Integer based buffer overflow vulnerability #320

Closed x00x00x00x00 opened this issue on Jun 6, 2021 · 1 comment

Labels bug

x00x00x00x00 commented on Jun 6, 2021

Hi Team,

Integer based buffer overflow caused by out-of-bound left shift is observed in miniaudio.h while fuzzing **MINIAUDIO (v0.10.35 and master branch)** using **UBSAN** enabled in **AFL FUZZER**

Vulnerable code from miniaudio.h -

```
DRWAV_API drwav_uint32 drwav_bytes_to_u32(const drwav_uint8* data)
{
return (data[0] << 0) | (data[1] << 8) | (data[2] << 16) | (data[3] << 24);
}
```

Steps to Reproduce -

```
cd examples
```

```
afl-gcc -fsanitize=address -fsanitize=leak -fsanitize=undefined simple_looping.c -o simple_looping -ldl -lm -lpthread
```

```
./simple_looping POC2
```

Download link to POC2

OUTPUT -

```
../miniaudio.h:52991:73: runtime error: left shift of 128 by 24 places cannot be represented in type 'int'
```

Request team to implement proper patch and validate

mackron added a commit that referenced this issue on Jun 11, 2021

Update dr_wav. ··· 73e1589

mackron commented on Jun 11, 2021 Owner

Thank for the report. This is coming from the dr_wav project in dr_libs. I've gone ahead and pushed an update to the dev branch and it'll be released soon. Feel free to reopen this issue if the issue still hasn't been fixed.

mackron closed this as completed on Jun 11, 2021

mackron added the bug label on Jun 11, 2021

Assignees

No one assigned

Labels

bug

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants