit Electron client app is supported on:

- Windows 7 or newer (both 32 and 64-bit)
- Android 4.1 or newer (ARM, ARMv7, ARM64, x86 and x86_64 processors)
- Ubuntu 14.04 and 16.04 LTS (amd64)

It is technically prepared to work, however not released yet for:

- Raspbian Jessie (armhf)
- Mac OS X (Intel 64-bit)
- iOS

Virtual Sound Cards

We have been testing and successfully using RadioKit Electron client app with the following virtual sound cards:

- AXIA IP-Driver for Windows,
- VoiceMeeter for Windows

Hardware Sound Cards

Windows

On Windows any sound card that supports Direct Sound drivers (which mean virtually any sound card) should work.

We are relying on DirectSound instead of chasing the newest drivers such as WASAPI to provide maximum backwards compatibility. Windows 7 are still not uncommon in many radio stations and we need to ensure that these users can use RadioKit Electron. Even if we add support for WASAPI, ASIO at some point, we will provide backwards compatibility for DirectSound.

Sound cards and devices that are known to work:

- ESI Juli@
- Tascam US 2x2

- Behringer XENYX Q1202USB
- Behringer XENYX Q1204USB

Of course we support any soundcard that is detected as Generic Audio Device and built-in soundcards.

Android

On Android we are limited to boundaries of the device and particular operating system build of the device vendor.

It does not limit support for microphone plugged into mic/headphone socket. But that means that newer and more decent phones will support USB sound cards (but only these that are detected as Generic Audio Device and have quite low power consumption), but some will refuse to do so, despite having recent version of the Android OS.

Sound cards/phones known to work:

- LG Nexus 4 + Android 5.0 + Behringer UCA202

Notes

We will do our best to expand the list of devices known to work.

We encourage our users to share their experiences. Please send us an e-mail to info@radiokit.org if you want to contribute.

Installation

Windows

Download and install the latest version of the client [here](#).

Linux

See [installation-linux](#).

Restarting client application

Generally speaking, RadioKit Electron's client application is supposed to run as a background system service, at least on platforms that support doing so.

However, you may want to restart the service.

Please follow instructions appropriate to the client device's operating system:

Windows

Attempt to kill process in the task manager will not succeed. You have to use windows service manager in order to restart the application.

- Log in as a user with administrative privileges
- Press **WindowsKey+R** so "Run" window will appear

- In the Service Manager find **RadioKit Electron 1.2**, select it and start the service using restart button located in the toolbar.

After a while the service should be restarted.

Resetting configuration

If you want to reset configuration of existing device, please do the following steps. That may help if you think that

Please note that the application after this process will recognize that it is running at the same device. If you want to totally remove the device configuration, not only local one, but also one stored in the cloud, you have to also remove it from the Patchbay in the web console.

Windows

- Log in as a user with administrative privileges
- Press **WindowsKey+R** so “Run” window will appear
- Type **services.msc** in the text field and click OK
- In the Service Manager that will pop up, find **RadioKit Electron 1.2**, select it and stop the service using stop button located in the toolbar.
- Delete the file `C:\Windows\System32\config\systemprofile\AppData\Local\radiokit-electron\config.json`
- In the Service Manager find **RadioKit Electron 1.2**, select it and start the service using start button located in the toolbar.
- Once you open <http://localhost:20000> you should see password prompt exactly as it looked like after installing the application for the first time.

Usage

Connecting devices via Patchbay

Patchbay is the place where you can connect and manage all your devices that have Electron client installed on them. To access it follow the next steps:

1. Open the [RadioKit console](#).
2. Click on the **Electron** box.
3. Then click on the **Patchbay** tab.

Here you can see the list of all your accounts. If you have multiple accounts, select the one you want to connect your device with. If you have only one account, the system will redirect you to the Patchbay of your account on its own.

If you are entering the Patchbay for the first time you will only see white empty board. All your devices with Electron client will be shown here after you add them to the Patchbay. Note that in order for devices with Electron client to be able to communicate with each other you have to add each one of them into the Patchbay. A single device with Electron client (example: Android phone) can not communicate with the system if the system (example: your streaming computer) does not have the Electron installed on it as well.

Add a new device in Patchbay

In order to connect your device to the system you have to first add it to the Patchbay. In the Patchbay screen, you will see three buttons in the upper left corner. The buttons are: Add device, Edit device and Delete (garbage bin icon).

- 1. Click on the **Add device** button. A pop-up screen will come up asking you to name your new device.
- 2. Insert the name of your new device (example: Journalist one) and click the Add device button.

Another pop-up screen will come up containing the set up code for your new device.

Bear in mind that each code is valid for only 15 minutes after which you will have to start the set up again if you need a new working code

- 3. Copy or write down your code and click Close button.

Your device will now appear on the Patchbay board. It will look like a blue rectangle and it will have the name you gave it, but **it won't be active until you put the code in the Electron client.**

Insert the Patchbay code in the Electron client

Open your Electron client by opening <http://localhost:20000> in the web browser (Firefox or Chrome) on the computer on which you have installed the application. The welcome screen will ask you to insert the code you have just received in the previous step. Insert the code and click the **Sign in** button. Your client should authenticate and show up after a while in the Patchbay.

Your client will now show you the screen with 2 ports, to capture or playback the sound. However it will still not be able to transmit audio until you complete the next step.

Connecting your devices in Patchbay

After you have completed the previous step and inserted the code in your device go back to the Patchbay in the RadioKit console. Your device in the Patchbay (blue rectangle) will now show all of its audio interfaces (inputs and outputs) in the form of white squares. The number of squares will vary depending on how many audio inputs/outputs your device has. By clicking one of the interfaces of your device, its square will change the color to yellow and allow you to connect it to another device in your Patchbay. Choose the input/output on the second device and click that square. Now you should be able to see a black line connecting these two interfaces (squares). Voila! Your devices are now connected and ready to communicate.

Editing names of devices in Patchbay

If you want to edit the name of one of your devices from Patchbay, simply click on it (it will become red) and click the **Edit device** button in the upper left corner of Patchbay screen.

Disconnecting devices in Patchbay

If you want to disconnect the communication link between your devices in Patchbay, simply click on it (the black line) and click the delete button in the upper left corner of Patchbay screen (garbage bin icon).

Deleting devices in Patchbay

If you want to remove one of your devices from Patchbay, simply click on it (it will become red) and click the delete button in the upper left corner of Patchbay screen (garbage bin icon).

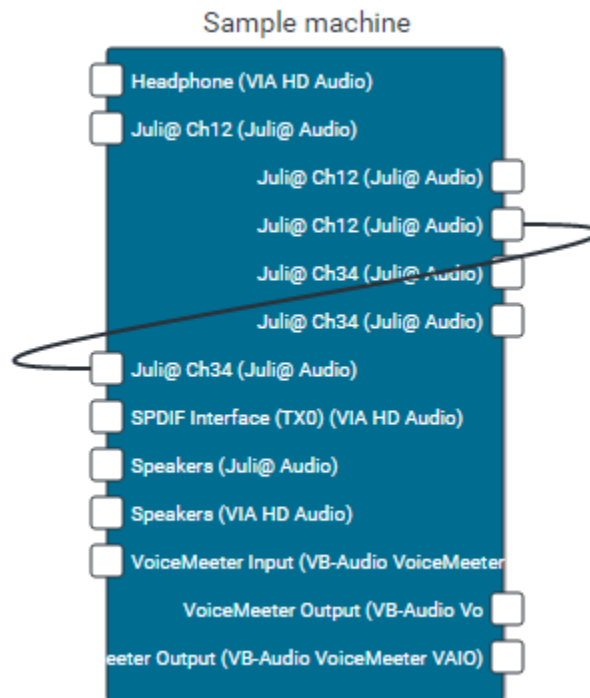
Troubleshooting

Only one channel is transmitted

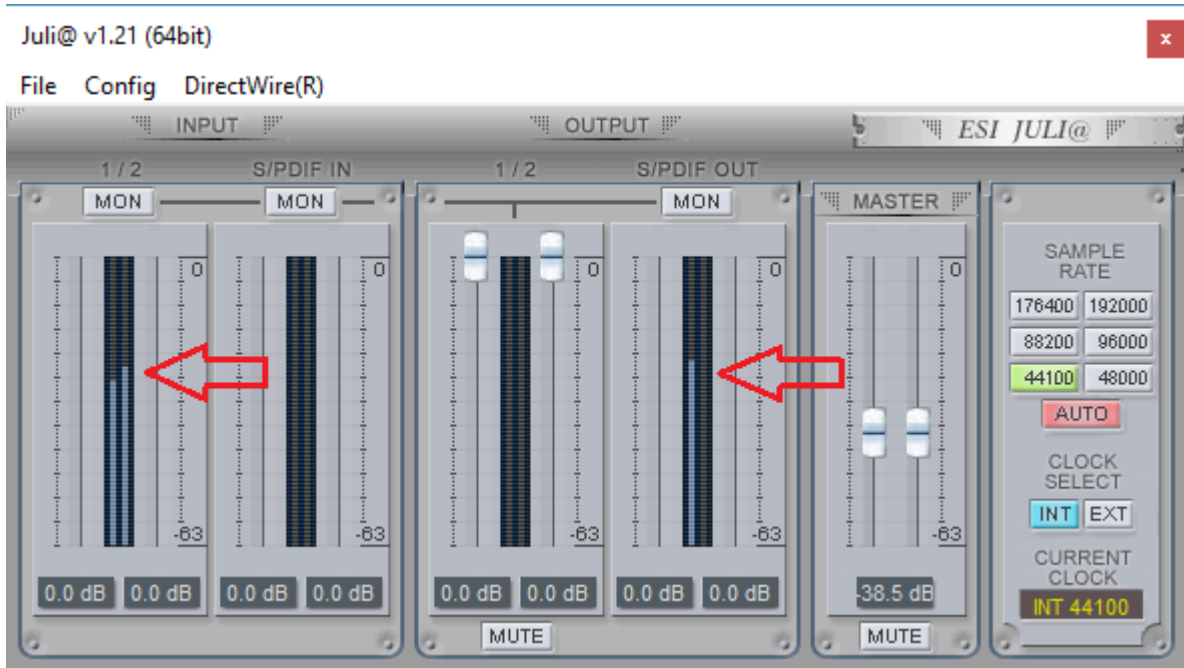
Please check your sound card configuration. If it supports an option to route individual channels between inputs and outputs, ensure that this setup is valid. With some sound cards even peakmeters in the sound card monitoring tool will show stereo input, but it does not mean that what is actually available to the applications is a stereo unless routing is set up.

Such issue is known for ESI Juli@ soundcard. Consider the following example:

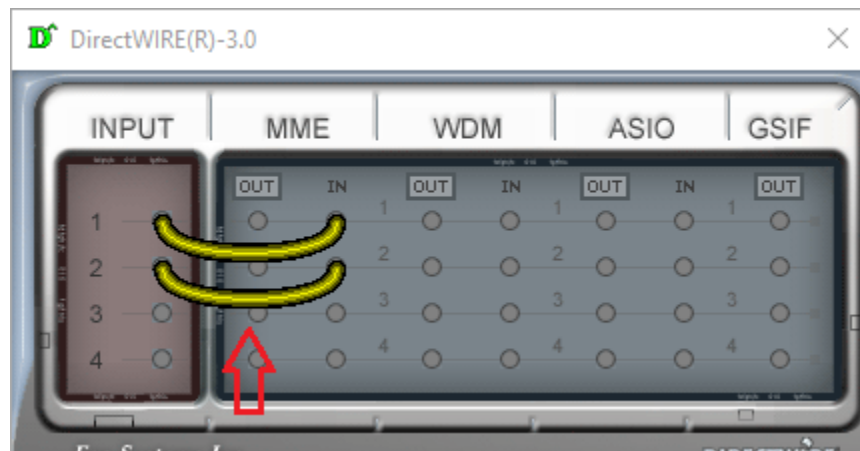
We have set up a loop link in the Patchbay:



so we should hear what is recorded on the *Juli@ Ch12* input on the *Juli@ Ch34* output, routed through Electron servers. But what we actually get by default is stripped to the left channel:



The resolution is to open DirectWire settings:



and link physical inputs with actual driver's inputs. Then it will properly pass stereo data to connected applications.

We are unable to test all existing soundcards but we believe that similar pattern can be present with other multichannel devices.

Support

If you need any support while using RadioKit Electron, please do not hesitate to contact us at info@radiokit.org

ChangeLog

This ChangeLog applies to the application installed on Android devices, PCs etc., not the web-based control panel known as RadioKit Console.

1.0.2 (19 April 2016)

General:

- Fixed bug that prevented transmission from restarting in rare cases.
- Significantly improved stability of audio link from the cloud to the device.
- Numerous stability fixes.
- Enabled Packet Loss Concealment while receiving audio.
- Added IPv6 support.

Android:

- Keep screen on during transmission.
- Properly ask for permissions on Android 6.

1.0.1 (8 April 2016)

General:

- Fixed bug that prevented transmission from stopping if it was reporting error.
- Added missing translations in the user interface.
- Fixed playing back audio feed transmitted from the cloud to the device.
- Reduced CPU usage.
- Increased Opus encoder quality.
- Stability improvements.

Android:

- Fixed bug that was causing crashes while rotating device.
- Fixed invalid launcher icon on devices with HDPI screen resolution.

1.0.0 (4 April 2016)

- Initial release