tf.string_to_hash_bucket_strong

```
string_to_hash_bucket_strong(
    input,
    num_buckets,
    key,
    name=None
)
```

Defined in `tensorflow/python/ops/gen_string_ops.py`.

See the guide: Strings > Hashing

Converts each string in the input Tensor to its hash mod by a number of buckets.

The hash function is deterministic on the content of the string within the process. The hash function is a keyed hash function, where attribute `key` defines the key of the hash function. `key` is an array of 2 elements.

A strong hash is important when inputs may be malicious, e.g. URLs with additional components. Adversaries could try to make their inputs hash to the same bucket for a denial-of-service attack or to skew the results. A strong hash prevents this by making it difficult, if not infeasible, to compute inputs that hash to the same bucket. This comes at a cost of roughly 4x higher compute time than `tf.string_to_hash_bucket_fast`.

Args:

- `input`: A `Tensor` of type `string`. The strings to assign a hash bucket.
- `num_buckets`: An `int` that is `>= 1`. The number of buckets.
- `key`: A list of `ints`. The key for the keyed hash function passed as a list of two uint64 elements.
- `name`: A name for the operation (optional).

Returns:

A `Tensor` of type `int64`. A Tensor of the same shape as the input `string_tensor`.

Last updated November 2, 2017.