

Tvheadend - Bug #5128

When tuner is configured as a slave device, after reboot settings are lost

2018-06-10 14:47 - Trex the Daemon

Status:	Fixed	Start date:	2018-06-10
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Configuration	Estimated time:	0.00 hour
Target version:		Affected Versions:	
Found in version:	4.3		
Description			
I have 3 tv tuners: - Dvbsky S960 DVB-S2, - Sundtek Skytv ultimate v6 DVB-S2 - Dvbsky T330 DVB-C+DVB-T			
I have usals and multiple sattelites configured. Now the DVBSKY and Sundtek are connected through a splitter to the same LNB, therefore I wanted to set the DVBSKY as a slave. After configuring everything, after reboot the settings are lost and the slave tuner reverts back to "Universal only" and no network configured. I've attached the whole configuration, only the passwords are deleted for security reasons.			

Associated revisions

Revision 937a5fb7 - 2019-03-20 13:45 - Jaroslav Kysela

linuxdVB: create the mux instances also for the slave tuners, issue #5128

Revision 453ee8df - 2019-03-20 13:47 - Jaroslav Kysela

linuxdVB: use the right configuration root for the slave tuners (loading), issue #5128

History

#1 - 2018-06-10 21:38 - Jaroslav Kysela

You may try to swap master/slave devices. Apparently, if the order is wrong (slave is first at the load time), the config is not restored properly, because master does not exist.

#2 - 2018-06-11 09:28 - Trex the Daemon

Jaroslav Kysela wrote:

You may try to swap master/slave devices. Apparently, if the order is wrong (slave is first at the load time), the config is not restored properly, because master does not exist.

Thanks. I will have to create some udev rules to specify the adapter names statically, so that the master device will always get the /dev/adapter0 entry.

#3 - 2018-06-12 14:28 - Trex the Daemon

Trex the Daemon wrote:

Jaroslav Kysela wrote:

You may try to swap master/slave devices. Apparently, if the order is wrong (slave is first at the load time), the config is not restored properly, because master does not exist.

Thanks. I will have to create some udev rules to specify the adapter names statically, so that the master device will always get the /dev/adapter0 entry.

Well, I've just checked the startup log and what we've assumed here seems to not be the reason:

Jun 12 15:22:45 raspberrypi tvheadend[21472]: linuxdVB: adapter added /dev/dvb/adapter2

Jun 12 15:22:45 raspberrypi tvheadend[21472]: linuxdvb: adapter added /dev/dvb/adapter1
Jun 12 15:22:45 raspberrypi tvheadend[21472]: linuxdvb: adapter added /dev/dvb/adapter0

So adapter 2 is the master adapter, and adapter 1 is the slave. If we take the order of the addition of the adapters, the master does exist before the slave is added.

Could you tell me where is this part of the code implemented ? Maybe I can have a look to see if I can find something there..

#4 - 2018-06-13 08:48 - Jaroslav Kysela

It seems that the order should not matter. The master tuner should be identified just using the uuid string. Check linuxdvb_frontend_create() linuxdvb_frontend_save() in src/input/mpegts/linuxdvb/linuxdvb_frontend.c . Look for 'master for #' strings.

#5 - 2018-06-13 08:50 - Trex the Daemon

Jaroslav Kysela wrote:

It seems that the order should not matter. The master tuner should be identified just using the uuid string. Check linuxdvb_frontend_create() linuxdvb_frontend_save() in src/input/mpegts/linuxdvb/linuxdvb_frontend.c . Look for 'master for #' strings.

I've just checked in the config file the UUID and they are correct.
I'll have a look in the code...

#6 - 2018-06-13 09:45 - Jaroslav Kysela

There must be something wrong - linuxdvb_frontend_dvbs_slave_class is not probably used for the slave.

#7 - 2018-07-02 19:52 - Trex the Daemon

Jaroslav Kysela wrote:

There must be something wrong - linuxdvb_frontend_dvbs_slave_class is not probably used for the slave.

I've checked the code but I'm not quite getting what here is meant:

```
/* Tuner slave */  
snprintf(buf, sizeof(buf), "master for #%d", number);  
if (conf && type == DVB_TYPE_S) {  
    muuid = htmsg_get_str(conf, buf);  
    if (muuid && uuid && !strcmp(muuid, uuid))  
        muuid = NULL;  
}
```

It gets the uuid of the master I suppose, and then checks to make sure that the master and the slave uuids are different ? What's the reason for that?

Later on, depending if we have a master uuid, it will use the correct pointer:
if (type == DVB_TYPE_S)
 idc = muuid ? &linuxdvb_frontend_dvbs_slave_class : &linuxdvb_frontend_dvbs_class;

so this should be fine..

#8 - 2018-11-24 15:53 - Trex the Daemon

Trex the Daemon wrote:

Jaroslav Kysela wrote:

There must be something wrong - linuxdvb_frontend_dvbs_slave_class is not probably used for the slave.

I've checked the code but I'm not quite getting what here is meant:

```
/* Tuner slave */  
snprintf(buf, sizeof(buf), "master for #%d", number);  
if (conf && type == DVB_TYPE_S) {  
    muuid = htmsg_get_str(conf, buf);  
    if (muuid && uuid && !strcmp(muuid, uuid))  
        muuid = NULL;  
}
```

It gets the uuid of the master I suppose, and then checks to make sure that the master and the slave uuids are different ? What's the reason for that?

Later on, depending if we have a master uuid, it will use the correct pointer:

```
if (type == DVB_TYPE_S)
idc = muuid ? &linuxdvb_frontend_dvbs_slave_class : &linuxdvb_frontend_dvbs_class;
```

so this should be fine..

Well, this issue is still present, now on a different device..

#9 - 2018-11-24 18:53 - Jaroslav Kysela

Have you had verified (use printf or gdb) if linuxdvb_frontend_dvbs_slave_class is really used for the slave frontend?

#10 - 2019-03-06 19:36 - Trex the Daemon

- File 65cce2b6dc875ed17dffe0c5ee91c5c5 added
- File 44128f8e1ad13b8417dac31e4e12521e added
- File Capture.PNG added

Jaroslav Kysela wrote:

Have you had verified (use printf or gdb) if linuxdvb_frontend_dvbs_slave_class is really used for the slave frontend?

Sorry, did not have time to debug this.

Meanwhile I've changed the dvbsky to another sundtek tuner, and the same issue happens also here, after reboot, the tuner that was configured as slave loses its settings and will become master again with "Universal LNB".

Interesting is, that in the config file it looks ok.

44128f8e1ad13b8417dac31e4e12521e is the master, and 65cce2b6dc875ed17dffe0c5ee91c5c5 is set as slave (see the logs).

Tvheadend version is 4.3-1774, but it's already present in previous 4.3 version.

However, in stable 4.2 it's not!

#11 - 2019-03-07 08:27 - Trex the Daemon

Trex the Daemon wrote:

Jaroslav Kysela wrote:

Have you had verified (use printf or gdb) if linuxdvb_frontend_dvbs_slave_class is really used for the slave frontend?

Sorry, did not have time to debug this.

Meanwhile I've changed the dvbsky to another sundtek tuner, and the same issue happens also here, after reboot, the tuner that was configured as slave loses its settings and will become master again with "Universal LNB".

Interesting is, that in the config file it looks ok.

44128f8e1ad13b8417dac31e4e12521e is the master, and 65cce2b6dc875ed17dffe0c5ee91c5c5 is set as slave (see the logs).

Tvheadend version is 4.3-1774, but it's already present in previous 4.3 version.

However, in stable 4.2 it's not!

Also, I've checked that if in the list, the slave tuner is the second, it will also show as master, so the position during initialization does not matter.

#12 - 2019-03-11 21:43 - Deep Thought

The bug is caused as follows: In linuxdvb_frontend_create, on entry, conf points to "frontends", but its value is overwritten

by the following code:

```
/* Internal config ID */
snprintf(id, sizeof(id), "%s #%d", dvb_type2str(type), number);
if (conf)
    conf = htmsg_get_map(conf, id);
```

The key,value pair for "master for #0" is part of the "frontends" map but it is now looked up in "DVB-S #0" instead, and is not found.

The solution is to save the initial value of conf, and then to use it for looking up "master for #0". E.g.:

```
linuxdvb_frontend_t *
linuxdvb_frontend_create
```

```

( htmsg_t *conf, linuxdvb_adapter_t *la, int number,
  const char *fe_path, const char *dmx_path, const char *dvr_path,
  dvb_fe_type_t type, const char *name )
{
  const idclass_t *idc;
  const char *str, *uuid = NULL, *muuid = NULL;
  char id[16], lname[256], buf[256];
  linuxdvb_frontend_t *lfe;
  htmsg_t *scconf;
  ssize_t r;
  int fd;
  htmsg_t * oldconf =conf;

  /* Internal config ID */
  snprintf(id, sizeof(id), "%s %d", dvb_type2str(type), number);
  if (conf)
    conf = htmsg_get_map(conf, id);
  if (conf)
    uuid = htmsg_get_str(conf, "uuid");

  /* Tuner slave */
  snprintf(buf, sizeof(buf), "master for %d", number);
  if (conf && type == DVB_TYPE_S) {
    muuid = htmsg_get_str(oldconf, buf);
    if (muuid && uuid && !strcmp(muuid, uuid))
      muuid = NULL;
  }
  ...
}

```

#13 - 2019-03-11 22:05 - Deep Thought

However, the above code change is not sufficient.

A slave tuner should be able to tune to the same network as its master tuner. In reality it refuses to tune to anything, unless a list of networks is filled in the adapter config file. However, there is no way of setting this list of networks from the web ui.

I also noticed that even after setting the field "master tuner", the web ui still shows information for switches, which it should not do.

How is this supposed to work?

I assume that the user should NOT set networks for the slave tuners, but then the code should use the networks from the master and it does not. Perhaps still another bug?

#14 - 2019-03-12 09:14 - Trex the Daemon

Deep Thought wrote:

However, the above code change is not sufficient.

A slave tuner should be able to tune to the same network as its master tuner. In reality it refuses to tune to anything, unless a list of networks is filled in the adapter config file. However, there is no way of setting this list of networks from the web ui.

I also noticed that even after setting the field "master tuner", the web ui still shows information for switches, which it should not do.

How is this supposed to work?

I assume that the user should NOT set networks for the slave tuners, but then the code should use the networks from the master and it does not. Perhaps still another bug?

Hello,

Well, all I can say is that on 4.2, once you've set the tuner to "slave" and restarted tvheadend, from the gui all other settings, as network mapping disappeared. I've tested by occupying one of the tuners and then requested a second channel, from the same polarity and orbital position. Tvheadend then used the slave tuner to tune and serve the second request. So for some reason, this has been broken in the 4.3 versions.

#15 - 2019-03-12 20:13 - Deep Thought

Yes, my comment was about 4.3. With the above patch, the slave tuner status is permanent (which

is good). The settings which should be hidden eventually are hidden, but not immediately.

I did discover that if you add the networks of the master in the config file of the slave, the slave tuners can be used. However, this does not sound like the correct solution, as the slave tuner should simply use the same network as the master tuner.

#16 - 2019-03-20 14:06 - Jaroslav Kysela

Retest with v4.3-1787-g453ee8dfd .

#17 - 2019-03-20 21:35 - Deep Thought

I have tested and can confirm that this problem is now solved.

smile.png
Thanks

#18 - 2019-03-23 18:31 - Deep Thought

There still is one problem: when initially configuring a slave tuner, you need to check some boxes for each of the wanted networks. However, after a restart of tvheadend, the screen to change these choices is missing.

#19 - 2019-03-24 19:46 - Jaroslav Kysela

- Status changed from New to Fixed

#20 - 2020-01-09 11:53 - Victor S

Same problem Configured as slave - LNB types/diseq menus are still present until reboot. Now configured back to "This tuner" master, and diseq menus are not coming back, reboot didn't help, stuck as blank. So annoying.

#21 - 2020-01-12 09:22 - Victor S

Manual fix of how to recover after Slave-to-Master setting:
Open with text editor /home/hts/.hts/dd570f6f5b314dbed3920c96e5b858fa (see your ID in the folder of the adapter in question)
find "master for #0": "" and remove the line, save and restart tvheadend.

FIX IT FOR GOD'S SAKE, ticket is 1 y old!!!

#22 - 2020-01-12 10:15 - Flole Systems

Victor S wrote:

FIX IT FOR GOD'S SAKE, ticket is 1 y old!!!

It is fully open source software which is provided for free. You should stop demanding something from others and start fixing it on your own!

#23 - 2020-05-02 04:50 - Victor S

- File Screen Shot 2020-05-01 at 4.49.29 PM.png added

Flole Systems wrote:

Victor S wrote:

FIX IT FOR GOD'S SAKE, ticket is 1 y old!!!

It is fully open source software which is provided for free. You should stop demanding something from others and start fixing it on your own!

Are you fucking serious? Turn one tuner as slave and then turn it back as master. What the hell you see happened to it? Where is damn switch configuration dropdown menu has gone? Fix the obvious shit. Shame on you.

#24 - 2020-05-02 04:58 - Victor S

If you make an opensource, then make it useable and don't ignore stuff years old reported, this type of ignorance just pisses off. And you waiting credits for that?

#25 - 2020-05-02 11:39 - Mark Clarkstone

Victor S wrote:

If you make an opensource, then make it useable and don't ignore stuff years old reported, this type of ignorance just pisses off. And you waiting credits for that?

It's not that the 'bug' is 'ignored' as you put it, but rather there aren't enough developers that have *free* time to work on tvheadend right now. You "swearing and screaming bloody murder" helps no one, in fact it puts people off, users and developers alike, so thanks for that.

As for your 'bug', it's a very simple one and I myself suffer with it. I did some testing and found that the sub-devices of the pci card actually change or swap dvb device (/dev/dvb) positions on each reboot. This is down to the kernel and/or driver not using set positions for each sub-device. Tvheadend could handle this better but how is it supposed to detect which device is supposed to be master or slave when they keep doing musical chairs?

#26 - 2020-05-02 18:44 - Flole Systems

Victor S wrote:

Shame on you.

It's more shame on you. You haven't contributed a single line of code and demand that something is fixed by others. I am not sure what gave you the impression that you can tell others what they should do but that impression is completely wrong.

As Mark mentioned correctly: By complaining in a rude manner you will only make sure that this issue is not fixed. Nobody wants to help someone who says "shame on you". Obviously you also didn't read the license which clearly states that there is absolutely no warranty, not even implicit warranty that this software is useful. And that is exactly what you got, so I am not sure what exactly gave you the impression that this software should work for your purpose, the license explicitly states that it may or may not work. If you want something that you can complain about and demand to be working you need to find a company that focuses on software development and have them write a software for you that works for your purpose.

Files

backup_hts_10-06-2018.zip	8.21 MB	2018-06-10	Trex the Daemon
65cce2b6dc875ed17dffe0c5ee91c5c5	722 Bytes	2019-03-06	Trex the Daemon
44128f8e1ad13b8417dac31e4e12521e	1.12 KB	2019-03-06	Trex the Daemon
Capture.PNG	51.7 KB	2019-03-06	Trex the Daemon
Screen Shot 2020-05-01 at 4.49.29 PM.png	42.8 KB	2020-05-02	Victor S