AV LINUX 2017.3

USER MANUAL

by Glen MacArthur

BOOKMARKS :

AV LINUX 2017 – A FRESH START! :

What AV Linux 2017 is, and what it ain't...:

This **'2017'** version of AV Linux is *completely new* from the ground up, it represents numerous significant changes and differences from previous versions briefly detailed in these notes. Essentially AV Linux is a downloadable and installable shared snapshot ISO image based on Debian/GNU Linux pre-configured to facilitate use as an Audio and Video production workstation OS. It is really intended to just be that simple, an AV production system prepared and shared by a single user with some practical experience in setting such things up. It is NOT intended to be a bona-fide full and proper "Linux Distribution", as its sole maintainer I have *no* special training or *any* credentials as a Linux System Administrator and am merely doing this in my spare time as a hobby. The implications of this are that **AV Linux is provided 'as-is' with absolutely NO guarantee of any kind to function on any given computer, or any guarantee of regular, sustained or future development .**

Now hold on a second, that sounds pretty bleak doesn't it…? Bear in mind that you are reading this statement in a complete User Manual and can voice your concerns in an <u>Online User Forum</u>, so please let's just mutually understand that I *like* doing this, I *want* to do this, but 'real life' may intervene and I may occasionally need to keep my focus elsewhere and therefore I can't in good conscience make any *promises* about the permanence of AV Linux. OK, phew…I feel better now… are we good?

Components of AV Linux and differences from stock Debian/GNU Linux :

- ➢ Originates from 32bit and 64bit snapshots of Debian Testing… **PLUS**
- ➢ Trulan Martin's Custom Realtime Preempt Kernel.
- ➢ CPU governed for performance by default.
- ➢ Optional choices of Liquorix or Debian Kernels.
- ➢ Some special packaging not found in default Repositories.
- ➢ Full XFCE4 Desktop Environment with attractive customizations.
- ➢ falkTX's KXStudio Repositories for up to date Audio Applications.
- ➢ Extensive Audio/Video and Administrator-friendly Thunar Custom Actions.
- ➢ Complete JACK Audio environment with automatic JACK MIDI and ALSA Loopback.
- ➢ Robust Environment for developers: Compilers, Package scripts, GIT, SVN, BZR, + dev libs.
- ➢ Choice of GCC5 or GCC6 Compilers.
- ➢ Mozilla builds of Firefox and Thunderbird.
- ➢ An abundance of various GTK and Window Manager Themes.
- ➢ Customized scripts for convenience and tons of other stuff I probably have forgotten..

ISO INFO :

ISO Size Info :

 Why is it so big!? *<insert obligatory joke here>*. I often get asked why AV Linux ISO's are significantly larger than *<insert obligatory Distro name here>*. Here are a few reasons:

- ➢ My priority is functionality and attention to detail over size for it's own sake.
- ➢ AV Linux uses a more relaxed ratio for it's compressed Live filesystem than mainline Distros.
- ➢ 'Bundled' apps that are self-contained are much larger than apps that use the system libraries.
- ➢ Two versions of 'Wine' are required, one for Audio Plugins and one for Pipelight.
- ➢ AV Linux ships with a full development environment for developers and experienced users.
- ➢ Hundreds of hidden Audio Plugins are included, many are fairly large in size.
- ➢ None of the extra 'bulk' in AV Linux affects performance, it is still light and efficient to use.

Straight Talk on 32bit+PAE and 64bit :

 64bit has replaced 32bit as the defacto Desktop computer architecture, some other OS' now only support 64bit, for this reason and in response to user requests AV Linux now comes in both 32bit and 64bit architectures. If you have a reasonably new 64bit CPU computer with more than 4Gb of RAM then your first choice will be to use the 64bit version of AV Linux. That said... one big misconception that plagued the previous versions of AV Linux is that 32bit is *always* inferior and has no use-case with 64bit CPU equipped computers and this is not accurate. There are of course valid technical 'whitepaper' reasons that 64bit outperforms 32bit, however in the real world for users who aren't pushing the limits of their system memory then the differences between 32bit+PAE and 64bit are virtually invisible. For instance 64bit is unlikely to run more Audio tracks or Render a Video file any faster than 32bit+PAE on the same hardware, those are factors related to CPU clock speed and HDD performance, not necessarily the CPU architecture and memory handling capacity. The 32bit version of AV Linux has a kernel with Physical Address Extentions (PAE) which allow it to use up to 64Gb of RAM, the bottleneck with this is that any one running process can only access 4Gb of RAM, where misunderstanding comes into play is that a small percentage of Users will have a workflow under normal conditions that will require a single process to access more than 4Gb of RAM. Two possible scenarios where this *can* happen are using very large audio sample libraries (ie GIG, SFZ, Drumgizmo) that cache the samples to RAM and don't use disk streaming, assuming your system has 4Gb of RAM if the combined size of your running programs and the samples cached in RAM exceed 4Gb then the system will become unstable and be forced to run in it's 'swap'. Other common scenarios are if you are running multiple Virtual Machines (ie in 'VirtualBox') or doing heavy duty 3D Compositing in 'Blender', in cases like this you will not want accessible system RAM to be a limiting factor. On the other hand if you are doing most common types of Audio and Video work then you don't necessarily *have to* use the 64bit version of AV Linux on a 64bit computer. As an example if you want to use 'ArdourVST' (Ardour built to use Windows VST Plugins) then it is only available and possible to use on the 32bit+PAE ISO of AV Linux and can be used quite easily on a 64bit computer.

 The point here is not to confuse or promote one architecture over the other, simply to separate facts from fiction so you can make an informed choice of what is best for you..

Differences between the 32bit and 64bit ISOs :

  At a casual glance you are not likely to see much difference between the 32bit and 64bit versions of AV Linux, but there are a few differences noted here:

- ➢ The 32bit version includes an extra build of Ardour that supports Windows VST Audio Plugins.
- ➢ The 32bit version has the XFCE4 Panel postioned at the top and 64bit has it at the bottom.
- ➢ The 32bit version uses the 'Greybird' system theme and 64bit uses 'Zukitre'
- ➢ The 32bit version has no Chrome Browser because Google discontinued 32bit support.
- ➢ The 64bit version is 'multiarch' and may have some extra required 32bit packages included.
- ➢ The 64bit version has preliminary, incomplete Systemback support for UEFI Boot.
- ➢ *UEFI boot has NOT been tested, User Input Needed!

Verifying the ISO Download with MD5sums :

  When you download the AV Linux ISO you will see that in the server download folder is the actual '*isotester-avlXX-2016.X.iso*' file containing the operating system and an accompanying '.md5' text file. You should download the both the 'iso' and 'md5' files, you should be able to right-click and use 'Save target as' for the md5 file. **Before running AV Linux Live or installing it on your computer you should verify that the 'md5sum' of your ISO download matches the md5sum on the download server**. This will ensure that you have a complete and proper download of the ISO as well as confirming that you haven't downloaded a compromised ISO that has been maliciously tampered with.

  For verification we use the 'md5sum' Terminal command. As an example let's say you downloaded the 64bit ISO to the 'Downloads' folder in your user's home folder. Open a Terminal and 'change directory' to the 'Downloads' folder like this:

```
cd Downloads
```

Now run the md5sum command on the downloaded 64bit ISO file :

```
md5sum isotester-avl64-2017.3
```

  Of course if you've downloaded the 32bit version then change the filename to 'avl32' accordingly. The terminal will calculate the md5 'hash' and then print it in the terminal. It will be a long string of numbers and letters and show the ISO filename at the end. If the hash generated by the terminal matches the hash in the md5 text file you downloaded you are good to go.

RUNNING AV LINUX "LIVE"

AV Linux can be used "Live" in 2 different ways with no changes to the host machine operating it, by running it from a LiveDVD or using a Bootable USB Key.

LiveDVD:

- ➢ Download the AV Linux ISO file from http://www.bandshed.net/avlinux
- ➢ Burn the ISO file to a DVD-R or RW using your existing Burning Software.
- ➢ Set your Computer BIOS to Boot from your DVD Drive if it doesn't by default.
- ➢ Reboot into the AV Linux LiveDVD.

METHOD 1 - Bootable USB Key:

Requirements:

An existing AV Linux install OR a running AV Linux LiveISO session
PC Capable of booting from USB Key with block sizes greater than 1024mb
At least 4GB USB Key
AV Linux ISO image from here: http://www.bandshed.net/avlinux

Bootable USB Keys can be created with the Bootable USB Disk Tool found in the AV Linux Assistant. This unique tool creates an 'ISOhybrid' USB Key. Please note that this tool overwrites any existing USB FAT32 or other filesystem and writes the actual ISOhybrid image itself to the key. This means the Key will no longer be usable for Data storage, however you can use the USB Key Tool to overwrite the ISOhybrid image as many times as you want. This USB Key Tool will only work with ISOhybrid images. The AV Linux ISO comes ready to use as a formatted ISOhybrid image.

METHOD 2 - Unetbootin Method:

Requirements:

PC Capable of booting from USB Key
At least 4GB USB Key
Latest AV Linux ISO image from here: http://www.bandshed.net/avlinux

This guide is written assuming you are using an existing AV Linux install, however any current distro with GParted and Unetbootin will work. This operation will utilize the entire USB Key so use a key you don't want to share data on other computers with and remove any important data from the key. I would also suggest you use a clean key without any U3 or other proprietary launching software on it, most manufacturers provide utilities to remove this.

STEP 1.
Insert your USB Key into the USB Port but don't mount it (or unmount it) in the File Manager. From the menu go to 'Preferences'-->GParted and launch GParted.

STEP 2.
Using the drop down dialog in the upper right corner of GParted select your USB Key. MAKE SURE YOU KNOW WHICH DRIVE YOUR USB KEY IS!!

STEP 3. (DO WITH CAUTION)
Select any existing Partitions on the Key, Right Click on them and delete them until the Key is completely unallocated space.

STEP 4.
Create a new FAT32 partition on the USB Key equal to or slightly larger than the size of the AV Linux ISO. In the remaining unpartitioned space create an ext2 or ext3 partition and in the 'label' field enter 'live-rw'.

STEP 5.
Click on the checkmark icon at the top of GParted and apply the partitions you have created to the key.

STEP 6.
Once GParted has finished creating your new partitions right click on the FAT32 Partition and select 'Manage Flags' and from the list select 'boot'. When you close the Flags dialog Gparted should quickly rescan and display the boot flag on your FAT32 Partition. Close Gparted.

STEP 7.
Remove your USB Key and then plug it back in. Then open the File Manager, you should see 2 partitions available to mount relating to your USB Key you can mount them both but you only need to mount the FAT32 one. Hint... it is the one not called 'live-rw'.

STEP 8.
From the menu open 'System'--->Unetbootin. Select the Diskimage option and navigate to where you have downloaded or placed the ISO Image file and Click OK. Unetbootin will take several minutes to copy the image and bootloader to your Key.

**Adding Persistence to the Key:**

If you simply want AV Linux on a USB Key without persistent storage you can stop here. This will give you the partition AV Linux runs from and a small additional partition to store files on while you are running it. For persistence there are a few more steps and a few things to consider:

- Persistence will dramatically slow bootup time the first boot after enabling it.
- Persistence may cause the system response to be a little slower than a non-persistent Key.
- Installing AV Linux from a persistent Key has NOT been tested and may not work.
 - Do not remove your persistent key when prompted until it stops flashing.

STEP 9.
Mount the FAT 32 Partition of your Key (if it isn't already) in the File Manager and navigate to the 'syslinux.cfg' file. Right click on it and open it in Mousepad and modify its text in line 9 like the example below, once done save the edited syslinux.cfg file and you are ready to boot into your new Key. Enjoy!

```
append initrd=/ubninit boot=live config   threadirqs quiet splash persistent --
```
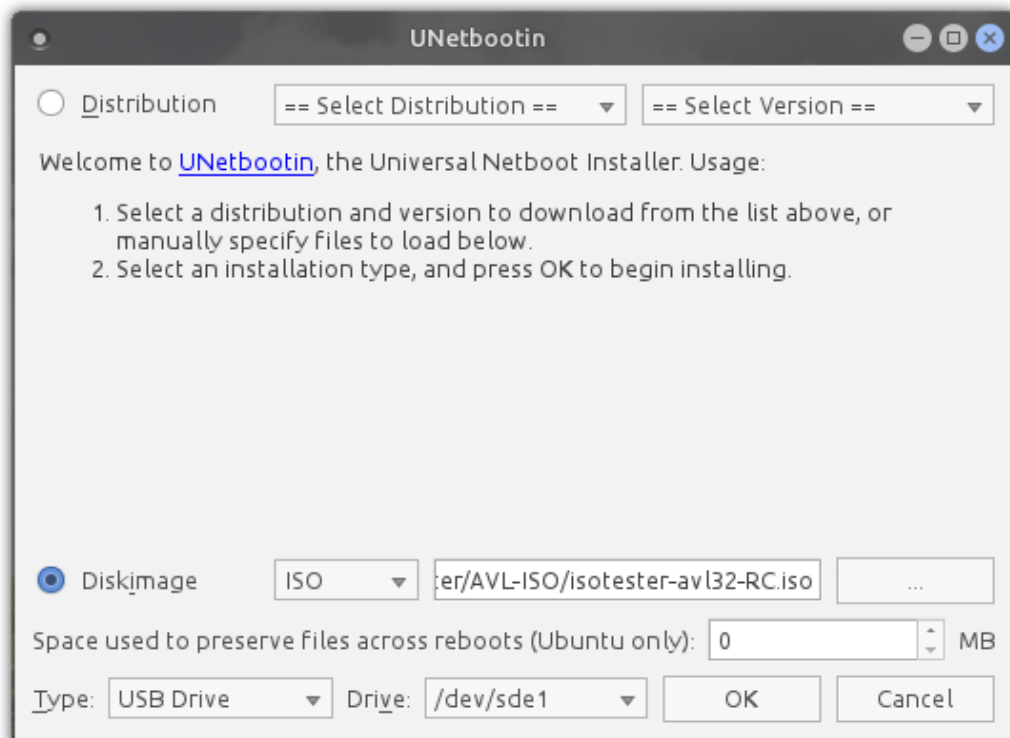
AV LINUX BASIC BOOTING BIBLE

When Booting an AV Linux ISO :

1. Be patient...This is a large ISO it can take several seconds to start the boot especially on LiveDVD media.

2. Watch for disk activity, if you see nothing immediately on the monitor but there is disk activity it is still working on the boot.

3. Closely observe the Kernel console text output for red text errors, warnings and even many regular console colored errors are normally harmless. AV Linux doesn't use a bootsplash is so the User can monitor what the Kernel is doing at boot.

4. The SliM Login window may take a few seconds to show up, possibly even longer when using a LiveDVD.

5. If the desktop partially loads OR the display resolution is wrong or corrupted it is most likely a Kernel Modesetting/Video Driver issue which is covered in more detail below.

Common Boot Failure Gremlins :

1. A corrupted or broken ISO image on the Project ftp server.
  This is extremely unlikely and rare and is easily ruled out by a single successful install by an end-user, AV Linux is pre-tested by a third party before any release is made public to make sure the ftp ISO images are intact and functional.

2. A bad download of the ISO image.
  This is not common, but quite possible and can be verified by using the accompanying ISO.md5 file from the ftp download folder. Another simple indication is if the file sizes on both the ftp and the downloaded file match perfectly.

3. A bad burn of the ISO to disk media.
  This is usually the most likely cause of boot failure and is generally evident when the disk does nothing at all or freezes after displaying a few lines of console text which may be accompanied by numerous console red error messages. Always burn any compressed filesystem media like ISO images at low burn speeds (i.e. 4X or less) and make sure to use good disk media. If a LiveDVD fails try it on a different computer, this is the number one way to discern if your media is bad or not or whether your computer has a specific hardware issue with the LiveDVD.

4. Incompatible Video Hardware and Kernel Mode Setting (KMS).
  This issue is generally evidenced by the Kernel completing its console output and dying at the login or unsuccessfully trying to load the Desktop and/or displaying the Desktop at a very low or garbled resolution. AV Linux 6.0.4 can be booted to disable KMS by selecting the 'Failsafe' option from the ISOlinux LiveDVD boot splash screen.

AV LINUX LiveISO LOGIN :

NEW! **Login Username and Password Required:**

AV Linux 2017 now requires a login to the LiveISO, previous versions used an auto-login. When the LiveISO finishes booting and proceeds into the SliM login screen you will need to enter the following Username and Password:

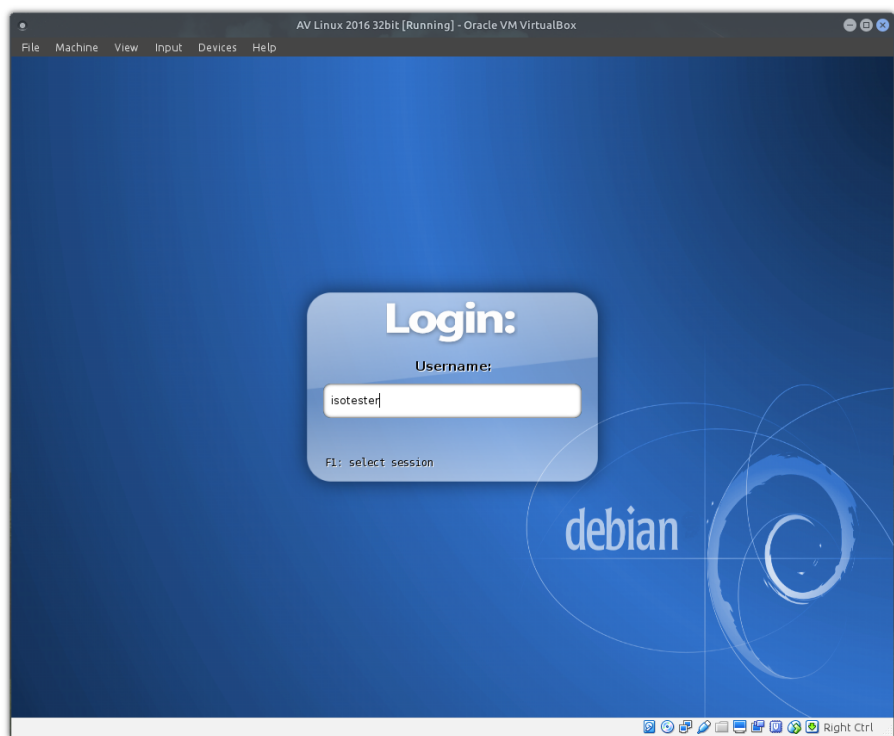**AV Linux 32bit :**

Username: isotester
Password: avl32
Root Password: avl32admin

AV Linux 64bit :

Username: isotester
Password: avl64
Root Password: avl64admin

LIMITATIONS OF LIVE SESSIONS :

Due to some changes in how AV Linux LiveISO's are built there are now some limitations when running AV Linux live from both DVDs and USB keys. The live session should be looked at as an opportunity to kick the tires of AV Linux, to see how it works, if it supports your PC hardware and to demo some of the great included applications. It is not really recommended to use AV Linux live for actual production for the following reasons:

➢ **The audio latency performance of Live sessions is noticeably higher than that of installs to HDD.** The cause for this is unknown but it seems to relate to how the Bootable ISOs are created by Systemback. By default the Live ISO is set for a conservative latency of 1024fpp, depending on your hardware AV Linux once installed to hard disk should easily run at significantly lower latencies.

➢ The LiveISO contains a well-rounded collection of applications but there are literally thousands more in the included repositories that are available after installing AV Linux, if you require applications that are not included on the ISO as provided then you will need an internet connection to enable installing and updating applications.

➢ By default the AV Linux LiveISO runs in English with the English keyboard map, if you need a different keyboard map during a Live session please see the 'Changing the Keyboard Settings' section of the Manual.

➢ By default the 'Performance' CPU Governor is enabled, this is an absolute '*must have*' setting for working with Audio and may negatively impact battery life to a small degree on laptops.

IMPORTANT PRE-INSTALLATION INFO!

Systemback and GPT Partition Table Support:

 AV Linux 2017 utilizes the Systemback Installer which at the time of this writing does **not** support formatting of and installation to Hard Drives with 'GPT' partition tables, many recent Windows 7,8,10 computers use GPT and you will not be able to install AV Linux 2017 as a dual-boot on these machines unless you change the partition table to an 'MSDOS' partition table.

Installation on UEFI Secure Boot Computers:

 In the past computer boot management and system hardware configuration outside of the installed Operating System was handled by the 'BIOS' (Basic Input Output System) but in recent years as a security feature the 'UEFI' (Unified Extensible Firmware Interface) has come to be the new standard. UEFI manages booting with 'Secure Boot' software and requires any new Operating Systems to be installed on the computer to have a digital signature that can be verified and validated by the Secure Boot software. At the time of this writing Systemback has no UEFI support for 32bit at all and has 64bit support only partially implemented, so AV Linux 2017 will likely not be validated for installation on most recent Windows 7,8,10 computers.

Summary and Workarounds:

 Generally speaking installation of AV Linux 2017 will currently be best suited to computers with Hard Drives formatted to the MSDOS partition table and BIOSes that pre-date UEFI OR have options in the BIOS to disable or circumvent Secure Boot.
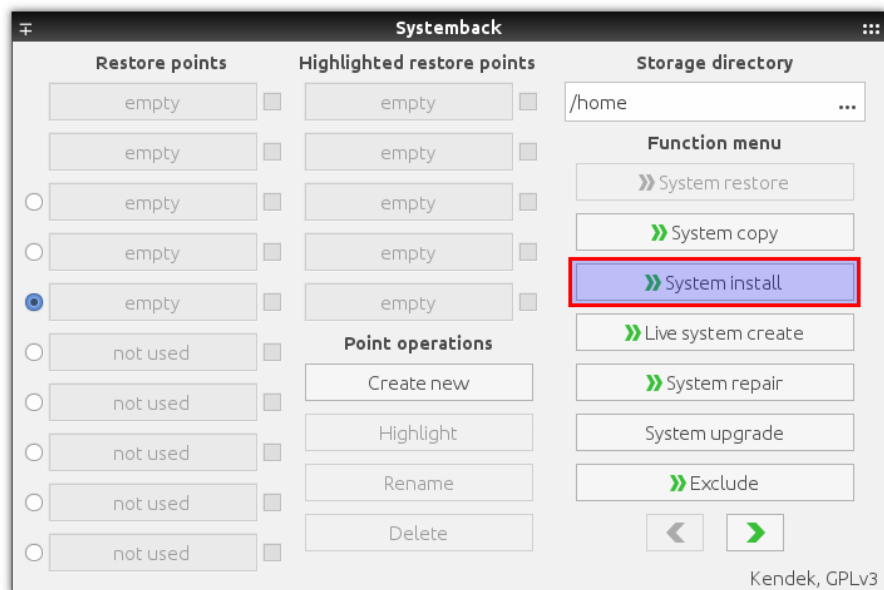
INSTALLING AV LINUX :

WARNING ! - The following section involves major system changes like hard drive partitioning etc. There is always a risk of Data loss or corruption when installing a new or different Operating System. *ALWAYS* back up any important Data before making changes to your existing Hard Drive. If you are completely unfamiliar with the concept of drive partitioning, or terms like "Master Boot Record" it is recommended that you don't attempt to install AV Linux.

Systemback Installation Info:

  Installation of AV Linux is now handled by 'Systemback' which is an excellent Linux Backup utility which contains an Installer as well. While it is not really intended to be an ISO tool for Distribution creation it does the job required to get AV Linux from ISO to the target system. Systemback's installer is not like a complete Distribution installer and does not provide setting up of locales etc. during the install process, it merely provides some basic disk formatting tools, copies the Live ISO Image as-is to the target machine and then installs the GRUB Bootloader. Users who require a locale and language support other than the default English will now find that installing AV Linux and setting it up for non-English locales is a 2-step process: Step 1 is to install the Live ISO. Step 2 is to use the ' AV Linux Assistant' to set up language, keyboard and locales after installing.


PLEASE NOTE! AV Linux does not use 'sudo' it uses the pure Debian Root Superuser and User method. When installing and setting up AV Linux you will need to have a separate and unique password for your Root Superuser to administrate the system and a regular User password for normal system use.

Installing From a Running Live Session

To access the Systemback Installer launch the Whisker Menu from the Panel AV button and either navigate to System → Systemback or type Systemback in the Menu search bar. Then click on the Systemback icon to launch it.
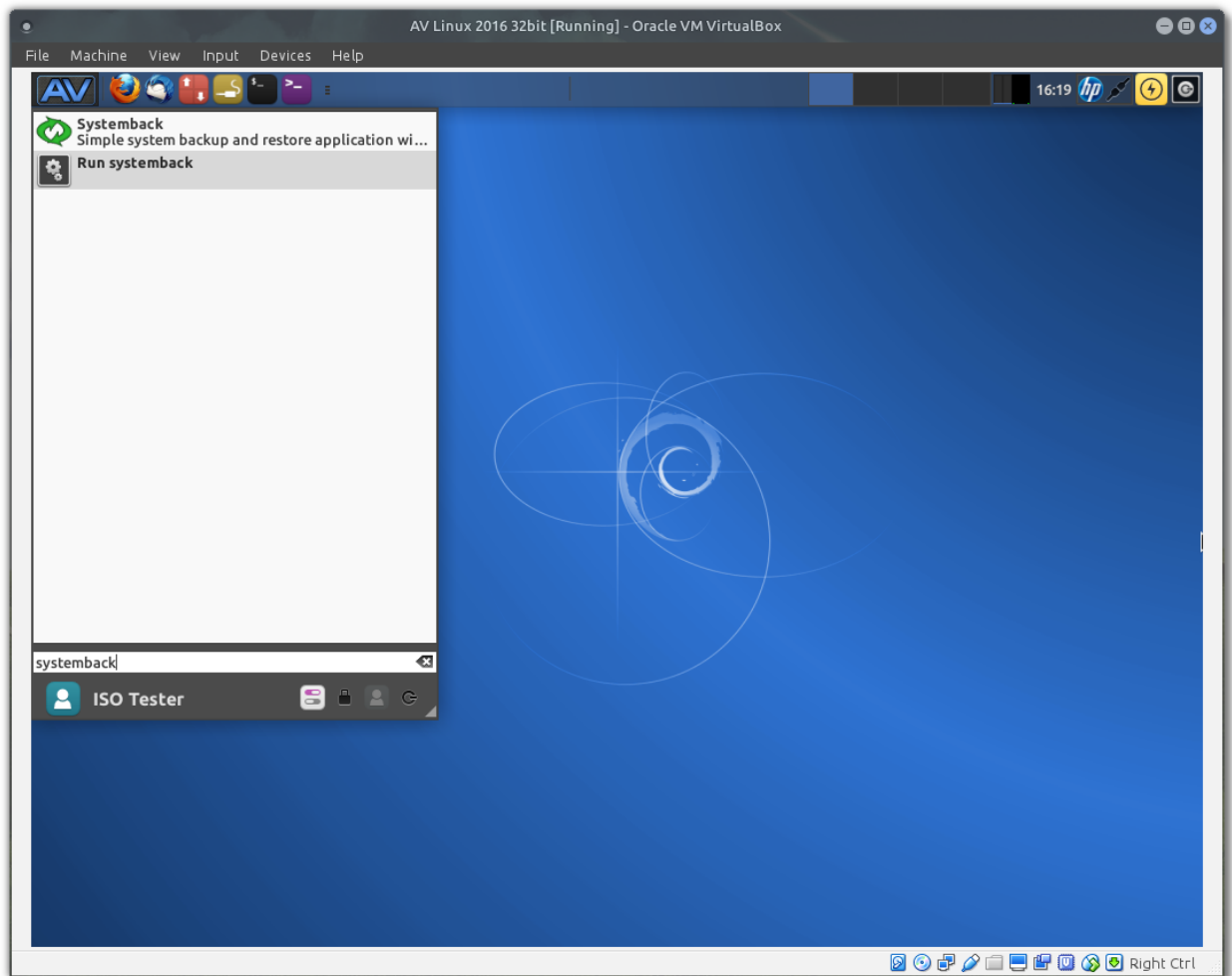
Once the Systemback window opens select the 'System install' button.
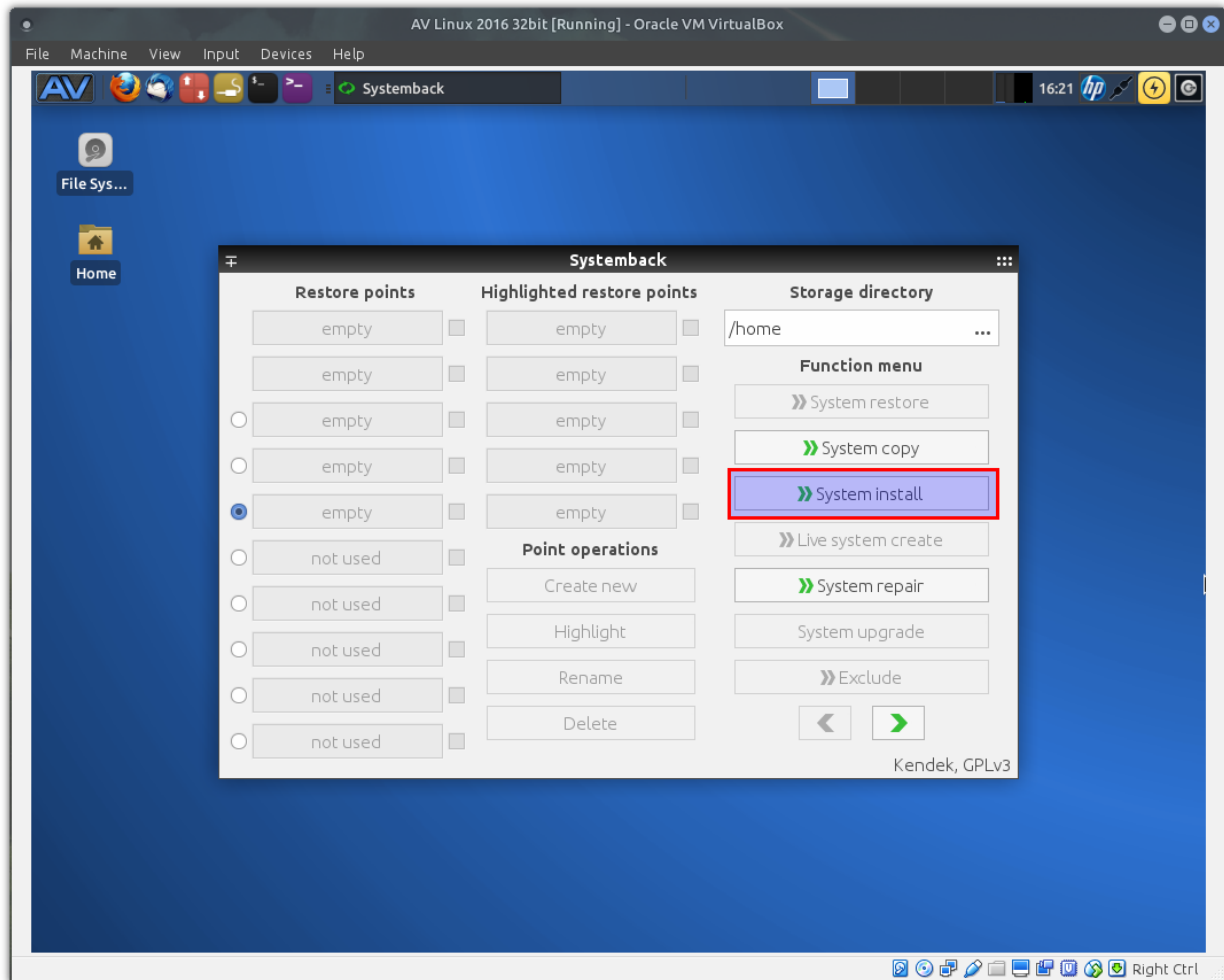
The window to set up your System User will open . You are required to fill in all the fields and will need to create separate, distinct passwords for both User and Root accounts. Fill in whatever you like for the hostname, this is how the computer will identify itself on the Network as well as what will be displayed at the command prompt in an open Terminal. When you are done filling in the info click on the 'Next' button.
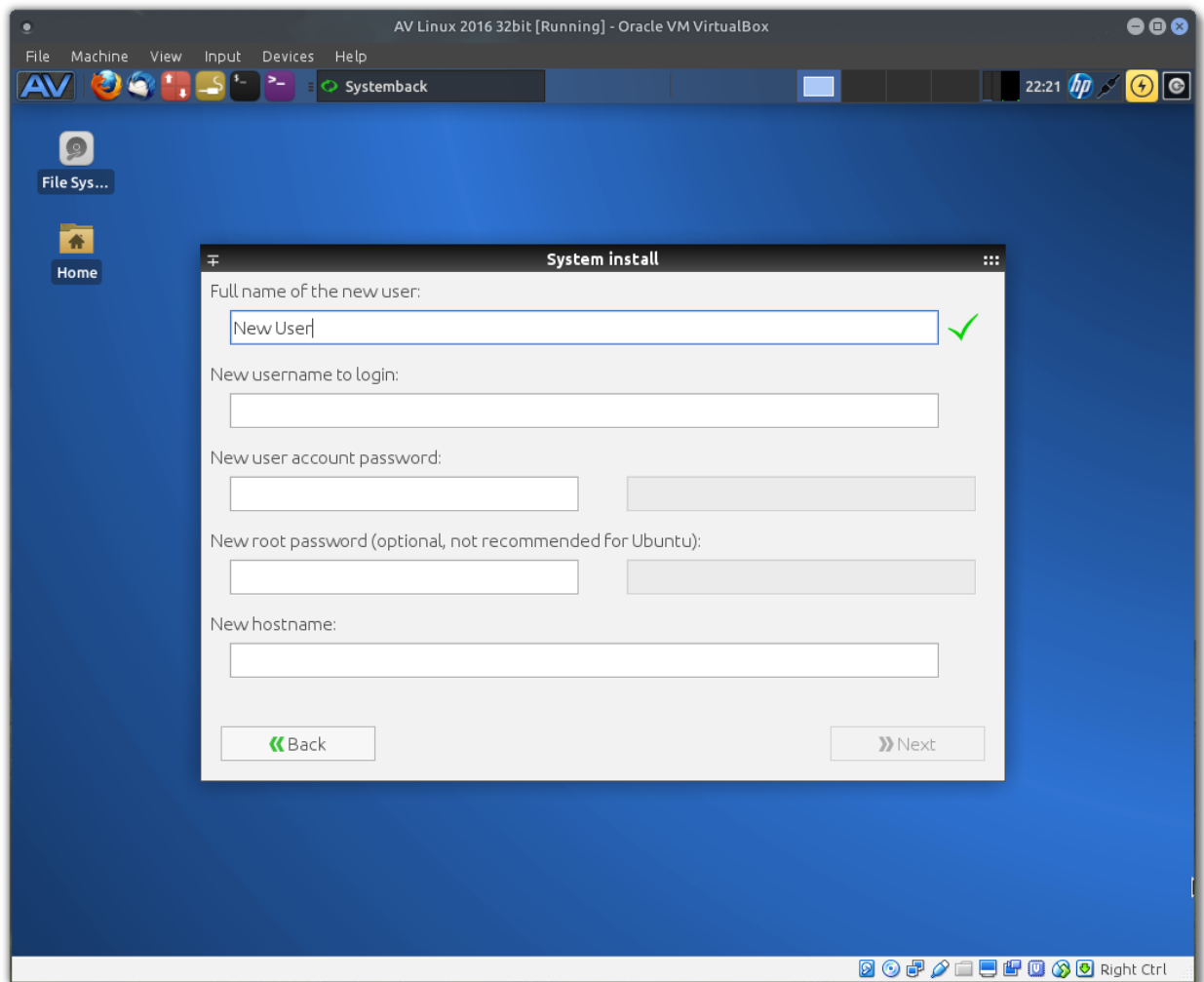
The 'Partition settings' window will open and contains a simple partitioning tool that will display all mounted disks and partitions on the system. **At this point you need to know what partitions are safe to format or modify on your system!** For this example we have a single 23GB 'ext4' primary partition and a 1GB swap primary partition, these are considered the basic, minimum requirements for installing AV Linux and in this example we are installing the entire OS (including the User's home) on the root('/') of the 23GB 'sda1'. *Of course your own system will differ in the sizes and numbers of partitions so adapt these instructions to your situation as needed.* To prepare any partitions you wish to target for an install it is suggested to unmount them by clicking the 'Unmount' button in the upper right (it will then toggle as '!Delete!'). Then highlight the target partition ('sda1' in this example) by clicking on it and then from the dropdown box under 'Mount point:' select the mount point (in this case '/') since we are targeting this partition for the install we want to erase any data on it so we select the 'Format' option. To apply your changes click the left-facing green Arrow button in the center-right, note that the circular green Arrow button beside it will reset your partitions to their original settings if you change your mind. Now do the same for the 1GB 'sda2' partition except select 'SWAP' as the filesystem. There is no need to copy any user config and data files over from a Live session and when AV Linux is installed it will automatically load some recommended system defaults so make sure to uncheck this option. Leave the GRUB 2 Bootloader set to 'Auto'. When you are satisfied with the settings click 'Next'.
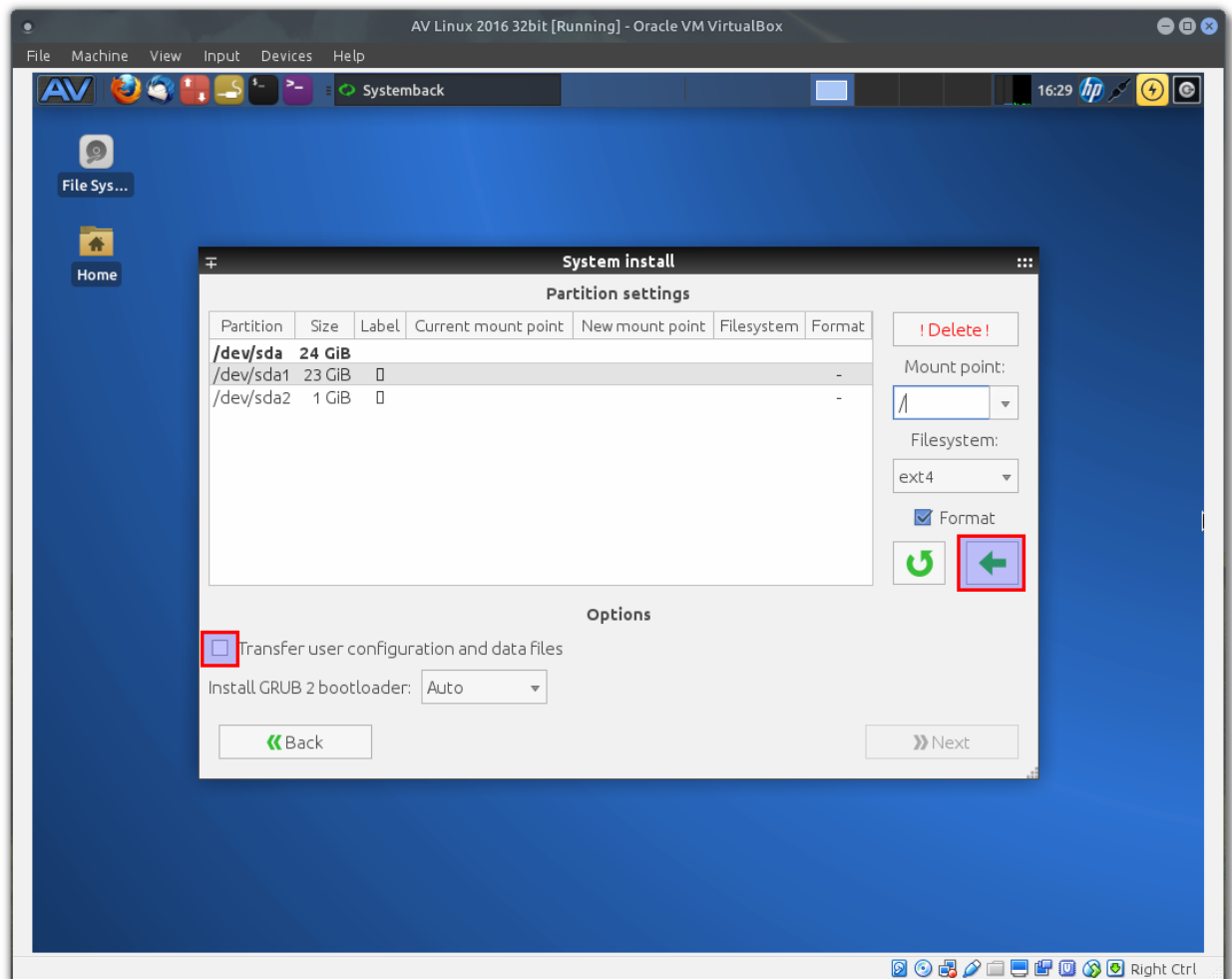
You will now be prompted to start the install, click 'Start' and a progress bar will appear and keep you informed of how its going, note that LiveUSB keys will install much faster than LiveDVDs. At the end of the process the GRUB2 bootloader will be installed.
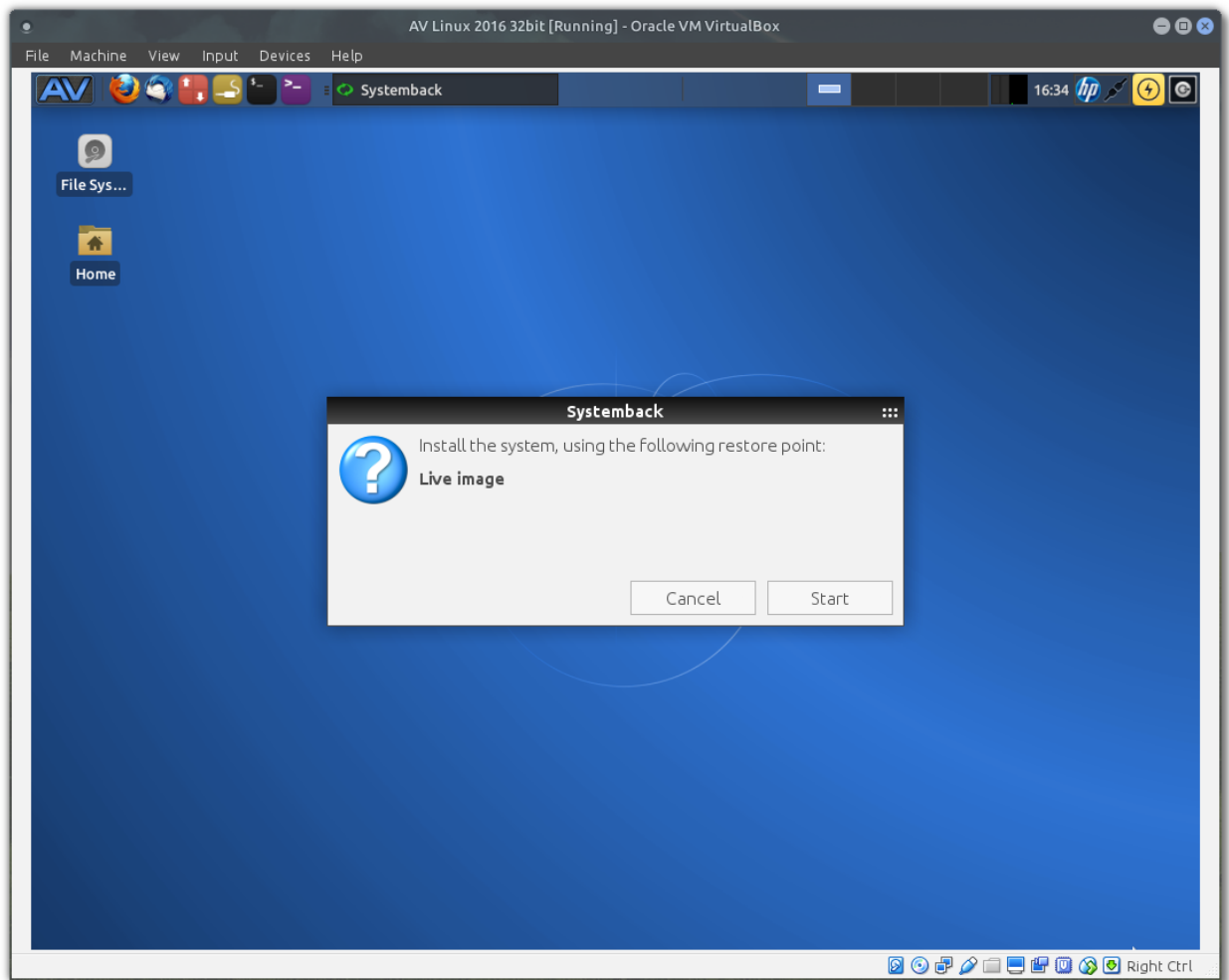
Once the install process is complete you will see this message dialog. At this point you can reboot into the installed system... well hopefully you can anyway.
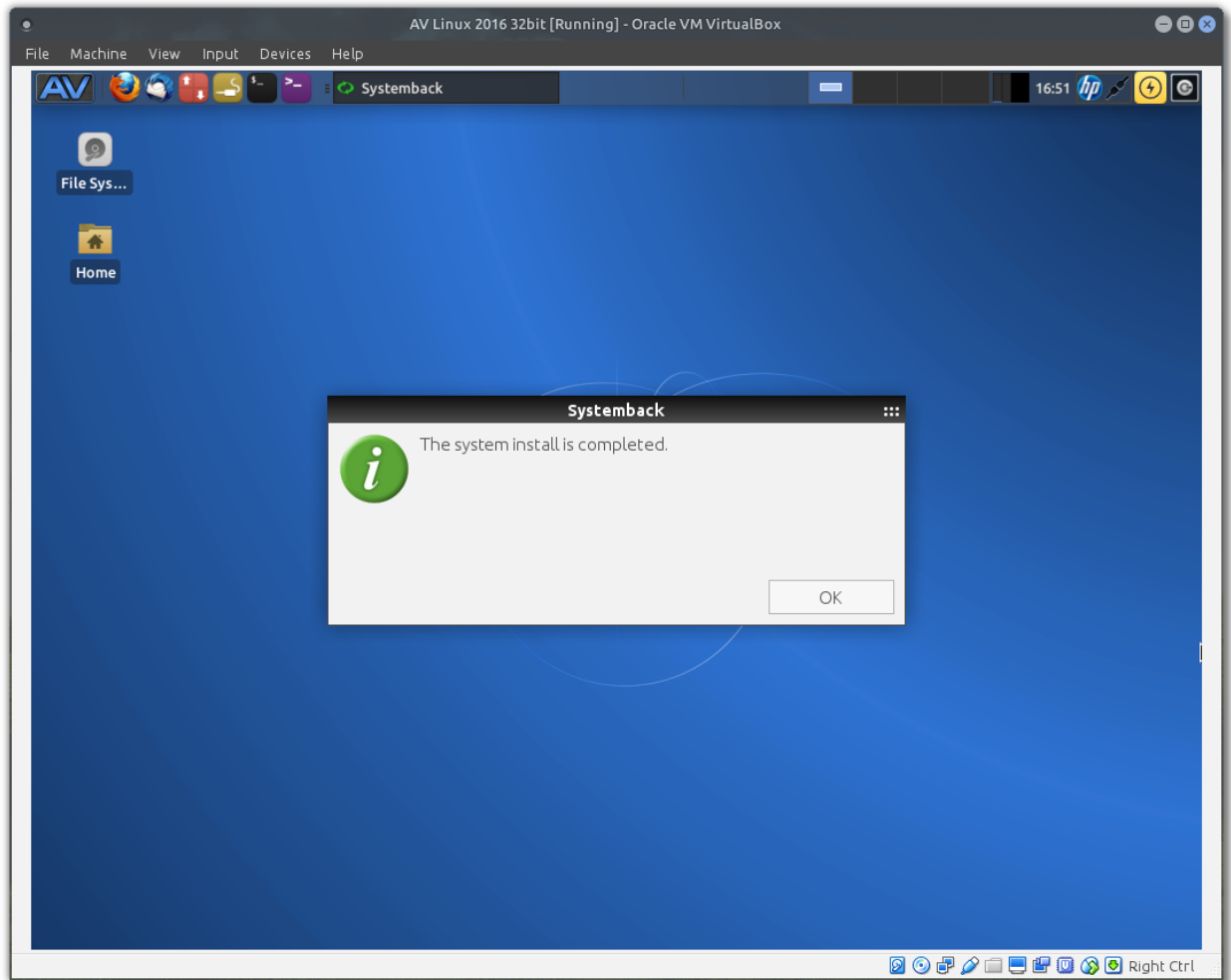
ADDING LOCALES AFTER INSTALL :

Post-Install Setup using the AV Linux Assistant :

  The Systemback installer used to install the Live system to HDD simply copies the running Live session as-is. If you require a different language and keyboard layout once AV Linux is installed you can use the 'AV Linux Assistant' to assist you with adding your locale and system keyboard settings.
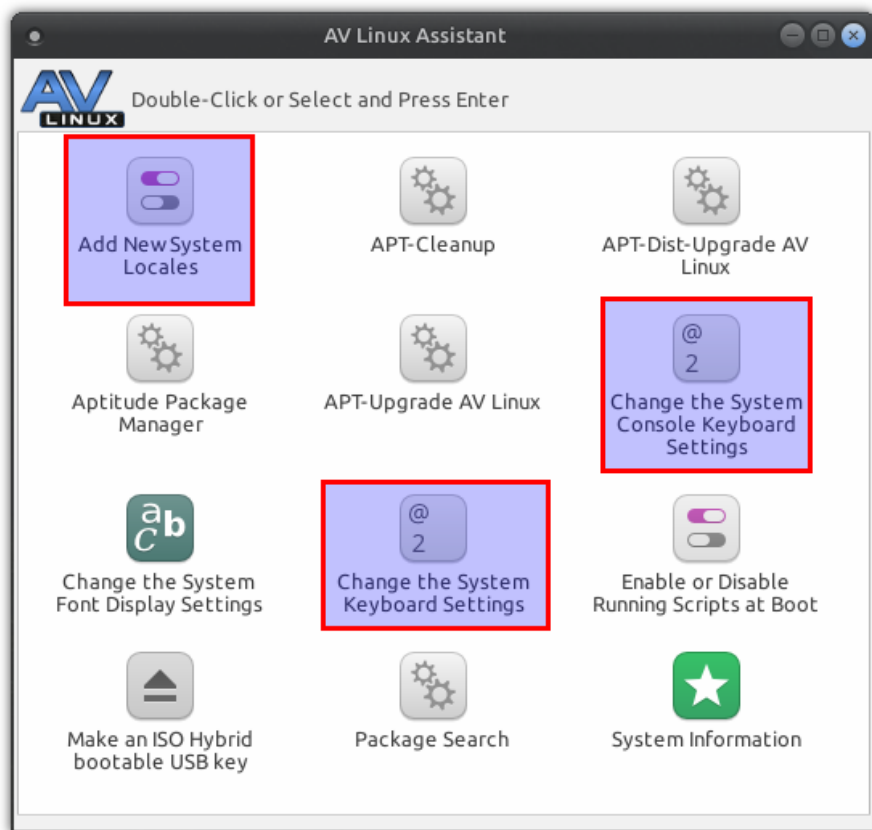
WHISKER MENU :

Customized Menu:

  AV Linux 2017 uses the XFCE4 'Whisker Menu' which allows you to browse Applications as you would in a regular Menu as well as providing a Search bar that you can type the name of the Application into. The Menu categories are customized using the KXStudio Menu which has been prepared to help users get to the applications they are looking for quickly and differs significantly from the stock XFCE4 menu.
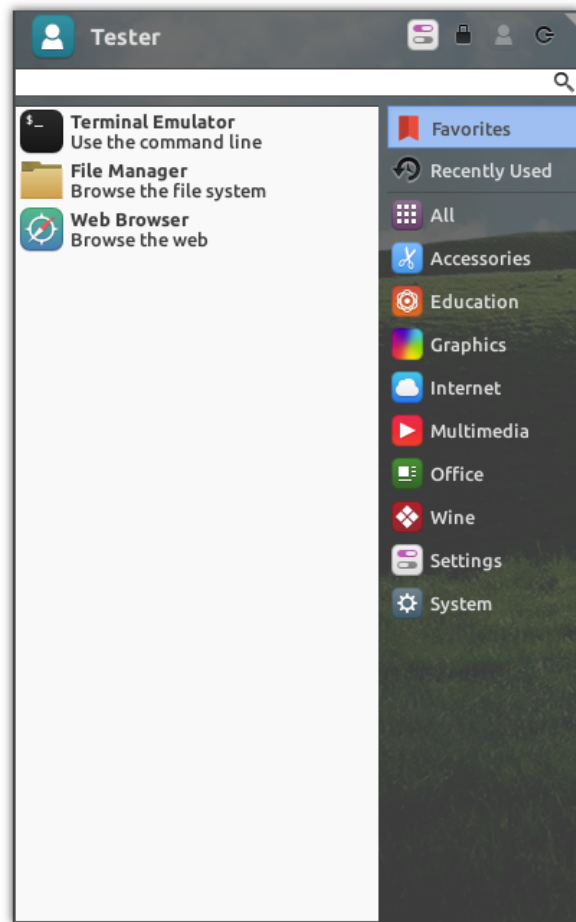
XFCE4 SETTINGS :

  The settings for XFCE4 are accessed by using the 'Settings Manager'. Find it by either typing 'Settings' in the Whisker Menu search bar or navigate to 'Settings' → 'Settings Manager'.

You can change how the system looks with the 'Appearance' settings. This will allow you to set the GTK theme, system icons, system fonts and some menu related items. AV Linux comes pre-loaded with a large amount of themes and colors to choose from.
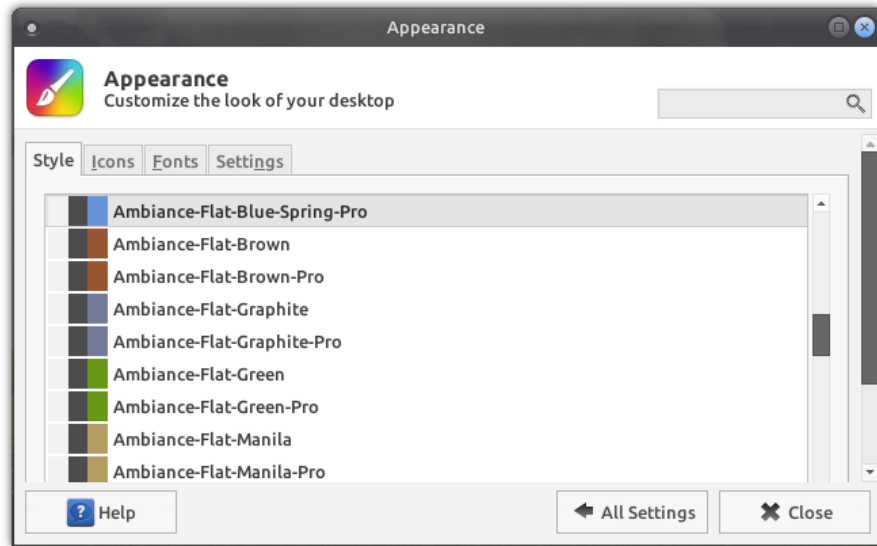
The Window Manager themes are not set in the 'Appearance' dialog, they have a separate 'Window Manager' applet in the Settings Manager.
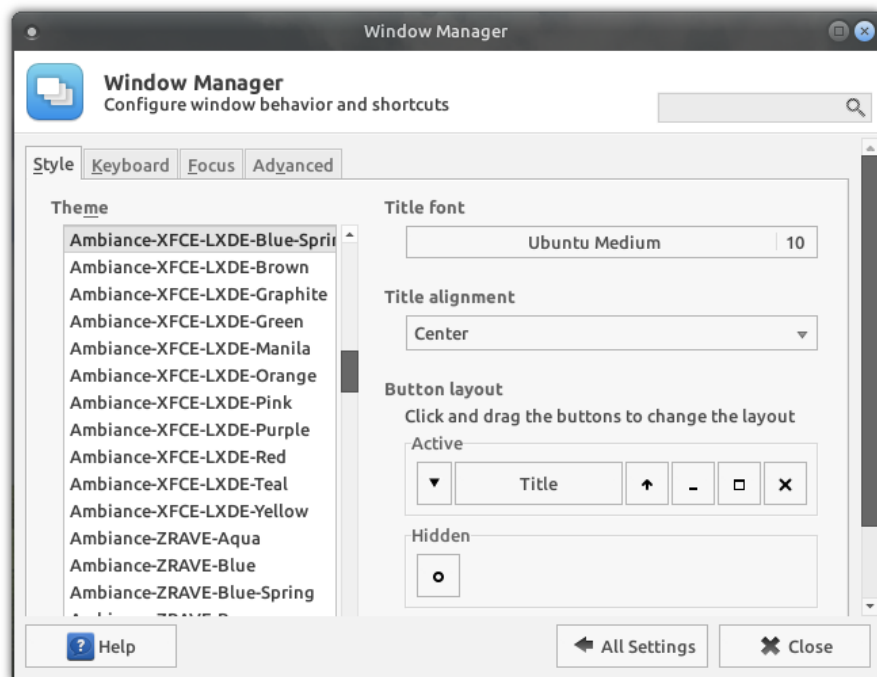
Change your Desktop wallpaper and other settings with the 'Desktop' applet in the Settings Manager or alternatively right-click on the Desktop and select 'Desktop Settings' from the Menu. Most common image file formats will also give you the ability to 'Set As Wallpaper' when you right-click on them.

THUNAR FILE MANAGER :

  The Thunar File Manager included in XFCE4 is extremely useful and for AV Linux has been customized and extended with Custom Actions to do MUCH more than simply manage files. As you navigate the System you will see there are numerous options when you right-click on the Desktop and in folders , more different options appear when you right -click on certain file types. Custom Actions enable many tasks including system administration, media playing, media conversions, networking and more! A summary of the Custom Actions is shown below, most options are self-explanatory. Some will be explained further on following pages.
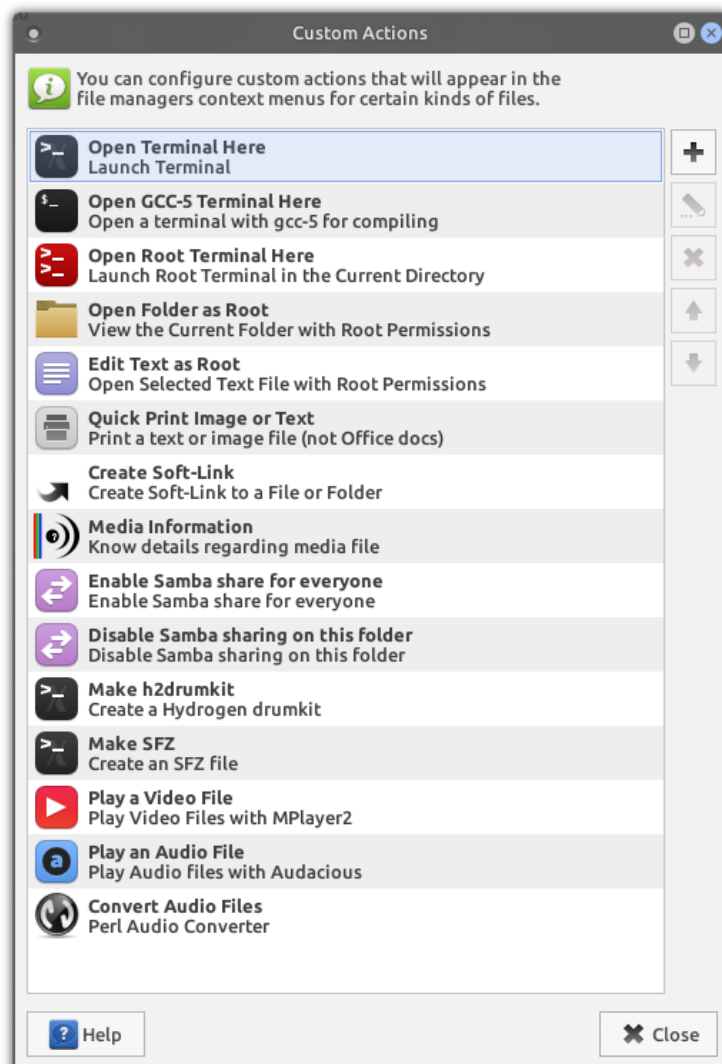
Choice of Compilers :

  A unique Custom Action feature of interest to developers and those who like to compile their own programs is the ability to choose which 'gcc' version to compile software source code with. The AV Linux kernel is currently compiled with gcc-4.9 so to build modules (ie for VirtualBox) it is important that gcc-4.9 is used and for that reason it is the default gcc version on AV Linux. You may find that for compiling newer versions of programs you require gcc-5 or even gcc-6 which has significant ABI changes while older programs may need to be compiled with a gcc-4.9. Don't forget AV Linux is OOTB developer-friendly containing most common build systems, source code tools and development libraries.
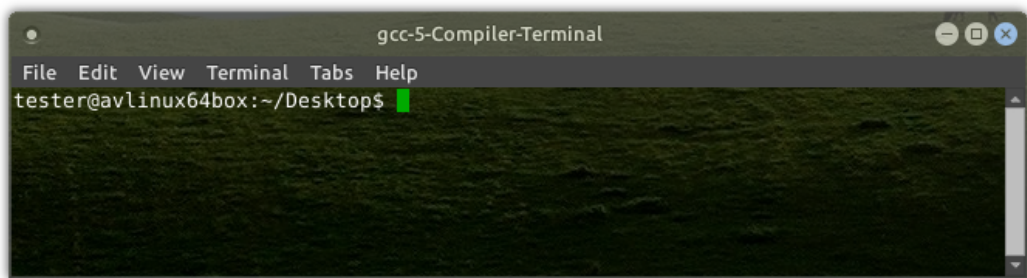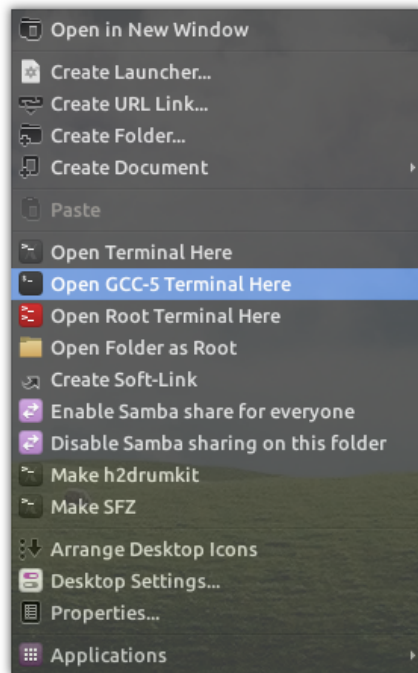
One-Click Samba Sharing on Your Network :

* Note this will expose any folders you enable for sharing on your Network with permissions for ALL users on your Network, if you are concerned about the security of your network, or are not sure which files you are making available DO NOT enable Samba sharing! It is also not recommended to share the root directory of a drive and all its contents on a Network.

 Right-clicking on a folder in Thunar will allow you to 'one-click' share it on your network.
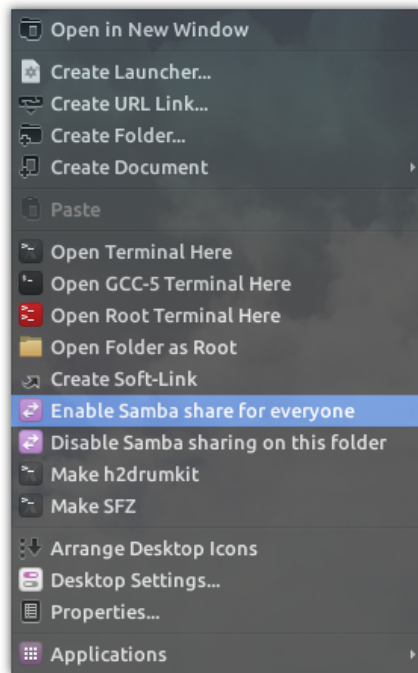

 Once you have enabled sharing on the folder you should be able to see it on your network by going to 'Browse Network' at the bottom of the left-hand pane of the main Thunar Window. The computer you are using should appear in the Network by its hostname. If you select the named Computer the shared file should appear with any other shared folders on the system. It is important to note that the folder you right-clicked on to share will not look any different when viewed in Thunar. In order to remind yourself that you have shared it you may want to right-click on it, select 'Properties' → 'Emblems' and attach a sharing emblem to the folder.
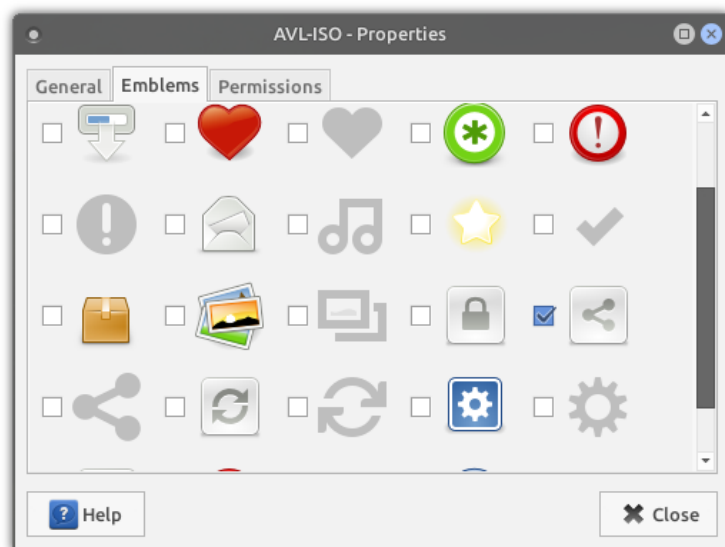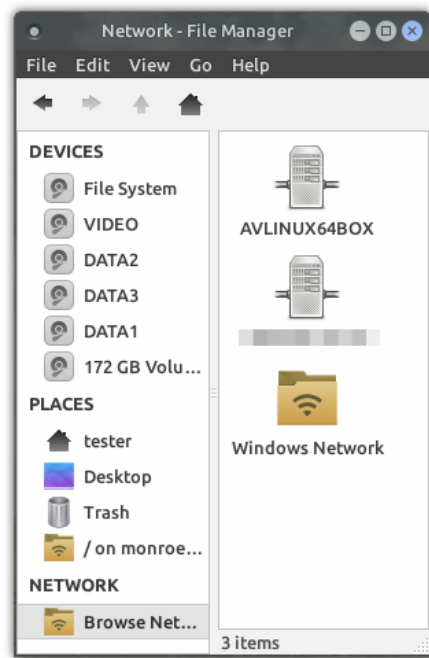
Creating an SFZ Library:

Two very handy Custom Action Menus exist to quickly and easily create two kinds of Sound Libraries right from within Thunar, the workflow in principle is similar for both but for the Manual I'll quickly go over an example to create an SFZ Sound Library.

First you'll need all your recorded samples collected together in one folder, the better job you do of sequentially naming your samples the more accurate the end result will be. In the example picture below you will see a folder of samples to be made into an SFZ. In this case the intent is to have 5 velocity-layered samples assigned to each key so taking '*36-CajonCenter-1.wav*' as an example you can see the '36' is the intended key note number, the actual sample name is next and '1' is the layer number, meaning on key 36 the quietest sample to be played is '*36-CajonCenter-1*' and the loudest sample to be played will be '*36-CajonCenter-*5'.
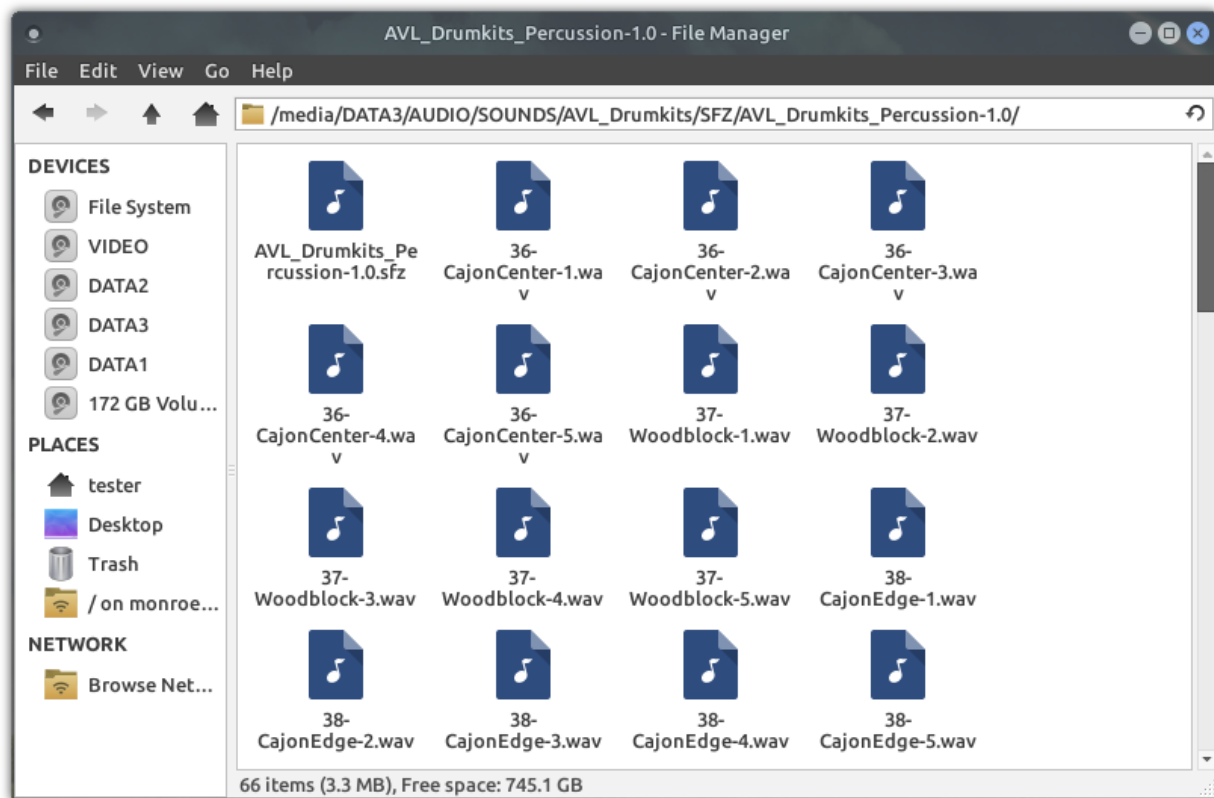
After your samples are organized and ready navigate to your samples folder with Thunar and right-click within the folder.
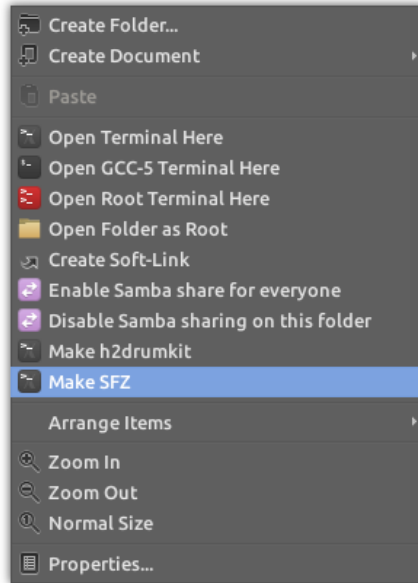

You will then see an interactive X-Terminal open a script with prompts to follow, it will be very beneficial to know ahead of time how you want your SFZ formatted. Some options you will have during the process will be to choose if you want every individual sample mapped to a single key or multiple samples mapped to a key, whether you want 'one-shot' or 'looped' modes and many more SFZ-specific features and details.
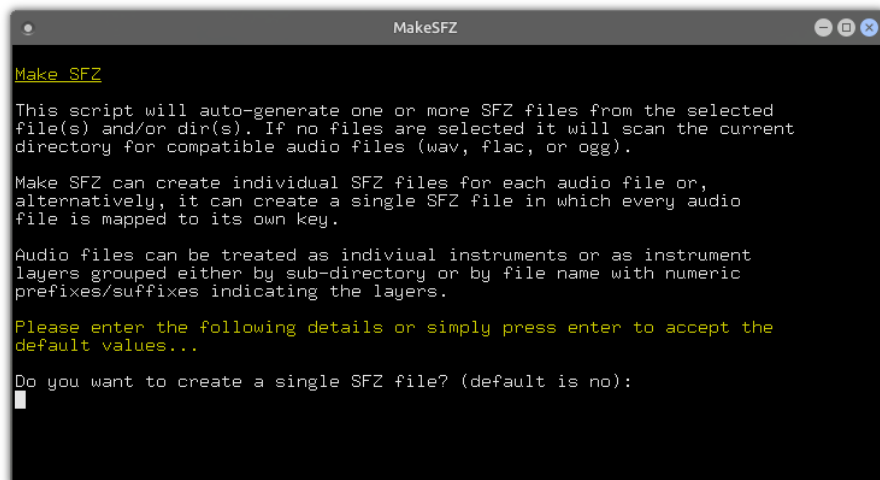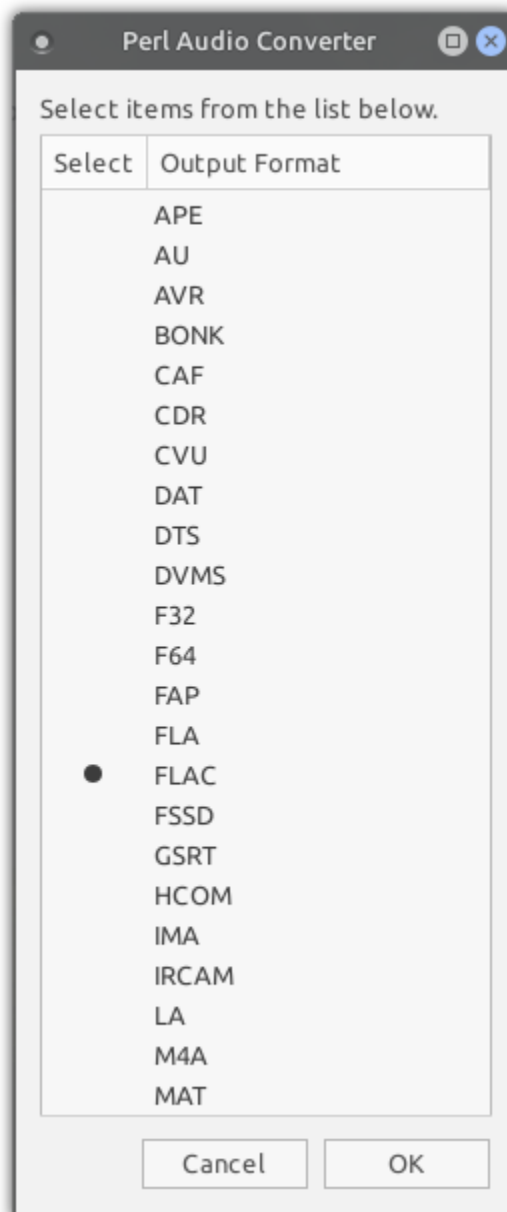
Convert Audio Files With PACPL :

  You can quickly and easily convert Audio files from one format to another with the integrated PACPL Custom Action. When you right click on an Audio file in a folder you will will see the option to "Convert Audio Files" as one of the menu options. Pretty much all common options for conversion are supported.
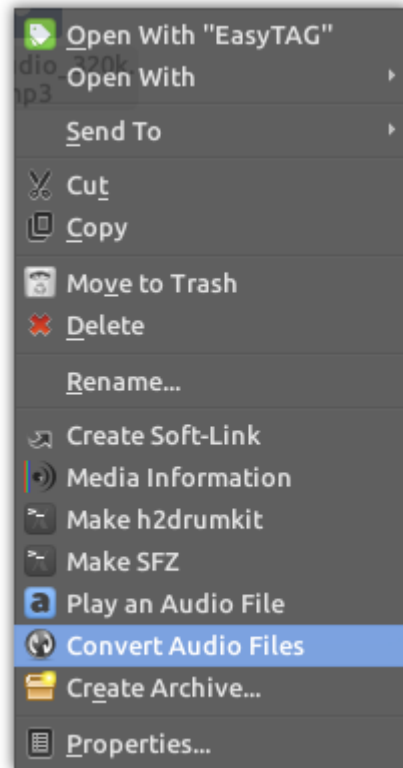
  When you select this option from the menu a dialog will appear with a listing of available codecs to choose from, clicking on the left side of the dialog will select the codecs you want. If you want to tweak the bitrate options for some codecs you can edit the configuration file for PACPL in /etc/pacpl/pacpl.conf. To open this file for editing right-click on it and select 'Edit Text As Root' which will prompt you for your Root password and then open it for editing in 'Mousepad'. Once you have made your changes don't forget to save them in Mousepad.

Perl Audio Converter

Select items from the list below.

Select	Output Format
	APE
	AU
	AVR
	BONK
	CAF
	CDR
	CVU
	DAT
	DTS
	DVMS
	F32
	F64
	FAP
	FLA
●	FLAC
	FSSD
	GSRT
	HCOM
	IMA
	IRCAM
	LA
	M4A
	MAT

Cancel   OK

DEBIAN SYSTEM ADMINISTRATION BASICS :

Simple Commands for SysAdmin Tasks :

  As a Linux user I used to wonder why XFCE4 and other light Desktop Environments didn't have any fancy graphical tools for common tasks like adding or removing users and password changes. After learning how simple the actual Terminal commands were I could see that trying to put these tasks into a GUI could potentially make them more confusing and time consuming than they really need to be. In this section I've tried to demonstrate the most commonly used SysAdmin commands.

  To add a new User to your system and create a new folder for them in the System '/home'folder you need to know what the new User's name will be and run this command in a Root Terminal:

`adduser <USERNAME>`

  Obviously you replace '<USERNAME>' with the actual User name, when you execute this command you will be prompted for some more information and when finished the new users folder will be created in the system '/home' folder.

  To add a system user to a 'Group' on the system (ie the 'lpadmin' Group to allow printing) then use this command in a Root Terminal:

`adduser <USERNAME> <GROUP>`

To add a Group to the System use this command in a Root Terminal:

`addgroup <GROUPNAME>`

To remove a user from the System you use this command in a Root Terminal:

`deluser <USERNAME>`

To remove the User and their User folder from the systems '/home' add this:

`deluser <USERNAME> --remove-home`

To remove a User from a Group use this command:

`deluser <USERNAME> <GROUP>`

To change your 'Root' Password use this command in a Root Terminal:

`passwd`

To change a User's Password use this command in a Root Terminal:

`passwd <USERNAME>`

CHANGING USER AND GROUP ID's

How to change 'UID' and 'GID' for System Users:

   By default the primary user on a Linux system is assigned a numerical User and Group ID (a.k.a. 'UID' and 'GID') of '1000', additional users are assigned numbers in excess of 1000 (ie 1001, 1002, etc). File and folder ownership permissions are assigned to the User and Group that created them. A scenario where this may come into play is if you are a secondary user and the primary user has been removed from the system now making you the new primary user/System Administrator, the files of the primary user will all have UID's and GID's of '1000' so since you are a secondary user with a UID/GID of '1001' you will not have ownership permissions of any files associated with User or Group '1000'. To fix this you'll need to change your UID/GID.

 AV Linux 2017 has scripts to ease the changing of User and Group ID's for situations where it is necessary, ***it is highly recommended to not use these scripts unless you know exactly what you are doing.***

  In order to be used these scripts must be run as Root outside of the 'X' graphical environment in the the basic system Console this ensure that no files to be changed are being used by the graphical Desktop Environment. There are 2 ways to login to the bare Console:

1. Boot the system and select 'Advanced Options' from the initial GRUB Boot menu, then select 'Recovery Mode' in the Kernel Boot choices. When the computer completes it's boot in recovery mode you will arrive at a login prompt to the Console, in some cases you may need to hit the 'Enter' key on your computer keyboard to initiate the login prompt. Once prompted enter your Username as 'root' and then enter your Root password. You are now logged in to the Console as Root and can proceed to run the scripts

2. If you are using your computer you can use the Ctrl+Alt+F3 key to get out of X, you will then be presented with a bare Console login prompt and can enter the Username 'root' and your Root password at the Console prompt.

Using the UID/GID Scripts:

 It is very important to note that you use these scripts for the regular Username you log into the system with. *As an example we are changing the UID/GID for a User named 'tester' from '1001' to '1000'.* Once logged in to the Console as Root enter this command to change your UID:

ChangeUID.sh

This will launch an interactive script which will indicate the syntax required.

ChangeUID.sh [username] [oldUID] [newUID]

  To demonstrate our example from above complete the command and fill in the required information like the example below, then hit 'Enter':

ChangeUID.sh tester 1001 1000

Depending on the size of your Hard Drives and number of files on your system it could take the scripts several seconds to a few minutes to find and change the file ownerships. It is normal to see some warnings and terminal output about files located in '/proc'.

  We are not done yet, so far only the User ID's have been changed, we now need to to also change the Group ID's. **You must change both User ID's and Group ID's.** Group ID's are done with a different script, as before we will use the example of the 'tester' Group being changed from '1001' to '1000'.

`ChangeGID.sh`

This will launch an interactive script which will indicate the syntax required.

ChangeGID.sh [username] [oldGID] [newGID]

  To demonstrate our example from above complete the command and fill in the required information like the example below then hit 'Enter':

`ChangeGID.sh tester 1001 1000`

As before, depending on the size of your Hard Drives and number of files on your system it could take the scripts several seconds to a few minutes to find and change the file Group ownerships. It is normal to see some warnings and terminal output about files located in '/proc'.

CHANGING KEYBOARD SETTINGS :

```
@
2
```

Changing the Keyboard Layout for Other Languages:

  To set the keyboard for a different language during a LiveISO session open a Terminal and use the 'setxkbmap' command with the accompanying 2 letter code for your language. The screenshot below shows the command for French (setxkbmap fr).
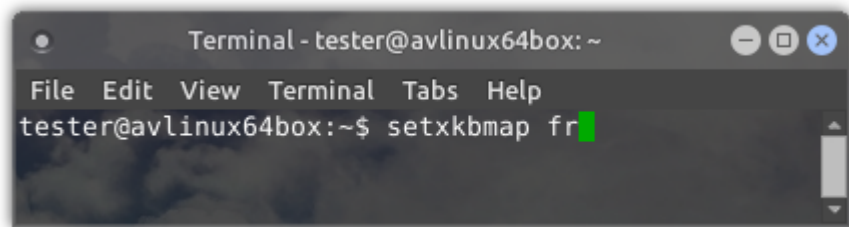
  To permanently change the keyboard setting in an AV Linux install use the Keyboard setting utilities found in the AV Linux Assistant. You can also change the System keyboard manually by opening a Root Terminal from the Accessories menu and editing the configuration file in /etc/default/keyboard with this command:

mousepad /etc/default/keyboard

Change the Model and Country codes within the quotes to the correct ones for your Country and save the changes.

XKBMODEL="pc104"

XKBLAYOUT="us"

XKBVARIANT=""

XKBOPTIONS=""

SliM LOGIN MANAGER :

Setting SliM For Autologin:

AV Linux features 'SliM' as its default login manager. SliM has a very light system footprint and is still quite configurable by editing its configuration file in /etc/slim.conf. The example below will cover the most popular configuration request...autologin.

First open a Root Terminal from the 'Accessories' Menu and enter:

`mousepad /etc/slim.conf`

To enable autologin two separate variables must be changed in slim.conf. Scroll down the file to find the following line:

`# default_user`

To set yourself as the default user remove the '#' comment from the beginning of the line and then add your username, USERNAME is used here as an example:

`default_user USERNAME`

If we leave the file like this then on next login it will automatically enter your username but still require your password, to have autologin work fully you need to edit the autologin line:

`# auto_login no`

Once again remove the '#' comment and change the 'no' to a 'yes' like this:

`auto_login yes`

When you have completed editing the slim.conf file click 'Save' in the Mousepad Text Editor and autologin should be active on your next login.

*SliM does NOT support remote login, if you require remote login it is recommended to install 'lightDM' from the Debian repositories and configure it as the default login manager.

Setting SliM Default Session Using Other Desktop Environments:

Open a Root Terminal from the 'Accessories' Menu and enter:

mousepad /etc/slim.conf

Scroll down the file to find the following line:

login_cmd   exec /bin/bash -login /etc/X11/Xsession %session

Change '%session' to the start command of your desired other default Desktop Environment.

Command Examples For Some Other Common Light Desktops:
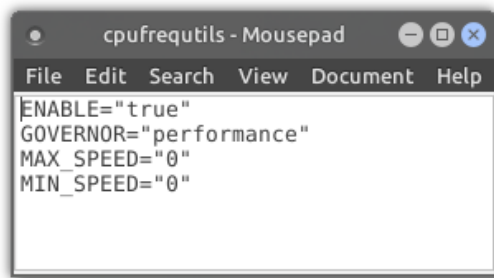
default,startxfce4,openbox,ion3,icewm,wmaker,blackbox,awesome
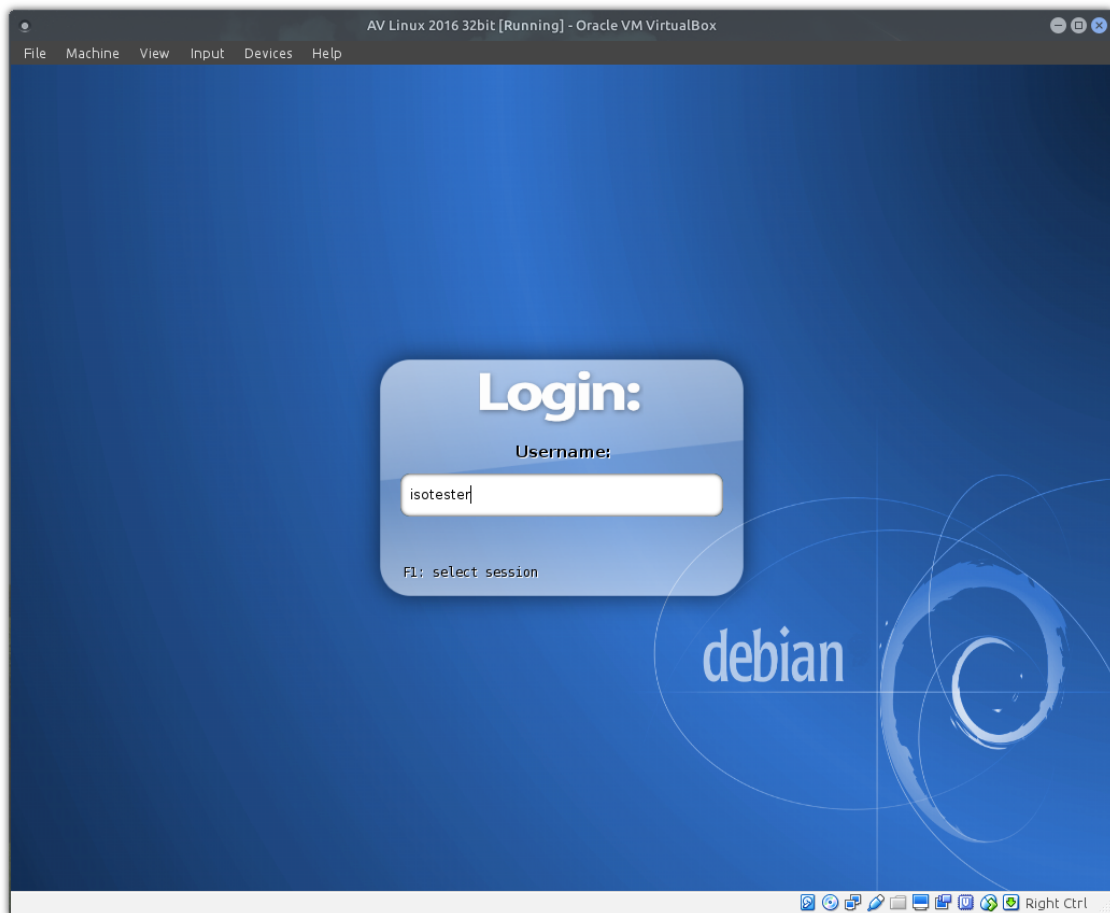
PERFORMANCE SETTINGS :

CPU Governor Settings:

Governing the CPU in AV Linux is set by the 'cpufrequtils' configuration file found in /etc/default. For Audio work it is NOT recommended to change this setting.

Power Management:

To change settings related to System power consumption use the XFCE4 Power Manager.

MONITOR AND DISPLAY SETTINGS :

AV Linux uses an application called 'Arandr' to handle setting the monitor screen resolution and also for setting up Dual-head displays. Arandr can be found in the 'Settings' menu. To change and select the monitor resolution right-click on the rectangle within the Arandr window that represents the Video output and display you are using. If you are using more than one output and display there will be more than one rectangle present within the Arandr window.

PLEASE NOTE To use Dual-head for certain nVidia and AMD Video cards it may be necessary to have Proprietary Video Drivers installed which is not possible with the AV Linux RT Kernel.
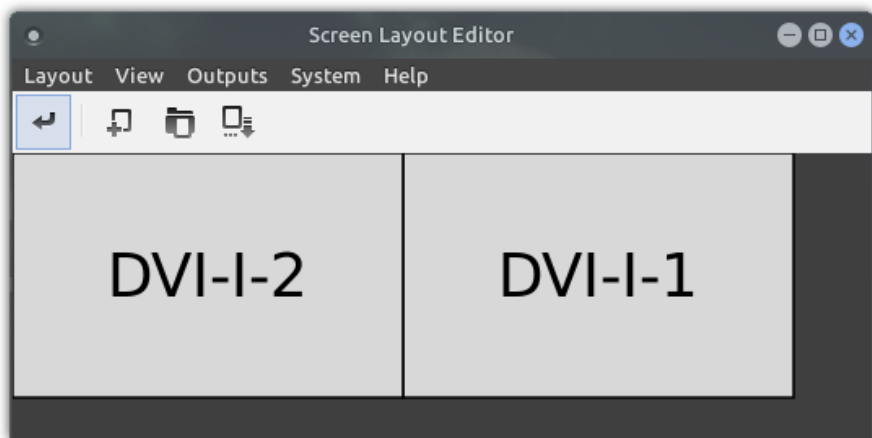
Making a Persistent Dual-Head Setup:

  If you have a permanent setup with Dual-head monitors you may want to have the Arandr configuration settings automatically start it up for you when you boot up your computer. This is possible but requires some extra steps and placing a '.desktop' launcher in /home/<YOURUSERNAME>/.config/autostart. Here's how to do it:

- ➢   Start Arandr configure the displays as desired.

- ➢   Save and name your configuration from the 'Save As' menu (ie dualhead.sh).

- ➢   'dualhead.sh' will be saved in /home/<YOURUSERNAME>/.screenlayout by default.

- ➢   Find your created 'dualhead.sh' in the '.screenlayout' folder and right-click on it.

- ➢   Go to 'Properties' -->'Permissions' and make sure 'Allow executing file as program' is checked.

- ➢   In your home folder right-click and 'Create Document' → 'Empty File' and name it 'dualhead.desktop'.

- ➢   Right-click on the 'dualhead.desktop' file you created and 'Open with Mousepad'.

- ➢   From the example below copy and paste the Launcher text into the 'dualhead.desktop' file.

- ➢   Change the yellow highlighted '<YOURUSERNAME>' to your actual User name.

- ➢   Save your changes in Mousepad

- ➢   Copy the 'dualhead .desktop' file to /home/<YOURUSERNAME>/.config/autostart.

- ➢   On reboot your ArandR configuration should be automatically loaded.

Example 'dualhead.desktop' Launcher File:

```
[Desktop Entry]
Encoding=UTF-8
Version=1.0
Type=Application
Categories=System
Terminal=false
Icon=
Name=Dualhead
Exec=/home/<YOURUSERNAME>/.screenlayout/dualhead.sh
Comment=Autostart Dual Monitors
```
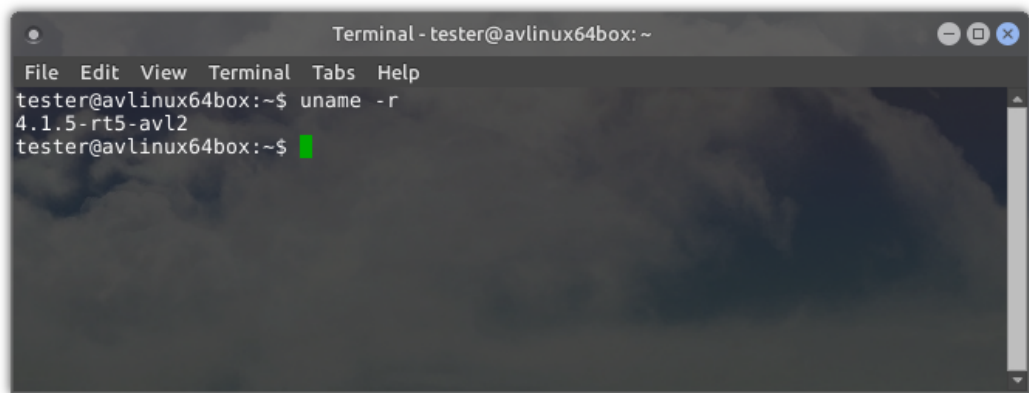
KERNEL NOTES :

Default RT Kernel:
** 'RT' Kernels are incompatible with nVidia and AMD 3rd party Proprietary Video Drivers !!

  AV Linux offers a range of choices for post-install Kernels, by default it features the performance advantages of a custom RT Kernel. Updated AV Linux Kernels can be installed from the AV Linux Kernel Repository which is enabled by default in the AV Linux 2017 Software Sources. If for some reason you require features that are not in the AV Linux kernel it is possible to install the 'Liquorix' kernels or the stock Debian kernels using Synaptic Package Manager. It is strongly recommended to only change or update your Kernel if absolutely necessary.


```
Terminal - tester@avlinux64box: ~

File   Edit   View   Terminal   Tabs   Help
tester@avlinux64box:~$ uname -r
4.1.5-rt5-avl2
tester@avlinux64box:~$
```

Cheatcodes Enabled By Default in AV Linux :

threadirqs - force-threaded irq handlers (realtime preemption)
 - Threaded irq's have long been a major part of the RT_PREEMPT patchset, and can offer significant benefits for low latency audio uses. Use this parameter unless you have a really good reason not to.

transparent_hugepage=never - turn off hugepages (a new memory management feature)
 - Transparent hugepages should be useful in memory intensive applications, such as virtual machines, databases, and possibly video editing. If ultra-low latency is desired, it seems to be best to disable them. Use this parameter if you want to squeeze the lowest latency possible out of your audio chain.

To Be Enabled/Disabled Optionally:

noautogroup - disable auto cgroup scheduling
 - The famous "200 line patch that does wonders", aka SCHED_AUTOGROUP, is useful when cpu-intensive tasks, such as make, are being done in a terminal, while the desktop is in use. Use this parameter to disable it if you think it is causing problems.

Example grub kernel command line:
 threadirqs transparent_hugepage=never quiet

Making GRUB-2 Add the Boot Flags Automatically:

1. As root, edit:
/etc/default/grub

2. Add the desired boot flags to the GRUB_CMDLINE_LINUX_DEFAULT line, like this:
GRUB_CMDLINE_LINUX_DEFAULT="threadirqs transparent_hugepage=never"

3. Then run: (also as root)
update-grub

4. Reboot

AUDIO & MIDI IN AV LINUX :
## *Recommended Reading!*

PulseAudio with JACK :

 It cannot be overstated that AV Linux differs *significantly* from your usual Linux Desktop in it's handling of Audio routing and subsystems. AV Linux relies on the low-latency Jack Audio Connection Kit (a.k.a JACK) for its powerful interconnectivity. JACK is a low-latency sound server which runs on top of ALSA (Advanced Linux Sound Architecture). ALSA is the basic 'driver' level of Audio hardware support and is both integrated in the Kernel itself as well as being a core component of the Operating System. In recent years PulseAudio can be made to peacefully coexist with JACK and AV Linux makes this easier by pre-configuring JACK and PulseAudio to work together with a custom 'pajackconnect' script .

Setting things up with JACK :

 The most initially confusing thing for many people getting started with Linux Audio is how to get the applications they want to use to 'see' (or perhaps hear) their Audio device. Reading this section will hopefully facilitate getting everything to 'just work'. The biggest step required of the user is to *identify* and *select* the Audio device they want to use, after that most of the complexity is looked after for you.

 The easiest way to setup the JACK Audio server is to use the JACK Control app (a.k.a. Qjackctl). It can easily be launched from the XFCE4 Panel or Whisker Menu. To setup your Audio device click the 'Setup' button.
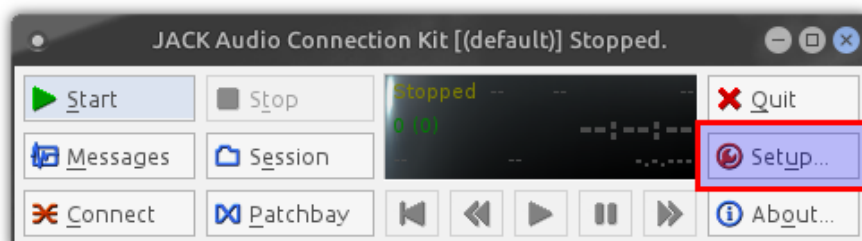
 The 'Setup' Window has multiple tabs which we will go over one at a time, the 'Settings' tab is where we initially select and set up the devicewe want to use with JACK in the 'Parameters' tab. To the left is the 'Driver' select dropdown box. Internal, PCI, and USB 1.1 and 2.0 Audio devices require the 'alsa' driver and FireWire devices require you to select the 'firewire' driver. It is essential that the 'Realtime' checkbox is selected. Next we need to select our desired Audio device from the 'Interface' dropdown box.
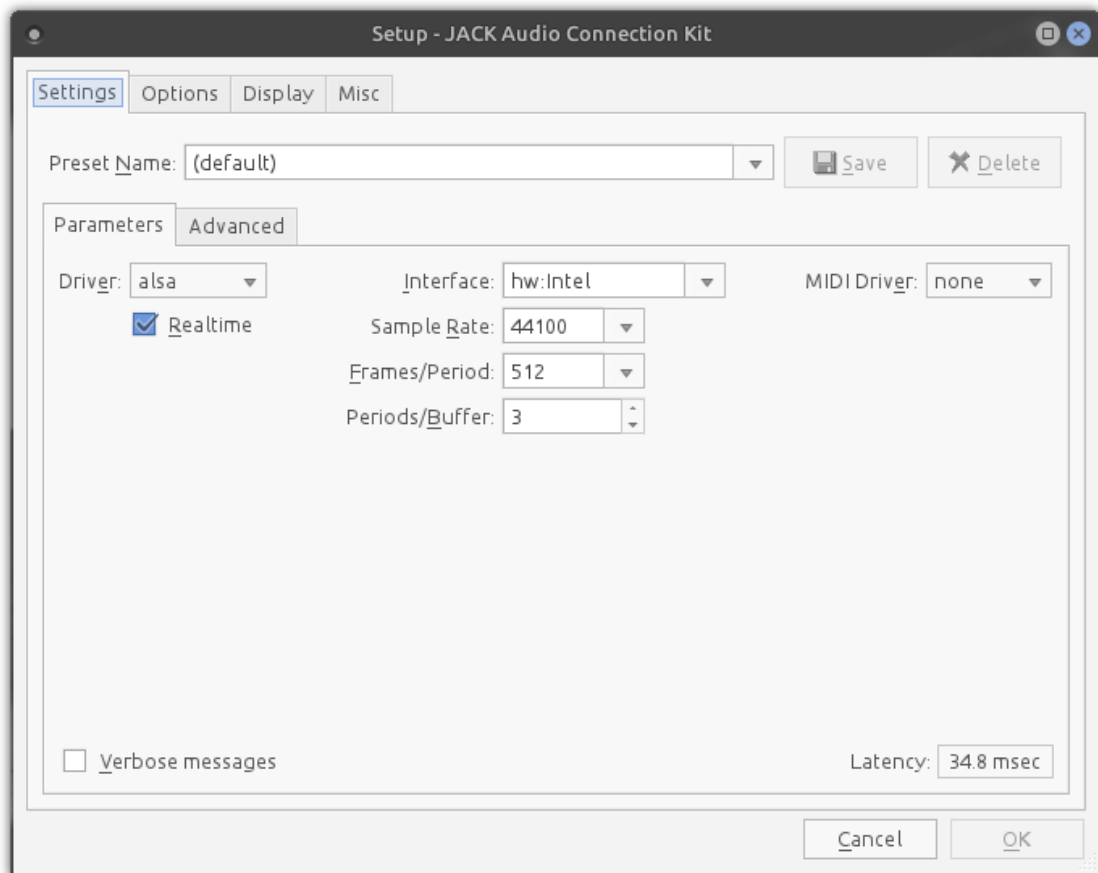
 In the 'Interface' dropdown box we will see a list of available Audio devices on the system, generally speaking whatever appears in the dropdown box should be supported to use with JACK, below is an example with multiple ALSA devices on a system to illustrate what to look for.
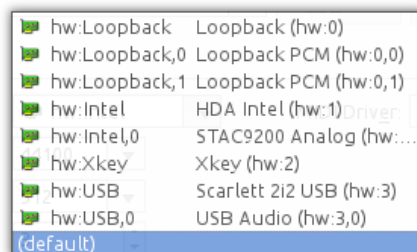
In the example picture above we can ignore the 'hw:Loopback' enties and see that the system has an internal Intel HDA Audio device at 'hw:1', A USB MIDI Keyboard Controller (Xkey) at 'hw:2' and a Scarlett 2i2 USB Audio device at 'hw:3'. Since the purpose of this dropdown box is to select an Audio (not MIDI) device we can also ignore the Xkey at 'hw:2'. So we can see that out of the displayed devices we can either choose the HDA Intel or the Scarlett 2i2 as our JACK Audio device.

Underneath the 'Interface' dropdown box we can set the 'Sample Rate' , 'Frames/Period' and 'Periods/Buffer'. By default AV Linux is pre-configured for a Sample Rate of 44100, 512 Frames/Period and 3 Periods/Buffer. Potentially these default settings will work OOTB for most users but may need to be changed depending on the performance of your hardware. The the most influencial number is 'Frames/Period'. If you are experiencing Audio dropouts or 'Xruns' you may need to increase from '512', conversely if you want to have lower 'latency' you can decrease from '512' until Xruns reappear. The 'Periods/Buffer' figure is dependent on the Audio device type, USB and Internal Audio devices generally perform best at a setting of '3' Buffers and PCI (e) Audio devices require a setting of '2'. Once you are happy with your settings you can start JACK by hitting the 'Start' button on Qjackctl.
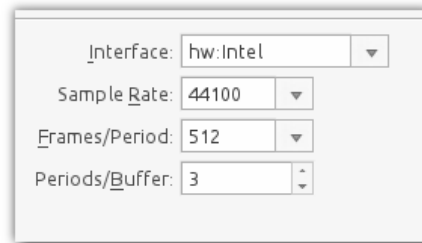
*__Added note__: 'Latency' is the delay incurred by your Audio/MIDI signals going into your Audio/MIDI device Inputs, following the signal path of both the hardware and software Applications being used and coming back out of the Audio/MIDI device's Outputs.

The 'Advanced' tab of the Setup window contains extra settings for experienced Users and in most cases these settings do not need to be changed. The ability to choose separate Output and Input devices may be of interest to people who want to use multiple Audio devices, a common scenario is using USB microphones to record and another System Audio device to play back. Another important JACK setting is the 'Timeout' value for running MIDI Plugins and editing a setting of 2000ms is recommended and set by default.
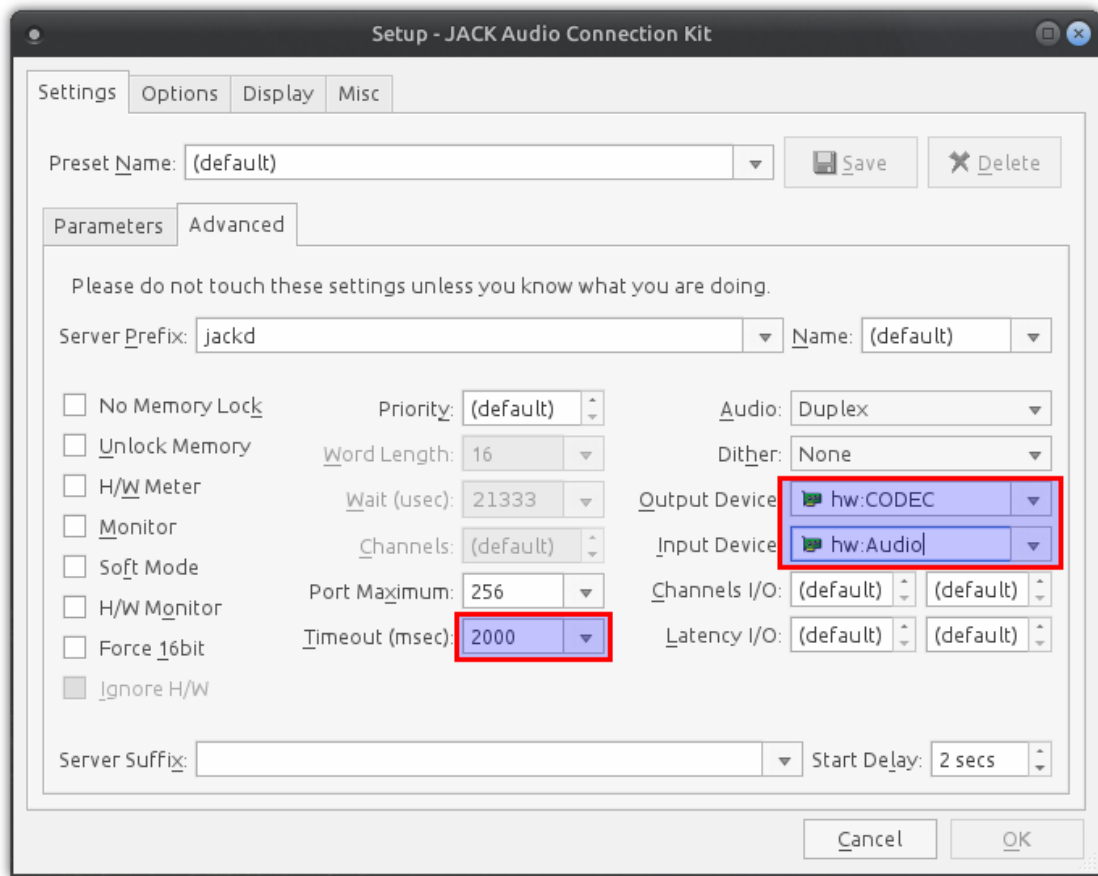
JACK's L'il Helpers, pajackconnect and a2jmidid:

 The next 'Options' tab is where we can have Qjackctl execute any optional scripts when it is started, by default AV Linux provides a '*pajackconnect'* script as one of two very important helper applications when Qjackctl starts JACK. One helper application is '*a2jmidid*'' (ALSA to JACK MIDI Daemon). In Linux there are 2 methods of handling MIDI I/O, one is at the ALSA driver level and the other is with JACK. This can be a problem if you are using an application with JACK and trying to connect a MIDI device with ALSA, the two methods cannot be used at the same time so the solution is to bridge any ALSA MIDI devices so they appear available as JACK MIDI and this is exactly what 'a2jmidid' is for, now any ALSA MIDI devices seamlessly appear to be selected as 'a2j' in Applications using JACK.

  The second helper application is a script known as '*pajackconnect*' and it's purpose is to integrate PulseAudio as a *source* and *sink* server and client routed through JACK. The ensures that applications using PulseAudio will route their Audio to the selected running Audio device you've selected in Qjackctl.
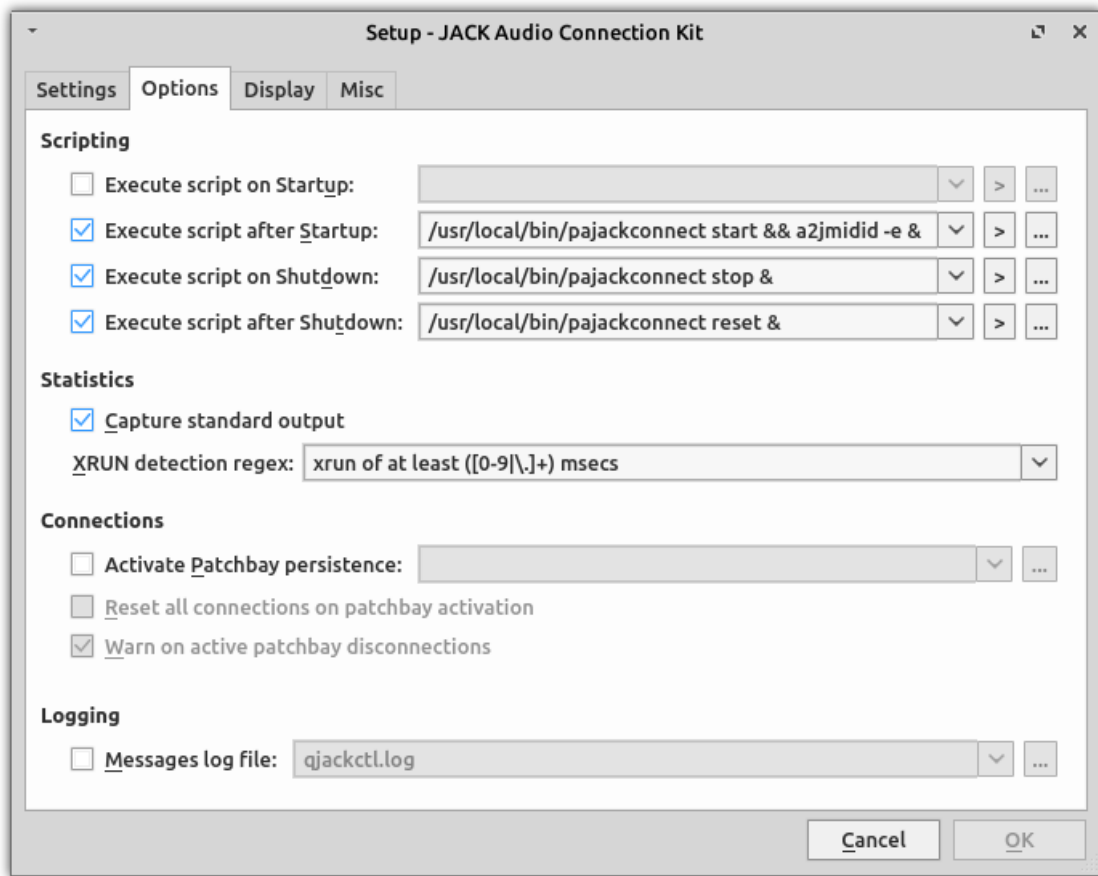
JACK Autostart and Other Settings :

   The 'Misc' tab of the Qjackctl 'Setup' window gives some other convenient options for getting JACK integrated smoothly. The '*Start JACK Audio server on application startup'* checkbox means as soon as you launch Qjackctl it will start JACK without you needing to hit the 'Start' button. You can also have Qjackctl run in the system tray on the XFCE4 panel when you close the GUI. If you have a setup where you are using one Audio device and it doesn't change you can have Qjackctl automatically start when you log in to your computer, for this to work you need to select the option to '*Start JACK Audio server on application startup'.* To have Qjackctl automatically start you need to add it to the XFCE4 'Session and Startup' list. Go to the Whisker Menu and type '*Session*' into the search bar and the top choice should be 'Session and Startup', go to the 'Application Autostart' tab and click the 'Add' button and create an entry for Qjackctl like the one shown on the next page.
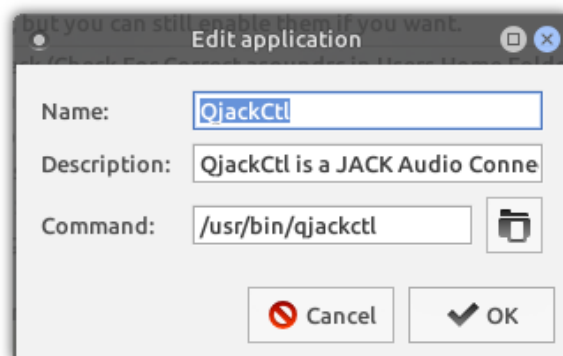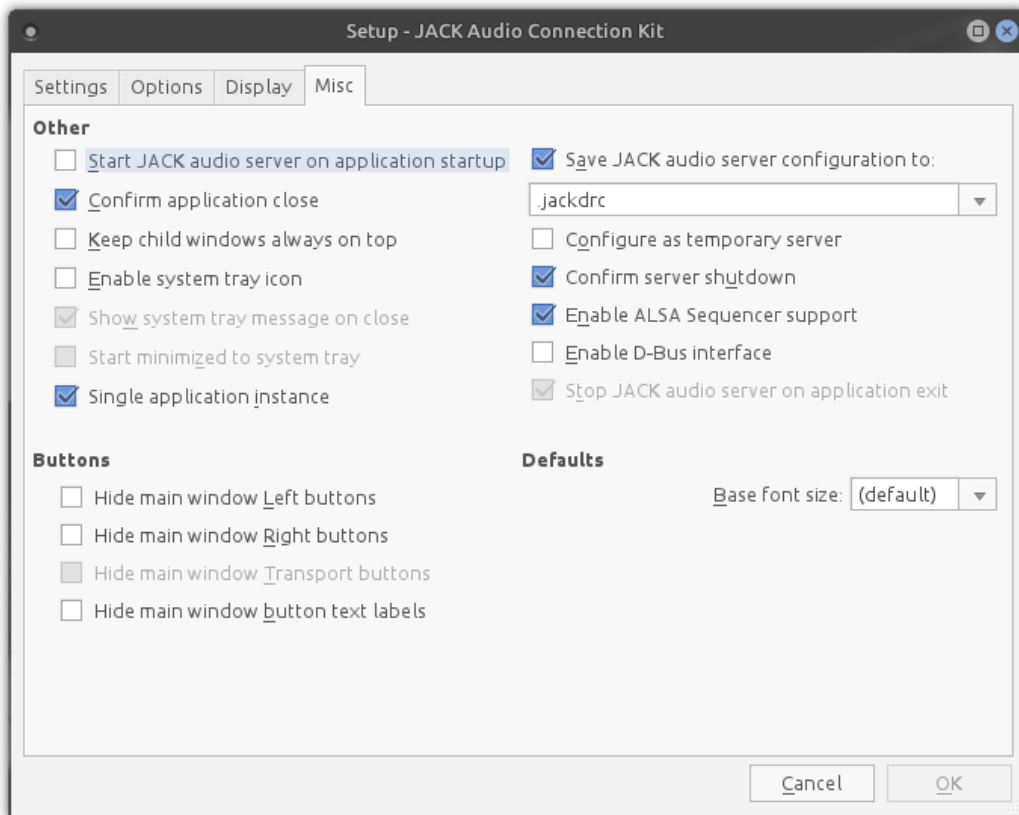
Getting Connected :

  If you are using a monolithic DAW like Ardour or Mixbus with JACK you will find that most of the connectivity you need is managed within the program itself so making manual connections is not necessary, in these cases you simply start JACK and then launch your program. However if you want to use and connect individual JACK-aware programs you may want to use the connection features of Qjackctl. It should be noted that AV Linux also comes with 'Patchage' another program to manage connections as well but for now we will stick with Qjackctl. To access the connections window click the 'Connect' button.

  There are three tabs in the 'Connections' window which pertain to three specific connection points. 'Audio' is for both hardware and software Audio (not MIDI) connections. When you are using the default AV Linux settings it is normal and expected for the 'PulseAudio JACK Sink' and 'PulseAudio JACK Source' clients to appear because PulseAudio is still active on the system. If you have a JACK-aware application running it should appear in the 'Readable or Writable Clients' panes and in most cases you will want to connect it to the 'system' playback or 'system' capture ports *not* the PulseAudio ports.
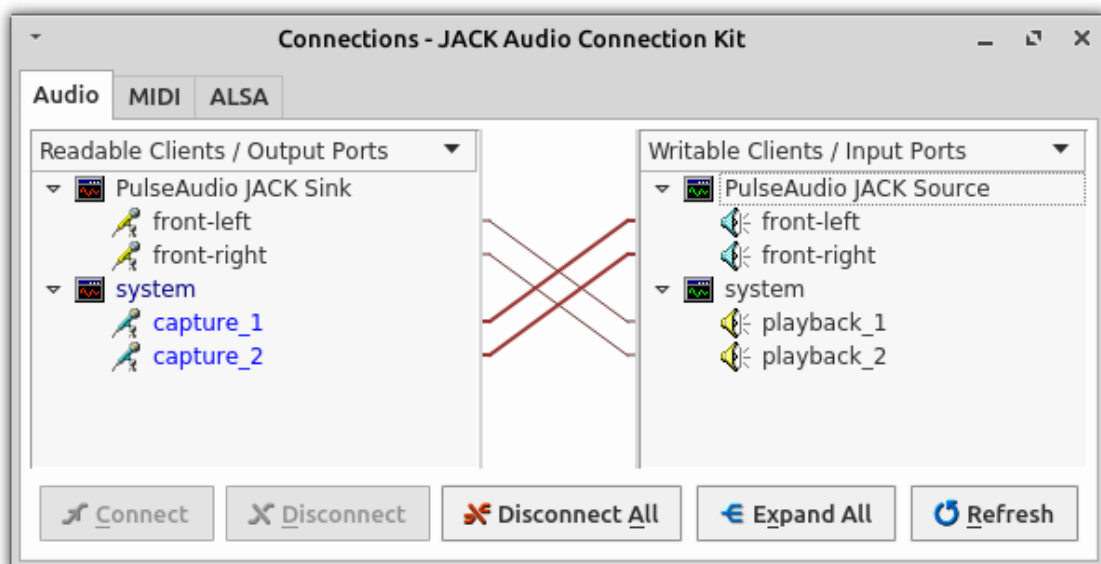
The center 'MIDI' tab is for 'JACK MIDI' connections, Because JACK MIDI is being bridged by 'a2jmidid' its ports appear under the 'a2j' Client. You can see the USB Xkey MIDI controller keyboard is visible in this tab ready to be connected to JACK-aware MIDI applications.
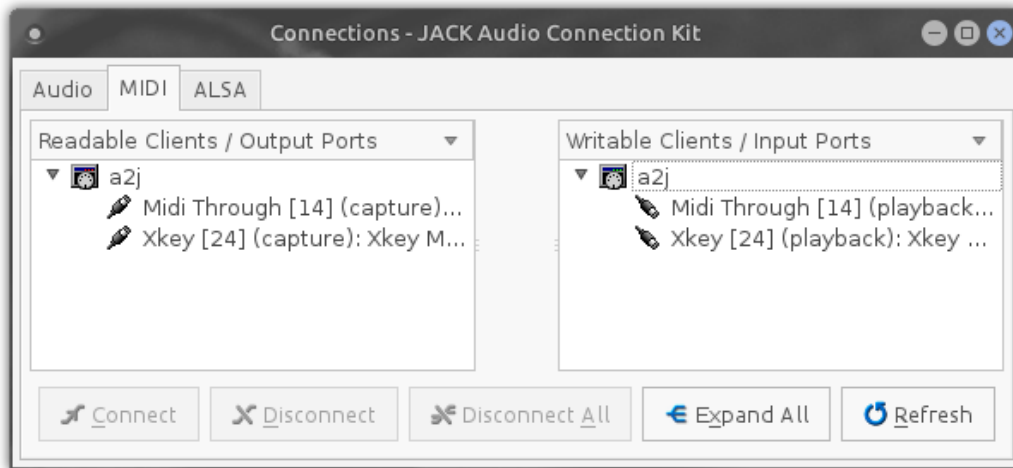
The 'ALSA' tab is for ALSA MIDI hardware and software to be connected, as you can see since the Xkey keyboard is supported by ALSA it also appears in the ALSA tab.
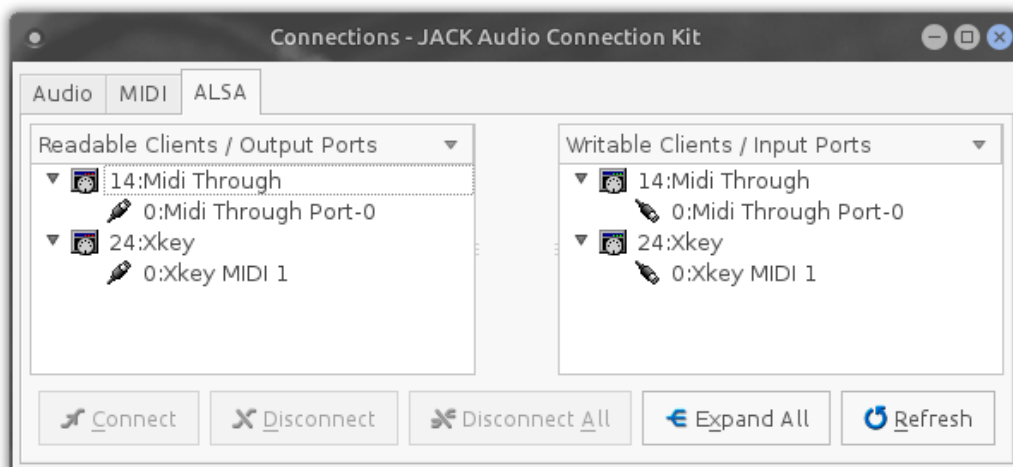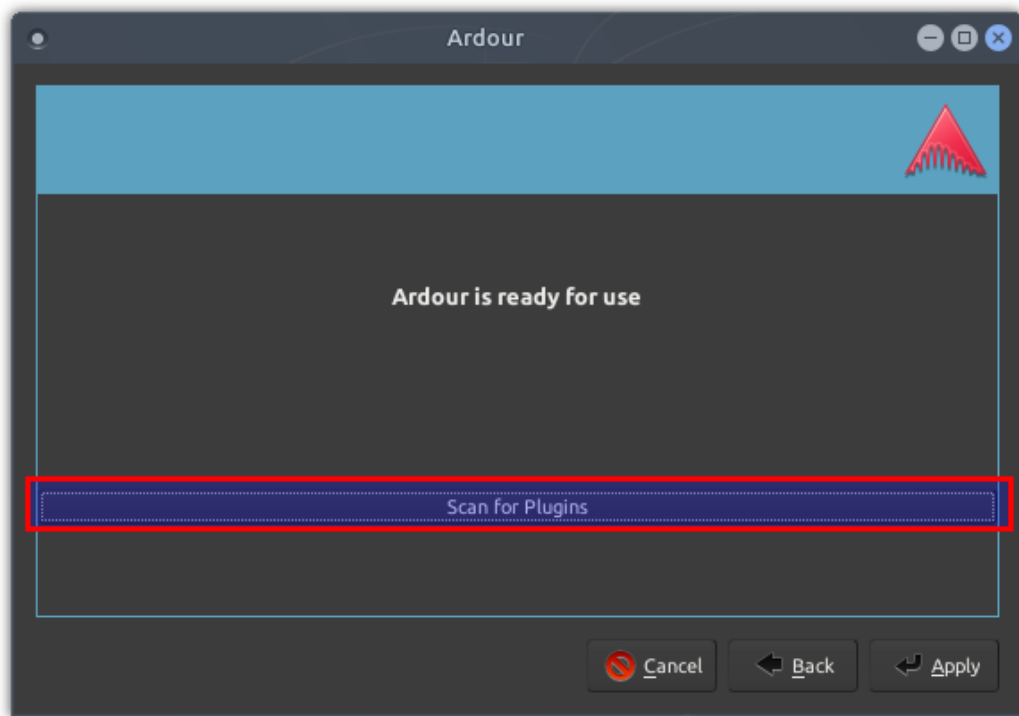
INITIAL SET UP OF ARDOUR/MIXBUS :

Choosing An Audio Backend :

  For many people new to Audio Recording with Linux it can be a bit confusing and daunting to initially set up the full-featured DAWs Ardour and Mixbus. Both powerful programs share the same core so we will use Ardour as an example. Recently Ardour and Mixbus have gained the ability to be used with either JACK or ALSA Audio 'backends'. **As stated in the previous chapter AV Linux is centered around using the JACK Audio server, because JACK has interconnectivity with other external JACK-aware programs it is recommended to prefer the JACK backend for using Ardour/Mixbus and to have JACK up and running before Ardour and Mixbus are launched.**

  When you initially launch Ardour or Mixbus either in a Live Session or a fresh HDD install you will be greeted by an initial welcome screen and walked through a few setup dialogs, in most cases except for what is detailed here you can accept the default setup choices as offered.
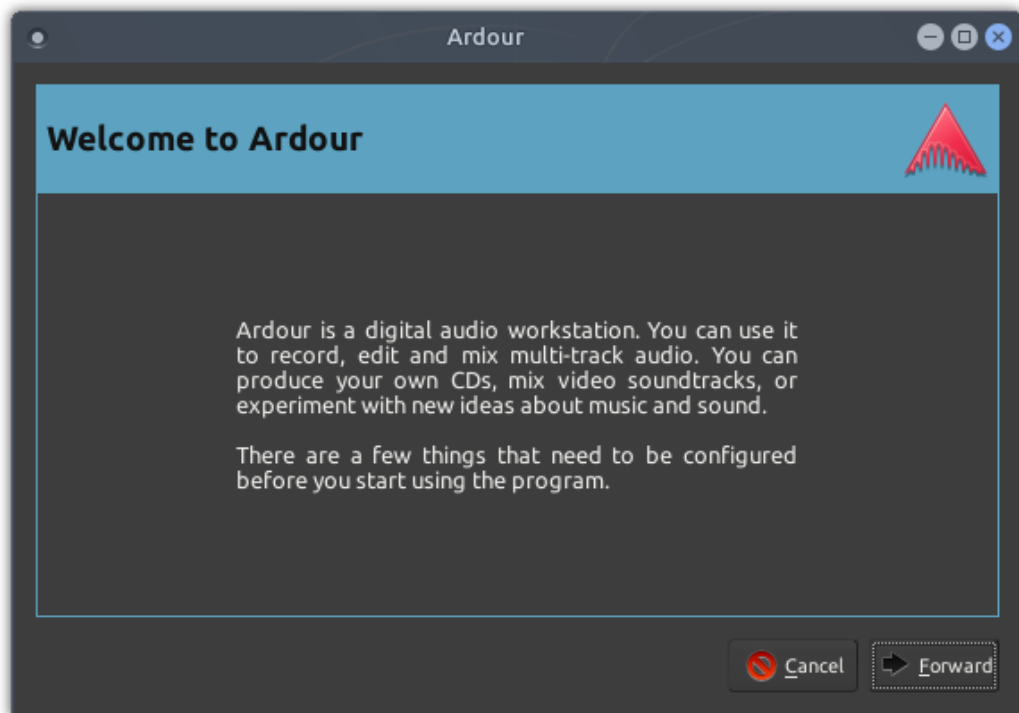
A very important part of the initial setup process is to have Ardour scan for Audio plugins, the scan doesn't take very long and doing this now will ensure that the hundreds of pre-installed Audio Plugins included in AV Linux will all be present and ready to use in your Ardour sessions. Ardour will auto-detect LADSPA and LV2 Audio plugins, LinuxVST plugins specifically will be found during the scan. It is important to note that any LinuxVST plugins you may have added yourself that aren't located in the standard paths of /usr/local/lib/vst or /usr/lib/vst will require you to to point Ardour/Mixbus to their folders by adding a custom path after launching the program, custom paths are set in 'Edit' → 'Preferences' → 'Plugins'.

*Mixbus now requires Plugins to be scanned from 'Edit' → 'Preferences' → 'Plugins' and does not offer a Scan option at first run.

When you set up your initial Ardour/Mixbus session you will be presented with an 'Audio/MIDI Setup' dialog, this is where you will choose whichever Audio backend you will use. As stated JACK is the *preferred* recommendation, ALSA can be selected but has the following caveat: No Audio/MIDI from any other external programs or applications can be routed in or out of Ardour/Mixbus. In the ALSA setup example below we are using the same USB Audio device for both Input and Output which is a suggested practice. It is possible to select a different Input than the Output, people using USB Microphones may want to use the mic with an Input and the System Audio device as an Output.
Take note that since this is a USB Audio device we set the 'Periods' dropdown selection to '3', for internal or PCI(e) Audio devices it is recommended to use a setting of '2'. If required it is also possible (but not usually necessary) to accurately calibrate Audio/MIDI latency using a cable looped from your Audio/MIDI outputs to the inputs to calculate the actual latencies. Once you have your settings selected click the 'Start' button to your upper-right.
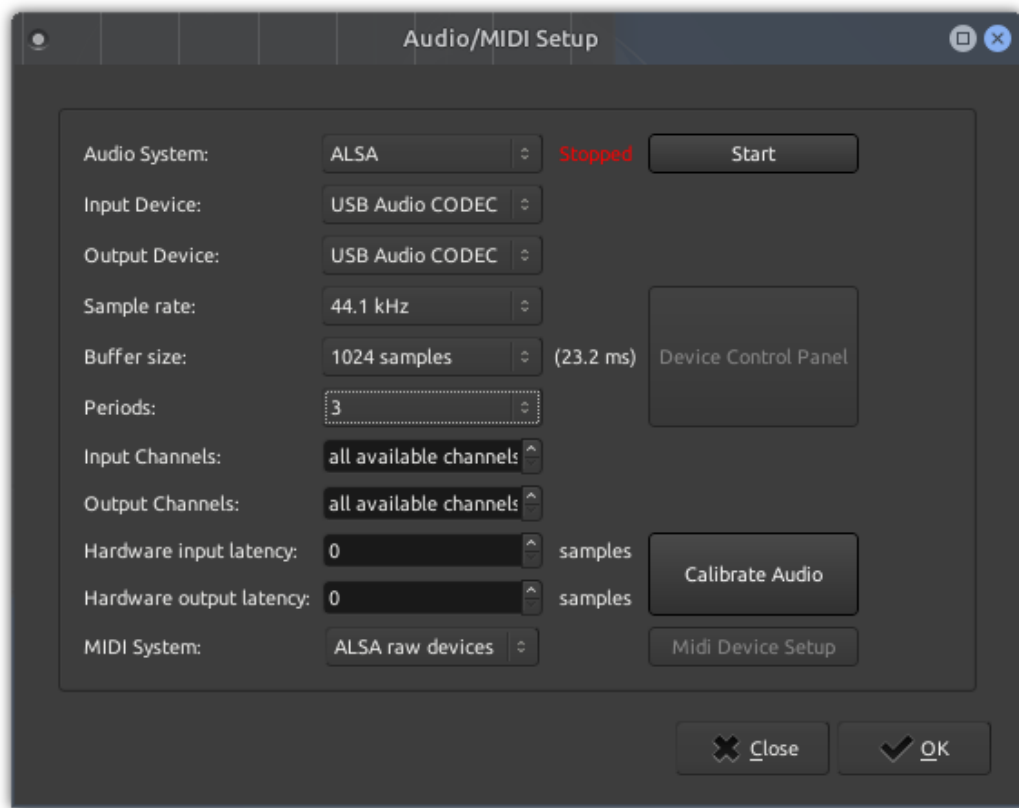
If you already have JACK up and running before setting up your intial Ardour/Mixbus session when you select the 'JACK' backend it will detect that JACK is already running and simply clicking the 'Connect to JACK' button will connect you to whatever settings and Audio device you've chosen in Qjackctl.
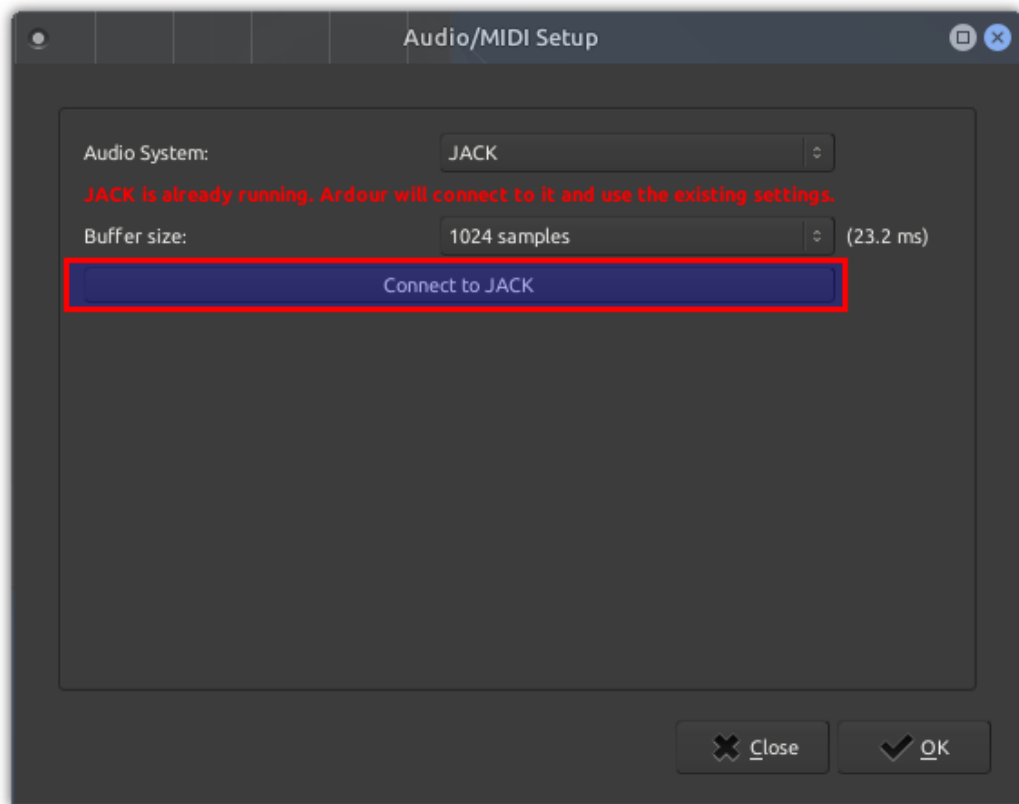
Further Info: The Ardour Reference Manual

CARLA PLUGIN HOST SETUP :

Carla... She's Quite a Gal :

  AV Linux includes 'Carla' which is a very useful host for all common Linux Audio plugin formats and comes with the optional extensions required to host Windows VST Audio plugins with Wine. Carla can be used to host plugins 'Standalone' or or also can be used as an LV2 or LinuxVST Plugin within Ardour or Mixbus to facilitate loading plugin formats that are not natively supported (ie DSSI). In addition Carla can also host Soundfont2, GIG and SFZ Sound Library formats. In order to get the best performance out of Carla it is advised to do an initial setup and scan for all Plugins available in various formats on the system. When this scan is performed with Carla running 'Standalone' it will then also ensure all Plugins scanned are available to Carla running as a Plugin when Ardour or Mixbus are used.

  If you have downloaded any outside plugins or Soundfont2, SFZ, or GIG Sound Libraries to your system you may first want to tell Carla where they are by adding the 'PATH' to Carla's scan. To do this, make sure you have JACK running and launch Carla from the Whisker Menu and select 'Configure Carla'.
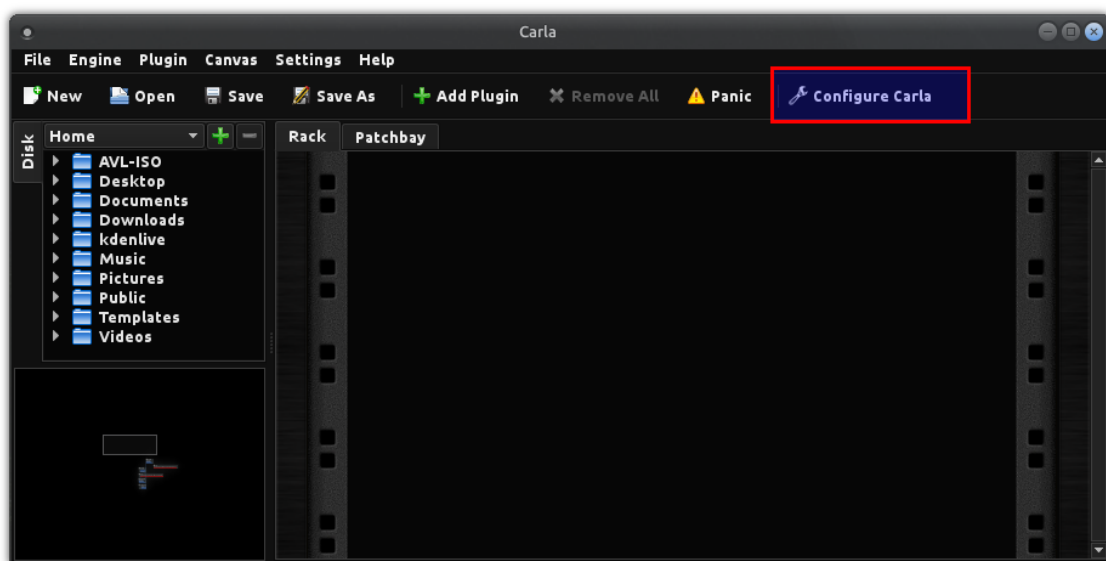
The 'Settings' window will open, select the 'Paths' Icon and from the Dropdown box select whichever type of Plugin or Sound Library you want to create a path for and if it isn't already listed use the '+Add' button to add your custom path for scanning. It is advised to restart Carla for the new paths to be taken into account.
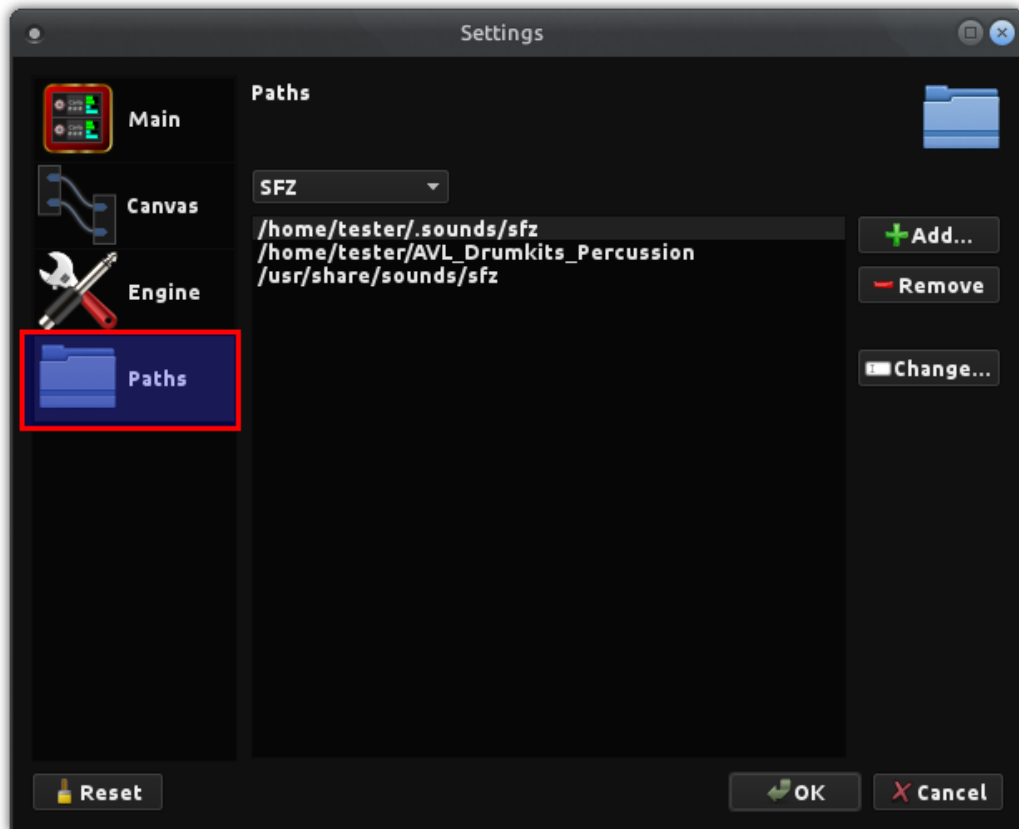
To access the scanning setup dialog click '+Add Plugin'.
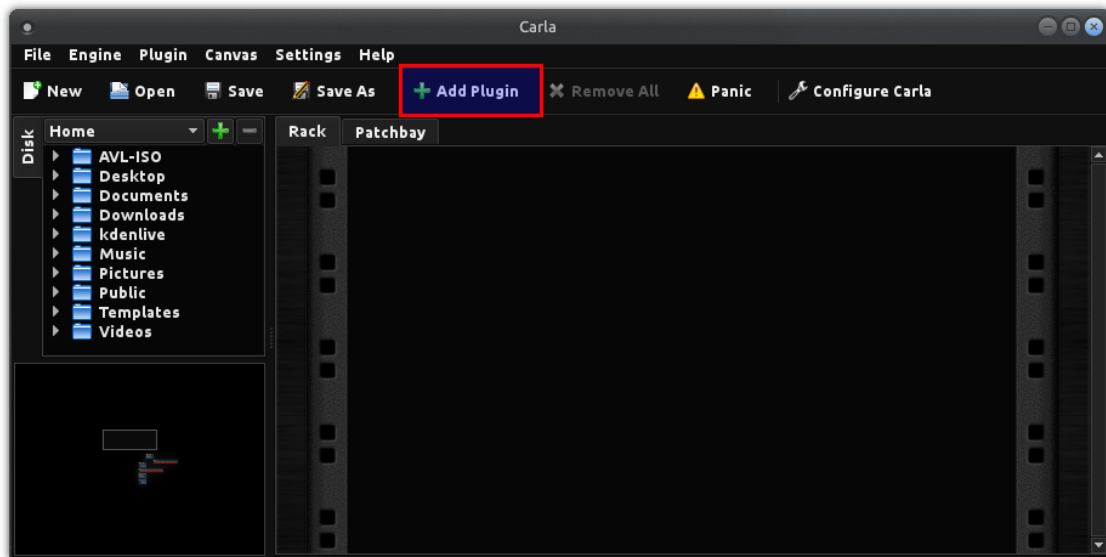
At this point Carla is showing what Plugins are already scanned and and available, to perform the actual scanning operations for new Plugins etc. that aren't available yet click the 'Refresh' button.
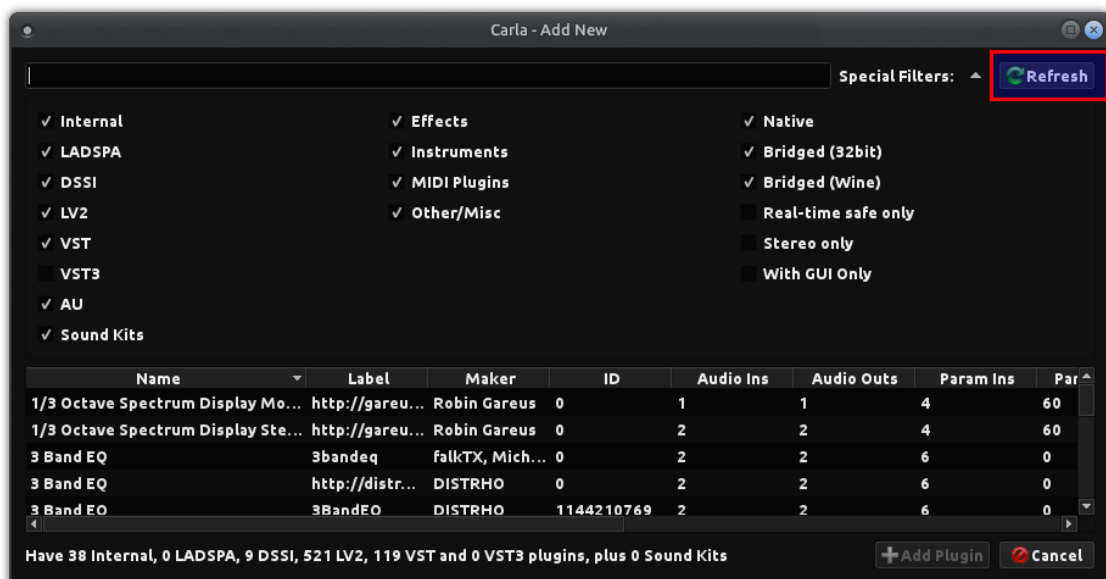
Check off as many or as few Plugin types as you would like Carla to scan for and then select the 'Start' Button to perform the actual scan. After scanning when you click the '+Add Plugin' on Carla's main window button you should be able to search and select your desired plugin by entering its name in the search field at the top of the 'Add New' window.
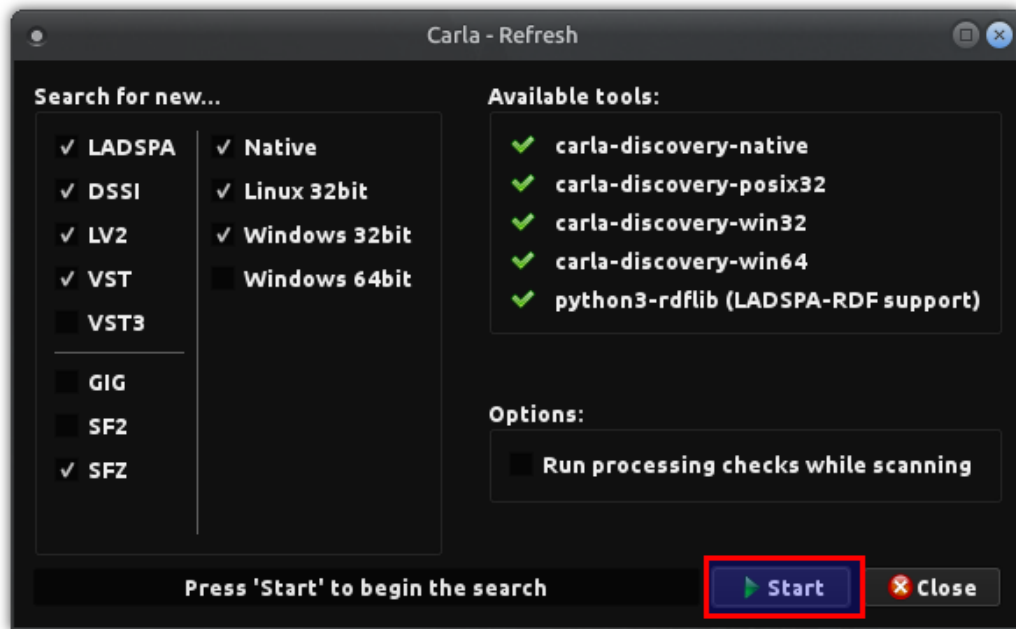
Once you have Carla set up and have the Plugins you need to use included in the list you will notice that when a Plugin is selected it appears in 'Carla Rack'. From here you can simply tweak the Plugin parameters visually in the presented Rack UI or for most LV2, LinuxVST and DSSI plugins you can click on the 'gear' Icon in the Rack slot and the plugin will show its own GUI to handle settings. In the example below ZynAddSubFX LV2 Plugin is loaded in Carla Rack. Note that LADSPA plugins and GIG, SFZ and SF2 Sound Libraries don't have GUI's and can have their settings modified by selecting the 'wrench' icon which will open a new parameters window.

Any plugins that are loaded into Carla running as a Standalone host will not automatically connect to the system Audio outputs, this differs from Carla running as a Plugin within Ardour or Mixbus in which case the connections are handled automatically within the DAW. When Carla is used by itself it provides its own Patchbay to make connections. In the example below we are using 'Carla Patchbay' to connect ZynAddSubFX to the system Audio Outputs. Carla can save both your Rack and Patchbay settings to a file to be retrieved later which makes preserving complicated setups easy.
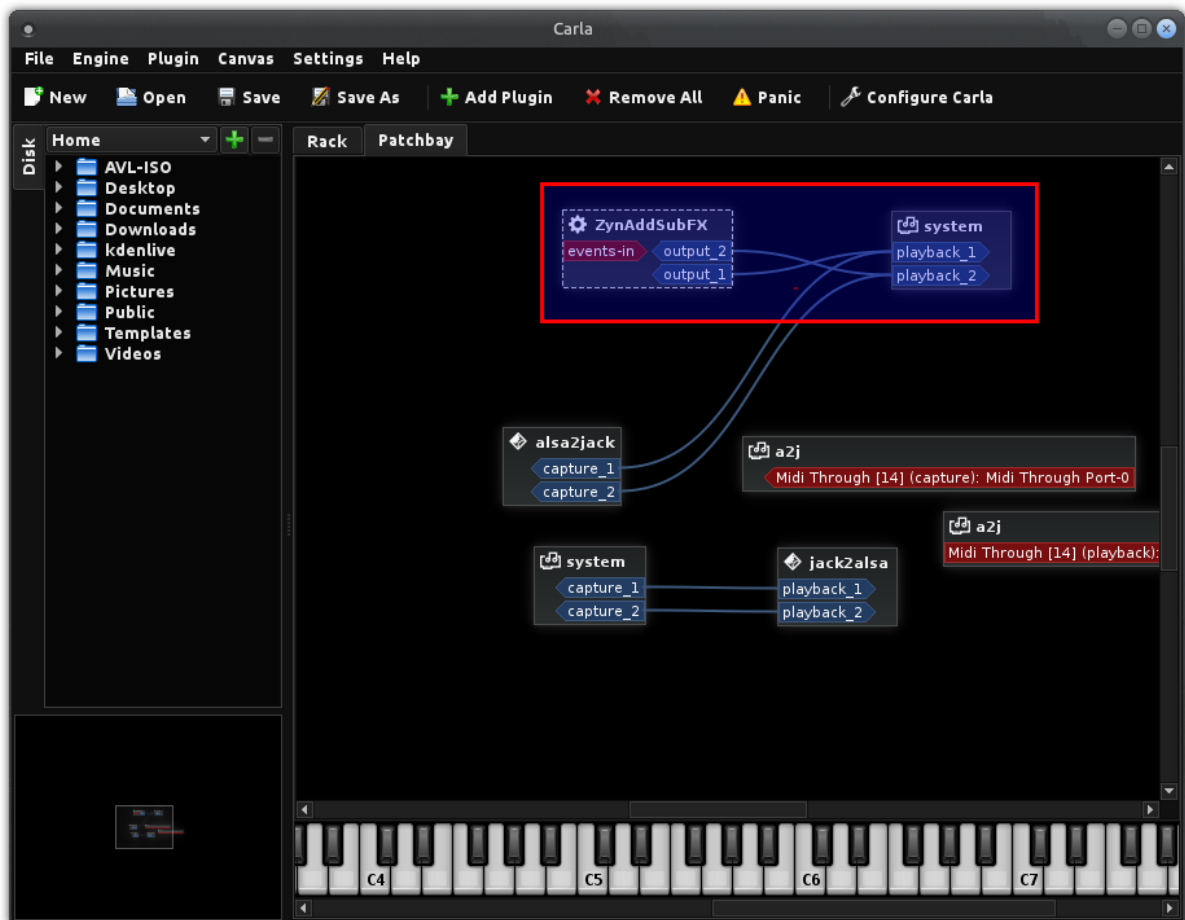
SAVING AND RESTORING JACK CONNECTIONS :

AJ-Snapshot for JACK-Aware External Application Connections :

  If you are using several JACK-Aware applications and connecting them together those unique
individual JACK connections will be lost when you you shut down the individual applications. This is a
much different scenario than when you are working within a DAW like Ardour, or a single host like Carla,
in that case the DAW remembers the connections when you save your session. What if you want to use
external JACK programs in various combinations with a complicated routing? You certainly don't want
to have to remember and manually re-connect every time you use those uniquely arranged programs.

  A very simple and effective tool to save and restore ALSA and JACK Routing is a little CLI utility called
aj-snapshot. In AV Linux I have added a very simple UI to make using aj-snapshot a little easier. With
JACK running and your applications connected simply launch 'AJ-Snapshot-UI' from the Whisker Menu,
it can be found in Multimedia → Audio Tools → AJ-Snapshot-UI. Select the option to 'Save an aj-
snapshot' and it will prompt you t enter the name of the snapshot and present a File Manager to save it
where you want. Later, with JACK running and the applications you wish to re-connect launched and
running simply open your saved Snapshot and the routing and connections will be restored.
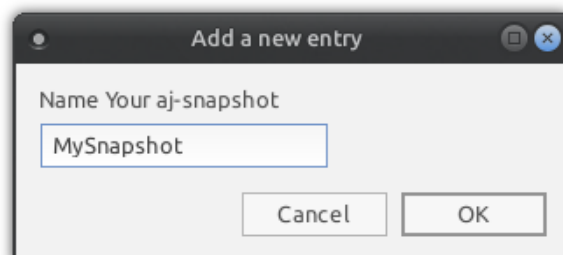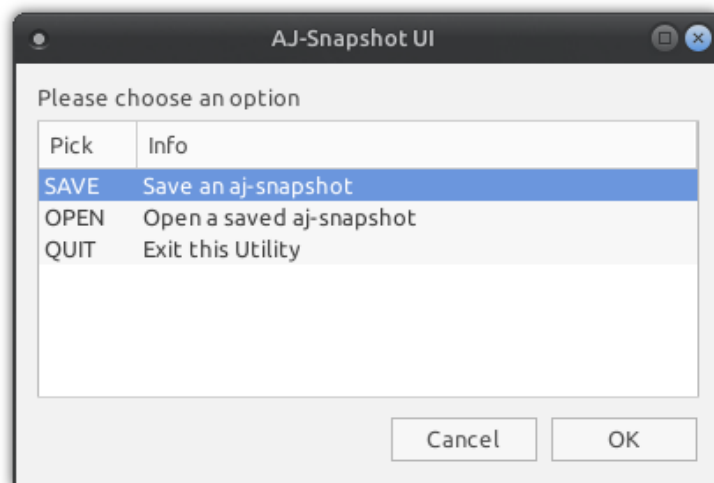
DRM WEB CONTENT :

Netflix and Other DRM Content on Linux :

  Until recently viewing Netflix and other Microsoft Silverlight DRM web browser content on a Linux computer was only possible through 3rd party applications and specially patched versions of 'WINE' (an emulation program to run Windows software on Linux). Very recently native Linux support for DRM Web content (ie Netflix) became fully provided through the Google Chrome web browser. Chrome is only available on the 64bit version of AV Linux and DRM support in the Mozilla Firefox Browser is not fully implemented yet at the time of this writing but hopefully will be soon, for this reason AV Linux 32bit which only features the Firefox Browser does not currently support viewing DRM Web Content.
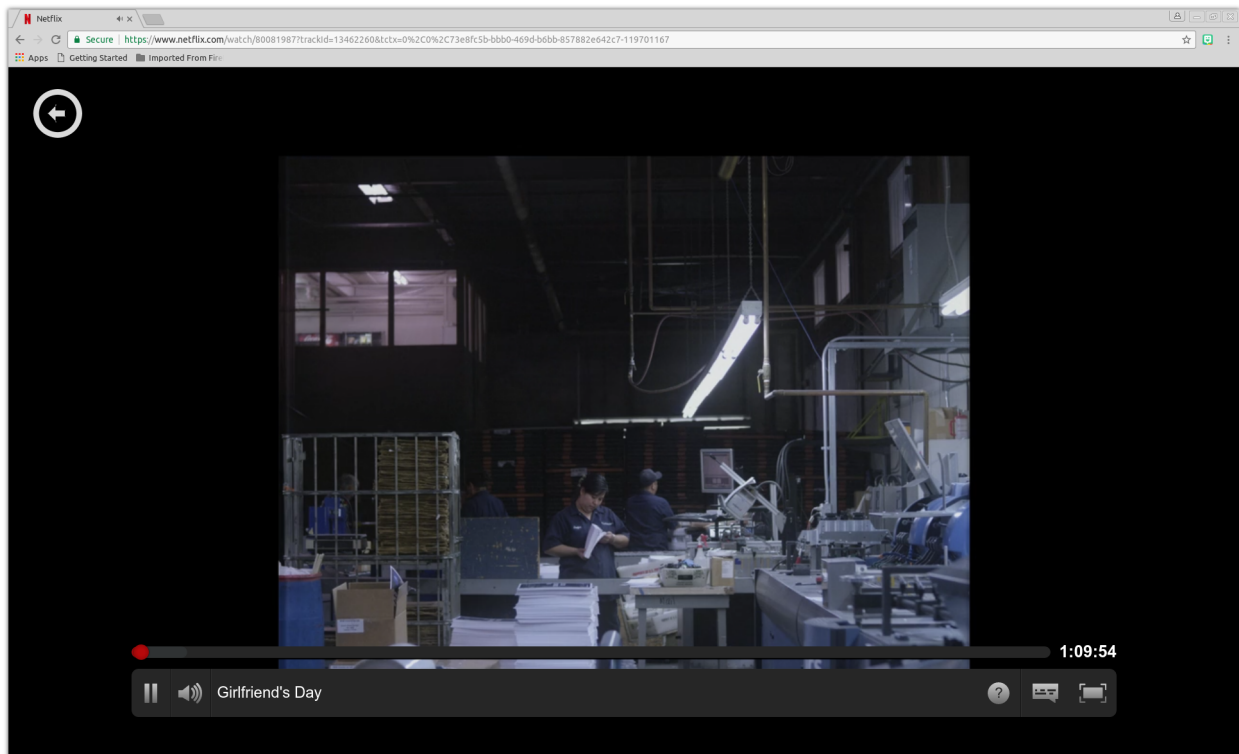
UPDATING FLASH PLUGIN :

Quick Update of Flash and Browsers:

AV Linux 2017 features a new utility to selectively update the Flash plugin and Web Browsers on the system. The rationale behind this is if during heavy production you want to avoid blanket upgrades of the system for the sake of stability but want the security of the latest Web Browser and Flash plugin you can use the '*Quick Update*' utility to upgrade only those applications.
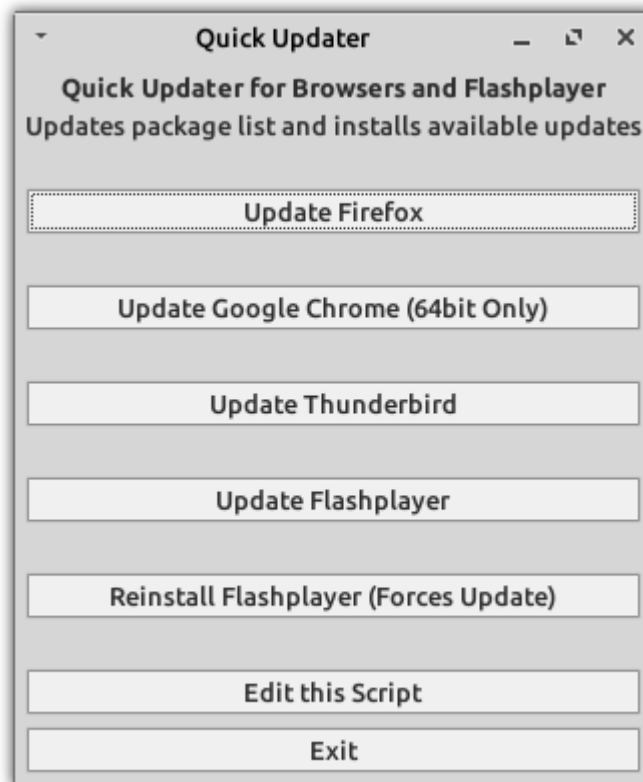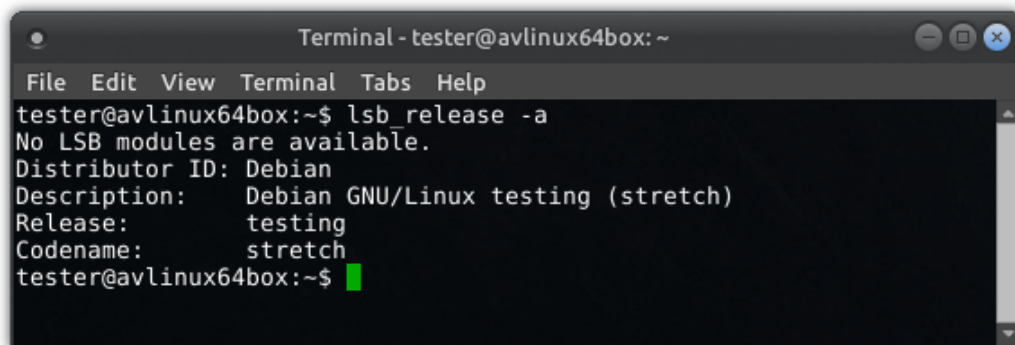
DEBIAN TESTING :

Background on the Debian Distribution Model :

 Debian/GNU Linux 'Testing' is the actual 'Distribution' that AV Linux is based on. Debian employs a multi-branch system to process the entry and inclusion of software which includes three separate branches commonly known as 'Unstable' or 'Sid' where software applications enter the Debian system, 'Testing' where software is refined and bugfixed and 'Stable' releases which are quite static in nature and provide users with a solid well-tested foundation as a finished Operating System. Previously AV Linux used the 'Stable' branch over an extended period of time which had the benefit of a very stable, solid and predictable underlying OS but required an immense amount of custom software packaging and maintenance to bring the latest and greatest applications to a static base.

Pro and Cons of Debian Testing :

 Moving to Debian Testing for AV Linux 2017 has had its share of challenges and has presented quite a learning curve. The upside of Testing is that it is positioned to have new and updated software applications enter its repositories roughly 2 weeks after being introduced into Debian 'Unstable' which means a very short wait to enjoy the latest applications and therefor requires very little extra custom packaging on my part, in addition many of the multimedia applications in AV Linux are now also provided by the KXStudio project's Repositories which also provide current multimedia packaging in a timely manner. The downside is the cyclical nature of how Testing is developed, since it is in effect a 'sandbox' to try new things and work out bugs presented by new software it bounces from bringing in very large amounts of new code to being 'frozen' with minimal updates when a new Stable release is being groomed. Influxes of new packages and big movements of multi-dependency items like Desktop Environments can occasionally leave updating Testing in a broken state for as long as a week or so, also the introduction of complete new or updated API's and ABI's often take a while to show the potential breakages they incur downstream. So in summary Testing ranges from long periods of stablity to sudden unpredictability.

```
Terminal - tester@avlinux64box: ~

File  Edit  View  Terminal  Tabs  Help
tester@avlinux64box:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Debian
Description:    Debian GNU/Linux testing (stretch)
Release:        testing
Codename:       stretch
tester@avlinux64box:~$
```

SOFTWARE UPDATES :
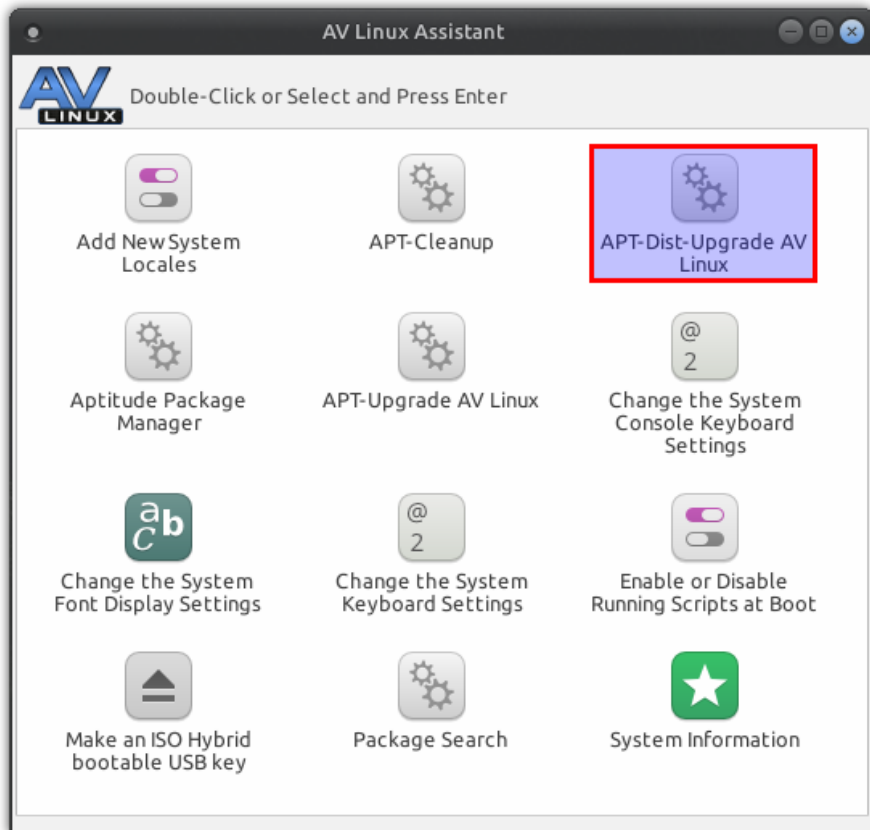*Recommended Reading!*

Balancing Updates and Risk :

  As detailed previously Debian Testing can at times be unpredictable, AV Linux is intended to provide a dedicated Audio/Video workstation OS and since it has a specific, intended use *stability* is a required trait especially when doing prolonged production work. How do we keep fresh software on the system and retain stability? There are no guarantees with Debian Testing and there is no surefire definitive answer to this question so here are two suggested ways to update your system. How to proceed will be up to the individual user and depend on the amount of experience using Debian Testing, the comfort level with risk and whether AV Linux is being used casually or for serious production.

METHOD 1: Full Dist-Upgrade – Most Risk/Newest Software :

  This is recommended for users who can live with occasional system breakage or upgrade failures, are familiar or experienced with Debian Testing and are not using AV Linux for vital production tasks..

  When running Debian Systems upstream from Debian Stable it is recommended to change how you approach and execute full System upgrades (a.k.a. "blanket upgrades"). Most users of Debian are quite familiar with the 'Synaptic Package Manager' (discussed in a later section) to install and remove software from their system. Synaptic is still a valuable tool and it can be used as a graphical help for installation, removal and to search for packages on the system but **not necessarily recommended as the best primary way to perform 'blanket upgrades'**. When using Debian Testing it is advised to perform blanket upgrades by using the 'APT' Command Line. To facilitate this, AV Linux has added a quick and easy way to perform this within the' AV Linux Assistant'. In 'Method 1' we are using 'Dist-Upgrade' which will upgrade the system and forcibly add or remove any newly required or obsolete dependencies to perform the upgrade. As an example if there is an update for 'Audacity' available and it requires the installation of some new supporting dependencies on the system that will remove and replace its existing dependencies then 'Dist-Upgrade' will update Audacity *and* install and remove whatever dependencies are necessary. This means deeper system changes (and more potential risk) than simply updating Audacity alone. To perform a Dist-Upgrade launch the AV Linux Assistant from the Whisker Menu or Panel and double-click on the 'APT-Dist-Upgrade AV Linux' launcher. This will prompt you for your Root password and then open a Terminal which will first update the Packages list and then perform the Dist-Upgrade, make sure to watch the terminal for any prompts or warnings.

METHOD 2: Safe-Upgrade – Better Stability/Reduced Update Potential :

 This is recommended for users who don't want to compromise stability, don't mind if specific application updates are held back (occurs rarely) and are in mid-production and don't want unexpected system changes or breakages.

 At a glance 'Method 2' appears quite similar to 'Method 1' however there is one key difference in how 'Safe-Upgrade' works. To return to the example above for the 'Audacity' package, a 'Safe Upgrade' would handle things differently. If Audacity is upgradeable with the current support dependencies on the system then it will simply upgrade as expected, however if new support dependencies are required that will either install new packages that aren't currently on the system or remove existing installed dependency packages then the Audacity upgrade will be held back in a 'Safe Upgrade'. It is important to note that in a multimedia-focused system like AV Linux that in most cases deep system support libraries are the types of things that get held back and 'end-user' applications and Plugins etc. are the types of packages that are less likely to get held back very often. To perform a 'Safe-Upgrade' launch the AV Linux Assistant from the Whisker Menu or Panel and double-click on the 'APT-Upgrade AV Linux' launcher. This will prompt you for your Root password and then open a Terminal which will first update the packages list and then perform the Upgrade, watch the terminal for any prompts or warnings.
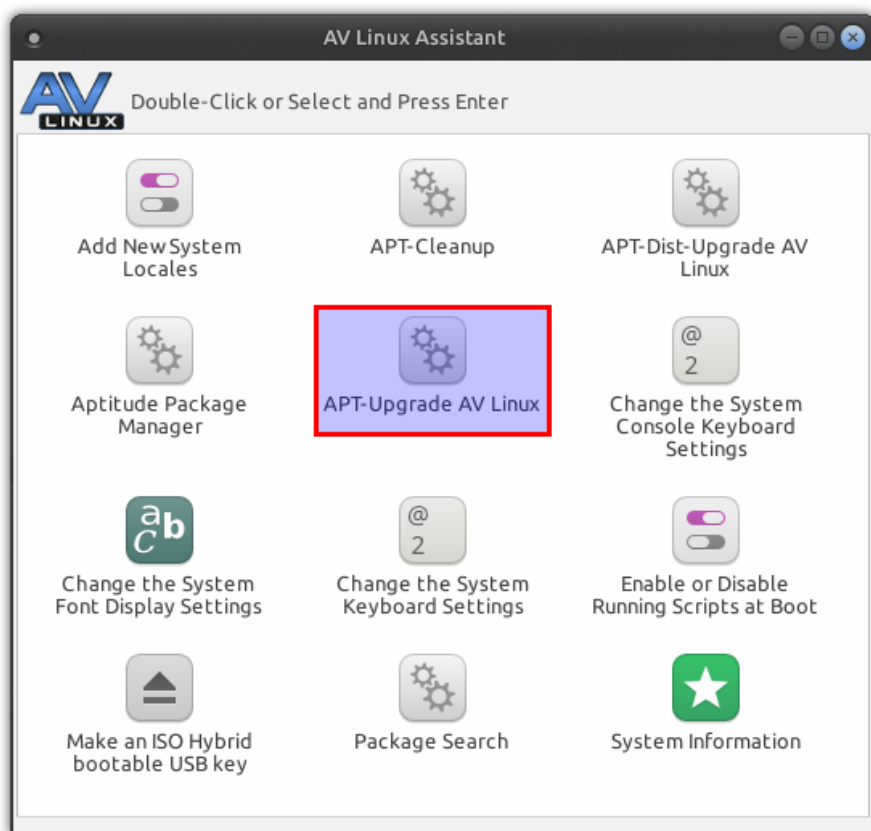
 *Added note: This section discussed basic overall system upgrading, methods of installing individual programs are covered in the next section...

SOFTWARE MANAGEMENT :

Installing Downloaded or Outside Packages :

  If you have a software 'Deb' package you have downloaded outside of the Repository system it can be installed by double-clicking on it which will automatically launch the 'Gdebi Package Installer'. **Be aware that you should only install outside packages from sources you trust.** Only one Package Management Tool can run at the same time so Gdebi will not work if you have Synaptic or Aptitude running. Gdebi will also not force the downgrade of an installed Package so if you want to roll back to an older version of an installed package you will need to manually install it in a Root Terminal using the 'dpkg' command line tool with a command like this:

`dpkg -i <PACKAGENAME>`

  If you have a folder of multiple Deb packages to install you can install them all at once by opening a Root Terminal within the folder and running dpkg with a 'wildcard' like this:

`dpkg -i *.deb`
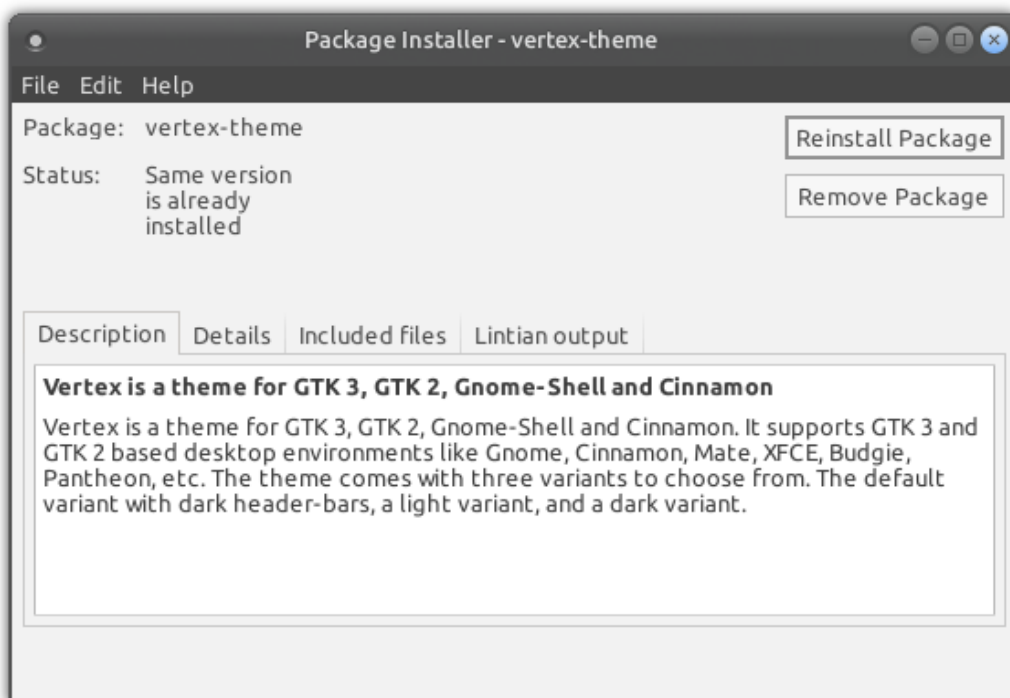
Aptitude Package Manager :

*Recommended for experienced Debian Testing or Unstable users.*

   Probably not the best first choice for new users but the Aptitude Package Manager is a powerful Package Management tool and is highly recommended by experienced Debian users for it's advanced dependency handling logic. It uses the 'Ncurses' terminal interface which can be navigated with the arrow keys on your computer keyboard. Using 'Ctrl+T' will allow you move across the menus and the up/down arrow keys will allow you to navigate, select items with the 'Enter' key. Aptitude can be used to install, remove and update Packages and perform Dist-Upgrades. Aptitude can be launched from the AV Linux Assistant where it is pre-configured to automatically update the Packages list before launching.
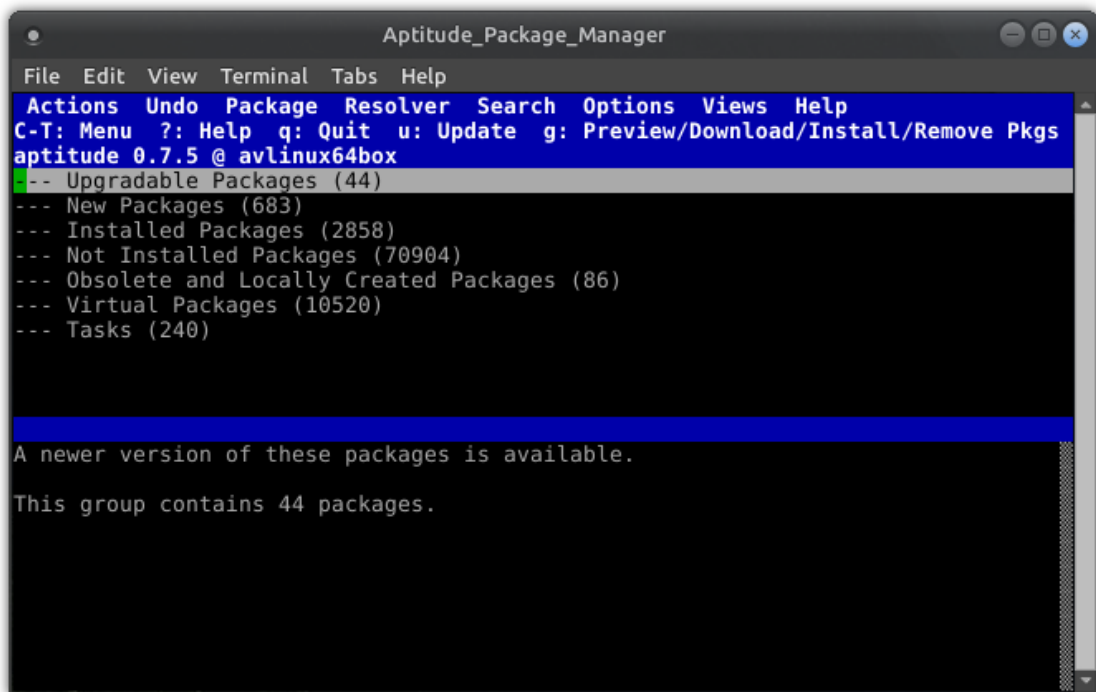
Synaptic Package Manager :

 The most familiar Graphical Package management tool to Debian Linux users is the Synaptic Package Manager. It has a well laid out and comprehensive interface to view what packages are on the system, the originating repositories they came from, and much more. It can be used for 'pinning' individual Packages to prevent them from being upgraded and has many other in-depth capabilities beyond the scope of this manual. In AV Linux it is recommended to use Synaptic to search for Packages, to install, remove, upgrade and autoremove individual packages *but not necessarily for blanket upgrades especially if you are using the 'Safe Upgrade' Method. When Safe Upgrade is used any 'held back' Packages will still be visible as 'Installed (upgradeable)' in Synaptic.* This is because Synaptic uses its own upgrade logic called 'Smart Upgrade' which is quite similar to 'Dist Upgrade' and will forcibly remove and add dependencies if requires for Package updates if you use the 'Mark All Upgrades' functionality.
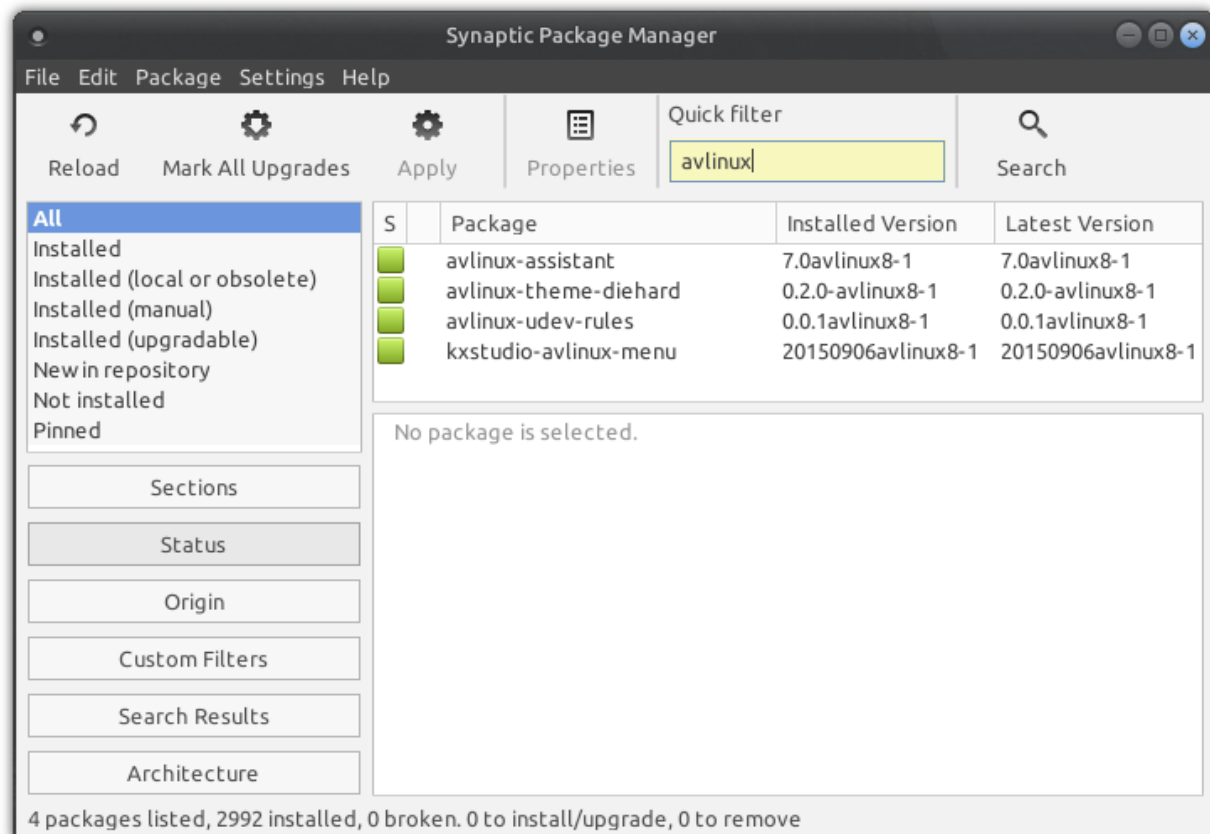
Cleaning Up After Package Operations ;

  When you refresh the Repository Package lists and upgrade, remove and install packages there are some residual cached downloaded Package and Package-list files that collect on your system. These leftover cached files can take up quite a bit of HDD space on your system over time. AV Linux comes with a program called 'Bleachbit' which can be used to clean these caches and many other system areas. In addition the AV Linux Assistant has a quick and easy launcher to clean the residual Packages and related files. Just select the 'APT-Cleanup' launcher from the AV Linux Assistant to delete these cached files.

MANAGING SOFTWARE REPOSITORIES :

Controlling What Software Is Available :

  In Linux, software is made available from online collections of software Packages called 'Repositories'. This system provides both applications and the required support dependencies to integrate and make them run properly on the system. Reputable Repositories usually provide a security key file that ensures the software is safe to download and install. AV Linux contains the Debian Testing Repositories along with various other optional 'Repos'. Of specific interest are the excellent KXStudio Repositories which provide many of the Audio Applications and Plugins.

  Enabling and disabling Repositories is another effective strategy to control the stability of your system. If you are a 'bleeding-edge' person you can leave them as-is, if you are in heavy production or happy with your current setup and don't want to be distracted by constant updates then you can simply disable selected Repositories as you wish. The selection of Repositories is handled by a 'Software & Updates' utility that can be accessed from the 'Repositories' launcher in the AV Linux Assistant.
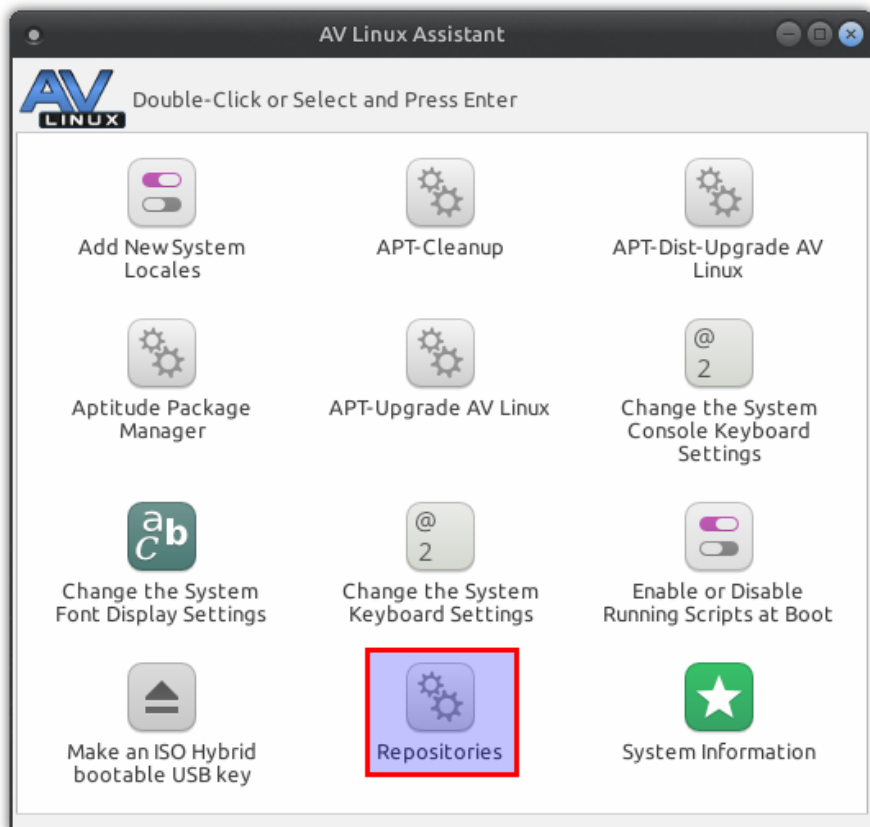
The 'Software & Updates' utility has multiple tabs, the first tab allows you to select the actual Debian Repositories you want, and the second tab is where you select other external Repositories.

PINNING AND HOLDING PACKAGES :

Preventing Unwanted Upgrades :

  It is possible to prevent unwanted upgrades of Software Packages on your System by using 'Pinning' within the Synaptic Package Manager or marking Packages to 'hold' using the 'APT' Command Line Interface. It is important to consider that pinning a Package will also prevent it's dependencies from upgrading. 'Pinning' is only recognized by the Synaptic Package Manager so if you are using 'Aptitude' or the 'APT' Command Line they will not be aware of the Packages you have pinned, if you are going to specifically use Pinning as a Package locking method it is recommended to only use Synaptic as your method of Package Management. To pin a Package in Synaptic find the the Package you want pinned and highlight it.
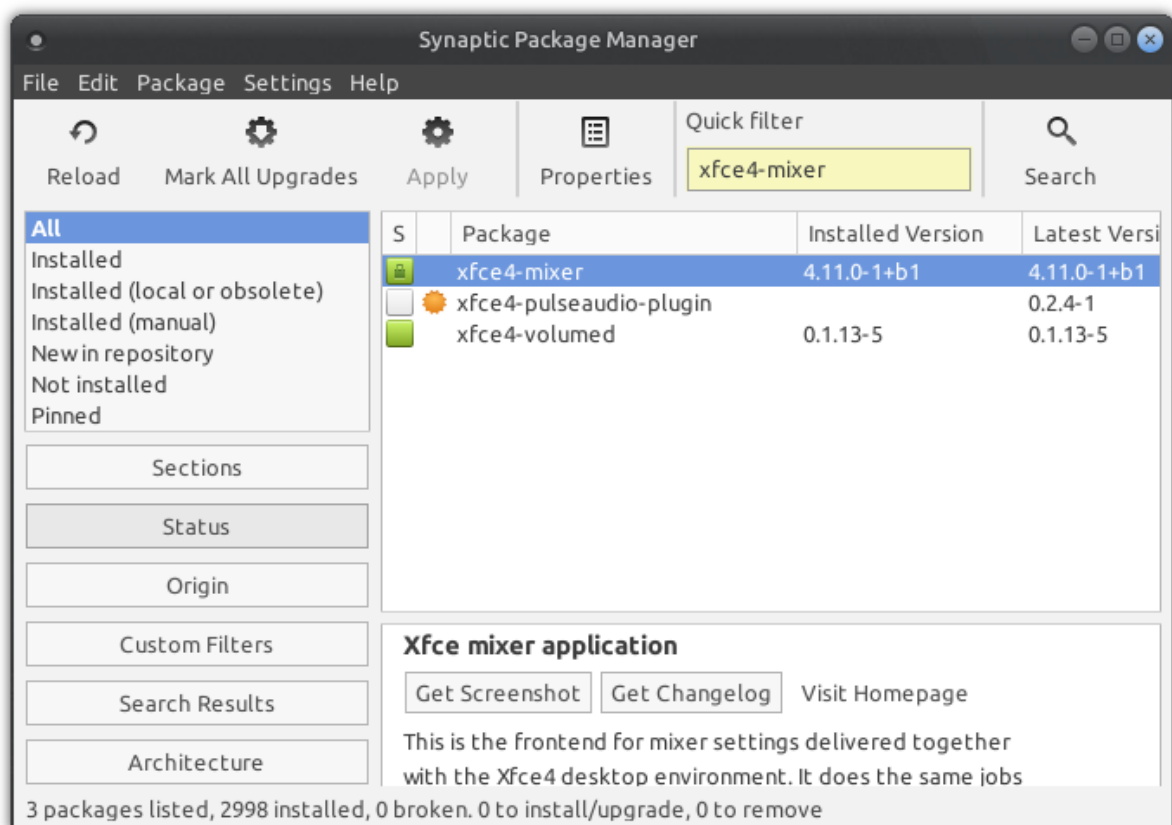
Navigate to the 'Package' menu in Synaptic and check off 'Lock Version'. The Package you have pinned will now appear with it's status as 'Pinned' in the left-hand pane of the Synaptic window when the 'Status' button is selected. To unpin simply uncheck 'Lock Version'.
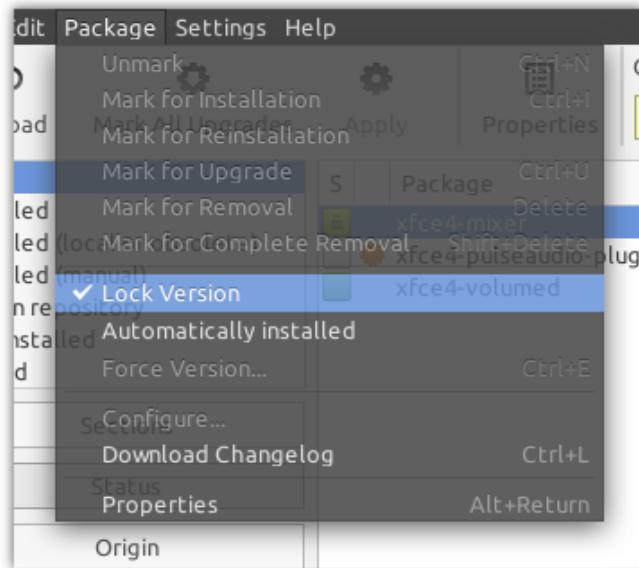
Using 'apt-mark' to Hold and Release Packages :

To use 'APT' Commands to place a Package on hold open a Root Terminal and use the following commands, note that <PACKAGENAME> requires the actual name of the Package but not the version number:

`apt-mark hold <PACKAGENAME>`

To remove the hold on a Package use this command:

`apt-mark unhold <PACKAGENAME>`

To see what packages are held on your system:

`apt-mark showhold`

To discover more useful functions of 'apt-mark' use this command:

`apt-mark --help`

Held Packages in AV Linux 2017 :

 AV Linux 2017 has some specific held packages in order to maintain compatibility with Systemback. This enables the AV ISO's to be created and also helps keep backup functionality intact. It is possible that these held packages may complicate future updates however it is not advised to to release the hold on these packages unless you are certain you do not want to use Systemback.

64bit held packages:

dosfstools extlinux genisoimage grub-common grub-pc grub-pc-bin isolinux libblkid1 libmount1 libsystemback live-boot live-boot-doc live-boot-initramfs-tools mount syslinux syslinux-common syslinux-utils systemback systemback-cli systemback-efiboot-amd64 systemback-locales systemback-scheduler

32bit held packages:

apt-mark hold dosfstools extlinux genisoimage grub-common grub-pc grub-pc-bin isolinux libblkid1 libmount1 libsystemback live-boot live-boot-doc live-boot-initramfs-tools mount syslinux syslinux-common syslinux-utils systemback systemback-cli systemback-locales systemback-scheduler
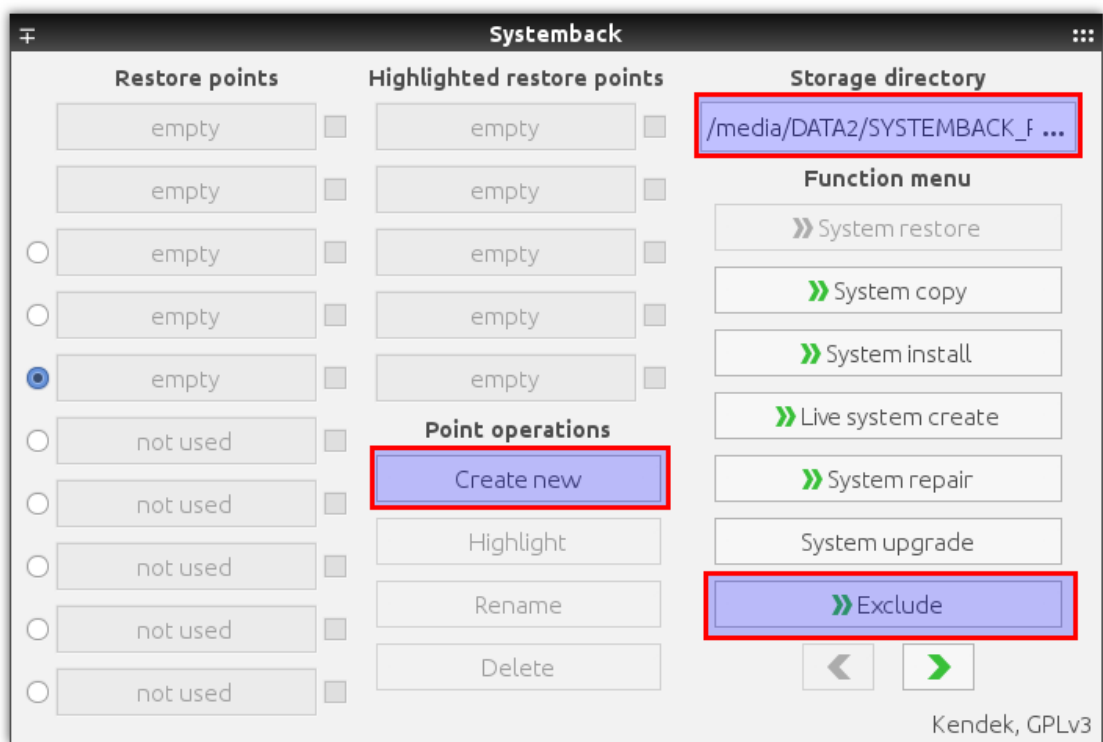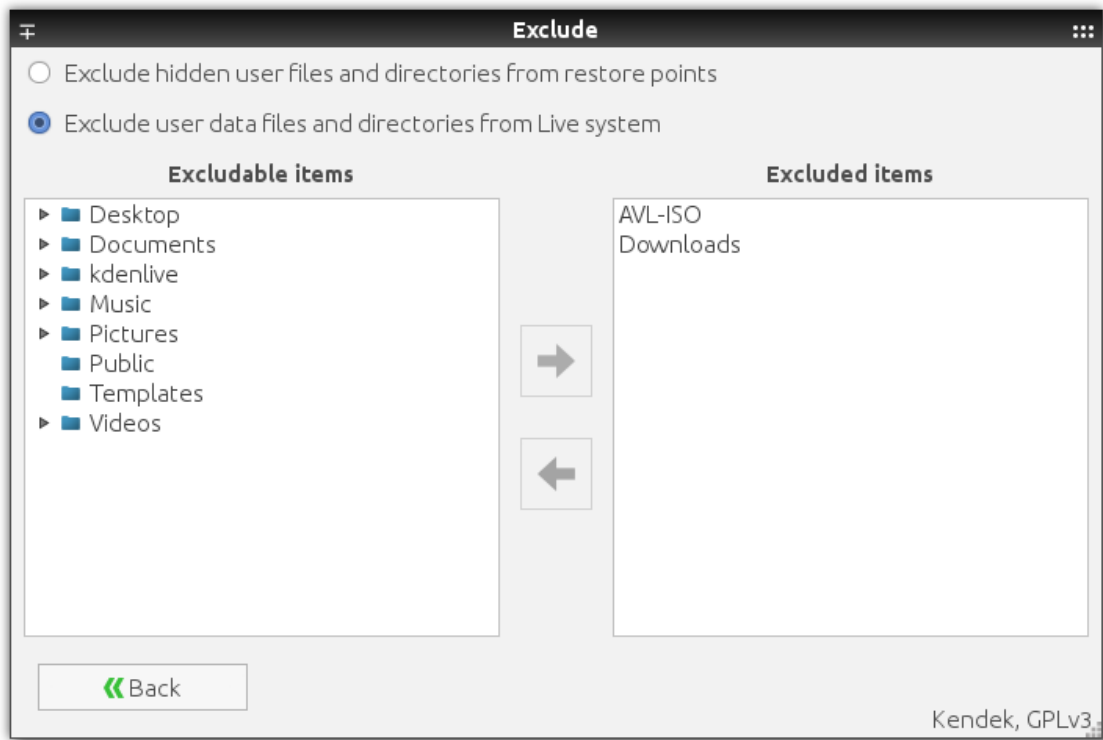
SYSTEMBACK RESTORE :

Comprehensive Backup and Restore ;

  AV Linux 2017 makes great use of Systemback for ISO creation, its Installer and probably its most useful daily function : System Restore. Systemback serves an extremely important need by helping mitigate some of the risk involved with using an 'upstream' Distribution like Debian Testing. By making regular restore points if there is some sort of system breakage incurred by faulty upgrades you can simply roll the system back to your last working restore point much like Windows computers have had as standard issue for many years. Systemback is quite easy to use and simply requires adequate writable HDD space to store its Restore data. By default Systemback will store them in the system '/home' folder, most mounted removable drives and partitions in '/media' can also be used if your User has write permissions.

  To get started launch Systemback from the Whisker Menu and enter your Root Password. The Restore window is shown below, here you can define where the Restore data goes in the 'Storage directory' field in the upper-right corner. I personally suggest not using the default of '/home' and selecting a drive or partition that AV Linux is not installed on just to keep the restore point in a safe and separate location. To set files and data that don't want to be part of the backup use the 'Exclude' feature.

It is important to note that excluded items will be excluded from both Restore points and any Live ISO's you create. Once you have completed your settings click the 'Create New' button and Systemback will take as long as it requires based on the amount of data you are backing up.
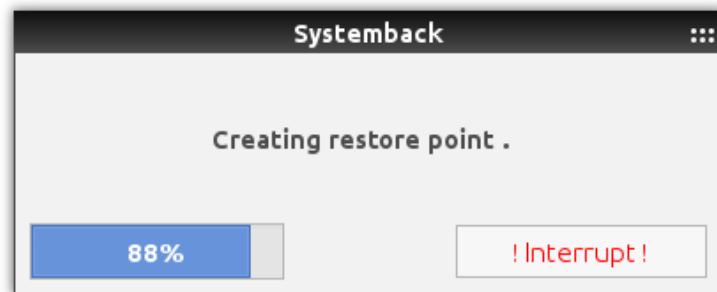
When creation of the Restore point is complete it will appear under the 'Restore points' column in the Systemback GUI. If you have created a point of some special significance (ie before an update to a complete new Desktop Environment) you can prioritize it by using the 'Highlight' button and then it will appear in the 'Highlighted restore points' column. If you need to Restore your system select the 'System restore' button and select your Restore options in the Restore window.
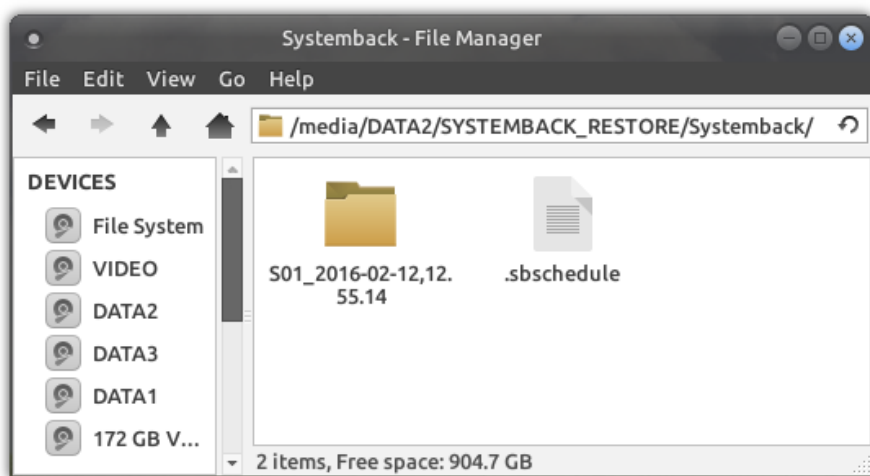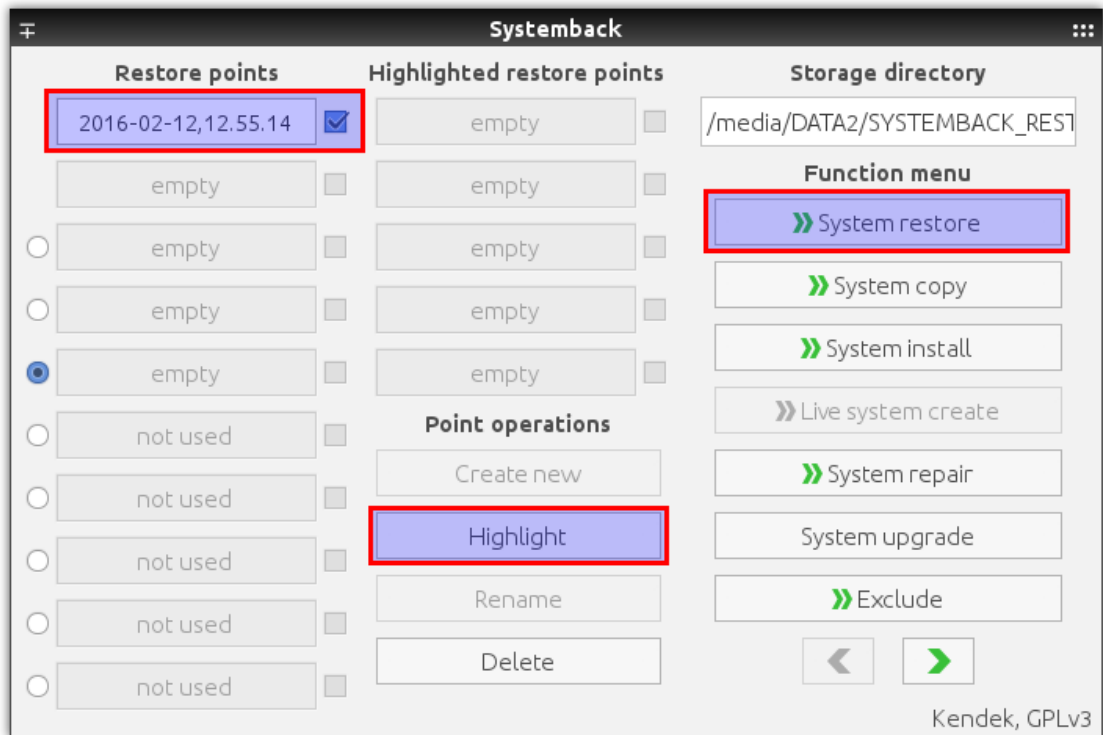
DEBIAN REFERENCE :

Many people who are new to Linux are reluctant to use the terminal and sometimes consider it to be either a primitive means of using the computer or too complicated. After some time to adjust to using it most Linux users come to appreciate the facility and speed afforded by using the CLI. The following section can be utilized by experienced Linux users and demonstrate some fundamentals to new Linux users. The apt and dpkg articles are modified from an excellent article by Matthew Danish.

Common APT usage :

apt-get install <package> Downloads <package> and all of its dependencies, and installs or upgrades them. This will also take a package off of *hold* if it was put on. See below for more info on *hold*.

apt-get remove [--purge] <package> Removes <package> and any packages that depend on it. --purge specifies that packages should be *purged*, see dpkg -P for more information.

apt-get update Updates packages listings from Debian mirrors, should be run at least once a day if you install anything that day, and every time after /etc/apt/sources.list is changed.

apt-get upgrade [-u] Upgrades all packages installed to newest versions available. Will not install new or remove old packages. If a package changes dependencies and requires installation of a new package, it will not be upgraded, it will be put on *hold* instead. apt-get upgrade will not upgrade packages put on *hold* (that is the meaning of *hold*). See below for how to manually put packages on *hold*. I suggest the `-u' option as well, because then you can see what packages are going to be upgraded.

apt-get dist-upgrade [-u] Similar to apt-get upgrade, except that *dist-upgrade* will install or remove packages to satisfy dependencies.

apt-cache search <pattern> Searches packages and descriptions for <pattern>.

apt-cache show <package> Shows the full description of <package>.

apt-cache showpkg <package> Shows a lot more detail about <package>, and its relationships to other packages.

Common Dpkg Usage :

dpkg -i <package.deb> Installs a Debian package file; one that you downloaded manually, for example.

dpkg -c <package.deb> Lists the contents of <package.deb>, a .deb file.

dpkg -I <package.deb> Extracts package information from <package.deb>, a .deb file.
dpkg -r <package> Removes an installed package named <package>

dpkg -P <package> Purges an installed package named <package>. The difference between *remove* and *purge* is that while *remove* only deletes data and executables, *purge* also deletes all configuration files in addition.

dpkg -L <package> Gives a listing of all the files installed by <package>. See also dpkg -c for checking the contents of a .deb file.

dpkg -s <package> Shows information on the installed package <package>. See also apt-cache show for viewing package information in the Debian archive and dpkg -I for viewing package information extracted from a .deb file.

dpkg-reconfigure <package> Reconfigures an installed package, if it uses *debconf* (*debconf* provides that consistent configuration interface for package installation). You can reconfigure *debconf* itself if you want to change the front-end or priority of questions asked. For example, to reconfigure *debconf* with the dialog front-end, you simply run:

dpkg-reconfigure --frontend=dialog debconf

echo ``<package> hold" | dpkg --set-selections Put <package> on *hold* (command line method)

dpkg --get-selections ``<package>" Get the current status of <package> (command line method)

dpkg -S <file> Searches for <file> in package database, telling you which packages have that file in them.

Building Debian Packages from Source:

apt-get source [-b] <package> Download the source Debian package for <package> and extract it. You must have deb-src lines in your /etc/apt/sources.list for this to work. If you supply the `-b' option and you are currently root, then the package will be automatically built if possible.

apt-get build-dep <package> Download and install the packages necessary to build the source Debian package <package>. This feature is only present in apt version 0.5 and up. Currently this means that woody and above contain this functionality. If you have an older version of apt then the easiest way to find out the build dependencies is to look in the debian/control file in the source package directory. A common usage of this command is in conjunction with apt-get source -b. For example (as root):

apt-get build-dep <package>
apt-get source -b <package>

Will download the source package, all of its build dependencies, and attempt to compile the source package.

dpkg-source -x <package.dsc> If you have downloaded the source package for a program manually, which includes several files such as a .orig.tar.gz (or .tar.gz if it is Debian native), a .dsc, and a .diff.gz (if it is not Debian native), then you can unpack the source package using this command on the .dsc file.

dpkg-buildpackage Builds a Debian package from a Debian source tree. You must be in the main directory of the source tree for this to work. Sample usage:

dpkg-buildpackage -rfakeroot -uc -b

Where `-rfakeroot' instructs it to use the fakeroot program to simulate root privileges (for ownership purposes), `-uc' stands for ``Don't cryptographically sign the changelog'', and `-b' stands for ``Build the binary package only''

debuild A handy wrapper script around dpkg-buildpackage that will automatically take care of using fakeroot or not, as well as running lintian and gpg for you.

Fixing dependencies :

dpkg --configure --pending If dpkg quits with an error while apt-get install, upgrade, or dist-upgrading try running this to configure the packages that were already unpacked. Then try apt-get install, upgrade, or dist-upgrade -f, and then try apt-get install, upgrade, or dist-upgrade again. Repeat as needed. This usually resolves most dependency problems (also, if it mentions a specific package for some reason, you might want to try installing or removing that package)

apt-get install -f
apt-get upgrade -f
apt-get dist-upgrade -f

Attempt to fix dependencies while doing one of the above. Note that apt-get install -f does not require a <package> argument.

HANDY TERMINAL COMMANDS :

$_

SYMLINK TO A FILE OR FOLDER ELSEWHERE ON SYSTEM (AS ROOT) :
ln -s /path/file/exists /path/link/placed

REBUILD FONT CACHE (AS ROOT) :
fc-cache -f -v

FONT ANTIALIASING AND HINTING SETUP:
dpkg-reconfigure fontconfig-config

DVD RIPPING WITH MPLAYER :
*Creates a file called "dump.vob" in your home folder assuming movie is title 1.
mplayer dvd://1 -dumpstream -dumpfile dump.vob

VIRTUALBOX MODULE SETUP (AS ROOT) :
*Make sure linux-headers are installed and linked
/etc/init.d/vboxdrv setup

PRINT A 'selections.txt' FILE TO DESIGNATE HELD PACKAGES (AS ROOT)
dpkg --get-selections * > selections.txt
*Then edit selections.txt as Root

SAVE THE 'selections.txt' FILE TO DPKG (AS ROOT)
dpkg --set-selections < selections.txt

BUILD DEBIAN PACKAGE WITH GIT-BUILDPACKAGE:
git-buildpackage --git-upstream-branch=origin/upstream

BUILD DEBIAN PACKAGE WITH DPKG:
dpkg-buildpackage -rfakeroot -uc -b

LINKING KERNEL HEADERS FOR MODULE BUILDING:
ln -s /usr/src/linux-headers-$(uname -r) /lib/modules/$(uname -r)/build

SET A DIFFERENT C COMPILER THAN DEFAULT
*Replace X.X with desired gcc version
export CC=/usr/bin/gcc-X.X

APPLY A PATCH TO SOURCE CODE OR KERNEL
patch -p1 < PATCHFILENAME

CHECK SOUND DEVICES INSTALLED ON SYSTEM
cat /proc/asound/cards

CHECK AUDIO PLAYBACK DEVICES ON SYSTEM
aplay -l

THANKS AND ACKNOWLEDGEMENTS :