TestDevLab

How We Test Audio Quality In VoIP Applications

Rihards Skrebelis




Why it has to be tested?

What goes into VoIP application?

- Signaling
  - Establish call
  - Handover
  - Multiple person call
- Audio/Voice
  - Codec
  - Performance
  - Quality
  - Handling various quality internet
  - Handling internet interruptions
  - Echo cancelling
  - Background noise reduction
  - etc.

What has to be tested in VoIP application

- Signaling
  - Establish call
  - Handover
  - Multiple person call
- Audio/Voice
  - Codec
  - Performance
  - Quality
  - Handling various quality internet
  - Handling internet interruptions
  - Echo cancelling
  - Background noise reduction
  - etc.

Our specialization

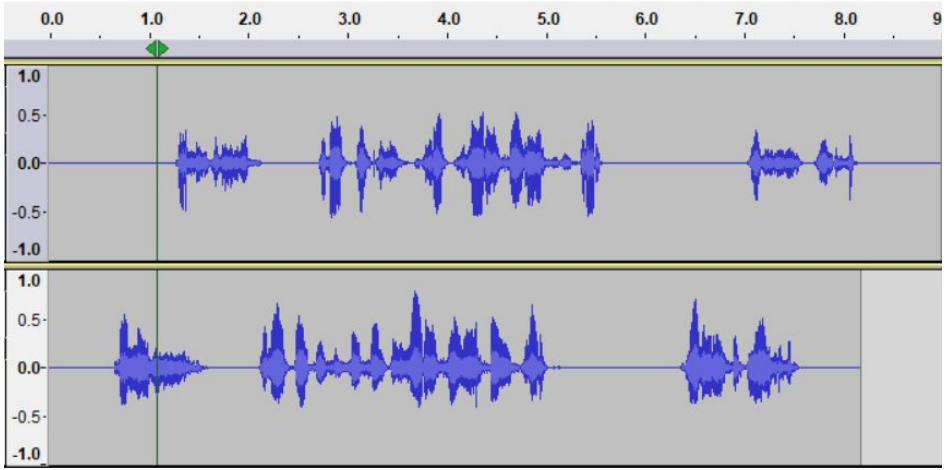
MOS and Delay under various network conditions

MOS

- Started as manual evaluation
- Taken over by algorithm
- Requires reference and degraded file
- In our case we purchased licensed software

Delay

Simple end-to-end delay, between time when sound is sent and when it is received on other device.

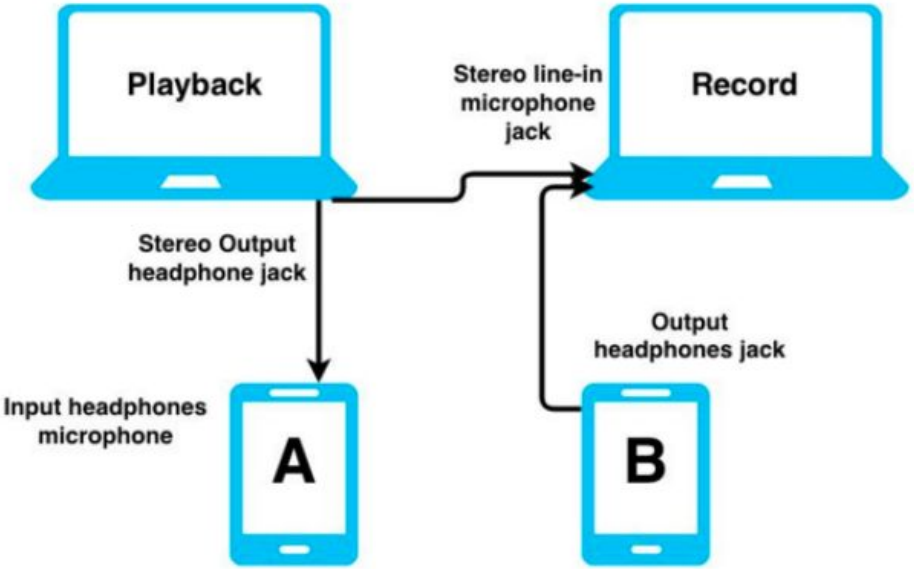


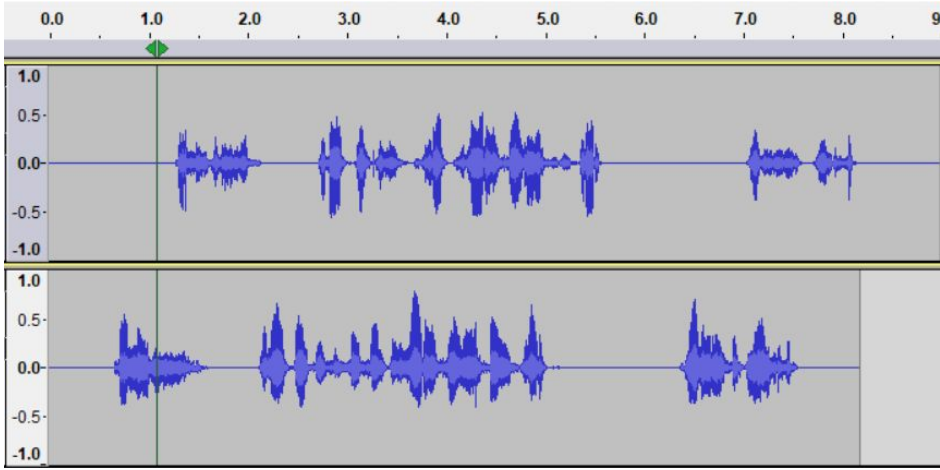
Methodology

We need to

- Control audio source
- Record audio source
- Record output audio
- Have recorded audio spend the smallest possible time outside of devices
- Control volume for input and output
- Platform independent

Audio hardware






SAMSUNG

Network conditioning

- Custom OS on network router
- Change parameters like
  - packet loss
  - bandwidth
  - jitter
  - delay
- Accessible UI or SSH

Making our work easier- automating as much as we can

- Things we have to do
- Start a call
- Start audio playback
- Start audio recording
- End recording call and playback
- Split audio track
- Give audio to tool
- Gather results
- Change network conditions

Things we can automate

- Start a call
- Start audio playback
- Start audio recording
- Split audio track
- Give audio to tool
- Gather results
- Change network conditions

```
./play $1 & ./rec $2 trim 0 9
sox $2 "$2_original.wav" remix 1
sox $2 "$2_degraded.wav" remix 2
sox "$2_original.wav" -b 16 "$2_original1.wav"
sox "$2_degraded.wav" -b 16 "$2_degraded1.wav"

echo "----- Remote server MOS evaluation -----"
curl -X POST -H "Cache-Control: no-cache" -H "Content-Type: multipart/form-data;
boundary=----WebKitFormBoundary7MA4YWxkTrZu0gW" -F "1=@$2_original1.wav" -F
"2=@$2_degraded1.wav" "http://10.1.5.58:8080/MOS/UploadServlet"

rm "$2_original.wav"
rm "$2_degraded.wav"
rm $2
```

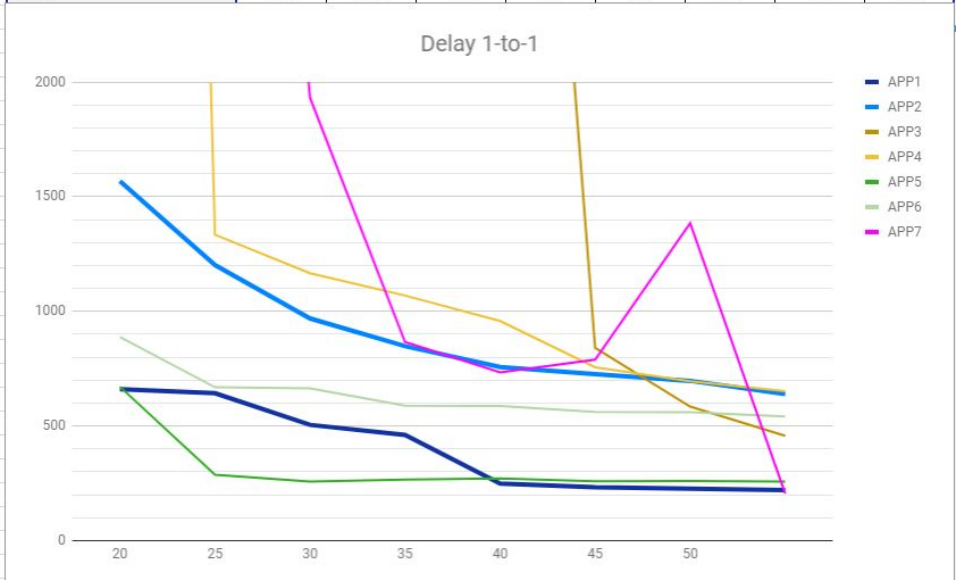
And we end up with this

MOS

	20	25	30	35	40	45	50	Unlimited
APP1	2.49	2.5	2.92	3.15	3.3	3.35	3.36	3.46
APP2	2.03	2.26	2.84	3.05	3.17	3.26	3.38	3.42
APP3	1.04	-	-	1.75	2.15	3.58	3.54	3.71
APP4	1.03	1.99	2.2	2.35	2.79	3.41	3.57	3.6
APP5	2.62	3.2	3.61	3.59	3.6	3.63	3.61	3.64
APP6	2.7	2.98	3.37	3.43	3.23	3.27	3.41	3.38
APP7	1.38	2.13	2.29	3.05	3.13	3.11	2.55	3.68

Delay

	20	25	30	35	40	45	50	Unlimited
APP1	661	643	505	461	249	233	227	220
APP2	1568	1202	969	848	757	726	697	639
APP3	10760	-	-	7370	6135	841	585	457
APP4	16250	1334	1166	1069	958	756	695	652
APP5	671	287	258	267	271	259	260	258
APP6	888	669	664	589	588	561	560	541
APP7	6673	6401	1931	866	734	789	1385	205



More topics if we have time

- Jitter influence on quality and how it is handled
- The road to where we are now
- Improvements on the automation and scripts

Questions?