WeakNet Linux WEAKERTHAN

System Administration Manual

Version 4.1k

Written by Trevelyn (Douglas@WeakNetLabs.com) June 8th, 2010.

Index:

How to Use this Guide:

This is a free reference guide. You can use this guide if you are not familiar with FluxBox, the Instant Server technology used in WNLAv4, or Terminal. This guide does not cover the how to use the security tools available in *WeakNet Linux WEAKERTHAN*, unless software was developed, or modified, by the author. This document WILL be made available without warranty, or without promised services, paid for by donations or not. Any references are available at the end of this document, for further reading.

All commands will be in a `monospace` font, without the bash prompt displayed.
All filenames, including directories (folders), will be displayed in `Courier New` font.
All subsections will be this color.

The password is "weaknet" (without quotes) for everything.

This guide assumes you have a fresh live environment, either via USB, CDROM, or a fresh installation of *WeakNet Linux WEAKERTHAN*.

You can get The Kernel Headers from the WeakNet Linux WEAKERTHAN Applications page:
http://weaknetlabs.com/main/?page_id=251

Forward:

WeakNet Linux was a small project I was mulling over for about a year when I first started writing software under the name WeakNet Labs. The first two versions (1 and 2) were simply to provide people in the Lab with a live environment that included all software I or anyone else under the name WeakNet Labs had written. This included all dependencies and added in all the hacking tools we were using at the time for network security auditing and computer penetration testing in the Lab.

I was given the idea to create a "Lite" version of WeakNet Linux from Tully, and released version three *exclusively* as "Lite." This meant that I had to cram as much stuff down to the size of just under 700MB to fit on a CD. This was no easy task, and helped me in the understanding of how Operating Systems worked and how wonderfully simple and beautiful UNIX based OS's are. The UNIX operating system is extremely text file based. This gives light to the phrase "Perl is powerful." Perl is an amazing text parser / manipulation programming language that even fits the bill for networking applications. Perl is what I used for ALL of my applications. The GUI applications were coded using the Perl::Tk module, and are, for the most part, just text manipulation applications.

When I started thinking about releasing a version 4 of WeakNet Linux, I was fed up with a few things. One of which I regretfully left in the kernel – Nouveau. I decided to create my own kernel based off of the Ubuntu 9.04 generic kernel. In this light, I removed a lot of things and added a lot as well, including debugging for hardware devices in 802.11 I felt since this was my first attempt at creating a Linux, that I should imaginatively name it "*WEAKERTHAN*." This turned out to be a charm, as my end-users seemed to have many problems, mostly with hardware. A lot of legacy hardware devices were not supported in the kernel I decided to use when creating *WEAKERTHAN*. I decided to stick with FluxBox as the default desktop manager, which made things even more complex for the Linux beginner. *WEAKERTHAN* is not for beginners, at all.

Supporting the OS, up to this point, has been rather painless and I have, without slacking, provided my %100 focus on support via Email in the best of my ability. This isn't easy to do for someone who wants a tiny social life as well :P

I hope, that in creating this document, I can resolve any issues a beginner may have at using WNLAv4. Please keep in mind, that this OS is still only a hack. It is a customized version of the ever popular Ubuntu 9.04 and the community at Ubuntu is very diligent in resolving issues with their handy little OS. If I cannot provide you with a solid answer, you can always ask them too. Who knows, maybe one day you will end up a tech support guy/girl yourself in the meritocracy of Ubuntu!

Thank you for giving the OS a test or a second try. Please keep in mind, that I am only one guy. I try my best to help. I try my best to support what I create. And I teach those who are passionate enough to listen, anything I know. Most importantly, I just wanna have fun.

~Trevelyn – a.k.a. Douglas Berdeaux – Douglas@WeakNetLabs.com WeakNetLabs@Gmail.com

Part 1. FluxBox

1.0 Menus and Tabs

FluxBox Menu's are easy to access, you simply right click anywhere on the desktop.

Figure 1.0: The Fluxbox menu.

To create a stack of tabbed applications, you simply click the title bar of an application with both right and left mouse buttons, (or middle mouse button) and drag the window over another dropping it on it. This will produce a column of tabs on the left side of the window. Each tab has a title, and corresponds to the application it is for.

Here is a screen shot of tabs. Both are for Eterm windows, and you can see them on the left side.

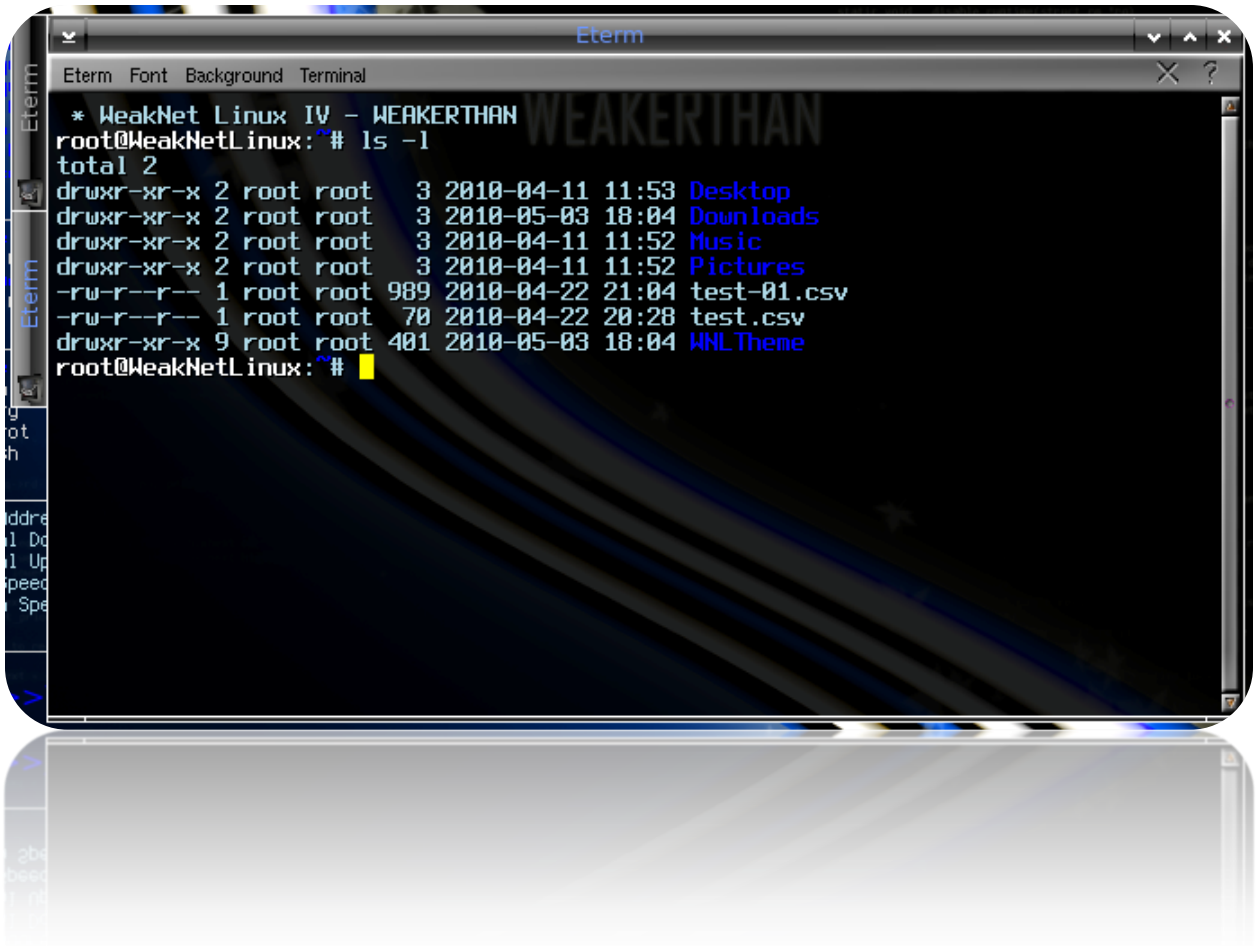


Figure 1.1: The FluxBox Tabs.

1.1 – Editing Menus

Editing the FluxBox menu is quite simple. You can use a text editor such as `vi` or `leafpad` to open the file `~/.fluxbox/menu` from here you can follow my syntax to add submenu's or add extra applications, quite easily. Just don't forget to close the submenu, or menus with the **[end]** line:

```
leafpad ~/.fluxbox/menu
```

Here is a generic form of the syntax to creating a submenu:

```
[submenu] (Menu name) </path/to/icon/icon.png>
[end]
```

If you would like to put an executable in the submenu, use the following generic syntax:

```
[exec] (Application Name) { application command line syntax }
</path/to/icon/icon.png>
```

Here is a trick to running a cli application in Eterm and have the window stay open after execution:

```
[exec] (Application Name) { Eterm -e sh -c "echo 'hello world!'; bash"  }
</path/to/icon/icon.png>
```

1.2 – Terminal

The default Terminal is Eterm. Eterm is an incredibly customizable lightweight terminal emulator, similar to Gnome-Terminal, but without the fat. You can access Terminal from the Desktop menu, the Dock, or even from a terminal. A lot of WeakNet Linux *WEAKERTHAN's* applications are executed in the Terminal. The Terminal is the most efficient, professional way to perform any system administration task. Please don't be afraid of the terminal; once you get into the swing of using it, you won't use anything else!

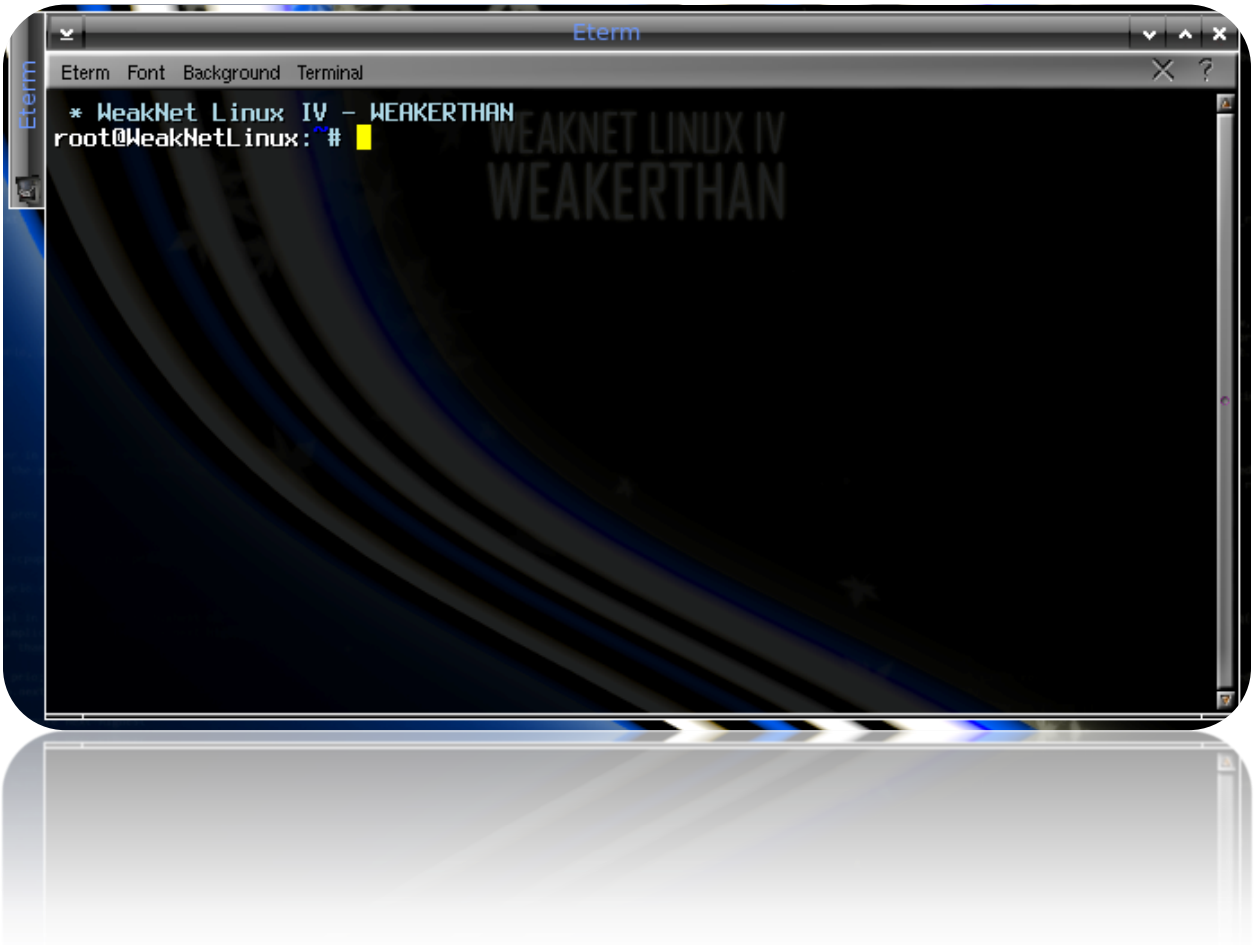


Figure 1.2: Eterm

The menu-bar at the top has two added buttons on the right side; a "?" button and an "X" button. These buttons will open a new Terminal and display the manual page, or exit the Terminal closing all of your non captive child processes, respectively. To exit the Terminal, you could also issue a CTRL+D.

1.3 – The Dock

The Dock is a Wbar dock, and can be edited right from the menu. This will open a text editor "nano" and the file `/usr/share/wbar/dot.wbar` Once again, you can easily see from my syntax how to add applications, including icons and even the background bar. The syntax is pretty simple, i: - icon, c: - command, t: - title. The first set of lines displays the background.

If you want, you can even change the font of the WBAR titles. You can get free fonts from the 1001freefonts.com website and unzip them using the following syntax:

```
Unzip <file>.zip
```

Now, simply put the TTF file in the `/usr/share/wbar/iconpack/wbar.osx/` directory as the name "font.ttf" with the following:

```
cp /usr/share/wbar/iconpack/wbar.osx/font.ttf
/usr/share/wbar/iconpack/wbar.osx/font.ttf.backup && cp /path/to/new/font.ttf
/usr/share/wbar/iconpack/wbar.osx/font.ttf
```

Figure 1.3: The WBAR dock in WeakNet Linux WEAKERTHAN

1.4 – Installing Applications

You can install Applications via Synaptic or from the command line. Most *WEAKERTHAN* applications, and applications included in the OS are for advanced Linux users. You may find that you would like to have a different GUI (Gnome or KDE) which has a full networking manager, to assist you in getting an IP from your network that is, say, WPA Enterprise, or WPA2 Enterprise enabled. A few applications were left off of the CD simply because it was intended to be a live CD for penetration testing. To fit all of the hacking tools onto the CD we had to forgo such applications as: OpenOffice.org ~360MB, GIMP, CUDA Toolkit, Nessus, Vim, Gnome, Gnome-Terminal, and more.

To begin installing applications with Synaptic, simply go through the Desktop Menu and click "Applications" then "Application Installation" and "Synaptic"

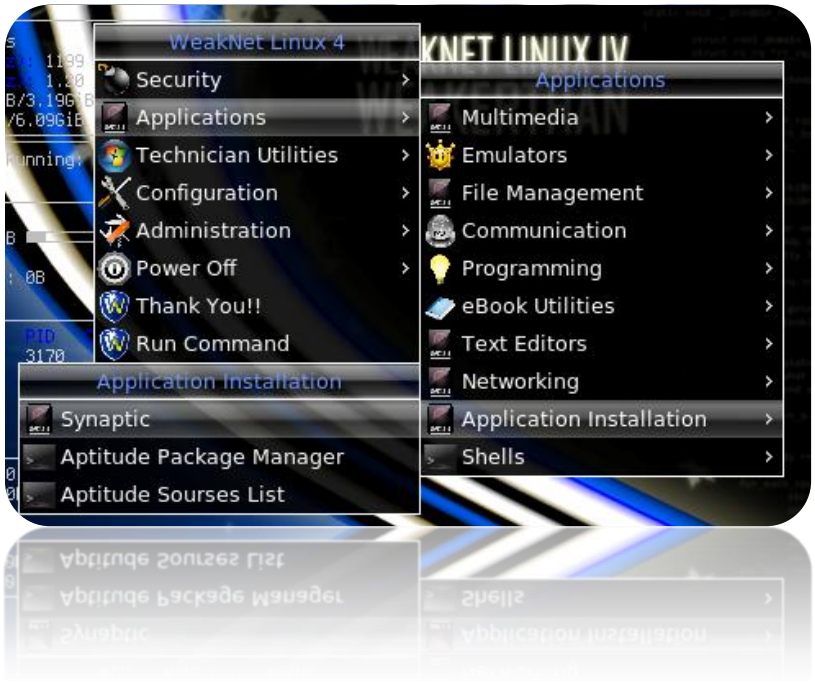


Figure 1.4: Synaptic Package Manager in FluxBox menu.

This will bring up a new window:

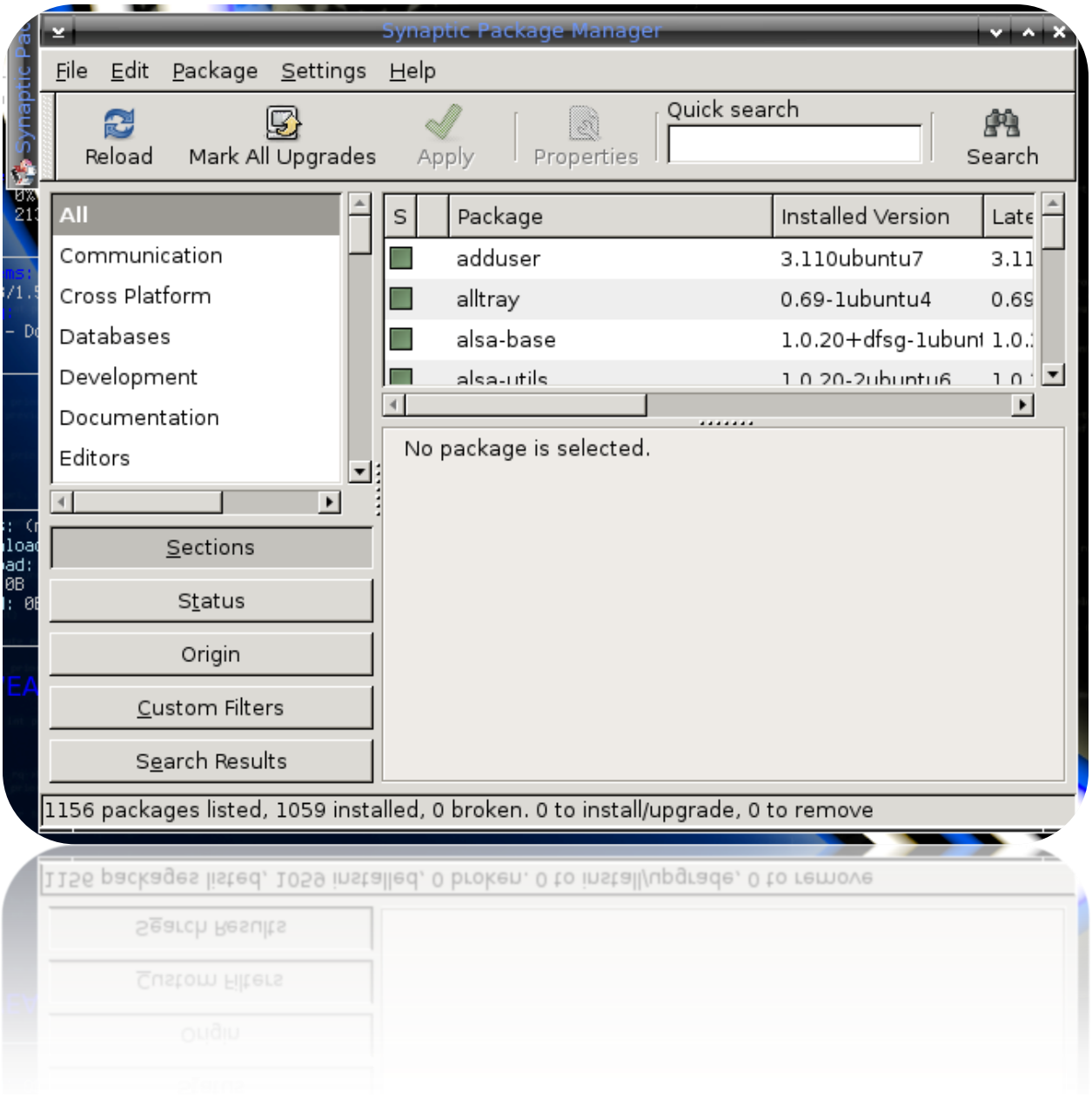


Figure 1.5: Synaptic Package Manager

The first thing we need to do is add online "repositories" to the application. I have added them in the `/etc/apt/sources.list file` for you, all you have to do is click "Settings" in the menu bar above then click "Repositories."

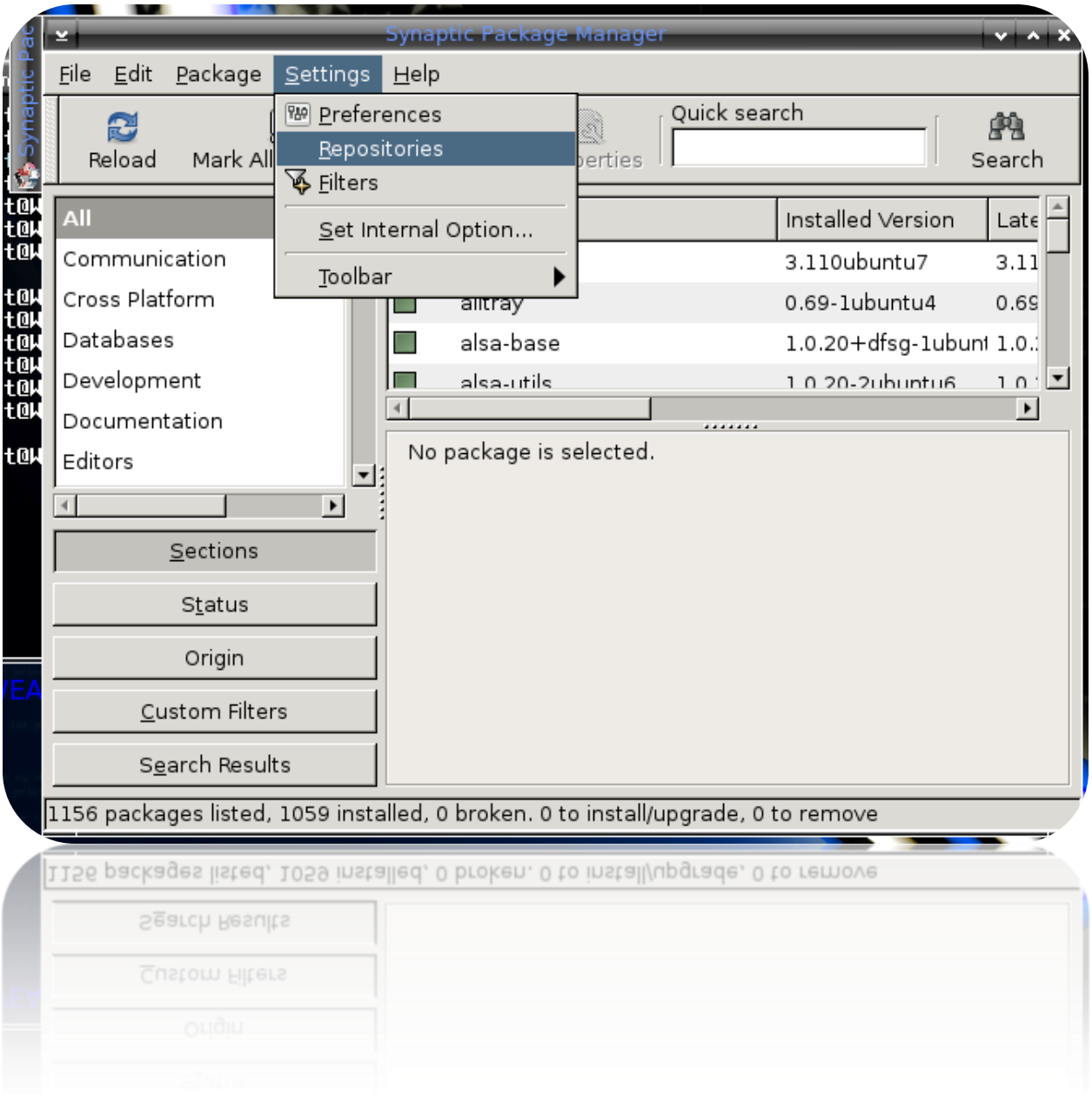


Figure 1.6: Add repos to Synaptic Package Manager.

This requires a network connection, because what it does is, it downloads lists from each "repository" which contain all applications available in the repository. Then, when it is finished, you click "Reload" and all the applications available to install will show up in the bottom right side of the window.

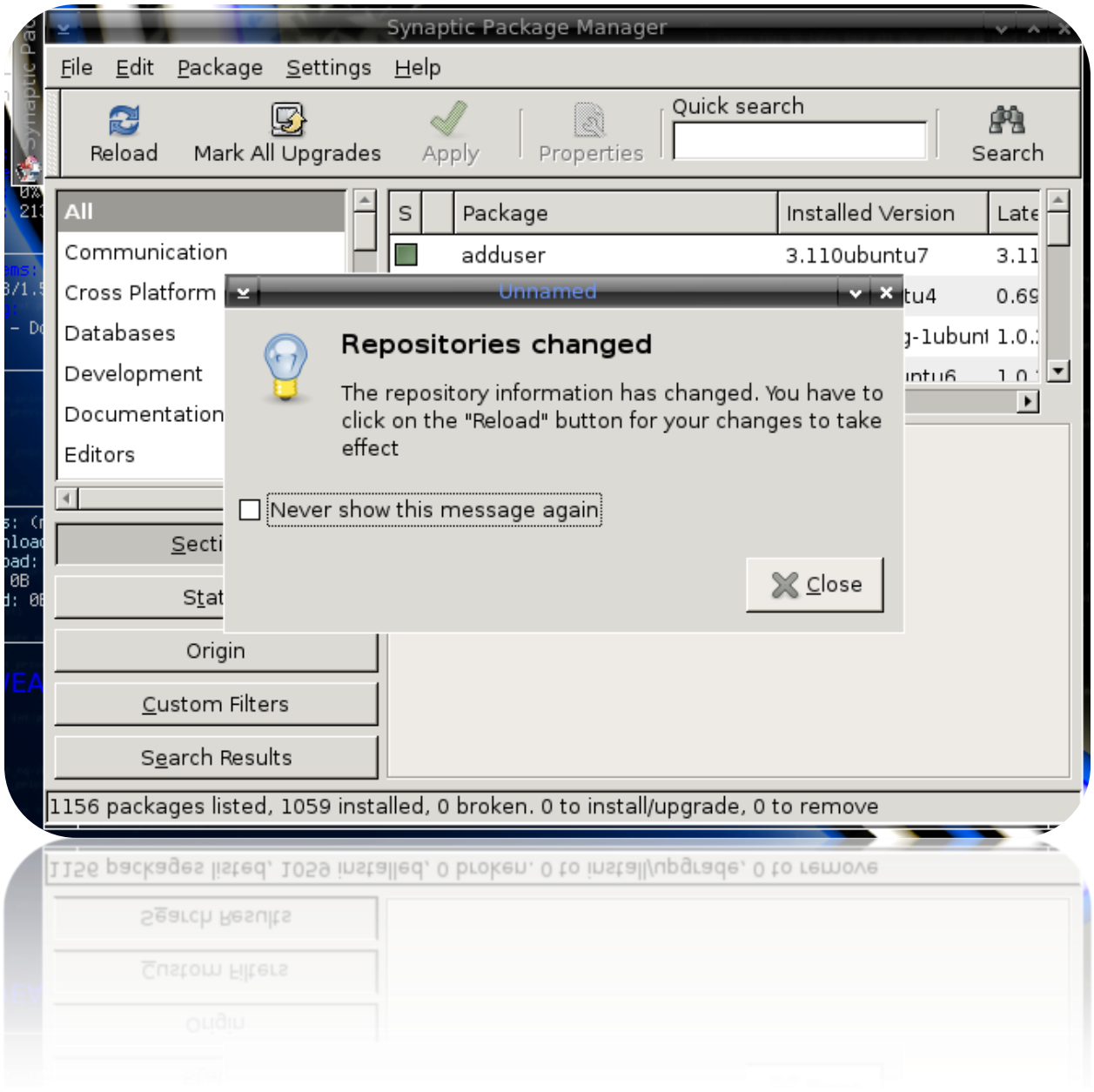


Figure 1.7: The automatic process of adding all repos to the Synaptic Package Manager.

The best way to find what you are looking for to install is by using the "Quick Search" bar at the top right side.

To install applications from the Terminal, simply do:

```
agi <package name>
```

And

```
acs <string>
```

To search for the proper name. Please note, that these commands are simply aliases of the `apt-get install <package name>` and `apt-cache search <string>`

1.5 – Application Examples Including Text Editors

LeafPad can be used to edit text in the FluxBox GUI and can be started by clicking it's icon in the Dock. You can also use xedit, or command line text editors such as Nano or Vi.

Part2. Networking

2.0 – Listing Devices

You can list your devices using iwconfig. In the screen capture below you can see that I have 2 networking devices. The extra piped output to grep a Regular Expression was simply used to show ONLY the devices.

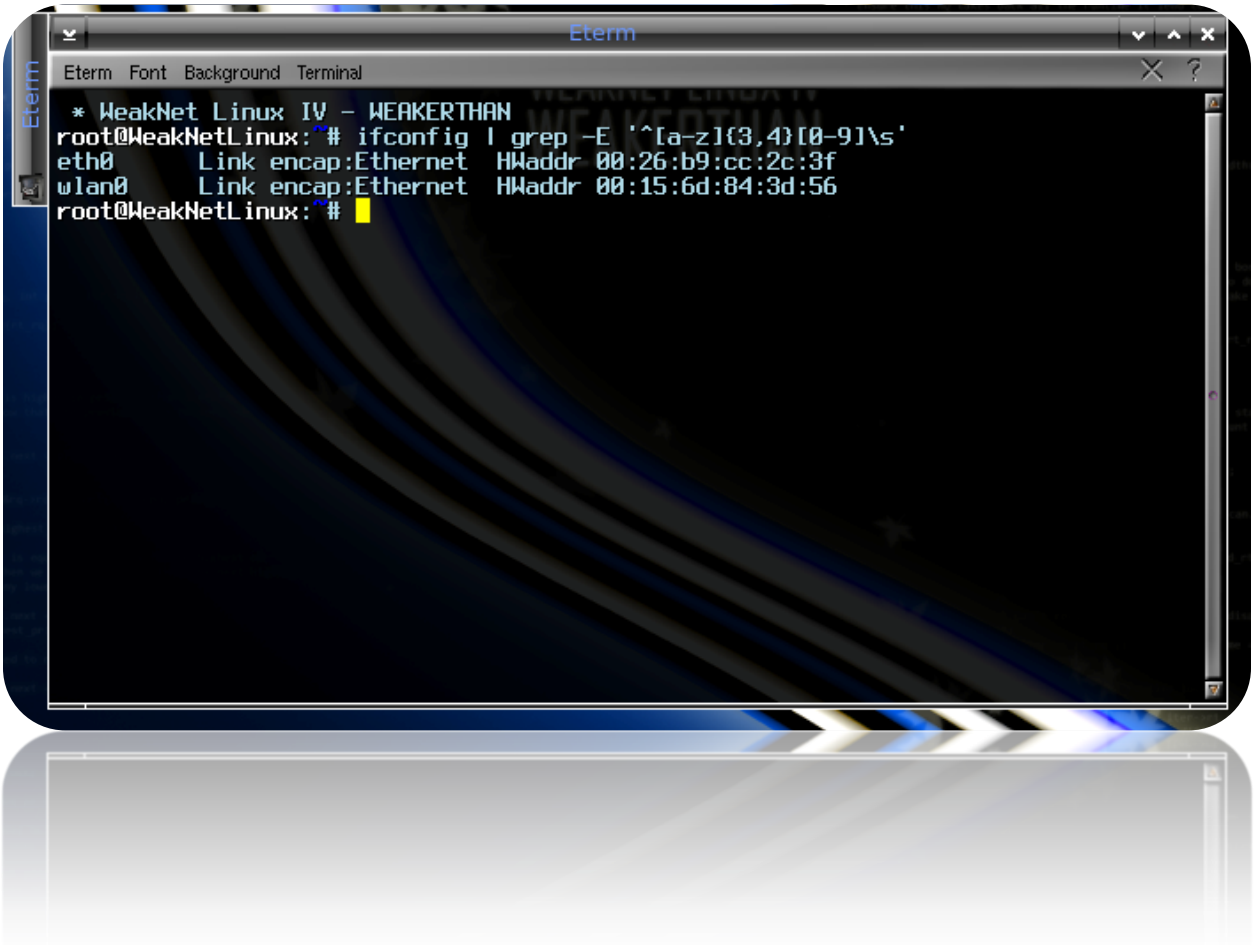


Figure 2.0: Showing all known NIC (Network Interface Card) devices in the system.

You can also list devices by using lscpi:

```
lspci
```

And optionally, you can also add the `-v` or `-vv` options to `lspci` to view even more information about your detected hardware.

Figure 2.1: WiFi Devices listed in the "About This Machine" application located in the Desktop Menu.

2.1 – Acquiring an IP address on a Wired Network

You can get an IP address by using `dhclient`.

```
dhclient <device name>
```

Dhclient is an amazing thing to have, as it sets up your IP from the DHCP server (most likely a router), sets up your DNS locations (places that change names into IP's to be read by the browser or other applications that require hostnames) in the `/etc/resolv.conf` file, and is very good at explaining any errors it comes across.

2.2 – WPA Assistant

You can use the WPA Assistant application which is a GUI frontend for WPA_Supplicant. This small application takes all of your information and simply runs the WPA_Supplicant commands. You can access the tool from the Desktop Menu by clicking "Administration" and "WPA Assistant."

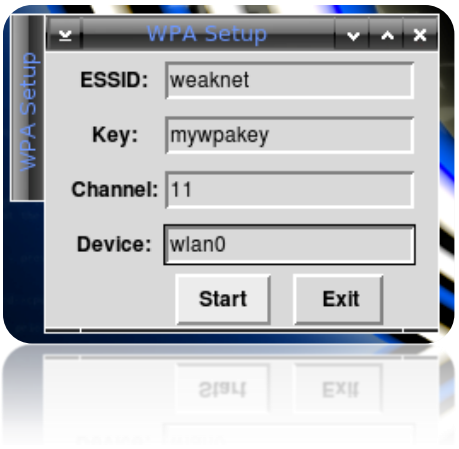


Figure 2.2: the WPA Assistant tool.

2.3 – WPA_Supplicant

WPA_Supplicant is installed on WeakNet Linux IV. You can run the commands right from the Eterm Terminal. First you need to create the PSK:

```
wpa_passphrase your_ssid your_psk
```

Copy the output and save it for later. Next, you need to edit the configuration file

```
leafpad /etc/wpa_supplicant.conf
```

And add the following information that pertains to your wireless network:

```
ctrl_interface=/var/run/wpa_supplicant
#ap_scan=2

network={
        ssid="your_ssid"
        scan_ssid=1
        proto=WPA RSN
        key_mgmt=WPA-PSK
        pairwise=CCMP TKIP
        group=CCMP TKIP
        psk=your_psk
}
```

CCMP is for WPA2, and TKIP is for WPA.

Now this command:

```
pre-up wpa_supplicant -Bw -D wext -i <device name> -c /etc/wpa_supplicant.conf
```

Where `wext` is the driver to use, which is generic and should work across most wireless devices. You should then be able to run **`dhclient`** and receive an IP from your router.

```
dhclient <device name>
```

2.4 – WEP

You enable a device to use your WEP key enabled network by issuing the following commands:

```
Iwconfig <device name> channel <channel number> essid <name of router> key
<WEP key>
```

Not necessarily in that order. Next, you will need to run dhclient to get an IP from the router.

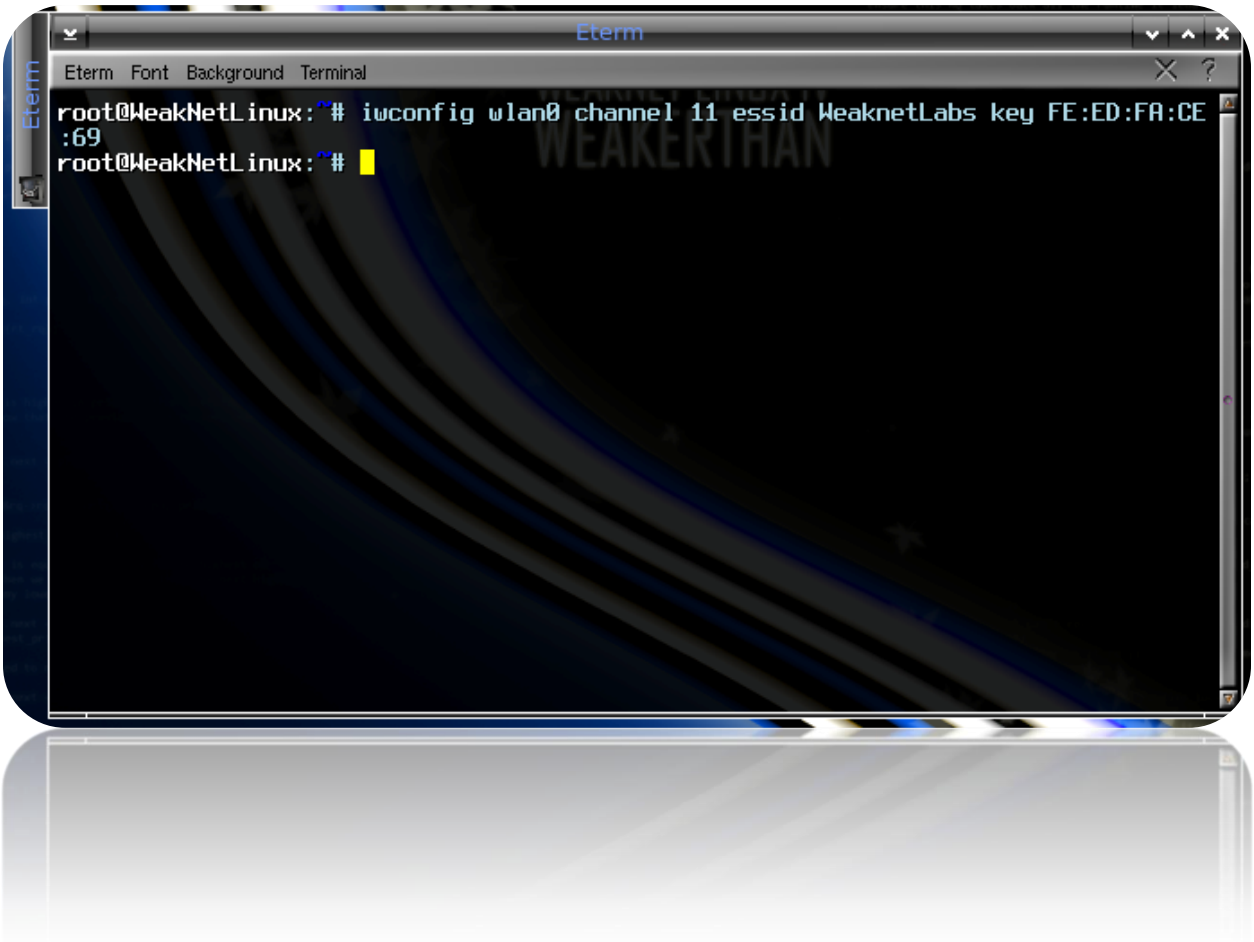


Figure 2.3: I am setting my wireless device up for WEP enabled BSS.

2.4 – Monitor Mode (Hacking)

Some applications require to you to put your WLAN card into monitor mode. Monitor mode isn't really a "mode" for wireless networking, it simply allows you to sniff raw data from the device through the air, and in some cases inject packets into a BSS or wireless network. The command:

```
airmon-ng start <device>
```

should work for most devices. This will create a VAP, or Virtual Access Point, which is a wireless device in non-managed mode. If "monitor" is not really a "mode" for wireless networking, then the only two modes are "managed" or "master." Master mode is for Ad-hoc networks were a station actually acts as an AP or wireless router.

This has been tested with Atheros Ath5k, Ath9k, Broadcom's b43, Ralink rt73USB, and a few others. If the device does not create an actual VAP, then it simply puts the device itself into monitor mode. You can later destroy the VAP with the following command:

```
airmon-ng stop <device>
```

Or simply remove the kernel module after turning the adapter off:

```
ifconfig <device name> down
rmmod <device driver>
modprobe <device driver>
iwconfig
```

2.5 – Monitor Mode (Hacking).

```
airmon-ng start <dev name>
```

This command will start your wireless face in monitor mode, or simply create a VAP (Virtual AP), to use in monitor mode. All Virtual AP's are on the same channel as the main wireless device at the same time. Most VAP's created start with the letters "mon." So your monitor mode enabled device may be called "mon0," for instance.

2.6 – Updating Aircrack-NG with one command.

Use this command to update Aircrack-NG Suite with this tremendous command:

```
Cd /appdev && svn co http://trac.aircrack-ng.org/svn/trunk/ aircrack-ng && cd
aircrack-ng && make && make install && airodump-ng-oui-update
```

2.7 – Drivers.

The WeakNet Linux WEAKERTHAN release has no madwifi-ng candidate. You can install the madwifi-ng by issuing the following, after getting an IP and internet connection via Ethernet:
First we need the Kernel Headers for WEAKERTHAN:

```
Cd /appdev/ && wget http://weaknetlabs.com/linux/devel/linux-headers-2.6.31-20-weakerthan_2.6.31-20.58_i386.deb && dpkg -i linux*
```

This will produce an error, as we need the generic Linux headers for 2.6.31-20:

```
Apt-get –f install
```

Now we are finally ready to make the driver and install it:

```
svn co http://madwifi-project.org/svn/madwifi/trunk madwifi && cd madwifi && make && make install
```

Now start the driver with:

```
Modprobe ath_pci
```

Please not, that if you have the ath9k or ath5k or ath9170 in place, you will need to put hose drivers down first:

```
Modprobe –r <driver>
```

The new revision of WEAKERTHAN WeakNet Linux IV has a new feature which will allow you to see exactly which drivers are in use. Simply access the Desktop menu and click Administration, then "WiFi Drivers Check"

Figure 2.4: Wireless Drivers Check.

2.8 WICd Network Manager Tool

WEAKERTHAN WeakNet Linux version 4.1k now features the WICd networking tool. To access the tool simply access the Desktop menu, then click Administration->WICd Network Daemon.

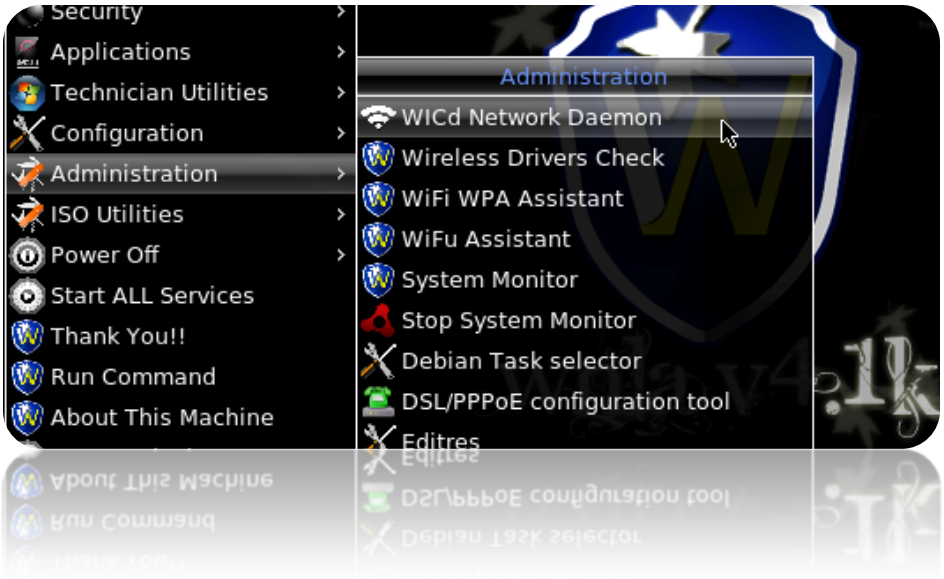


Figure 2.5WICd in the menu.

Once started, you will see an icon in the top taskbar on the right:

Figure 2.6 – The WICd tray icon.

Now, you can access the client for settings by clicking the icon. You will see a new window open up and you can click preferences then set up the device:

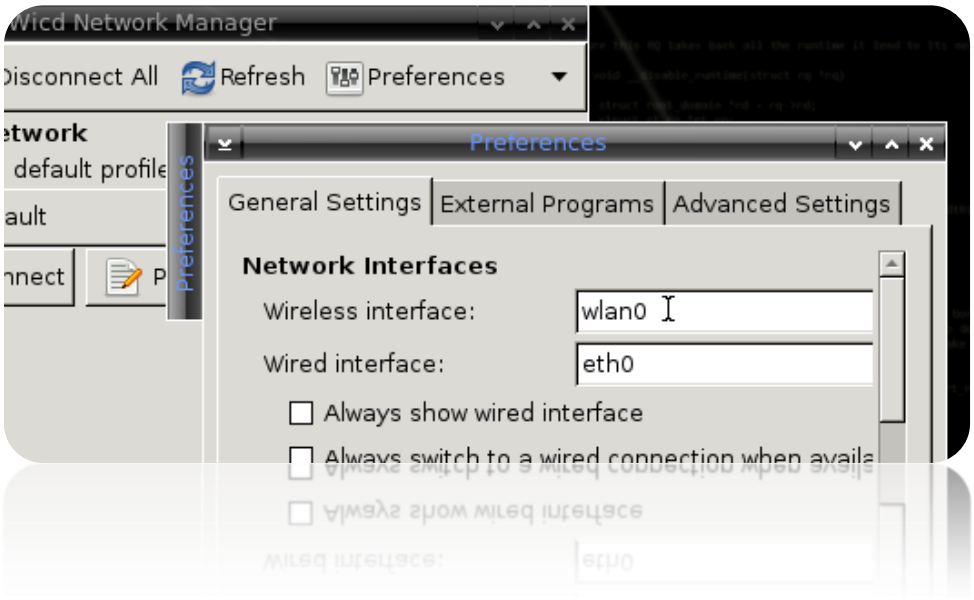


Figure 2.6 Using wlan0 as my device.

You can find your device by running `iwconfig`, the "Wireless Drivers Check" utility or `lspci`.

Part 3. Instant Servers

3.0 – Server Status (and why servers are important.)

Mostly all web-based applications use local or remote servers. This can include HTTP, MySQL, HTTP/PHP, and more. The Web Hacking Portal version 5 (WHPv5) software requires you have these services started. If you click on the blue WeakNet shield logo in the dock at the bottom of the screen, the servers are automatically started for you. This starts HTTP+PHP and MySQL, then brings up a browser which points you right to the WHPv5 root directory. If you are to click on the icon for WeakNet Social Network (WSN) or WeakNet CodeRepo, without clicking the blue WeakNet shield logo, you will get an empty browser stating that it cannot find the URL given. This is because the servers are not started. Here are a few screenshots of what the WeakNet Code Repo looks like without the services started and with it started:

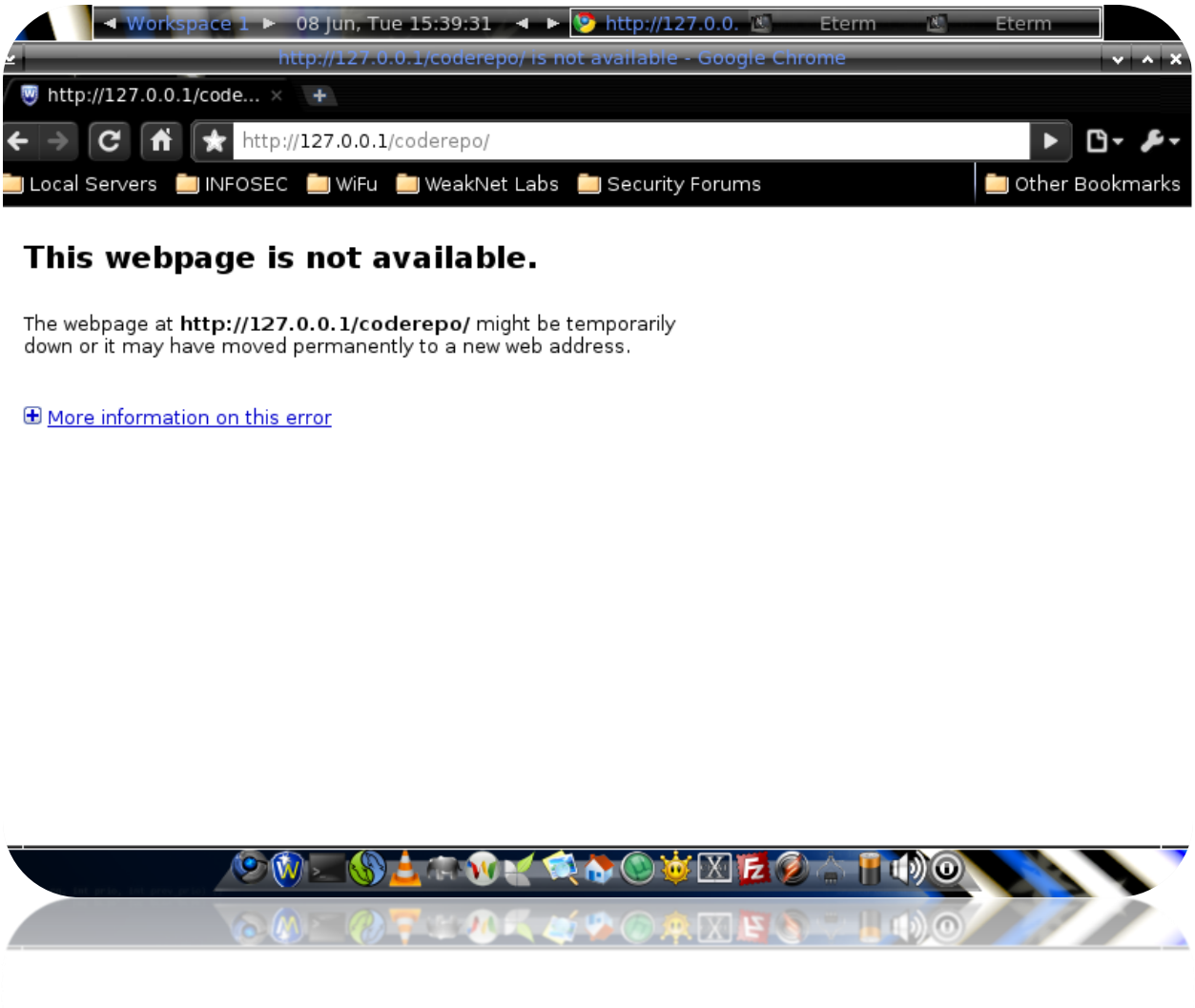


Figure 3.0: The WeakNet Code Repo without the HTTP Server started

Figure 3.1: The WeakNet CodeRepo with HTTP/PHP+MySQL servers started

In the following few sections, I will be giving you a few examples on how to start and stop the servers.

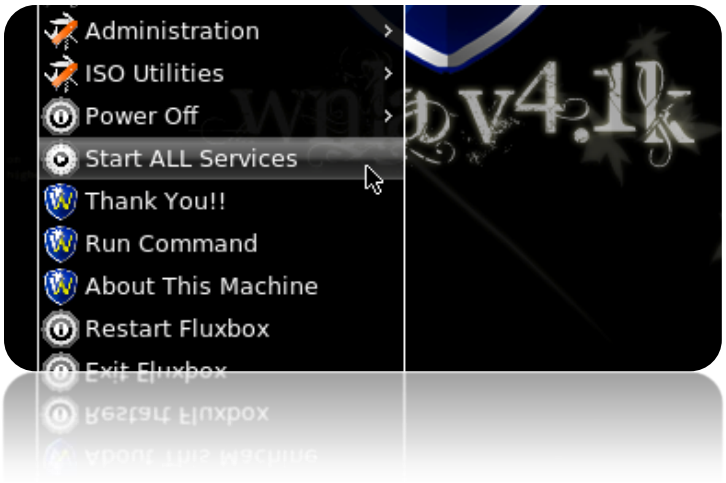
3.1 – Starting Servers via the FluxBox Menu

You can start the servers from scripts that were added to the FluxBox menu. Simply click on "Security" then click "Instant Servers" and then click on the servers you intend to use. In our case above, we should be using PHP-CGI Enabled HTTP (Lighttpd custom compiled for WNLAv4) and MySQL.

Figure 3.2: Starting the servers directly from the FluxBox menu

3.2 – Start ALL Services

With the release of WEAKERTHAN WeakNet Linux 4.1k+ you will have a new option in the desktop menu to start "ALL" services. This simply starts all the services needed to run the web hacking portal. These services do not automatically start due to the author trying to shorten the amount of time it takes to boot the WNLAv4.1k live CD.



3.3 – WHPv5

To access the Web Hacking Portal Version 4.0, simply click the blue WeakNet shield icon in the dock. This will give you access to Catchme-NG!, WardriveSQL, the WNL Phishers, and more.

3.4 – Stopping Servers

You can see the servers that are running on your localhost by issuing an Nmap command from the command line like so:

```
nmap localhost
```

Here is what the system will look like before starting the HTTP server on port 80:

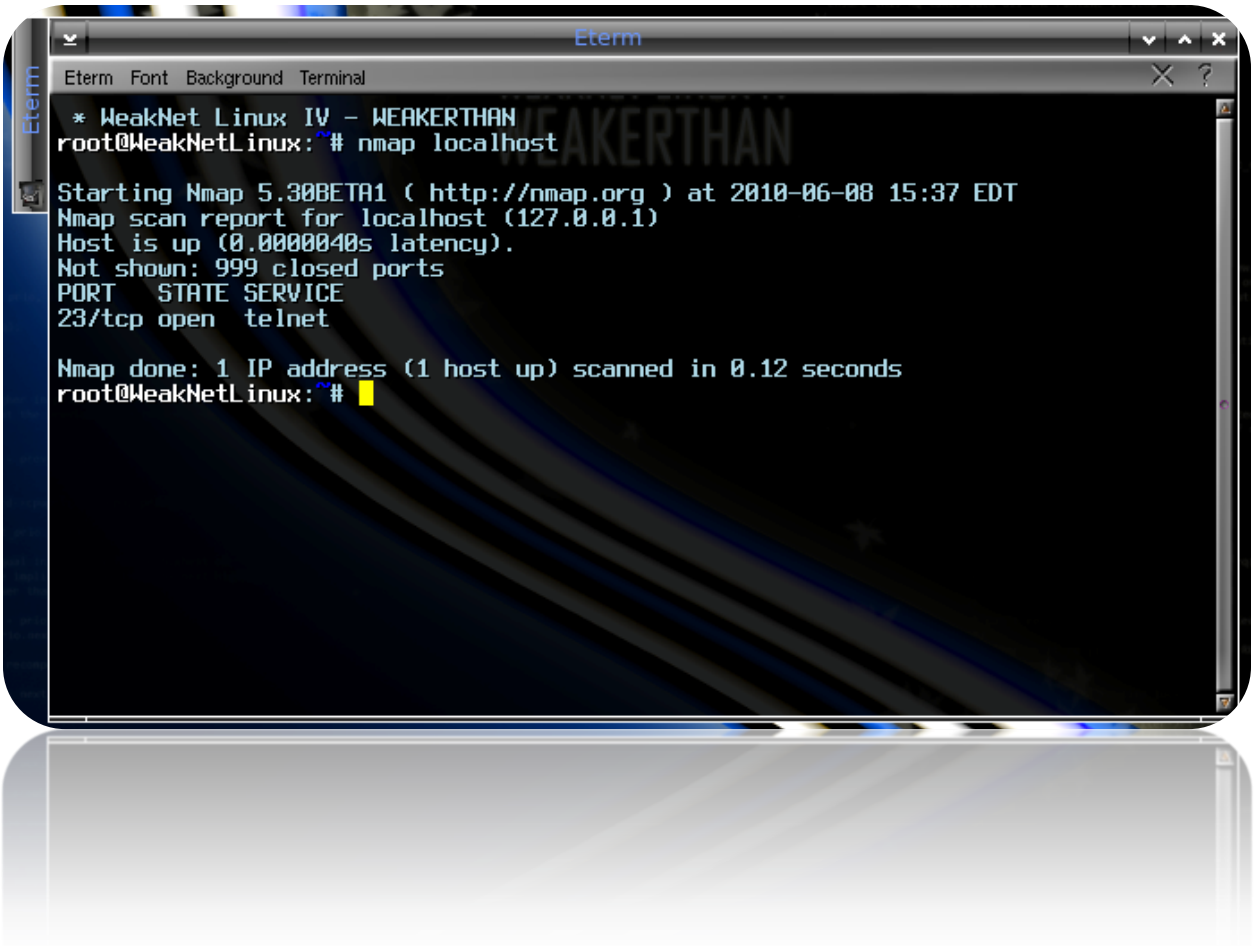


Figure 3.3: No HTTP Server here!

Here is an Nmap scan after I started the server:

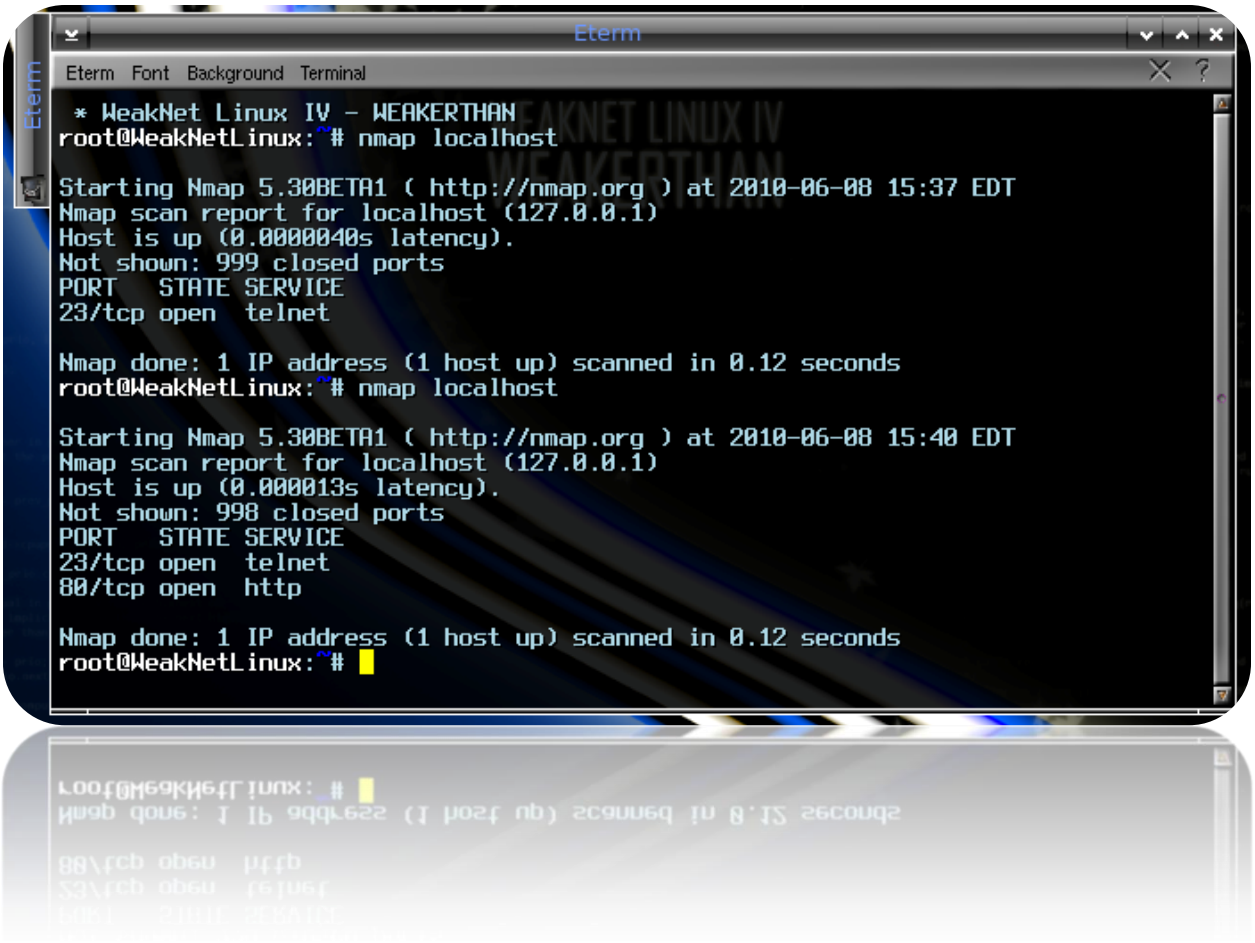


Figure 3.4: Now we have the server running on port 80!

Once again, the web based applications will not run properly without these servers. To stop the servers, you simply do:

```
/etc/init.d/<service name> stop
```

Or click the Stop Service script from the FluxBox menu that corresponds to your service. You can see in Figure 16 above that the "Stop Local Servers" menu is located at the red boomerang icon.

Figure 3.5: The Web Hacking Portal Version 5

Part 4. Hardware

4.0 – Listing your Hardware

You can list your Hardware with the lspci command:

```
lspci -v
```

This will give you manufacturer information as well as model names and much more. If you add the –v or –vv flags to the command the output changes and gives you much more information (being more verbose.) Armed with this information, you can then search the web with your favorite search engineto find troubleshooting with Ubuntu or Debian based OS's for that particular hardware.

4.1 – Mounting USB Drives

You can mount a USB drive my using the following method. First we create a directory to mount the USB drive to:

```
mkdir /mnt/usb
```

Next, we fin dthe drive name. plug it into the system and wait just about 10 seconds for the serial bus to settle. Then issue the dmesg command and only view the last few lines:

```
dmesg | tail
```

You should see something like `[numbers] sd [numbers] [sd<letter here>] Assuming drive cache…` The `sd<letter here>` part is the name the OS gives the physical device. In my example, the device is `sdb`. If you happen to have more slices or partitions on the USB drive, you can mount those individually, and they will be labeled `sdb1, sdb2, … sdbn` In my case I only have 1 partition, and I mount `sdb`:

```
mount /dev/sdb /mnt/usb
```

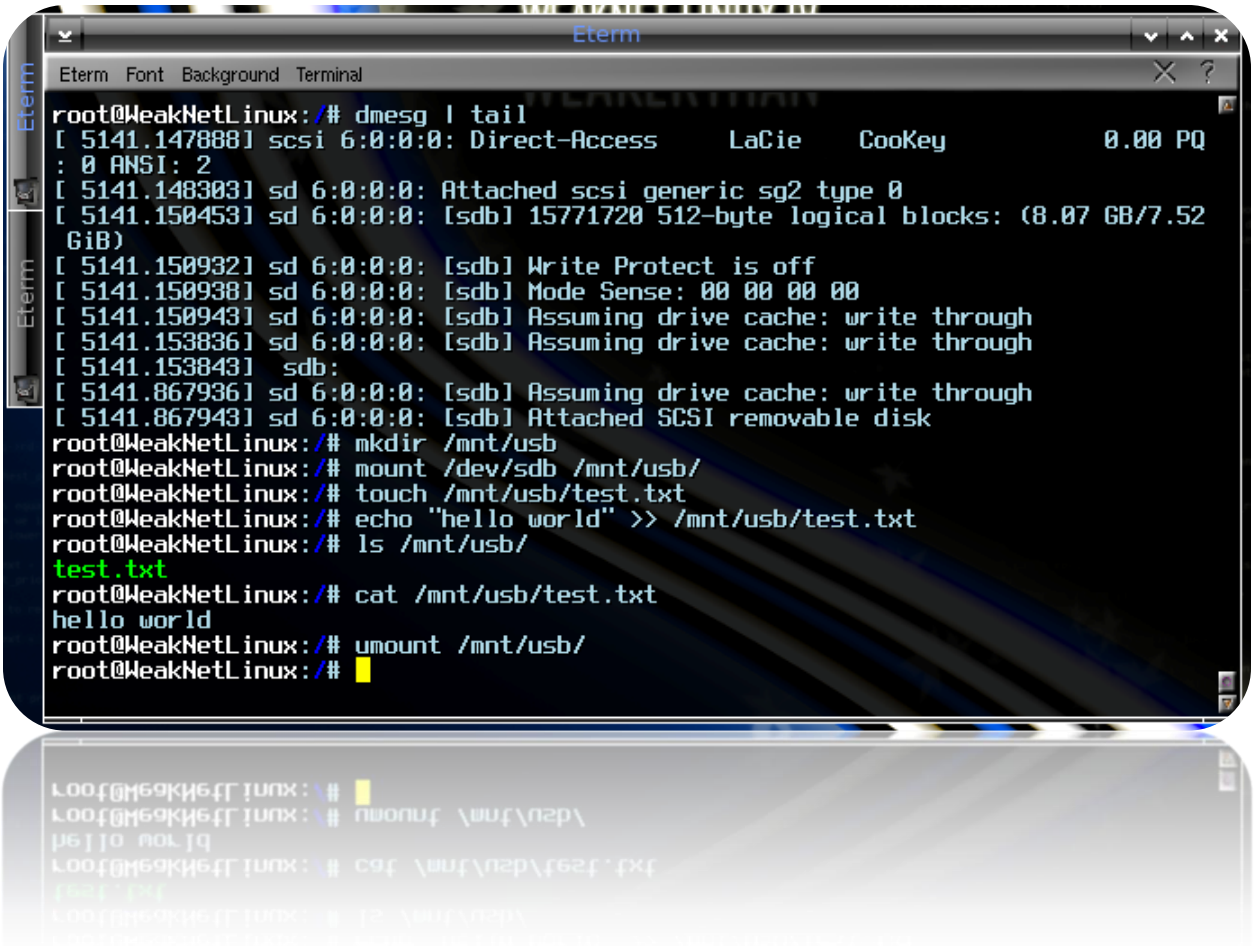


Figure 4.0: Mounting the USB Drive via Terminal

To un-mount the device you can use the `umount` command, as I do in Figure 18.

4.2 – Mount with PCMANFM

You can also mount a USB drive using the file-manager: PCMANFM simply click on the home icon in the wbar:

Figure 4.1: The PCMANFM icon in the WBAR:

Then click the drive's name on the left side window:

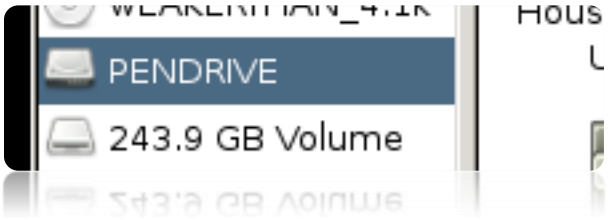


Figure 20: The drive is labeled but unmounted on the left side.

Now, you will see the drives files in the right side of the window. To unmount the drive, you can simply right click on the drive in the left side and click Unmount:

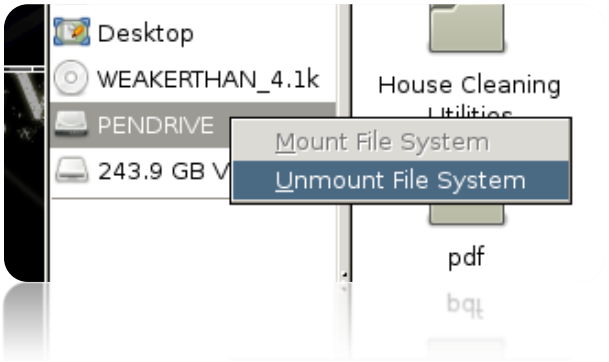


Figure 4.2: The unmount function.

4.3 – Battery Monitor

The Battery Monitor is available in the Dock at the bottom of the screen and runs from the text files located in /proc for the battery BAT0. If you have a second battery, this will not be shown.

4.4 – Sound / Volume

The Volume icon in the Dock will bring up an Eterm window with an Aslamixer session to change the volume levels:

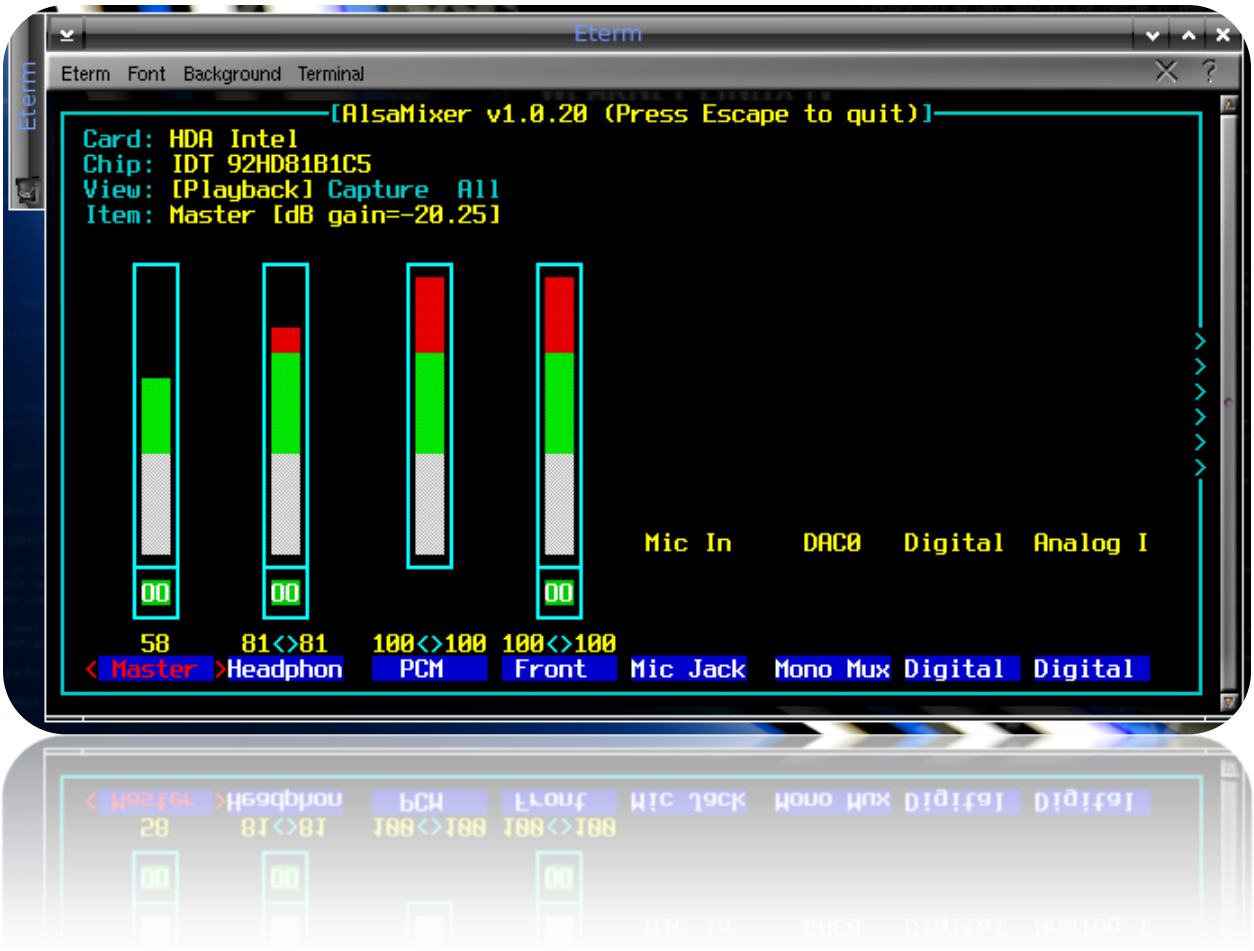


Figure 4.3: The Alsamixer application.

If you want more advanced options, hit the TAB key:

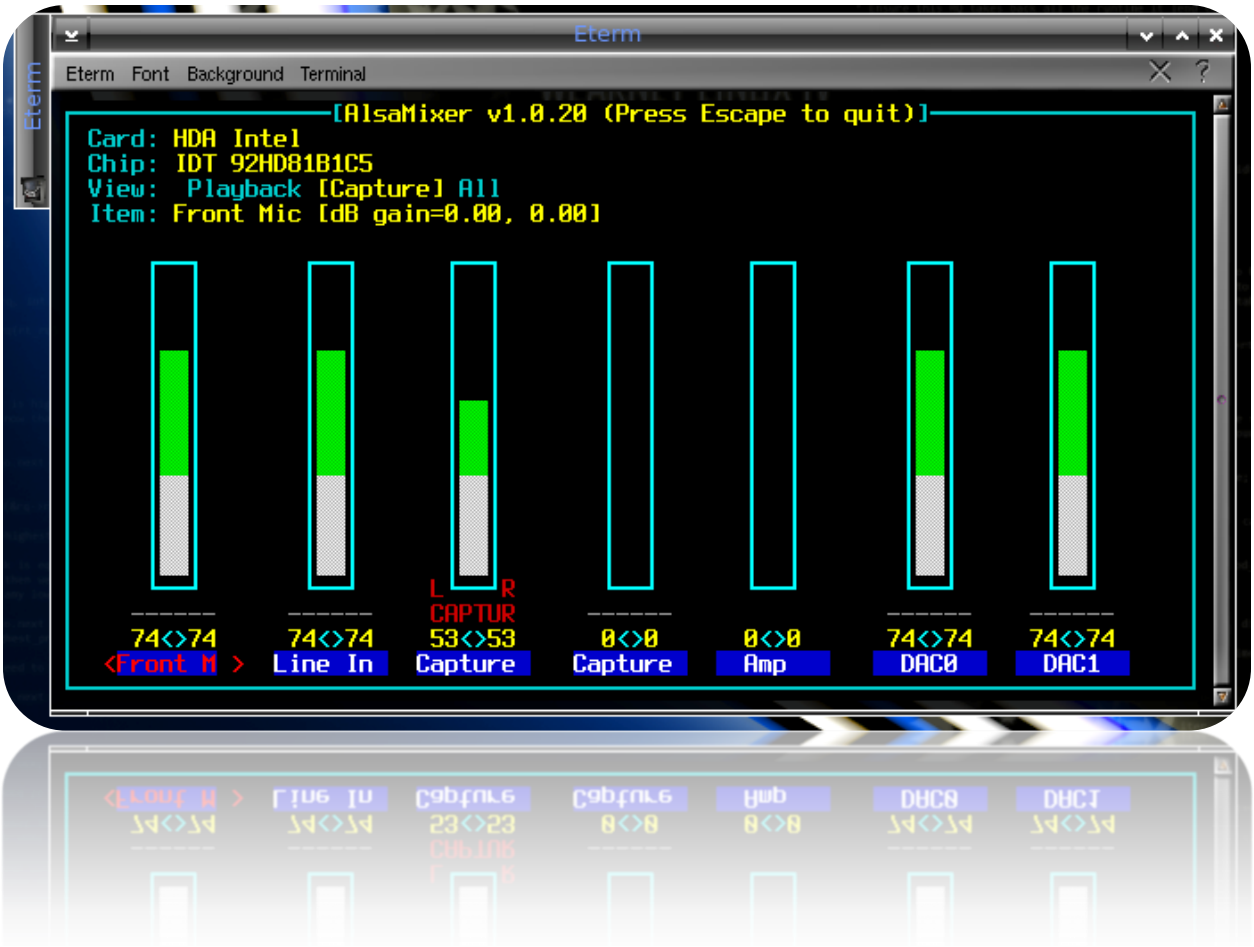


Figure 4.4: The Alsamixer with extra options.

Navigation in Alsamixer is easy; you simply hit the right arrow, or left arrow. To change the volume of a particular device, you simply hit the up or down arrows. To exit the Alsamixer, hit ESC.

Part 5. Installation

5.0 – Install to Hard Disk

You can install to a hard disk by using the following methods. First, we will cover how to install to a hard disk. Click on the Desktop and bring up the menu. Then. click "Run Command."

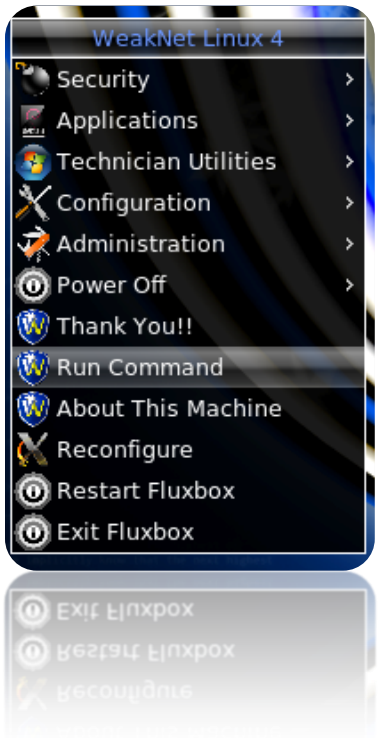


Figure 5.0: The Run Command

Then, in the box type `ubiquity` and hit "Run!"

Figure 5.1: enter "ubiquity" and hit Run!

From this point on, all you have to do is follow the prompts, just as you would to install Ubuntu, or Debian.

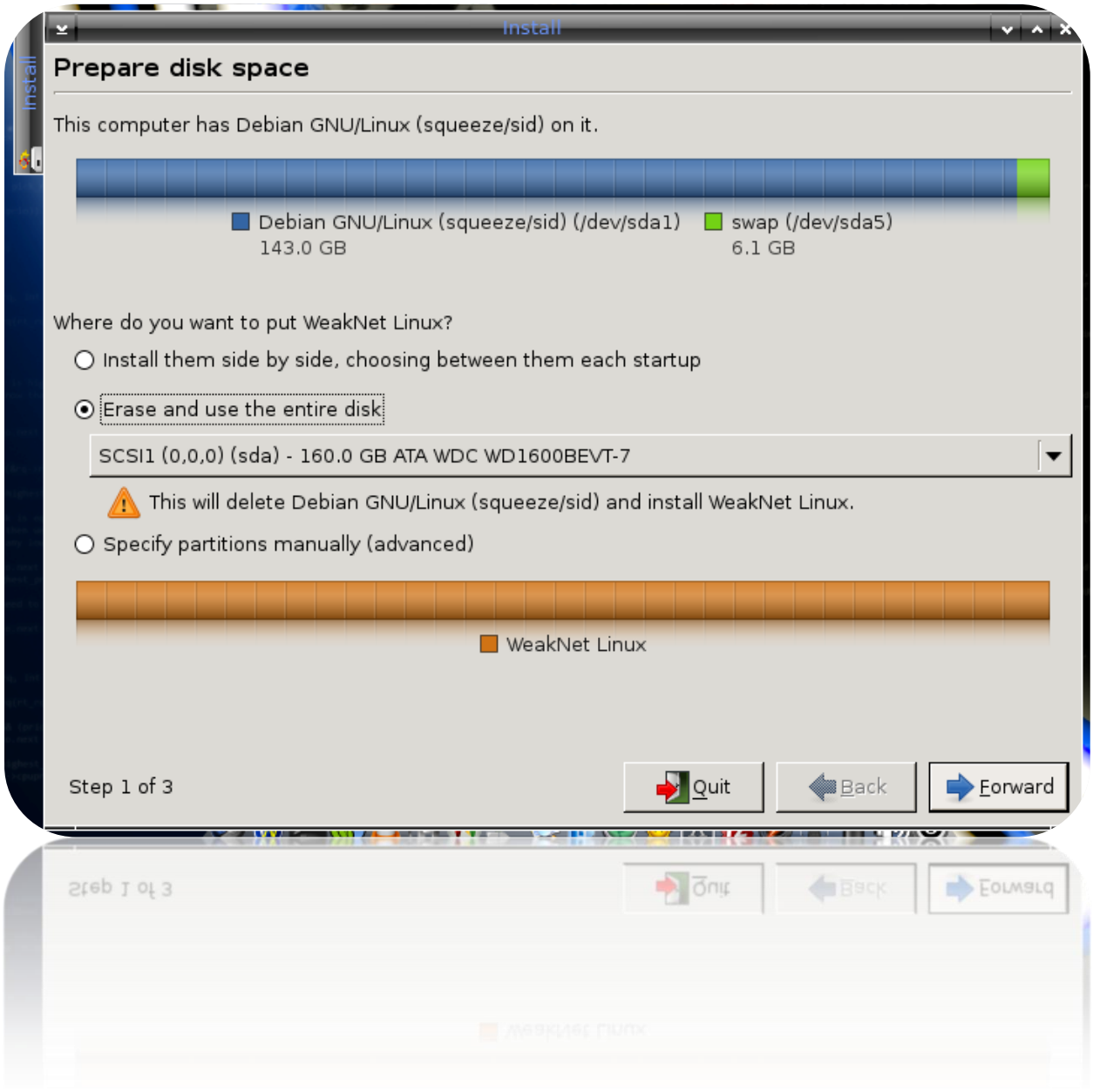


Figure 5.2: Follow the prompts from here to install the OS.

5.1 – Install to USB Drive (Unetbootin) [untested]

You can install to a USB drive if you have a network connection, by doing the following. Open Terminal and type:

```
apt-get update
apt-get install unetbootin
```

Now, without mounting the drive, plug the USB drive into the system and run the command:

```
unetbootin
```

This will bring up a window that is quite easy to follow. If you would like *WNLAv4 WEAKERTHAN* installed on your USB drive, simply mount another USB drive that contains the ISO or re-download it, and run the unetbootin utility and point the ISO option in the middle of the window to the ISO.

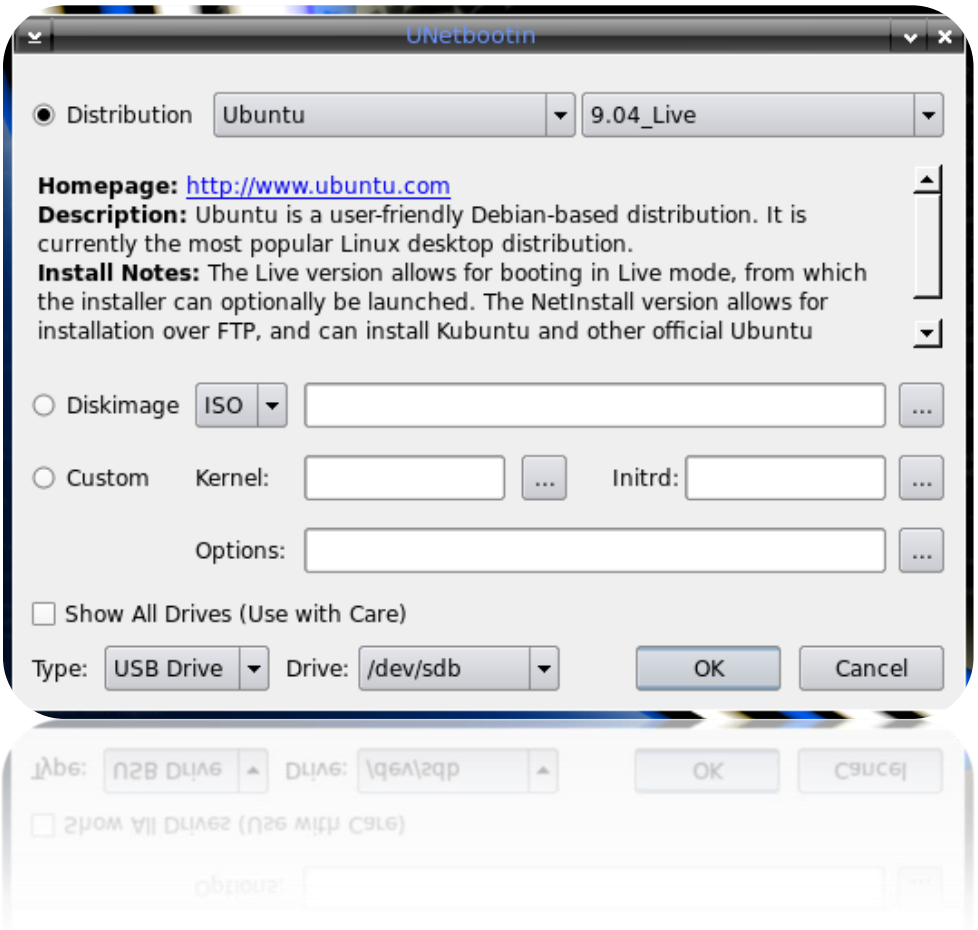


Figure 5.3: Unetbootin install

Next, just follow the prompts and make sure you choose your destination drive carefully!

BUGs

BUGs.0 Nouveau kernel driver error

This BUG has a work around for the NVIDIA NVS 3100m chip. You simply hit TAB while selecting your "live cd" in Grub and add the following options:

```
Nouveau.modeset=0 xforcevesa
```

before the – (double hyphen) in the kernel options. If you don't add the xforcevesa part, there will be no display, just text and the Terminal.

BUGs.1 Google-Chrome – No IP, no Server.

This BUG happens only occasionally. If you don't have an IP, you cannot run PHP files from the local lighttpd server.

BUGs.0 Google-Chrome – WardriveSQL refresh rate.

This BUG is awful. I ran into it after I released WNLAv4. I guess the coders of GC decided that if you refresh the page, you get refreshed back to the top of the page and not where you were! The only work around is that a GUI version of WardriveSQL is in the makes as it is included in the WiFiCake application soon to come from WeakNet Labs.

References

WBAR:

http://freshmeat.net/projects/wbar/

FluxBox:

http://fluxbox.org/

Ubutnu:

http://www.ubuntu.com/

Gnome:

http://www.gnome.org/

OpenOffice.org

http://www.openoffice.org/

Google Chrome for Linux:

http://www.google.com/chrome/intl/en/landing_chrome.html?hl=en&hl=en&brand=CHMA&utm_campaign=en&utm_source=en-ha-na-us-bk&utm_medium=ha

Remastersys Project:

http://www.geekconnection.org/remastersys/index.html

Aircrack-NG:

http://www.aircrack-ng.org/

Unetbootin:

http://unetbootin.sourceforge.net/

CUDA Zone:

http://www.nvidia.com/object/cuda_home_new.html

1001 Free fonts:

http://www.1001freefonts.com/

WiFiCake:

http://weaknetlabs.com/main/?p=368

WardriveSQL:

http://wardrivesql.info/

WPA_Supplicant Ubuntu Howto:

http://ubuntuforums.org/showthread.php?t=263136