

Install IPFire on a Raspberry Pi CM4 on DFRobot Carrier Board over Serial Console

This tutorial covers how to install a [IPFire](#) Firewall on to a [Raspberry Pi Compute Module 4](#) that is carried on [DFRobot's Router Carrier Board Mini](#) using a **serial console**.



*Disclaimer: all provided links in this article **aren't** sponsored!*

Tutorial tested against a Raspberry Pi Compute Module 4 (4GB, rev. 1.2) and IPFire Core Update 179. During the setup a second Raspberry Pi 3B+ with local attached keyboard and monitor was used.

→ IPFire's wiki can be found [here](#).

→ And the DFRobot's wiki [here](#).

Preparation / Requirements

- SD-Card (32 GB) + SD-Card-Reader
- Raspberry Pi Compute Module 4 (4GB RAM recommended)
- Jumper Wires (Female to Female)
- Official Raspberry Pi USB-C Powersupply (CM4 requires 5V **3A!**)
- Configured and running Raspberry Pi (either with SSH-access or working display output)

Depending on your Setup:

- *HDMI-Cable*
- *Keyboard*
- *or USB2TTL Adapter*

Download and flash the Image to the SD-Card

Download the aarch64 Flash Image from IPFire's webpage: ipfire.org/download. Then flash the Image on another computer to the SD-Card. Therefore you can use [Win32DiskImager](#) (Windows), the official [Raspberry Pi Imager](#) (Linux, Windows and MacOS) or [BalenaEtcher](#) (Web, Linux and Windows).

- **The compressed IMG is required, the ISO image will not work!**
- If experiencing boot issues when using newer hardware revisions have a look at the official IPFire wiki: wiki.ipfire.org!

Before ejecting the SD-Card perform the following changes:

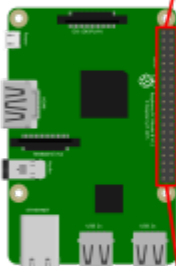
→ Enable the serial console: add (if not present) `enable_uart=1` to `config.txt`.

→ Check if `SERIAL-CONSOLE` in `uENV.txt` is `SERIAL-CONSOLE=ON`.

Connect both Pi's with jumper cables



GPIO#	2nd func.	Pin#	Pin#	2nd func.	GPIO#
2	+3.3 V	1	5	+5 V	
3	SDA1 (I2C)	4	6	+5 V	
4	SCL1 (I2C)	5	6	GND	
4	CLK	9	10	TXD0 (UART)	14
17	GEN0	11	12	RXD0 (UART)	15
27	GEN2	13	14	GND	18
22	GEN3	15	16	GEN4	23
10	MOSI (SPI)	17	20	GEN5	24
9	MISO (SPI)	19	20	GND	25
11	SCLK (SPI)	21	22	CE0_N (SPI)	8
	GND	25	26	CE1_N (SPI)	7



3.3V PWR	1	2	5V PWR
GPIO2 (SDA1, I2C)	3	4	5V PWR
GPIO3 (SCL1, I2C)	5	6	GND
GPIO4 (GPIO_GCLK)	7	8	(UART_TXD0) GPIO14
GND	9	10	(UART_RXD0) GPIO15
GPIO17 (GPIO_GEN0)	11	12	(GPIO_GEN1) GPIO18
GPIO27 (GPIO_GEN2)	13	14	GND
GPIO22 (GPIO_GEN3)	15	16	(GPIO_GEN4) GPIO23
3.3V PWR	17	18	(GPIO_GEN5) GPIO24
GPIO10 (SPI0_MOSI)	19	20	GND
GPIO9 (SPI0_MISO)	21	22	(GPIO_GEN6) GPIO25
GPIO11 (SPI0_CLK)	23	24	(SPI_CE0_N) GPIO8
GND	25	26	(SPI_CE1_N) GPIO7
ID_SD (I2C EEPROM)	27	28	ID_SC (I2C EEPROM)
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21

Now connect the following pins on your carrier board's GPIO and your second Pi's GPIO with three jumper wires (female to female):

Carrier Board (CM4)	2nd Raspberry Pi / USB2TTL	Use
6	6 / GND	Ground (GND)
10	8 / TX	Receive and send
8	10 / RX	Send and receive

When connected properly you can power up the carrier board.

(It is recommended to first power up the second Pi and start the serial console before powering up the carrier board.)

Images sourced from siocours.lycees.nouvelle-aquitaine.pro and wiki.dfrobot.com - CM4 DFRobot Carrier Board

Open serial console using screen

Before you can open a serial connection: serial console must also be enabled on the Pi from which you wish to connect. Therefore check if your `/boot/config.txt` contains `enable_uart=1`. If not, add it at **top (!)** and reboot.

When using a Raspberry Pi with bluetooth as second device also add the following lines to your `config.txt`:

```
# disable bluetooth via uart
dtoverlay=pi3-disable-bt
dtoverlay=pi3-miniuart-bt
dtoverlay=disable-bt
dtoverlay=miniuart-bt
```

This will disable the bluetooth module, because this may break your serial connection as it also uses the uart console.

In general there might be issues with the serial connection, it's recommended to firstly open the serial console on the second Pi and then booting up the CM4. Additionally **do not** dis- and reconnect to a running console session, otherwise your console might not be readable anymore for this session.

Then you can execute the following command from the second Pi to connect to your serial console. **It's strongly recommended to use a local attached keyboard and monitor on a Pi that boots to CLI(!) rather than creating the serial connection via SSH / Desktop.**

```
screen /dev/ttyS0 115200
```

(root-permission required!) → you might install screen before by running the following command:

```
sudo apt install screen
```

If `/dev/ttyS0` doesn't work, use `/dev/serial0`. In case you are using a USB2TTL-Card use `/dev/ttyUSB0`.

Finally perform the setup of IPFire as usual!

With `Ctrl+A` and `D` you can quit the screen-session.

Select correct boot method

When booting for the first time, there will be three entries in the grub bootloader.

Select the 3rd option, that contains serial console!

Adjusting interfaces (important!)

As described also in this [blog post](#) in IPFire's official forum, there are assignment issues with the carrier board's NICs. It seems that the second PCIe NIC gets a self-assigned MAC everytime the device boots up. This is followed by the host OS no longer recognizing the NIC. To solve this issue there is a more or less fancy workaround here in this wiki: [Set persistent MAC for RTL8111/8168/8411 \(r8169\) with udev](#) (internal link).

It's also recommended to completely shut down again and reboot to verify the changes!

Resources used: [cyberciti.biz - linux serial console](#), [scribles.net - uart communication between to Raspberry Pis](#) and [wiki.ipfire.org - Raspberry Pi 4 Model B](#)

From:

<http://fixes.brecht-schule.hamburg/> - Fixes | Public BIT Wiki

Permanent link:

<http://fixes.brecht-schule.hamburg/raspberrypi/ipfire-on-rpicm4?rev=1713629427>

Last update: **2024/04/20 18:10**

