Tvheadend - Bug #5649

CA PMT descriptor parsing changed in recent oscam

2019-06-08 23:20 - nautilus7 nautilus7

Status:	Fixed	**Start date:**	2019-06-08
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:			
Found in version:	latest version - current source code	**Affected Versions:**	

Description

In the ca pmt creation code, the 0x82 descriptor which informs the ca application (softcam) about the used demux device (and adapter) is misused and filled with different information depending on its position in the ca pmt message. This a totally wrong approach and should be avoided.

See
https://github.com/tvheadend/tvheadend/blob/76dd042e0d3bb93e1102eae65c2d23aa31233274/src/descrambler/capmt.c#L2247
https://github.com/tvheadend/tvheadend/blob/76dd042e0d3bb93e1102eae65c2d23aa31233274/src/descrambler/capmt.c#L2267

The purpose of each descriptor is to carry SPECIFIC data, independently of its position in the message.
The exact structure of a descriptor is determined by its tag.
Imagine what would happen if a DVB SI descriptor carried different information depending on its position in a TS section.

The misuse of this descriptor was hidden in softcams like oscam until recently, when a rewrite of the oscam's ca pmt descriptor parsing code exposed this issue.

If there is a need in tvheadend to send different information each time, then you should use different descriptors.
If none of the already supported ones is appropriate, you should create a new one. Currently oscam supports the following private descriptors in the ca pmt:

1. enigma namespace - carries enigma namespace, tsid and onid - descriptor_tag 0x81 descriptor_length 0x08

2. demux and ca_mask - carries ca device (in bit mask form) and demux device - descriptor_tag 0x82 descriptor_length 0x02 (deprecated)

3. adapter - carries adapter index - descriptor_tag 0x83 descriptor_length 0x01

4. pmt pid - carries pmt pid of the service - descriptor_tag 0x84 descriptor_length 0x02

5. service type mask (live tv, recording, etc) - descriptor_tag 0x85 descriptor_length 0x04

6. demux (added recently) - carries demux device - descriptor_tag 0x86 descriptor_length 0x01

Most of them (if not all) come from enigma and are already supported by tvheadend.
The main problem is the 0x82 descriptor as already mentioned, because it is used with two different ways in tvheadend...

I propose the use of the new 0x86 for sending the demux device in use and drop the deprecated 0x82 completely.
The 0x83 can be also used for sending the adapter info.

AFAIK, dvb adapters for pc do not have any hardware ca devices, so I am not sure how this should be handled.
I guess the virtual ca device should be the same as the demux device, e.g. for demux0 -> ca0, demux1 -> ca1, etc.
In this case, the 0x86 descriptor is enough.
Oscam assumes the ca device index is equal to the demux device when it receives the 0x86 descriptor. (This is also the case for modern enigma boxes)

Associated revisions

Revision d4860273 - 2019-06-12 14:47 - Jaroslav Kysela

capmt: fix for the oscam r11520+, fixes #5649

- allow to force the PMT composing, too

Revision bc769bfa - 2019-06-12 15:06 - Jaroslav Kysela

capmt: fix for the oscam r11520+, fixes #5649

- allow to force the PMT composing, too

History

#1 - 2019-06-09 08:16 - Jaroslav Kysela

The code is a bit hacky, but it supports wide range of oscam versions (including the emu ones). Previous oscam versions have tricky conditions assuming the fixed SI tag order. What's broken with the recent oscam exactly? I don't see any different information for 0x82 tag. The first data byte overflows for adapter numbers >= 8, so tvh sends zero there.

#2 - 2019-06-09 09:52 - nautilus7 nautilus7

Hi, thanks for the quick response.

What is the wide range of oscam versions?
Oscan svn and oscam-emu always shared (and still share) exactly the same code regarding parsing the ca pmt descriptors.
As you mentioned, the previous code was based in a specific tag order. Now this has changed and it can work with any order.

How is your two 0x82 descriptors the same???
In the first case in line 2249, the 1st byte is "0" and the 2nd is the "adapter".
In the second case in line 2269 the 1st byte is "1 << adapter" and the 2nd is "adapter or 0 based on wrapper", which I don't know what it is. And this is sent in a different position than the other one. So, previously oscam checked the position of the 0x82 descriptor in the capmt and parsed it accordingly. No this cannot be done now (and shouldn't be done anyway).

Problem is this:
A user with RPi has tvheadend 4.3-1792~g466a01431 installed. 0x82 descriptor is "82 02 01 00".
In previous oscam versions, this descriptor was parsed as demux0 and adapter0, but now this is parsed as demux1 and adapter0.
See: http://www.streamboard.tv/wbb2/thread.php?threadid=46306

The important question here is **what information is tvheadend supposed to send inside the 0x82 descriptor**? As soon as as this is cleared I will fix oscam.
But this is not clear to me, because there are 2 different 0x82 descriptor with different info that can be sent by tvh, in different scenarios...

#3 - 2019-06-09 17:22 - Jaroslav Kysela

- File a.patch added

The explanation is really simple.

The pmt_pid parsing for the 0x82 tag was added in revision 11396 for the PC (network) type of connection in oscam. So the only way to set the pmt_pid before this revision was to use the default code path in the getDemuxOptions() function where the first data byte was used as ca_mask. Apparenly, oscam removed this default path (no backward compatibility) which makes tvh code incompatible now. So, the different meaning of the first byte (ca_mask or demux_index) was in the original getDemuxOptions() function.

I can probably add some auto-detection of revision 11396+ from the DVBAPI_SERVER_INFO and don't send ca_mask for this newer oscam version. Patch attached. The question, if oscam should behave consistently for new versions, is not answered with this.

More references: https://tvheadend.org/issues/4793

#4 - 2019-06-09 21:31 - nautilus7 nautilus7

oscam always supported the 0x84 descriptor for sending the pmt pid. This is how enigma2 does it since ever...
Anyway.. Let's focus to the present...

The CA PMT message structure is described in p.30 of this document: https://www.dvb.org/resources/public/standards/En50221.V1.pdf
The private descriptors (0x81, 0x82, 0x83, 0x84, 0x85 and 0x86) were created by enigma2 and are also included among with the ca descriptors in the program level of the CA PMT. This is what enigma2 does for over a decade now...

A month ago I rewrote the CA PMT parsing code in oscam, so I doesn't care for the order of these descriptors inside the CA PMT. This means that you (tvh) can send whatever descriptor is needed, in whatever order. This is how it is supposed to be done for all descriptors found in the PMT, CAT, PAT, ECM, EMM and other TS sections. This is how it should have been done in oscam from the beginning. Better late than never, I would say...

So, why having useless code for detecting oscam version and making your code look bad and difficult to maintain???

Just send what ever information is needed using the appropriate descriptors.
Just stick to whatever information each descriptor is supposed to carry and send each one only a single time. So:

1. Use the 0x83 descriptor to send the adapter device. It should look like this: 83 01 AA, where 01 is the length and AA the adapter id (e.g. 00, 01, 02, ..., FF).

2. Use the new 0x86 descriptor to send the demux device. It should look like this: 86 01 DD, where 01 is the length and DD the demux device (normally equal to 00).

3. let's create a new descriptor with tag 0x87 and use it for the ca device. (syntax 87 01 CA). Alternatively, instead of creating a new descriptor, if boxtype in osscam is set to "pc" or "pc-nodmx", then oscam will assume that the ca device is the same as the adapter device. I think this should also work.

4. Use the 0x84 descriptor for sending the pmt pid. It should look like this: 84 02 PP PP, where 02 is the length and PP PP is the 2 byte pmt pid.

Doing so, will make your code very much cleaner, without extra checks and conditions.
You could send these descriptors in whatever order you feel like...
In my honest opinion this solution will be the best for both tvh and oscam users.

Regards,
nautilus7

#5 - 2019-06-10 08:49 - Jaroslav Kysela

Look again to the old oscam code - original getDemuxOptions() function. Reordering or rewriting the pmt tag data will not work with old oscam versions. If you show me the code which will work correctly with the old oscam, I am sure that I can apply it. But you introduced the incompatibility although the implementation may be better in oscam now.

#6 - 2019-06-10 09:21 - Jaroslav Kysela

- Subject changed from CA PMT descriptor used incorrectly to CA PMT descriptor parsing changed in recent oscam

#7 - 2019-06-10 09:54 - nautilus7 nautilus7

So we should keep writing shitty code just for compatibility?

Why do you care about old oscam versions?
It's just a matter of seconds for someone to update oscam in his machine!

You can do this change in your development (master) branch. You should not care about compatibility in your dev branch.
OSCam does not have a development branch, so every change can be considered experimental or not-compatible with the previous code. None has a problem with that.

Furthermore, you have missed what is the cause of this problem. The problem is not oscam. The problem is tvh which uses the 0x82 descriptor differently depending on its position in the capmt message.

I explained that in my first post. I also asked you to explain what information is tvh supposed to send in the 0x82 descriptor, so I could fix the problem from the oscam side.

You just said that you always send the same info, which is clearly not true. Your code has 2 different placements for the 0x82 descriptor with different info each time. Which one is sent to oscam? And what is the purpose of the other one?

#8 - 2019-06-10 12:35 - Jaroslav Kysela

nautilus7 nautilus7 wrote:

> So we should keep writing shitty code just for compatibility?
>
> Why do you care about old oscam versions?

I just care.

> **Furthermore, you have missed what is the cause of this problem. The problem is not oscam. The problem is tvh which uses the 0x82 descriptor differently depending on its position in the capmt message.**

The PMT message composition is based on the previous oscam code.

> I explained that in my first post. I also asked you to explain what information is tvh supposed to send in the 0x82 descriptor, so I could fix the problem from the oscam side.

I already answered this. It is oscam issue when you change the parsing of PMT data. It is a bit irrelevant, if you are trying to cleanup something and break things working for years.

> You just said that you always send the same info, which is clearly not true.

Nope. See my answer in comment 3. I know that there are two different first data byte usages, but everything is based on the old oscam code:

```
FIRST (ca_mask):
```

```
        if(buffer[17] == 0x82 && buffer[18] == 0x02)
        {
                // enigma2
                *ca_mask = buffer[19];
                uint32_t demuxid = buffer[20];
                if (demuxid == 0xff) demuxid = 0; // tryfix prismcube (0xff -> "demux-1" = error! )
                *demux_index = demuxid;
                if (buffer[21]==0x84 && buffer[22]==0x02) *pmtpid = b2i(2, buffer+23);
                if (buffer[25]==0x83 && buffer[26]==0x01) *adapter_index=buffer[27]; // from code cahandler.cp
p 0x83 index of adapter
        }
```

SECOND (demux_index):

```
        if((cfg.dvbapi_boxtype == BOXTYPE_PC || cfg.dvbapi_boxtype == BOXTYPE_PC_NODMX || cfg.dvbapi_boxtype =
= BOXTYPE_SAMYGO)
                 && buffer[7] == 0x82 && buffer[8] == 0x02)
        {
                *demux_index = buffer[9]; // it is always 0 but you never know
                *adapter_index = buffer[10]; // adapter index can be 0,1,2
                *ca_mask = (1 << *adapter_index); // use adapter_index as ca_mask (used as index for ca_fd[] a
rray)
                if (buffer[21]==0x84 && buffer[22]==0x02) *pmtpid = b2i(2, buffer+23);
        }
```

> Your code has 2 different placements for the 0x82 descriptor with different info each time. Which one is sent to oscam? And what is the purpose of the other one?

I also already answered this. Just compare the old and new oscam code.

#9 - 2019-06-10 13:03 - nautilus7 nautilus7

Jaroslav Kysela wrote:

> I know that there are two different first data byte usages, but everything is based on the old oscam code:

You keep referring to the old oscam code. The old oscam code is based on enigma2.
enigma2 guys created the private descriptors (0x81, 0x82, 0x83, x084, 0x85) over a decade ago.

They use them in a specific way.

They have the right to do so, because they are the creators.
Everybody else (including oscam and tvh) should conform to that.

You just admitted that you fill the first data byte in the 0x82 descriptor differently based on your preference/usage.
If you want to send different information, you just create a new descriptor.
You can't misuse already existing descriptors, because then you start breaking things or you're forcing people do hacks in their code to support what you're doing wrong.

Just look into your ca pmt code. Does it look clean? I see lots of various checks...
I made a suggestion to simplify your code by a huge amount (which will also make oscam code 3 lines sorter).
I wont insist further. If you like what you see in your code, it's fine for me.

Regards.

#10 - 2019-06-10 18:24 - Jaroslav Kysela

nautilus7 nautilus7 wrote:

> Jaroslav Kysela wrote:
>
> You just admitted that you fill the first data byte in the 0x82 descriptor differently based on your preference/usage.

Not my/tvh but enigma2 usage. The old code for 0x82 (especially)/0x84/0x83 defined the interface. You changed this. I really wonder how your change does work with all enigma2 boxes (including the old versions).

#11 - 2019-06-10 18:55 - nautilus7 nautilus7

Jaroslav Kysela wrote:

> Not my/tvh but enigma2 usage. The old code for 0x82 (especially)/0x84/0x83 defined the interface. You changed this. I really wonder how your change does work with all enigma2 boxes (including the old versions).

For the last time... enigma2 defined the interface, not oscam.

enigma2 still works with oscam, because oscam code is absolutely correct. Your tvh is wrong here. I am saying this from the first post.

The fast that you don't understand WHY enigma still works with latest oscam, shows that you are unaware and ignorant of what a descriptor is and how the descriptor parsing code works.

#12 - 2019-06-10 20:20 - Jaroslav Kysela

Ok, I see the problem now. You don't parse tvh (network dvbapi - PC type) 0x82 message in the enigma2 block but in the PC block which is incompatible:

```
  New code:
```

```
                               if(descriptor_length == 0x02 && (cfg.dvbapi_boxtype == BOXTYPE_PC ||
                                        cfg.dvbapi_boxtype == BOXTYPE_PC_NODMX || cfg.dvbapi_boxtype == BOXTYP
E_SAMYGO))
                                {
```

```
  Old code:
```

```
     if((cfg.dvbapi_boxtype == BOXTYPE_PC || cfg.dvbapi_boxtype == BOXTYPE_PC_NODMX || cfg.dvbapi_boxtype == B
OXTYPE_SAMYGO)
                 && buffer[7] == 0x82 && buffer[8] == 0x02)
```

So enigma2 works like before, but the network dvbapi clients are broken because dvbapi_boxtype checks. TVH assumes that the 0x82 message will be parsed in the enigma2 block for the specific PMT data order:

```
                                else if(descriptor_length == 0x02) // enigma2
                                {
                                        *ca_mask = buffer[pos + 2];
                                        uint8_t demux_tmp = buffer[pos + 3];

                                        if(demux_tmp >= 8 && *ca_mask == 0) // openpli based images
                                        {
                                                *ca_mask = 1 << demux_tmp;
                                        }

                                        if(demux_tmp == 0xFF) // tryfix prismcube (0xFF -> "demux-1" = error!)
                                        {
                                                demux_tmp = 0;
                                        }
                                        *demux_index = demux_tmp;
                                        is82Parsed = true;
                                }
                                break;
```

I omit "ignorant" word from my response. You don't offer any clean solution than to theck the oscam revision to keep compatibility with the older oscam versions. Perhaps, oscam might add a short note to the capabilities that the PMT parsing changed (look for DVBAPI_SERVER_INFO).

#13 - 2019-06-10 22:04 - nautilus7 nautilus7

Yes, the problem is what you described.

But why it was created after the 11520 change???
The answer is: because tvh can send the 0x82 descriptor either in the form of line 2247 or in the form of line 2267, what you call "pc" form/block and "enigma" block/form. This is what I am telling you since my first post!!! This shouldn't happen!

So, I ask again:

If tvh assumes the "enigma2" block/form is used in oscam, then why there is another form? When is it used? Perhaps by other softcams (cccam, mgcamd, etc)? (I don't know if tvh supports other softcams). Is the "pc" block/form send by tvh to oscam at any situation, or the "enigma" form is always sent to oscam?

Please answer these questions as clearly as you can. I can then propose clean solutions for both oscam and tvh.

#14 - 2019-06-11 05:03 - saen acro

With witch code TVH work with WiCard?

#15 - 2019-06-11 06:27 - Jaroslav Kysela

See my comment 3. The blocks are combined, because older oscam before revision 11396 didn't parse PMT PID for the PC code block, so the engima2 code block was forced for adapters 0-7 for the better functionality. Dot.

#16 - 2019-06-11 09:29 - nautilus7 nautilus7

Oscam code before 11520 was just wrong! It was not wise to parse the descriptors assuming a fixed position for each one. That's why each descriptor has a different tag. So you can identify them from that. Not from its position.

Instead of asking the oscam guys to properly fix the problem back then, you started adding tons of "bad" code in tvh. Or maybe the oscam guys were not willing to change their code, I don't know. And it doesn't matter now.

Now, we come to the present day, where the oscam parsing code is proper/fixed and reads the descriptors correctly without caring about their position in the capmt message. Oscam realized the problem and it got it fixed in 11520. Finally!

What I see here is an oportunity for you as well to get rid of all the dirty code, and make your life easier. Just drop all the bad code and write a few lines of proper code.

Unfortunately, you are not willing to understand that and you want to continue writing improper and dirty code, with the excuse of backwards compatibility. I can't get it. Why a programmer would prefer his code being that bad is beyond me.

You of course have the right to do whatever you want with your code, but in this case, if oscam wants to support tvh, it needs to make its code a little bit worse than it currently is.

#17 - 2019-06-12 15:07 - Jaroslav Kysela

- Status changed from New to Fixed

- % Done changed from 0 to 100

Applied in changeset tvheadend|bc769bfa9260bad6e1caa0c95591b70ae25f47bf.

#18 - 2019-06-12 15:07 - Jaroslav Kysela

We have just different opinion in the compatibility behavior. I prefer to not break things working for years, but apparently, you're on a different side. Fixed in recent 4.2 and 4.3 (master) trees.

#19 - 2019-06-13 09:36 - nautilus7 nautilus7

Your ca pmt code checks for oscam version and acts accordingly. It checks for 4-5 different oscam versions, most of which are **4-5-6 years old**. Do you realize that?

Nobody uses these ancient oscam versions. This is not called backwards compatibility... It is called "not willing to move forward". Your thinking is like your code: deprecated.

Regards.

#20 - 2019-06-13 18:14 - matias segurola

nautilus7 nautilus7 wrote:

> Your ca pmt code checks for oscam version and acts accordingly. It checks for 4-5 different oscam versions, most of which are **4-5-6 years old**. Do you realize that?
>
> Nobody uses these ancient oscam versions. This is not called backwards compatibility... It is called "not willing to move forward". Your thinking is like your code: deprecated.
>
> Regards.

haha have compatibility with oscam 5 years ago is a stupid Mr. developer is not a fool and work on improving the code not to have compatibility with those clunks of oscam from years ago, work so that tvheadend support namespace as an enigma hash asi work with oscam emu for a simpler softcam.key file

Jaroslav Kysela wrote:

> Applied in changeset tvheadend|bc769bfa9260bad6e1caa0c95591b70ae25f47bf.

#21 - 2019-06-13 19:04 - Jaroslav Kysela

Guys, if you are so clever, go ahead and be the tvh developer. Anyway, I feel that my tvh contributions are at the end those months. My interest is lower and lower.

#22 - 2019-06-13 19:56 - Pablo R.

matias segurola wrote:

> haha have compatibility with oscam 5 years ago is a stupid Mr. developer is not a fool and work on improving the code not to have compatibility with those clunks of oscam from years ago, work so that tvheadend support namespace as an enigma hash asi work with oscam emu for a simpler softcam.key file

Is it necessary to underestimate the work of others and the hours of dedication disrespecting others? I do not think so. sad.png

What's more, it seems to me that the work done on tvheadend in recent years has been several times superior to oscam, which had no updates (or poor updates) for years. Now they want to change any code that crosses them.

#23 - 2019-06-13 20:34 - matias segurola

Pablo R. wrote:

> matias segurola wrote:
>
>> haha have compatibility with oscam 5 years ago is a stupid Mr. developer is not a fool and work on improving the code not to have compatibility with those clunks of oscam from years ago, work so that tvheadend support namespace as an enigma hash asi work with oscam emu for a simpler softcam.key file
>
> Is it necessary to underestimate the work of others and the hours of dedication disrespecting others? I do not think so. sad.png
>
> What's more, it seems to me that the work done on tvheadend in recent years has been several times superior to oscam, which had no updates (or poor updates) for years. Now they want to change any code that crosses them.

friend nobody could ever speak ill of the very good work of the team of tvheadend is respected only in this case if that improvement would help tvheadend and users, and the old oscam then to update nothing more nothing can reproach tvheadend since that if it would be lack of education and moral ethics

#24 - 2019-06-13 20:38 - matias segurola

it is only asked to modify that part of the code that one day helped many of us and today unfortunately it hurts and leaves without many benefits of oscam to the users of tvheadend it is not required to restart the code only to modify integrate it with oscam to one hundred to improve nothing else , you do not have to take that badly if very good, this is as if the creators of the common nafta opened said not so that to create the super if in that way we leave many cars outside always forwards the back ones go to the view or they will use compatible headend and oscam views and today we use these improvements that would help us and much

Files

a.patch	1.25 KB	2019-06-09	Jaroslav Kysela