

# Advance Linux Usage

Author: Hackuin[@]ymail[.]com

Date:2011/07/05



## Notes:

- This article is no means to/or/an attempt to compare Linux with any other operating systems [or] any other Linux distrubition.
- This is derived/compiled from my personal experience and few of my buddies experiences at <http://www.criticalsecurity.net>.
- This article purely assumes, you are have a basic understanding of setting up any Linux distribution.
- Examples given in the article are tested on Ubuntu 11.04 [Natty]
- This article also covers few understanding and approaching towards Linux Certifications.
- Grab your Coffee/Beer, this article is huge one.
- Enjoy reading!! :]

## Disclaimer:

- I am no expert, and does not guarnty you that, by reading this article you will/or become a Linux expert. It solely depends upon you and your usage and your likes and comfortabilites.
- Do not think of this as a stand point or reference for any kind of Linux certifications, this is purely meant for customizing Linux for personal comfortabilites.

One idiotic thing, actually made me to start exploring Linux OS. Ages ago, when Fedora wasn't part of Red-Hat, and the times when, "ping of death" is soo facinating, and I was total n00b at Info Security. I read some article about ping of death, and after reading few of the comments over that article, one of the person commented something like, "on linux environment, you don't need to mentioned -t option with ping utility for continious pings to a host" and little more praising of linux environment for programming matter and all, so I taught of trying out Linux, and it was "Fedor Core 3" which had 4 CD, eventough DVD's writer at those time was much higher priced. And even downloading them was like, living you pocket roll. 1 hour charges for internet cafe was like, Rs.60/- and internet speed even sucks quite harder.

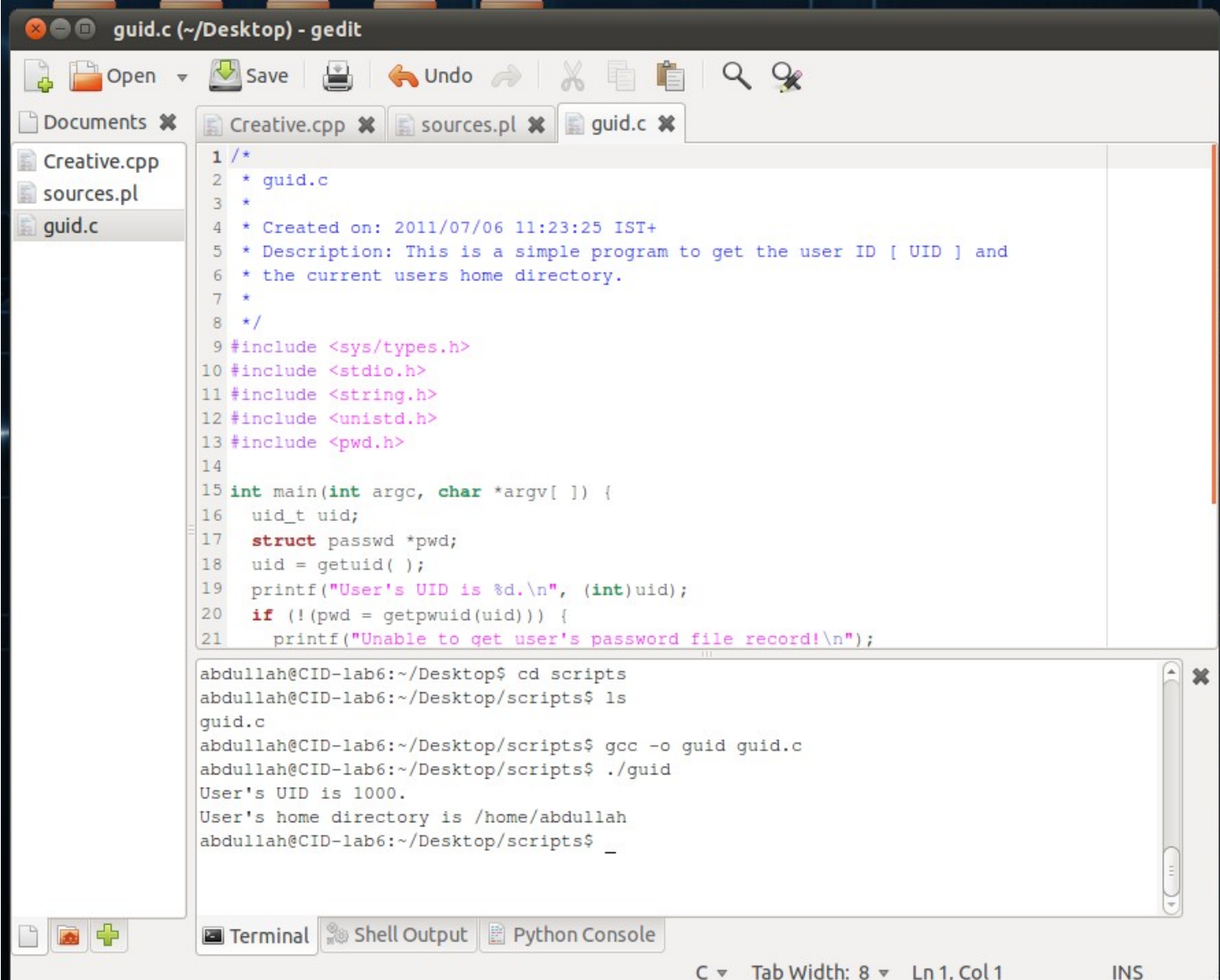
The point is your dedications towards Linux, make it note and note it very well, neither you are born Linux expert nor does the Linus Trovards. It takes time and patience, the beauty of Linux lie's in itself, I mean, its regular usage, accomplishing the problem you face while using Linux Environment. However, as pointed in notes, this article assumes you are average user of Linux. So, be it. Let start exploring the Linux Comfortableness with slow pace.

As you have been reading this article on a infoSec/Hacking relateing forums, the very important aspect of infosec/Hacking is the knowledge about programing. I assume you spend little atleast little time over programing environment. If you are not, start choosing a programing language and start learning it right now, its very important aspect of infosec/hacking. So, as you spend time on programing/coding. You need confortabilites. Lets start with it.

### The Glory of Gedit [ Makeing it closer to IDE ]:

Gedit is a default text editor on Gnome environment. And by default it has a very simple look as a Notepad. As, I do spend my time on notepad programming/codeing, I need to confort my self, consider it as lazyness or something, but when you start using the way I do, you will love it very quickly. Usually, when I code something, I simple hate jumping to terminal and compiling and running. With customising, I get a little terminal under my bottom "pane". If you don't like the pane everytime, you can just Ctrl+f9 to show/hide. Auto-completion plugin save's my hell out of time, and Bracket completing plugin stiff my references for brackets, used with Highlight matching brackets. Snippets are very very hand for simple example: If you want to include the GPL License just customise your snippet to short cut key/or just type gpl and tab creates GPL License. :]

Below is a screen shot of my Gedit.



The screenshot shows the Gedit text editor with the file `guid.c` open. The editor has a menu bar with `Open`, `Save`, `Undo`, and other options. The left sidebar shows a file list with `Creative.cpp`, `sources.pl`, and `guid.c`. The main editing area contains the following C code:

```
1 /*
2  * guid.c
3  *
4  * Created on: 2011/07/06 11:23:25 IST+
5  * Description: This is a simple program to get the user ID [ UID ] and
6  * the current users home directory.
7  *
8  */
9 #include <sys/types.h>
10 #include <stdio.h>
11 #include <string.h>
12 #include <unistd.h>
13 #include <pwd.h>
14
15 int main(int argc, char *argv[ ]) {
16     uid_t uid;
17     struct passwd *pwd;
18     uid = getuid( );
19     printf("User's UID is %d.\n", (int)uid);
20     if (!(pwd = getpwuid(uid))) {
21         printf("Unable to get user's password file record!\n");
```

At the bottom of the window, there is a terminal pane showing the execution of the program:

```
abdullah@CID-lab6:~/Desktop$ cd scripts
abdullah@CID-lab6:~/Desktop/scripts$ ls
guid.c
abdullah@CID-lab6:~/Desktop/scripts$ gcc -o guid guid.c
abdullah@CID-lab6:~/Desktop/scripts$ ./guid
User's UID is 1000.
User's home directory is /home/abdullah
abdullah@CID-lab6:~/Desktop/scripts$ _
```

The terminal pane has tabs for `Terminal`, `Shell Output`, and `Python Console`. The status bar at the bottom indicates `Tab Width: 8`, `Ln 1, Col 1`, and `INS`.

## Customizing Gedit:

By default, Gedit looks like a normal text editor like notepad, and the plugin available are very little. First thing need to do is, install the additional plugins for Gedit.

```
user@server:~$sudo apt-get install gedit-plugins
```

After installing the gedit-plugins, you can navigate and customize the gedit.

1)Select "Edit -> Preferences -> plugins"

And select the appropriate plugins according to your confort.

Importants plugins:

- a)Code Comment
- b)Commander
- c)Embedded Terminal
- d)External Tools
- e)File Browser Pane
- f)Insert Date/Time
- g)Multi-Edit
- h)Python Console
- i)Session Saver
- j)Snippets
- k)Word Completion

2)Select "View" and check "side pane" & "bottom pane".

3)Have a look at "Tools -> Manage Snippets.." and customise your snippets according to your pefered programming language and comforts like shortcut keys, which reduces your hell out of pains while codeing.

Beside all this you can opt for the Emac editor, it is not installed by default, and its interface sucks really hard atleast for me.

Next, This is actually a common question on Linux environments, when you install a Linux distribution, the partisions are not auto mounted when you restart your machine. Beside that, I have few custom \*.iso images lying around which I almost use them regularly example, is sumari.

## Basic concept:

If you have a iso/image files you can mount them to a location (directory).

For example:

I have file name ccie.iso, I use this image file allmost everyday. It is in my home directory. I can issue a command.

```
user@server:~$sudo mkdir /media/mounts/ccie
user@server:~$sudo mount -o loop ccie.iso /media/mounts/ccie
```

The iso image is mounted on /media/mounts/ccie directory you can navigate and use the files in the directory. But, the problem is, every I had to issue this command, I mean, when I boot my PC, I have issue this command again to navigate and use the sumari files... There are couple of work arounds for this.

### Simple one actually ironic one:

Once you mount the iso image, you just copy files to a folder. And then just delete the iso imagefile and use those files as simple files then.. lulz.

### Nasty one, make you or force you to learn:

However, if you don't wanna do it, as I used to download allot of huge sized iso images, as the download itself takes a lot of time and then again I need to mount it and start copying the files to a folder, one time, its fine, but everytime its not good and time consuming one, soo, what I do is, I have created a bash script log-time ago made it a startup script.

At the time, I mean after downloading iso files, I mount the downloaded file and use it. But, inseed of copying the file to a folder which take time (consider an iso image of 3GB). I just set my scripts variables, which dont even take few seconds. And its done and always available for you.

Script:

```
#!/bin/bash
: '
* Auto-mounts.sh
*
* Date: 2009/16/09 23:18:49 IST 2009
* Author: Hackuin[@]Ymail[dot]com
* Description: Auto mounts the specified iso images. Please change the specific
* Directory or change the $DIR or check your $PATH carefully.
*
* Copyright (C) 2009 under GNU GPL Licence.
*
* '
DIR=/media/60GB/linuxiso                                ## Change as per your Directory PATH.
FILENAME=ccie.iso                                       ## Change as per your file name.
NDIR=/media/mountpoints/network                         ## Change as per your mountpoint.

C_R="\e[01;31m"                                         ## Colors Red,Darkblack & Green.
C_B="\e[01;30m"
C_G="\e[01;32m"
C_END="\e[00m"

file $NDIR >/dev/null                                   ## Checking if directory EXIST's or NOT.
if [ "$?" != 0 ]; then
    mkdir $NDIR
fi

echo -e "Mounting *.iso images....."

mount -o loop $DIR/$FILENAME $NDIR    ## Mounting the iso image.

if [ "$?" = 0 ]; then
    echo -e "Mounting" "$C_B$FILENAME$C_END" is mounted to $CB$NDIR$C_END" "\t\t\t$C_G[ OK ]$C_END"
else
    echo -e "Could not Mount" "\e[01;30m$FILENAME\e[00m" "\t\t\t\t\t\e[01;31m[ FAIL ]\e[00m"
fi

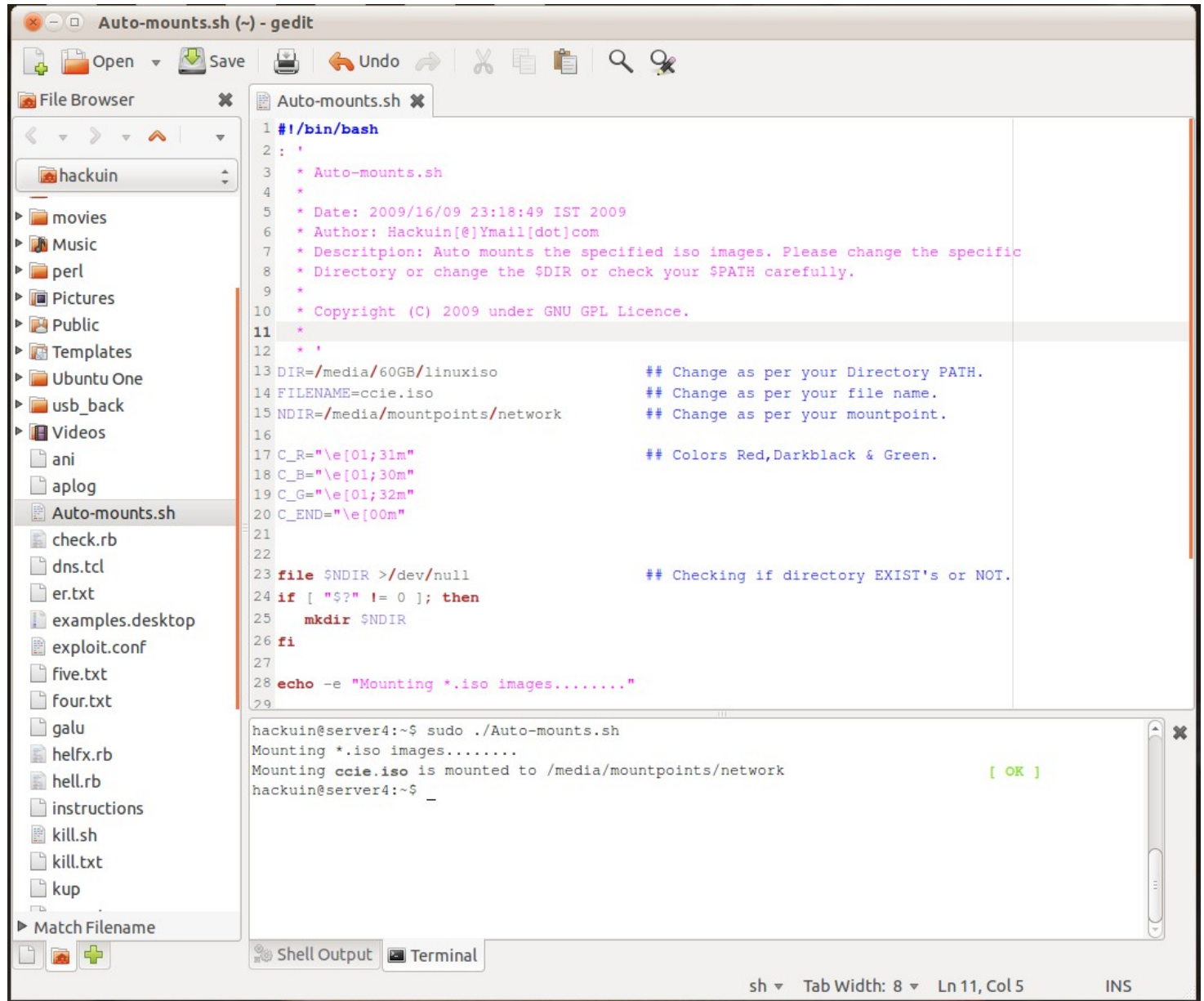
#END
```

Just copy the script to `"/etc/init.d/"` and update rc with defaults once.

```
user@server:~$sudo cp Auto-mounts.sh /etc/init.d/
user@server:~$sudo update-rc Auto-mounts.sh defaults
```

Make sure the file is executable I mean, set +x permission to the file(in our case Automounts.sh).

Below is simple output of the script.



The screenshot shows a gedit window titled "Auto-mounts.sh (~) - gedit". The left sidebar displays a file browser with a list of files and folders, including "hackuin", "movies", "Music", "perl", "Pictures", "Public", "Templates", "Ubuntu One", "usb\_back", "Videos", "ani", "aplog", "Auto-mounts.sh", "check.rb", "dns.tcl", "er.txt", "examples.desktop", "exploit.conf", "five.txt", "four.txt", "galu", "helix.rb", "hell.rb", "instructions", "kill.sh", "kill.txt", "kup", and "Match Filename". The main editor area shows the content of "Auto-mounts.sh", which is a bash script with comments and code. The script includes a header with a date, author, description, and copyright notice. It defines variables for the directory (DIR), filename (FILENAME), and mountpoint (NDIR). It then checks if the directory exists and creates it if it does not. Finally, it echoes a message indicating the mounting process.

```
1 #!/bin/bash
2 :
3 * Auto-mounts.sh
4 *
5 * Date: 2009/16/09 23:18:49 IST 2009
6 * Author: Hackuin[.]Ymail[.]com
7 * Description: Auto mounts the specified iso images. Please change the specific
8 * Directory or change the $DIR or check your $PATH carefully.
9 *
10 * Copyright (C) 2009 under GNU GPL Licence.
11 *
12 *
13 DIR=/media/60GB/linuxiso          ## Change as per your Directory PATH.
14 FILENAME=ccie.iso                ## Change as per your file name.
15 NDIR=/media/mountpoints/network   ## Change as per your mountpoint.
16
17 C_R="\e[01;31m"                  ## Colors Red,Darkblack & Green.
18 C_B="\e[01;30m"
19 C_G="\e[01;32m"
20 C_END="\e[00m"
21
22
23 file $NDIR >/dev/null             ## Checking if directory EXIST's or NOT.
24 if [ "$?" != 0 ]; then
25     mkdir $NDIR
26 fi
27
28 echo -e "Mounting *.iso images....."
29
```

The terminal output at the bottom shows the execution of the script:

```
hackuin@server4:~$ sudo ./Auto-mounts.sh
Mounting *.iso images.....
Mounting ccie.iso is mounted to /media/mountpoints/network
hackuin@server4:~$
```

That was actually out of curiosity way of doing in our own ways, let just use the standard way.

### Holy Glory of fstab:

Ironically, not even our partitions are automounted by default. How will our residing iso images get mounted on start up?  
To mount your partitions [or] iso images, auto-matically, you need to append fstab file.

As above, let us take iso file ccie.iso which is located at /media/60GB/linuxiso/ directory, we just need to type a single commaond [ long one though ] and its.

```
user@server:~$sudo echo -e "/media/60GB/linuxiso/ccie.iso /media/mountpoints/network
udf,iso9660 user,loop 0 0 " >> /etc/fstab
```

And its done.

The reason I posted different methods is, to just make a point, that there always a solution and please concentrate not just on solution, make it prular, I mean, think of solution[b]'s[/b]. That way you will defenetely master the concept.

### **OH! INSTALLED IT AND LEFT:**

This is most amazingly common thing I have noticed on plenty of forums, people start posting about the behaviours of OS.

When you/we install Linux, however, 90% of desktop instalations are kept as standard and default mode. I mean, they just install it through GUI and make the instalation automode. Yep, it quite safer method, if you don't want to mangle around at-least at before the OS is installed. However, you can/should take care of thing with aren't required at-least after installation.

Few things:

- 1) Go to you service setting and un-check the services you don't actually need, like, if you are not using printer, there is no point to start the service, cups/print queue applet on your boot-up, there are plenty of un-needed services. The reason for doing that is optimization, and as this is not an article for optimizing Linux or something, the actual reason is becuae, we are going to installed some applications which actually starts at boot time. So, the more applications/startup scripts/services, higher is the start-up time, why not just removing use-less servies and reduce our statup time?

With Ubuntu Natty[11.04] Unity is introduced as default. There are plenty of glintches, like for example, we use terminal allot, uh.uh.. teminals I would say, If we put/keep the terminal in launcher, we can just click and use the terminal, if we want to get another terminal, if you got click it twice/thrice/zillion time, what you will get is the same terminal. Idiomaticity at best, I read one article, to add "new tab", "new terminal" & "root terminal" to launcer, I mean, when you right click you will have plenty of options like, "open new tab", "open New terminal", "open root terminal" etc., and you know what is the procedure?

```
Copy GNOME-Terminal launcher file to your home directory.cp /usr/share/applications/gnome-terminal.desktop ~/.local/share/applications
```

Open and edit the file using Gedit.

```
gedit ~/.local/share/applications/gnome-terminal.desktop
```

**Find**(Ctrl + f) the following line from the file.

```
OnlyShowIn=GNOME;
```

**Replace** it with the following line.

```
OnlyShowIn=GNOME;Unity;
```

Now, simply copy-paste the following lines towards the **bottom** of the file.

```
X-Ayatana-Desktop-Shortcuts=NewWindow;NewTab;Top;Root;
```

```
[NewWindow Shortcut Group]
```

```
Name=New Window
```

```
Exec=gnome-terminal --window
```

```
TargetEnvironment=Unity
```

```
[NewTab Shortcut Group]
Name=New Tab
Exec=gnome-terminal --tab
TargetEnvironment=Unity
```

```
[Root Shortcut Group]
Name=Root Terminal
Exec=gksudo gnome-terminal
TargetEnvironment=Unity
```

```
[Top Shortcut Group]
Name=Top
Exec=gnome-terminal --command top
TargetEnvironment=Unity
```

Close all Terminal windows, right-click on Terminal launcher icon and **untick** 'Keep In Launcher' option. Finally, browse to the `~/.local/share/applications/` from Nautilus file browser and simply **drag and drop** the new Terminal launcher (file name: `gnome-terminal.desktop`) onto the Unity Launcher. Done.

WTF? O.o, When the terminal is open, why not just use the short-cut key? Alt+f, b (New tab?) and so on..? And, In the first place, set you regular application a short cut key.

Like, I used to set, super+z for terminal, Alt+b for browser. Set your keyboard short cuts, according to your comforts.

One dis-advantage of doing so is the addictions, believe me, I am so much addicted to my Home PC, My Note-book and my Office PC, that, if I use some other persons laptop, or someone else PC, I feel like I am using a strange machine and get frustrated very soon by the behaviour of how it operates, the point is, its not the fault of the that persons PC, its actually, how we made/customize our machines for acting quick, and we are habituated to it.

But, we do so, very rarely don't we. ;]

- 2) If you have understood my first point clearly, if not this second point indicates you, how valid my first point for you.

Few Good Applications:

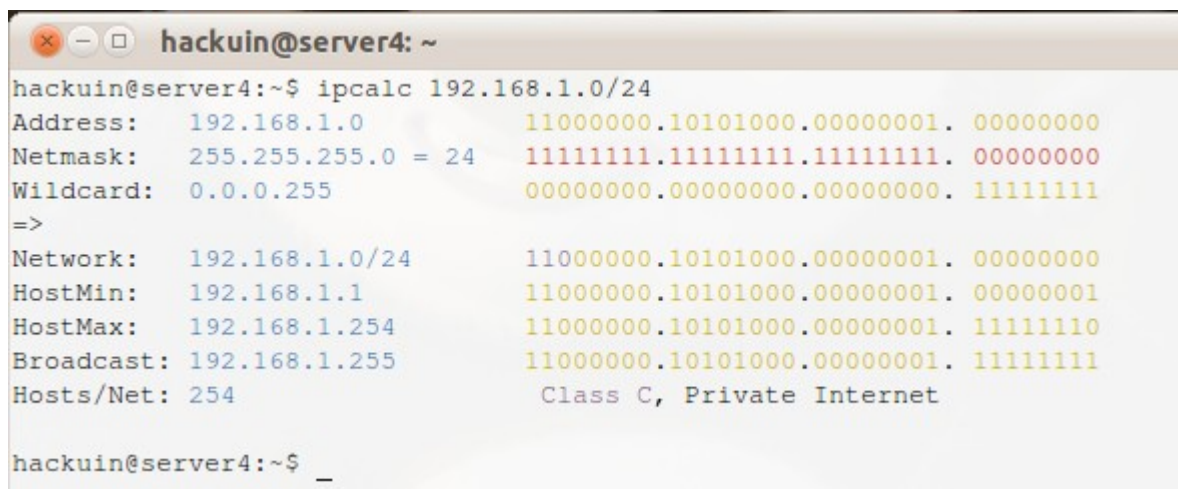
- In the very begining, we made our editor somewhat closer to IDE. But, few people don't prefer that way, IDE is IDE thought, but what I have noticed is, installing IDE, has little performance issue on avaragely configured machines. Tough! If you are just addicted to use IDE, I would say go for Net-beans, that is excellent IDE, I have ever seen. I spend my time coding usually at C++ (as I am in love with C++) from long time. It really make me common sense to install a light weight IDE instaed of installing a consumptions kind of IDE like (Net-bean/Ellepcsies), I use code::blocks IDE and also prefer CodeLite lightweight C/C++ IDE. Basically my point is, if you spend more time on programming any perticular language, don't go for huge IDE like Net-bean/Elipsis, instead look for alternative native language IDE for your language.



- Filezilla, this application is really very very excellent one for sharing/downloading/uploading files via FTP/FTPS/SFTP. A must FTP client for you.
- As I used be busy with either with my IDE/e-books/Videos-tuts (for CCIE), Gmail Notify simply delevs me with important mails. There are plenty of notifier available, but at one point or other, they have glintches either with Memory usage or crash quickly, but, Gmail Notify is simply good enough!!
- Plenty of people use different type of IRC clients, I like pidgin, while there are similar features available with other IRC clients. Its just personal specification, pidgin by default saves all log of chat in ".purple" directory in your home folder.
- Networking is actually beautiful concept, its like my whole life dedecated to it. After completing my CNNA, it was more interested but, took little long for completing my CCSP, and seriously, studing for CCIE and honestly, believe me or not, once you get more involved to netwoking, the more you get addicted to it, Although, going with vendor neutral about netwoking, I found, it quite amazing, when playing around with networking stuff, with Linux OS.

Discussing about every aspect of Linux Networking is beyond scope of this article, However, I am spending few time over few of the important concepts, while being studing for CCNA, one important concept is "subnetting", however, as being in Cisco world we have plenty of methodologies to calculate subnets, however, thats for certification where we don't have access to tools/applications, however, at job/work place, when configuring network/s, we do have quicker methods but not always quicket though, and when people are not from the Cisco/Networking world, like (RHCE's) and for people who are not quick enough at calculating, they can use the tool "ipcalc" for calculating subnets and everything about network and planning.

Below is screen shot:



```

hackuin@server4: ~
hackuin@server4:~$ ipcalc 192.168.1.0/24
Address:    192.168.1.0      11000000.10101000.00000001. 00000000
Netmask:    255.255.255.0 = 24 11111111.11111111.11111111. 00000000
Wildcard:   0.0.0.255        00000000.00000000.00000000. 11111111
=>
Network:    192.168.1.0/24    11000000.10101000.00000001. 00000000
HostMin:    192.168.1.1      11000000.10101000.00000001. 00000001
HostMax:    192.168.1.254    11000000.10101000.00000001. 11111110
Broadcast:  192.168.1.255    11000000.10101000.00000001. 11111111
Hosts/Net:  254              Class C, Private Internet

hackuin@server4:~$ _

```

Ask, a CCNA guy/Network admins the importance of these kind of tools.



Basically, when network comes, especially at Linux world, there stands, a concept call Firewall, which resembles Iptables, however, Iptable is beautiful concept, it desperately takes time to understand and master the Iptables, Iptables is actually a front end of what soo called Net-Filter. If you are a home user, you can start learning about UFW(Uncomplicated Firewall), ufw is quite fasinating for home user, especially for user who have no plan's for serving, Simple two commands and it fairly secured box, at-least at network layer.

```
user@server:~$sudo ufw enable
user@server:~$sudo ufw default deny
```

^those two commands, self explanatory, one it enables firewall, and second command tells ufw to deny all incoming packet (means, if the packet is inbound/initilizing connection to this machine. This rule does not stop anykind of reverse connections. However, we can harden it, consult the Net-filter documentations or learn Iptables rules stratigies.

### **Let me give a real world example:**

My work is jumbled one, and I am kind of lazy, and that turn into my frequent need of my home machine access from my work place.

So, I have configured SSH access to my home PC from my Office, so that, whenever I need something from my home PC, I could go access it and use it and get it.

### **Here is scenario/Problem/s:**

My Home PC, internet connection Dynamic (IP keep on changing frequently at boot time).  
My Office PC, internet connection Static IP. (XXX.XXX.XXX.XXX)

The problem is my Dynamic IP, if my system got restart ( which usually get down due to power cut), I could not get access to my machine, even if It get powered on when power comes up, because of Dynamic IP. So, again, a script comes to rescue, but before looking at script, we would see how actually we accomplished it with out a script.

As a home user I do not expect anyone connecting to me so I do have "default deny" rule.

So, let do it in a standard way:

Assumptions:

My IP Address: 1.2.3.4(Non-reaiable for me, as dynamic)

My Office IP: 9.8.7.6(Static IP)

```
me@mypc:~$sudo ufw disable
me@mypc:~$sudo ufw allow proto tcp from 9.8.7.6 to 1.2.3.4 port 22
me@mypc:~$sudo ufw default deny
me@mypc:~$sudo ufw enable
```

Next, check your status of UFW, with:

```
me@mypc:~$sudo ufw status
```

You would look out something like:

Status: active

To	Action	From
--	-----	----
1.2.3.4 22/tcp	ALLOW	9.8.7.6



```

#: For this again, we need to get the IP assigned to us. Its simple.
#: As, the default user who login after a minute automatically is not a root user.
#: soo, I must need to change firewall rules, so that it updates my dynamic IP.
#: For this I have a another bash script for updating rules, reason for another script is,
#: ufw rules update/delete are interactive, If am not at machine,
#: I could not interact with the firewall prompts.
#: Check the ufw-ssh-home.sh
#: So, for interacting with ufw-ssh.home.sh, I use expect script.
#: Check the expect script.

except auth.ufw

#: Million Dollar question now.
#: We authenticate ISP, upadte our ufw and set ssh to be accessed by our Office IP.
#: As, I am being in office, how would I know, what is the IP_HOME or what my dynamic
#: IP address has changed to?

ifconfig eth0 |grep "inet add" |cut -c 21-34 > /home/hackuin/Ubuntu\ One/ip.txt

#: Although, there are plenty of other options for us, like.
#: We can instead of creating a file/updating the file at Ubuntu One. We could
#: Just, send an email with "mutt/mail".
#: Or, we can use any of logging oriented service, say, a PHP script on a webhost, to log visitors IP.
#: Or, as I do have a OSSEC IDS, I can do a connect to some port which is being watched which alerts.
#: Or, As, we have few system Taps, running around, can also be used.

#:END

```

ufw-ssh-home.sh script below:

```

#!/bin/bash
: '
* ufw-ssh-home.sh
*
* Date: 2011/16/02 12:16:22 IST 2011
* Author: Hackuin[@]Ymail[dot]com
* Descriptpion: Script to be used with or source to expect script for
* interacting with ufw prompts and updating the ufw rules for
* accessing the home PC from Office.
*
* Copyright (C) 2011 under GNU GPL Licence.
*
* '

IP_OFFICE=9.8.7.6
IP_HOME=`ifconfig eth0 |grep "inet add" |cut -c 21-34`

sudo ufw disable          #sudo will prompt for password.
sudo ufw delete 1         #Delete the rule 1, which is actually old IP_HOME.

sudo ufw allow proto tcp from $IP_OFFICE to $IP_HOME port 22
sudo ufw default deny
sudo ufw enable

#: END

```

Expect script auth.ufw is below:

```
#!/usr/bin/expect

#: auth.ufw
#: Date: 2011/16/02 12:40:56 IST 2011
#:
#: Description: Script to change Firewall ( UFW ) RULES, and to interact with
#: ufw prompts.
#: Used by the home-ssh.sh script. For escalating privileges.
#: spawns ufw-ssh-home.sh script to update rules of UFW.
#:
#: Copyright (C) 2011 under GNU GPL Licence.

set Passwd MyPassword
set yesno y

spawn ./ufw-ssh-home.sh
expect "password:"
send "$Passwd\r"
expect "(y|n)?"
send "$yesno\r"

expect eof
```

The above scripts demonstrates the issues/problem to access my Home PC from my office, with my specific scenario, usually we come across this type of similar issues. It solely depends upon your approach. And, yes, you need to make the script start the script at your startup session.

More Ironically I could just have a remote desktop. But, again, I definitely, need a script to know my changed dynamic IP, if power intrupts.

There are infinite number of applications available, mentioning all of them over here is beyond the scope of this paper, however, the problem is people don't explore there software center, I recommend you to spare a little time browsing all the applications/tools available in your software center. You will be amazed.

For the addition notes, I had few post over different forums about Linux, which may interest you, links below:

[Linux + Security](#)

[People Never understood what it is](#)

- 3) Being a Linux user, you must and should start learning a shell scripting. It saves your hell out of pain in your ass. However, you won't use it often, as a home user, but if you have a little taught of administrating Linux environment, you should have good knowledge in shell scripting. Even though, shell scripting is not a "Rocket Science" to learn. It all require is your command at Linux shell or in other words, experience over your commanding skills of shell [either, C/KSH/BASH etc.], there is not much difference between these different shell scripting except few, for example:

Consider a simple bash script:

```
#!/bin/bash
#
#: for.sh
#: Example of C-Style for loop.
#:
XVAL=1
for((i=3; i>=XVAL; i--))
do
    echo -n "$i...."
    sleep 2
done
echo
echo -e "START"
```

The script is very self explanatory, it loop till, i is greater than or equal to the value of XVAL variable. Then echo's the value at i and waits 2 seconds and iterates again, output would be

```
3.... 2.... 1....
START
```

However, if you try to run this script using a C-shell, you will get an error! Because, C-shell does not support "for" loop, instead it has a "foreach" loop.

Few Key references or books:

- Pro Bash Programming
- Linux Shell Scripting Cookbook
- Advance Bash Scripting

I guess, you know where to find these books ;]

#### 4) Glory of X-Forward/SSH GUI:

It is quite facinating thing, Forward all GUI with SSH. Before doing that, first understand, what we are doing. Have you ever run, multiple GUI session, I mean, when you hit Alt+Ctl+F1, All you get is a black screen with a command prompt. And, if you want to switch back to GUI mode, you use, Alt+Ctl+F7, what happens if you hit Alt+Ctl+F9 or F10? Just a blank screen? These can used for another GUI session, for example open up a terminal and check with command.

```
user@server:~$sudo xinit -- :1 vt9
```

You will get a X-server running on TTY9 that is Alt+Ctl+F9 but, with simple basic X terminal, you go ahead and start your GUI session, for example: You can start, fluxbox or you can start Gnome-Session etc., Its gives you full featured, GUI at tty9.  
Now, similarly you could use, This feature for forward the X-server to your machine, with your ssh session.

Try this once:

```
user@server:~$ssh -2fX user@remotehost gedit
```

This will just launch a simple program gedit which you connect to remote system, if you need full GUI, you follow the same on as about that xinit command and have a full GUI with SSH.

Few Interesteing References:

- [Thetan Article](#) ( Criticalsecurtiy Member )
- [Six User Linux System](#)

I Am gona stop now, and will make the Part II for this paper. As, I am going to be busy for next 2 weeks.

Forgive me for spelling  
Hackuin!!  
:END: