

## Общее

В данном руководстве будет описан процесс переключения CNI драйвера с `flannel` на `cilium` на версии deckhouse CSE v1.64.0

## Переключение CNI

Сначала необходимо включить модуль `cni-cilium`

```
$ kubectl create -f - << EOF
apiVersion: deckhouse.io/v1alpha1
kind: ModuleConfig
metadata:
  name: cni-cilium
spec:
  version: 1
  enabled: true
  settings:
    tunnelMode: VXLAN
EOF
```

Затем проверяем, что все агенты `cilium` в статусе `Running`

```
$ kubectl get po -n d8-cni-cilium
NAME                                READY   STATUS    RESTARTS   AGE
agent-5zzfv                          2/2    Running   5 (23m ago) 26m
agent-gqb2b                           2/2    Running   5 (23m ago) 26m
agent-wtv4p                           2/2    Running   5 (23m ago) 26m
operator-856d69fd49-mlglv            2/2    Running   0           26m
safe-agent-updater-26qpk             3/3    Running   0           26m
safe-agent-updater-qlbrh             3/3    Running   0           26m
safe-agent-updater-wjlr5             3/3    Running   0           26m
```

Если агенты `cilium` не стартуют, то порядок действия для их починки следующий

1. Сохраняем манифест `daemonset` для `kube-proxy` (для возможности восстановления):

```
$ kubectl -n kube-system get ds d8-kube-proxy -oyaml > d8-kube-proxy.yaml
```

2. Выполняем следующие команды:

```
$ kubectl delete validatingadmissionpolicies.admissionregistration.k8s.io label-objects.deckhouse.io
$ kubectl -n kube-system delete ds d8-kube-proxy
$ kubectl -n d8-cni-cilium delete po -l app=agent
```

Затем необходимо дождаться пока все агенты `cilium` перейдут в состояние `Running`

Далее необходимо отключить модуль `cni-flannel`

Выключение `cni-flannel`:

```
$ kubectl apply -f - << EOF
apiVersion: deckhouse.io/v1alpha1
```

```
kind: ModuleConfig
metadata:
  name: cni-flannel
spec:
  enabled: false
EOF
```

После этого необходимо выполнить `reboot` мастер нод, а затем по очереди остальных нод кластера.

*Если агенты `cilium` не переходят в статус `Running`, то нужно выполнить `reboot` проблемных нод*

После ребута проверяем состояние подов `cilium-agent` :

```
$ kubectl get po -n d8-cni-cilium
```

Далее нужно включить модуль `node-local-dns`

Включение `node-local-dns` :

```
$ kubectl apply -f - << EOF
apiVersion: deckhouse.io/v1alpha1
kind: ModuleConfig
metadata:
  name: node-local-dns
spec:
  enabled: true
EOF
```

Чтобы убедиться, что переключение прошло успешно, необходимо:

1. Проверить очередь `deckhouse`  
В случае одной мастер ноды

```
$ kubectl -n d8-system exec -it deploy/deckhouse -c deckhouse -- deckhouse-controller
queue list
```

В случае мультимастер инсталляции

```
$ kubectl -n d8-system exec svc/deckhouse-leader -c deckhouse -- deckhouse-controller
queue list
```

2. Проверить агенты `cilium`. Они должны быть в статусе `Running`

```
$ kubectl get po -n d8-cni-cilium
NAME                READY   STATUS    RESTARTS   AGE
agent-5zzfv         2/2     Running   5 (23m ago) 26m
agent-gqb2b         2/2     Running   5 (23m ago) 26m
agent-wtv4p         2/2     Running   5 (23m ago) 26m
```

3. Проверить, что модуль `cni-flannel` выключен

```
$ kubectl get modules | grep flannel
cni-flannel          35      Disabled  Embedded
```

4. Проверить, что модуль `node-local-dns` включен

```
$ kubectl get modules | grep node-local-dns
node-local-dns          350      Enabled  Embedded  Ready
```

5. Проверить, что модуль kube-proxy выключен

```
$ kubectl get modules | grep kube-proxy
kube-proxy             36       Disabled Embedded
```