kubectl and the Rancher CLI

Lab 21

What are you Learning?

In this lab you'll learn about different ways to interact with Kubernetes clusters running in Rancher. First, you'll connect to them with Kubectl in the Kubernetes UI. Next, you'll use kubectl with a kubeconfig file, provided by Rancher. Lastly, you'll interface with Kubernetes via the Rancher CLI, with an API Token, provided by Rancher.

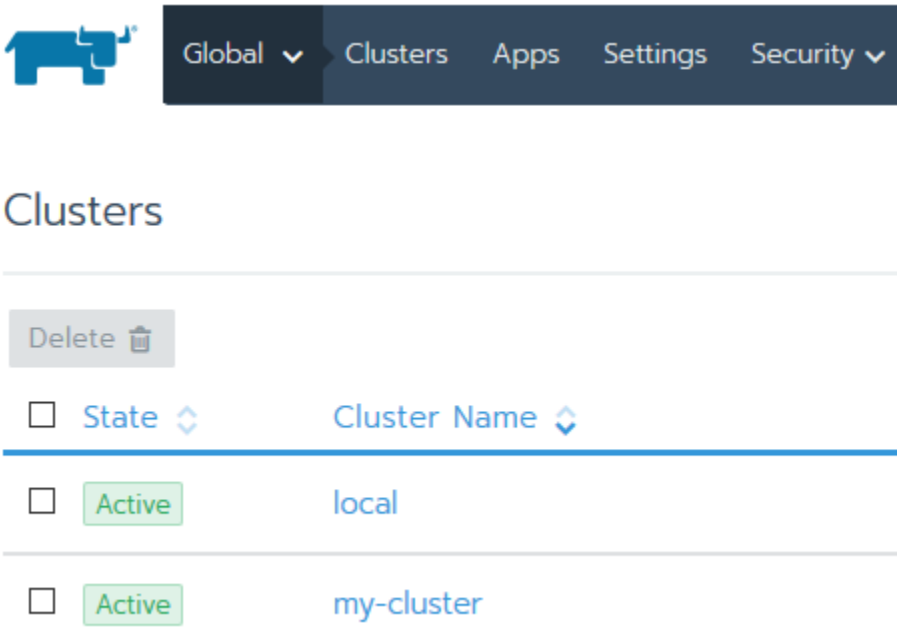
Why is it important?

The Rancher CLI can interact with Kubernetes via kubectl directly, but it also has many of the features provided by the Rancher UI. This is very useful for automating workflows via scripts, or quick access to your clusters without having to log into Rancher via a web browser.

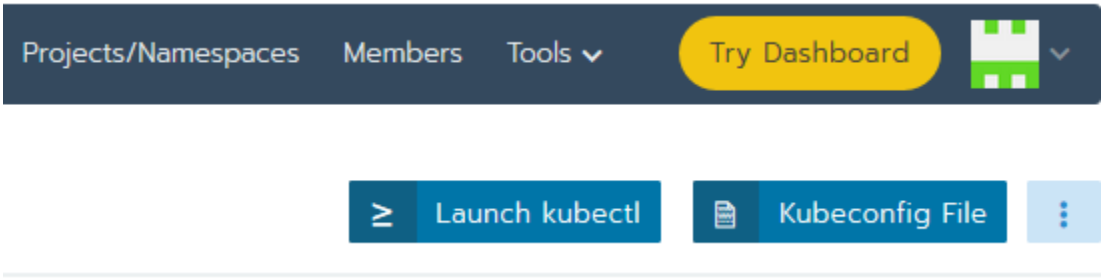
Using kubectl

From the UI

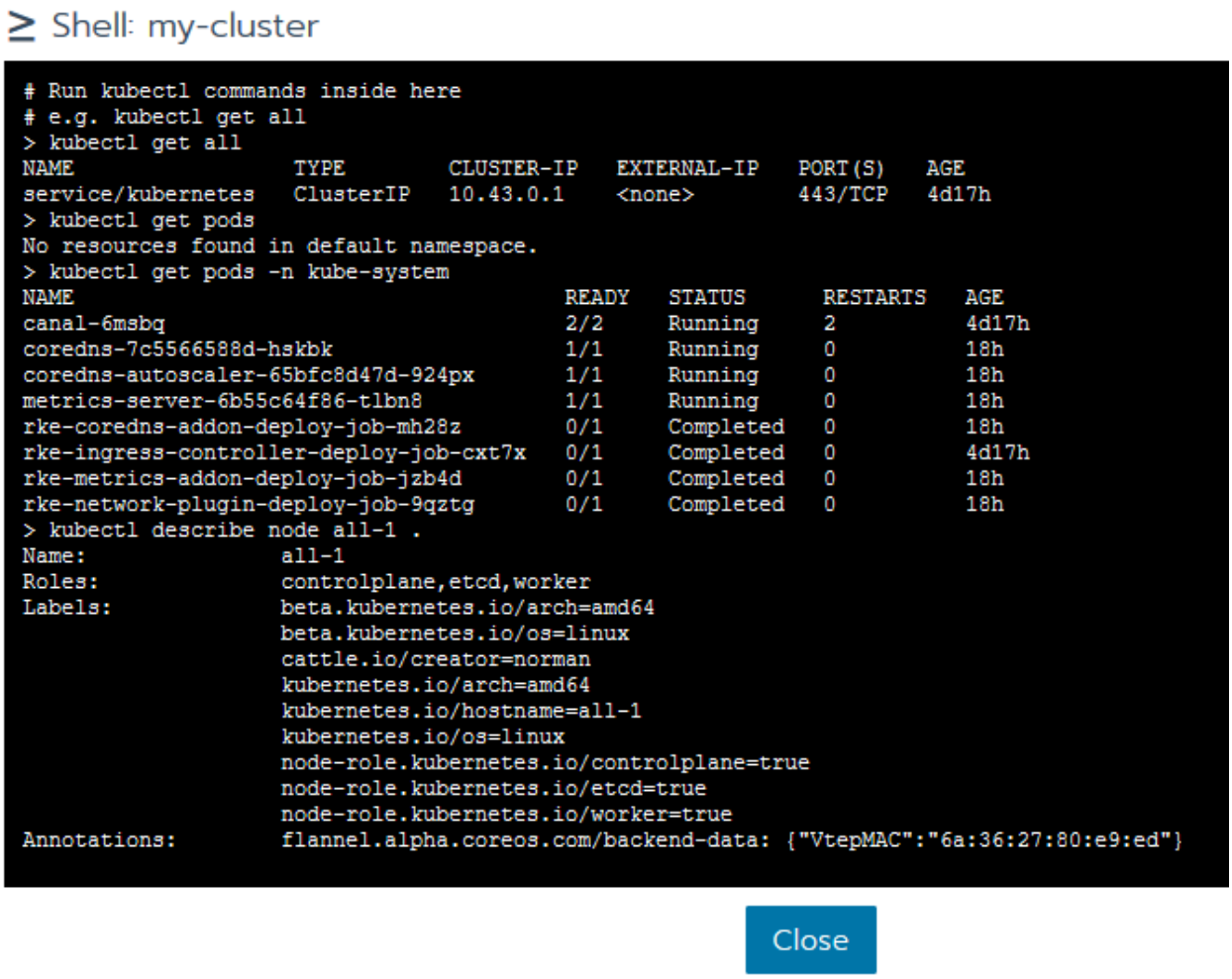
1. Click on the cluster with which you wish to interact.



2. Click on Launch Kubectl



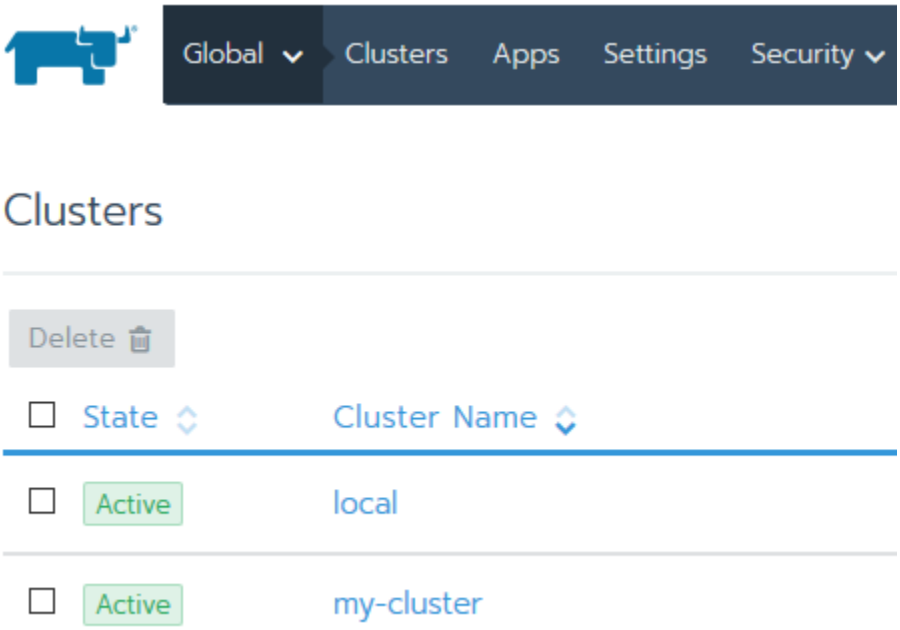
3. Try out a few commands, command-line completion has been activated as well. The click Close



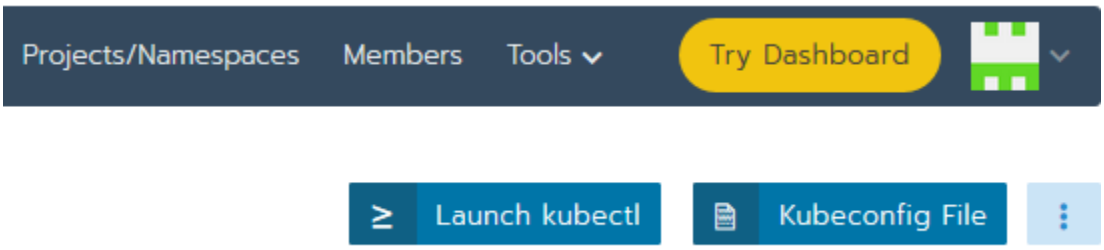
From kubectl

1. If you haven't already install kubectl

2. Click on the cluster with which you wish to interact.



3. Click on Kubeconfig File



4. Either copy this file to ~/.kube/config or setup your environment variables to use multiple configuration files.
5. At this point you should be able to run similar commands from the CLI.

```
$ kubectl get all --all-namespaces
NAMESPACE       NAME                                                READY
STATUS      RESTARTS   AGE
cattle-system   pod/cattle-cluster-agent-86796fdbd8-jslwj     1/1
Running     0          12d
cattle-system   pod/cattle-node-agent-929gv                   1/1
Running     0          12d
cattle-system   pod/rancher-fb7c56d9d-f4g8k                   1/1
Running     1          12d
cattle-system   pod/rancher-fb7c56d9d-fv4tc                   1/1
Running     2          12d
cattle-system   pod/rancher-fb7c56d9d-jqsxs                   1/1
Running     1          12d
```

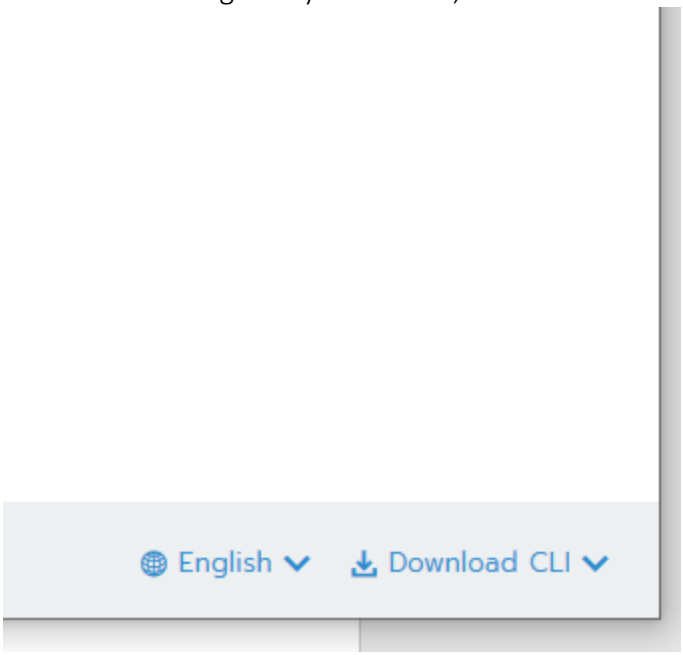
```
cert-manager    pod/cert-manager-5d9cd85cbb-k6q9z             1/1
Running     1          13d
cert-manager    pod/cert-manager-cainjector-95c885477-w7x45   1/1
Running     11         13d
cert-manager    pod/cert-manager-webhook-6ff9487489-q56w2     1/1
Running     4          13d
ingress-nginx   pod/default-http-backend-67cf578fc4-hw7mc      1/1
Running     0          13d
ingress-nginx   pod/nginx-ingress-controller-zrvj6            1/1
Running     3          12d
kube-system     pod/canal-vrvbt                               2/2
Running     0          12d
kube-system     pod/coredns-7c5566588d-gv6jk                  1/1
Running     0          12d
kube-system     pod/coredns-autoscaler-65bfc8d47d-g2bdr       1/1
Running     0          12d
kube-system     pod/metrics-server-6b55c64f86-r47vg           1/1
Running     0          12d
kube-system     pod/rke-coredns-addon-deploy-job-2vvtr        0/1
Completed   0          12d
kube-system     pod/rke-ingress-controller-deploy-job-qb6h7   0/1
Completed   0          12d
kube-system     pod/rke-metrics-addon-deploy-job-zdf46        0/1
Completed   0          12d
kube-system     pod/rke-network-plugin-deploy-job-pljts       0/1
Completed   0          12d

NAMESPACE       NAME                            TYPE        CLUSTER-
IP      EXTERNAL-IP   PORT(S)                   AGE
cattle-system   service/rancher                 ClusterIP
10.43.136.179   <none>        80/TCP                    12d
cert-manager    service/cert-manager            ClusterIP
10.43.93.81     <none>        9402/TCP                  13d
cert-manager    service/cert-manager-webhook    ClusterIP
10.43.20.62     <none>        443/TCP                   13d
default         service/kubernetes              ClusterIP   10.43.0.1
<none>        443/TCP                   13d
ingress-nginx   service/default-http-backend    ClusterIP
10.43.186.16    <none>        80/TCP                    13d
kube-system     service/kube-dns                ClusterIP
10.43.0.10      <none>        53/UDP,53/TCP,9153/TCP    13d
kube-system     service/metrics-server          ClusterIP
10.43.2.20      <none>        443/TCP                   13d

NAMESPACE       NAME                                      DESIRED
CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELE
```

```
CTOR            AGE
cattle-system   daemonset.apps/cattle-node-agent          1
1         1       1            1           <none>
                12d
ingress-nginx   daemonset.apps/nginx-ingress-controller   1
1         1       1            1           <none>
                13d
kube-system     daemonset.apps/canal                     1
1         1       1            1           kubernete
s.io/os=linux   13d

NAMESPACE       NAME                                     READY
UP-TO-DATE   AVAILABLE   AGE
cattle-system   deployment.apps/cattle-cluster-agent     1/1     1
1           12d
cattle-system   deployment.apps/rancher                  3/3     3
3           12d
cert-manager    deployment.apps/cert-manager             1/1     1
1           13d
cert-manager    deployment.apps/cert-manager-cainjector  1/1     1
1           13d
cert-manager    deployment.apps/cert-manager-webhook     1/1     1
1           13d
ingress-nginx   deployment.apps/default-http-backend     1/1     1
1           13d
kube-system     deployment.apps/coredns                  1/1     1
1           13d
kube-system     deployment.apps/coredns-autoscaler       1/1     1
1           13d
kube-system     deployment.apps/metrics-server           1/1     1
1           13d

NAMESPACE       NAME
DESIRED   CURRENT   READY   AGE
cattle-system   replicaset.apps/cattle-cluster-agent-86796fdbd8
1         1         1       12d
cattle-system   replicaset.apps/rancher-fb7c56d9d
3         3         3       12d
cert-manager    replicaset.apps/cert-manager-5d9cd85cbb
1         1         1       13d
cert-manager    replicaset.apps/cert-manager-cainjector-95c885477
1         1         1       13d
cert-manager    replicaset.apps/cert-manager-webhook-6ff9487489
1         1         1       13d
ingress-nginx   replicaset.apps/default-http-backend-67cf578fc4
1         1         1       13d
```

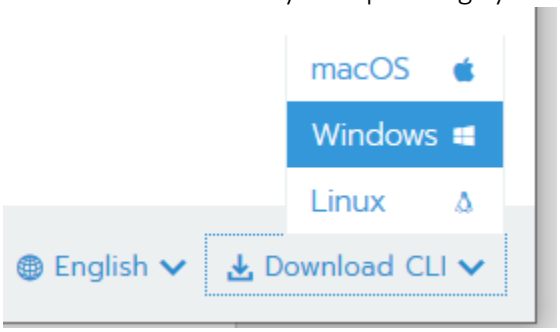
```
kube-system     replicaset.apps/coredns-7c5566588d
1         1         1       13d
kube-system     replicaset.apps/coredns-autoscaler-65bfc8d47d
1         1         1       13d
kube-system     replicaset.apps/metrics-server-6b55c64f86
1         1         1       13d

NAMESPACE     NAME
COMPLETIONS   DURATION   AGE
kube-system   job.batch/rke-coredns-addon-deploy-job        1/1
2s         12d
kube-system   job.batch/rke-ingress-controller-deploy-job   1/1
3s         12d
kube-system   job.batch/rke-metrics-addon-deploy-job        1/1
3s         12d
kube-system   job.batch/rke-network-plugin-deploy-job       1/1
2s         12d
$
```

From the Rancher CLI

1. On the bottom right of your screen, there is the Download CLI option.



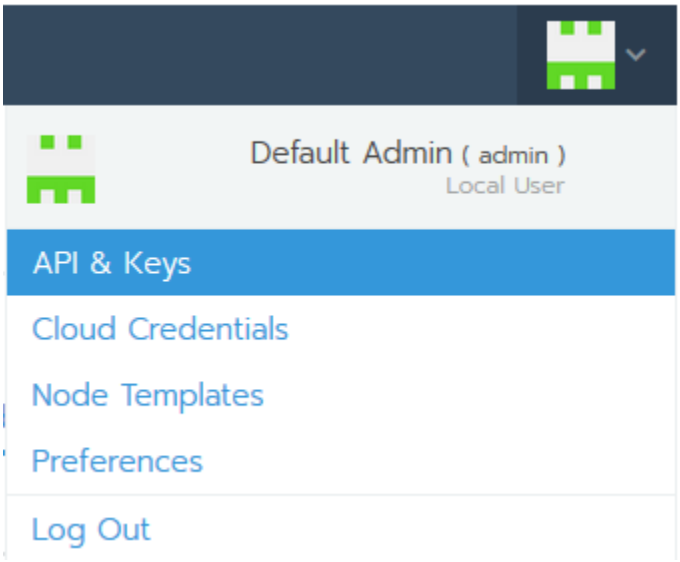
2. Select the version for your operating system.



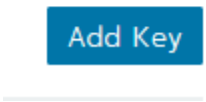
3. Install this and add it to your path. This will vary for your operating system.

```
$ rancher --version
rancher version v2.4.0
```

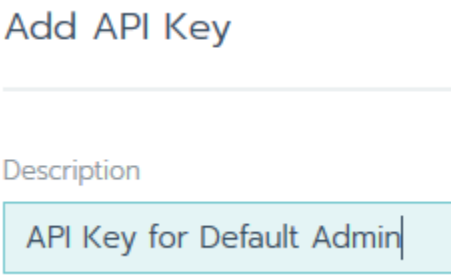
4. Now we'll need an API Token. Click on your avatar icon and select API & Keys



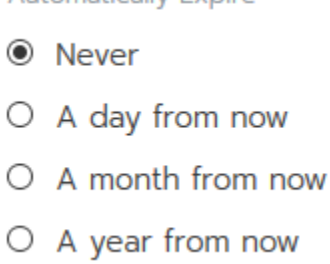
5. Select Add Key



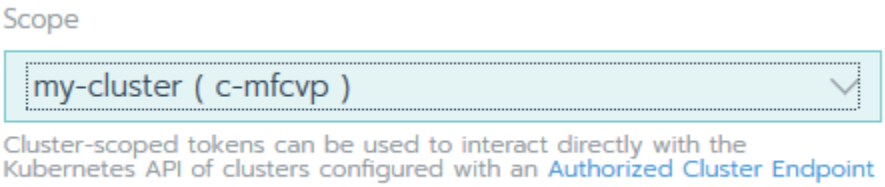
6. Give it a reasonable description



7. Select a reasonable expiration date, consistent with your IT Operations teams' secret rotation policy.



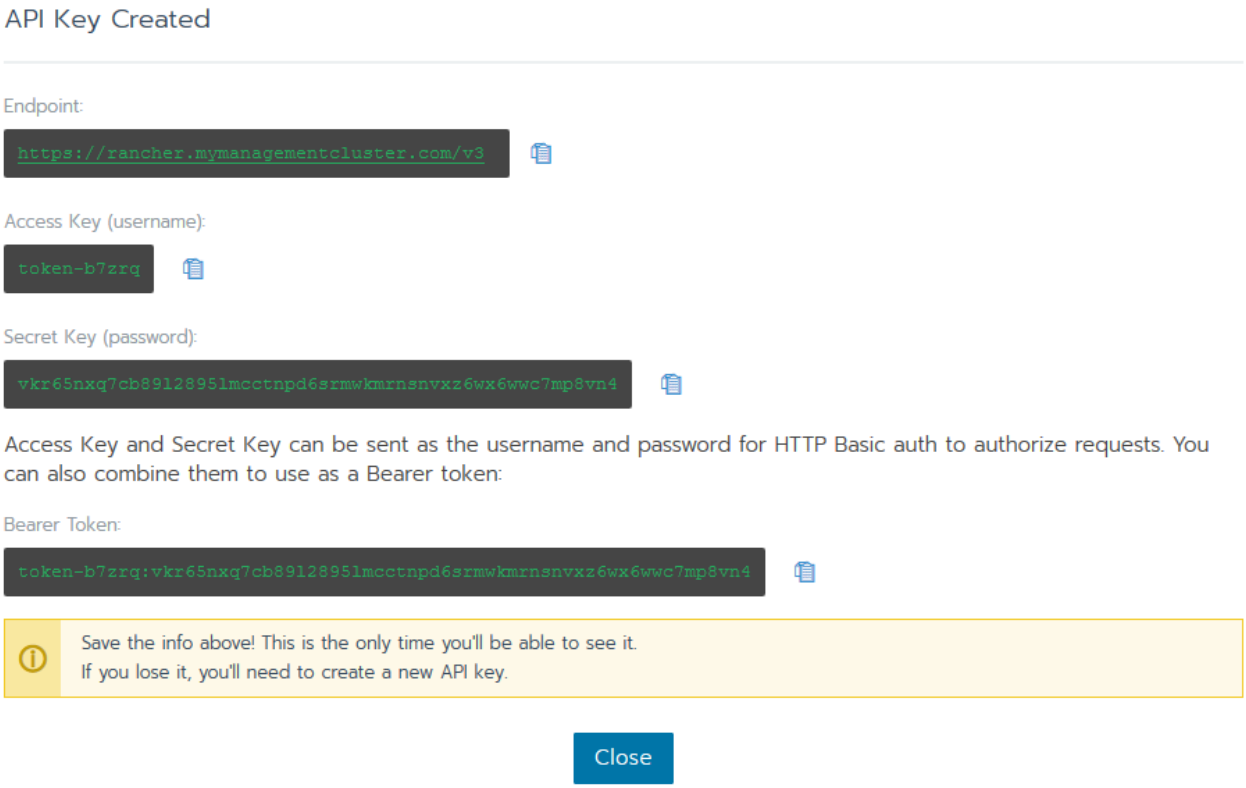
8. Select an appropriate scope for this key. No scope will be able to interact with any cluster where you have access. That being said, a cluster-scoped key reflects least-privilege and can access the Authorized Cluster Endpoint. ***For this lab, use a no scope key.***



9. Click Create

10. Save these items to a file as this is the last time you will see them. Keep them safe, anyone who has access to this token has access to your credentials and RBAC for the

scope of this API token.



11. Click Close, you'll see your API Token on the list

State	Access Key	Description	Scope	Expires	Created
Active	helm-token-user-s6t99	token for helm chart deployment	N/A	Never	12 days ago
Active	kubectl-shell-user-s6t99	Access to kubectl shell in the browser	N/A	Never	14 days ago
Active	telemetry	telemetry token	N/A	Never	17 days ago
Active	token-b7zrq	API Key for Default Admin	Cluster	Never	a few seconds ago

12. From your terminal, we'll use the token to login into Rancher and my-cluster. Notice I added a space at the start of my command so the token is not saved to my terminal's history log. Also, if your cluster is using self-signed certificates, you'll want to use the skip-verify option.

```
$  rancher login --token token-7vvhc:k8l4npmgr97hvjbk6pq9j4h2fkkzf71
65h6zrzcfbrfjl9hrx29nr8 --name rancher-cert
https://rancher.mymanagementcluster.com/v3
NUMBER    CLUSTER NAME   PROJECT ID        PROJECT NAME   PROJECT
DESCRIPTION
1         my-cluster     c-mfcvp:p-pw5sv   System         System
project created for the cluster
2         my-cluster     c-mfcvp:p-v5dgv   Default        Default
project created for the cluster
```

```
3         local          local:p-jcks8     System          System
project created for the cluster
4         local          local:p-nkv4q     Default         Default
project created for the cluster
```

13. You'll notice a list of clusters and projects. A Rancher CLI context is the combination of a cluster and a project.
14. Select the Default Project on my-cluster, or whatever the name is you gave you your cluster.

```
Select a Project:2
INFO[0239] Saving config to /home/jason/.rancher/cli2.json
```

15. Notice your credentials are saved in the cli2.json file. This contains your token keep it safe.

```
$ cat ~/.rancher/cli2.json
{"Servers":{"rancher-cert":{"accessKey":"token-
7vvhc","secretKey":"k8l4npmgr97hvjbk6pq9j4h2fkkzf7l65h6zrzcfbrfjl9hr
x29nr
8","tokenKey":"token-
7vvhc:k8l4npmgr97hvjbk6pq9j4h2fkkzf7l65h6zrzcfbrfjl9hrx29nr8","url":
"https://rancher.mymanagementcl
uster.com","project":"c-mfcvp:p-
v5dgv","cacert":""}},"CurrentServer":"rancher-cert"}
```

16. To run kubectl against this context use rancher kubectl

```
$ rancher kubectl get nodes
NAME    STATUS   ROLES                      AGE     VERSION
all-1   Ready    controlplane,etcd,worker   4d23h   v1.17.5
$ rancher kubectl get all --all-namespaces
NAMESPACE       NAME                                          READY
STATUS      RESTARTS   AGE
cattle-system   pod/cattle-cluster-agent-58d884cfcd-m6rrn     1/1
Running     0          4d23h
cattle-system   pod/cattle-node-agent-2qtjs                   1/1
Running     0          4d23h
cattle-system   pod/kube-api-auth-w9vv6                       1/1
Running     0          4d23h
ingress-nginx   pod/default-http-backend-67cf578fc4-jtr6r     1/1
Running     0          4d23h
ingress-nginx   pod/nginx-ingress-controller-5ttvj            1/1
Running     0          4d23h
kube-system     pod/canal-9j7kd                               2/2
Running     0          4d23h
kube-system     pod/coredns-7c5566588d-rtg2b                  1/1
Running     0          4d23h
kube-system     pod/coredns-autoscaler-65bfc8d47d-2ggf5       1/1
Running     0          4d23h
kube-system     pod/metrics-server-6b55c64f86-fpwhv           1/1
```

```
Running     0          4d23h
kube-system     pod/rke-coredns-addon-deploy-job-cb79j        0/1
Completed   0          4d23h
kube-system     pod/rke-ingress-controller-deploy-job-srbp2   0/1
Completed   0          4d23h
kube-system     pod/rke-metrics-addon-deploy-job-d9zvx        0/1
Completed   0          4d23h
kube-system     pod/rke-network-plugin-deploy-job-662dj       0/1
Completed   0          4d23h

NAMESPACE       NAME                              TYPE        CLUSTER-
IP      EXTERNAL-IP   PORT(S)                   AGE
default         service/kubernetes                ClusterIP   10.43.0.1
<none>        443/TCP                   4d23h
ingress-nginx   service/default-http-backend   ClusterIP
10.43.244.173   <none>        80/TCP                    4d23h
kube-system     service/kube-dns                  ClusterIP
10.43.0.10      <none>        53/UDP,53/TCP,9153/TCP   4d23h
kube-system     service/metrics-server            ClusterIP
10.43.243.14    <none>        443/TCP                   4d23h

NAMESPACE       NAME                                        DESIRED
CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELE
CTOR            AGE
cattle-system   daemonset.apps/cattle-node-agent            1
1         1       1            1           <none>
                4d23h
cattle-system   daemonset.apps/kube-api-auth                1
1         1       1            1           <none>
                4d23h
ingress-nginx   daemonset.apps/nginx-ingress-controller   1
1         1       1            1           <none>
                4d23h
kube-system     daemonset.apps/canal                        1
1         1       1            1           kubernete
s.io/os=linux   4d23h

NAMESPACE       NAME                                    READY   UP-
TO-DATE   AVAILABLE   AGE
cattle-system   deployment.apps/cattle-cluster-agent   1/1     1
1           4d23h
ingress-nginx   deployment.apps/default-http-backend   1/1     1
1           4d23h
kube-system     deployment.apps/coredns                1/1     1
1           4d23h
kube-system     deployment.apps/coredns-autoscaler     1/1     1
```

```
1           4d23h
kube-system     deployment.apps/metrics-server         1/1     1
1           4d23h

NAMESPACE       NAME
DESIRED   CURRENT   READY   AGE
cattle-system   replicaset.apps/cattle-cluster-agent-58d884cfcd   1
1         1       4d23h
ingress-nginx   replicaset.apps/default-http-backend-67cf578fc4   1
1         1       4d23h
kube-system     replicaset.apps/coredns-7c5566588d                1
1         1       4d23h
kube-system     replicaset.apps/coredns-autoscaler-65bfc8d47d     1
1         1       4d23h
kube-system     replicaset.apps/metrics-server-6b55c64f86         1
1         1       4d23h

NAMESPACE     NAME
COMPLETIONS   DURATION   AGE
kube-system   job.batch/rke-coredns-addon-deploy-job        1/1
13s        4d23h
kube-system   job.batch/rke-ingress-controller-deploy-job   1/1
11s        4d23h
kube-system   job.batch/rke-metrics-addon-deploy-job        1/1
12s        4d23h
kube-system   job.batch/rke-network-plugin-deploy-job       1/1
10s        4d23h
```

17. To always use rancher api access you can alias kubectl.

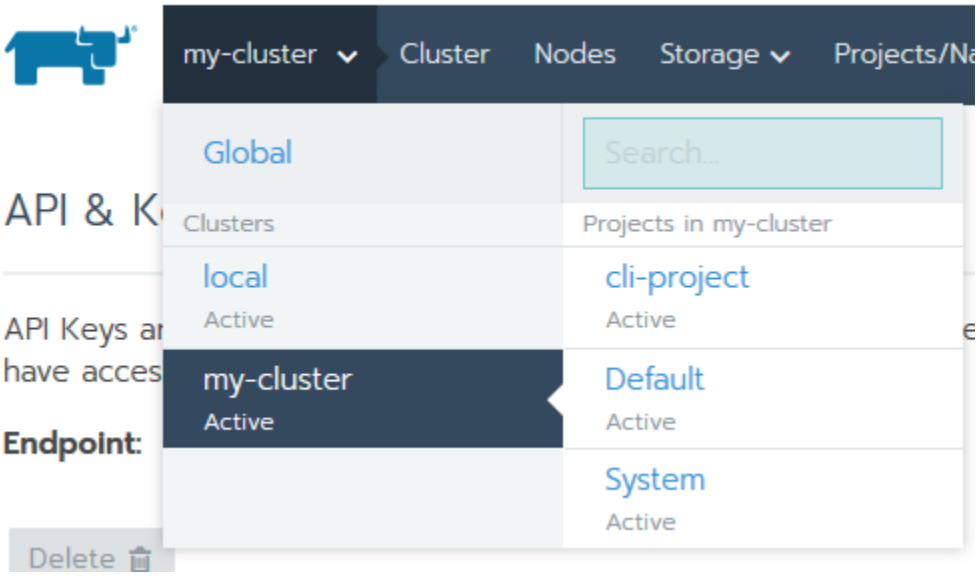
```
$ alias kubectl="rancher kubectl"
$ kubectl get nodes
NAME    STATUS   ROLES                      AGE     VERSION
all-1   Ready    controlplane,etcd,worker   4d23h   v1.17.5
```

Testing That It Works

1. Create a project using the CLI

```
$ rancher project create --description "this project was created vi
a the CLI." cli-project
```

2. Check your web UI to see the new project



This is great! Now your Rancher CLI can modify both Rancher and Kubernetes Clusters. You can learn more about the Rancher CLI commands and their options in the Rancher documentation. Spend some time experimenting.

References

- Install and Setup kubectl - https://kubernetes.io/docs/tasks/tools/install-kubectl/
- Configure Access to Multiple Clusters - https://kubernetes.io/docs/tasks/access-application-cluster/configure-access-multiple-clusters/
- How the Authorized Cluster Endpoint Works - https://rancher.com/docs/rancher/v2.x/en/cluster-admin/cluster-access/ace/
- Project Administration - https://rancher.com/docs/rancher/v2.x/en/project-admin/
- Use the Rancher Command Line Interface - https://rancher.com/docs/rancher/v2.x/en/cli/