

# How get Windows 10 running on a Raspberry Pi

Written by: Petr Fedchenkov LF Edge Lead Developer, Vladimir Suvorov, LF Edge R&D Lead, and Aaron Williams, Community Manager and Developer Advocate for LF Edge.

## Did you know that EVE is currently the only way to get Xen to run on a Raspberry Pi?

## Well, now EVE is the simplest way to get Windows 10 running on a Raspberry Pi.

To get Windows 10 running on a Raspberry Pi is actually quite easy, but right now there are a couple of limitations that we know of, and probably a couple more that we don't know about yet. But we wanted to get this out there so our community could start playing with it and help us improve upon what we already have. This was taken and expanded from our instructions on how to get started with [Eden](#) on Github.

[EVE](#) on Github.

### Limitations:

- need to run it via RDP or VNC
- Not sure if the RPi's Bluetooth or WiFi is working

### What you will need:

- RPi 4 B- 8 GB is better, 4 GB will work, but it is slow. If you use 4 GB, you will need to lower the GB given to Windows below.
  - This [kit](#) has everything that you need, minus the SD card
- 32GB SD card (at least), you will want a fast one for a camera, this [one](#) is 128 GB
- ~~Networking Cable~~ (Native WiFi is working)
- WiFi name (SSID) and password.

### What to do if you have questions

Don't you hate it when you try to do a tutorial and you get stuck or something isn't clear? We do too. Thus, if this happens, please visit us on our Slack channel [LF Edge #eve-help](#) and ask your question. Since we are a community, we just ask that if you see a question that you can answer, that you do that too.

And while you are there, go to [#eve](#) or [#eve-rpi](#) and tell us what you building.

### Instructions for a Mac

#### Assumptions:

- Need Go installed (version 1.15.3)
- Need docker community (version 2.5.0.1)
- Qemu- version 4 or greater

### Step 0: Setting up your SD card

The first thing that you need to do is to format your SD card. The easiest way is just to go and get the [SD Card Formatter](#) app.

Once you have formatted your disk, lets get it ready for EVE.

1. find your SD cards disk and unmount it

```
diskutil list
diskutil unmountDisk /dev/<diskname> //mine is diskutil unmountDisk /dev/disk2
```

2. We are going to write 0's (zeros) 1 GB - just incase the SD Card Formatter doesn't do a good job

```
sudo dd if=/dev/zero of=/dev/<diskName> bs=1m count=1000 //mine is sudo dd if=/dev/zero of=/dev/disk2 bs=1m count=1000
```

The above step can take a couple of minutes with out feedback from the command line (mine took just over one minute)

## Step 1: Install EVE on Raspberry Pi

EVE will act as the OS for the RPi. You then can add your own containerized OS and application(s) on top of it.

### Prepare Raspberry image

1. On your computer, navigate in your terminal application to the location where you would like place EVE and Eden
2. Clone Eden's github

```
git clone https://github.com/lf-edge/eden.git
cd eden
```

3. Build Eden

```
make clean  <--This cleans up if you have already built something and does nothing if you haven't
make build
```

4. Set Configuration

There are a couple of ways that you could do this. Here are a couple of options. Let's start with WiFi

#### A. No WiFi

```
./eden config add default --devmodel RPi4
```

#### B. With WiFi

```
./eden config add default --devmodel RPi4 --ssid <mywifinet> //no quotation marks around the SSID unless there is a space
in the SSID, the password will be asked for later
```

If you would like to run the Xen hypervisor instead of the default kvm, after doing one of the above, add this:

```
./eden config set default --key eve.hv --value xen
```

5. Build the EVE image

```
./eden setup
```

This is the step that will ask you for the password to your WiFi.

When it is done, Eden will output information about the EVE certificates and the SSID and password that you entered. It will also give you the location of the live.img image that you will put on the SD card.

6. Start Eden

When you are ready, start Eden

```
./eden start
```

### Copy the Image to the SD Card

MacOS:

```
diskutil list
diskutil unmountDisk /dev/<your_disk_name> <--your disk might be unmounted already if you haven't removed after
step 0
sudo dd bs=1m if=fullpath_of_your_image.img of=/dev/<your_disk_name>; sync
sudo diskutil eject /dev/<your_disk_name>
```

The copying might take a minute or two, depending on your machine. You will not see anything on the screen until it is done. For me, it takes about 30 seconds.

## Step 2: Onboarding the RPi and Running a Small Application to Test It

1. unplug anything attached to the USB on the RPi (keyboards, mouse, etc.)
2. Put in SD card into Raspberry and power it on
3. Onboard our RPi into eden so that we can control it remotely

On the terminal window of your computer, type:

```
./eden eve onboard
```

This can take 5 minutes. It will output a large JSON file when it is done.

When your RPi is onboarded, type the below to get the status of Eden

```
./eden status
```

4. Deploy the nginx server, as a test

```
./eden pod deploy -p 8028:80 docker://nginx
```

5. run eden's status command to get the RPi's IP address

It is listed under "EVE REMOTE IPs"

```
./eden status
```

6. Open a browser and point it to `http://EVE's_IP_Address:8028`

### Step 3: Pushing Windows to your RPi

1. The final part is to push Windows from Eden to EVE on your RPi.  
`./eden pod deploy -p 8027:3389 docker://itmoeve/eci-windows:2004-compressed-arm64 --vnc-display=1 --memory=6GB --cpus=2`

The memory = 6 GB flag means that you are telling EVE to give Windows 6 GB of RAM, thus if you are running on a 4 GB RPi, you will want to reduce this to 3 GB. The more memory that you give to Windows, the better it will perform, but you will want to save 1 GB for EVE. EVE only needs a little less than 500 MB, but save 1 GB. Also, remember if you are going to put other applications on your RPi via EVE (say you want to also install Ubuntu), you will need to save memory for them.

This is the longest part of the install. It can take about an hour to download and then about 10-15 min for it to be installed and ready to go.

2. Deploying Windows to EVE will take a while. To check on the progress, use the pod ps command.  
`./eden pod ps`

The results will start with the "percent downloaded", then "loading", "creating volume", and "finally running".

### Step 4: Using Microsoft's Remote Desktop to Connect to Your Pi

1. Download and install RDP for your OS from [here](#).
2. When you set up your connection make sure you use the port 8027. (XX.XX.XX.XX:8027)
3. Username: IEUser and Password: Passw0rd!

### Step 4b: Using VNC to Remote into Your Pi

You will need to download VNC to your laptop. Then use the IP address of your Pi, plus the port of 5901 192.123.1.14:5901

### Step 5: Play

You now have Windows running on a Raspberry Pi! Pretty cool, eh? Please visit us on slack to tell us what you have done.

This was mostly done as a PoC, so we are really looking for help from the community to improve this project. Here are some ideas from us:

- What about Android? or Ubuntu? (yes for Ubuntu you could put it natively on the RPi, but if you are running an event with 10-50 RPi's, it would be much nicer to reset everything via EVE than to re-burn all of the SD cards.)
- What about running EdgeX or Fledge on Eden/EVE? The instructions would basically work. We just need to work on what ports to turn on.

Limitations/things we haven't tested as of Nov 25, 2020

- Using VNC- Lining up of the mouse. This is really a limitation of VNC. RDP works well.
- WiFi/Bluetooth- We haven't tested using the native WiFi or the native Bluetooth on the device, but it doesn't seem to be working
- When the Pi is plugged into a monitor, you see EVE and not windows. We are working on this.
  - Plugging in a keyboard and mouse interacts with EVE (same as above)
- GPIO pins, they might work, but the limitation is probably on Windows. We need to test this.
- USB- They are not quite working yet, we are working on it.