

GBOX Installation HOWTO

For REEL BOX Receivers

mami_hacky
January 2th 2007

Introduction

OK! I'm a beginner, Interested in reelbox receiver, because of its power, and great hardware architecture and beautiful ideas used in its design.

The creators of reelbox do their best to provide good support to customers, just ask a question about reelbox in <http://www.reelbox-support.co.uk/index.php> or its german equivalent and be sure that you will receive a full reply in less than one day.

Reelbox provides you a complete linux based computer, which with, you can do every thing you want. Even you can download the firmware source codes and compile your own firmware. The main available firmwares for reelbox are based on VDR Linux. (<http://www.vdr-portal.de>, <http://vdr.blux.org/index.php> , <http://www.4freeboard.to/board/>)

Unfortunately reelbox support is very weak in some areas. These areas are usually related to illegal material such as SoftCams and Card sharing. The creators of reelbox are doing their best to prevent reelbox users from using their boxes illegally. So, when you go through famous reelbox forums and sites, you will never find a reelbox image with a card sharing client integrated into it. All of these kind of images have been removed from the net. One famous example is <http://www.reel-pirates.ws/wbb2/>. This site released two or three reelbox images with gbox client integrated into it in the past, but now, they no more provide such firmwares.

There are two famous card sharing protocols in todays world, 1- GBOX and 2- NewCamd. There are other protocols available but they are not used widely. (examples camd3, newcs and ...)

So, I'm here going to talk about how you should add gbox to your reelbox. What changes you need to do, and which files should be added.... I'm not going to describe how to build a new image from scratch. This is not needed at all. I'm going to just describe howto add gbox to an existing reelbox image.

Yes, I read the letter from the original creator of gbox (available in one of GBOX distros) carefully. And yes, I know that he do not want his program to be used in public. But GBOX is so general now, many people around the world are using it. Then why an owner of a reelbox receiver should not be able to use it? So, I begin with what you need...

Preparing for installation

1. REELBOX digital receiver
2. REELBOX image, based on VDR with VDR-SC support. VDR SC is a plugin for VDR Linux. SC is designed to implement video decryption routines in software. For obtaining such an image goto <http://www.reel-pirates.ws/wbb2/> or <http://reelbox.funfiles.cc/>. There you can download reel's images with VDR SC compiled into it.
3. GBOX software. GBOX software has different distributions. It is available for different kinds of CPUs. Original GBOX was developed for dreambox which was based on PowerPC CPU. Another versions of gbox were created for other CPUs then. We will use an x86 based version of GBOX. It's main executable is called gboxx86. This version of gbox is also used for DVB cards. I provide you two sources for obtaining gbox:

<http://www.skystar.org/arsiv/index.php?dir=cardsharing/gbox> and <http://rapidshare.com/files/9946010/gbox.tar.bz2.html> I have uploaded the second one myself.

4. A valid account on a gbox server. For this one you need to contact some one who has a set of cards and want to share these cards with others. You will give him your dns address, and you also open a UDP port on your router (ADSL router at home, using NAT). He will create a file for you which is called cwshare.cfg

For reelbox image, I use RBC Image 2713, with SC 0.5.12 compiled into it. So the first step is to download the image , to write into onto a CD and to update your reel's firmware.

GBOX Installation

Now, you have a RBC 2713 image running , please note that this firmware loads SC automatically. Some of the firmwares do not load SC automatically and you should set some special settings in the setup section of the reel. (The section which loads when you press the setup key on your remote)

Now, download the latest gbox or download my provided link. Telnet into your reelbox, please note that you should have configured the network connections for your reel properly. For me, I have set the IP address for reel to 10.10.10.175, the gateway is 10.10.10.1 and name server is also 10.10.10.1, this is the address of my ADSL router. Reel is connected to a wireless switch in my case and I can connect to it using my notebook. Now,

```
telnet 10.10.10.175
cd /mnt/hd
mkdir keys
```

As you see I telnet into the box, then I go to the location where reel's hard drive is located, this is /mnt/hd. Please not that your reelbox should have a hard drive. If you have purchased reelbox both no hard drive this solution does not work for you. I place all of the gbox related files on the hard drive. The reason is simply, there is no enough space on the Disk-On-Chip memory of reel. (2713 image, but older images have more free space and you can put gbox on Disk-On-Chip) now copy all of the files in gbox.tar.bz2 file into the keys folder on reel's hard drive. Here is a list of the files, I put the list here so that you be able to use other gboxx86 versions than the one I'm providing here:

drwxr-xr-x	2	root	root	16	Apr	17	02:16	cs2gbox
-rwxrw-rw-	1	root	root	175	Apr	17	02:05	cwshare.cfg
-rw-r--r--	1	root	root	134	Apr	17	02:36	ecm.info
-rwxrw-rw-	1	root	root	6.4k	Apr	17	02:05	gbox_cfg
-rwxrw-rw-	1	root	root	1.1M	Apr	17	02:05	gboxx86
-rw-r--r--	1	root	root	18.3k	Apr	17	02:16	ident.info
-rw-r--r--	1	root	root	10.9k	Apr	17	02:16	ignore.list
-rw-r--r--	1	root	root	23.7k	Apr	17	02:16	knowns.ini
-rw-r--r--	1	root	root	16.0k	Apr	17	02:16	rom02.b
-rw-r--r--	1	root	root	4.0k	Apr	17	02:16	rom02eep.b
-rw-r--r--	1	root	root	544	Apr	17	02:16	rom02ram.b
-rw-r--r--	1	root	root	16.0k	Apr	17	02:16	rom03.b

-rw-r--r--	1	root	root	4.0k	Apr 17 02:16	rom03EEP.b
-rw-r--r--	1	root	root	544	Apr 17 02:16	rom03RAM.b
-rw-r--r--	1	root	root	16.0k	Apr 17 02:16	rom07.b
-rw-r--r--	1	root	root	4.0k	Apr 17 02:16	rom07EEP.b
-rw-r--r--	1	root	root	544	Apr 17 02:16	rom07RAM.b
-rw-r--r--	1	root	root	22.5k	Apr 17 02:16	rom10.b
-rw-r--r--	1	root	root	8.0k	Apr 17 02:16	rom10EEP.b
-rw-r--r--	1	root	root	1.0k	Apr 17 02:16	rom10RAM.b
-rw-r--r--	1	root	root	22.5k	Apr 17 02:16	rom11.b
-rw-r--r--	1	root	root	8.0k	Apr 17 02:16	rom11EEP.b
-rw-r--r--	1	root	root	1.0k	Apr 17 02:16	rom11RAM.b
-rw-r--r--	1	root	root	202	Apr 17 02:16	s2g_cfg
-rwxrwxrwx	1	root	root	54.3k	Apr 17 02:16	s2ISSUER.b
-rw-r--r--	1	root	root	4.5k	Apr 17 02:16	s2PROVID.b
-rw-r--r--	1	root	root	3.9k	Apr 17 02:16	softcam.cfg

Now , the main files in the above list are: gboxx86 which is the main executable, gbox_cfg, this is the configuration file for gbox and cwshare.cfg which as I said will be provided by the one who is sharing his card with you. (I don't want to talk about cwshare.cfg here) other files are not really important and even you can omit them :D Please look at the permissions for these files, your permissions should be the same as the above. If not, go to the directory and enter the following command:

```
chmod 755 *
```

Please do not forget to put a root password for your box exactly after you finished with the installation of each new image. As you know the root user is very easily accessible at the beginning because there is no password for it. You can use passwd command for this purpose. OK, now we should make a symbolic link:

```
cd /var
ln -s /mnt/hd/keys keys
```

this makes a symbolic link /var/keys to point to the main location of gbox files, /mnt/hd/keys. /var changes each time you boot the system from beginning. So, I add the following line in the bottom of the /etc/profile file, this way , the symbolic link will come up automatically the system comes up.

```
ln -s /mnt/hd/keys keys
```

In addition if you want gbox to be executed each time your box boots, you can add the following line at the end of /etc/profile file:

```
/var/keys/gboxx86 &
```

now let's do another important step. VDR SC has a config file called cardclient.conf. (/etc/video/plugins/cardclient.conf) This file indicates for SC Plugin to where to search for new keys. Using this file we should tell it to connect to GBOX and get the keys. This is the modified cardclient.conf file:

```

#
# Comment lines can start with # or ;
#
# every client line starts with the client name, followed by some arguments:
# 'hostname' is the name of the server
# 'port' is the port on the server
# 'emm' is a flag to allow EMM transfers to the server
# (0=disabled 1=enabled)
# 'caid' (optional) caid on which this client should work
# 'mask' (optional) mask for caid e.g. caid=1700 mask=FF00 would allow
# anything between 1700 & 17FF.
# Default is 1700 & FF00. If only caid is given mask is FFFF.
# You may give multiple caid/mask values comma separated
# (e.g. 1702,1722,0d0c/ff00).
# 'username' is the login username
# 'password' is the login password
#
# radegast client
#radegast:hostname:port:emm/caid/mask
#
# aroureos client
# 'hexbase'
# 'hexserial' card data for which EMM updates should be send
#aroureos:hostname:port:emm/caid/mask:hexbase:hexserial
#
# camd33 client (tcp protocol)
# 'aeskey' is the AES key (32bytes), disable encryption if missing
#camd33:hostname:port:emm/caid/mask:username:password:aeskey
#
# camd35 client (udp protocol)
#camd35:hostname:port:emm/caid/mask:username:password
#
# cardd client
#cardd:hostname:port:emm/caid/mask:username:password
#
# buffy client
# 'aeskey' is the AES key (32bytes), disable encryption if missing
#buffy:hostname:port:emm:username:password:aeskey
#
# newcamd client
# 'cfgkey' is the config key (28bytes)
#newcamd:hostname:port:emm/caid/mask:username:password:cfgKey
#
# gbox client
#
# NOTE: hostname & port will be ignore. GBOX must be runnning on the local
# machine. For convinience you should choose localhost:8004
#gbox:hostname:port:emm/caid/mask
gbox:localhost:8004:1/0000/0000

```

As you see I have un-commented the gbox line.

Test The Installation

Now, we are going to test if every thing is OK. First of all. Let's try the gbox. As I said you should get a cwshare.cfg file from the one who is sharing you the keys. This file should also be copoied to /var/keys (/mnt/hd/keys). now telnet into the box and start gbox by running /var/keys/gboxx86. A list of messages like these should be generated:

```

gbox 2.25/Linux@X86 ( May 21 2006, 18:56:23 )
Seasonport=0 speed=9600
OSD (dbox2) IP = 10.10.10.175

```

```

RS232gate=0 speed=115200
mode 03
AU:02/update:01/KeyFile:00 Hash:01 DispECM:02/EMM:02 UDPIInit:00 OSD:00
could not open ./irdeto
could not open ./seca
could not open ./via
could not open ./nagra
could not open ./conax
could not open ./crypto
trying to get online, please wait ...
My IP XXX.XXX.XXX.XXX ID CCCC (NNNNNNNNN.dyndns.org) pass YYYYYYY, we are online
IP   YYY.YYY.YYY.YYY Port   ZZZZZZ-ZZZZZZ ID   4444 pass   VVVVVVV sharelevel 3/3
(VVVVVVV.dyndns.tv)
enx_conf = 3 ; reset on zap=0
Total 1 ports to monitor
monitoring port: 17543
CPU 80X86 compatible
using LinuxTV api3
TPSCrypt 1 PTS 1/1
Could not open /dev/sci0: Success
Could not open /dev/sci1: Success
mode: any pid
entering main loop
ReRead pmt.tmp
->HelloL to VVVVVVV.dyndns.tv (YYY.YYY.YYY.YYY:ZZZZZ)
ReRead pmt.tmp
<-Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ
**** hashTableECM 0 elements ****
**** hashTableEMM 0 elements ****
ReRead pmt.tmp
->HelloL to VVVVVVV.dyndns.tv (YYY.YYY.YYY.YYY:ZZZZZ)
**** hashTableECM 0 elements ****
**** hashTableEMM 0 elements ****
<-Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ
<=Hello from VVVVVVV.dyndns.tv port ZZZZZ

```

Now, you should make sure that SC is working properly and it can see GBOX, so, you need a complete log of its operation. For this reason, we will kill all of the VDR processes and we will start them again.

Log into your reelbox using a separate telnet login. Do not use your previous login and let GBOX to be alive. So make a new telnet connection and :

```

killall -9 vdr
killall -9 reelvdr

```

Now start the VDR process:

```
/etc/init.d/reelvdr -bsp-started
```

Now look at the generated lines, you should see a line with these info it:

```

INFO: loading cardclient config from
/etc/video/plugins/cardclient.conf

```

```
cc: hostname=localhost port=8004 emm=1 emmCaids 0000/0000
netwatch: up
socket: binding to 127.0.0.1:8003/udp (127.0.0.1)
cc-link: client 'gbox' ready
cc-link: created 1 client(s)
```

As you see, SC sees gbox and can get connected to it. Now, it is the time to have some real tests.

As you know your reelbox has at least two tunners, some of them have four. Mine has two tunners and by experience I know that my second tuner is in a better operational condition than the first one so, I do my tests with the second tuner. For this purpose I put a task to the first tuner and make it busy and then I send another task to reel which will be done by the second tuner. I ask the first tuner to show the "BBC World" Channel. In my notebook I type:

```
telnet 10.10.10.175 2001
```

As you know, VDR Linux provides a telnet like interface on port 2001. using this telnet interface you can control the operation of VDR remotely. Now, in the telnet command prompt I type:

```
CHAN BBC World
```

I used the CHAN command to change the program of the first tuner of reel. Now, using VLC I ask reel to play "TPS Star" channel for me, first I connect to <http://10.10.10.175:3000> which gives me a complete channel listing, I obtain the link for "TPS Star" and then using VLC I try to watch the channel, let's look at the SC and GBOX logs then. This is a portion of what SC shows:

```
cc-gbox: no CA descriptor for caid 0500 sid 1204 prov 20010
cc: client gbox (localhost:8004) ECM failed (0 ms)
ecmhandler 1.0: try system Viaccess (0500) id 20010 with ecm 1d7 (cached) (pri=-10)
ecmhandler 1.0: stopping console log until valid key is found
action: read logger 1/1: Buffer overflow
action: read logger 1/1: Buffer overflow
action: read logger 1/1: Buffer overflow
cc: cc-loop
cc: now trying client gbox (localhost:8004)
cc-gbox: no CA descriptor for caid 0500 sid 1204 prov 20010
cc: client gbox (localhost:8004) ECM failed (1 ms)
cc: cc-loop
cc: now trying client gbox (localhost:8004)
ecmhandler 1.0: correct key found
REMUX SYNCED
Enabled packer optimization
```

This log shows that after some tries SC could finally found the correct key to decrypt the video. Now let's have a look at GBOX's log:

```
Restarting pid 05FF c:501
Restarting pid 05FF c:501
Restarting pid 05FF c:501
Restarting pid 05FF c:501
ReRead pmt.tmp
*ECM: CA ID: 0x0500 Viaccess (008300) -> CA PID: 0x01D5
*ECM: CA ID: 0x0500 Viaccess (008100) -> CA PID: 0x01D0
*ECM: CA ID: 0x0500 Viaccess (007C00) -> CA PID: 0x01CC
```

```

*ECM: CA ID: 0x0500 Viaccess      (022600)      -> CA PID: 0x01CE
*ECM: CA ID: 0x0500 Viaccess      (020000)      -> CA PID: 0x01D4
*ECM: CA ID: 0x0500 Viaccess      (022610)      -> CA PID: 0x01CD
*ECM: CA ID: 0x0500 Viaccess      (020010)      -> CA PID: 0x01D7
**** hashTableECM 7 elements ****
**** hashTableEMM 0 elements ****
->ECM send to VVVVVV.dyndns.tv (ZZZZ ZZZZ ZZZZ ZZZZ ZZZZ ZZZZ )
<-CWb (<-3) received from VVVVVV.dyndns.tv (5566) (414 ms)
ECM: CA ID: 0x0500 Viaccess      (008300)      -> CA PID: 0x01D5
ECM: CA ID: 0x0500 Viaccess      (008100)      -> CA PID: 0x01D0
ECM: CA ID: 0x0500 Viaccess      (007C00)      -> CA PID: 0x01CC
ECM: CA ID: 0x0500 Viaccess      (022600)      -> CA PID: 0x01CE
ECM: CA ID: 0x0500 Viaccess      (020000)      -> CA PID: 0x01D4
ECM: CA ID: 0x0500 Viaccess      (022610)      -> CA PID: 0x01CD
*ECM: CA ID: 0x0500 Viaccess      (020010)      -> CA PID: 0x01D7
**** hashTableECM 1 elements ****
**** hashTableEMM 0 elements ****
==== Network ECM on CaID 0x0500, pid 0x01D7 ===
prov: 5566
CW0:  BB 8E 8C D5 37 56 78 05
CW1:  94 01 28 BD 77 5C 67 3A
=====

```

So, you see that GBOX has got a CW from server and has passed it to SC to decrypt the video.

Notes

Delay of your network is very important in the performance of this method. If the rate of key change is high (e.g. Irdeto channels such as NOVA), then the delay from the server to you home should be as small as possible to make a good card sharing possible. SC generates a time out message when ever it could not get the correct key in correct time.

GBOX can be directly integrated into VDR without the need for SC. The experts say that this makes the delay slower, they say that SC acts very slowly in some of the conditions and this is a problem. GBOX can be directly connected to VDR in premium cards and I don't know if such a thing is possible for reel at all.