Dual Booting Haiku R1/beta4 and UEFI Windows 11 on bare metal using rEFInd

Last edited 5/10/2025 by Solvitor

The following is how to do what the title says, **using only software on your Haiku R1Beta5 Live instal (on your USB) and your non-secure boot Windows 11 box** to get Haiku onto your existing Windows 11 box. This is the path an average noob migrating from *Windows*, with little *Haiku* knowledge and not coming through the Linux route will probably attempt. That means no *GParted*, or understanding of *Haiku* and EFI.

Given I was running *Haiku* on a fast USB3.1 Stick and USB3.1 Port, I am not sure it was worth it for a Beta release, as I doubt the average *Windows* user would find it an adequate replacement daily driver. I needed the BeFile System to be as fast as possible and large as possible.

This guideline is simply a record of what I did (over a few days of bashing my head) and is not elegant and probably has stuff that isn't needed or documented right, but it hopefully makes it "easy" for you to get on *Haiku* bare metalled. I followed what others far cleverer have kindly put online.

Caveat: This is written by an non-expert noob with limited documenting ability who believes it should work, because it did for him. He can not be held responsible in any way if you wipe your Windows instal, personal data or other undesirable outcomes. If you do not have the skills to protect against losses and bad outcomes do not follow these instructions. This is very much what I did FYI. If you want any form of recourse for things that go wrong use Windows/Apple.

The procedure is not relevant to legacy MBR instals, but the solution to the *no boot path: "Select boot volume (Current: None)"* problem covered at the end (The solution) is the same issue from what I read, so the same fix should work.

The procedure follows Matt Lacey's clear guide https://mattlacey.com/posts/2022-12-30-booting-haiku-and-windows/ until *6A: The Fix*. I have simply written what I figured, that other noobs might find helpful.

The official Haiku Website provides https://www.haiku-os.org/guides/uefi_booting/ which is to create a HHD/SSD from scratch and is equivalent where it is not the dual boot with *Windows* instal.

Step 1: instal Haiku to USB then

Get *Haiku R1Beta5* (haiku-r1beta5-x86_64-anyboot.iso) booting from a Live USB drive instal. This was dead easy and is well explained everywhere (although I used RUFUS, because Balena Etcher does not have a 32Bit version). Set up your Live USB, use it, instal what you want, and set it up however you like. Because it is persistent and *Haiku Installer* instals your current complete set-up configured as is, your instal boots up on the target PC as you left it (brilliant). If you fill it with lots of large files and applications, expect subsequent instals to take longer than a few seconds to copy. Figure out the best balance between checking out what *Haiku* can do for you (testing packages) on your live USB, before going to a bare metal instal.

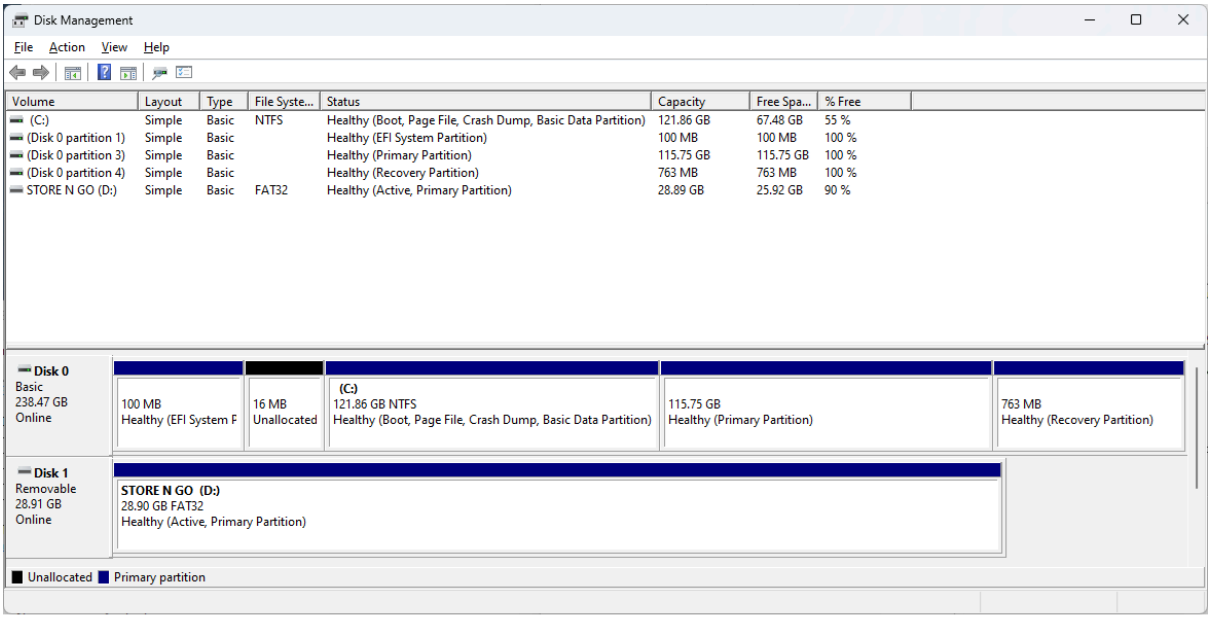
Check your USB Live instal works on the target machine. Given the limited drivers and hardware compatibility this is necessary. I found the *Haiku installer* would not recognise the internal hard drive/SSD on a couple of my cheap NUCs, though I have never had a refusal to boot.

Step 2: Create your partition

Before starting, my experience was that *Windows 11* bloatware on 64 GB is insufficient to do anything except house the bloat and daily enforced bugfixes. It is a major reason I am looking at Haiku, even though it is chalk and cheese when it comes to "out of the box“ “just works”.

64GB seems huge for testing the current *Haiku* release. My *Haiku* instal, which has more installed apps than most testers would load, is currently using 25GB on disk. If you were doing something like a large database you should add your data files to the size you require. I do not know BeFS yet, but I read that BtrFS gets nasty if you exceed 80%, so 32GB seems adequate for Haiku testing IMHO. The ISO is only 1.37GB (the size of a light desktop and software suite Linux ISO), so if you are doing something small it can be fitted in much less.

You will need to shrink existing partitions[1] in *Windows* using *Disk Management* and create your *Haiku* partition, if you do not already have one. Only one extra partition is needed because the boot partition already exists. The file system format you select does not matter as the *Haiku Installer* will prompt reformatting to BeFS using *DriveSetup*.



The new partition is the target volume to select when running *Haiku Installer*, so ensure you know how to identify it, and not the *Windows* volume (Don't overwrite your *Windows* instal).

Step 3: Run *Installer*

Reboot into your Live *Haiku* USB instal where you will find *Installer* in the Applications menu.

Details for "installing" *Haiku* are available on the *Haiku* website, but it is as simple as running *Installer* and filling in the blanks.

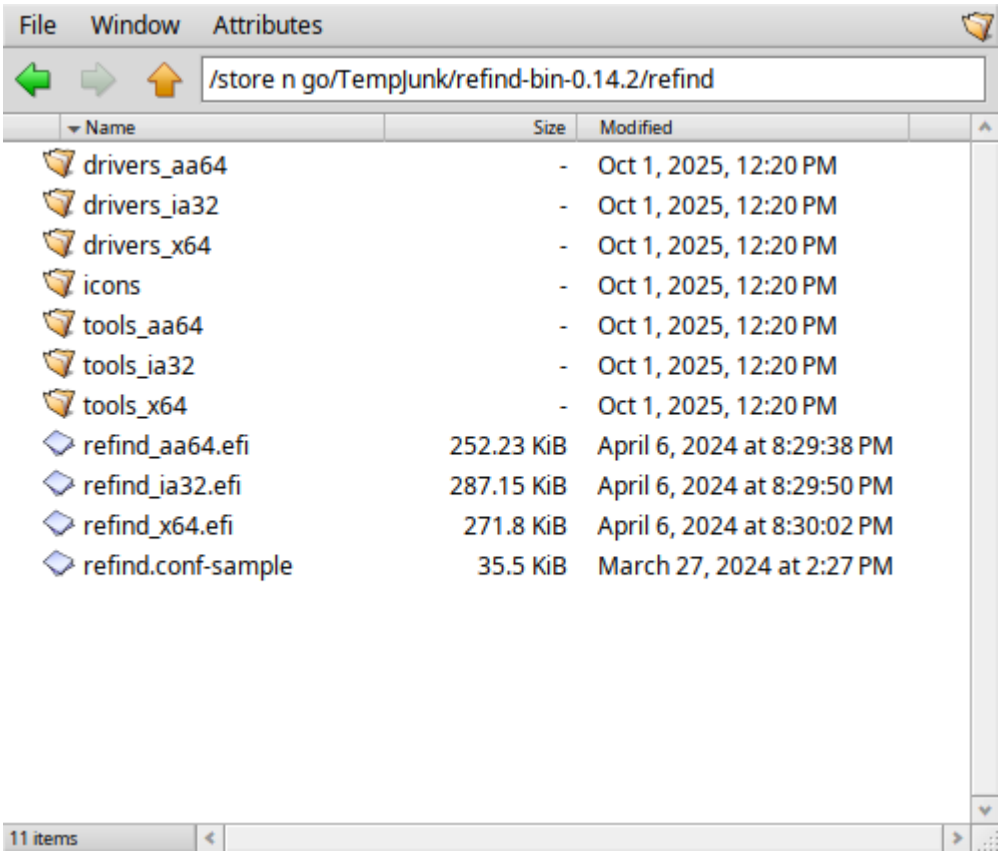
---

[1] Good luck with partition creation as Windows writes hidden files to block you doing just this (at least it seems like that) and this was my biggest single time waster. Dr Google "You cannot shrink a volume beyond the point where any unmovable files are located"
provides loads of advice. In my case it was \$Extend\$UsnJrnl:$J:$DATA hidden file found by looking in Event Viewer at the latest Defrag entry (On a SSD, Defrag is actually a Trim operation). That required the Page File System turning off, but there was a lot more I turned off before I finally did the Latest Defrag entry check. Doing that first, and Googling the offending file would be my advice. Type "Event Viewer" in Search in the Windows Bar and mouse click your way from there. I found most things that way because Windows has so many settings, accessible in so many ways, it is now hard to find settings using the menus.
The 16MB Unallocated block seen in the Disk Management image inexplicably "appeared" when I was hacking to get this partition created with Partition type Attribute "BeFS file System" and I could not figure out how to concatenate it back. It should not be there.

While flashing the ISO to the USB was fully automated, getting *Haiku* to boot from an internal drive is not. Running *Installer* does not set up the booting. That is why manually installing *rEFIind* is required.

Step 4: Download *rEFIind*

You can find details on http://www.rodsbooks.com/refind/getting.html and download the "binary zip file" (Link provided to GitHub). Extract the ZIP file and in *Tracker* view the extracted contents that you need in `…/refind-bin-0.14.2/refind.`



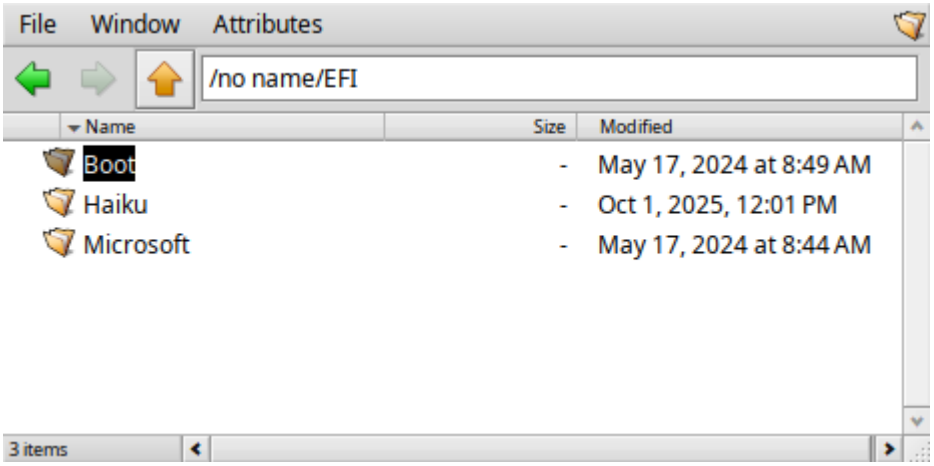
Step 5: Manually instal *rEFInd* and *haiku_loader.efi*

Installing rEFInd on *Haiku* requires you to manually copy the correct files for your computer architecture to its existing EFI partition (exists already to boot Windows). (The included installers are sadly for *Linux* and *OSX* only, but maybe some wonderful developer will fix that). The instructions below show the X64 files for 64Bit machines.

Mount the EFI partition in your *Haiku* USB loaded system and identify the correct partition using *DriveSetup*. It is the one with *Partition type*: *EFI system data*. Usually at the start of a disk, *volume name: non name, 100MiB, FAT 32, Read Only* in my case. Close *DriveSetup* without making any changes

Right-click on the desktop and *Mount* the EFI partition as Read/Write. When opened in *Tracker* you will find a directory called EFI in the root directory, and inside that the *Microsoft* directory.

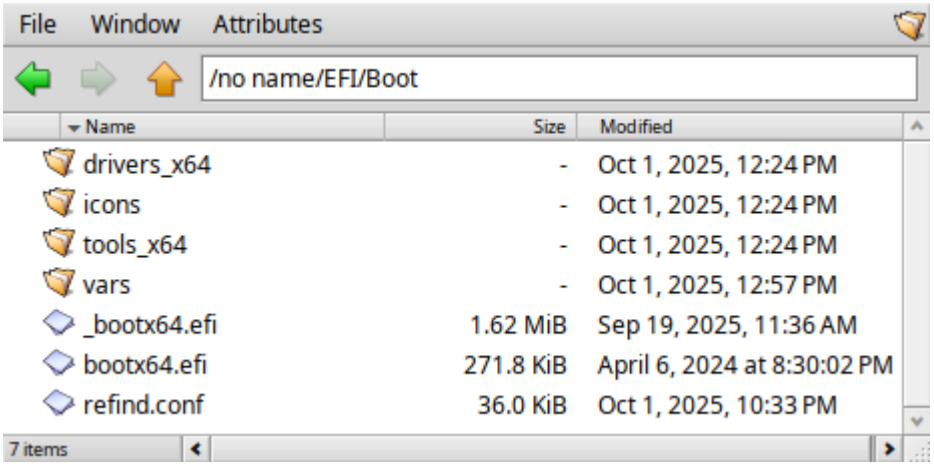
Within `/EFI` Create new folder `/Haiku` and copy the *Haiku* EFI bootloader `haiku_loader.efi` from `/boot/system/data/platform_loaders/` into it.



Now CD to `/EFI/BOOT`. Rename the existing `boot/bootx64.efi` (in case something doesn't work and you need to boot as before) or delete it if you don't care. Copy `refind/refind_x64.efi` to `/EFI/BOOT` and rename it `bootx64.efi`

Copy `refind/refind.conf-sample` rename it `refind.conf`

Your finished `EFI/Boot` directory will look as follows:



I think the drivers are not needed as they are for other file systems, and the `tools_x64` contains `gptsync_x64.efi` which I do not think `haiku_loader.efi` uses.

The directory structure shown is according to Matt's guide. I do not think the boot loaders will find the files so long as they are in the partition.

Step 6: Test

Shutdown *Haiku* Live, reboot with your USB removed and it should, after a delay, display the colourful *rEFInd* boot menu, with *Haiku* and *Window* menu icons, plus other options there for clicking.

Because rEFInd searches so well for other boot options it will display anything it can boot (`.efi` files it finds), including your Live *Haiku* USB if you leave it inserted (Icon is Haiku with a USB symbol on it: very elegant.)

But it is only a partial success :-(

You will almost certainly get Haiku Boot manager telling you “no bootpath is set”, and this is where Matt’s guide ended!

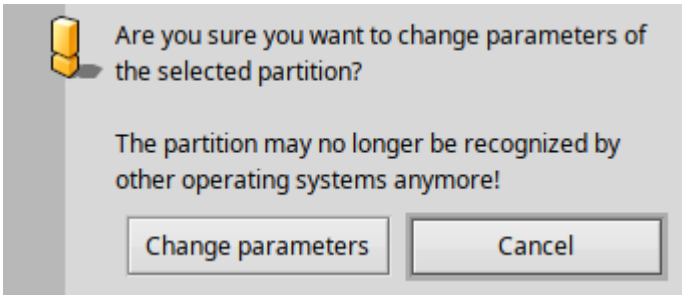
Haiku Boot Loader will kindly find it for you as you just click away at the options it throws up (it does not provide a default *“boot to Current: Latest State”*) and you will then return to *continue booting* and voila: you have *Haiku* running, but it is a pain to do this for every boot!

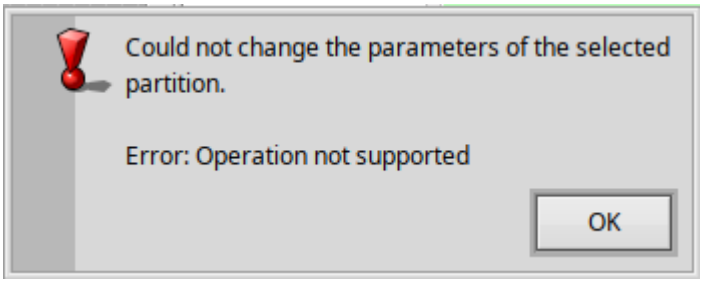
Step 6A: The Fix

Jump to The solution if you are rushing

After a couple of days Googling, repartitioning, reinstalling and hacking to get the *Partition type* in *DriveSetup* to show “Be File System” , I conclude that with *Windows 11 Disk Management* and *Haiku Beta5 DriveSetup* it is impossible to get the *Partition*

type right. Maybe it was the “boot” partition types I selected (based on Googling my impression was you had to make the partition bootable, but this is not relevant to EFI), When attempting to set it *Haiku DriveSetup* returned:





despite it being an unmounted, unused partition. If *Haiku DriveSetup* actually would change the *Partition type*, the whole *Haiku* installation and setup would have been a stroll in the park.

I agree that **@waddlesplash** Mar 2023 https://discuss.haiku-os.org/t/could-not-find-boot-partition/13155/3 is correct: This *DriveSetup* limitation appears to be an unfortunate bug, causing the grief.

The solution was to finally implement the simple but “hard Core” @Javanx fix. From the same thread.

@Javanx Feb 2023 took the partition GUID from haiku/gpt_known_guids.h at 8f16317a5b6db5c672f331814273e5857555020f · haiku/haiku · GitHub

Then using DISKPART in Windows CMD[2] to set the partition type id. Start DiskPart, `select drive ??, select partition ??` and then `set id=42465331-3BA3-10F1-802A-4861696B7521`. Read the help files and warnings about “OEM installers use only”, but any hacker of modest reading ability can do it (back up stuff you don’t want to lose first).

[2] In the Windows start bar Search type command, and you'll see Command Prompt . Choose the "Run as administrator." option at the right, or Right-click that Command Prompt and do the same.

@Javanx has done all the clevers[3], so you just need to cut and paste the number he provided! (`42465331-3BA3-10F1-802A-4861696B7521`)

Haiku should now correctly boot though the *rEFInd* boot menu without having to manually choose the path each session.

If you check in *Haiku Drive-Setup* you will see the *Partition type* is now identified as "Be File System".

Well Identified Problem of "Select boot volume (Current: None)"

https://discuss.haiku-os.org/t/solved-why-no-boot-path-found-is-occurred-on-efi-booting/16476 and other threads all point to the partition type being incorrectly set for the auto search Haiku .efi boot algorithm. Everyone has identified the failure of *DriveSetup* to allow the *Partition type* Attribute when you reformat to BeFS as A MAJOR UNNECESSARY complication in getting Haiku to boot smoothly on a bare metal instal. The *Windows DISKPART* route is explained above, while others cover using other partitioning software to change the *Partition type*. The post saying use GParted in the procedure work I think because GParted generates a partition type that `haiku_loader.efi` checks (I did not have GParted so can not definitively say).

Set 6B: Editing the rEFInd Configuration

It all works automagically, but if you want to improve, this is one place you can do it.

When trying to bypass the **"no bootpath is set"** problem I hack edited `/EFI/BOOT/refind.conf`. You can manually pass options and provide paths in this file, so it seemed a possible solution.

As an example you can search/find the Windows entry (at the bottom). Ensure the loader and icon files specs match where you put them.

```
menuentry "Windows" {
    loader \EFI\_Microsoft\Boot\bootmgfw.efi
    icon \EFI\BOOT\icons\os_win.png
}
```

Cut and Paste the Windows entry to proforma a Haiku entry

```
menuentry "Windows" {
    loader \EFI\Haiku\boot\haikuloader.efi
```

---

[3] @Javanx points out the GUID is formatted slightly differently in Haiku codebase as it's divided in 4 chunks, while diskpart wants it in 5 chunks, but it's as easy as adding the extra dash.

```
        icon \EFI\BOOT\icons\os_haiku.png
    }
```

In addition to *rEIDind* the automagically found bootloaders, you will find two more menu selections now load per the above given instructions (Must add *manual* is added to *scanfor* ie: *scanfor manual,internal,external,firmware*). These extra menu entries make it clear what is being automated. You can turn off the automagically found menu options by changing the search options in `refind.conf`.

There are many settings in `refind.conf` that look like they could be tweaked to increase boot speed and make things work better for *Haiku*. One simple tweak is to remove the commenting # from in front of #enable_mouse and you can now select OS menu options with your mouse (see the mouse monitor in the image)! The *rEFInd README.txt* contains good explanations. *rEFInd* has most file systems covered but not BeFS, so I suspect that may be partly why my redirection attempts failed, along with not having a clue what arguments that *haikuloader.efi* accepts (I was copying the Linux path examples).

Maybe a *Haiku* expert can post an “optimal” `refind.conf` and even provide resources for rEFInd to incorporate Haiku as a known system.

Notes: If you Open your finished Haiku Partition, in Windows Disk Management it will show it as a RAW file system because it does not recognise BeFS, but at least it will warn you that it may contain something it doesn’t know about, before you do anything to it.

https://medium.com/@probonopd/my-fourth-day-with-haiku-installation-and-boot-woes-bd4f637c3ba0