

Slackware/Linux Experience

This page is about my experience using Linux in general, and specifically using [Slackware](#). I have also started setting up [Devuan](#) for my family to use.

Slackware Installation

Linux Notes (System)

Useful notes on Linux systems.

Linux Kernel

Kernel-related stuffs.

Intel Motherboard

On startup, got a kernel warning regarding a firmware bug, `TSC_DEADLINE disabled...` or something similar. The required 'fix' is to install `intel-microcode` package.

On [Devuan](#), install the 'parent' package - `firmware-linux` (which covers both `intel-microcode` and `amd64-microcode`)

```
# apt install firmware-linux
```

Initramfs will be auto regenerated.

On [Slackware](#), install `intel-microcode` using script from Slackbuilds.org. Create a new initramfs (initrd) using mknitrd (with `-P /boot/intel-ucode.cpio`).

Kernel Error Message 20210328

After an upgrade on Devuan, I got

```
pstore: crypto_comp_decompress failed, ret = -22!
pstore: decompression failed: -22
```

kernel message after rebooting.

Fixed it by deleting everything in /sys/fs/pstore.

Linux LVM

I have some experience playing with AIX and I am very fascinated by its LVM implementation (which is a MUST-USE). I also found out that Linux also has a slightly different LVM implementation - and obviously optional. I tried it on my home machine and I want to put my personal note here. I will simply put it as it is for now... and maybe discuss with a little more detail later.

[linux_lvm_notes.txt](#)

```
* DISCLAIMER: this is my personal note. use at your own risk.

* My home machine is running Devuan
- note: mount | grep -v -E "/sys|tmpfs|proc|devpts|gvfs"
- current disk layout on /dev/sda (GPT):
  = p1: UEFI partition (fat32) 127M (1M offset)
  = p2: Root partition (ext4) 30G
  = p3: Home partition (ext4) ~898G
  = p4: Swap partition (swap) 4G

@ TASK: Create a new lv from slice of p3
- make sure basic lvm terms/concepts are covered
  = volume group (vg)
  = physical volume (pv) [aix:pv MUST BE a disk]
  = physical element/extent (pe) [aix:pp=physical partition]
  = logical volume (lv)
  = logical element/extent (le) [aix:lp=logical partition]
- need to unmount /home
  = login as root and make sure no user logged in
- target disk layout:
  = p1: UEFI partition (fat32) 127M (1M offset)
  = p2: Root partition (ext4) 30G
  = p3: Home partition (ext4) 600G
  = p4: Xtra partition (lvm) ~298G
  = p5: Swap partition (swap) 4G
- current p3 content is <500G

* repartition disk to create space for new lv
# umount /home
# e2fsck -f /dev/sda3
# resize2fs -p /dev/sda3 500G
# swapoff /dev/sda4
# gdisk /dev/sda
> delete /dev/sda4
> note /dev/sda3 starting sector
> delete /dev/sda3
> create new /dev/sda3 (linux:8300)
  = make sure use same starting sector
```

```
= specify size +600G
> create new /dev/sda4 (linux lvm:8e00)
= default starting sector
= specify -4G to leave space for swap
> create new /dev/sda5 (swap:8200)
= should have 4G remaining
> exit
* reclaim /dev/sda3 space (600G)
# resize2fs -p /dev/sda3
* edit /etc/fstab accordingly
- e.g. specify new uuid for swap & /dev/sda3
* restore swap
# swapon -a
- OR, 'swapon /dev/sda5'

* install lvm
- using lvm2 implementation (uses device mapper)
# apt install lvm2

* create pv
# pvcreate /dev/sda4
* create volume group vg
# vgcreate xtravg /dev/sda4
* playing around (create/remove lv)
# lvcreate -L 100G -n store xtravg
# lvcreate -l +100%FREE -n movie xtravg
# lvremove movie
# lvremove store

* create single lv on vg
# lvcreate -l +100%FREE -n share xtravg
# vgdisplay xtravg
* create ext4 fs on lv
# mkfs.ext4 -L LVMSHARE /dev/xtravg/share
* create a mountpoint for the new lv
# mkdir /ext
* check entry in /dev/mapper/
= should have xtravg-share
* add entry in fstab
= e.g. /dev/mapper/xtravg-share /ext ext4 defaults 0 2
* manually mount
# mount /ext

@ TASK: Add new disk to existing lv
- got a new disk, using whole disk (1TB)
- extend existing lv

# pvcreate /dev/sdb
# vgdisplay xtravg
# vgextend xtravg /dev/sdb
# vgdisplay xtravg
```

```

* use all newly available space
# lvextend -l +100%FREE /dev/xtravg/share

@ TASK: Remove PV from VG
- need to use original /dev/sda4 for something else
- note: pvs -o+pv_used
- PV is in use, so need to free up

# umount /ext
# e2fsck -f /dev/xtravg/share
# resize2fs -p /dev/xtravg/share 800G
# lvreduce -L 850G /dev/xtravg/share
* reclaim space
# resize2fs -p /dev/xtravg/share
* check usage
# pvs -o+pv_used
- OR, pvdisplay
* IF any PEs still in use
# pvmove /dev/sda4
* remove pv from vg
# vgreduce xtravg /dev/sda4
* remove pv from system (no longer id as pv)
# pvremove /dev/sda4
* reclaim all lv space
# lvextend -l +100%FREE /dev/xtravg/share
# resize2fs -p /dev/xtravg/share

```

Linux AUFS

Dumped from my personal note on making AUFS-patched Linux kernel.

[linux_aufs_notes.txt](#)

```

-----
How I built an aufs-patched kernel
-----

- get kernel source
= got 5.4.26 (latest longterm atm)
$ getlinux 5.4.26 -x
- prepare kernel source
= extract
$ tar xf linux-5.4.26.tar.xz
= actually, i got 5.4.25 like a day before 5.4.26 is release
= got patch-5.4.25-26 and apply
$ cd linux-5.4.25
$ patch -p1 < ~/temp/patch-5.4.25-26
= renamed the path as linux-5.4.26

```

```
- get the required aufsX
= need aufs5 because i chose kernel v5
$ git clone git://github.com/sfjro/aufs5-standalone.git
- prepare aufs5
$ git checkout -b aufs5.4 origin/aufs5.4

- get aufs-util (for later use)
$ git clone git://git.code.sf.net/p/aufs/aufs-util
- prepare aufs-util
= aufs-util does not have exact kernel version (get closest lower)
$ git checkout -b aufs-5.0 origin/aufs5.0

- apply patches in aufs5-standalone into linux
= not all patches are required
= refer to README in aufs5-standalone
= i use method 1
$ cd linux-5.4.26
$ patch -p1 < ../aufs5-standalone/aufs5-kbuild.patch
$ patch -p1 < ../aufs5-standalone/aufs5-base.patch
$ patch -p1 < ../aufs5-standalone/aufs5-mmap.patch
$ patch -p1 < ../aufs5-standalone/aufs5-standalone.patch
= OR (in a single line command)
$ for a in kbuild base mmap standalone ; do patch -p1 < ../aufs5-
standalone/aufs5-$a.patch ; done
- need to copy some files
= still in linux-5.4.26
$ cp -av ../aufs5-standalone/Documentation/* Documentation/
$ cp -av ../aufs5-standalone/fs/* fs/
$ cp -av ../aufs5-standalone/include/uapi/linux/aufs_type.h
include/uapi/linux/
- repack this for easy reuse
= rename path as linux-5.4.26_aufs
$ cd .. ; mv linux-5.4.26 linux-5.4.26_aufs
$ cd linux-5.4.26_aufs
$ packthis

- prepare to build
= unpack to /usr/src
$ cd /usr/src ; tar xf linux-5.4.26_aufs.tar.xz
- configure kernel
$ cd linux-5.4.26_aufs
= use make_kernel
$ make_kernel -i -c
= that will run make menuconfig
= select aufs as module
- build kernel
= use make_kernel
$ make_kernel -x
= wait...
```

```
- pack kernel (slackware)
= use pack_kernel (duh!)
$ pack_kernel --all
= by default package will be at $HOME/temp
```

2023/08/29 13:04

Linux Desktop Environment

Some usage tips and fixes for Linux Desktop Environment(s)

JWM stuff

dumped...

[jwm_key_mask.txt](#)

```
jwm mask modifier
- mod key map can be checked using xmodmap -pm
```

My last used config - requires xdgmenumaker (getslackbuild)

[dot-jwmrc](#)

```
<?xml version="1.0"?>
<JWM>
    <RootMenu onroot="3">
        <Program icon="terminal.png" label="Terminal">st -g
110x35</Program>
        <Program icon="file-manager.png" label="File
Manager">rox</Program>
        <Program icon="text-editor.png"
label="Editor">geany_space</Program>
        <Program icon="browser.png" label="Browser">firefox</Program>
        <Separator/>
        <Program icon="drive-hddisk.png" label="Mount
Tool">my1pmount-gtk</Program>
        <Separator/>
        <Include>exec: xdgmenumaker -n -i -f jwm</Include>
        <Separator/>
        <Program icon="reload3.png" label="Reload Menu">jwm -
reload</Program>
        <Restart icon="reload.png" label="Restart JWM"/>
        <Exit icon="exit.png" label="Exit JWM"/>
        <Separator/>
```

```
<Program icon="redo.png" label="Reboot">logindctl  
reboot</Program>  
<Program icon="system-shutdown.png" label="Shutdown">logindctl  
poweroff</Program>  
</RootMenu>  
<Group>  
    <Name>st</Name>  
    <Option>cascaded</Option>  
</Group>  
<Tray x="0" y="0" autohide="off">  
    <TrayButton label="MY1">root:3</TrayButton>  
    <Spacer width="2"/>  
    <TrayButton label="Desktop">showdesktop</TrayButton>  
    <Spacer width="2"/>  
    <Pager labeled="true"/>  
    <TaskList maxwidth="256"/>  
    <Dock/>  
    <Clock format="%Y%m%d %H:%M"></Clock>  
</Tray>  
<WindowStyle>  
    <Font>Sans-10:bold</Font>  
    <Width>4</Width>  
    <Height>21</Height>  
    <Corner>3</Corner>  
    <Foreground>#FFFFFF</Foreground>  
    <Background>#555555</Background>  
    <Outline>#000000</Outline>  
    <Opacity>0.5</Opacity>  
    <Active>  
        <Foreground>#FFFFFF</Foreground>  
        <Background>#0077CC</Background>  
        <Outline>#000000</Outline>  
        <Opacity>1.0</Opacity>  
    </Active>  
</WindowStyle>  
<TrayStyle group="true" list="desktop">  
    <Font>Sans-10</Font>  
    <Background>#333333</Background>  
    <Foreground>#FFFFFF</Foreground>  
    <Outline>#000000</Outline>  
    <Opacity>0.75</Opacity>  
</TrayStyle>  
<TaskListStyle>  
    <Font>Sans-10</Font>  
    <Active>  
        <Foreground>#FFFFFF</Foreground>  
        <Background>#555555</Background>  
    </Active>  
    <Foreground>#FFFFFF</Foreground>  
    <Background>#333333</Background>  
</TaskListStyle>
```

```
<PagerStyle>
    <Outline>#000000</Outline>
    <Foreground>#555555</Foreground>
    <Background>#333333</Background>
    <Text>#FFFFFF</Text>
    <Active>
        <Foreground>#0077CC</Foreground>
        <Background>#004488</Background>
    </Active>
</PagerStyle>
<MenuStyle>
    <Font>Sans - 10</Font>
    <Foreground>#FFFFFF</Foreground>
    <Background>#333333</Background>
    <Outline>#000000</Outline>
    <Active>
        <Foreground>#FFFFFF</Foreground>
        <Background>#0077CC</Background>
    </Active>
    <Opacity>0.85</Opacity>
</MenuStyle>
<PopupStyle>
    <Font>Sans - 10</Font>
    <Foreground>#000000</Foreground>
    <Background>#999999</Background>
</PopupStyle>
<IconPath>/usr/share/icons/hicolor/16x16/apps</IconPath>
<IconPath>/usr/share/icons/Tango/16x16/apps</IconPath>
<IconPath>/usr/share/icons/Tango/16x16/actions</IconPath>
<IconPath>/usr/share/icons/Tango/16x16/devices</IconPath>
<IconPath>/usr/share/icons/Tango/16x16/categories</IconPath>
<IconPath>/usr/share/icons/Tango/16x16/mimetypes</IconPath>
<IconPath>/usr/share/icons/Adwaita/16x16/apps</IconPath>
<IconPath>/usr/share/icons/Adwaita/16x16/actions</IconPath>
<IconPath>/usr/share/icons/Adwaita/16x16/categories</IconPath>
<IconPath>/usr/share/pixmaps</IconPath>
<IconPath>/usr/share/jwm</IconPath>
<Desktops width="2" height="1">
    <!--<Background type="solid">#111111</Background>-->
    <Background
type="scale">/usr/share/jwm/slackwall.png</Background>
</Desktops>
<!-- Double click speed (in milliseconds) -->
<DoubleClickSpeed>400</DoubleClickSpeed>
<!-- Double click delta (in pixels) -->
<DoubleClickDelta>2</DoubleClickDelta>
<!-- The focus model (sloppy or click) -->
<FocusModel>click</FocusModel>
<!-- The snap mode (none, screen, or border) -->
<SnapMode distance="10">border</SnapMode>
<!-- The move mode (outline or opaque) -->
```

```
<MoveMode>opaque</MoveMode>
<!-- The resize mode (outline or opaque) -->
<ResizeMode>opaque</ResizeMode>
<!-- Key bindings -->
<Key key="Up">up</Key>
<Key key="Down">down</Key>
<Key key="Right">right</Key>
<Key key="Left">left</Key>
<Key key="h">left</Key>
<Key key="j">down</Key>
<Key key="k">up</Key>
<Key key="l">right</Key>
<Key key="Return">select</Key>
<Key key="Escape">escape</Key>
<Key mask="A" key="p">exec:dmenu_run -b</Key>
<Key mask="A" key="Tab">nextstacked</Key>
<Key mask="A" key="F4">close</Key>
<Key mask="A" key="#">desktop#</Key>
<Key mask="A" key="F1">root:3</Key>
<Key mask="A" key="F2">window</Key>
<Key mask="A" key="F5">move</Key>
<Key mask="A" key="F6">resize</Key>
<Key mask="A" key="F10">maximize</Key>
<Key mask="A" key="F11">fullscreen</Key>
<Key mask="A" key="Right">rdesktop</Key>
<Key mask="A" key="Left">ldesktop</Key>
<Key mask="A" key="Up">udesktop</Key>
<Key mask="A" key="Down">ddesktop</Key>
<Key mask="CA" key="Up">maxtop</Key>
<Key mask="CA" key="Down">maxbottom</Key>
<Key mask="CA" key="Right">maxright</Key>
<Key mask="CA" key="Left">maxleft</Key>
<Key mask="C4" key="v">maxv</Key>
<Key mask="C4" key="h">maxh</Key>
<Key mask="CA" key="l">exec:xlock -mode matrix</Key>
<Key mask="CA" key="m">exec:my1pmount-gtk</Key>
<Key mask="CA" key="plus">exec:volume +</Key>
<Key mask="CA" key="minus">exec:volume -</Key>
<Key mask="AS" key="Return">exec:st -g 110x35</Key>
<StartupCommand>nm-applet</StartupCommand>
<SnapMode>none</SnapMode>
</JWM>
```

DWM stuff

dumped...

[dwm_hotkeys.txt](#)

Mod = Alt

[Shift]+[Mod]+[Enter]	- launch terminal.
[Mod]+[b]	- show/hide bar.
[Mod]+[p]	- dmenu
[Mod]+[Enter]	- swap master with top stack
[Mod] + [j / k]	- focus on next/previous window in current tag.
[Mod] + [h / l]	- increases / decreases master size.
[Mod]+[2]	- moves your focus to tag 2.
[Shift]+[Mod]+[2]	- move active window to the 2 tag.
[Mod] + [i / d] master	- increases / decreases number of windows on master
[Mod] + [, / .] setup	- move focus between screens (multi monitor setup)
[Shift]+[Mod]+[, / .]	- move active window to different screen.
[Mod]+[0]	- view all windows on screen.
[Shift]+[Mod]+[0]	- make focused window appear on all tags.
[Shift]+[Mod]+[c]	- kill active window.
[Shift]+[Mod]+[q]	- quit dwm cleanly.
[Mod]+[t]	- tiling mode
[Mod]+[f]	- floating mode
[Mod]+[m]	- monocle mod
[Mod]+[R M B]	- to resize the floating window.
[Mod]+[L M B]	- to move the floating window around.
[Mod]+[Space]	- toggles to the previous layout mode.
[Mod]+[Shift]+[Space]	- to make an individual window float.
[Mod]+[M M B]	- to make an individual window un-float.

[st_hotkeys.txt](#)

[Shift]+[Mod]+[PageUp]	- increase font size
[Shift]+[Mod]+[PageDn]	- decrease font size
[Shift]+[Mod]+[Home]	- reset font size

Generic Stuff

Applicable to most (if not all) DE/WM.

Geany on multiple workspaces

[geany_space](#)

```
#!/bin/bash
# from:
https://unix.stackexchange.com/questions/246827/geany-open-a-new-instance-per-workspace-when-opening-a-file-in-that-workspace

socket=$(xprop -root _NET_CURRENT_DESKTOP)
socket=${socket##* }

if [ "$socket" ] ; then
    if [ "$DISPLAY" ] ; then
        socket="${DISPLAY%.}.$socket"
        socket=${socket#:}
    else
        socket="NODISPLAY.$socket"
    fi
    exec geany --socket-file "/tmp/geany_socket_$socket" "$@"
else
    exec geany "$@"
fi
```

[geany_space.desktop](#)

```
[Desktop Entry]
Type=Application
Version=1.0
Name=Geany (Workspace)
Comment=Geany wrapper to detect workspace
Exec=geany_space %F
Icon=geany
Terminal=false
Categories=Development;IDE;
MimeType=text/plain;text/x-chdr;text/x-csrc;text/x-c++hdr;text/x-
c++src;text/x-java;text/x-dsrc;text/x-pascal;text/x-perl;text/x-
python;application/x-php;application/x-htpd-php3;application/x-htpd-
php4;application/x-htpd-
php5;application/xml;text/html;text/css;text/x-sql;text/x-diff;
StartupNotify=true
Keywords=Text;Editor;
```

Useful Desktop Files

The *.desktop files provide menu entry for applications.

[arduino-ide.desktop](#)

```
[Desktop Entry]
Type=Application
Version=1.0
Name=Arduino IDE (App-Image)
Comment=Arduino IDE
Exec=/home/share/appx/app-image/arduino-ide %F
#Icon=
Terminal=false
Categories=Development;IDE;
MimeType=text/plain;text/x-chdr;text/x-csrc;text/x-c++hdr;text/x-
c++src;text/x-java;text/x-dsrc;text/x-pascal;text/x-perl;text/x-
python;application/x-php;application/x-htpd-php3;application/x-htpd-
php4;application/x-htpd-
php5;application/xml;text/html;text/css;text/x-sql;text/x-diff;
StartupNotify=true
#Keywords=Text;Editor;
```

[android-studio.desktop](#)

```
[Desktop Entry]
Version=1.0
Type=Application
Name=Android Studio
Comment=
Exec=env JAVA_HOME=/home/share/appx/android-studio/jbr
/home/share/appx/android-studio/bin/studio.sh
Icon=/home/share/appx/android-studio/bin/studio.png
Path=/home/share/appx/android-studio/bin
Categories=Development;IDE;
Terminal=false
StartupWMClass=jetbrains-studio
StartupNotify=true
```

[xfig.desktop](#)

```
[Desktop Entry]
Encoding=UTF-8
Version=1.0
Type=Application
NoDisplay=true
Exec=/usr/bin/xfig -nowrite_bak %f
Name=xfig -nowrite_bak
Comment=Custom definition for xfig -nowrite_bak
```

Using xdg-mime

Dumped

```
- get mime type of <filename>
$ xdg-mime query filetype <filename>

- get currently set default app for a <mime/type>
$ xdg-mime query default <mime/type>

- set a new default app for a <mime/type>
$ xdg-mime default app.desktop <mime/type>
```

User Applications

Some specific use-cases (for my personal reference)

- Video subtitle (*.srt) editor - gaopol (apt,slackbuilds)

XFCE: Slackware64-15.0

Icons for extra/unused partitions on my hard disk are shown on XFCE desktop. I DO NOT want this!

- Create a udev rule in udev/rules.d path

10-hdpart.rules

```
KERNEL=="sda3", ENV{UDISKS_IGNORE}="1"
KERNEL=="sda4", ENV{UDISKS_IGNORE}="1"
```

- add more lines as required
- Reload udev rules

```
udevadm control --reload-rules && udevadm trigger
```

XFCE: Clock Panel

Custom Digital Clock settings on XFCE

```
<span size='x-small'>%Y%m%d%n</span><b>%H:%M:%S</b>

<span size='x-small'><i>%Y-%m-%d</i>%n</span><span
size='small'><b>%T</b></span>

<span size='small'><b>%Y%m%d</b></span> %H:%M:%S
```

XFCE: Terminal Shortcut for Thunar

Note20230501 No longer needed? Simply go Edit>Configure custom actions... and set it there.

There is already a User Custom Action (UCA) defined to 'Open Terminal Here' - so we just need to add a shortcut.

- Find that UCA id from `~/.config/Thunar/uca.xml`
- Locate that id in `~/.config/Thunar/accels.scm`
 - uncomment and add shortcut key, e.g. F4

XFCE: Quake-Style Terminal

XFCE4 terminal can actually run in a drop-down mode (Quake-style):

```
xfce4-terminal --drop-down --hide-menubar
```

Use keyboard settings (Settings→Keyboard): Create new entry in 'Application Shortcuts' tab with the previous command, with `<F12>` as trigger key.

XFCE: Remove Suspend/Hibernate Button

```
xfconf-query -c xfce4-session -np '/shutdown>ShowSuspend' -t 'bool' -s  
'false'  
xfconf-query -c xfce4-session -np '/shutdown>ShowHibernate' -t 'bool' -s  
'false'  
xfconf-query -c xfce4-session -np '/shutdown>ShowHybridSleep' -t 'bool' -s  
'false'
```

Sometimes this is also annoying?

```
xfconf-query -c xfce4-session -np '/shutdown>ShowSwitchUser' -t 'bool' -s  
'false'
```

XFCE: Multi-User

Note: Latest version should be [here](#)

Copying XFCE settings for other users:

[clone-xfceconf](#)

```
#!/bin/bash
```

```
DSTUSER=$1
[ -z "$DSTUSER" ] && echo "** Missing target username!" && exit 1

SRCUSER=$(pwd)
SRCTEST=$(dirname $SRCUSER)

while [ 1 ] ; do
[ "$SRCTEST" == "/" ] &&
    echo "** Cannot find /home in current path!" && exit 1
[ "$SRCTEST" == "/home" ] && break;
SRCUSER=$(basename $SRCTEST)
SRCTEST=$(dirname $SRCTEST)
done
echo "-- Using user '$SRCUSER' as source!"

DSTHOME="/home/$DSTUSER"
SRCHOME="/home/$SRCUSER"

[ "$DSTUSER" == "$SRCUSER" ] &&
    echo "** Target ($DSTUSER) same as source ($SRCUSER)?" && exit 1

[ ! -d "$DSTHOME" ] &&
    echo "** Cannot find target user path '$DSTHOME'!" && exit 1

[ ! -d "$SRCHOME" ] &&
    echo "** Cannot find source user path '$SRCHOME'!" && exit 1

DST_GRP=$(groups $DSTUSER | sed 's/.*/\1.*$/\1/' )
echo -- Target:{$DSTUSER:$DST_GRP}
[ $UID -ne 0 ] && echo -e "\n** Must run as root!\n" && exit 1

CHKCONF="Thunar geany xfce4"
for conf in $CHKCONF ; do
    SRCCONF="$SRCHOME/.config/$conf"
    [ ! -d "$SRCCONF" ] && continue
    DSTCONF="$DSTHOME/.config/$conf"
    if [ -d "$DSTCONF" ] ; then
        echo -n "-- Deleting $DSTCONF... "
        rm -rf $DSTCONF
        [ $? -ne 0 ] && echo -n "**error0** "
        echo "done."
    fi
    echo -n "-- Copying $SRCCONF as $DSTCONF... "
    cp -a $SRCCONF $(dirname $DSTCONF)
    [ $? -ne 0 ] && echo -n "**error1** "
    chown -R $DSTUSER:$DST_GRP $DSTCONF
    [ $? -ne 0 ] && echo -n "**error2** "
    echo "done."
done
```

Cinnamon: Terminal Shortcut for Nemo

Shortcut for nemo file manager

- \$HOME/.gnome2/accels/nemo

```
(gtk_accel_path "<Actions>/DirViewActions/OpenInTerminal" "F4")
```

Cinnamon: Shortcuts

Note: Super key is the Windows key

[shortcuts.txt](#)

```
Ctrl+Alt+Down: Show the window selection screen
Ctrl+Alt+Up: Show the workspace selection screen
Ctrl+Alt+End: Shut down dialog
Super+D: Show the desktop
Alt+F2: Run dialog
Alt+F8: Resize window
Alt+F7: Move window
Super+Left: Push tile left
Super+Right: Push tile right
Super+Up: Push tile up
Super+Down: Push tile down
Ctrl+Super+Left: Push snap left +
Ctrl+Super+Right: Push snap right
Ctrl+Super+Down: Push snap down
Shift+Ctrl+Alt+Left: Move window to left workspace
Shift+Ctrl+Alt+Right: Move window to right workspace
Super+Shift+Left: Move window to left monitor
Super+Shift+Right: Move window to right monitor
Super+Shift+Up: Move window to up monitor
Super+Shift+Down: Move window to down monitor
Alt+F5: Un-maximize window
Alt+F4: Minimize window
Alt+Space: Activate window menu
Alt+F10: Toggle maximization state
```

[shortcuts_check.txt](#)

```
Super+P: Re-detect display devices
```

Cinnamon: Disable Recent Files

Option is in Menu > Preferences > Privacy

2023/08/29 13:04

Linux Notes (User?)

Some useful things to know in Linux.

DNS Privacy

Dumped

[dns_privacy.txt](#)

```
public/free dns:  
dns.quad9.net  
9.9.9.9  
149.112.112.112  
2620:fe::fe  
2620:fe::9
```

Recursive DNS Server Addresses and Features - Service based configuration:

Recommended: Malware Blocking, DNSSEC Validation (this is the most typical configuration)

IPv4
9.9.9.9
149.112.112.112

IPv6
2620:fe::fe
2620:fe::9

HTTPS
<https://dns.quad9.net/dns-query>

TLS
<tls://dns.quad9.net>

Secured w/ECS: Malware blocking, DNSSEC Validation, ECS enabled

IPv4

```
9.9.9.11  
149.112.112.11
```

```
IPv6  
2620:fe::11  
2620:fe::fe:11
```

```
HTTPS  
https://dns11.quad9.net/dns-query
```

```
TLS  
tls://dns11.quad9.net
```

```
Unsecured: No Malware blocking, no DNSSEC validation (for experts only!)
```

```
IPv4  
9.9.9.10  
149.112.112.10
```

```
IPv6  
2620:fe::10  
2620:fe::fe:10
```

```
HTTPS  
https://dns10.quad9.net/dns-query
```

```
TLS  
tls://dns10.quad9.net
```

Code Debugging

Note on using valgrind...

```
$ valgrind --leak-check=full --track-origins=yes --log-file=valgrind.rpt \  
./myasm85 ../myasm85/asm/monitor85.asm  
  
$ valgrind --tool=memcheck --leak-check=full --track-origins=yes -s \  
./myasm85 ../myasm85/asm/monitor85.asm
```

User Admin

To disable user login

```
# passwd -l username
```

To re-enable user login

```
# passwd -u username
```

OR,

To disable user login

```
# usermod -L -e 1 username
```

To re-enable user login

```
# usermod -U -e "" username
```

I prefer/use the first one.

Avahi/mDNS Stuff

List all hosts in *.local

```
$ avahi-browse -atr | grep hostname | sort -u
```

Terminal Display

To display extended ASCII (which is a non-standard), use Western-IBM850 encoding (instead of the currently common UTF-8). The extended ASCII is actually great in drawing boxes in text mode.

Manipulating PDF

To merge multiple PDFs (using pdftk):

```
pdftk $(ls *.pdf | sort) cat output ../output.pdf
```

Use xournal (or, maybe, xournalpp) to mark/sign - used export PDF to recreate PDF. If the created PDF is big (tends to be), import from LibreOffice Draw and re-export PDF.

Cleanup Stuff

Just found out about /proc/sys/vm/drop_caches kernel interface today. Need to look at this some time...

```
* clear page cache {flag:0x01}
# echo 1 > /proc/sys/vm/drop_caches
```

```
* clear dentries and inodes {flag:0x02}
# echo 2 > /proc/sys/vm/drop_caches
* can request simultaneously {flag:0x02|0x01}
# echo 3 > /proc/sys/vm/drop_caches
```

Setup User Development Space (Isolated?)

Got this info from somewhere online - I just put it into a script.

Note: Create access to an IPad... requires ifuse and libimobiledevice-utils packages (tested on Devuan).

setup_dev_env

```
#!/bin/bash

# setup_dev_env
# - got this info from internet, i just put it in a script
# - should source this file :p

WORK_PATH=`pwd`
[ -d "$1" ] && WORK_PATH=`cd $1;pwd`
echo "-- Setup environment in ${WORK_PATH}"

MAKE_PATH=0
add_path()
{
    [ -d "$1" ] && PATH="$1":$PATH && MAKE_PATH=1
}
add_path ${WORK_PATH}/bin
add_path ${WORK_PATH}/sbin
add_path ${WORK_PATH}/usr/bin
add_path ${WORK_PATH}/usr/sbin
[ $MAKE_PATH -eq 0 ] &&
    echo "** No binary path found!" && exit 1
export PATH

MAKE_PATH=0
add_ldlib_path()
{
    if [ -d "$1" ] ; then
        [ -z "$LD_LIBRARY_PATH" ] &&
            LD_LIBRARY_PATH="$1" ||
        LD_LIBRARY_PATH="$1":$LD_LIBRARY_PATH
            MAKE_PATH=1
    fi
}
add_ldlib_path ${WORK_PATH}/lib
add_ldlib_path ${WORK_PATH}/usr/lib
```

```
[ $MAKE_PATH -ne 0 ] && export LD_LIBRARY_PATH

MAKE_PATH=0
add_c_path()
{
    if [ -d "$1" ] ; then
        [ -z "$CPATH" ] &&
            CPATH="$1" || CPATH="$1":$CPATH
        MAKE_PATH=1
    fi
}
add_c_path ${WORK_PATH}/include
add_c_path ${WORK_PATH}/usr/include
[ $MAKE_PATH -ne 0 ] && export CPATH

MAKE_PATH=0
add_man_path()
{
    if [ -d "$1" ] ; then
        [ -z "$MANPATH" ] &&
            MANPATH="$1" || MANPATH="$1":$MANPATH
        MAKE_PATH=1
    fi
}
add_man_path ${WORK_PATH}/share/man
add_man_path ${WORK_PATH}/usr/share/man
[ $MAKE_PATH -ne 0 ] && export MANPATH

MAKE_PATH=0
add_pkcfg_path()
{
    if [ -d "$1" ] ; then
        [ -z "$PKG_CONFIG_PATH" ] &&
            PKG_CONFIG_PATH="$1" ||
PKG_CONFIG_PATH="$1":$PKG_CONFIG_PATH
        MAKE_PATH=1
    fi
}
add_pkcfg_path ${WORK_PATH}/lib/pkgconfig
add_pkcfg_path ${WORK_PATH}/usr/lib/pkgconfig
[ $MAKE_PATH -ne 0 ] && export PKG_CONFIG_PATH
```

Support for RTL8723DE WiFi Module

I have Slackware 14.2 on my HP laptop, which has an RTL8723DE wifi module hardware. This module is still not supported in the mainstream kernel, but the driver is available at https://github.com/lwfinger/rtlwifi_new.git.

- get source, go extended

```
$ git clone https://github.com/lwfinger/rtlwifi_new.git
$ cd rtlwifi_new
$ git checkout extended
```

- to install (as root)

```
$ make install
$ modprobe rtl8723de
```

- to uninstall module (e.g. for kernel update)

```
$ rmmod rtl8723de
$ make uninstall
```

- to get better signal strength, create file '/etc/modprobe.d/rtl8723de.conf' with:

```
options rtl8723de ant_sel=2
```

Update20200824 That repo is no longer available. Using [rtw88](#) instead (module is now named rtw88_8723de). Just to note that the rtw88 code is now in mainline kernel starting 5.2

KVM Stuff

Somehow, /dev/kvm is missing?? So, just create this udev rule:

[65-kvm.rules](#)

```
KERNEL=="kvm", GROUP="users", MODE="0660"
KERNEL=="vhost-net", GROUP="users", MODE="0660"
```

Text-manipulation

I mostly use sed / grep / cut for this. But I have found the need to use tr at times.

For example, to filter out non-ASCII characters

```
tr -cd '\000-\177'
```

Removing Idlinux.sys

When installing using extlinux, the file Idlinux.sys is created and cannot be removed even by root! The command chattr can be used to remove the flag that protects that particular file

```
# chattr -i ldlinux.sys
```

To check/confirm, run

```
# lsattr ldlinux.sys
```

After that, the file can be removed by root as usual.

GTK3 Stuff

Running my GUI code compiled with GTK3 on Artix causes a runtime warning message

```
dbind-WARNING **: 11:12:11.208: Couldn't register with accessibility bus:  
Did not receive a reply. Possible causes include: the remote application did  
not send a reply, the message bus security policy blocked the reply, the  
reply timeout expired, or the network connection was broken.
```

I found this simple fix somewhere in a forum - simply create an environment variable

```
export NO_AT_BRIDGE=1
```

To check screen dpi

```
$ xdpyinfo  
$ xrdb -query
```

Optimize hard disk using hdparm

Note to self: CHECK THIS OUT! For example, on T23 laptop

```
$ hdparm -q -d1 -c3 -W1 -u1 -m 16 /dev/hda  
$ hdparm -q -d1 -c3 -X66 /dev/hdc
```

Adding fonts in Linux

1. xorg.conf, "mkfontscale", "mkfontdir", "ttmkfdir" => old school, nobody needs them?
2. /etc/fonts/*, ~/.fonts.conf, fc-cache, fc-list => the way to go

To strip binary/library files

```
$ strip --strip-debug /lib/*
$ strip --strip-unneeded /{,s}bin/*
```

Create a patch file

```
$ diff -Naur [file1] [file2] > file.patch
**Note**
%%{{{@ -X,Y +J,K @@}}}% is a hunk where diff is
 X & J - starting line number
 Y & K - line counts
```

Script to create multiple users

create_users

```
#!/bin/bash

USERLIST="$1"

[ ! -f "$USERLIST" ] &&
echo "Cannot access user list '$USERLIST'! Abort!" && exit 1

# assume format is user:pass
for userpass in `cat $USERLIST` ; do
    user=`echo $userpass | sed 's/^\([:^]\):.*\1/'`
    pass=`echo $userpass | sed 's/.*:\(\.*\)/\1/'`
    what=`echo $userpass | sed 's/^#\(\#[^#]*\)\1/'`
    [ "$what" == "#" ] && continue
    what=`cat /etc/passwd | grep "$user"`
    [ "$what" != "" ] && continue
    echo "User:[$user],Pass:[$pass]"
    echo "${user}:{$pass}:100:User Account:/home/${user}:/bin/bash"
done
```

NTP (System Date/Time)

By default, date/time are set manually: e.g. as root,

```
ntpdate pool.ntp.org
```

If ntpdate is not available (being deprecated?), use ntp daemon:

```
ntp -gq
```

(May have to stop running ntp daemon)

For Slackware, a startup script for NTP daemon is available /etc/rc.d/rc.ntpd, but do not forget to modify /etc/ntp.conf and uncomment (or add custom) NTP server information. To query, use

```
ntpq -p
```

To still use ntpdate while NTP daemon is running, use the -d switch,

```
ntpdate -d pool.ntp.org
```

Useful /proc Interface

To display active partitions,

```
cat /proc/partitions
```

To display cpu information,

```
cat /proc/cpuinfo
```

To check for Intel-VT virtualization features,

```
grep --color vmx /proc/cpuinfo
```

To check for AMD-V virtualization features,

```
grep --color svm /proc/cpuinfo
```

Note: Some BIOS may disable this feature by default. Also, old Linux kernel may not support it either.

To display memory information,

```
cat /proc/meminfo
```

To check shared libraries used by a program

```
cat /proc/<proc_id>/maps
```

Playing with LDAP

I want to have LDAP-based central authentication for my two new Slackware machines used for my

GMC project. Things to do:

1. get LDAP server installed and running
 - Slackware only have openldap-client by default
 - I'm referring to [here](#) to do this
2. figure out how to setup the client side
 - I may need nss_ldap only - will check into this later

SSH X Forwarding

On server, edit /etc/ssh/sshd_config

```
AllowTcpForwarding yes
X11Forwarding yes
X11DisplayOffset 10
X11UseLocalhost yes
```

On client, edit /etc/ssh/ssh_config

```
ForwardAgent yes
ForwardX11 yes
```

To connect,

```
# allow remote access for display
ssh -Y user@host
export DISPLAY=localhost:10.0
# then, run any x-program
```

Display Access Control

Only the user on the main console gets control of display protocol. If we do su, the root user cannot use any GUI.

To disable and enable access control

```
$ xhost +
$ xhost -
```

Find and set DISPLAY value accordingly

```
$ echo $DISPLAY
$ export DISPLAY=:0.0
```

Remote VNC (Remote Desktop)

This enables us to have GUI access on a remote machine. I use screen because I want to keep the running terminal 'alive'.

Use virtual terminal, run screen on the remote machine

```
$ screen -S VNC
```

Start VNC server on a display port

```
$ vncserver :23
```

Note `~/.vnc/xstartup` will be executed

To stop the VNC server

```
$ vncserver -kill :23
```

Detach the virtual terminal by hitting `<CTRL+A>-<D>`. To resume the virtual terminal from any console

```
$ screen -r VNC
```

To tunnel through SSH, start an ssh session (with port forwarding [5900+DP] and going background)

```
ssh -L 5923:localhost:5923 -N -f user@remotehost
```

Connect as if the server is on localhost

```
vncviewer localhost:23
```

Note: Desktop response becomes too slow for me! Not using this..

Console Multi-Tasking

On Linux, multi-tasking is also available on console. To suspend a task, hit `ctrl+z`

To list all jobs **duh!**

```
$ jobs
```

To resume

```
$ fg %<num>
```

VirtualBox Stuffs

Rebuild VirtualBox kernel module

After a kernel upgrade... do a

```
# /etc/init.d/vboxdrv setup
```

Virtual Serial Port

We can actually use minicom to connect...

```
minicom -D unix\#/tmp/xxx
```

where /tmp/xxx is the host pipe



Convert Image to Disk

```
VBoxManage convertfromraw -format VDI <file.img> <file.vdi>
```

Compact Disk Image

```
VBoxManage modifymedium --compact <file.vdi>
```

Create Disk Image

```
VBoxManage createmedium disk --filename <file.vdi> --size <megabytes>
```

Video4Linux Stuffs

```
v4l2-ctl --list-devices
v4l2-ctl -d 0 --list-formats-ext
v4l2-ctl -d 0 --list-ctrls

v4l2-ctl --device=0 --set-ctrl=?

v4l2-ctl -d 0 --set-fmt-video=width=1920,height=1080,pixelformat=YUYV
```

Mounting FreeBSD Partition (Slice)

Note At the moment, only read-only access (no write permission)!

```
# mount -t ufs -o ufstype=ufs2 /dev/<partition> <mount-path>
```

Backing up dokuwiki pages on github

[dokuwiki2github.sh](#)

```
#!/bin/bash

DOKUWIKI=${DOKUWIKI:="$HOME/public_html/dokuwiki"}
WIKIPAGE=${WIKIPAGE:="$DOKUWIKI/data/pages"}

cd ${WIKIPAGE} && git add . && git add -u &&
git commit -a -m "Content update `date +'%H:%M %d/%m/%Y %Z'`" &&
git push origin master

# setup a cron job for hourly update:
#
# 0 * * * * dokuwiki2github.sh
```

Linux kernel patch for CH341 USB2Serial

Tested this on Slackware 14.2.

Using patch found [here](#), I modified it a bit for latest Linux kernel used in Slackware 14.2 (4.4.190)

To create this patch, I copied out ch341.c from kernel source to my user path. Make another copy as ch341_patched.c. Modify as required. Do,

```
$ diff -u ch341.c ch341_patched.c > linux_4.4.190_ch341.patch
```

I do not know if this last step is needed, but I modified the patch so that the first two lines have the same file name.

[linux_4.4.190_ch341.patch](#)

```
--- ch341.c    2019-10-24 15:02:10.039646991 +0800
+++ ch341.c    2019-10-24 15:06:14.244416258 +0800
@@ -358,6 +358,7 @@
     struct ch341_private *priv = usb_get_serial_port_data(port);
     unsigned baud_rate;
     unsigned long flags;
+    unsigned int par_flags;

     baud_rate = tty_get_baud_rate(tty);

@@ -371,6 +372,30 @@
     * (cflag & PARENB) : parity {NONE, EVEN, ODD}
     * (cflag & CST0PB) : stop bits [1, 2]
```

```

*/
+ /* CH340 doesn't appear to support variable stop bits or data bits */
+ if (C_PARENB(tty)) {
+     if (C_PARODD(tty)) {
+         if (tty->termios.c_cflag & CMSPAR) {
+             dev_dbg(&port->dev, "parity = mark\n");
+             par_flags = 0xeb;
+         } else {
+             dev_dbg(&port->dev, "parity = odd\n");
+             par_flags = 0xcb;
+         }
+     } else {
+         if (tty->termios.c_cflag & CMSPAR) {
+             dev_dbg(&port->dev, "parity = space\n");
+             par_flags = 0xfb;
+         } else {
+             dev_dbg(&port->dev, "parity = even\n");
+             par_flags = 0xdb;
+         }
+     }
+ } else {
+     dev_dbg(&port->dev, "parity = none\n");
+ par_flags = 0xc3;
+ }
+ ch341_control_out(port->serial->dev, 0x9a, 0x2518, par_flags);

    spin_lock_irqsave(&priv->lock, flags);
    if (C_BAUD(tty) == B0)

```

Steps (as root) in terminal:

1. get to location

```
$ cd /usr/src/linux/drivers/usb/serial
```

2. patch ch341.c

```
$ patch < /path/to/linux_4.4.190_ch341.patch
```

- path/to is the path where you save the above patch

3. get to top kernel source path

```
$ cd /usr/src/linux
```

4. compile module

```
$ make M=drivers/usb/serial modules
```

5. (OPTIONAL) remove previously loaded module

```
$ rmmod ch341
```

6. 'install' the module

```
$ cp drivers/usb/serial/ch341.ko /lib/modules/`uname -r`/kernel/drivers/usb/serial/
```

2023/08/29 13:04

Linux Shell

Useful command-line stuff.

Raw Disk Data

Using dd

```
$ dd if=ifile of=ofile skip=offs count=size status=progress
```

where skip and count values are in sector counts.

[dd_size.txt](#)

```
- note on byte-size & sector size (for 512-bytes sector size)
01M:    1048576byte @ 000002048sect
01G: 1073741824      @ 002097152
10G: 10737418240     @ 020971520
50G: 53687091200     @ 104857600
80G: 85899345920     @ 167772160
```

File Management

Using find... dumped

```
$ find . -type d -empty -print
$ find . -type d -empty -delete

$ find -type f \(-name "*zip" -o -name "*tar" -o -name "*gz"\) -size +1M -delete
$ find -type f \(-name "*zip" -o -name "*tar" -o -name "*gz"\) -size +1M -exec rm {} +
```

2025/01/21 07:40 · azman

Devuan Installation

My Devuan installation notes.

Latest Devuan Installation

202204100749

Using Devuan Chimaera DVD ISO (dd onto USB) → UEFI boot!

Note: If using netinstall or CD1 some firmware may not be available - which is a problem if using WiFi module.

The must-haves for MY Devuan installation:

- install build-essential (development tools)
- install linux-headers-amd64 (to compile kernel modules)
- install geany, git, gitk (coding stuffs)
- install libnss-mdns & avahi-daemon (system management) **INSTALLED BY DEFAULT**
- install curl (some of my scripts needs that...)

The stuffs I currently need:

- install cryptsetup (encrypted partitions)
- install xfig (i need xfig2dev)

Note: these are required to compile Linux kernel: *libncurses5-dev build-essential libssl-dev libelf-dev git bison flex* (need to check if these are already installed by the above selection, e.g. build-essential and git are covered)

work in progress...?

Last XFCE Installation

- using default xfce desktop
 - this is my primary choice, but I have installed cinnamon on other machines
- replace wicd with network-manager
 - # apt install network-manager network-manager-gnome
 - *network-manager-gnome* provides nm-applet (not needed if using nmcli)
 - # apt purge wicd wicd-gtk
 - # apt autoremove

Minimal Desktop Installation

I want a basic installation with dwm.

- install base (use netinstall iso... or, maybe use debootstrap?)
- install my usual stuffs

```
# apt install build-essential linux-headers-amd64 vim git curl
```

- install xorg stuffs

```
# apt install xorg libx11-dev libxft-dev libxinerama-dev
```

- install suckless stuffs

```
# apt install stterm suckless-tools
```

- install dwm from source
 - use mylubuild script
- looks good, but i need acpi stuffs

```
# apt install acpid
```

- for laptop, maybe add acpi-support
- if need command-line utility, add acpi

work in progress...

Devuan upgrade (chimaera to daedalus)

Personal note - basically from devuan.org.

[devuan_upgrade.txt](#)

```
upgrade chimaera to daedalus (from devuan.org)

- update/upgrade chimaera
# apt update
# apt upgrade

- update apt sources.list
# sed -i 's/chimaera/daedalus/g' /etc/apt/sources.list

- update pkg list from daedalus
# apt update

- kill screensaver (if running)
# killall xscreensaver

- upgrade/dist-upgrade
# apt upgrade
# apt dist-upgrade

- in case of failures, fix and rerun
```

```
# apt -f install
# apt dist-upgrade

- cleanup
# apt autoremove --purge
# apt autoclean

deb http://deb.devuan.org/merged daedalus main non-free-firmware non-
free contrib
deb http://deb.devuan.org/merged daedalus-security main non-free-
firmware non-free contrib
deb http://deb.devuan.org/merged daedalus-updates main non-free-
firmware non-free contrib

#deb-src http://deb.devuan.org/merged daedalus main
#deb-src http://deb.devuan.org/merged daedalus-security main
#deb-src http://deb.devuan.org/merged daedalus-updates main
```

Install Using debootstrap

- boot using my1live-devuan
- need to install gisk debootstrap
 - optionally, install lvm2
- run

```
# apt install gisk debootstrap lvm2
```

- prepare disk (/dev/sda) layout
 - 1 uefi partition (ef00)
 - 1 root partition (8300)
 - 1 home partition (8e00/8300)
 - 1 swap partition (8200)
- run

```
# gdisk /dev/sda
```

- format/mount root partition

```
# mkfs.ext4 -L MY1B00T /dev/sda2
# mount /dev/sda2 /mnt/disk
```

- run debootstrap

```
# debootstrap chimaera /mnt/disk http://deb.devuan.org/merged/
```

- while debootstrap runs, format other partitions

```
# mkdosfs -n MY1UEFI /dev/sda1
# mkfs.ext4 -L MY1B00T /dev/sda2
# mkswap -L MY1SWAP /dev/sda4
```

- will use lvm in this example

```
# pvcreate /dev/sda3
# vgcreate homevg /dev/sda3
# lvcreate -l +100%FREE -n home0 homevg
# mkfs.ext4 -L MY1HOME /dev/homevg/home0
```

- mount efi/home partitions

```
# mkdir -p /mnt/disk/boot/efi
# mount /dev/sda1 /mnt/disk/boot/efi
# mount /dev/homevg/home0 /mnt/disk/home
```

- chroot and install/setup

```
# chroot /mnt/disk
# apt update
# apt install linux-image-amd64 build-essential linux-headers-amd64 vim
git lvm2
# apt install firmware-linux firmware-iwlwifi firmware-atheros
firmware-realtek
# apt install cinnamon-desktop-environment
```

- update initramfs (add lvm support)

```
# update-initramfs -u -k all
```

- i prefer all-lowercase path names

```
# vi /etc/xdg/user-dirs.defaults
```

- setup locale

```
# apt install locales
# echo "en_US.UTF-8 UTF-8 >>/etc/locale.gen"
# locale-gen
```

- setup grub

```
# apt install grub-efi-amd64
# grub-install /dev/sda
# update-grub
```

- run efibootmgr to make sure

```
# efibootmgr
```

- edit fstab

```
# vi /etc/fstab
```

- setup root password

```
# passwd
```

- change hostname

```
# vi /etc/hostname
```

- reboot

Note: I got to Cinnamon Desktop and everything looks ok - BUT, I simply cannot run gnome-terminal! Well, I can if i run dbus-update-activation-environment first. So, I missed something coz when I reinstalled using the full dvd, everything works fine.

work in progress...

Devuan Setups

These depends on my need when using that particular machine.

- using virtualbox from oracle (just like my slackware setup)
- using texlive ([install using tlmgr](#))
 - alternatively, install texlive texlive-latex-extra texlive-science
- install freecad kicad openscad (project stuffs)
- install ntp ntpdate (system management)

Development

- getting OpenGL stuffs (glut): # apt install freeglut3-dev
- getting sqlite stuffs: # apt install sqlite3 libsqlite3-dev
- getting glade (will also get gtk library): # apt install glade
- getting wxwidgets stuffs: # apt install lib-wxgtk3.0dev
- getting mylimgpro stuffs: # apt install libavcodec-dev libavdevice-dev libavformat-dev libswscale-dev
- to compile sdcc: # apt install bison flex libboost-dev texinfo

note: [Setting up mingw-w64 cross-compiler](#)

Web Server

- Install webserver
 - # apt install apache2
 - default path for web is /var/www/html
 - edit /etc/apache2/apache2.conf

- add *ServerName* (remove annoying startup message!)
- my *my1apisrv* code need these for *www* dir config

```
Options FollowSymLinks
AllowOverride All
Require all granted
```

- Install php
 - # apt install php php-cgi libapache2-mod-php php-mysql php-sqlite3
 - edit /etc/php/7.0/apache2/php.ini to enable pdo support
- Create required links in mods-enabled and conf-enabled
 - my *my1apisrv* code need rewrite
 - both folders are in /etc/apache2
- Install database
 - # apt install mariadb-server
- If running dokuwiki
 - # apt install php-xml
- My API client php code needs this
 - # apt install php-curl

Multi-Arch (a.k.a. Multi-Lib)

To run 32-bit binary:

- Enable multi-arch: # dpkg --add-architecture i386
- Update package list: # apt update
- Most probably need libc: # apt install libc6:i386
- Install required libraries: (<package>:i386)

To build 32-bit binary:

- Install compiler(s): # apt install gcc-multilib g++-multilib
 - Notice that these are 64-bit packages (no :i386 suffix) → cross compilers!
- Use -m32 gcc option to compile!

Gaming

To play steam games:

- enable multi-arch
- install steam: # apt install steam (binary in /usr/games/steam)

NFS setup

Dumping this as it is for now:

- client

- install

```
# apt install nfs-common
```

- mount

```
# mount -t nfs <host>:/path <mount-point>
```

- server

- install

```
# apt install nfs-kernel-server
```

- [optional]

```
# mount --bind /path/to/share /mount/point
```

- edit /etc/exports

- start

```
service nfs-kernel-server start
```

Others

My iso2boot script need *isohybrid* from syslinux/isolinux project:

- # apt install syslinux-utils

Use Tips

Some are applicable to any APT-based distro.

Note for APT-based distro: To remove translations, create file */etc/apt/apt.conf.d/99translation* and insert *Acquire::Languages { "none"; };*

Note: look into *unattended-upgrades* (simply *apt install* and run *dpkg-reconfigure -priority=low unattended-upgrades???*)

List Installed Packages

Using apt tool:

```
apt list --installed 2>/dev/null | grep installed
```

Note that apt will issue a warning when piping its output in shell. Hence, the need to redirect stderr to /dev/null. We can further grep away the packages that were automatically installed.

Using the basic dpkg tool:

```
dpkg --get-selections | sed -n 's/^\\(([^\t]*\\)\t.*$\\1/ p'
```

This version, however, only provides package name. To extract similar output from apt (assuming output was redirected into a file called temp.txt), run

```
cat temp.txt | sed -n 's|^(.*)/.*$|\\1| p'
```

Adding More Repo

Get proper signature key from that source (*.asc file) and add to system using apt-key.

```
# cat <key-file.asc> | apt-key add -
```

Create a listing file for source URL in /etc/apt/sources.list.d/.

```
# echo "deb [arch=?] <new-repo-url> <version> main" >
/etc/apt/sources.list.d/new-repo.list
```

After doing an apt update, should be able to apt install <pkg>...

Reconfigure Package

Basically, run a dpkg-reconfigure <pkg>

e.g. To change timezone

```
# dpkg-reconfigure tzdata
```

Upgrading APT-based systems

To do an upgrade:

- Modify /etc/apt/sources.list and replace the release codenames

```
# sed -i 's/old_release/new_release/g' /etc/apt/sources.list
```

- skip -i option for a dry-run

- Update package list

```
# apt update
```

- Upgrade distribution

```
# apt dist-upgrade
```

- To make sure EVERYTHING is upgraded (avoids old packages held back)

```
◦ # apt full-upgrade
```

- Do house cleaning

```
# apt autoremove
# apt clean
```

That should do it!

Search package for specific binary

- There is a specific tool for that

```
# apt install apt-file
```

- Update the package/file mapping database

```
# apt-file update
```

- Search for “top”

```
# apt-file search --regexp '/top$'
```

Avoiding marking package as manually installed

- Use dry-run (simulation) option -s

```
# apt install -s <pg1> ... <pkgN> 2>/dev/null|grep manually
```

- Removed the package(s) displayed from your install list and re-run without -s

GRUB Stuffs

To prevent GRUB from looking for other OS every time it is updated:

- edit /etc/default/grub and insert

```
GRUB_DISABLE_OS_PROBER=true
```

- run update-grub

To add a custom entry:

- add a menuentry in /etc/grub.d/40_custom
- uses the same menuentry format as in /boot/grub/grub.cfg

- e.g. to boot my Slackware partition I can insert

```
menuentry 'Slackware' --class slackware --class gnu-linux --class
gnu --class os $menuentry_id_option 'my1part-<uuid>' {
    insmod part_gpt
    insmod ext2
    set root='hd0,gpt4'
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,gpt4
--hint-efi=hd0,gpt4 --hint-baremetal=ahci0,gpt4 <uuid>
    else
        search --no-floppy --fs-uuid --set=root 4d95a526-2518-4fd6-
a904-f7bd2729145d
    fi
    linux /boot/vmlinuz-huge-4.4.240 root=/dev/sda4
}
```

- use initrd (after linux line) to specify an initrd/initramfs
- of course, <uuid> should be a valid filesystem uuid
- run update-grub

KVM Stuffs

If KVM group is missing, simply create one...

```
# [ -z "$(cat /etc/group|grep kvm)" ] && addgroup --gid 125 kvm
```

Issues

Issues... and maybe fixes (if any).

Image Magick's convert error

I found an error when using image magick's convert to create EPS from fig that provides this message:

```
convert-im6.q16: attempt to perform an operation not allowed by the security
policy `EPS' @error/constitute.c/IsCoderAuthorized/421
```

Solution:

- (as root) edit /etc/ImageMagick-6/policy.xml
- modify following line (change none → read|write)

```
<policy domain="coder" rights="none" pattern="EPS" />
```

GRUB Prompt

In some cases, after an installation is complete, the system boots to a GRUB prompt. The issue may be caused by a different disk assignment during installation. Things is not so bad since GRUB prompt is very shell-like (e.g. double-tab for completion):

To list available disks:

```
> ls
```

To list content of first partition on the first disk (/dev/sda1) that is using GPT:

```
> ls (hd0,gpt1)
```

So, to boot an installation on second partition instead:

```
> linux (hd0,gpt2)/boot/vmlinuz<...>
> initrd (hd0,gpt2)/boot/initrd.img<...>
> boot
```

BCM Wireless

Device:

```
# lspci -nn | grep BCM
02:00.0 Network controller [0280]: Broadcom Limited BCM43142 802.11b/g/n
[14e4:4365] (rev 01)
```

Install: broadcom-sta-dkms, firmware-linux, firmware-b43-installer (b43-fwcutter)

note: only solves the Wireless Driver part... NOT the bluetooth hardware!

```
# lsusb | grep BCM
Bus 002 Device 004: ID 0a5c:216d Broadcom Corp. BCM43142A0 Bluetooth 4.0
# dmesg | grep blue
[    8.850007] bluetooth hci0: firmware: failed to load
brcm/BCM43142A0-0a5c-216d.hcd (-2)
[    8.850197] bluetooth hci0: Direct firmware load for
brcm/BCM43142A0-0a5c-216d.hcd failed with error -2
```

So, download BCM43142A0-0a5c-216d.hcd in [here](#) and place it /lib/firmware/brcm/ (create that path if it does not exist).

Xfce Desktop

- tapping on my laptop touchpad is NOT working

- moving around & button clicks are working
- need to create file /etc/X11/xorg.conf.d/90-touchpad.conf

[90-touchpad.conf](#)

```
Section "InputClass"
    Identifier "libinput touchpad catchall"
    MatchIsTouchpad "on"
    MatchDevicePath "/dev/input/event*"
    Driver "libinput"
    Option "Tapping" "on"
EndSection
```

- lid event does not trigger suspend (but manual suspend works)
 - this is a systemd-related aftermath: xfce power manager allows/assumes login to handle lid
 - fix:

```
$ xfconf-query -c xfce4-power-manager -p /xfce4-power-
manager/logind-handle-lid-switch -s false
```

- issues when logging out/shutdown
 - may be caused by intel graphics library? i915?
 - can install lightdm - but shutdown/restart always gets login page (=logout)
- wicd feature is an issue for system with multiple users
 - shared wifi password, no option to make private

Cinnamon DE

- login page (lightdm?) - cannot shutdown/reboot! read [here...](#)
 - found a fix [here](#)
 - i just needed to edit /etc/pam.d/lightdm-greeter and changed pam_systemd.so to pam_elogind.so (but, may cause issues with suspend/hibernate?)
- laptop battery quickly drain below 30 percent
 - setup higher critical value for power management

```
gsettings list-keys org.cinnamon.settings-daemon.plugins.power
gsettings set org.cinnamon.settings-daemon.plugins.power use-time-
for-policy false
gsettings set org.cinnamon.settings-daemon.plugins.power
percentage-low 30
gsettings set org.cinnamon.settings-daemon.plugins.power
percentage-critical 25
gsettings set org.cinnamon.settings-daemon.plugins.power
percentage-action 23
```

- or use dconf-editor

PulseAudio

Youtube videos keep resetting the volume settings to 100%! The culprit is `flat-volumes`. Modify `/etc/pulse/daemon.conf` and set `flat-volumes=no`.

(Re)-Compiling Syslinux

I need to recompile syslinux - just to remind myself, other than the usual development packages (build-essential, etc), I also need nasm and upx-ucl.

```
apt install nasm and upx-ucl
```

Unwanted Background Program

I found this in my old notes... I somehow found an autostart program `geoclue-demo-agent.desktop`, which I obviously do not need. So, simply remove that from autostart path

```
# rm /etc/xdg/autostart/geoclue-demo-agent.desktop
```

Sound muted on startup on my HP laptop

trying two options found... so far, not successful? keeping this here for reference.

1. option 1:
 - install alsa-utils
 - make sure sound is not muted and set to desired volume level
 - run (as root) alsactl store
2. option 2:
 - edit (as root) /etc/pulse/defaults.pa
 - can also copy this to home folder ??? for 1-user solution...
 - comment out 'load-module module-device-restore'

General Issues

- wifi firmware missing after first restart
 - manually install `firmware-ralink`
- network manager cannot connect using wifi
 - edit `/etc/NetworkManager/NetworkManager.conf`
 - add

```
[device]
wifi.scan-rand-mac-address=no
```

Using Wine on Devuan

Setup Wine (as root):

- enable multiarch
- install wine and winetricks

```
# apt install wine wine32 winetricks
```

Configure Wine (as user):

- if required, remove previous settings

```
$ rm -rf $HOME/.wine
```

- config for win32

```
$ WINEARCH=win32 winecfg
```

- setup nice fonts

```
$ winetricks corefonts
```

Note: Install fuseiso to enable mounting ISO as user

Game: Red Alert 2

Setup Installer ISO:

- create link to ISO as CDROM device

```
$ ln -sf /path/to/install.iso $HOME/.wine/dosdevices/d::
```

- mount ISO

```
$ fuseiso mount /path/to/install.iso $HOME/.wine/drive_d
```

- create link to mount path as CDROM drive

```
$ ln -sf $HOME/.wine/drive_d $HOME/.wine/dosdevices/d:
```

Run installer

- \$ wine D:\\Setup.exe

Fixes:

- menu does not show (solution available at [WineHQ](#))
 - download ddraw.dll from <https://github.com/CnCNet/ts-ddraw/releases>
 - place in RA2 install path
 - run winecfg add that to library (as window native, instead of builtin)

Application: LTSpice

- download windows version from ltspice website and run

```
$ wine /download/path/LTspiceXVII.exe
```

Devuan on Raspberry Pi

Check out [here](#).

RasPi400

Using rpi-devuan-chimaera-5.10.110-ext4-2022-04-16.zip

- boots ok (default hostname=bcm2711)
- login (root:toor)
- setup

```
# run-setup
```

- change hostname (edit /etc/hostname and /etc/hosts)
- reboot
- remove default user

```
# userdel -r devuan
```

- most development stuffs i want are preinstalled! yay!
- install vim

```
# apt install vim
```

- remove nano

```
# apt remove nano
```

- also

```
# rm .nanorc
```

- install xorg stuffs

```
# apt install xorg libx11-dev libxft-dev libxinerama-dev
```

- install suckless stuffs

```
# apt install stterm suckless-tools
```

- create user and login as that user
- get my1shell and my1ubuild

```
# git clone https://codeberg.org/azman/my1shell
# git clone https://codeberg.org/azman/my1ubuild
```

- install dwm from source
- install browser & font

```
# apt install surf fonts-liberation2
```

work in progress... 2 issues: {surf cannot validate cert} {reboot hangs}

Note: Will simply use Raspberry Pi OS for now...

2023/08/29 12:54

From:
<http://azman.unimap.edu.my/dokuwiki/> - Azman @UniMAP



Permanent link:
<http://azman.unimap.edu.my/dokuwiki/doku.php?id=linux:linux&rev=1739508448>

Last update: **2025/02/14 12:47**