Heap buffer overflow in `BandedTriangularSolve`

Low **mihaimaruseac** published **GHSA-2xgj-xhgf-ggjv** on May 12, 2021

Package

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

Affected versions

< 2.5.0

Patched versions

2.1.4, 2.2.3, 2.3.3, 2.4.2

Description

Impact

An attacker can trigger a heap buffer overflow in Eigen implementation of `tf.raw_ops.BandedTriangularSolve`:

```
import tensorflow as tf
import numpy as np

matrix_array = np.array([])
matrix_tensor = tf.convert_to_tensor(np.reshape(matrix_array,(0,1)),dtype=tf.float32)
rhs_array = np.array([1,1])
rhs_tensor = tf.convert_to_tensor(np.reshape(rhs_array,(1,2)),dtype=tf.float32)
tf.raw_ops.BandedTriangularSolve(matrix=matrix_tensor,rhs=rhs_tensor)
```

The implementation calls `ValidateInputTensors` for input validation but fails to validate that the two tensors are not empty:

```
void ValidateInputTensors(OpKernelContext* ctx, const Tensor& in0, const Tensor& in1) {
  OP_REQUIRES(
      ctx, in0.dims() >= 2,
      errors::InvalidArgument("In[0] ndims must be >= 2: ", in0.dims()));

  OP_REQUIRES(
      ctx, in1.dims() >= 2,
      errors::InvalidArgument("In[1] ndims must be >= 2: ", in1.dims()));
}
```

Furthermore, since `OP_REQUIRES` macro only stops execution of current function after setting `ctx->status()` to a non-OK value, callers of helper functions that use `OP_REQUIRES` must check value of `ctx->status()` before continuing. This doesn't happen in this op's implementation, hence the validation that is present is also not effective.

Patches

We have patched the issue in GitHub commit ba6822bd7b7324ba201a28b2f278c29a98edbef2 followed by GitHub commit 0ab290774f91a23bebe30a358fde4e53ab4876a0.

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 Ye Zhang and Yakun Zhang of Baidu X-Team.

Severity

Low

CVE ID

CVE-2021-29612

Weaknesses

No CWEs