

CTF Write-up — Jingle Bells

Category: Crypto / Audio

Points: 25

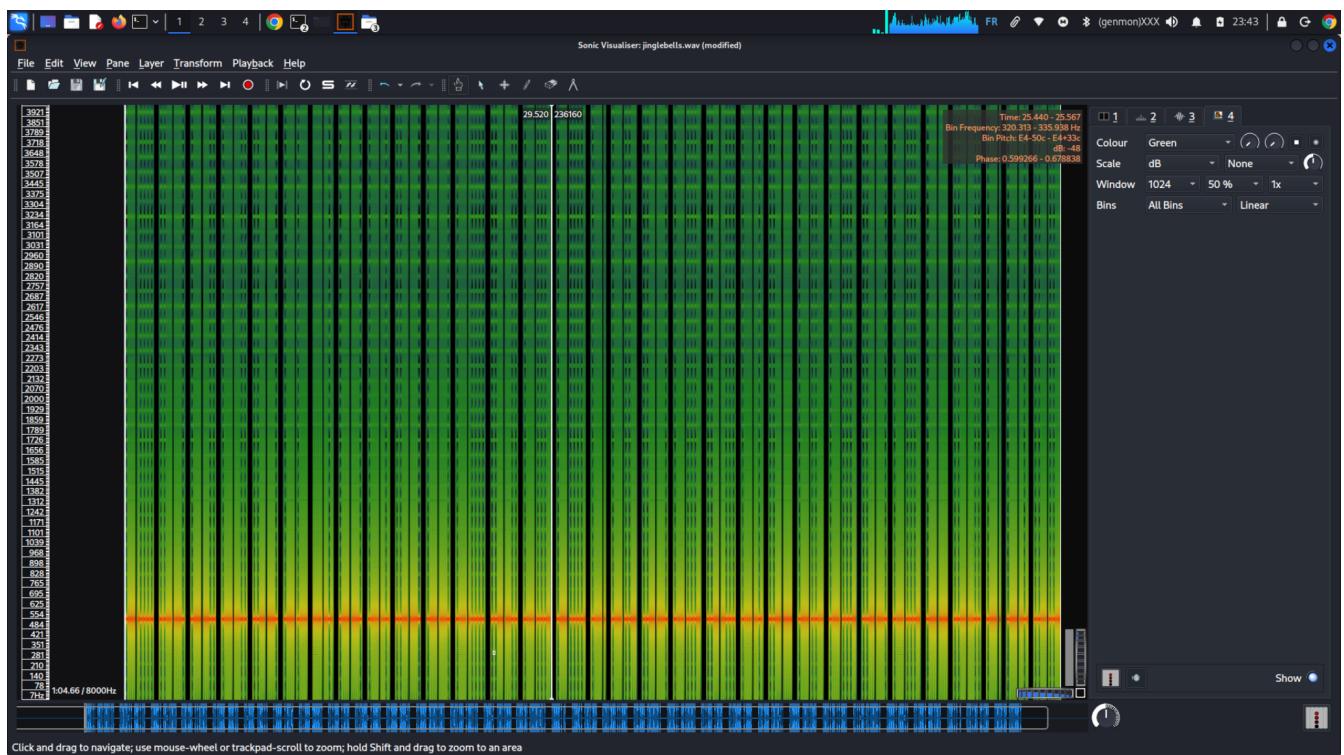
Author: p3f0rm3r ({x0x} Team)

Description

This challenge provides a WAV audio file that does not sound like Jingle Bells. The objective is to analyze the audio and extract the hidden message.

Step 1 — Spectrogram Analysis (Sonic Visualiser)

The audio file was opened in Sonic Visualiser and a spectrogram layer was added. The visualization revealed repeating short and long signals, indicating Morse code.



Step 2 — Morse Decoding (MorseXpress)

The audio file was uploaded to <https://morseexpress.com/morse-code-audio-decoder/>. The platform decoded the Morse signals and returned a hexadecimal string.

61 74 63 63 74 66 5F 64 6F 74 64 61 73 68 69 6E 67 74 68 72 6F 75 67 68 74 68 65 73 6E 6F 77

The screenshot shows the MorseXpress website interface. At the top, there's a navigation bar with links like 'Translator', 'Audio Decoder', 'Image Decoder', 'Decoding Tree', and a search bar. Below the navigation is a file upload section for 'jinglebells.wav'. Underneath, the 'Audio Visualization' section displays a barcode representation of the audio file, with a play button and some technical details: 'Total segments: 595 | Duration: 64.66s', 'Tone segments: 298 | Silence segments: 297', and 'Sample: #0 [0.00-0.19] tone | #1 [0.19-0.25] silence | #2 [0.25-0.32] tone'. The 'Decoded Text' section shows the Morse code output: '61 74 63 63 74 66 5F 64 6F 74 64 61 73 68 69 6E 67 74 68 72 6F 75 67 68 74 68 65 73 6E 6F 77'. There's also a 'Morse Code' section below it.

Step 3 — Hex Decoding (CyberChef)

The hexadecimal output was decoded using CyberChef with the 'From Hex' operation.

The screenshot shows the CyberChef interface. On the left, the 'Operations' sidebar lists various encoding and decoding options. In the center, under the 'Recipe' tab, the 'From Hex' operation is selected. The 'Input' pane shows two files: '1: PELCBFRPHEVGL' and '2: jinglebells.wav'. The 'jinglebells.wav' file is highlighted, showing its details: Name: jinglebells.wav, Size: 517 324 bytes, Type: audio/x-wav, Loaded: 100%. The 'Output' pane shows the decoded text: 'atcctf_dotdashingthroughthesnow'. At the bottom, there's a 'STEP' button with a 'BAKE!' button and an 'Auto Bake' checkbox.

Flag

Conclusion

This challenge combines audio analysis, Morse code decoding, and hexadecimal decoding. It demonstrates how multi-layered data can be hidden inside audio files.

Tools Used

- Sonic Visualiser
- MorseXpress
- CyberChef