In order to generate the certificates needed by Kubernetes, you must first provision a certificate authority. This lesson will guide you through the process of provisioning a new certificate authority for your Kubernetes cluster. After completing this lesson, you should have a certificate authority, which consists of two files: `ca-key.pem` and `ca.pem`.

Here are the commands used in the demo:

```
cd ~/
mkdir kthw
cd kthw/
```

UPDATE: cfssljson and cfssl will need to be installed. To install, complete the following commands:

```
sudo curl -s -L -o /bin/cfssl https://pkg.cfssl.org/R1.2/cfssl_linux-amd64
sudo curl -s -L -o /bin/cfssljson https://pkg.cfssl.org/R1.2/cfssljson_linux-amd64
sudo curl -s -L -o /bin/cfssl-certinfo https://pkg.cfssl.org/R1.2/cfssl-certinfo_linux-amd64
sudo chmod +x /bin/cfssl*
```

Use this command to generate the certificate authority. Include the opening and closing curly braces to run this entire block as a single command.

```
{

cat > ca-config.json << EOF
{
  "signing": {
    "default": {
      "expiry": "8760h"
    },
    "profiles": {
      "kubernetes": {
        "usages": ["signing", "key encipherment", "server auth", "client auth"],
        "expiry": "8760h"
      }
    }
  }
}
EOF

cat > ca-csr.json << EOF
{
  "CN": "Kubernetes",
  "key": {
    "algo": "rsa",
    "size": 2048
  },
  "names": [
    {
      "C": "US",
      "L": "Portland",
      "O": "Kubernetes",
      "OU": "CA",
      "ST": "Oregon"
    }
  ]
}
EOF

cfssl gencert -initca ca-csr.json | cfssljson -bare ca

}
```