Android USB tethering driver for Mac OS X

Last update: **Mar 21, 2022**

HoRNDIS(the USB tethering driver for Mac OS X)

HoRNDIS (pronounce: *"horrendous"*) is a driver for Mac OS X that allows you to use your Android phone's native USB tethering mode to get Internet access.

For more information, visit the home page for HoRNDIS on my site.

Installation

From Source/Binary

Get the installation package (Download or Build the installation package from source yourself)

Run the installation package

From Homebrew

```
brew cask install horndis
sudo kextload /Library/Extensions/HoRNDIS.kext
```

Configuration

Assuming that the installation proceeds without errors, after it completes, connect your phone to your Mac by USB.

Enter the settings menu on your phone.

In the connections section, below Wi-Fi and Bluetooth:

Select "More..."

Select "Tethering & portable hotspot"

Check the "USB tethering" box. It should flash once, and then become solidly checked.

Uninstallation

Delete the `HoRNDIS.kext` under `/System/Library/Extensions` and `/Library/Extensions` folder

Restart your computer

Building the source

`git clone` the repository

Simply running xcodebuild in the checkout directory should be sufficient to build the kext.

If you wish to package it up, you can run `make` to assemble the package in the build/ directory

Debugging and Development Notes

This sections contains tips and tricks for developing and debugging the driver.

USB Device Information

Mac OS System Menu -> About This Mac -> System Report --> Hardware/USB
Lists all USB devices that OS recognizes. Unfortunately, it does not give USB descriptors.

`lsusb -v`
It prints USB configuration, such as interface and endpoint descriptors. You can print it for all devices or limit the output to specific ones. In order to run this command, you need to install *usbutils*.

Homebrew users: `brew install mikhailai/misc/usbutils`
Please *do not* install *lsusb* package from Homebrew Core, it's a different utility with the same name.

Macports users: `sudo port install usbutils`

OS Logging

The `LOG(....)` statements, sprinkled throughout the HoRNDIS code, call the `IOLog` functions. On Mac OS *El Capitan* (10.11) and earlier, the log messages go into `/var/log/system.log`. Starting from *Sierra* (10.12), these messages are no longer written to `system.log` and instead can be viewed via:

GUI, using *Console* application, located in *Utilities* folder. You need to enter `process:kernel` in the search box in order to filter the relevant messages.

Command Line, using the `log` command. For example:

```
log show --predicate process==\"kernel\" --start "2018-12-11 22:54:00"
```

The start value needs to be several minutes in the past, so it would not flood the console. Here is a convenient command that prints the messages from the past 3 mintes:

```
log show --predicate process==\"kernel\" --start "$(date -v-3M +'%F %T')"
```

I've observed that Mac OS logging is unreliable (especially in *Sierra*). In some cases, the messages may come out garbled (looking like bad multi-threaded code). In other cases, either GUI or Command Line may be missing messages that were emitted. Sometimes, reloading the driver may fix the problem.

GitHub

https://github.com/jwise/HoRNDIS

Comments

1. 10.11

Not working on OSX El Capitain:
Jun 18 21:35:33 MBPRetiDeuschle kernel[0]: HoRNDIS: init: HoRNDIS tethering driver for Snow Leopard+, by Joshua Wise Jun 18 21:35:33 MBPRetiDeuschle kernel[0]: 006028.794737 IOUSBHostHIDDevice@: IOUSBHostHIDDevice::start: unable to set device idle milliseconds Jun 18 21:35:33 MBPRetiDeuschle kernel[0]: 006028.794750 IOUSBHostHIDDevice@: IOUSBHostHIDDevice::start: start has failed Jun 18 21:35:33 MBPRetiDeuschle kernel[0]: 006028.797808 IOUSBHostHIDDevice@: IOUSBHostHIDDevice::start: unable to set device idle milliseconds Jun 18 21:35:33 MBPRetiDeuschle kernel[0]: 006028.797820 IOUSBHostHIDDevice@: IOUSBHostHIDDevice::start: start has failed Jun 18 21:35:33 MBPRetiDeuschle kernel[0]: IOHIDDevice::newUserClient failed to get a workloop

Reviewed by DiegoGiovany at 2015-06-19 00:47

---

2. 32-bit build needed for Macs running a 32-bit kernel by default

was: Mac OS X 10.6.8 Install issues - "no code for running kernel's architecture"
Hi, I downloaded the HoRNDIS-rel1.pkg installer and ran it, and it appeared to work, but my Mac didn't detect my phone when I plugged it in, even after rebooting. I fired up console.app and searched for "horndis" and found this:

```
11/16/12 2:42:42 PM Installer[69496]    HoRNDIS  Installation Log
11/16/12 2:42:42 PM Installer[69496]    Opened from: /Volumes/Old Macintosh HD/downloads/HoRNDIS-rel1.pkg
11/16/12 2:42:56 PM Installer[69496]        Install: "HoRNDIS"
11/16/12 2:42:56 PM Installer[69496]        Install: "HoRNDIS Kernel Extension"
11/16/12 2:43:08 PM com.apple.kextd[17] Can't load /System/Library/Extensions/HoRNDIS.kext - no code for running kernel's architecture.
11/16/12 2:43:08 PM com.apple.kextd[17] Failed to load /System/Library/Extensions/HoRNDIS.kext - (libkern/kext) requested architecture/executable not found.
11/16/12 2:43:08 PM installd[69532] Installed "HoRNDIS" ()
11/16/12 2:47:53 PM com.apple.kextd[17] Can't load /System/Library/Extensions/HoRNDIS.kext - no code for running kernel's architecture.
11/16/12 2:47:53 PM com.apple.kextd[17] Failed to load /System/Library/Extensions/HoRNDIS.kext - (libkern/kext) requested architecture/executable not found.
11/16/12 2:47:53 PM com.apple.kextd[17] Load com.joshuawise.kexts.HoRNDIS failed; removing personalities.
```

I plugged in the phone at the last step of the install when it had instructions up, so I re-ran the installer with the phone disconnected the entire time and got this:

```
11/16/12 2:58:01 PM Installer[1751] HoRNDIS  Installation Log
11/16/12 2:58:01 PM Installer[1751] Opened from: /Volumes/Old Macintosh HD/downloads/HoRNDIS-rel1.pkg
11/16/12 2:58:11 PM com.apple.kextcache[689]    /System/Library/Extensions/HoRNDIS.kext doesn't support architecture i386; omitting from prelinked kernel.
11/16/12 2:58:13 PM Installer[1751]     Upgrade: "HoRNDIS"
11/16/12 2:58:13 PM Installer[1751]     Upgrade: "HoRNDIS Kernel Extension"
11/16/12 2:58:24 PM com.apple.kextd[17] Can't load /System/Library/Extensions/HoRNDIS.kext - no code for running kernel's architecture.
11/16/12 2:58:24 PM com.apple.kextd[17] Failed to load /System/Library/Extensions/HoRNDIS.kext - (libkern/kext) requested architecture/executable not found.
11/16/12 2:58:24 PM installd[1782]  Installed "HoRNDIS" ()
```

I don't think there's anything really special about my mac - it's a 2010 model with an i7 - it should run i386 code just fine. So I'm not sure what to make of that error.
The phone is a rooted Samsung Galaxy S III running Clean Rom (basically stock but without all the trials and Verizon crap). It has a USB Tethering checkbox in the menu and it puts a "USB Tethering, Tap to configure" line in the Ongoing section of my top pull-down menu.
Any ideas?

Reviewed by nfriedly at 2012-11-16 20:05

---

3. creates a new Ethernet connection on each phone restart

I have already 10 Ethernets right now after using it for 10 days.In my network section.So this might not be related to your package and it could be related to something to change on the phone to get this fixed.But anyway if anyone's going to tell what's the problem and how to fix it,I think it could be maybe here.
How do I make it not create any new Ethernets?
I do a lot of setup on the ethernet setting up dns proxies and ... and new Ethernet comes on with no configuration.So I have to do all over again.

Reviewed by Stevemoretz at 2020-03-22 06:26

```
High Speed device @ 7 (0x1D110000): ..........................................   Composite device: "Android Phone"
    Port Information:   0x1018
           Not Captive
           External Device
           Connected
           Enabled
           Connected to External Port
    Number Of Endpoints (includes EP0):
        Total Endpoints for Configuration 1 (current):   6
    Device Descriptor
        Descriptor Version Number:   0x0200
        Device Class:   0   (Composite)
        Device Subclass:   0
        Device Protocol:   0
        Device MaxPacketSize:   64
        Device VendorID/ProductID:   0x0BB4/0x0FFC   (HTC Corporation)
        Device Version Number:   0x0228
        Number of Configurations:   1
        Manufacturer String:   2 "HTC"
        Product String:   3 "Android Phone"
        Serial Number String:   4 "HT13PTJ08628"
    Configuration Descriptor (current config)
        Length (and contents):   98
            Raw Descriptor (hex)    0000: 09 02 62 00 03 01 00 80  FA 08 0B 00 02 02 06 FF
            Raw Descriptor (hex)    0010: 07 09 04 00 00 01 02 02  FF 05 05 24 00 10 01 05
            Raw Descriptor (hex)    0020: 24 01 00 01 04 24 02 00  05 24 06 00 01 07 05 82
            Raw Descriptor (hex)    0030: 03 08 00 09 09 04 01 00  02 0A 00 00 06 07 05 81
            Raw Descriptor (hex)    0040: 02 00 02 00 07 05 01 02  00 02 00 09 04 02 00 02
            Raw Descriptor (hex)    0050: FF 42 01 00 07 05 83 02  00 02 00 07 05 02 02 00
            Raw Descriptor (hex)    0060: 02 00
        Number of Interfaces:   3
        Configuration Value:   1
        Attributes:   0x80 (bus-powered)
        MaxPower:   500 mA
        Interface Association   Communications-Control
            First Interface   0
            Interface Count   2
            Function Class   2   (Communications-Control)
            Function Subclass   6
            Interface Protocol   255
            Function String   7 "RNDIS"
        Interface #0 - Communications-Control ..........................................   "RNDIS Communications Control"
            Alternate Setting   0
            Number of Endpoints   1
            Interface Class:   2   (Communications-Control)
            Interface Subclass;   2
            Interface Protocol:   255
            Comm Class Header Functional Descriptor
                Raw Descriptor (hex)   0000: 05 24 00 10 01
            Comm Class Call Management Functional Descriptor
                Raw Descriptor (hex)   0000: 05 24 01 00 01
            Comm Class Abstract Control Management Functional Descriptor
                Raw Descriptor (hex)   0000: 04 24 02 00
            Comm Class Union Functional Descriptor
                Raw Descriptor (hex)   0000: 05 24 06 00 01
            Endpoint 0x82 - Interrupt Input
                Address:   0x82  (IN)
                Attributes:   0x03  (Interrupt)
                Max Packet Size:   8  (8 x 1  transactions opportunities per microframe)
                Polling Interval:   9 (256 microframes (32 msecs) )
        Interface #1 - Communications-Data/Unknown Comm Class Model ....................................   "RNDIS Ethernet Data"
            Alternate Setting   0
            Number of Endpoints   2
            Interface Class:   10   (Communications-Data)
            Interface Subclass;   0   (Unknown Comm Class Model)
            Interface Protocol:   0
            Endpoint 0x81 - Bulk Input
                Address:   0x81  (IN)
                Attributes:   0x02  (Bulk)
                Max Packet Size:   512
                Polling Interval:   0 ( Endpoint never NAKs)
            Endpoint 0x01 - Bulk Output
                Address:   0x01  (OUT)
                Attributes:   0x02  (Bulk)
                Max Packet Size:   512
                Polling Interval:   0 ( Endpoint never NAKs)
        Interface #2 - Vendor-specific
            Alternate Setting   0
            Number of Endpoints   2
            Interface Class:   255   (Vendor-specific)
            Interface Subclass;   66   (Vendor-specific)
            Interface Protocol:   1
            Endpoint 0x83 - Bulk Input
                Address:   0x83  (IN)
                Attributes:   0x02  (Bulk)
                Max Packet Size:   512
                Polling Interval:   0 ( Endpoint never NAKs)
            Endpoint 0x02 - Bulk Output
                Address:   0x02  (OUT)
                Attributes:   0x02  (Bulk)
                Max Packet Size:   512
```

```
        Device MaxPacketSize:   64
        Number of Configurations:   1
        bReserved:   0
    Other Speed Configuration Descriptor
        Length (and contents):   98
            Raw Descriptor (hex)    0000: 09 07 62 00 03 01 00 80  FA 08 0B 00 02 02 06 FF
            Raw Descriptor (hex)    0010: 07 09 04 00 00 01 02 02  FF 05 05 24 00 10 01 05
            Raw Descriptor (hex)    0020: 24 01 00 01 04 24 02 00  05 24 06 00 01 07 05 82
            Raw Descriptor (hex)    0030: 03 08 00 20 09 04 01 00  02 0A 00 00 06 07 05 81
            Raw Descriptor (hex)    0040: 02 40 00 00 07 05 01 02  40 00 00 09 04 02 00 02
            Raw Descriptor (hex)    0050: FF 42 01 00 07 05 83 02  40 00 00 07 05 02 02 40
            Raw Descriptor (hex)    0060: 00 00
        Number of Interfaces:   3
        Configuration Value:   1
        Attributes:   0x80 (bus-powered)
        MaxPower:   500 mA
        Interface Association   Communications-Control
            First Interface   0
            Interface Count   2
            Function Class   2   (Communications-Control)
            Function Subclass   6
            Interface Protocol   255
            Function String   7 "RNDIS"
        Interface #0 - Communications-Control ..........................................   "RNDIS Communications Control"
            Alternate Setting   0
            Number of Endpoints   1
            Interface Class:   2   (Communications-Control)
            Interface Subclass;   2
            Interface Protocol:   255
            Comm Class Header Functional Descriptor
                Raw Descriptor (hex)   0000: 05 24 00 10 01
            Comm Class Call Management Functional Descriptor
                Raw Descriptor (hex)   0000: 05 24 01 00 01
            Comm Class Abstract Control Management Functional Descriptor
                Raw Descriptor (hex)   0000: 04 24 02 00
            Comm Class Union Functional Descriptor
                Raw Descriptor (hex)   0000: 05 24 06 00 01
            Endpoint 0x82 - Interrupt Input
                Address:   0x82  (IN)
                Attributes:   0x03  (Interrupt)
                Max Packet Size:   8
                Polling Interval:   32 ms
        Interface #1 - Communications-Data/Unknown Comm Class Model ....................................   "RNDIS Ethernet Data"
            Alternate Setting   0
            Number of Endpoints   2
            Interface Class:   10   (Communications-Data)
            Interface Subclass;   0   (Unknown Comm Class Model)
            Interface Protocol:   0
            Endpoint 0x81 - Bulk Input
                Address:   0x81  (IN)
                Attributes:   0x02  (Bulk)
                Max Packet Size:   64
                Polling Interval:   0 ms
            Endpoint 0x01 - Bulk Output
                Address:   0x01  (OUT)
                Attributes:   0x02  (Bulk)
                Max Packet Size:   64
                Polling Interval:   0 ms
        Interface #2 - Vendor-specific
            Alternate Setting   0
            Number of Endpoints   2
            Interface Class:   255   (Vendor-specific)
            Interface Subclass;   66   (Vendor-specific)
            Interface Protocol:   1
            Endpoint 0x83 - Bulk Input
                Address:   0x83  (IN)
                Attributes:   0x02  (Bulk)
                Max Packet Size:   64
                Polling Interval:   0 ms
            Endpoint 0x02 - Bulk Output
                Address:   0x02  (OUT)
                Attributes:   0x02  (Bulk)
                Max Packet Size:   64
                Polling Interval:   0 ms
```

Reviewed by grishka at 2013-01-21 21:38

---

5. Should match with IOUSBInterface, not IOUSBDevice

Devices like Droid RAZR, which have a device class of 0/0/0 (rather than 224/0/0), will not work, since currently we match on a device class of 224/0/0. We should match on an interface class to make sure we get what we want; if we do that, then we can match on the more specific interface class, as well, to avoid picking up CDC devices.
Original:
Subj: Tethering error with Mac 10.7.5 and Android 4.0.4
MacBook Pro 13" with OS X 10.7.5

Steps:

Installed HoRNDIS and rebooted (tried install/reboot again just to be sure)

Plugged in my phone via USB

Enabled USB tethering and it says "tethering error" on the phone

---

There are also no changes in the Network preferences on the Mac

---

Via *kextstat* I saw it wasn't loaded so I manually loaded it:

```
sudo kextload /System/Library/Extensions/HoRNDIS.kext
```

Tried to tether again and noticed these errors in *kernel.log*:

```
Nov 14 19:05:28 Robs-MacBook-Pro kernel[0]: 0 1 AppleUSBCDCACMData: start - Find CDC driver for ACM data interface failed Nov 14 19:05:28 Robs-MacBook-Pro kernel[0]: 0 1 AppleUSBCDCECMData: start - Find CDC driver for ECM data interface failed
```

Reviewed by robolmos at 2012-11-15 04:16

---

6. Bonjour/Zeroconf not working

I am unable to get Bonjour/Zeroconf to work over the HoRNDIS interface. Bonjour works on Windows RNDIS , but services being broadcast are not detected when using HoRNDIS. Does the HorDIS driver support multicast DNS?

Reviewed by danyocom at 2014-07-27 13:55

---

7. Can't get to working on 10.13.5 and Nokia 7 plus

I've used the tethering successfully on previous intallations of os x with different android phones, but now that I'm trying it with my latest setup, I can't seem to get the tethering detected in any way. Nothing appears in the network connections and obviously there is no internet connection.
My setup:

Macbook Pro (Retina, 13-inch, Early 2015)

MacOS High Sierra version 10.13.5

Nokia 7 plus, Android 8.1

Using the latest driver release, i.e. 9.1. Tied installing via package and homebrew and rebooting the laptop a lot, but to no avail.
Any ideas what to try or how to debug?
Disabling system integrity protection is a no go.

Reviewed by Riimu at 2018-07-25 08:30

---

8. Samsung S7 Edge (Android 8.0.0)

After quite some time of investigating why my Samsung S7 Edge running Android 8.0.0 didn't work in MacOS High Sierra 10.13.2, even after a complete reset of the device, installation of ADB, csrutil disabled, tested successfully with Sony Xperia Z1 running 5.1.1 and tested successfully with Samsung S8 running Android 8.0.0, I tried switching HoRNDIS from release 9 to release 8 and that worked. Do you some kind of vendor/model filtering in your code that makes release 9 unsable with Samsung S7 Edge ?

Reviewed by flemming-fridthjof at 2018-07-11 13:48

---

9. Doesn't work with my Samsung Galaxy S II

I have S2 with Android 2.3.6 Gingerbread. USB tethering works fine with Windows.
Machine : MacBook Air 2011 13" OS X version : 10.7.5 HoRNDIS pkg : HoRNDIS-rel2.pkg

Love, Bonito Hong Kong

The driver doesn't load startup, so I load it with kextload. When I plug it in, it appears as a Modem in Network Preferences (as before).

Love, Bonito Hong Kong

I attached the log files from kernel.log and USB Prober below. Thanks a lot.

kernel.log

After connecting the phone to USB port kernel[0]: USBMSC Identifier (non-unique): 304D19BA84548C7E 0x4e8 0x685e 0x400 kernel[0]: AppleUSBCDCACMData: Version number - 4.1.22, Input buffers 8, Output buffers 16 kernel[0]: AppleUSBCDC: Version number - 4.1.22 **After enabling USB tethering** kernel[0]: 0xffffff801ce5da00/(5) Device not responding kernel[0]: 0 ff AppleUSBCDCACMControl: configureACM - ACM Control interface has vendor specific protocol kernel[0]: 0 0 AppleUSBCDCACMControl: start - configureACM failed kernel[0]: AppleUSBCDCACMData: Version number - 4.1.22, Input buffers 8, Output buffers 16 kernel[0]: AppleUSBCDC: Version number - 4.1.22

USB Bus Probe

High Speed device @ 4 (0xFD120000): ..... Communication device: "SAMSUNG_Android" Port Information: 0x1018 Not Captive External Device Connected Enabled Number Of Endpoints (includes EP0):
Total Endpoints for Configuration 1 (current): 4 Device Descriptor
Descriptor Version Number: 0x0200 Device Class: 2 (Communication) Device Subclass: 0 Device Protocol: 0 Device MaxPacketSize: 64 Device VendorID/ProductID: 0x04E8/0x6863 (Samsung Electronics Co., Ltd.) Device Version Number: 0x0400 Number of Configurations: 1 Manufacturer String: 1 "SAMSUNG" Product String: 2 "SAMSUNG_Android" Serial Number String: 3 "304D19BA84548C7E" Configuration Descriptor (current config)
Length (and contents): 75 Raw Descriptor (hex) 0000: 09 02 4B 00 02 01 00 C0 30 08 0B 00 02 02 06 00
Raw Descriptor (hex) 0010: 0A 09 04 00 00 01 02 02 FF 08 05 24 00 10 01 05
Raw Descriptor (hex) 0020: 24 01 00 01 04 24 02 00 05 24 06 00 01 07 05 86
Raw Descriptor (hex) 0030: 03 08 00 09 09 04 01 00 02 0A 00 00 09 07 05 8B
Raw Descriptor (hex) 0040: 02 00 02 00 07 05 0A 02 00 02 00 Number of Interfaces: 2 Configuration Value: 1 Attributes: 0xC0 (self-powered) MaxPower: 96 ma Interface Association Communications-Control First Interface 0 Interface Count 2 Function Class 2 (Communications-Control) Function Subclass 6 Interface Protocol 0 Function String 10 "RNDIS" Interface #0 - Communications-Control ..... "RNDIS Communications Control" Alternate Setting 0 Number of Endpoints 1 Interface Class: 2 (Communications-Control) Interface Subclass; 2 Interface Protocol: 255 Comm Class Header Functional Descriptor
Raw Descriptor (hex) 0000: 05 24 00 10 01 Comm Class Call Management Functional Descriptor
Raw Descriptor (hex) 0000: 05 24 01 00 01 Comm Class Abstract Control Management Functional Descriptor
Raw Descriptor (hex) 0000: 04 24 02 00 Comm Class Union Functional Descriptor
Raw Descriptor (hex) 0000: 05 24 06 00 01 Endpoint 0x86 - Interrupt Input
Address: 0x86 (IN) Attributes: 0x03 (Interrupt no synchronization data endpoint) Max Packet Size: 8 (8 x 1 transactions opportunities per microframe) Polling Interval: 9 (256 microframes (32 msecs)) Interface #1 - Communications-Data/Unknown Comm Class Model ..... "RNDIS Ethernet Data" Alternate Setting 0 Number of Endpoints 2 Interface Class: 10 (Communications-Data) Interface Subclass; 0 (Unknown Comm Class Model) Interface Protocol: 0 Endpoint 0x8B - Bulk Input
Address: 0x8B (IN) Attributes: 0x02 (Bulk no synchronization data endpoint) Max Packet Size: 512 Polling Interval: 0 (Endpoint never NAKs) Endpoint 0x0A - Bulk Output
Address: 0x0A (OUT) Attributes: 0x02 (Bulk no synchronization data endpoint) Max Packet Size: 512 Polling Interval: 0 (Endpoint never NAKs) Device Qualifier Descriptor
Descriptor Version Number: 0x0200 Device Class 2 (Communication) Device Subclass 0 Device Protocol 0 Device MaxPacketSize: 64 Number of Configurations: 1 bReserved: 0
Other Speed Configuration Descriptor
Length (and contents): 75 Raw Descriptor (hex) 0000: 09 07 4B 00 02 01 00 C0 30 08 0B 00 02 02 06 00
Raw Descriptor (hex) 0010: 0A 09 04 04 00 01 02 02 FF 08 05 24 00 10 01 05
Raw Descriptor (hex) 0020: 24 01 00 01 04 24 02 00 05 24 06 04 05 07 05 86
Raw Descriptor (hex) 0030: 03 08 00 20 09 04 05 00 02 0A 00 00 09 07 05 8B
Raw Descriptor (hex) 0040: 02 40 00 00 07 05 0A 02 40 00 00 Number of Interfaces: 2 Configuration Value: 1 Attributes: 0xC0 (self-powered) MaxPower: 96 ma Interface Association Communications-Control First Interface 0 Interface Count 2 Function Class 2 (Communications-Control) Function Subclass 6 Interface Protocol 0 Function String 10 "RNDIS" Interface #4 - Communications-Control ..... "RNDIS Communications Control" Alternate Setting 0 Number of Endpoints 1 Interface Class: 2 (Communications-Control) Interface Subclass; 2 Interface Protocol: 255 Comm Class Header Functional Descriptor
Raw Descriptor (hex) 0000: 05 24 00 10 01 Comm Class Call Management Functional Descriptor
Raw Descriptor (hex) 0000: 05 24 01 00 01 Comm Class Abstract Control Management Functional Descriptor
Raw Descriptor (hex) 0000: 04 24 02 00 Comm Class Union Functional Descriptor
Raw Descriptor (hex) 0000: 05 24 06 04 05 Endpoint 0x86 - Interrupt Input
Address: 0x86 (IN) Attributes: 0x03 (Interrupt no synchronization data endpoint) Max Packet Size: 8 Polling Interval: 32 ms Interface #5 - Communications-Data/Unknown Comm Class Model ..... "RNDIS Ethernet Data" Alternate Setting 0 Number of Endpoints 2 Interface Class: 10 (Communications-Data) Interface Subclass; 0 (Unknown Comm Class Model) Interface Protocol: 0 Endpoint 0x8B - Bulk Input

After connecting the phone to USB port

34.747 [2] AppleUSBEHCI[0xffffff800a888000]::PollInterrupts - deferring checking for RHStatus until we are running again 34.771 [3] AppleUSBHub[0xffffff800a8d5000]::DecrementOutstandingIO(2118), outstandingIO(0), _interruptReadPending(false) - rearming read 34.771 [3] AppleUSBHub[0xffffff800a8d5000]::DecrementOutstandingIO(2118), outstandingIO(0), _interruptReadPending(false) - rearming read 34.771 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateInterruptTransfer - root hub interrupt transfer - clearing unneeded memDesc (0xffffff800a8c4e00) from dmaCommand (0xffffff800a8b9980) 34.771 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateInterruptTransfer - root hub interrupt transfer - clearing unneeded memDesc (0xffffff800a8c4e00) from dmaCommand (0xffffff800a8b9980) 34.771 [3] AppleUSBEHCI[0xffffff800a888000]::UIMRootHubStatusChange got bitmap (0x2) 34.771 [3] AppleUSBEHCI[0xffffff800a888000]::UIMRootHubStatusChange got bitmap (0x2) 34.771 [3] AppleUSBHub[0xffffff800a8d5000]::ChangeRaisedPowerState(+) now (1) 34.771 [3] AppleUSBHub[0xffffff800a8d5000]::ChangeRaisedPowerState(+) now (1) 34.783 [3] AppleUSBHubPort[0xffffff800a8afc00]::WakeSuspendCommand calling commandWakeUp 34.783 [3] AppleUSBHubPort[0xffffff800a8afc00]::WakeSuspendCommand calling commandWakeUp 34.783 [3] AppleUSBHub[0xffffff800a8d5000]::ChangeRaisedPowerState(-) now (0) 34.783 [3] AppleUSBHub[0xffffff800a8d5000]::ChangeRaisedPowerState(-) now (0) 34.783 [3] AppleUSBHub[0xffffff800a8d5000]::DecrementOutstandingIO(2124), outstandingIO(0), _interruptReadPending(false) - rearming read 34.783 [3] AppleUSBHub[0xffffff800a8d5000]::DecrementOutstandingIO(2124), outstandingIO(0), _interruptReadPending(false) - rearming read 34.783 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateInterruptTransfer - root hub interrupt transfer - clearing unneeded memDesc (0xffffff800a8c4e00) from dmaCommand (0xffffff800a8b9980) 34.783 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateInterruptTransfer - root hub interrupt transfer - clearing unneeded memDesc (0xffffff800a8c4e00) from dmaCommand (0xffffff800a8b9980) 34.784 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2125), outstandingIO(0), _interruptReadPending(false) - rearming read 34.784 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2125), outstandingIO(0), _interruptReadPending(false) - rearming read 34.799 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (1) 34.799 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (1) 34.837 [3] AppleUSBHub[0xffffff800a9c7000]::HubAreAllPortsDisconnectedOrSuspended - port 2 still initing, status changing, or adding a device (0/1/0) 34.837 [3] AppleUSBHub[0xffffff800a9c7000]::HubAreAllPortsDisconnectedOrSuspended - port 2 still initing, status changing, or adding a device (0/1/0) 34.901 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (2) 34.901 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (2) 34.901 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (1) 34.901 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (1) 34.901 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2130), outstandingIO(0), _interruptReadPending(false) - rearming read 34.901 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2130), outstandingIO(0), _interruptReadPending(false) - rearming read 34.903 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (0) 34.903 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (0) 34.927 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (1) 34.927 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (1) 35.039 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateControlEndpoint(0, 0, 64, 2 @(0, 0)) 35.039 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateControlEndpoint(0, 0, 64, 2 @(0, 0)) 35.042 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateControlEndpoint(4, 0, 64, 2 @(0, 0)) 35.042 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateControlEndpoint(4, 0, 64, 2 @(0, 0)) 35.045 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (0) 35.045 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (0) 35.045 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2135), outstandingIO(0), _interruptReadPending(false) - rearming read 35.045 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2135), outstandingIO(0), _interruptReadPending(false) - rearming read

After enabling USB tethering

37.272 [2] IOUSBPipe[0xffffff801db70800]::Read(sync) returned 0xe00002ed (device is not responding) - stalling pipe 37.272 [2] IOUSBPipe[0xffffff801db70800]::Read(sync) returned 0xe00002ed (device is not responding) - stalling pipe 37.272 [3] IOUSBInterfaceUserClientV2[0xffffff8020680800]::ReadPipe - returning err e00002ed, size read: 0 37.272 [3] IOUSBInterfaceUserClientV2[0xffffff8020680800]::ReadPipe - returning err e00002ed, size read: 0 37.272 [2] IOUSBPipeV2[0xffffff801db70800]::Abort setting status to 0 37.272 [2] IOUSBPipeV2[0xffffff801db70800]::Abort setting status to 0 37.295 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (1) 37.295 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (1) 37.333 [3] AppleUSBHub[0xffffff800a9c7000]::HubAreAllPortsDisconnectedOrSuspended - port 2 still initing, status changing, or adding a device (0/1/0) 37.333 [3] AppleUSBHub[0xffffff800a9c7000]::HubAreAllPortsDisconnectedOrSuspended - port 2 still initing, status changing, or adding a device (0/1/0) 37.390 [3] AppleUSBHub[0xffffff800a9c7000]::HubAreAllPortsDisconnectedOrSuspended - port 2 still initing, status changing, or adding a device (0/1/0) 37.390 [3] AppleUSBHub[0xffffff800a9c7000]::HubAreAllPortsDisconnectedOrSuspended - port 2 still initing, status changing, or adding a device (0/1/0) 37.397 [2] IOUSBDevice(SAMSUNG_Android)[0xffffff800c43e700]::terminate - making sure all endpoints are enabled 37.397 [2] IOUSBDevice(SAMSUNG_Android)[0xffffff800c43e700]::terminate - making sure all endpoints are enabled 37.401 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - Mass Storage[0xffffff80240e1800] is not in _OPEN_CLIENTS 37.401 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - Mass Storage[0xffffff80240e1800] is not in _OPEN_CLIENTS 37.401 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - CDC Abstract Control Model (ACM)[0xffffff80200cb000] is not in _OPEN_CLIENTS 37.401 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - CDC Abstract Control Model (ACM)[0xffffff80200cb000] is not in _OPEN_CLIENTS 37.401 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - CDC ACM Data[0xffffff8010688200] is not in _OPEN_CLIENTS 37.401 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - CDC ACM Data[0xffffff8010688200] is not in _OPEN_CLIENTS 37.401 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - IOUSBInterface[0xffffff800d7a6400] is not in _OPEN_CLIENTS 37.401 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - IOUSBInterface[0xffffff800d7a6400] is not in _OPEN_CLIENTS 37.405 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - Mass Storage[0xffffff80240e1800] is not in _OPEN_CLIENTS 37.405 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - Mass Storage[0xffffff80240e1800] is not in _OPEN_CLIENTS 37.408 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - CDC Abstract Control Model (ACM)[0xffffff80200cb000] is not in _OPEN_CLIENTS 37.408 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - CDC Abstract Control Model (ACM)[0xffffff80200cb000] is not in _OPEN_CLIENTS 37.413 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - CDC ACM Data[0xffffff8010688200] is not in _OPEN_CLIENTS 37.413 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - CDC ACM Data[0xffffff8010688200] is not in _OPEN_CLIENTS 37.418 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - IOUSBInterface[0xffffff800d7a6400] is not in _OPEN_CLIENTS 37.418 [2] SAMSUNG_Android[0xffffff800c43e700]::handleIsOpen - IOUSBInterface[0xffffff800d7a6400] is not in _OPEN_CLIENTS 37.421 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (2) 37.421 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (2) 37.421 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (1) 37.421 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (1) 37.421 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2140), outstandingIO(0), _interruptReadPending(false) - rearming read 37.421 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2140), outstandingIO(0), _interruptReadPending(false) - rearming read 37.422 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (0) 37.422 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (0) 37.455 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (1) 37.455 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(+) now (1) 37.492 [3] AppleUSBHub[0xffffff800a9c7000]::HubAreAllPortsDisconnectedOrSuspended - port 2 still initing, status changing, or adding a device (0/1/0) 37.492 [3] AppleUSBHub[0xffffff800a9c7000]::HubAreAllPortsDisconnectedOrSuspended - port 2 still initing, status changing, or adding a device (0/1/0) 37.568 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateControlEndpoint(0, 0, 64, 2 @(0, 0)) 37.568 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateControlEndpoint(0, 0, 64, 2 @(0, 0)) 37.571 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateControlEndpoint(4, 0, 64, 2 @(0, 0)) 37.571 [3] AppleUSBEHCI[0xffffff800a888000]::UIMCreateControlEndpoint(4, 0, 64, 2 @(0, 0)) 37.573 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (0) 37.573 [3] AppleUSBHub[0xffffff800a9c7000]::ChangeRaisedPowerState(-) now (0) 37.573 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2145), outstandingIO(0), _interruptReadPending(false) - rearming read 37.573 [3] AppleUSBHub[0xffffff800a9c7000]::DecrementOutstandingIO(2145), outstandingIO(0), _interruptReadPending(false) - rearming read

Reviewed by baugp at 2012-12-18 11:33

I have an old MacBook Pro (2008) that I'd love to install this on but I'm getting "the operation couldn't be completed (com.apple.installer.pagecontroller error -1)"
Any workarounds?

Reviewed by Husky2222 at 2018-11-10 18:13

---

11. GPLv2+ or GPLv3?

COPYING.rtf says GPLv3, but the source files say GPLv2+. I might like to distribute this driver, but will probably only be allowed to do so if I can distribute under GPLv2 not be required to use GPLv3. Thanks!

Reviewed by swarren at 2019-03-06 21:13

---

12. brew hint on about page is out of date.

brew hint on about page is out of date. % brew cask install horndis
Error: `brew cask` is no longer a `brew` command. Use `brew <command> --cask` instead.
This trys to work and fails. % brew install --cask horndis ==> Downloading https://github.com/jwise/HoRNDIS/releases/download/rel9.2/HoRNDIS-9.2.pkg ==> Downloading from https://objects.githubusercontent.com/github-production-release-asset-2e65be/6651247/f07fac90-95dc-11e8-9939-daf33
100.0% Error: macOS El Capitan, Sierra, High Sierra or Mojave is required for this software. Catalina 10.15.7

Reviewed by niftyhacking at 2022-03-18 15:55

---

13. Support for Monterey 12.1 M1 Chip

PLZ build the HoRNDIS in such a way that it natively supports M1 chip and Monterey 12 or later.

So that we do not need to disable SIP for installation.....

Reviewed by SoumadeepChoudhury at 2022-01-30 13:55

---

14. I know Joshua no longer develops, but Chris1111 has archived his port already

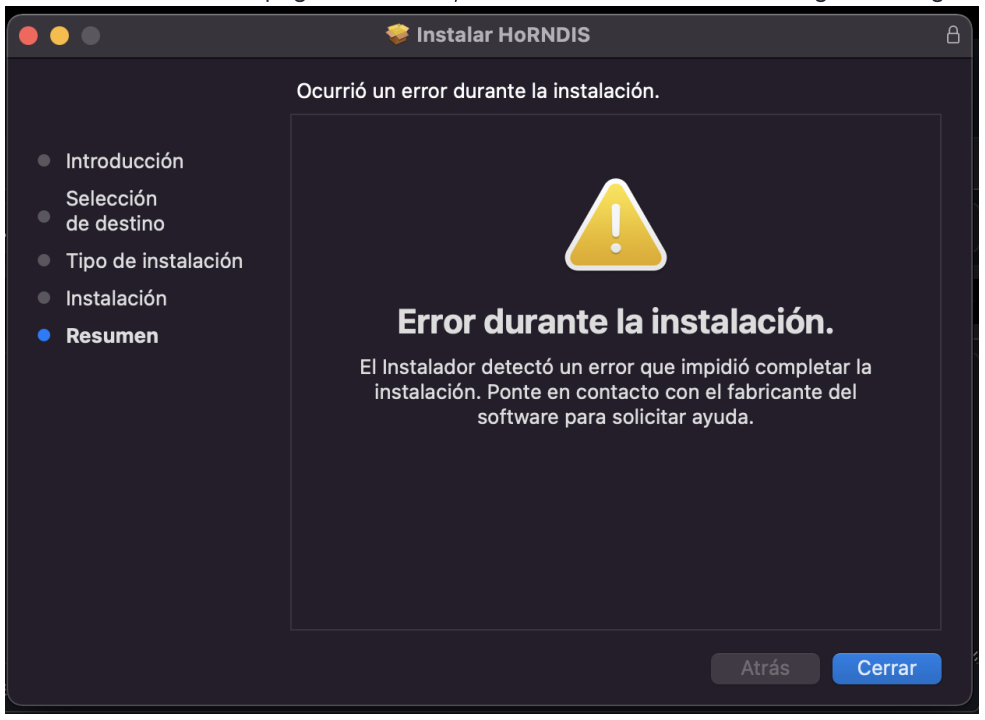
i know Joshua no longer develops Horndis, and Chris1111 has archived his port already so i can not tell Chris of the issue i have. In the instructions it says to type the command below followed by Enter, and i follow this exactly.
/usr/sbin/spctl kext-consent add 54GTJ2AU36
and the terminal returns this error: /usr/sbin/spctl kext-consent add 54GTJ2AU36 spctl: failed to store new configuration.
what is the issue please ?

Reviewed by zaphodd at 2022-01-19 04:09

---

15. Couldnt make it in Moterrey 12.1

At first the links on the page didn't work, had to find a workaround through an incognitos windows.



give me an error message and transfer to the link bellow.
https://support.apple.com/en-us/HT210999

Reviewed by larts85 at 2021-12-19 14:54

Related tags

System HoRNDIS

Native MongoDB driver for Swift, written in Swift

Installation | Tutorial | Basic usage | About BSON | Codable | Community | How to help A fast, pure swift MongoDB driver based on Swift NIO built for

Mar 11, 2022

Native macOS client for Gameboy cartridge readers & writers

Features Native macOS ?? Quick transfer speeds ?? Copy ROMs from your physical cartridges ?? Write ROMs to your own flash carts ⚡ Backup / Restore /

Mar 6, 2022

📻 LocalRadio is "Radio for Cord-Cutters" – a Software-Defined Radio (SDR) app for your Mac and mobile devices. With an inexpensive RTL-SDR USB device, LocalRadio provides a casual, home-based radio listening experience for your favorite local frequencies - FM broadcasts/free music/news/sports/weather/public safety aviation scanner/etc.

LocalRadio The latest pre-release alpha versions of the LocalRadio app are available for download at this link. LocalRadio is "Radio for Cord-Cutters"

Mar 17, 2022

Receive Android notifications on your mac. (w/PushBullet) ⛵

What is Noti? Noti is your little companion living in your mac menu, relaying any notifications you get on your Android phone with Pushbullet installe

Mar 17, 2022

iOS/Android app deployment tool for macOS

TransporterPad Download from here! TranspoterPad is app package deployment tool for everyone of development team. App package file (.ipa/.apk) is very

Feb 12, 2022

Joplin - an open source note taking and to-do application with synchronization capabilities for Windows, macOS, Linux, Android and iOS. Forum: https://discourse.joplinapp.org/

Joplin is a free, open source note taking and to-do application, which can handle a large number of notes organised into notebooks. The notes are sear

Mar 17, 2022

AndroidUtilCode 🔥 is a powerful & easy to use library for Android

AndroidUtilCode ?? is a powerful & easy to use library for Android. This library encapsulates the functions that commonly used in Android development which have complete demo and unit test. By using it's encapsulated APIs, you can greatly improve the development efficiency. The program mainly consists of two modules which is utilcode, which is commonly used in development, and subutil which is rarely used in development, but the utils can be beneficial to simplify the main module. ??

Mar 17, 2022

One-click screenshots, video recordings, app installation for iOS and Android

Android tool for Mac Download Download the latest version here Copy AndroidTool.app to your Applications folder Right click AndroidTool and choose Ope

Mar 15, 2022

Next generation Brave browser for Android, Linux, macOS, Windows.

Brave Browser Overview This repository holds the build tools needed to build the Brave desktop browser for macOS, Windows, and Linux. In particular, i

Mar 18, 2022