Missing validation results in undefined behavior in `QuantizedConv2D`

Low **mihaimaruseac** published **GHSA-pqhm-4wvf-2jg8** on May 17

Package

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

Affected versions	Patched versions
< 2.9.0	2.6.4, 2.7.2, 2.8.1, 2.9.0

Description

Impact

The implementation of `tf.raw_ops.QuantizedConv2D` does not fully validate the input arguments:

```
import tensorflow as tf

input = tf.constant(1, shape=[1, 2, 3, 3], dtype=tf.quint8)
filter = tf.constant(1, shape=[1, 2, 3, 3], dtype=tf.quint8)

# bad args
min_input = tf.constant([], shape=[0], dtype=tf.float32)
max_input = tf.constant(0, shape=[], dtype=tf.float32)
min_filter = tf.constant(0, shape=[], dtype=tf.float32)
max_filter = tf.constant(0, shape=[], dtype=tf.float32)

tf.raw_ops.QuantizedConv2D(
  input=input,
  filter=filter,
  min_input=min_input,
  max_input=max_input,
  min_filter=min_filter,
  max_filter=max_filter,
  strides=[1, 1, 1, 1],
  padding="SAME")
```

In this case, references get bound to `nullptr` for each argument that is empty (in the example, all arguments in the `bad args` section).

Patches

We have patched the issue in GitHub commit 0f0b080ecde4d3dfec158d6f60da34d5e31693c4.

The fix will be included in TensorFlow 2.9.0. We will also cherrypick this commit on TensorFlow 2.8.1, TensorFlow 2.7.2, and TensorFlow 2.6.4, 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 Neophytos Christou from Secure Systems Lab at Brown University.

Severity

Low

CVE ID

CVE-2022-29201

Weaknesses

No CWEs