Continued from PART III

UPshare and DOWNshare: limiting access to Caid's, Providers, and Services.

CCcam offers the possibility to set each cards access independently. The syntax is as follows:

F: <username> <password> <uphops> <shareemus> <allowemm> { caid:id:downhops, caid:id:downhops, ... } { caid:id:sid, caid:id:sid, ... }

To understand how to limit access to each card, let's look at the following example

F: user2 pass2 0 1 0 { 0100:000080, 0622:000000:1, 0500:000000:2 }

<uphops> is set to 0, the client can only access the server local cards
<shareemus> is set to 1, the client can have access to the server local key file
<allowemm> is set to 0, the client cannot do any remote EMM
<caid:id> is set to 0100:000080, the client does not have access to Canal+ France card
<caid:id:downhops> is set to 0622:000000:1, the client receives Canal+ NL, but cannot reshare it
<caid:id:downhops> is set to 0500:000000:2, the client receives a Viacces provider, and can reshare it 1 hop further

Therefore, in addition to limiting server access, <caid:id:downhops> allows the server to define how far the client can reshare the server cards. This is set in the downhops field. Let's look at the following example:

F: user3 pass3 5 0 1 { 0:0:3, 0100:000080:1 }

<uphops> is set to 5, the client receives all the cards up to a distance of 5 hops from the server
<shareemus> is set to 0, the client does not have access to the server local key file
<allowemm> is set to 1, the client receives remote EMM
<caid:id:downhops> is set to 0:0:3, the client receives all the cards, and can reshare them 2 hops further
<caid:id:downhops> 000080:1, with the exception of this card, that cannot be reshared

The option <caid:id:sid> allows the server to limit access even to a specific TV service.

PART IV

Start and stop scripts

So far, we used Telnet to start CCcam and check our tests.
We shall now look at the scripts that start and stop CCcam. The script procedure changes according to the image in use.

Gemini

All the scripts for Gemini images are found in the /usr/script directory. It is possible to have listed in Blue Panel any EMU related script, if these are starting with _cam (e.g. CCcam_cam.sh).
Note: to enable any script, its attributes must be set to 755
The Gemini original scripts start with a list of CAMID's. A sector is allocated to each EMU. This list of CAMID's is only providing information and can be ignored.
CAMID must be set as follows:

CAMID=6000

CAMID is required to sort the Blue Panel list.Each CAMID must be listed only once. This will enable the Gemini image to identify the EMU by its number.
If the Dreambox requires it, then Gemini will search the script according to the number saved during its creation and will initialize it. If Gemini does not find the script, the following error message will be returned “camid not found”
This is not a fatal error. It means that no script has been found with that CAMID number.
The available scripts (ending with _cam.sh) can be read when starting Enigma. They can also be read from the Blue Panel, Cam Settings, ALL cam, Update.
Therefore, no need to worry if an EMU is not listed in the Blue Panel.

CAMNAME="CCcam-1.2.0"
CAMNAME represents the name that will be displayed in the Blue Panel. The length of the name must be set according to the skin in use. Else, it may not appear properly, or completely

ZAPTIME=6
When changing from the current EMU to a different EMU, ZAPTIME sets the time required by the EMU in use to display a service

INFOFILE="ecm.info"
This is the file where the current ECM are saved

case "$1"
This is where the start/stop script begins

The start/stop setting is read at the start of the script

```
start)
echo "[SCRIPT] $1: $CAMNAME"
/var/bin/CCcam -v
;;
```

start)
is the value of the setting, and will start the following script

ECHO "(script) $1: # $$CAMNAME>"
this is simply displaying the script name, the setting value and the name of the CAM

/var/bin/CCcam – v
Cccam is now started; in the example the –v (verbose) setting is used

;;
End of the start script

```
stop)
echo "[SCRIPT] $1: $CAMNAME"
killall CCcam
;;
```

stop)

this is the setting value and the following script will be launched

ECHO "(script) $1:" # $$CAMNAME>"-

killall CCcam
the EMU stops

;;
End of the script

```
*)
$0 stop
exit 1
;;
esac
```

If, in addition to CCcam, you want to start NewCS, then you must add it. However, it is necessary to hold CCcam for 3 seconds. This will prevent CCcam to read the CI slots before NewCS.

```
echo "[SCRIPT] $1: $CAMNAME"
/var/bin/newcs
sleep 3
/var/bin/CCcam -v
..
```

If you want to stop them

```
..
echo "[SCRIPT] $1: $CAMNAME"
killall newcs
killall CCcam
..
```

The names must be correct, respecting also the capital letters.
To try the script with Telnet simply type

/var/skripts/emu_cam.sh start