`CHECK`-fail in `tf.raw_ops.EncodePng`

Low mihaimaruseac published GHSA-3qxp-qjq7-w4hf 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 trigger a `CHECK` fail in PNG encoding by providing an empty input tensor as the pixel data:

```
import tensorflow as tf

image = tf.zeros([0, 0, 3])
image = tf.cast(image, dtype=tf.uint8)
tf.raw_ops.EncodePng(image=image)
```

This is because the implementation only validates that the total number of pixels in the image does not overflow. Thus, an attacker can send an empty matrix for encoding. However, if the tensor is empty, then the associated buffer is `nullptr` . Hence, when calling `png::WriteImageToBuffer` , the first argument (i.e., `image.flat<T>().data()`) is `NULL` . This then triggers the `CHECK_NOTNULL` in the first line of `png::WriteImageToBuffer` .

```
template <typename T>
bool WriteImageToBuffer(
    const void* image, int width, int height, int row_bytes, int num_channels,
    int channel_bits, int compression, T* png_string,
    const std::vector<std::pair<std::string, std::string> >* metadata) {
  CHECK_NOTNULL(image);
  ...
}
```

Since `image` is null, this results in `abort` being called after printing the stacktrace. Effectively, this allows an attacker to mount a denial of service attack.

Patches

We have patched the issue in GitHub commit 26eb323554ffccd173e8a79a8c05c15b685ae4d1.

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 Yakun Zhang and Ying Wang of Baidu X-Team.

Severity

Low

CVE ID

CVE-2021-29531

Weaknesses

No CWEs