A namespace is stuck in the *Terminating* state

The next steps describe how to manually delete your namespace that is stuck in a terminating state. Before you delete your namespace, troubleshoot to find out the root cause of stuck namespace. See the *namespace that is stuck* section on the Known issues page.

Resolving the problem

1. Run the following command to view the namespaces that are stuck in the *Terminating* state:

   ```
   kubectl get namespaces
   ```

2. Select a terminating namespace and view the contents of the namespace to find out the `finalizer`. Run the following command:

   ```
   kubectl get namespace <terminating-namespace> -o yaml
   ```

   Your YAML contents might resemble the following output:

   ```
   apiVersion: v1
   kind: Namespace
   metadata:
     creationTimestamp: 2018-11-19T18:48:30Z
     deletionTimestamp: 2018-11-19T18:59:36Z
     name: <terminating-namespace>
     resourceVersion: "1385077"
     selfLink: /api/v1/namespaces/<terminating-namespace>
     uid: b50c9ea4-ec2b-11e8-a0be-fa163eeb47a5
   spec:
     finalizers:
     - kubernetes
   ```

```
status:
  phase: Terminating
```

3. Run the following command to create a temporary JSON file:

```
kubectl get namespace <terminating-namespace> -o json >tmp.json
```

4. Edit your `tmp.json` file. Remove the `kubernetes` value from the `finalizers` field and save the file.

   Your `tmp.json` file might resemble the following output:

```
{
    "apiVersion": "v1",
    "kind": "Namespace",
    "metadata": {
        "creationTimestamp": "2018-11-19T18:48:30Z",
        "deletionTimestamp": "2018-11-19T18:59:36Z",
        "name": "<terminating-namespace>",
        "resourceVersion": "1385077",
        "selfLink": "/api/v1/namespaces/<terminating-namespac
        "uid": "b50c9ea4-ec2b-11e8-a0be-fa163eeb47a5"
    },
    "spec": {
       "finalizers":
    },
    "status": {
        "phase": "Terminating"
    }
}
```

5. To set a temporary proxy IP and port, run the following command. Be sure to keep your terminal window open until you delete the stuck namespace:

```
kubectl proxy
```

   Your proxy IP and port might resemble the following output:

```
Starting to serve on 127.0.0.1:8001
```

6. From a new terminal window, make an API call with your temporary proxy IP and port:

```
curl -k -H "Content-Type: application/json" -X PUT --data-binary
@tmp.json http://127.0.0.1:8001/api/v1/namespaces/<terminating-
namespace>/finalize
```

Your output might resemble the following content:

```
{
  "kind": "Namespace",
  "apiVersion": "v1",
  "metadata": {
    "name": "<terminating-namespace>",
    "selfLink": "/api/v1/namespaces/<terminating-namespace>/fi
    "uid": "b50c9ea4-ec2b-11e8-a0be-fa163eeb47a5",
    "resourceVersion": "1602981",
    "creationTimestamp": "2018-11-19T18:48:30Z",
    "deletionTimestamp": "2018-11-19T18:59:36Z"
  },
  "spec": {

  },
  "status": {
    "phase": "Terminating"
  }
}
```

Note: The `finalizer` parameter is removed.

7. Verify that the terminating namespace is removed, run the following command:

```
kubectl get namespaces
```

Continue to follow the steps for other namespaces that are stuck in the *Terminating* state.