Tvheadend - Bug #4519

MUXs allowed with duplicated transport_stream_id on the same network

2017-08-05 14:59 - J Blanc

Status:	New		**Start date:**	2017-08-05
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	DVB		**Estimated time:**	0.00 hour
Target version:	4.6			
Found in version:	4.2.3-19~g490b6f213		**Affected Versions:**	

Description

transport_stream_id (Reported within TVH as TransponderID) is a unique identifier within a network, but TVH allows there to be multiple MUX definitions on the same network with the same transport_stream_id .

This can be reproduced by tuning twice on the same transponder, by setting slight changes in the original tuning value. For instance 11777H and 11778H will tune to the same transponder. The same transponder will be tuned to by initial tuning values close enough within the seeking range of the hardware tuner. The original tuning value is not the actual tuning to which the hardware device settles, because that changes every tuning as the device seeks the best signal, and it should not be used as a unique identifier.

This can occur naturally, when transponders are reorganised at the source, or more commonly when the NIT is updated with slightly changed tuning values to reduce STB tuning seek times. Because it appears the TVH does not use transport_stream_id as unique identifiers, this means these updates can create duplicates of the same transponder.

This may be the root cause of a lot of bugs, most obvious is the "duplicate Bouquet channels" issue that can occur with duplicate transponders present.

History

#1 - 2017-08-06 17:39 - Jaroslav Kysela

- Target version set to 4.4

At the time, the user intervention is required - the invalid muxes (incorrect parameters) should be set to 'Ignore' (enable field). Unfortunately, the mentioned inconsistency is from the broadcaster(s). I plan to add more logic to have enabled only one mux for the 'overlapped' parameters and migrate services to this mux, but I have not time to implement this yet.

#2 - 2017-08-06 17:56 - J Blanc

This is not a broadcaster error, broadcasters are explicitly allowed to change the tuning values for transport_stream sections in the NIT, this is intended functionality to allow broadcasters to update tuning values on STBs. The unique identifier for a transport_stream is the transport_stream_id.

There should never be two transponders with the same transport_stream_id on a network, and TVH is in error for allowing that. I know of no DVB networks that allow this, because it would break some basic utility of STBs.

Special logic for detecting "overlapping" transponders isn't actually needed. Use the transport_stream_id as the key identifier for a transponder, not the tuning parameter, and consistency should just happen.

Is there a reason why you can't change the schema for storing transponders to have the key field be network_id and transport_stream_id?

#3 - 2017-08-06 21:09 - Jaroslav Kysela

Have you tried to analyze these mux parameters broadcasted in NIT on some big satellites like 19.2E, 13E etc.? There are many inconsistencies (mostly small like different rolloff, but some changes are major). The broadcasters forgot to remove the old info on some transponders for example. You're speaking about an ideal world.

#4 - 2017-08-06 21:39 - J Blanc

The correct way to handle where there's a broken NIT stream from specific transponders, is to allow the user to blacklist transponders with incorrect NITs. Otherwise there is no way you can actually use NIT streams to properly update transponders, because you need to use the unique ID as a unique ID.

I'm unable to check 19.2 E, because I can only get reception from 28.2 E, which has correct and consistent unique network_id values. I would be surprised that 19.2 E has inconsistent NIT streams when both it and 28.2 E are SES operated. Is this in 0x40 tables, or 0x41 or other descriptors? It's not explicit in the spec, but only 0x40 are 'actual network' authoritative descriptors.

#5 - 2017-08-16 21:50 - J Blanc

Also, it occurs to me that a Transponder will always be authoritative in identifying it's own tuning values. Tuning values that contradict those provided by the Transponder it's self should be deactivated or discarded?

#6 - 2017-08-17 09:01 - Jaroslav Kysela

Some muxes does not have the mux tuning parameters.

#7 - 2017-08-17 09:40 - J Blanc

For each network_id+transport_stream_id pair, store the last N seen satellite_delivery_system_descriptor entries (unique by origin network_id+transport_stream_id pair). Actual tuning value used for the network_id+transport_stream_id pair is the plurality of common satellite_delivery_system_descriptor entries.

That should maintain consistency as long as N is more than twice the number of common faulty entries. Can you estimate a good value of N based on the number of common faulty entries you've seen on 19.2 E?

#8 - 2019-10-23 20:55 - Joe User

While this is a problem, I would not consider it a tvheadend "bug". First due to reality vs theory and second due to tvheadend's concept of "network" is not necessarily the same as yours. With tvheadend, I often use one "network" to include muxes from different providers on the same satellite. It would be terrible to have to create a new network for each mux. Also one would have to create a new network for each "feed" on some satellites if tvheadend was to restrict transport id to be unique. For example, on 4.9E, muxes at 12322 and 12360 both use 94 for network id and 1 for transport id, but are from the same "provider". Yes, it is a set of feeds, but it is not uncommon. Also one cable provider has some duplicate tsids and although not according to "spec", it is not tvheadend's job to enforce a "spec", it is to make it work. smile.png

#9 - 2019-10-23 22:35 - J Blanc

Using TIDs as a unique identifier for a network is a required part of correctly handling MUXs changing their tuning values. If you want fully automatic bouquet management, it's a requirement.

Proper handling of networks that, for what ever reason, have intentional duplicate NID TID pairs for different transponders, would be to flat ignore the network table for transponder updates because it is incorrect, and allow duplicates.

If full tuning update management is implemented, there probably should be an "Allow duplicate TIDs" option that also disables updates for these problematic providers?

#10 - 2019-10-24 08:42 - Joe User

J Blanc wrote:

> Using TIDs as a unique identifier for a network is a required part of correctly handling MUXs changing their tuning values. If you want fully automatic bouquet management, it's a requirement.
>
> Proper handling of networks that, for what ever reason, have intentional duplicate NID TID pairs for different transponders, would be to flat ignore the network table for transponder updates because it is incorrect, and allow duplicates.

I understand that, but the tricky part (as always) is how to handle bad/inconsistent data??

> If full tuning update management is implemented, there probably should be an "Allow duplicate TIDs" option that also disables updates for these problematic providers?

Maybe that is not necessary, you just need to check if the "Network ID (limit scanning)" option is set to 0, then do not try to enforce. And/or one could simply disable network discovery.

#11 - 2019-10-24 11:22 - Joe User

see also here:
https://tvheadend.org/issues/4261

#12 - 2021-02-20 22:32 - Flole Systems

- Target version changed from 4.4 to 4.6