

Steam Deck

This guide is a slight modification to Hysteria's linux guide to enable online support for the Steam Deck. Note, when Valve releases an update to the Deck it may wipe any of the following changes. Additionally, these steps bypass the security Valve has put into place to protect your device, these steps should be taken at your own risk. This guide will not work for wireless connections, you will need to plug an ethernet cable with a USB-C adapter into the Deck

Step 1:

Click the "Steam" button and select "Switch to Desktop Mode"

Open "Konsole" and run the following commands.

Step 2: (DO THIS AT YOUR OWN RISK)

```
sudo steamos-readonly disable
```

Step 3:

```
sudo pacman-key --init
```

```
sudo pacman-key --populate archlinux
```

STEP 4:

```
sudo pacman -S openvpn dolphin bridge-utils
```

STEP 5:

```
sudo openvpn --mktun --dev Dolphin0 --dev-type tap --user deck --group deck
```

```
sudo brctl addbr br0
```

```
sudo brctl addif br0 Dolphin0
```

```
sudo brctl addif br0 enp4s0f3u1 (this may be variable, run the command "ip link" to determine the name of your ethernet connection and replace enp4s0f3u1 with the name on your device, it should look similar)
```

```
sudo ip l set Dolphin0 up
```

```
sudo ip l set br0 up
```

```
sudo ip l set Dolphin0 promisc on
```

STEP 6:

```
sudo dhclient -v br0
```

Running Dolphin on Linux

STEP 1 installs Dolphin itself, OpenVPN (which allows the creation of your TAP interface), and bridge_utils, which allows the creation of the network bridge. Bridge_utils is generally considered deprecated but I like it anyway. If you want something considered more 'in-date' you can try the iproute2 package.

STEP 2 creates the TAP interface and the bridge, and links it all to your ethernet network.

STEP 3 compensates for making br0 your NIC - it now needs an IP address. This step will give it one.

STEP 1:

```
sudo pacman -S openvpn dolphin_emu bridge-utils
```

STEP 2:

```
sudo openvpn --mktun --dev Dolphin0 --dev-type tap --user abcde --group users #replace the value given to 'user' with your login name
```

```
sudo brctl addbr br0
```

```
sudo brctl addif br0 Dolphin0
```

```
sudo brctl addif br0 eno1 #if eno1 is not the name of your ethernet device, replace it with what is
```

```
sudo ip l set Dolphin0 up
```

```
sudo ip l set br0 up
```

```
sudo ip l set Dolphin0 promisc on
```

STEP 3:

```
sudo dhclient -v br0
```

Compiling on Apt Based Linux

1: If not already installed, install gcc and gpp from software center.

From Terminal

2: `sudo apt install --no-install-recommends ca-certificates qtbase5-dev qtbase5-private-dev git cmake make gcc g++ pkg-config libavcodec-dev libavformat-dev libavutil-dev libswscale-dev libxi-dev libxrandr-dev libudev-dev libevdev-dev libsfml-dev libminiupnpc-dev libmbdts-dev libcurl4-openssl-dev libhidapi-dev libsystemd-dev libbluetooth-dev libasound2-dev libpulse-dev libpugixml-dev libbz2-dev libzstd-dev liblzo2-dev libpng-dev libusb-1.0-0-dev gettext`

3: `sudo apt install git`

```
sudo apt install cmake
```

```
sudo apt install build-essential
```

4: `mkdir source`

```
cd source
```

```
git clone https://github.com/dolphin-emu/dolphin.git
```

5: Keep your terminal open, we'll be back here later. When the source code is downloaded, go into your newly created source/dolphin folder through your file explorer program (KDE's is called Dolphin, funnily enough), and enter the `.git` directory (note that this is a hidden directory, but if you just add `.git` to the end of the filepath in the file explorer you'll get there). Open the config file and add the below line to the bottom of the `[remote=origin]` tag (don't delete any lines). Save the file when done. There is doubtlessly a way to do this through git itself but all this remotey origin and HEAD stuff confuses me, so I didn't even try :smile: Note the spaces before the word 'fetch' should be a tab.

```
fetch = +refs/pull/*/head:refs/remotes/origin/pr/*
```

6: `cd dolphin`

```
git fetch origin
```

```
git checkout pr/10564
```

```
git submodule update --init
```

```
mkdir build
```

```
cd build
```

```
cmake .. -DLINUX_LOCAL_DEV=true
```

```
make
```

```
sudo make install
```