Step 2. Install the Dialup modem software

There are 5x .deb files containing apps which need to be installed.
These 5x .deb files are located at :

[Terminal] `$ /usr/share/local-repository/binary/` (or)

[GUI] Places -> Computer -> File System -> usr -> share -> local-repository -> binary

The 5x .deb files are :

(1) libwvstreams4.6-base
(2) libwvstreams4.6-extras
(3) libuniconf4.6
(4) wvdial_1.61
(5) gnome-ppp-0.3.23

These 5x .deb files are in your LM13 installation, only, the applications they contain, are not installed from default. You have to do the installation yourself. Installation must be in the numbered order.

(You don't need to be online to do this.)

When navigating to the **binary** directory via the GUI (easiest way), then double-click a .deb file to begin the install process.

In a new screen, the **Package Installer (PI)** may tell you that the same application is also in a software channel (repository). The **PI** prefers to download from a repository because these files are 'clean'. The **PI** cant be sure that a .deb file located on your system and not sourced from the repository, is clean, and this message is a subtle way of reminding you of this. You can't download from a repository at this point, because you are not online, so disregard this message.

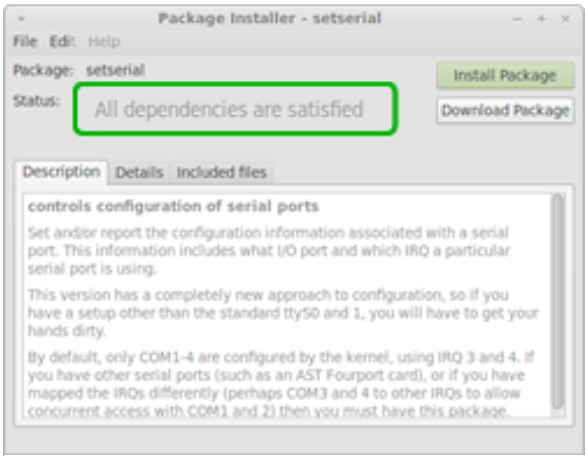
The **PI** will also ask you for your administrator password for each .deb file, so it has permission to install.

Step 2 : IMG1

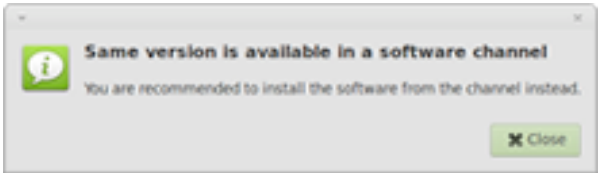
The **binary** folder / directory containing the 5x .deb packages you need to install.

Step 2 : IMG2

The **Package Installer** will tell you if all dependencies are satisfied for each installation. A .deb package will only install if its dependencies are present. It is for this reason that installation order is important.



Step 3 : IMG3
The **Package Installer** asks you to install from a software channel. You cant do this because you are not online. Ignore this message.

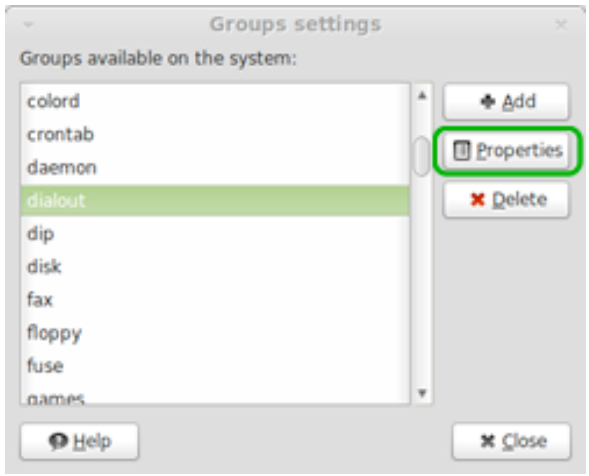


Step 3. (a) Put yourself (user) into dialout, and dip and root groups.
(b) Give yourself permissions to all activities on your machine.

(a) Put yourself into the **dialout**, **dip** and **root** groups.

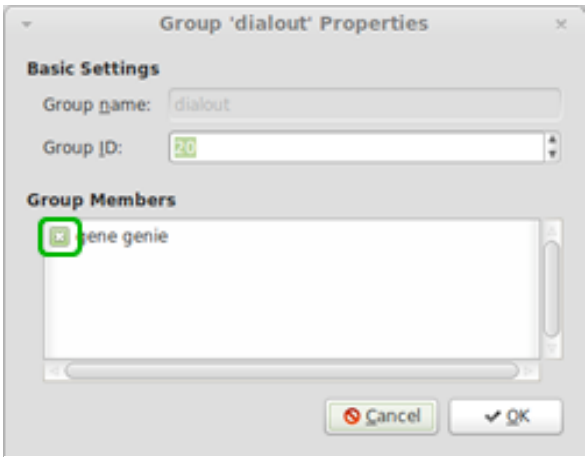
Step 3 : IMG1
Go to **System -> Administration -> Users and Groups -> User settings**. Select **Manage Groups**

Step 3 : IMG2
In the **Group Settings** window, select the **dialout** group and then select **Properties.**



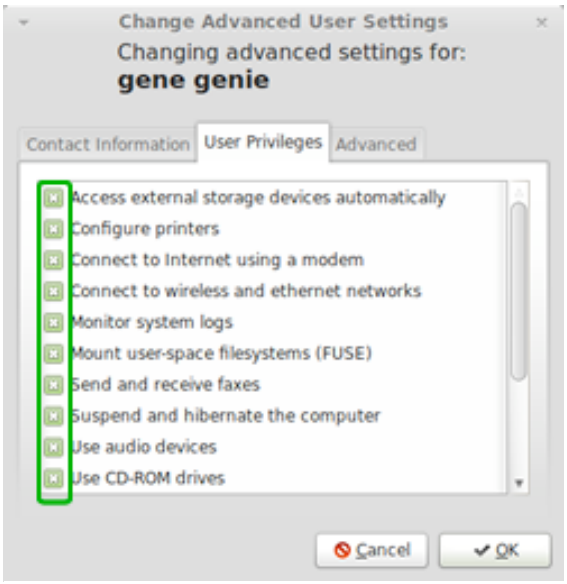
Step 3 : IMG3
In the **Group Properties** window, make sure the current user (you) is checked, **[X]** for the **dialout** group.

Now repeat Step 3 : IMG2 and IMG3 for each of the **dip** and also **root** groups.



(b) Give yourself permissions for all items/activities on your machine. Select **Advanced Settings**, and make sure everything is checked. One of the selections is to be able to use modem or dialout. This is very important to be checked on.

Step 3 : IMG4
Under **Users Settings** select **Advanced Settings**. Then select **User Privileges**. Make sure all items/activities are selected **[X]** for the current user (you).



Note 3

Important files/apps to be aware of.
You may visit these files more than once via the Terminal.

The following files/apps and locations are relevant to successful dialup, so I have listed them here.
You will probably refer back to them.

There is some information floating around that 'there is a second wvdial.conf'.
I have listed it here (file 7 in list), but I have not touched anything to do with file 7 in order to get dialup working.

1	wvdialconf	`/usr/bin/wvdialconf`
2	wvdial.conf	`/etc/wvdial.conf` (created by wvdial)
3	pap-secrets	`/etc/ppp/pap-secrets`
4	chap-secrets	`/etc/ppp/chap-secrets`
5	pppd	`/usr/sbin/pppd`
6	wvdial	`/usr/bin/wvdial`
7	wvdial.conf	`/.wvdial.conf`

Step 4. Check and then change permissions for items 1 through 5 of the list in Note 3 above.

Each of the files numbered 1-5 in Note 3, need their permissions changed.
This change will give all users, all permissions to these files.
For each file I did a **3 step process**.
I always check the original permissions, so that I know what it was if I need to change it back.

a) Check permissions using `ls -l`
b) Change permissions using `sudo chmod go+rwx` (or) `a+rwx` (or) a variation of this as required
All users (a) or group (g) or others (o) get read (r), write (w), execute (x) privileges.
c) Check permissions again using `ls -l`

This is all done via the **Terminal**.

1	wvdialconf	`$ ls –l /usr/bin/wvdialconf` `-rwxr-xr-x 1 root root 42724`	type+enter result
		`$ sudo chmod go+w /usr/bin/wvdialconf`	type+enter
		`$ ls –l /usr/bin/wvdialconf` `-rwxrwxrwx 1 root root 42724`	type+enter result
2	wvdial.conf	`$ ls –l /etc/wvdial.conf` `-rw-r--r-- 1 root dialout 248`	type+enter result
		`$ sudo chmod a+rwx /etc/wvdial.conf`	type+enter
		`$ ls –l /etc/wvdial.conf` `-rwrwxrwx 1 root dialout 248`	type+enter result
3	pap-secrets	`$ ls –l /etc/ppp/pap-secrets` `-rw------- 1 root root 1628`	type+enter result
		`$ sudo chmod a+rwx /etc/ppp/pap-secrets`	type+enter
		`$ ls –l /etc/ppp/pap-secrets` `-rwxrwxrwx 1 root root 1628`	type+enter result
4	chap-secrets	`$ ls –l /etc/ppp/chap-secrets` `-rw------- 1 root root 80`	type+enter result
		`$ sudo chmod a+rwx /etc/ppp/chap-secrets`	type+enter
		`$ ls –l /etc/ppp/chap-secrets` `-rwxrwxrwx 1 root root 80`	type+enter result
5	pppd	`$ ls –l /usr/sbin/pppd` `-rwsr-xr-- 1 root dip 273272`	type+enter result
		`$ sudo chmod go+rwx /usr/sbin/pppd`	type+enter
		`$ ls –l /usr/sbin/pppd` `-rwsrwxrwx 1 root dip 273272`	type+enter result

Quit the **Terminal.**

Step 5. Reboot the computer

Step 6. Create data into wvdial.conf

wvdial.conf is a file which now exists, but it has no modem data written into it.
You can check contents of **wvdial.conf** by opening the file with **pluma** text editor ...

[Terminal] `$ pluma /etc/wvdial.conf` type+enter

To write data into **wvdial.conf**, you must run **wvdialconf** via the **Terminal**.
(Note that the data file is **wvdial.conf**, (with dot) and the application which writes to it, is **wvdialconf** (no dot).

[Terminal] `$ wvdialconf` type+enter

Step 7. Examine and edit contents of wvdial.conf

The text editor **pluma** opens and reads the data-file.

[Terminal] `$ pluma /etc/wvdial.conf` type+enter

The **pluma** text file should read something like this, (below) but not always in the order shown...

```
[Dialer Defaults]
Init 1 = ATZ
Init 2 = ATQ0 V1 E1 SO=0 &C1 &D2 +FCLASS=0
Modem Type = Analog Modem
Baud = 115200
NewPPD = yes
Modem = /dev/ttys0
ISDN = 0
; Phone = <Target Phone Number>
; Password = <Your Password>
; Username = <Your Login Name>
```

The edits you need to do in the text file, are as follows :

Remove : semi-colons and space for last 3 lines shown
Add : Phone Number, Password and Login Name where it is indicated.
Remove : < and > characters
Add a new line and type on that line Stupid Mode = yes

Save file.
Do not change the file name.
Quit **pluma**.

Note 4

The instructions in Note 4 will not get you connected.
Instructions in Note 4 WILL write nameserver data to the current file located at /etc/ppp/resolv.conf.
This is important to Step 8 which follows.

I have a detailed Note 4 because many ppl seem to end up with one of the results listed here in Note 4, and get *somewhat* connected, but not completely, (as described here).

At this point you can try to connect using **Applications -> Internet -> Gnome-PPP**.

a) In the first **Gnome-PPP** screen, you need to enter your correct dial-up phone number, and username, and password.

b) Under the **Setup -> Modem** tab, you need to **Detect modem**, and it should be found under /dev/ttyS0.

c) Try to **Connect** via **Gnome-PPP** GUI at this point.
You should be able to connect for just a few seconds, and then **Gnome-PPP** **will terminate itself**.

d) Under the **Setup -> Options** tab, you need to check **[X] Ignore terminal settings (Stupid mode)**.

e) Try to connect via **Gnome-PPP** GUI at this point.
Gnome-PPP should connect, the connection timer should count, but there will be **no data traffic**.

Quit **Gnome-PPP**.

Carrying out this process has written nameserver data into **/etc/ppp/resolv.conf**.
This will be useful to **Step 8**.

Step 8. Create a file called resolve.conf in the /etc directory, write nameserver data to that file.

(a) Generate and copy the nameserver data

Correct nameserver data needs to be found in the **resolv.conf** file, which needs to in the **/etc** directory, not anywhere else, and not in **/etc/ppp/**

I had a **resolv.conf** file located here... **/etc/ppp/resolv.conf**
Useful to know, but it's not in the **/etc** directory, so is not useful to maintain a dialup connection.

Open this same file with the text editor **pluma**...

[Terminal] `$ pluma /etc/ppp/resolv.conf` type+enter

A **pluma** text file is opened, and the **resolv.conf** file is empty, unless you completed **Note 4.**

IMPORTANT : Follow Note 4 above, to generate nameserver data into /etc/ppp/resolv.conf.

Open **resolv.conf** with the text editor **pluma** after completing **Note 4**...

[Terminal] `$ pluma /etc/ppp/resolv.conf` type+enter

In a **pluma** text file you should see something like this...

```
nameserver 203.xx.xxx.xx
nameserver 201.xx.xxx.xx
```

Copy these data via **Edit -> Copy**

Quit **pluma.**

(b) Fix permission to /etc and write resolv.conf into /etc folder.

The **/etc** directory will have 'permission denied' to copy files into it, or create files in it.
Examine the permissions, allow all permissions, then examine the permissions again.
If necessary you can change the permissions back after completing this step.
This is all done in the **Terminal**.

`$ ls -l /` `drwxr-xr-x 148 root root ....etc`	type+enter result : you will get a list of all files and directories within the root directory. The listing for **/etc** directory will look like this.
`$ sudo chmod go+w /etc`	type+enter
`$ ls -l /` `drwxrwxrwx 148 root root ....etc`	type+enter result : you will get a list of all files and directories within the root directory. The listing for **/etc** directory will look like this.

(c) Write resolv.conf and its contents into /etc folder.

You now have permission to write a file to the **/etc** folder.
A useful way to create a small file is to use the **cat** utility.

`$ cd /etc`	type+enter result : moves you into the **/etc** directory where subsequent work is to be done.
`$ cat > resolv.conf`	type+enter result : **cat** command creates **resolv.conf** file, and places following **Terminal** input into that file.
`nameserver 203.xx.xxx.xx` `nameserver 201.xx.xxx.xx`	Select **Edit -> paste** to paste the nameserver data into the **Terminal** screen, and into the **resolv.conf** file that **cat** has just created.
`Control + D`	press result : terminates input and closes file.
`$ pluma resolv.conf`	type+enter result : **pluma** opens **resolv.conf** file in new window. File should read something like... nameserver 203.xx.xxx.xx nameserver 201.xx.xxx.xx
`$ ~`	type+enter result : Go back to home or root directory
`$ sudo chmod go-w /etc`	return the **/etc** directory permissions to its normal state

Step 9. Connect using Gnome-PPP

You should be able to connect now, the connexn should be stable, AND you should get data traffic.