
GhostBSD

Eric Turgeon

Nov 13, 2021

USER GUIDE

- 1 Overview 3**
- 2 Our Goal 5**
 - 2.1 Getting started 5
 - 2.2 Installation Guide 9
 - 2.3 Getting to know the system 9
 - 2.4 Troubleshooting 9
 - 2.5 Frequently Asked Questions 9
 - 2.6 Providing feedback 11
 - 2.7 Get Involved 11

This documentation portal has all of the information users need to get going with their journey using GhostBSD. It also contains information for people that want to help the community, report bugs, and contribute to the project.

OVERVIEW

GhostBSD is a simple, elegant, and friendly BSD operating system for desktops and laptops based on FreeBSD. GhostBSD is a slow-rolling release while some GNU/Linux distros are on the bleeding edge side; we tried to offer a stable update and release cycle. The official desktop environment is MATE. The system comes with a graphical application to install software and update your system. Most codecs to play multimedia files are pre-installed. The installer leverages OpenZFS makes it easy to install and is suitable for newcomers to BSD. With modest hardware requirements, GhostBSD is ideal for modern workstations and 64-bit single-board computer hardware.

OUR GOAL

In general, BSD is also considered beyond the average computer user's knowledge. We try to simplify BSD to lower the entry-level of using BSD on a desktop or laptop. We provide all the benefits of the FreeBSD operating system benefits combined with the OpenRC inits system.

2.1 Getting started

2.1.1 Status

GhostBSD is available for general use.

2.1.2 System requirements

- 2 GHz dual core Intel/ARM 64-bit processor
- 4 GiB RAM (system memory for physical and virtualized installs)
- VGA capable of 1024x768 screen resolution
- Network connection
- 15 GiB of storage
- Either a CD/DVD drive or a USB port for booting the installer media

In the future, there may also be builds for other processor architectures. We would like to bring down the RAM requirement considerably.

Please refer to [FreeBSD Hardware Compatibility](#) for more information on individual components.

Tested hardware

GhostBSD is known to boot to a graphical desktop on the following machines. Auxiliary functionality such as wireless networking, sound over HDMI, sleep, graphics acceleration, etc. has not yet been tested systematically.

Please contact us if you would like to sponsor the project with a hardware donation. We are especially looking for Lenovo devices from the previous generations that should be available second-hand inexpensively.

To see Hardware Probes of systems running GhostBSD, please see the [GhostBSD Hardware Database](#) provided by [bsd-hardware.info](#). It is reasonable to assume that every system listed there can at least successfully boot GhostBSD. Auxiliary functionality such as wireless networking, sound over HDMI, sleep, graphics acceleration, etc. may or may not be working.

Networking hardware

Not all networking devices may be supported by GhostBSD yet. In those cases, you may want to consider using a USB based networking devices. GhostBSD developers currently have access to the following USB based networking devices which are known to work:

- USB 802.11n WLAN Adapters based on ID `0bda:8176` Realtek Semiconductor Corp. RTL8188CUS
- USB Wired Ethernet Adapters based on ID `0b95:772b` ASIX Electronics Corp. AX88772B

Virtualization environments

We recommend running GhostBSD on real hardware ("bare metal") if possible. This [↪](#) should give you the best possible performance and hardware support.

Users have reported success in running GhostBSD in the following virtualization environments:

- VirtualBox host (on FreeBSD and macOS), known to work in BIOS and EFI mode
- VMware host (on Windows), possibly only working in BIOS mode?
- QEMU host (on Linux), works in both BIOS and EFI modes (see below). Note that for acceptable performance, QEMU needs KVM which is currently not available on FreeBSD hosts yet
- Parallels host, reported to work in EFI mode (see below)
- Proxmox VE

Please note:

- The VM needs to be **64-bit**
- The VM needs **at least 4 GB of RAM**
- The VM needs **at least 2 CPU cores**
- The boot process takes longer than you might expect; boot in verbose mode to see the details
- For best results set **EFI/UEFI** boot mode (not BIOS)

Please report back about the results on your virtualization environment.

QEMU

Create an 8 GiB (or larger) `ghostbsd.img` image file on which you can install the system:

```
$ pwd
/home/user
$ mkdir -p .qemu/ghostbsd
$ fallocate -l $(( 8*1024*1024*1024 )) .qemu/ghostbsd/ghostbsd.img
```

Then, boot GhostBSD:

```
qemu-system-x86_64 -machine type=q35,accel=kvm \
-enable-kvm -cpu host -smp 2 -m 4096 \
-device virtio-net,netdev=vmnic -netdev user,id=vmnic,hostfwd=tcp::5222-:22 \
-vga std -soundhw hda -no-quit \
-drive format=raw,file=${HOME}/.qemu/ghostbsd/ghostbsd.img \
```

(continues on next page)

(continued from previous page)

```
-drive format=raw,file=${HOME}/Downloads/ghostbsd.iso \  
-boot menu=on
```

When QEMU starts, press `esc` and select 2 to boot the ISO.

Use the **Install GhostBSD** utility to install GhostBSD do the disk image.

Then restart QEMU, you now remove the last two options from the above command.

Notes

- The `host.fwd=` option creates a port forward from your host port 5222 to the Qemu VM port 22.
- Unfortunately the `qemu-system-x86_64` USB tablet options do not work; you will need to press `Ctrl+Alt+g` to release the mouse pointer from the QEMU window
- To make QEMU full screen, press `Ctrl+Alt+F`

To boot/install GhostBSD in UEFI mode, first install [OVMF Open Virtual Machine Firmware](#) on your host side. The package name for Fedora 32 is `edk2-ovmf`

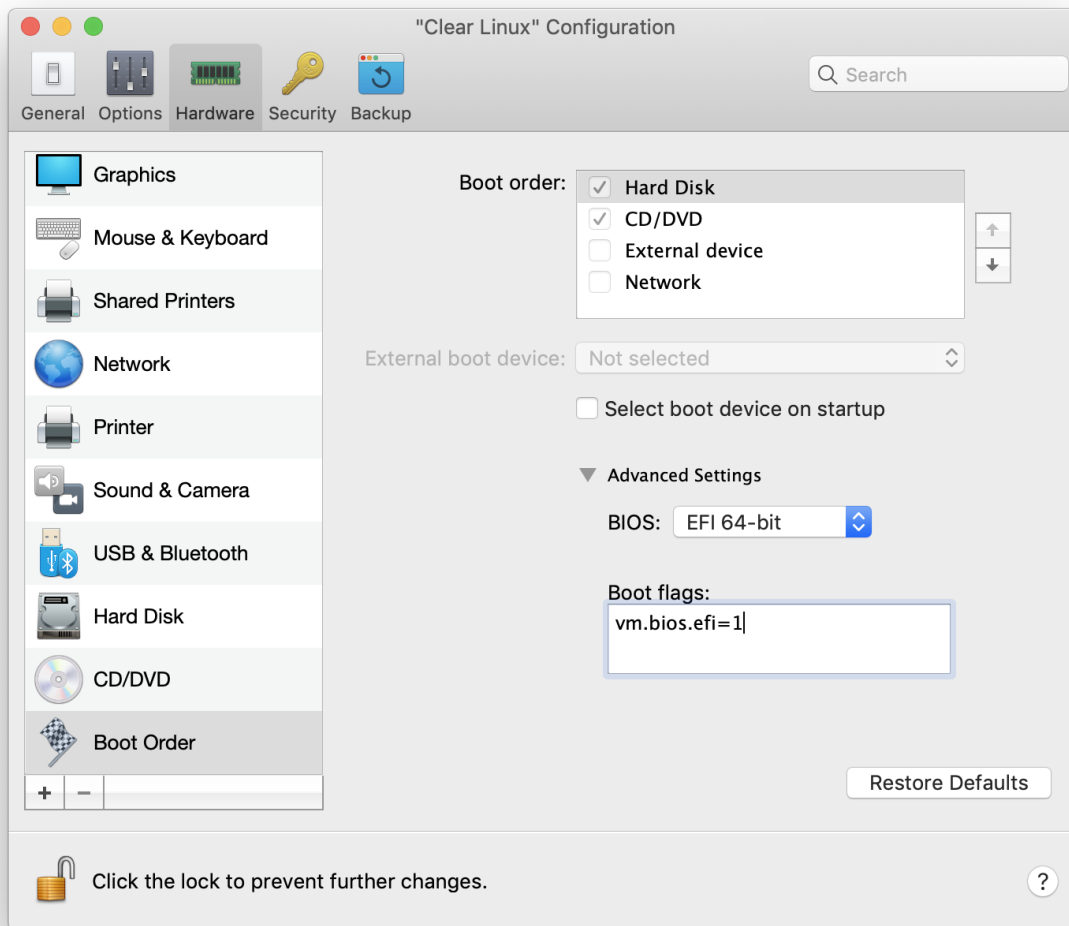
Then add these two `qemu-system-x86_64` options:

```
-bios /usr/share/edk2/ovmf/OVMF_CODE.fd \  
-smbios type=0,vendor=0vendor,version=0version,date=0date,release=0.0,uefi=on \  

```

Parallels

- Select Hardware > Boot Order.
- Expand **Advanced Settings**. Set **BIOS** to “EFI 64-bit” and in the Boot flags field, enter `vm.bios.efi=1`.



Proxmox VE

- Memory: 4GB (not ballooned)
- Processors: 2 (1 socket 2 cores)
- BIOS: OVMF (UEFI)
- Display: Default (VGA)
- Machine: q35
- SATA Controller: VirtIO SATA for attaching virtual disk to (to install the system on)
- CD Drive: GhostBSD ISO
- Hard Disk: At least 8GB Raw
- Network Device: VirtIO

To set resolution, press F2 at boot to access OVMF settings. Select 'Device Manager > OVMF Platform Configuration > Change Preferred', save and reboot.

2.1.3 Downloading

The **GhostBSD** ISO image is available for download [here](#).

Experimental development images are available for download [here](#).

2.2 Installation Guide

2.2.1 Introduction

GhostBSD comes with a Graphical installer call GBI.

After reading this guide, you will know:

- How to make a USB memory stick.
- How to set up a Virtual Machine.
- How to install GhostBSD.
 - How to install GhostBSD using the entire disk drive.
 - How to install alongside other operating system.
- Troubleshooting the installer and live media.

2.3 Getting to know the system

2.4 Troubleshooting

2.5 Frequently Asked Questions

This document aims to cover the most frequently asked questions concerning the GhostBSD operating system. Although initially intended to reduce bandwidth and avoid the same old questions being asked repeatedly, FAQs have become recognized as valuable information resources.

Every effort has been made to make this FAQ as informative as possible; if you have any suggestions about improving it, please feel for this GitHub and create a [Pull Request on GitHub](#).

2.5.1 Introduction

What is GhostBSD?

Briefly, GhostBSD is a UNIX®-like operating system. It works on AMD64 and Intel® platforms. GhostBSD is based on FreeBSD, which is based on U.C. Berkeley's "4.4BSD-Lite" release, with some "4.4BSD-Lite2" enhancements. It is also based indirectly on William Jolitz's port of U.C. Berkeley's "Net/2" to the i386, known as "386BSD", though very little of the 386BSD code remains. At this time, GhostBSD provides MATE desktop as default version, and a community release with XFCE desktop.

Why is it called GhostBSD?

GhostBSD was developed as an operating system to hack on Gnome using FreeBSD technology. After a while, it became what was referred to as **(G)nome (host)ed by Free(BSD)**, which means that Gnome is hosted on the FreeBSD system. Today GhostBSD name is still relevant to the past since MATE is a continuation of Gnome 2.

On which FreeBSD branch is GhostBSD based on?

GhostBSD is based on the FreeBSD 13.0-STABLE branch.

Who can use GhostBSD?

GhostBSD can be used by companies, researchers, data scientists, computer professionals, students, and home users all over the world in their work, education, and recreation.

Does the GhostBSD license have any restrictions?

Yes. Those restrictions do not control how you use the code, merely how you treat the GhostBSD Project itself. If you have serious license concerns, read the actual license. For the simply curious, the license can be summarized like this:

- Do not claim that you wrote this.
- Do not sue us if it breaks.

Can GhostBSD replace my current operating system?

For most users, yes.

Most people do not use an operating system. Instead, they use applications, as these are what make a computer useful. GhostBSD is designed to provide a desktop and full-featured environment for applications. It supports various web browsers, office suites, email readers, graphics programs, programming environments, network servers, and just about everything else you might want. Most of these applications can be built using the FreeBSD Ports Collection.

Suppose you need to use an application that is only available on one operating system. In that case, you cannot easily replace that operating system (this situation is referred to as vendor lock-in). However, the chances are that there is a very similar application on GhostBSD.

If you migrate to GhostBSD from some other UNIX®-like environment, you already know most of what you need to know. However, if your background is in graphic-driven operating systems such as Microsoft Windows® and older versions of MacOS®, expect to invest additional time learning the UNIX way of doing things.

2.5.2 Support

What's the best way to get support?

At this point, the best ways to contact other GhostBSD users or developers are:

- On [Telegram](#) you will meet developers and users
- On [Element](#) which is connected with Telegram
- On [The GhostBSD Forums](#)

2.5.3 Software Installation

How do I install new software?

In GhostBSD software can be installed in three different ways:

- Software Station is a GTK+ based tool to manage (search, install and uninstall) binary packages on GhostBSD.
- FreeBSD's package manager PKG can be used to manage packages using the command line.

Can I use Linux software on my GhostBSD system?

First, you should know that most Linux software is open-source software. That is, it is freely available for many operating systems, not just Linux. Therefore, most software that runs on Linux should also run on GhostBSD.

Like FreeBSD, GhostBSD uses something called Linux Binary Compatibility. In a nutshell, that means you can run many Linux applications as-is. This [section](#) of the FreeBSD Handbook explains this compatibility in more detail and describes when it does not work.

However, you do not have to set up Linux Binary Compatibility on your GhostBSD system as it is already configured for you and should “just work”.

2.5.4 System and software updates

How should I update GhostBSD to latest updates?

To upgrade GhostBSD, please use the Update Station! Do not use pkg to upgrade if you are not aware of how the updates work with GhostBSD, or your system will most likely get broken.

2.6 Providing feedback

2.7 Get Involved

Welcome to the GhostBSD Community! By getting involved, you will meet new friends, learn new skills and make a difference for many users while working with people from all around the globe. This page will give you a brief introduction to things everyone contributing to GhostBSD should know and help you get started with contributing.

We want to ensure that GhostBSD contributors remain a welcoming and friendly place where people can feel comfortable. Therefore, we ask you to abide by the GhostBSD Code of Conduct when interacting with the rest of the GhostBSD Community.

2.7.1 Getting Started!

Beyond is the starting point to find the area you want to contribute. Start at the top of this list and work your way down. We want to make it easy for people to get involved in various areas of GhostBSD. If you need help to get started, please ask any questions you may have on [Telegram](#) or [Element](#)!

Issue Reporting

If you've found a problem or have an idea for an improvement, report it to the developers! QA is critically important to ensure quality, and you can be involved to make sure that our users are happy with the final products.

Bug Triaging

If you'd like to take the next step in contributing to GhostBSD, help triage all those GitHub tickets that people are filing! By separating the wheat from the chaff, you will help developers figure out which issues they need to work on and help them get the information they need to fix them. You don't need any programming experience to triage GitHub tickets, and it's a perfect introduction if you want to become a developer eventually but don't feel like your programming experience is good enough yet.

Development

If you'd like to contribute code for the GhostBSD project, there are many small tasks available, so you don't even need to be a programmer to start developing GhostBSD! In the process, you'll learn valuable new skills like C, Python, GTK, Shell Scripting, Creating and maintaining FreeBSD ports, and so on.

Documentation

Even with the best-designed software, users will need documentation to help them do what they want and get the most out of it. Good-quality documentation is essential and has one of the most significant impacts on end-users.

User Support

Sometimes users will run into trouble, have questions, and not find help or answers on their own. They will need someone to provide guidance and help. The GhostBSD Community provides several platforms for users to ask for support from other users, so other users must be ready to give that support.