GunP4ng / **README.md**

Created 3 days ago

Code Revisions 1

CVE-2025-70072

README.md

CVE-2025-70072

> Public reference for CVE-2025-70072 (Issue: out-of-bounds read in Assimp FBX multi-material mesh conversion).

Product / Affected Versions

Vendor: Assimp project Product: Assimp Open Asset Import Library Affected version(s): Assimp 6.0.2 as tested/reported Fixed version: TBD/Unknown Component(s): FBX importer, `FBXConverter::ConvertMeshMultiMaterial`, `code/AssetLib/FBX/FBXConverter.cpp`

Summary

Assimp's FBX importer converts multi-material meshes by iterating material indices and face counts together. In the reported vulnerable path, the code assumes that the material-index array and the face-count array have the same length. If a crafted FBX file provides mismatched arrays, the face-count iterator can advance past the end of the `faces` array while the material-index loop continues.

This can cause an out-of-bounds read and crash the importing process, resulting in denial of service.

Vulnerability Type / CWE

Type: Out-of-bounds Read CWE: CWE-125 (Out-of-bounds Read)

Impact

Denial of Service: Yes (crafted FBX input can crash an application that imports it through Assimp.) Attack requirements: The target application must import an attacker-supplied FBX file using Assimp.

High-level Verification / Reproduction (no weaponized details)

High-level reproduction conditions:

1. Build Assimp 6.0.2 with the FBX importer enabled.
2. Import a crafted FBX file whose material index count and face index count differ.
3. Reach `FBXConverter::ConvertMeshMultiMaterial`.
4. Observe out-of-bounds read behavior when the code iterates `mindices` and `faces` in lockstep.

The current local `crashes/final` corpus in this workspace was rematched and does not contain the reproducer for this CVE; those local files reproduce CVE-2025-70070 instead. They should not be cited as evidence for CVE-2025-70072.

Technical Notes / Root Cause (for triage)

Location:

- File: `code/AssetLib/FBX/FBXConverter.cpp`
- Function: `Assimp::FBX::FBXConverter::ConvertMeshMultiMaterial`

Relevant vulnerable pattern:

```
itf = faces.begin();
for (MatIndexArray::const_iterator it = mindices.begin(), end = mindices.end(
    const unsigned int pcount = *itf;
```

The loop advances `itf` with the material-index iterator without first validating that `mindices.size()` and `faces.size()` are equal. A malformed FBX file can therefore make `itf` walk past `faces.end()`.

Mitigation / Fix Guidance

- Validate that the material-index and face-count arrays have compatible lengths before lockstep iteration.
- Reject malformed meshes with mismatched material and face metadata.
- Add regression coverage for mismatched material index count versus face count.

Upstream Fix / References for Triage

Upstream fix commit: TBD/Unknown Upstream pull request: TBD/Unknown Release note status: TBD/Unknown

Credits

Discovered by: TaeYong LEE (GunP4ng)

References

- Assimp project: https://github.com/assimp/assimp