⚠ A translation for your language preference does not exist.

How to control Node updates in OCP 4

⊘ **SOLUTION** 已验证 - 已更新 2021年五月13日00:03 - English ▾

环境

- Red Hat OpenShift Container Platform (RHOCP)
  - 4

问题

- Is there a way to control Node updates in OCP 4?
- Could I set the `MaxUnavailable` Node in a OpenShift 4.x to 0 and wait until I have approval and then set this to 1 ? Would that work?
- Is it possible to continue upgrading other nodes in the cluster, if one node cannot be drained?

决议

Red Hat OpenShift Container Platform 4 has functionality (`MaxUnavailable`) to control the amount of Red Hat OpenShift Container Platform - Node(s) that can be taken offline at a given time within a `MachineConfigPool` . By default this is set to 1, meaning only 1 Red Hat OpenShift Container Platform - Node per `MachineConfigPool` can be unavailable. If `MaxUnavailable` is set to 0, no Red Hat OpenShift Container Platform - Node can be taken down, which would prevent any update from happening. To change `MaxUnavailable` in any given `MachineConfigPool` , execute `oc patch` as shown below:

```
oc patch --type merge machineconfigpool/<machineconfigpool> -p '{"spec":
{"MaxUnavailable":"<value>"}}'
```

Since `MaxUnavailable` is primarily meant to specify the percentage or constant number of Red Hat OpenShift Container Platform - Node that can be updating at any given time, there is additional functionality available to prevent a `MachineConfigPool` from being updated. In `MachineConfigPool`, `paused` can be set to either `true` or `false`. If set to `true`, changes to this `MachineConfigPool` should be stopped. This includes generating new `desiredMachineConfig` and update of Red Hat OpenShift Container Platform - Node part of that `MachineConfigPool`. To set `paused` to either `false` or `true` in a specific `MachineConfigPool` use the below procedure:

```
oc patch --type merge machineconfigpool/<machineconfigpool> -p '{"spec":
{"paused":"<value>"}}'
```

Important: if the `spec.paused` field is not restored to `false`, or the `spec.MaxUnavailable` set to a value greater than `0`, the nodes will not be rebooted by configuration changes in the cluster like certificates renewal, and it can cause that the cluster may not work if the old certificates expire. Refer to disable autoreboot when the certificates of kube-apiserver-to-kubelet-signer will be rotated at around the 80 percent of that one year for additional information.

Additionally, there is a RFE to add the ability to continue with nodes upgrade if a node cannot be drained.

根源

There is a need to control Red Hat OpenShift Container Platform - Node update behaviour either by amount of Red Hat OpenShift Container Platform - Node that can be taken offline at the same time in one `MachineConfigPool` respectively related to the timing when the Red Hat OpenShift Container Platform - Node can be updated and hence need to be `paused` until further action is taken.

产品（第）   **Red Hat OpenShift Container Platform**

元件   **kubernetes**   **openshift-node**   **RHCOS**   **类别**   **Upgrade**

标记   **ocp_4**   **openshift**   **shift_upgrade**   **shift_usability**   **update**   **upgrade**

People who viewed this solution also viewed

MachineConfigPool is stuck on updating status
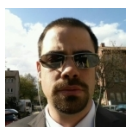
Solution - 2022年12月8日

`invalid value for IntOrString: invalid type:` when updating MachineConfigPool in OpenShift 4

Solution - 2022年5月9日

What is the difference between MachineSets and MachineConfigPool Red Hat OpenShift Container Platform 4

Solution - 2020年3月10日

---

3 评论

25 June 2020 12:28 PM

Oliver Guggenbühl

COMMUNITY
MEMBER

47 Points

We need to have a way to control reboots by nodes. if we starting to bring operator base workload example DBs we shouldn't reboot by accident the wrong nodes. this is not good enough currently. Additional to that not all workload running on ocp will be stateless.

↩ 回复

5 February 2021 3:29 PM

Simon Reber

SR
RED HAT

Oliver Guggenbühl, you can for each machine create a `MachineConfig` pool and therefore better control when, what Node is rebooted. Although we don't recommend doing that. To better control how applications are impacted by OpenShift - Node reboots, we recommend to use PodDisruptionBudget.

↩ 回复

---

23 June 2022 1:08 PM

Christian Benstein

The instructions specifies to patch `MaxUnavailable`. The correct identifier however is: `maxUnavailable` (refering to `machineconfigpool.spec.maxUnavailable`). Running the patch action as described will not patch or change existing machineconfigpool's

↩ 回复