



v.0.0.1

# itadOS

Guides

Karl-Markus Oismann  
7-15-2025

## Contents

<b>Create itadOS ISO .....</b>	<b>2</b>
Prerequisites: .....	2
<b>Helpful links for live-build: .....</b>	<b>2</b>
Step 1 – Create a directory .....	3
Step 2 – Create live-build config directory .....	3
Step 3 – Insert itadOS into live-build .....	4
Step 4 – Give itadOS execution permission .....	4
Step 5 – Add the dependencies .....	5
Step 6 – Start itadOS on boot .....	6
Step 7 – Allow laptop lid to be closed.....	7
Step 8 – Legacy boot .....	8
Step 9 – UEFI boot.....	10
Step 10. – Build ISO .....	11
<b>Modify itadOS .....</b>	<b>12</b>
Step 1 – Clean config.....	12
Step 2 – Modify itadOS settings.....	13
Step 3 – Issue config command.....	14
Step 4 – Build ISO .....	14
<b>Create itadOS USB disk.....</b>	<b>16</b>
Step 1 – Get flashing utility. ....	16
Step 2 – Get itadOS ISO.....	16
Step 3 – Flash USB disk with itadOS .....	17

## Create itadOS ISO

### Prerequisites:

- Computer or virtual machine with Debian operating system.
  - The following packages installed:
    - nvme-cli
    - lshw
    - dd (coreutils)
    - hdparm
    - rtcwake (util-linux)
    - mmc-utils
    - whiptail
    - shred (coreutils)
    - smartmontools
    - pciutils
    - fop
    - xsltrpoc
    - live-build
    - isolinux

### Helpful links for live-build:

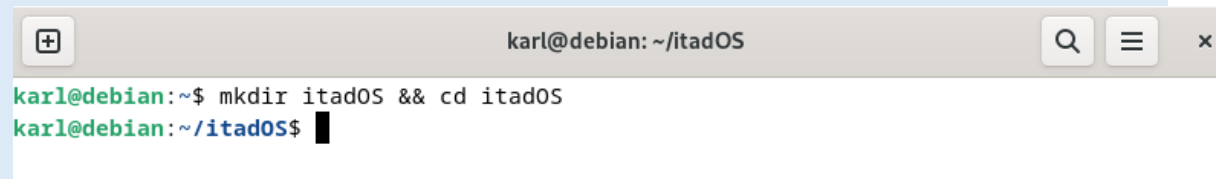
<https://debian-install-notes.pages.dev/netinstall/live-build#3>

[https://manpages.debian.org/testing/live-build/lb\\_config.1.en.html](https://manpages.debian.org/testing/live-build/lb_config.1.en.html)

## Step 1 – Create a directory

- Create a directory and change directory to it, as an in an example shown in figure 1.1.

```
mkdir itadOS && cd itadOS
```

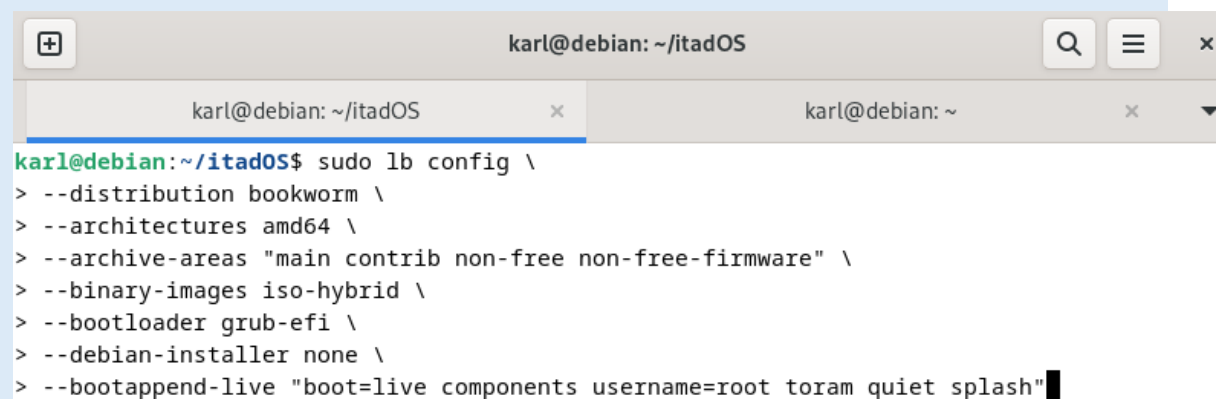


**Figure 1.1.** Directory created for live-build and directory changed to it.

## Step 2 – Create live-build config directory

- Enter the following command, as shown in figure 1.2:

```
sudo lb config --distribution bookworm --architectures amd64 --archive-areas "main contrib non-free non-free-firmware" --binary-images iso-hybrid --bootloader grub-efi --debian-installer none --bootappend-live "boot=live components username=root toram quiet splash"
```



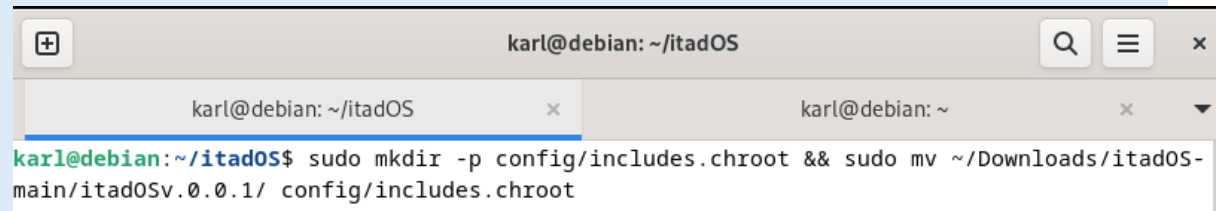
**Figure 1.2.** Live-build config command issued.

## Step 3 – Insert itadOS into live-build

- **Insert itadOS to config/includes.chroot, as shown in figure 1.3.**

(in this example itadOS was downloaded from github: <https://github.com/karloismann/itadOS>)

```
sudo mkdir -p config/includes.chroot && sudo mv ~/Downloads/itadOS-main/itadOSv.0.0.1/ config/includes.chroot
```

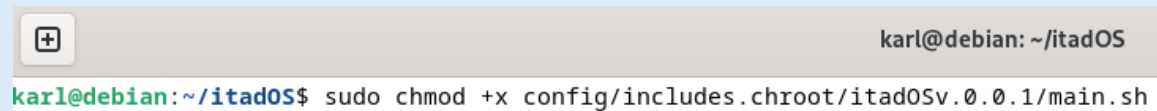


**Figure 1.3.** Directory made for itadOS and it is being moved into it.

## Step 4 – Give itadOS execution permission

- **Issue chmod command, as shown in figure 1.4.**

```
sudo chmod +x config/includes.chroot/itadOSv.0.0.1/main.sh
```

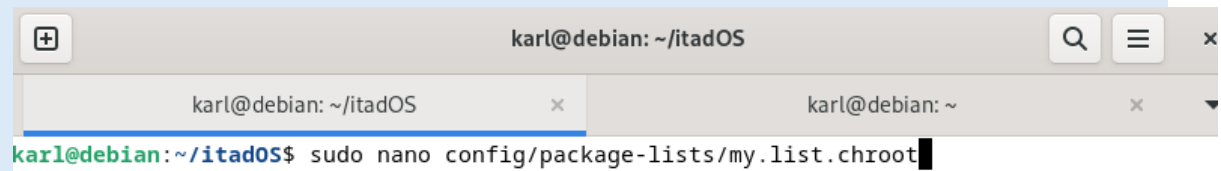


**Figure 1.4.** ItadOS given execution permissions using chmod command.

## Step 5 – Add the dependencies

- Create a file, as shown in figure 1.5.

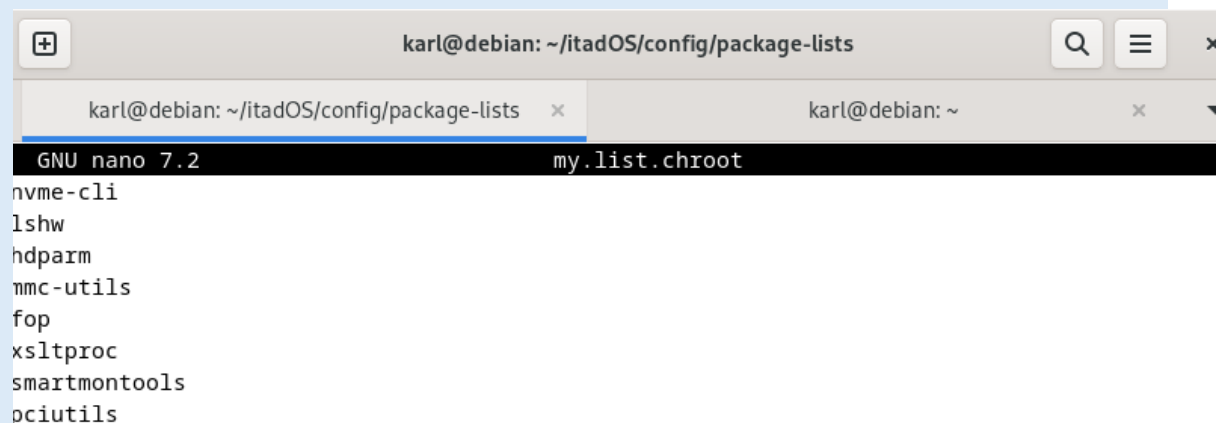
```
sudo nano config/package-lists/my.list.chroot
```



**Figure 1.5.** File for packages is created.

- Insert the following to my.list.chroot, as shown in figure 1.6:

```
nvme-cli  
lshw  
hdparm  
mmc-utils  
fop  
xsltproc  
smartmontools  
pciutils
```

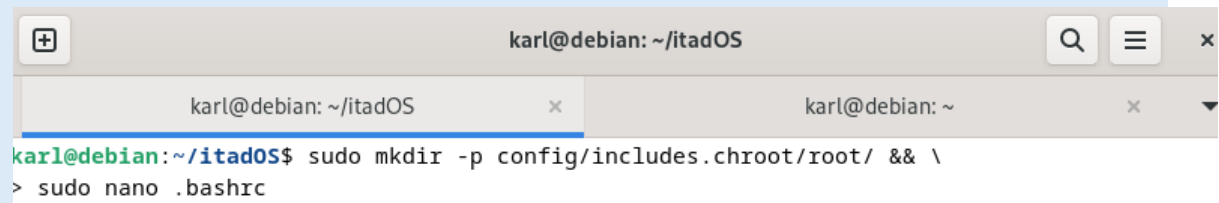


**Figure 1.5.** my.list.chroot filled with required packages.

## Step 6 – Start itadOS on boot

- **Create path and file '.bashrc' and start editing it, as shown in figure 1.6.**

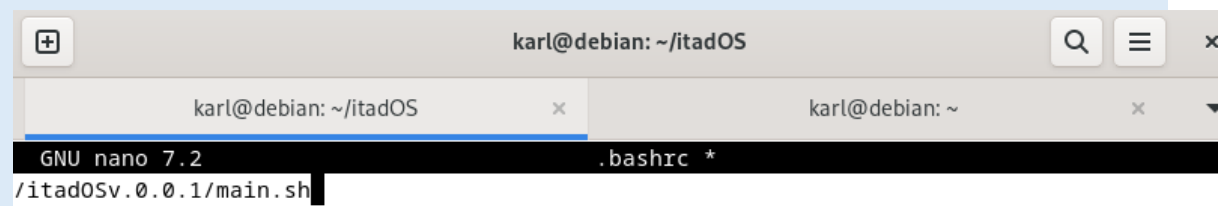
```
sudo mkdir -p config/includes.chroot/root/ && sudo nano .bashrc
```



**Figure 1.6.** Path for .bashrc created and nano command issued.

- **Insert the following line into .bashrc, as shown in figure 1.7:**

```
/itadOSv.0.0.1/main.sh
```

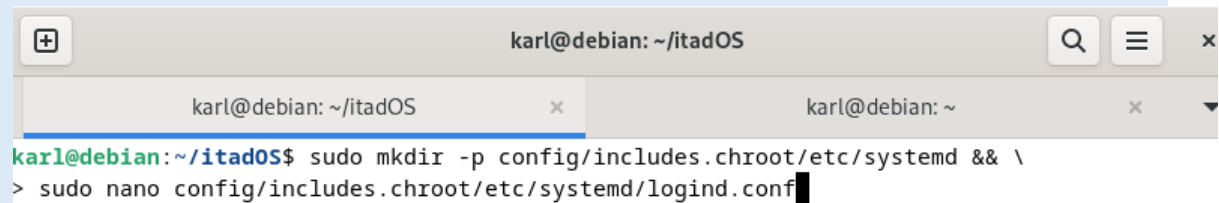


**Figure 1.7.** Bashrc contents.

## Step 7 – Allow laptop lid to be closed

- Create path and file 'logind.conf' and start editing it, as shown in figure 1.8.

```
Sudo mkdir -p config/includes.chroot/etc/systemd && sudo nano  
config/includes.chroot/etc/systemd/logind.conf
```

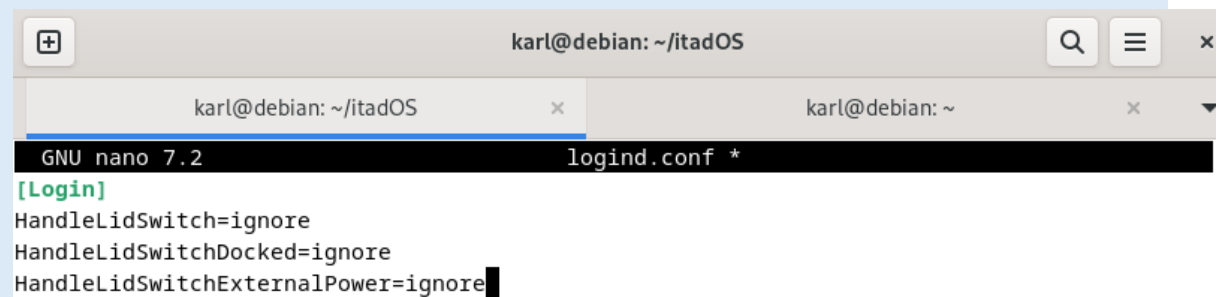


A terminal window titled 'karl@debian: ~/itadOS' showing the command: `karl@debian:~/itadOS$ sudo mkdir -p config/includes.chroot/etc/systemd && \> sudo nano config/includes.chroot/etc/systemd/logind.conf`. The cursor is at the end of the second line.

**Figure 1.8.** Path for logind.conf created and nano command issued.

- Add the following to logind.conf, as shown in figure 1.9:

```
[Login]  
HandleLidSwitch=ignore  
HandleLidSwitchDocked=ignore  
HandleLidSwitchExternalPower=ignore
```



A terminal window titled 'karl@debian: ~/itadOS' showing the nano editor editing 'logind.conf'. The header bar says 'GNU nano 7.2 logind.conf \*'. The content is: `[Login]  
HandleLidSwitch=ignore  
HandleLidSwitchDocked=ignore  
HandleLidSwitchExternalPower=ignore`. The cursor is at the end of the last line.

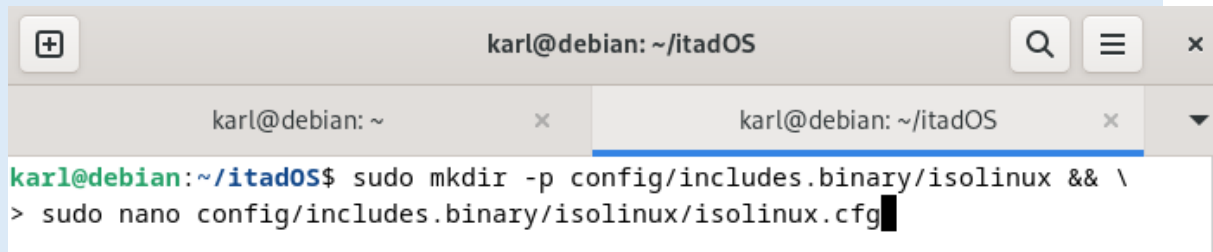
**Figure 1.9.** Logind.conf contents.



## Step 8 – Legacy boot

- **Create path and file, as shown in figure 1.10:**

```
sudo mkdir -p config/includes.chroot/isolinux && sudo nano
config/includes.chroot/isolinux/isolinux.cfg
```



A terminal window titled 'karl@debian: ~/itadOS' showing the command: `karl@debian:~/itadOS$ sudo mkdir -p config/includes.binary/isolinux && \> sudo nano config/includes.binary/isolinux/isolinux.cfg`. The cursor is at the end of the second line.

**Figure 1.10.** Path for `isolinux.cfg` created and `nano` command issued.

- **Insert the following to `isolinux.cfg`, as shown in figure 1.11:**

```
UI vesamenu.c32
PROMPT 0
MENU TITLE itadOS Boot Menu
TIMEOUT 50
DEFAULT live

LABEL live
    MENU LABEL Boot itadOS (Live)
    KERNEL /live/vmlinuz-6.1.0-37-amd64
    APPEND initrd=/live/initrd.img-6.1.0-37-amd64 boot=live components
username=root toram splash
```

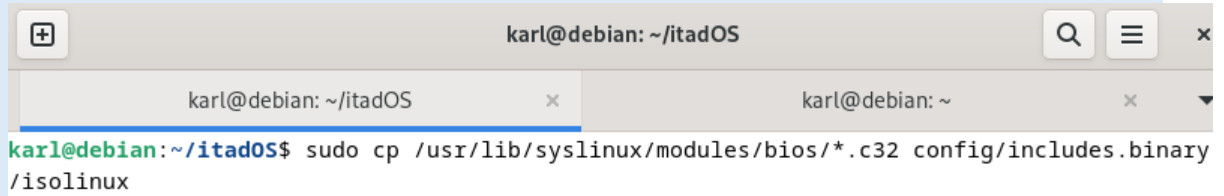


A terminal window titled 'karl@debian: ~/itadOS/config/includes.binary/isolinux' showing the contents of the `isolinux.cfg` file in nano editor. The text matches the code block above. The window title bar includes 'GNU nano 7.2' and 'isolinux.cfg'.

**Figure 1.11.** `isolinux.cfg` contents.

- Also add required binaries from syslinux, as shown in figure 1.12:

```
cp /usr/lib/syslinux/modules/bios/*.c32  
config/includes.binary/isolinux/
```

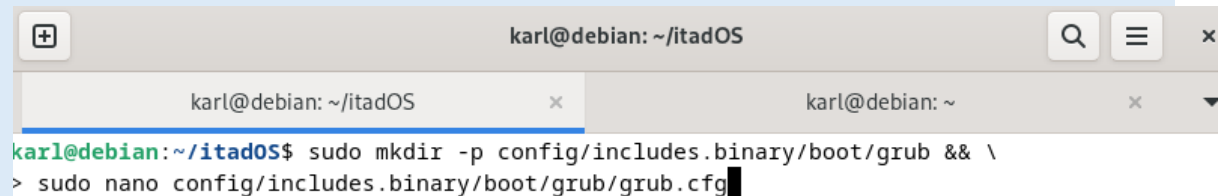


**Figure 1.12.** COM32 modules copied to the destination.

## Step 9 – UEFI boot

- **Create path and file, as shown in figure 1.13:**

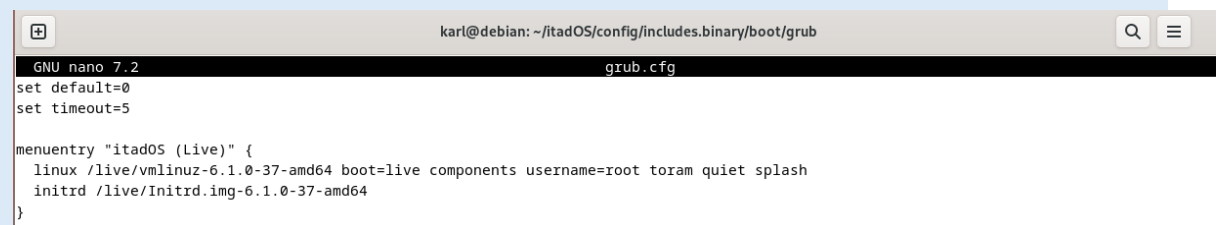
```
sudo mkdir -p config/includes.binary/boot/grub && sudo nano  
config/includes.binary/boot/grub/grub.cfg
```



**Figure 1.13.** Path for grub.cfg created and nano command issued.

- **Insert the following to grub.cfg, as shown in figure 1.14:**

```
set default=0  
set timeout=5  
  
menuentry "itadOS (Live)" {  
    linux /live/vmlinuz-6.1.0-37-amd64 boot=live toram  
    initrd /live/Initrd.img-6.1.0-37-amd64  
}
```

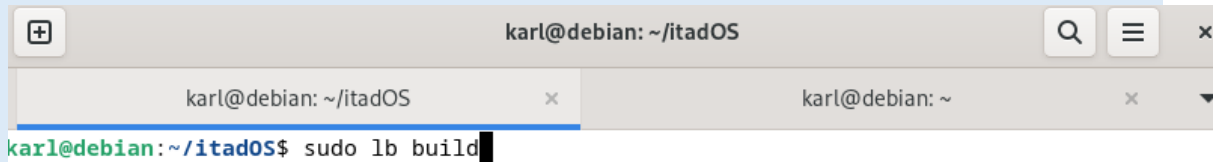


**Figure 1.14.** grub.cfg contents.

## Step 10. – Build ISO

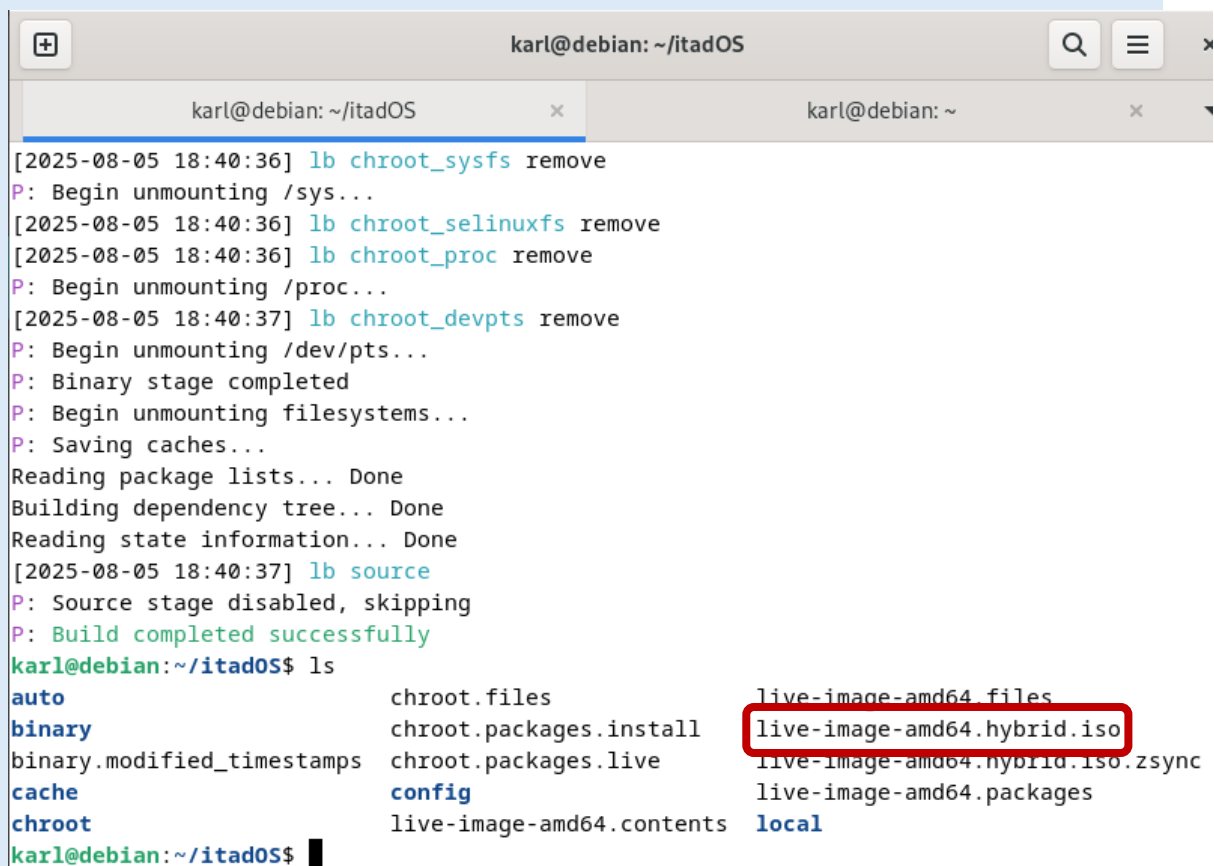
- Enter the following command, as shown in figure 1.15:

```
sudo lb build
```



**Figure 1.15.** Sudo lb build command issued.

- ISO file is now created as shown in figure 1.16.



**Figure 1.16.** ItadOS ISO file is highlighted.

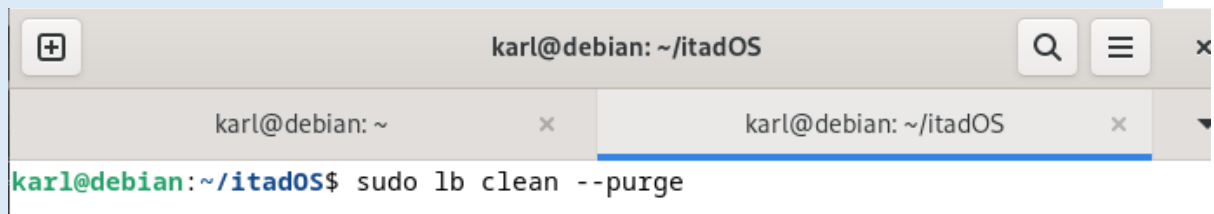
## Modify itadOS

!!! Must complete ‘[Create ISO file](#)’ before continuing. !!!

### Step 1 – Clean config

- Go into the directory created [here](#) and insert the following command, as shown in figure 2.1:

```
sudo lb clean --purge
```

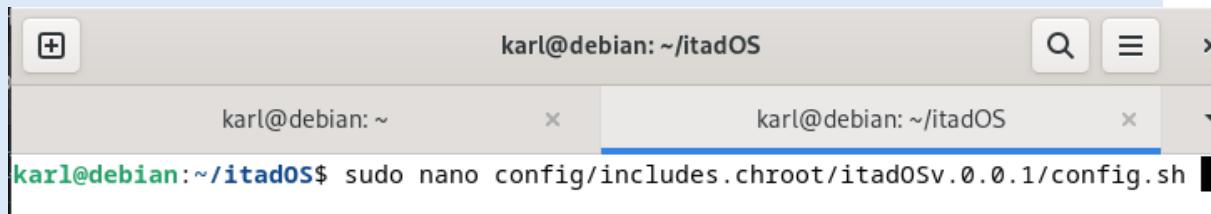


**Figure 1.2.** *sudo lb clean --purge* command issued.

## Step 2 – Modify itadOS settings

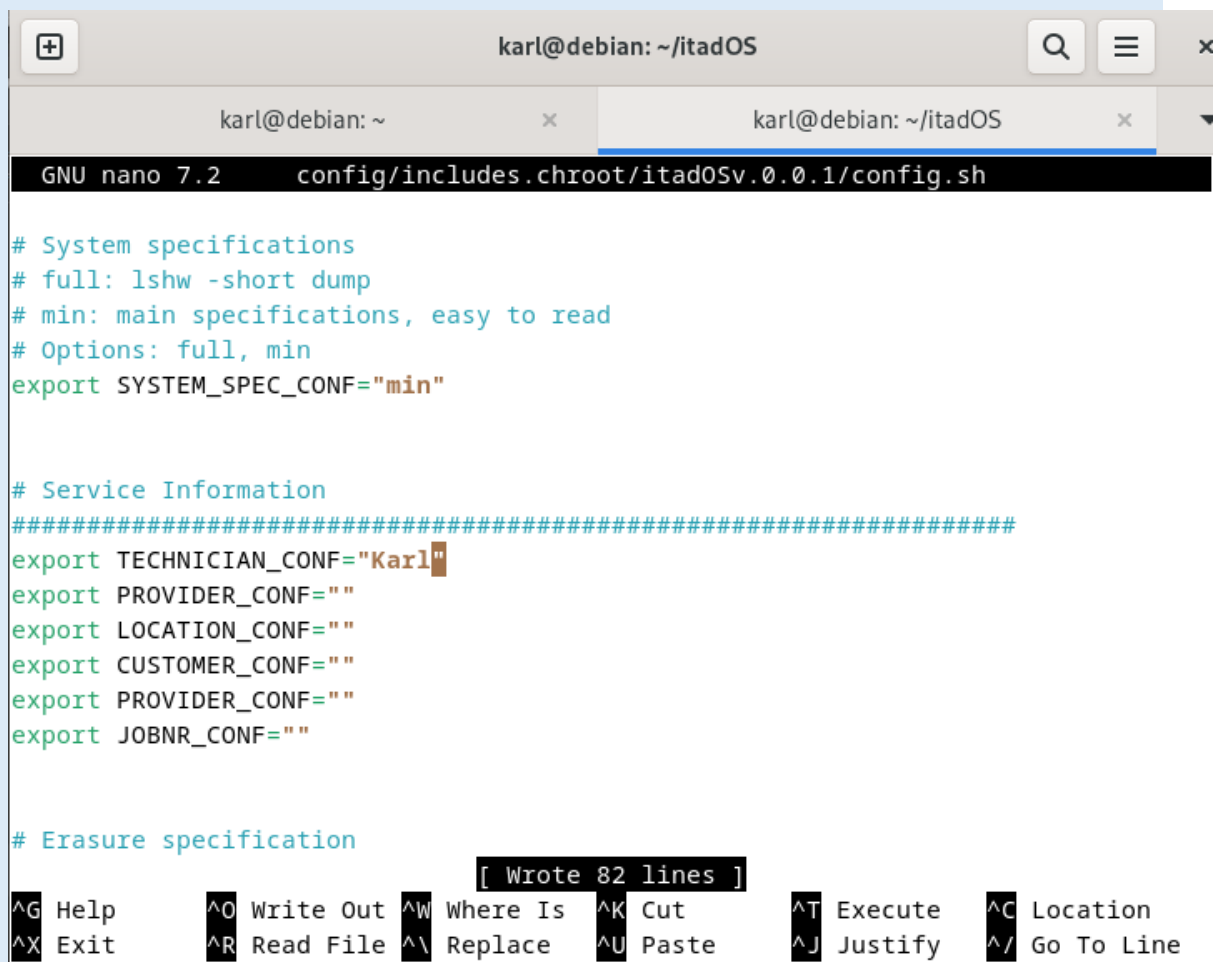
- **Modify settings in config.sh, as shown in figure 2.2.**

```
sudo nano config/includes.chroot/itadOSv.0.0.1/config.sh
```



**Figure 2.2.** Nano command issued for config.sh

- **In this example, I am setting default technician's name to 'Karl', as shown in figure 2.3.**

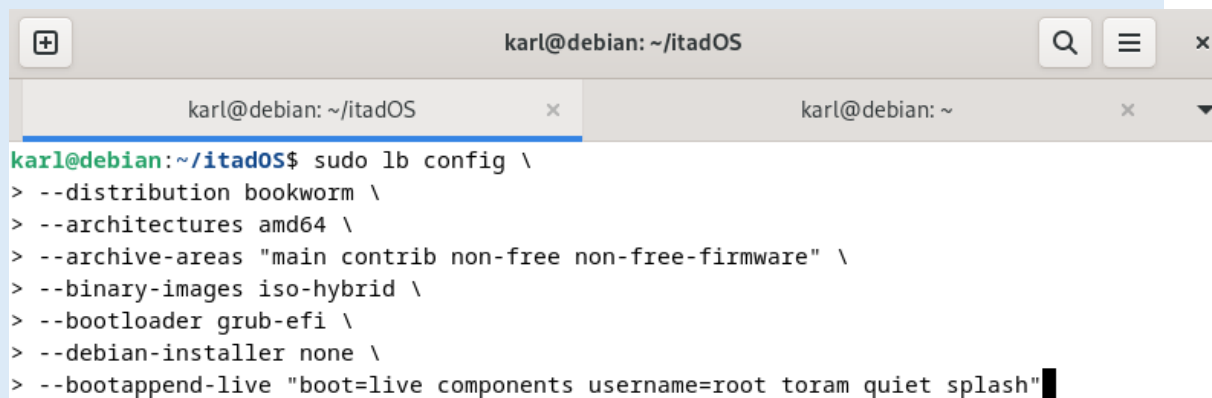


**Figure 2.3.** ItadOS default technician changed to 'Karl'.

## Step 3 – Issue config command

- Enter the following command, as shown in figure 2.4:

```
sudo lb config --distribution bookworm --architectures amd64 --archive-areas "main contrib non-free non-free-firmware" --binary-images iso-hybrid --bootloader grub-efi --debian-installer none --bootappend-live "boot=live components username=root toram quiet splash"
```



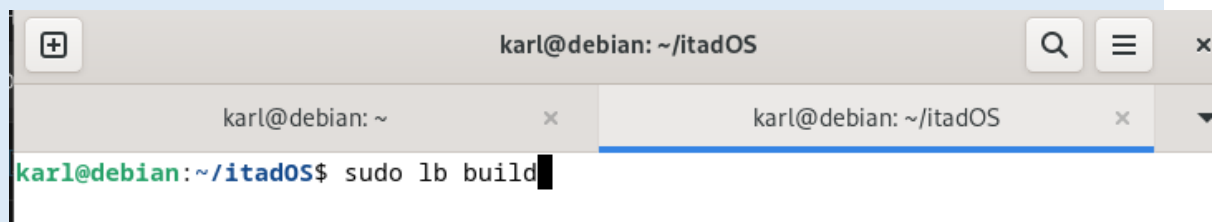
A terminal window titled 'karl@debian: ~/itadOS' showing the execution of the 'sudo lb config' command. The command is entered line by line, with backslashes indicating continuation. The options specified are: --distribution bookworm, --architectures amd64, --archive-areas "main contrib non-free non-free-firmware", --binary-images iso-hybrid, --bootloader grub-efi, --debian-installer none, and --bootappend-live "boot=live components username=root toram quiet splash". The prompt 'karl@debian: ~/itadOS\$' is visible at the start of the command line.

**Figure 2.4.** Live-build config command issued.

## Step 4 – Build ISO

- Enter the following command, as shown in figure 2.5:

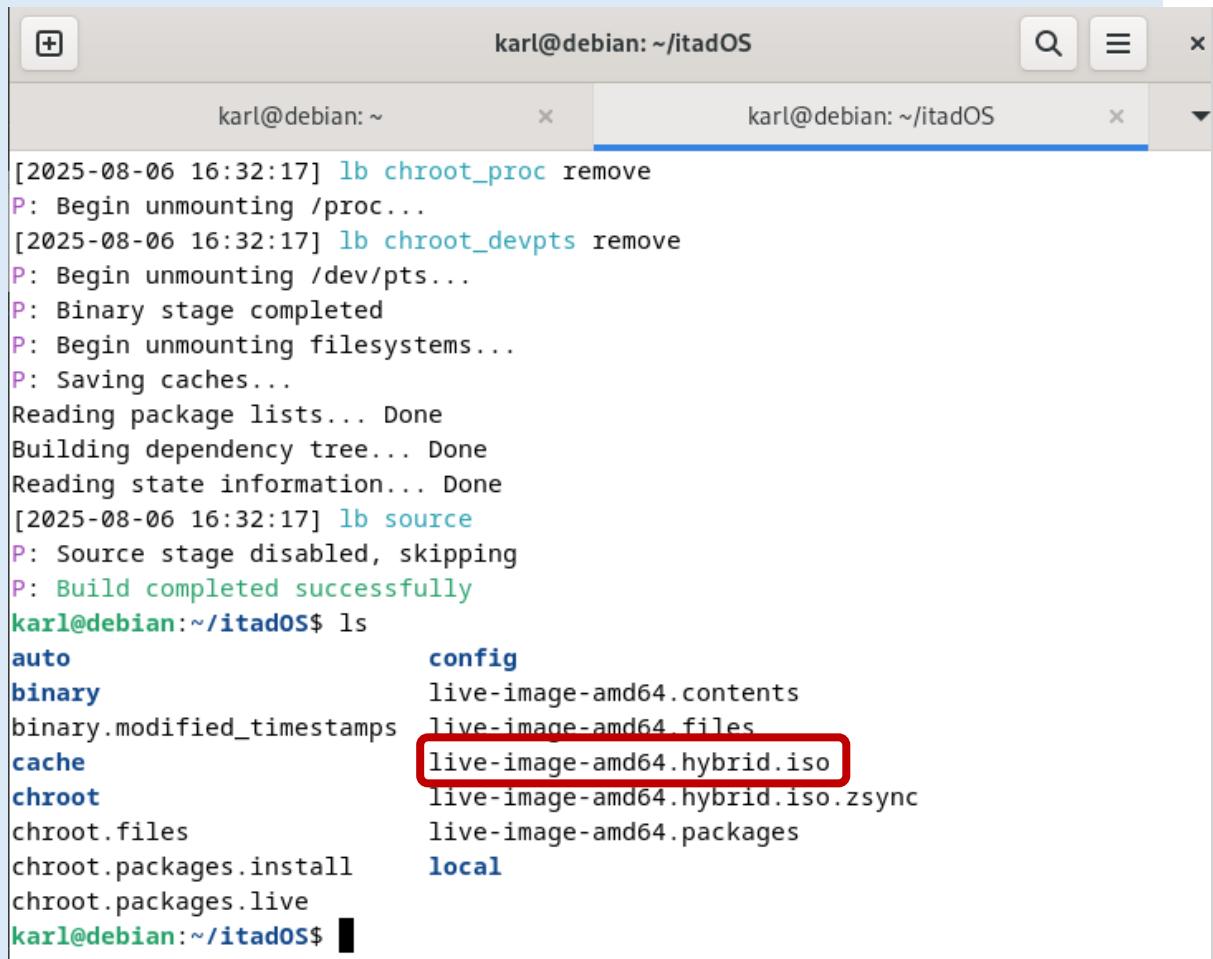
```
sudo lb build
```



A terminal window titled 'karl@debian: ~/itadOS' showing the execution of the 'sudo lb build' command. The command is entered at the prompt 'karl@debian: ~/itadOS\$'. The prompt is highlighted in green. The terminal window has tabs for 'karl@debian: ~/itadOS' and 'karl@debian: ~'.

**Figure 2.5.** Sudo lb build command issued.

- ISO file is now created as shown in figure 2.6.



The image shows a terminal window titled 'karl@debian: ~/itadOS'. The terminal output shows the following sequence of commands and responses:

```
[2025-08-06 16:32:17] lb chroot_proc remove
P: Begin unmounting /proc...
[2025-08-06 16:32:17] lb chroot_devpts remove
P: Begin unmounting /dev/pts...
P: Binary stage completed
P: Begin unmounting filesystems...
P: Saving caches...
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
[2025-08-06 16:32:17] lb source
P: Source stage disabled, skipping
P: Build completed successfully
karl@debian:~/itadOS$ ls
auto                config
binary              live-image-amd64.contents
binary.modified_timestamps live-image-amd64.files
cache                live-image-amd64.hybrid.iso
chroot               live-image-amd64.hybrid.iso.zsync
chroot.files         live-image-amd64.packages
chroot.packages.install local
chroot.packages.live
```

The file `live-image-amd64.hybrid.iso` is highlighted with a red rectangle in the terminal output.

**Figure 1.16.** ItadOS ISO file is highlighted.



## Create itadOS USB disk

This guide is utilising rufus on Windows PC.

Step 1 – Get flashing utility.

- **Install Rufus (windows) or other alternative software.**

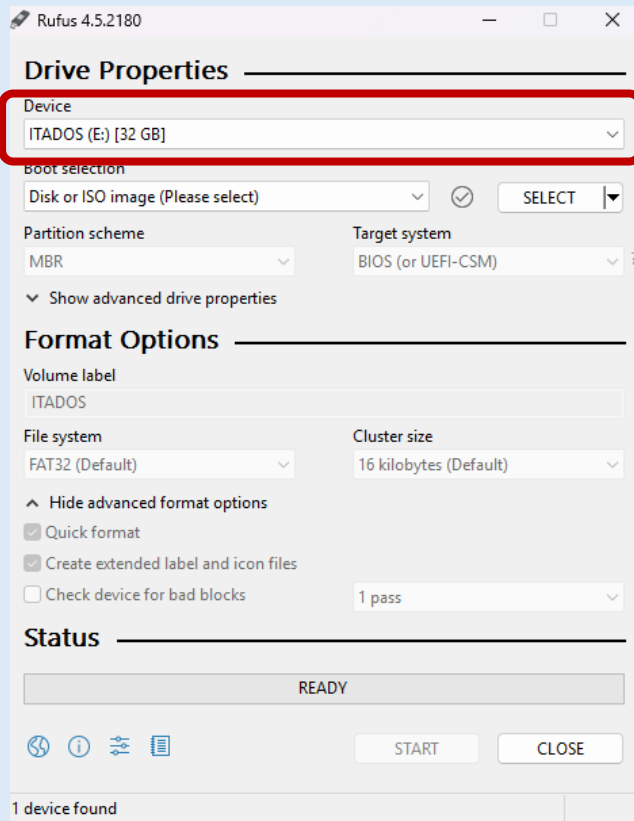
Rufus : <https://rufus.ie/en/>

Step 2 – Get itadOS ISO

- **Download itadOS ISO from [here](#) OR create ISO [here](#).**

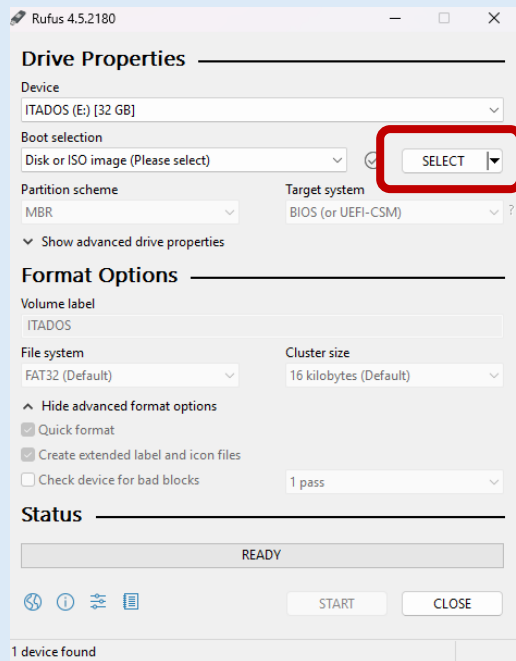
## Step 3 – Flash USB disk with itadOS

- Insert the USB drive into the PC and choose it from the list highlighted in figure 3.1.



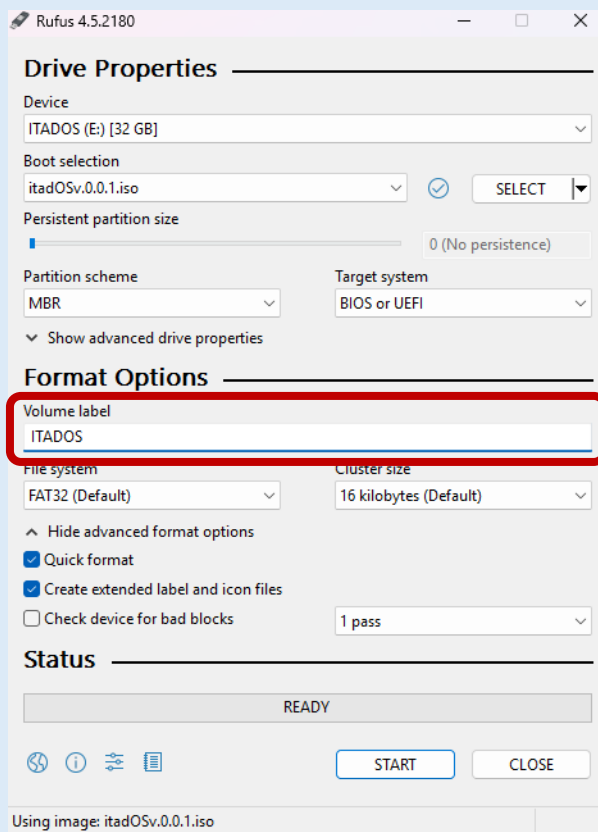
**Figure 3.1.** Rufus' USB drive list highlighted.

- Click on the highlighted button in figure 3.2 and select itadOS ISO.



**Figure 3.2.** ISO selection button in Rufus.

- Name the volume label as 'ITADOS' as highlighted in figure 3.3, and press 'START'.
  - This helps itadOS to identify boot disk.



**Figure 3.3.** Volume label named as 'ITADOS' in Rufus.