# Getting Linux Online with "Reliance"

Sunil Beta

November 10, 2004

**Abstract**

This document intends to be a HowTo targeted at users of Linux on installing internet connectivity with their "Reliance India" phone. The major portion of this document is derived from an earlier, more comprehensive HowTo on the same topic. *This document is being written to help a friend get started out on Linux and use his Reliance phone to get connected.*

## 1 Prerequisites

### 1.1 You

Some prior experience in using computers, a minimal comfort level with using the "terminal" is necessary. Any *nix experience, programming experience is a plus.

### 1.2 The Phone

**Models supported by Reliance's "rconnect-cmdline" script** Here's a list of phones that are supported by the script. (rconnect-cmdline-1.0.tar)

- GTRAN GCP 4020
- KYOCERA 7135
- LG LSP -340E
- LG LST 250
- LG RD 2030
- LG RD 2130
- LG RD 2230
- LG RD 7130
- SAMSUNG SCH A563
- SAMSUNG SCH A603
- SAMSUNG SCH A663
- SAMSUNG SCH N191

If you don't have one of these phones, you can still get it to connect, but probably not without some extra work.

**Cabling** You either get a Serial cable (RS232C D connector) or a USB cable. The former is simpler to use, however if you use a laptop you'll probably be looking for the USB connector. Some models are not supported with USB connectors in which case, you can buy the serial cable and buy a USB2serial connector yourself (most of which use a pl2303 chipset.)

**Activating Internet Services** You will have to explicitly activate internet services to your phone by dialing `*412`. You should receive a confirming response in the next 15 seconds - 2 minutes.

**Setting up Baud Rate** All models seem to support a maximum baud rate setting of 230400 bps, however a more realistic (and practical) setting is 115200 bps which has worked on most of the hardware I have tried it on. The "rconnect-cmdline" script reminds you to do this during installation.

## 1.3 Desktop/PC

Any desktop or laptop running Linux is fine. You should be having a serial (RS232C) connector or a USB connector.

## 1.4 Software

**Linux Distribution** You'll need a Linux distribution installed on your Desktop/PC/Laptop. A Linux Kernel version higher than 2.2 (including 2.4 and 2.6) would suffice for our purposes. This would mean - you could use Debian GNU/Linux 3 (woody) or higher, SuSE GNU/Linux 7.0 (or higher), Redhat Linux 6.x (or higher), Fedora Core (any version). Knoppix, Gnoppix, Ubuntu, any of the newer linux distributions are also good.

In most cases, distribution has already taken care of which versions of the kernel and supporting applications to install. The following paragraphs help you find out which versions have been installed and whether they need to be upgraded before trying to connect to the Internet with your Reliance phone.

**The Kernel** It would be preferable if you have linux kernel v2.4 installed, v2.6 would work great. If you're using v2.4, please try using v2.4.20 or newer. If you are using v2.6, please use a newer release (later than v2.6.3).

Finding out which version of the linux kernel you have is simple:

```
username@hostname> uname -a
Linux hostname 2.6.9 #7 Mon Oct 25 17:27:11 IST 2004 i686 i686 i386 GNU/Linux
```

I use SuSE GNU/Linux 9.0 as my distribution and things seem to work fine for me. I've also tried doing this on a machine running Redhat GNU/Linux 9.0 and things work fine.

You need to make sure that your kernel has support for connecting your cable. If you have a RS232C (Serial) cable, you will require serial support in the kernel. If you have a USB cable, you'll minimally need USB host support to be compiled in.

You can find out the devices currently supported by your kernel.

```
username@hostname> cat /proc/devices
Character devices:
  1 mem
  2 pty
  3 ttyp
  4 /dev/vc/0
  4 tty
  4 ttyS
  5 /dev/tty
  5 /dev/console
  5 /dev/ptmx
  7 vcs
  9 st
 10 misc
 13 input
 14 sound
 21 sg
 29 fb
116 alsa
128 ptm
136 pts
180 usb
188 ttyUSB
254 pcmcia

Block devices:
  1 ramdisk
  3 ide0
  7 loop
 11 sr
 22 ide1
```

This would probably be different on your machine, and also depend on the kernel you are running. But what you would be looking for is "ttyS" which is the serial device support and "ttyUSB" which is usb-serial support. If you need to use a USB cable, you'll still have to get a few more things in place.

**Applications** Here are the applications you will need to have installed, their minimal versions and reccommended versions.

- pppd 2.x or newer

  ```
  username@hostname> pppd --version
  pppd version 2.4.1
  ```

- bash (any version should do fine)

```
username@hostname> bash --version
GNU bash, version 2.05b.0(1)-release (i586-suse-linux)
Copyright (C) 2002 Free Software Foundation, Inc.
```

- minicom (any version should do fine) *needed only for debugging if things go wrong.*

```
minicom version 2.1 (compiled Sep 23 2003)
Copyright (C) Miquel van Smoorenburg.

This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version
2 of the License, or (at your option) any later version.
```

**Upgrading software** If you do not have the reccommended versions of the kernel, or the software, you may need to upgrade them. With most linux distributions, a well established packaging and distribution mechanism is used for this purpose. For eg. in debian, you would use `$ apt-get install pppd` to get the latest possible version. With Redhat (or any rpm based distribution), you would have to locate the *.rpm file that provides ppp support, and use `$ rpm -Uvh ppp-2.4.1.rpm`

If your kernel lacks the devices required as listed earlier(1.4), then you most likely need to build a custom kernel.

Building your own custom kernel is beyond the scope of this document. It is reccommended that you read the Linux Kernel build howto. You can also take a look at one of my presentations on kernel building[1] which also lists other relevant literature that you should read.

## 1.5 Downloads

**"rconnect" users** If you intend to use Reliance's "rconnect" which is distributed under their exclusive propreitary license, you will need to obtain the current release of the script from their website.[2] At the time of writing this document, the current version of the script is 1.0, distributed as **rconnect-cmdline-1.0.tar** on Reliance's page.

**using your own setup** Even if you decide not to use the "rconnect" script, you should be able to connect with ease. You can either have your own script to connect or use a program like "kppp", "rh-internet" to connect. I will present information on writing your own script (borrowed from an earlier howto by Ramakrishnan [3].) and "kppp". Other net connectivity programs should be quite similar to customize.

[1] http://www.codito.com/ beta/kernelbuild/
[2] http://www.relianceinfo.com/Infocomm/Rim/rconnect_dc_linux.html
[3] http://www.hackgnu.org/ril-howto.html

For those of you who have no means of alternately connecting to the net with the same computer, you'll have to request someone to download the "rconnect-cmdline-1.0.tar" file and transfer it to your desktop/laptop/workstation with a CD, a local network link or (in the worst case) a floppy drive.

# 2 Connecting using "rconnect-cmdline"

## 2.1 Obtaining and Verifying "rconnect-cmdline" script

For most users, using the rconnect script will be the quickest way to go about it. You can download it from the location listed above[2]. After downloading, you can ensure that the file has not been corrupted in anyway (this is specific to rconnect-cmdline-1.0.tar).

```
username@localhost> ls -l rconnect-cmdline-1.0.tar
-rw-r--r--    1 username  users      153600 2004-11-09 23:18 rconnect-cmdline-1.0.tar

username@localhost> md5sum rconnect-cmdline-1.0.tar
e37ddf1ad9ce29ba5a6f2a0d63f930d8  rconnect-cmdline-1.0.tar
```

If the MD5SUM of your download does not match `"e37ddf1ad9ce29ba5a6f2a0d63f930d8"`, you may want to a) either try downloading again, or b) confirm that reliance has changed something in their script distribution without incrementing the version.

## 2.2 uncompressing and preparing the "rconnect-cmdline" script

You will now need to uncompress the tar ball you have downloaded. This is the easy part.

```
username@localhost> tar xvf rconnect-cmdline-1.0.tar
rconnect1.0/
rconnect1.0/usbwizard
rconnect1.0/phones/
rconnect1.0/phones/LG_RD_2130/
rconnect1.0/phones/LG_RD_2130/LG_RD_2130_COM/
rconnect1.0/phones/LG_RD_2130/LG_RD_2130_COM/LG_RD_2130_COM
rconnect1.0/phones/LG_RD_2130/LG_RD_2130_COM/LG_RD_2130_COM_ACTIVATE.TXT
rconnect1.0/phones/LG_RD_2230/
rconnect1.0/phones/LG_RD_2230/LG_RD_2230_USB/
rconnect1.0/phones/LG_RD_2230/LG_RD_2230_USB/LG_RD_2230_USB
rconnect1.0/phones/LG_RD_2230/LG_RD_2230_USB/LG_RD_2230_USB_ACTIVATE.TXT
rconnect1.0/phones/LG_RD_2230/LG_RD_2230_COM/
rconnect1.0/phones/LG_RD_2230/LG_RD_2230_COM/LG_RD_2230_COM
rconnect1.0/phones/LG_RD_2230/LG_RD_2230_COM/LG_RD_2230_COM_ACTIVATE.TXT
rconnect1.0/phones/KYOCERA_7135/
rconnect1.0/phones/KYOCERA_7135/KYOCERA_7135_USB/
rconnect1.0/phones/KYOCERA_7135/KYOCERA_7135_USB/KYOCERA_7135_USB_ACTIVATE.TXT
rconnect1.0/phones/KYOCERA_7135/KYOCERA_7135_USB/KYOCERA_7135_USB
rconnect1.0/phones/KYOCERA_7135/KYOCERA_7135_COM/
```

```
rconnect1.0/phones/KYOCERA_7135/KYOCERA_7135_COM/KYOCERA_7135_COM
rconnect1.0/phones/KYOCERA_7135/KYOCERA_7135_COM/KYOCERA_7135_COM_ACTIVATE.TXT
rconnect1.0/phones/LG_LSP_-340E/
rconnect1.0/phones/LG_LSP_-340E/LG_LSP_-340E_COM/
rconnect1.0/phones/LG_LSP_-340E/LG_LSP_-340E_COM/LG_LSP_-340E_COM
rconnect1.0/phones/LG_LSP_-340E/LG_LSP_-340E_COM/LG_LSP_-340E_COM_ACTIVATE.TXT
rconnect1.0/phones/LG_LST_250/
rconnect1.0/phones/LG_LST_250/LG_LST_250_COM/
rconnect1.0/phones/LG_LST_250/LG_LST_250_COM/LG_LST_250_COM
rconnect1.0/phones/LG_LST_250/LG_LST_250_COM/LG_LST_250_COM_ACTIVATE.TXT
rconnect1.0/phones/GTRAN_GCP_4020/
rconnect1.0/phones/GTRAN_GCP_4020/GTRAN_GCP_4020_USB/
rconnect1.0/phones/GTRAN_GCP_4020/GTRAN_GCP_4020_USB/GTRAN_GCP_4020_USB
rconnect1.0/phones/GTRAN_GCP_4020/GTRAN_GCP_4020_USB/GTRAN_GCP_4020_USB_ACTIVATE.TXT
rconnect1.0/phones/SAMSUNG_SCH_A563/
rconnect1.0/phones/SAMSUNG_SCH_A563/SAMSUNG_SCH_A563_USB/
rconnect1.0/phones/SAMSUNG_SCH_A563/SAMSUNG_SCH_A563_USB/SAMSUNG_SCH_A563_USB
rconnect1.0/phones/SAMSUNG_SCH_A563/SAMSUNG_SCH_A563_USB/SAMSUNG_SCH_A563_USB_ACTIVATE.TXT
rconnect1.0/phones/SAMSUNG_SCH_A603/
rconnect1.0/phones/SAMSUNG_SCH_A603/SAMSUNG_SCH_A603_USB/
rconnect1.0/phones/SAMSUNG_SCH_A603/SAMSUNG_SCH_A603_USB/SAMSUNG_SCH_A603_USB
rconnect1.0/phones/SAMSUNG_SCH_A603/SAMSUNG_SCH_A603_USB/SAMSUNG_SCH_A603_USB_ACTIVATE.TXT
rconnect1.0/phones/SAMSUNG_SCH_A663/
rconnect1.0/phones/SAMSUNG_SCH_A663/SAMSUNG_SCH_A663_USB/
rconnect1.0/phones/SAMSUNG_SCH_A663/SAMSUNG_SCH_A663_USB/SAMSUNG_SCH_A663_USB
rconnect1.0/phones/SAMSUNG_SCH_A663/SAMSUNG_SCH_A663_USB/SAMSUNG_SCH_A663_USB_ACTIVATE.TXT
rconnect1.0/phones/SAMSUNG_SCH_N191/
rconnect1.0/phones/SAMSUNG_SCH_N191/SAMSUNG_SCH_N191_USB/
rconnect1.0/phones/SAMSUNG_SCH_N191/SAMSUNG_SCH_N191_USB/SAMSUNG_SCH_N191_USB
rconnect1.0/phones/SAMSUNG_SCH_N191/SAMSUNG_SCH_N191_USB/SAMSUNG_SCH_N191_USB_ACTIVATE.TXT
rconnect1.0/phones/LG_RD_2030/
rconnect1.0/phones/LG_RD_2030/LG_RD_2030_USB/
rconnect1.0/phones/LG_RD_2030/LG_RD_2030_USB/LG_RD_2030_USB
rconnect1.0/phones/LG_RD_2030/LG_RD_2030_USB/LG_RD_2030_USB_ACTIVATE.TXT
rconnect1.0/phones/LG_RD_2030/LG_RD_2030_COM/
rconnect1.0/phones/LG_RD_2030/LG_RD_2030_COM/LG_RD_2030_COM
rconnect1.0/phones/LG_RD_2030/LG_RD_2030_COM/LG_RD_2030_COM_ACTIVATE.TXT
rconnect1.0/phones/LG_RD_7130/
rconnect1.0/phones/LG_RD_7130/LG_RD_7130_USB/
rconnect1.0/phones/LG_RD_7130/LG_RD_7130_USB/LG_RD_7130_USB
rconnect1.0/phones/LG_RD_7130/LG_RD_7130_USB/LG_RD_7130_USB_ACTIVATE.TXT
rconnect1.0/phones/LG_RD_7130/LG_RD_7130_COM/
rconnect1.0/phones/LG_RD_7130/LG_RD_7130_COM/LG_RD_7130_COM
rconnect1.0/phones/LG_RD_7130/LG_RD_7130_COM/LG_RD_7130_COM_ACTIVATE.TXT
rconnect1.0/common/
rconnect1.0/common/check
rconnect1.0/common/sheller.sh
rconnect1.0/common/rconnect
rconnect1.0/common/options
rconnect1.0/common/monitor
```

```
rconnect1.0/common/a.ppprc
rconnect1.0/common/rdisconnect
rconnect1.0/common/rupdate
rconnect1.0/common/runinstall
rconnect1.0/common/detectmodem
rconnect1.0/install.sh
rconnect1.0/LICENCE.TXT
rconnect1.0/README
rconnect1.0/comwizard
rconnect1.0/.README.swp
```

If your file had no corruption and things went well, this would work fine for you. Now you have all the requisites for installation and just need to go ahead with an installation.

## 2.3 Beginning the Installation

If you follow the instructions listed in Reliance's website, you'll probably get the following output.

```
username@hostname> sh install.sh
Welcome to RConnect Installation Wizard
---------------------------------------

[STEP : 1/5] Select your Reliance IndiaMobile

Only root user can install
Login as username-->root then try installation again
RConnect Installation terminated
```

**Installing "rconnect" scripts** Here's what you'll have to be doing, to become user 'root' and start the installation. (at least this is how I prefer doing it.)

```
username@hostname> su -
password:<type your root password here>
root@hostname> cd /home/username/path/to/rconnect1.0/
root@hostname> bash install.sh
```

Now things start looking better. The script installation has started. Please remember that this script installation will work only so long as your phone hardware is supported.

```
---------------------------------------
Welcome to RConnect Installation Wizard
---------------------------------------

[STEP : 1/5] Select your Reliance IndiaMobile

 1. GTRAN GCP 4020
```

```
 2. KYOCERA 7135
 3. LG LSP -340E
 4. LG LST 250
 5. LG RD 2030
 6. LG RD 2130
 7. LG RD 2230
 8. LG RD 7130
 9. SAMSUNG SCH A563
10. SAMSUNG SCH A603
11. SAMSUNG SCH A663
12. SAMSUNG SCH N191

To choose Your phone just type the corresponding serial number :
```

I chose to use '3', which is the model I am currently connecting to. You'll be selecting your appropriate model. The model selection is important to select the correct modem initialization string, so ensure that you have selected the right model.

```
Selected phone : LG LSP -340E
Have you selected the correct phone (Y/N) :
```

We can happily select 'Y' here, or say 'N', reselect the right phone, come back here and press 'Y' all over again. To get to the next step (assuming you selected the right phone), press 'Y'.

```
---------------------------------------
Welcome to RConnect installation Wizard
---------------------------------------

[STEP : 2/5] Activating R Connect

Follow the below instruction to activate R Connect

        1. Power on the phone.
        2. Press [MENU].
        3. Press 2 [Additional].
        4. Press 4 [SIO Baud Rate].
        5. Select 115,200 bps option.

Have you activated R Connect (Y/N) :
```

This is also probably the easier part, so you go ahead and say Y once you've done this. Trying higher Baud Rate settings is asking for trouble [at least, avoid doing that first time]

```
---------------------------------------
Welcome to RConnect installation Wizard
---------------------------------------

[STEP : 3/5] Setup your Reliance IndiaMobile as Modem
```

```
Connecting your phone to your computer

         1. Power on your phone
         2. Connect the phone end of data cable to the phone
         3. Connect the computer end of data cable to the computer

COM ports on your computer :
No com ports found on your computer

Select the port to which your phone is attached :
```

This is the most likely case if you are about to use USB to connect to the net. There's a lot more we need to get ready or rely on the USB wizard. For this model at least you need an alternate USB-Serial converter (not provided by reliance). On the phone we've selected however, this is what you get:

```
---------------------------------------
Welcome to RConnect installation Wizard
---------------------------------------

[STEP : 3/5] Setup your Reliance IndiaMobile as Modem

Connecting your phone to your computer

         1. Power on your phone
         2. Connect the phone end of data cable to the phone
         3. Connect the computer end of data cable to the computer

COM ports on your computer :
1. COM1
2. COM2

Select the port to which your phone is attached :
```

I chose 2, coz that's where my machine is connected to. Now the program tries an autodetect. If this is successful it prints out a `MODEM DETECTED`. In case it failed, it prints out a `MODEM DETECTION FAILED`.

The problem with this autodetect is that if some other program had opened the modem earlier (for whatever reason), this would fail. This has worked for me the first time I ran it, but not the second, and while I'm writing this document against time, I'm proposing a nasty hack.

### 2.3.1 Disabling the Modem Detection

**hacking 'detectmodem'** Not reccommended if things have worked and your modem got detected. (or you're rerunning things after doing this.)

```
root@localhost> cd /path/to/rconnect1.0/
root@localhost> cd common
root@localhost> mv detectmodem detectmodem.old
root@localhost> echo \#\!/bin/bash > detectmodem
root@localhost> echo exit 0 >> detectmodem
root@localhost> chmod a+x detectmodem
root@localhost> cd ..
```

Re run from the installation step (2.3). If things didn't work, you need to do this only once. You can obviously skip this part hoping you've selected the right model and port combination. The reason we are doing this is, `detectmodem`, instead of actually talking to the modem with the AT command sequence attempts to establish a ppp connection straight off. In short, this script needs to and will be rewritten.

### 2.3.2 Making the connection

**step 4** The next step is trying out the connection. If you have done the `detectmodem` hack, things may not always turn out right. But so long as you selected the right serial port and the rest of them, it should work great. You only need your phone number as Username and Password. All STD codes are to be used with no '0' prefixes.

```
----------------------------------------
Welcome to RConnect installation Wizard
----------------------------------------

[STEP : 4/5] Connecting to internet using R Connect

Making a Connection
        <Username>-Enter your phone number with the STD code prefixed to it
        [Example: 2231790000]
        <Password>-Enter your phone number with the STD code prefixed to it
        [Example: 2231790000]

Please enter username : 80123456
Please enter Password :
```

**step 5** Well, step 5 which never seemed to have happened here is actually asking permission to Connect. You would probably press 'Y' now to find out if the connection is established.

```
Connecting to Device.........
Validating username password
Successfully Connected
```

Your Reliance web phone's display should now show something about a PPP data connect and the data Receive and Transmit rates. You can try out your favorite browser (mine is Firefox) to connect and browse the net. With everything else being fine, this should work out of the box. In case you have doubts about your connectivity, just read the next section and the one on testing (2.5).

## 2.4 Connecting and Disconnecting

**connecting** This is probably what you would be doing more frequently once the installation is complete. You will need to be root and run /sbin/rconnect, every time you want to connect to the net. You can setup a shortcut on whatever desktop (KDE or GNOME and also set the suid bit to run the script as root from any user.)

```
username@hostname> su
password:<type your root password here>
root@hostname> /sbin/rconnect

Connecting to Device.........
Validating username password
Successfully Connected
```

**disconnecting** You will have to manually disconnect, or the connection would get disconnected only when you explicitly shutdown your machine. You probably want to disconnect when you would no longer be requiring the connection. This you can do as follows.

```
username@hostname> su
password:<type your root password here>
root@hostname> /sbin/rdisconnect
```

**avoiding root** You can avoid doing everything as root by using setuid, though this is not reccommended if security is a major concern for your desktop. For simplicity, this is how you'd setuid the connect and disconnect scripts.

```
username@hostname> su
password:<type your root password here>
root@hostname> chmod a+s /sbin/rconnect
root@hostname> chmod a+s /sbin/rdisconnect
root@hostname> chmod a+s /usr/sbin/pppd
```

## 2.5 Testing the connection

Now, that you have everything configured, you'll want to try out your connection. You can start with the following:

```
username@hostname> ping www.google.com -c 4
PING www.google.akadns.net (64.233.171.104) 56(84) bytes of data.
64 bytes from 64.233.171.104: icmp_seq=1 ttl=236 time=699 ms
64 bytes from 64.233.171.104: icmp_seq=2 ttl=236 time=689 ms
64 bytes from 64.233.171.104: icmp_seq=3 ttl=236 time=869 ms
64 bytes from 64.233.171.104: icmp_seq=4 ttl=236 time=779 ms

--- www.google.akadns.net ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3619ms
rtt min/avg/max/mdev = 689.724/759.547/869.734/72.604 ms
```

If things went well, you'd see output like what's shown above. This means your machine can connect to google. You can immediately open your favorite browser and try connecting to a few sites.

**trouble with the connection** If things didn't work out right, here's what you'd most likely be seeing:

```
username@hostname> ping www.google.com -c 4
ping: unknown host www.google.com
```

At this point, I'd want to you to run `/sbin/rdisconnect` once, rerun `/sbin/rconnect` make sure that your phone is showing up a PPP data transfer screen (with Receive, Transmit rates) and retry the `ping`. To verify whether the actual PPP connection is up, you can try to check as follows.

```
username@hostname> /sbin/ifconfig
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:10 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:700 (700.0 b)  TX bytes:700 (700.0 b)

ppp0      Link encap:Point-to-Point Protocol
          inet addr:220.226.45.121  P-t-P:97.239.2.5  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:14 errors:0 dropped:0 overruns:0 frame:0
          TX packets:15 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:911 (911.0 b)  TX bytes:372 (372.0 b)
```

If you aren't seeing `ppp0` here and `/sbin/rconnect` said "**successfully connected**", then you'll probably want to custom configure everything yourself and not use the script. Other simple commands you can run to verify if the ppp daemon is on is `pgrep pppd` which will show you the process number of pppd if it is running. If no number comes up, something fundamental is not working. Else this means there's more work to do.

# 3 Conclusion

**inconclusive conclusion ...** This document is far from over. I haven't dealt with all possible issues here, so this first draft should still help you get connected to the net with little trouble. I also provide a version of the rconnect-cmdline-1.0.tar with the `detectmodem` hacked because it seems to fail for me more than once and I can't wait for a proper rewrite.

**contacting me...** http://www.geocities.com/sunilbetabaskar/contact.html should tell you how to get to me.