absorptions

random hacks & assorted infodumps. by windytan [oona räisänen]

Pea whistle steganography

by Oona Räisänen Wednesday, October 07, 2015

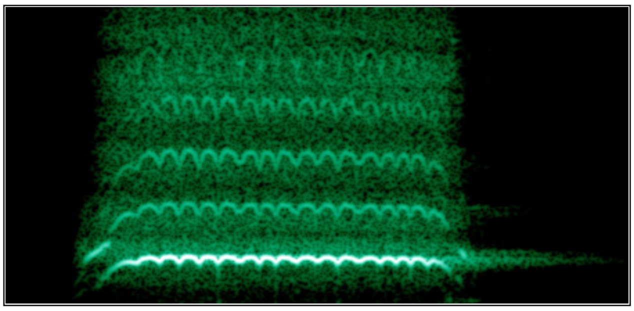
Would anyone notice if a referee's whistle transmitted a secret data burst?

I do really follow the game. But every time the pea whistle sounds to start the jam I can't help but think of the possibility of embedding data in the frequency fluctuation. I'm sure it's alternating between two distinct frequencies. Is it really that binary? How random is the fluctuation? Could it be synthesized to contain data, and could that be read back?

I found a staggeringly detailed Wikipedia article about the physics of whistles – but not a single word there about the effects of adding a pea inside, which is obviously the cause of the frequency modulation.

To investigate this I bought a metallic pea whistle, the Acme Thunderer 60.5, pictured here. Recording its sound wasn't straightforward as the laptop microphone couldn't record the sound without clipping. The sound is incredibly loud indeed – I borrowed a sound pressure meter and it showed a peak level of 106.3 dB(A) at a distance of 70 cm, which translates to 103 dB at the standard 1 m distance. (For some reason I suddenly didn't want to make another measurement to get the distance right.)

Later I found a microphone that was happy about the decibels and got this spectrogram of a 500-millisecond whistle.



▶ 0:01 🔊

The whistle seems to contain a sliding beginning phase, a long steady phase with frequency shifts, and a short sliding end phase. The "tail" after the end slide is just a room reverb and I'm not going to need it just yet. A slight amplitude modulation can be seen in the oscillogram. There's also noise on somewhat narrow bands around the harmonics.

The FM content is most clearly visible in the second and third harmonics. And seems like it could very well fit FSK data!

Making it sound right

I'm no expert on synthesizers, so I decided to write everything from scratch (whistle-encode.pl). But I know the start phase of a sound, called the attack, is pretty important in identification. It's simple to write the rest of the fundamental tone as a simple FSK modulator; at every sample point, a data-dependent increment is added to a phase accumulator, and the signal is the cosine of the accumulator. I used a low-pass IIR filter before frequency modulation to make the transitions smoother and more "natural".

Adding the harmonics is just a matter of measuring their relative powers from the spectrogram, multiplying the fundamental phase angle by the index of the harmonic, and then multiplying the cosine of that phase angle by the relative power of that harmonic. SoX takes care of the WAV headers.

Getting the noise to sound right was trickier. I ended up generating white noise (a simple `rand()`), lowpass filtering it, and then mixing a copy of it around every harmonic frequency. I gave the noise harmonics a different set of relative powers than for the cosine harmonics. It still sounds a bit too much like digital quantization noise.

Embedding data

There's a limit to the amount of bits that can be sent before the result starts to sound unnatural; nobody has lungs that big. A data rate of 100 bps sounded similar to the Acme Thunderer, which is pretty much nevertheless. I preceded the burst with two bytes for bit and byte sync (`0xAA 0xA7`), and one byte for the packet size.

Here's "OHAI!":

▶ 0:01 🔊

Sounds legit to me! Here's a slightly longer one, encoding "Help me, I'm stuck inside a pea whistle":

▶ 0:04 🔊

Decoding

Writing a receiver is left as an exercise, but it should be as simple as receiving FSK. The frequency can be determined using `atan2`, a zero-crossing detector, or FFT, for instance. The synchronization bytes are meant to help decode such a short burst; the alternating 0s and 1s of `0xAA` probably give us enough transitions to get a bit lock, and the `0xA7` serves as a recognizable pattern to lock the byte boundaries on.

6 comments:

konaya 07 October, 2015 07:47

”Would anyone notice if a referee's whistle transmitted a secret data burst?” — You, if anyone, would!

Anonymous 07 October, 2015 14:00

This whistle ["Help me, I'm stuck inside a pea whistle"] sounds pretty much like data transmission on an old modem which was plugged into the telephone line. So I guess one would notice that if it were used extensively.

Ron 08 October, 2015 04:21

Notice that there are dark bands horizontally between the sections of noise looking vertically, suggesting that the noise itself is generating harmonics. Maybe you could hide a signal in the harmonic noise leaving the fundamental alone to help preserve the character of the sound?

Oona Räisänen 08 October, 2015 11:20

The noise harmonics are parameterized in the encoder (line 26), so it shouldn't be difficult. But I think the noise sounding wrong is what already gives away the synthetic nature of the sound, and modulating it with data wouldn't necessarily improve that.

Andrew Skretvedt 15 October, 2015 14:29

Have you looked into the pealess designs? I love their simple physics for an un-jammable trill. I own a Fox 40 Classic. It's very loud. The package claims 115 dB and one soccer ref site measured it at 124 dB (one must be careful...sonic weapons here). I confirmed it as just barely audible from 700 m away during a gale. Mfg. claims up to 1-mile/1600m, which I believe if blown forcefully and in still air. I'd love to get Acme's Metropolitan 15. Heck, I'd like to start a collection. http://acmewhistles.ca/ http://www.youtube.com/fox40world

Oona Räisänen 16 October, 2015 10:37

Hmm, no I haven't. Do they also exhibit the frequency shift behavior?