OsmoHNBGW - Bug #2435

Osmocom asn1c/libasn1c is based on old fork

08/14/2017 06:32 AM - laforge

Status:	New	**Start date:**	08/14/2017
Priority:	Normal	**Due date:**	
Assignee:	sysmocom	**% Done:**	0%
Category:			
Target version:			
Spec Reference:			

Description

We have a fork of asn1c/libasn1c that's from April 2015.

We should rebase our changes on current upstream master (which has picked up development speed again) and keep rebasing.

Related issues:

Related to OsmoHNBGW - Feature #2436: Test cases for asn1c APER encoding	**New**	**08/14/2017**
Related to OsmoHNBGW - Bug #2437: synchronize different asn1c APER forks	**New**	**08/14/2017**
Related to OsmoHNBGW - Feature #3862: Add support for asn1scc	**Rejected**	**03/26/2019**

History

#1 - 08/14/2017 06:34 AM - laforge

- Related to Feature #2436: Test cases for asn1c APER encoding added

#2 - 08/14/2017 06:38 AM - laforge

- Related to Bug #2437: synchronize different asn1c APER forks added

#3 - 01/20/2018 01:34 PM - laforge

FYI: I just used current asn1c master 4cc779fd9bd7f556699b5863cf111b359da10b66 (last commit November 21, 2017) to successfully compile HNBAP and RANAP from wireshark.gig:

```
asn1c -fcompound-names -fline-refs -funnamed-unions HNBAP-CommonDataTypes.asn  HNBAP-Constants.asn  HNBAP-Cont
ainers.asn  HNBAP-IEs.asn  HNBAP-PDU-Contents.asn  HNBAP-PDU-Descriptions.asn
asn1c -fcompound-names -fline-refs -funnamed-unions RANAP-CommonDataTypes.asn RANAP-Constants.asn RANAP-Contai
ners.asn RANAP-IEs.asn RANAP-PDU-Contents.asn RANAP-PDU-Descriptions.asn
asn1c -fcompound-names -fline-refs -funnamed-unions RUA-CommonDataTypes.asn  RUA-Constants.asn  RUA-Containers
.asn  RUA-IEs.asn  RUA-PDU-Contents.asn  RUA-PDU-Descriptions.asn
```

This means it is capable to parse the full extent of information object classes contained in the relevant 3GPP specs, without relying on any asn1tostruct.py hacks!

It still is missing the APER and type prefixing which we need, so those would have to be forward-ported, and our existing code written against the assumptions of asn1tostruct.py ported over.

#4 - 02/25/2018 11:44 PM - neels

note that there already is a branch aper-prefix-onto-upstream which I had rebased onto the upstream master at that time. Last time I checked it worked well (with our current asn1tostruct.py affairs). It is probably far behind the current master, but would be a good basis to avoid some of the merge conflicts already resolved there.

Also note that our current asn1c.git repository contains **two completely separated histories**. In my gitk --all it is visible by showing the aper-prefix-onto-upstream branch, with a new "Initial" commit right above that and a looong history towering on top of that; notice there is no line connecting the two, so it is a completely separate initial commit.

About all of this, I wrote:
http://lists.osmocom.org/pipermail/openbsc/2016-July/009488.html
"merging iu specific branches to master"
Wed Jul 6 16:58:40 UTC 2016

```
> >   asn1c: * aper-prefix

First of all, it looks like we imported from svn, but Lev Walkin later did
another migration to git, which we fetched as well. So our master is on our old
svn import, while the upstream master has different hashes in its history.  Our
aper-prefix branch is based on the upstream history, not our "stale" master,
which made a rebase a bit easier.

Furthermore, on https://github.com/vlm/asn1c/commits/master there are scores of
new commits we don't have in our asn1c.  our last commit from Lev Walkin is
from 2015-04-28, "?= was confusing some environments", 62913d8b8e1eb96d74315ff

I have thus:
* fetched upstream master from github's vlm/asn1c,
  pushed as new branch vlm/master to our git.osmocom.org/asn1c
* rebased our aper-prefix branch to that master (with minor conflicts),
  pushed as new branch aper-prefix-onto-upstream

We should probably:
* reset --hard our master to vlm/master
* reset --hard our aper-prefix to aper-prefix-onto-upstream (after testing)
```

The last of which hasn't happened yet but still makes sense to me today, followed by pulling in the updated current master and rebasing our (still required) changes onto that.

(I guess we also might want to get rid of that disjunct evil twin history in our asn1c.git somehow.)

#5 - 03/26/2019 03:10 PM - msuraev

- Related to Feature #3862: Add support for asn1scc added