NPTEE Howto
Using pipes to outstream Allstar audio

One of the major problems when using ezstream to send audio to Broadcastify is the dependence it has on its connection to Allstar. Unless you properly shutdown the processes it can hang the system so updates with restarts or reboots are a real pain. The other problem is the ability to have multiple streams.

nptee or "named pipe tee" solves this problem by providing a process in between Allstars outstream command and the actual program that is receiving the audio data like lame or ezstream or any other audio process.

nptee is located in the /usr/local/bin directory of Hamvoip as of the latest update. Here is how it is used with two pipes. You can use up to 10 simultaneous pipes.

In the Allstar rpt.conf file the outstream command would look like this -

outstreamcmd=/usr/local/bin/nptee,pipe1,pipe2

The stream named pipes in this example are:

/tmp/outsound/pipe1
/tmp/outsound/pipe2

The pipe names are arbitrary. Here is an example with all 10

outstreamcmd=/usr/local/bin/nptee,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10

which would create theses files -

/tmp/outsound/p1
/tmp/outsound/p2
/tmp/outsound/p3
/tmp/outsound/p4
/tmp/outsound/p5
/tmp/outsound/p6
/tmp/outsound/p7
/tmp/outsound/p8
/tmp/outsound/p9
/tmp/outsound/p10

Then using the first example you could create two paths to ezstream using these commands in a script -

lame --preset cbr 16 -r -m m -s 8 --bitwidth 16 /tmp/outsound/pipe1 - 2>/tmp/status | ezstream -qvc /etc/ezstream1.xml

lame --preset cbr 16 -r -m m -s 8 --bitwidth 16 /tmp/outsound/pipe2 - 2>/tmp/status | ezstream -qvc /etc/ezstream2.xml

Note that these scripts use two different ezstream xml files going to two different accounts and passwords as specified in the ezstream1 and ezstream2 xml files.

This could be repeated for up to 10 scripts and streams.

The nptee program isolates asterisk/app_rpt from the client applications. So, if lame or ezstream needs restarting you can just kill and restart them. Asterisk would not be affected.

The ntpee program also includes 1 minute of audio buffering so if a process is stopped and restarted within the minute no audio would be lost.

Since the pipe names are arbitrary naming them for the destination would make it more readable. Like -

outstreamcmd=/usr/local/bin/nptee,broadcastify1,broadcastify2

This creates two pipes in /tmp -

/tmp/outsound/broadcastify1
/tmp/outsound/broadcastify2

Then you create a script "start_ezstream"with these commands, recommend putting it in /root -

#!/bin/bash

lame --preset cbr 16 -r -m m -s 8 --bitwidth 16 /tmp/outsound/broadcastify1 - 2>/tmp/status | ezstream -qvc /etc/ezstream1.xml

lame --preset cbr 16 -r -m m -s 8 --bitwidth 16 /tmp/outsound/broadcastify2 - 2>/tmp/status | ezstream -qvc /etc/ezstream2.xml

echo -e "\nBroadcastify started\n"

#end script

Then make the script executable -

chmod 750 start_ezstream

Then call this in /etc/rc.local so it starts at boot -

/root/start_ezstream

Now your Allstar and Broadcastify stuff are not linked so that you can easily restart Allstar without interrupting Broadcastify.

Typically you would only have one Broadcastify stream going out so it would look like this in rpt.conf -

outstreamcmd=/usr/local/bin/nptee,broadcastify

and the script would have -

lame --preset cbr 16 -r -m m -s 8 --bitwidth 16 /tmp/outsound/broadcastify - 2>/tmp/status | ezstream -qvc /etc/ezstream.xml