After an install, I only have the postgres-0 pod!

Issue

- After installing Element On-Premise, I only have a `postgres-0` in the `element-onprem` namespace:

```
[user@element element-enterprise-installer-1.0.0]$ kubectl get pods -n element-onprem
NAME       READY   STATUS    RESTARTS   AGE
postgres-0   1/1     Running   0          3m33s
```

- Installer hangs while trying to connect to the local microk8s registry.
- `calico-kube-controllers` in the `kube-system` namespace throwing this error:

```
[FATAL][1] main.go 114: Failed to initialize Calico datastore error=Get
https://10.152.183.1:443/apis/crd.projectcalico.org/v1/clusterinformations/default: context deadline
exceeded
```

(N.B. You must include the hash behind calico-kube-controllers to get the logs. So in the event that your pod is named `calico-kube-controllers-f7868dd95-dqd6b` then you would need to run `kubectl logs -n kube-system calico-kube-controllers-f7868dd95-dqd6b` to get the logs.)

Environment

- Element Enterprise Installer 1.0.0
- Red Hat Enterprise Linux 8.5.0

Resolution

- On **Ubuntu**, edit `/etc/modules` and add in there a new line:
  `br_netfilter`
- On **Red Hat Enterprise Linux**, edit `/etc/modules-load.d/snap.microk8s.conf` and add in there a new line:
  `br_netfilter`
- Run:
  `microk8s stop`
- Edit `/var/snap/microk8s/current/args/kube-proxy` and remove the `--proxy-mode` line completely.
- Run: `sudo modprobe br_netfilter`
- Then run: `microk8s start`
- After this, wait a little bit for all of the pods to finish creating and bring the rest of the stack up.

Root Cause

- Looking at all my pods, there are several errors:

```
[user@element element-enterprise-installer-1.0.0]$ kubectl get pods -A
NAMESPACE          NAME                                READY   STATUS             RESTARTS   AGE
kube-system        coredns-7f9c69c78c-9g5xf            0/1     Running            0          8m3s
kube-system        calico-node-l8xmn                   1/1     Running            0          11m
container-registry registry-9b57d9df8-xjcf5            0/1     Pending            0          2m8s
kube-system        coredns-ddd489c4d-bhwq5             0/1     Running            0          2m8s
kube-system        dashboard-metrics-scraper-78d7698477-pcpbg   1/1     Running            0
2m8s
kube-system        hostpath-provisioner-566686b959-bvgr5       1/1     Running            0          2m8s
kube-system        calico-kube-controllers-f7868dd95-dqd6b     0/1     CrashLoopBackOff   10
11m
element-onprem     postgres-0                          1/1     Running            0          2m9s
kube-system        kubernetes-dashboard-85fd7f45cb-m7lkb       1/1     Running            2          2m8s
ingress            nginx-ingress-microk8s-controller-tlrqk     0/1     Running            3          2m9s
operator-onprem    osdk-controller-manager-644775db9d-jzqnb    1/2     Running            2
2m8s
kube-system        metrics-server-8bbfb4bdb-tlnzk      1/1     Running            2          2m8s
```

- Looking at the logs for `calico-kube-controllers` in the `kube-system` namespace:

```
[user@element ~]$ kubectl logs -n kube-system calico-kube-controllers-f7868dd95-swpst
2022-05-09 15:18:10.856 [INFO][1] main.go 88: Loaded configuration from environment
config=&config.Config{LogLevel:"info", ReconcilerPeriod:"5m", CompactionPeriod:"10m",
EnabledControllers:"node", WorkloadEndpointWorkers:1, ProfileWorkers:1, PolicyWorkers:1,
NodeWorkers:1, Kubeconfig:"", HealthEnabled:true, SyncNodeLabels:true,
```

```
DatastoreType:"kubernetes"}
W0509 15:18:10.857670       1 client_config.go:541] Neither --kubeconfig nor --master was specified.
Using the inClusterConfig.  This might not work.
2022-05-09 15:18:10.858 [INFO][1] main.go 109: Ensuring Calico datastore is initialized
2022-05-09 15:18:20.859 [ERROR][1] client.go 255: Error getting cluster information config
ClusterInformation="default" error=Get
https://10.152.183.1:443/apis/crd.projectcalico.org/v1/clusterinformations/default: context deadline
exceeded
2022-05-09 15:18:20.859 [FATAL][1] main.go 114: Failed to initialize Calico datastore error=Get
https://10.152.183.1:443/apis/crd.projectcalico.org/v1/clusterinformations/default: context deadline
exceeded
```

The reason that this is happening is under certain scenarios, microk8s fails to load the `br_netfilter` kernel module and this allows the calico networking to fall back to user space routing, which fails to work in this environment and causes the calico-kube-controllers pod to not start, which cascades into the rest of the stack not really coming up. More on this specific issue can be seen here: https://github.com/canonical/microk8s/issues/3085. The microk8s team does expect to release a fix and we will work to incorporate it in the future.

Revision #8
Created 9 May 2022 15:22:41 by Karl Abbott
Updated 6 November 2024 12:49:21 by Kieran Mitchell Lane