# Out of bounds access in TFLite implementation of segment sum

High **mihaimaruseac** published **GHSA-hx2x-85gr-wrpq** on Sep 24, 2020

Package

tensorflow-lite (tensorflow)

| Affected versions | Patched versions |
|---|---|
| 2.2.0, 2.3.0 | 2.2.1, 2.3.1 |

Description

### Impact

In TensorFlow Lite models using segment sum can trigger writes outside of bounds of heap allocated buffers by inserting negative elements in the segment ids tensor:

tensorflow/tensorflow/lite/kernels/internal/reference/reference_ops.h
Lines 2625 to 2631 in 0e68f4d

```
2625      memset(output_data, 0, sizeof(T) * output_shape.FlatSize());
2626
2627      for (int i = 0; i < input_shape.Dims(0); i++) {
2628        int output_index = segment_ids_data[i];
2629        for (int j = 0; j < segment_flat_size; ++j) {
2630          output_data[output_index * segment_flat_size + j] +=
2631              input_data[i * segment_flat_size + j];
```

Users having access to `segment_ids_data` can alter `output_index` and then write to outside of `output_data` buffer.

This might result in a segmentation fault but it can also be used to further corrupt the memory and can be chained with other vulnerabilities to create more advanced exploits.

### Patches

We have patched the issue in `204945b` and will release patch releases for all affected versions.

We recommend users to upgrade to TensorFlow 2.2.1, or 2.3.1.

### Workarounds

A potential workaround would be to add a custom `Verifier` to the model loading code to ensure that the segment ids are all positive, although this only handles the case when the segment ids are stored statically in the model.

A similar validation could be done if the segment ids are generated at runtime between inference steps.

If the segment ids are generated as outputs of a tensor during inference steps, then there are no possible workaround and users are advised to upgrade to patched code.

### For more information

Please consult our security guide for more information regarding the security model and how to contact us with issues and questions.

### Attribution

This vulnerability has been discovered from a variant analysis of GHSA-p2cq-cprg-frvm.

Severity

High

CVE ID

CVE-2020-15212

Weaknesses

No CWEs