

Esteban Herrera

DIY Linux

MASTER YOURSELF PURE
BLEND DEBIAN DESKTOPS
FROM A SCRATCH

v1.0.1

Esteban Herrera

DIY Linux

MASTER YOURSELF PURE
BLEND DEBIAN DESKTOPS
FROM A SCRATCH

v1.0.1

Copyright © Esteban Herrera

ISBN 978-000-000-000-2

Published by Esteban Herrera Publishing.

Printed in Herreras' Printhouse.

2017

Table of Contents

Table of Contents.....	4
Meet the software installation sources and options available.....	10
Where can I find software for GNU/Linux?.....	10
Software sources available.....	10
Alternative software deployments.....	11
Automated installation processes.....	11
Installation from image files.....	12
Installation on Virtual Machines, emulators and chroot environments.....	13
Binary installation.....	13
Convert other distro binaries into DEB packages.....	14
Debian binaries.....	14
Flatpak, X-Apps or Snap packages Debian equivalent.....	15
Full compatible/portable Linux/Debian binaries.....	16
Source Code installation.....	17
Debian source code.....	17
Source code & Debian forks source code.....	19
Binaries or Source Code from other operating systems.....	26
MacOS binaries or source code.....	26
Source code built for a Windows or DOS version.....	26
Windows or DOS executable binaries.....	27
Multiplatform installation.....	27
Desktop Extensions.....	27
Multiplatform Desktop Applications.....	28
Dominate activities regarding the software installation and management.....	29
Kernel.....	29
Build the Linux Kernel.....	29
DKMS (Dynamic Kernel Module Support).....	31
Find out unfound drivers.....	32
Working with Kernel modules.....	33
Package Management.....	34
Build packages from source.....	34
Multi Architecture.....	40
Static, shared dynamic and loadable libraries.....	41
System Package Managers: Synaptic, Aptitude, DPKG, APT, Gdebi, Complementary.....	42
Verify download integrity, PGP, GPG, GNUPG, Seahorse.....	49
Software installation table.....	55
Repositories.....	68
Create a local Debian repository.....	68
Manage traditional software repositories.....	68
Build a Minimal Debian Live DVD from source code.....	74
Debian from a scratch.....	74
Set up and maintain a perfectly functional operating system.....	80

Back-up, Emergency and Restore.....	80
Cloning and directory and system back-ups.....	80
Data recovery.....	82
Emergency keys combinations.....	83
System Rescue.....	84
Filesystems.....	84
Boot Loaders and Dual Boot.....	84
Clean up the system.....	86
Partitioning.....	86
Sudo.....	86
Users, groups and terminals.....	87
Working with directories.....	91
Gnome Desktop.....	92
Gnome Desktop Icons, Themes and Cursors.....	92
Gnome Shell Extensions.....	92
Networking.....	97
Basic networking.....	98
Bluetooth.....	107
Share system directories across the network.....	109
Sync devices in the Cloud.....	110
Optimization and Power Management.....	113
ACPI battery monitor.....	113
Optimization and performance, speed up computer, save battery power.....	113
Peripherals.....	118
Camera tethering.....	118
Game Controllers and Joystick.....	119
Printer.....	120
Scanner.....	121
Smartphones.....	122
Video VIM.....	122
Settings.....	123
Change the default login screen.....	123
Check hardware information.....	124
Create a Bootsplash screen.....	126
Decode Man pages.....	126
Fonts.....	127
Make a program the default option.....	128
Set Date, Time and Timezone.....	129
Set Keyboard, Local options, Region and Language.....	134
Systemd.....	136
Wine.....	142
Sound.....	145
Sound.....	145
SSDs/HDS/CDs.....	146

Logical Volume Management (LVM).....	146
Mount and repair CD-DVD ROM ISO or HDs-SSDs.....	148
Multiboot USB Pendrive.....	151
Set up software RAID.....	151
Solid State Drives (SSDs).....	151
Tune Hard Disk with hdparm.....	154
Updates and Upgrades.....	155
Update the system and applications.....	155
System version upgrade procedure.....	155
Video.....	156
3D Acceleration and VR.....	156
Brightness.....	164
Color calibration.....	165
Reconfigure Xserver hardware (Video Card, Keyboard, Mouse, Monitor).....	165
Video card.....	166
Webcam.....	187
Complete a decent environment by testing the most of software installation options available.....	189
3D Modeling, CAD and Animation.....	189
Blender.....	190
Openscad.....	190
Synfig Studio.....	190
AI and Machine Learning.....	190
LiSpeak.....	191
Sirius.....	212
Audio and Music.....	223
Audacity.....	223
Frescobaldi.....	223
KXStudio.....	223
Linthesia.....	226
Music Players.....	226
Sound Recorder.....	226
Browsers.....	226
Firefox.....	227
Google Chrome.....	230
Icedtea.....	232
Broadcasters.....	234
OBS.....	234
CD Burners.....	234
K3b.....	234
Compressors.....	235
WinRAR.....	235
Desktop Enhancers.....	235
Cairo-dock.....	235
Compiz-Fusion.....	239

Conky.....	240
Screenlets.....	244
Xscreensaver.....	245
Desktop Environments.....	249
Fluxbox.....	249
KDE.....	258
Xfce.....	259
Development Editors and IDEs.....	259
Aptana Studio 3.....	259
Bluefish.....	260
Brackets.....	261
Code::Blocks.....	264
MonoDevelop.....	264
Netbeans IDE.....	265
Rstudio.....	266
Sublime Text Editor.....	266
TeXstudio.....	267
Development Languages.....	268
Java.....	268
NET Framework.....	273
R.....	273
Diagramming.....	274
Dia.....	274
Umbrello.....	275
Electronic.....	275
Proteus.....	275
Email.....	275
Icedove.....	275
Emulators.....	276
Genymotion.....	276
Firefox OS.....	276
Mupen64Plus.....	276
Games.....	276
Steam.....	276
Graphics Design and Photography.....	279
Darktable.....	279
GIMP.....	280
Photoshop.....	280
RawTherapee.....	281
Scribus.....	281
Illustrators.....	283
Inkscape.....	283
Krita.....	283
My Paint.....	284

Library.....	284
Adobe Digital Editions.....	284
Calibre.....	284
Management.....	284
ProjectLibre.....	284
Meeting and Chat.....	285
Pidgin.....	286
Ring.....	286
Skype.....	286
Xchat.....	287
News Feed.....	287
RSSOwl.....	287
Office.....	287
Gimagereader.....	287
Kingsoft Office.....	287
Libre Office.....	288
Remote Desktops.....	288
Teamviewer.....	288
Science.....	289
Calculator.....	289
Stellarium.....	289
Scilab.....	289
Screen Recorders.....	290
Screenshot.....	290
SimpleScreenRecorder.....	290
Video and Audio Tools.....	294
Ffmpeg.....	294
Handbrake.....	295
Video Editors.....	295
Cinelerra.....	295
Kdenlive.....	295
Lightworks.....	295
Natron.....	296
OpenShot.....	296
Video Players.....	297
Flash Player.....	297
KODI.....	301
Popcorn Time CE.....	301
VLC Player.....	301
Virtualization.....	301
KVM.....	301
VirtualBox.....	312
Learn to install web development software for the MacOS.....	321
Ruby.....	321

Linuxbrew.....	323
Linuxbrew – Nodejs.....	328
NPM – Grunt.....	329
NPM - Bower – JQuery.....	333
RVM.....	339
Linuxbrew – Taps.....	435
Linuxbrew – Git.....	446
Linuxbrew - Git tools.....	460
Linuxbrew – Nginx.....	466
Linuxbrew – MongoDB.....	470
Linuxbrew – Cassandra.....	483
Linuxbrew - libsass and sassc.....	483
Ruby - sass and Bootstrap-sass.....	484
Security first.....	484
Improve the system security on local.....	484
Chrootkit.....	484
ClamAV.....	485
LUKS.....	485
Maldet.....	485
OpenSSL.....	485
Rkhunter.....	485
SELinux.....	485
Sysctl.....	486
Tripwire.....	486
VeraCrypt.....	486
Improve the security of the system over the networks and Internet.....	488
Ad blockers.....	488
Bastille.....	770
Darkstat.....	770
Etherape.....	770
Fail2Ban.....	770
Ip.....	770
Netfilter.....	770
Netstat.....	780
Nmap.....	780
OpenSSH server.....	781
OpenVPN.....	782
Ping.....	782
Proxies.....	783
Screen.....	783
Site Advisors and page checkers.....	785
Speedtest.....	785
Tcpdump.....	785
TOR.....	786

Traceroute.....	786
Wireshark.....	786
Improve the security of the system since the network.....	786
Kali.....	786
Network Firewall.....	787
OSSIM.....	787
REMnux.....	787
Esteban Herrera.....	787

Meet the software installation sources and options available

Where can I find software for GNU/Linux?

Software sources available

In Debian GNU/Linux it's possible to classify the software installation in several ways. Here is one which consist of the list of Software sources available.

Software sources available:

- Disc image
- Automatic installation frameworks
- Distribution DVDs
- Netinstall CD
- Localhost or Local Area Network Debian repository
- Debian repository servers
- Debian PPA repositories
- Web site, P2P and Developers repositories
- Ubuntu PPA (Known exceptions)
- Any RPM or alternative Linux binary package repo or source
- Alternative Package Managers sources
- Microsoft's Windows world executables sources
- Microsoft's Windows world source code sources
- Chrooted Operating System or VM app sources
- Desktop Extensions sources
- Multiplatform Desktop Applications sources

The concept of Official Debian repositories:

To be clear, when I speak of Debian Official repositories (or repos) I mean exactly the repos configured in the directory `/etc/apt/sources`, especially in the file `sources.list`. The "fountain" of this sources can be the Debian DVDs, the Netinstall CD, any of the repositories in the Debian repository servers, or a LAN or a repository allocated in the localhost, which is also possible to prepare.

LAN Debian repos:

The Local Area Network or localhost options are going to improve the package installation security because no one can intercept and replace real packages with actually time bombs as occurs over public Networks.

Create a local Debian repository to improve the security, and standardize and cache of the installation packages.

References:

Topic: Create a local Debian repository.

Guides:

Create a local Debian repository

Debian PPA repositories:

Debian PPA is a project under development. In the meantime, to download PPA source use the PPAs of Debian forks distros like Mint or Ubuntu.

With a Debian PPA the programs installation would be as easy as add the new PPA to the system and then install from the package/source in a simple step with a tool as APT.

I would like to search for and navigate for the community packages on a website.

Package Verification:

You can also verify the packages installed already in the system and the new ones right after they downloaded, before begin their installation process.

References:

Topic: Verify the integrity of packages.

Guide:

Verify download integrity, PGP, GPG, GNUPG, Seahorse

Alternative software deployments

Automated installation processes

Software sources available:

- Automatic installation frameworks

Installation examples:

- Installation from FAI (Fully Automatic Installation framework):

References:

Web: https://debian-administration.org/article/240/FAI_the_Fully_Automatic_Installation_framework_for_linux

File: FAI the Fully Automatic Installation framework for linux.pdf

- Installation from Ansible:

References:

Web: <http://www.tecmint.com/install-and-configure-ansible-automation-tool-in-linux/>

File: How to Install and Configure 'Ansible' Automation Tool for IT Management - Part 1.pdf

- PXE Boot server configuration:

PXE Boot is very useful when we looking to re-image / install more clients yet a time

References:

Web: <http://arkit.co.in/linux/pxe-boot-server-configuration/>

File: PXE Boot server configuration step by step Guide.pdf

Installation from image files

Software sources available:

- Disc image

Installation examples:

This type of installation is not "installing" anything at all but deploying the complete system or a part of it from a system image or a partition image, thanks to shells scripts or a specialized software as Clonezilla, Partimage or Bacula

Shell scripts:

Always my best choice. I use my own scripts to backup complete system partitions. However I am not going to publish my scripts by the moment.

Clonezilla:

Clonezilla, based on drbl, partclone and udpcast, allows you to do bare metal backup and recovery. This package provides Clonezilla SE (server edition) which is for massive deployment: it can clone many (40 plus!) computers simultaneously.

References:

Topic: Clonezilla.

Web: <https://packages.debian.org/sid/clonezilla>

File: Clonezilla - Details in Sid.pdf

Web: <http://drpaudel.com.np/?p=221>

File: Clonezilla tutorial for Debian 6.pdf

Partimage:

Partimage is a client-server application, but you can use only the client and an external HD to replace the server. The process without the server takes much less time to complete the restoration from teh backups. In short, works as follows: You install the server where you will have the partition backups, include complete copies of systems like this Desktop. When the sever is ready, boot the client from a Live Distro CD, which comes with a copy of Partimage pre-installed (I recommend the distro Knoppix which DEbian compatible). In the client create the backup. the client will create a compressed image in the server. When you want to re-install the partition or the whole system from teh backups in the server, turn on the server and re-insert the Live Distro CD to run Partimage and ask him to restore everything.

References:

Topic: Backup and restore Linux partitions with Partimage.

Web: <http://www.debianadmin.com/backup-and-restore-linux-partitions-using-partimage.html>

File: Backup and restore Linux partitions with Partimage.pdf

Topic: Linux Knoppix

Web: <http://www.knopper.net/knoppix/index-en.html><http://www.knopper.net/knoppix/index-en.html>

In both cases, Clonezilla and Partimage, the concept of restoration makes a second sense: We are talking about a system and software installation instead of a simple restoration. Even though there is not data in a new internal hard drive you can deploy on it a complete copy of a system in 20 minutes or less, meanwhile the targeted hardware and the hardware where the installation of the system to copy occurred are identical.

Installation on Virtual Machines, emulators and chroot environments

Software sources available:

- Chrooted Operating System or VM app sources
- Distribution DVDs
- Disc image
- Automatic installation frameworks
- Distribution DVDs
- Netinstall CD

Installation examples:

- Photoshop
- Run Android apps in the Debian OS
- Create a chroot environment to install Linux Mint inside

I am not running any Windows programs on my Linux Box, but Photoshop in a Virtual Machine.

You are free to change my method by installing and setting up your Wine accordingly. You have to know that many Adobe Adobe Photoshop versions run very well in Linux with Wine.

You can install other system rather than Debian inside Debian using chroot environments and then run the distro apps normally no matter that Debian is the main OS.

References:

Guides: Photoshop, VirtualBox, KVM

Binary installation

Convert other distro binaries into DEB packages

Software sources available:

- Any RPM or alternative Linux binary package repo or source

Use the command 'alien' to Debianize RPM packages, I mean to create a DEB package from RPM package. This option is not going to work with all the packages, but supports conversion among packages such as Linux Standard Base (LSB), RPM, DEB, Stampede and Slackware.

Some people says the command is some old and it is better to build the source instead of translating the package to be able to install it on Debian.

This installation example works:

Guide:

ProjectLibre

References:

Topic: Installing and using the command 'alien' in Debian. Converting .rpm Packages To Debian/Ubuntu .deb Format.

Web: https://www.howtoforge.com/converting_rpm_to_deb_with_alien

File: Converting .rpm Packages To Debian_Ubuntu .pdf

Debian binaries

Software sources available:

- Distribution DVDs
- Netinstall CD
- Localhost or Local Area Network Debian repository
- Debian repository servers
- Debian PPA repositories
- Web site, P2P and Developers repositories
- Ubuntu PPA (Known exception)

Installation examples (DPKG):

- Google Chrome
- Skype
- Teamviewer

Installation examples (APT):

- VLC
- Firefox
- Screensaver
- Xchat
- VirtualBox

Ubuntu PPA (Known exception).DEB packages compatible with Debian:

- KXStudio

For Debian packaged binaries:

They are recognized by the extension .deb, I.E: package.deb

Use DPKG, the Debian Package Manager.

For Debian packaged binaries to be downloaded from any deb repo:

Use APT, Aptitude or Synaptic (GUI).

In our case we are going to use this method to save time at installing the most of the system software, except for when the packages aren't in the Official Debian repos or the experimental.

I recommend not to install experimental packages as possible on this kind of Desktop to do not compromise its integrity.

We are not going to use this method to install the Web apps and libraries, to match the Apple's Mac engineering/development Team.

Flatpak, X-Apps or Snap packages Debian equivalent

Software sources available:

- Yet unavailable

Installation examples:

- Flatpak (Is not distro specific, previously called xdg-app, available in Debian Experimental repos)

- x-apps (Linux Mint)

- snap (Ubuntu)

They work like Android apps. They don't depend on any or many of the options like the desktop environment, libraries or libraries version like GTK 3.x, the distro or certain packages.

They are bigger than programs because they contain all the packages required to work out of the box, hopefully in any distribution/version.

There isn't official Debian development for these apps yet, but no matter they lack of separate installer like the Ubuntu's installer there are similar applications working similarly, like the listed in the section "Full compatible/portable Linux/Debian binaries".

Snap example:

```
$ snap find
```

```
$ sudo snap install <package-name>
```

References:

Web: <http://www.webupd8.org/2016/04/ubuntu-1604-lts-to-offer-updates-via.html>

File: Ubuntu 16.pdf

Web: <http://segfault.linuxmint.com/2016/02/the-first-two-x-apps-are-ready/>

File: The first two X-Apps are ready Segfault.pdf

Full compatible/portable Linux/Debian binaries

This type of applications distribution includes approaches like "Self contained installers" and "Applmage". ON advantage is that you can install the newest app version available, no matter whether you have old or new system libraries required to run the app. One disadvantage is the security, since the point of view that the distro application passed stability and other security tests. A second disadvantage is the space, due to every application have its own libraries copy, which differs of the shared libraries approach.

References:

Guide:

Static, shared dynamic and loadable libraries

Software sources available:

- Web site, P2P and Developers repositories
- Ubuntu PPA (Known exception)

Installation examples:

- Netbeans IDE
- Sublime Text Editor
- Aptana Studio 3 (Has a internal Java, Hot Spot)
- Java (Cross platform applications)
- Ruby (Cross platform applications)
- C++ (Cross platform applications)
- Some Themes, Icons and Cursors packages like Numix (For Jessie use Ubuntu 14.04 package)
- Natron
- AMD APP SDK

Requirements:

Make sure that the binaries are compatible with your distro, distro version and architecture (32 or 64 bit) before proceed with any installation, and that if your system meets all the dependencies required, otherwise, the app is not going to work.

Download:

This package binaries are fetched from somewhere else rather than the Official Debian repos.

Package distribution:

Many of these apps come as self contained tar archive with extensions such as *.tar.xz and a self-contained installer in Bash. In the case just extract the tarball file and then run the Installer or directly the executable binaries.

Java, Ruby and similar languages:

The default Debian installation has a free version of Java runtime to run Java applications, but to run Ruby applications you have to install ruby runtime. One method to install Ruby runtime is by installing KDE. See the KDE installation guide ([LINK](#)).

Self contained installers:

References:

Web: <http://www.matteomattei.com/create-self-contained-installer-in-bash-that-extracts-archives-and-perform-actitions/>

File: Create a self-contained installer in Bash Matteo Mattei.pdf

Applimages:

References:

Topic: Linux apps that run anywhere

Web: <http://appimage.org/>

File: AppImage _ Linux apps that run anywhere.pdf

Topic: List of Apps that use AppImage

Web: <https://bintray.com/probono/AppImages>

File: AppImages - Generic – Bintray.pdf

Source Code installation

Debian source code

Software sources available:

- Localhost or Local Area Network Debian repository
- Debian repository servers
- Debian PPA repositories

Installation examples:

- Apache2. Patching and recompilation or simple compilation from sources will improve security if made adding new security module to enable xinit to manage the service to add anti DDOS ability.
- wine-development. Install development version of Wine from patched source will allow the version to use the winetricks package.
- Video drivers. Compilation and maybe patched of proprietary drivers are sometimes requirements to make them work.
- Linux kernel. Debianized kernel is required to add new functionality to the system.

For Debian source code:

You should use APT to download, manage the compilation (building process) and install packages. Puritan users install all the software they can this way. Unfortunately, not every package source we want is available. My only excuse to install binaries directly from the Debian repos is I am saving time by skipping the source compilation to complete the system environment as fast as possible.

I don't have any examples yet, but almost all packages in the Official Debian repositories installed by APT have sources available to download, build and install in one step.

This method is very useful in servers to compile custom package versions with extra or modified features. It's the case of Apache2 which can be compiled from a scratch or be patched before the recompilation to add new Apache2 modules, like modules required to run a number of the Apache2 service instances in parallel to add them to xinetd.

Don't confuse builds from Debian source code and builds from source code. For example, going back to Apache, whether your choice is building from source you have to choose between Debian or another source code. In case of source code analyze the pros and cons of the code chosen.

Mini Howto - Building a Debian package from Debian sources:

Before installing Debian source you need to compile the code, and next I recommend to create a DEB package to install with DPKG.

To succeed, use one of these tutorials. I ordered every reference from the most appropriate to the less appropriate to me:

References:

Topic: Building Debian packages from Debian sources - tutorial.

Web: <https://wiki.debian.org/BuildingTutorial>

File: Building Debian packages from Debian source tutorial.pdf

References:

Topic: Working with Debian source packages.

Web: <https://wiki.debian.org/SourcePackage>

File: Working with Debian source packages.pdf

References:

Topic: Building Debian packages with sbuild, the tools I use to build from Debian forks source code.

Guide:

Build packages from source

SimpleScreenRecorder

References:

Topic: Look for the topic Deb builder in this guide.

Guide: Source code & Debian forks source code

References:

Topic: Rebuild a package using one of the methods described in the Debian package maintainers

Guide:

- Complete (re)build

Output:

pts (+ the screen ID)

pts (+ the screen ID)

...

Enter to specific screen (resume, if status = detached):

Use the screen name and ID as flag to parameter -r

```
# screen -r 1860.pts-0.foobar
```

Exit (again) from screen (to avoid nesting screens behind screen):

CTRL + A + D

Output:

Detached from pts (+ the screen ID)

Detach from a screen from screens list:

```
# screen -d 1860.pts-0.foobar
```

List existent screens:

```
# screen -r
```

Look for condition (Detached)

Output:

pts (+ the screen ID) (Detached)

pts (+ the screen ID) (Detached)

Remove a screen:

Enter to a screen and then:

```
# exit
```

Kill screen based on process id:

```
screen -r
```

```
# kill <PID>
```

Example:

```
# kill 4169
```

New screens with custom name:

```
# screen -S <name>
```

```
# screen -S ntp
```

After you use custom names to create screens the option # screen -r will don't work and show an error message.. by then to list use:

```
# screen -list
```

Site Advisors and page checkers

Bitdefender TrafficLight:

From the Google Chrome web store:

"Bitdefender TrafficLight is a Chrome and Firefox browser EXtension which adds a strong and non-intrusive layer of security to your browsing experience.

TrafficLight is taking the security-while-browsing concept where it belongs: in the browser.

This extension will add a strong layer of security over your browsing experience without inducing speed penalties or system resource consumption. Part of the processing is done in the cloud with some intelligent small engines that make various checks on pages you're visiting enabling you to have top notch protection in case you run into fraud, phishing or simply dangerous websites."

Speedtest

Installation:

```
$ apt-cache search speedtest-cli
$ sudo nano /etc/apt/sources.list
$ sudo apt-get update
$ sudo apt-get install speedtest-cli
$ speedtest-cli
$ speedtest-cli --bytes --share
```

Tcpdump

For ping information, read the guide: "Ping"

Who pings me?:

Reference:

Web: <http://askubuntu.com/questions/430069/how-to-monitor-who-is-pinging-me>

File: Who is pinging me.pdf

Useful commands:

Pending:

Paste here book "Entrenamiento profesional en redes" (Spanish)

Send results to screen and file like this:

Web: <http://unix.stackexchange.com/questions/4782/how-to-pass-the-output-of-one-command-as-the-command-line-argument-to-another>

File: bash - How to pass the output of one command as the command-line argument to another - Unix & Linux Stack Exchange.pdf

Web: <http://askubuntu.com/questions/38126/how-to-redirect-output-to-screen-as-well-as-a-file>

File: bash - How to redirect output to screen as well as a file in a yet-to-be-created directory - Ask Ubuntu.pdf

Web: <https://felixmilea.com/2014/12/running-bash-commands-background-properly/>

File: Running bash commands in the background properly - Felix Milea-Ciobanu.pdf

TOR

TOR (The Onion Router), is the largest implementation of onion routing, which is a method for transmitting data anonymously over the Internet. Run by volunteers, there are approximately a thousand Tor proxy servers on the Internet that provide the routing paths.

- Debian and Tor Services available as Onion Services:

The Debian project is working with the Tor privacy network to set up anonymous access to Debian's infrastructure. "We, the Debian project and the Tor project, are enabling Tor onion services for several of our sites. These sites can now be reached without leaving the Tor network, providing a new option for securely connecting to resources provided by Debian and Tor. The freedom to use open source software may be compromised when access to that software is monitored, logged, limited, prevented, or prohibited. As a community, we acknowledge that users should not feel that their every action is trackable or observable by others. Consequently, we are pleased to announce that we have started making several of the various web services provided by both Debian and Tor available via onion services." A list of Debian services available through the Tor network and how to access them can be found in the announcement.

References:

Web: <https://bits.debian.org/2016/08/debian-and-tor-services-available-as-onion-services.html>

File: Bits from Debian - Debian and Tor Services available as Onion Services.pdf

- TorBrowser

I am gonna use TOR for secure navigation as Edward Snowden recommends, but he says you should use the distribution Tails.

(Pending)

Traceroute

Pending

Wireshark

Pending

Improve the security of the system since the network

Kali

Pending

Network Firewall

Pending

OSSIM

IDS

IPS

SIEM

You can run it on your network to help protecting you from almost everything.

Main source for Alien Vault

References:

File: OSSIM - Wikipedia, the free encyclopedia.pdf

Directory: ../IT Securty/AlienVault_OSSIM

REMnux

Pending

Esteban Herrera

He is a Computers Consultant. Went to Intel to teach Linux. Worked for important media companies in Costa Rica, as Radio U and the newspaper La Nación. Received the National Award of Films. In his spare time creates new electronic and rock music. You can send a fan mail or Tweet him @esteban_records

Meet the software installation sources and options available
Dominate activities regarding the software installation and management
Build a Minimal Debian Live DVD from source code
Set up and maintain a perfectly functional operating system
Complete a decent environment by testing the most of software installation options available
Learn to install web development software for the MacOS
Security first



Esteban Herrera

He is a Computers Consultant. Went to Intel to teach Linux. Worked for important media companies in Costa Rica, as Radio U and the newspaper La Nación. Received the National Award of Films. In his spare time creates new electronic and rock music. You can send a fan mail or Tweet him @esteban_records