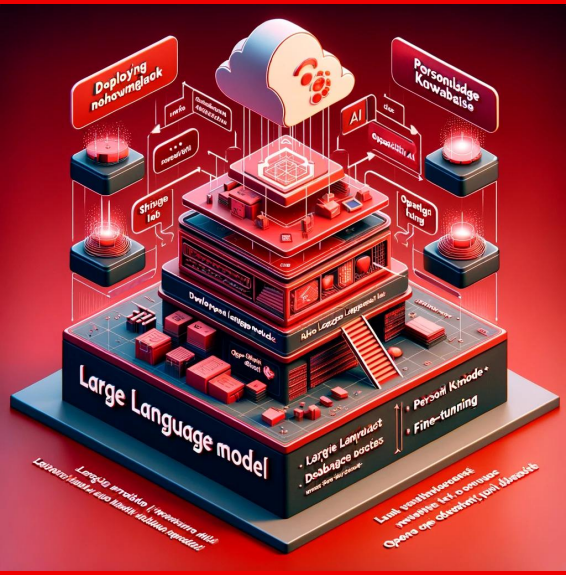
Machine Config Operator Certificate/Key Rotation Logic Comparing

between 4.12 and 4.16

Zheng(George) Wang
zhengwan@redhat.com

agenda



1. the question
2. how things works in 4.12
3. how things works in 4.16
4. test it out
5. what changed?
6. key code logic
7. conclusion

Red Hat

The question: how the cert rotation logic changed between ocp 4.12 and ocp 4.16

when cert rotation on ocp 4.12, the mco logs looks like:

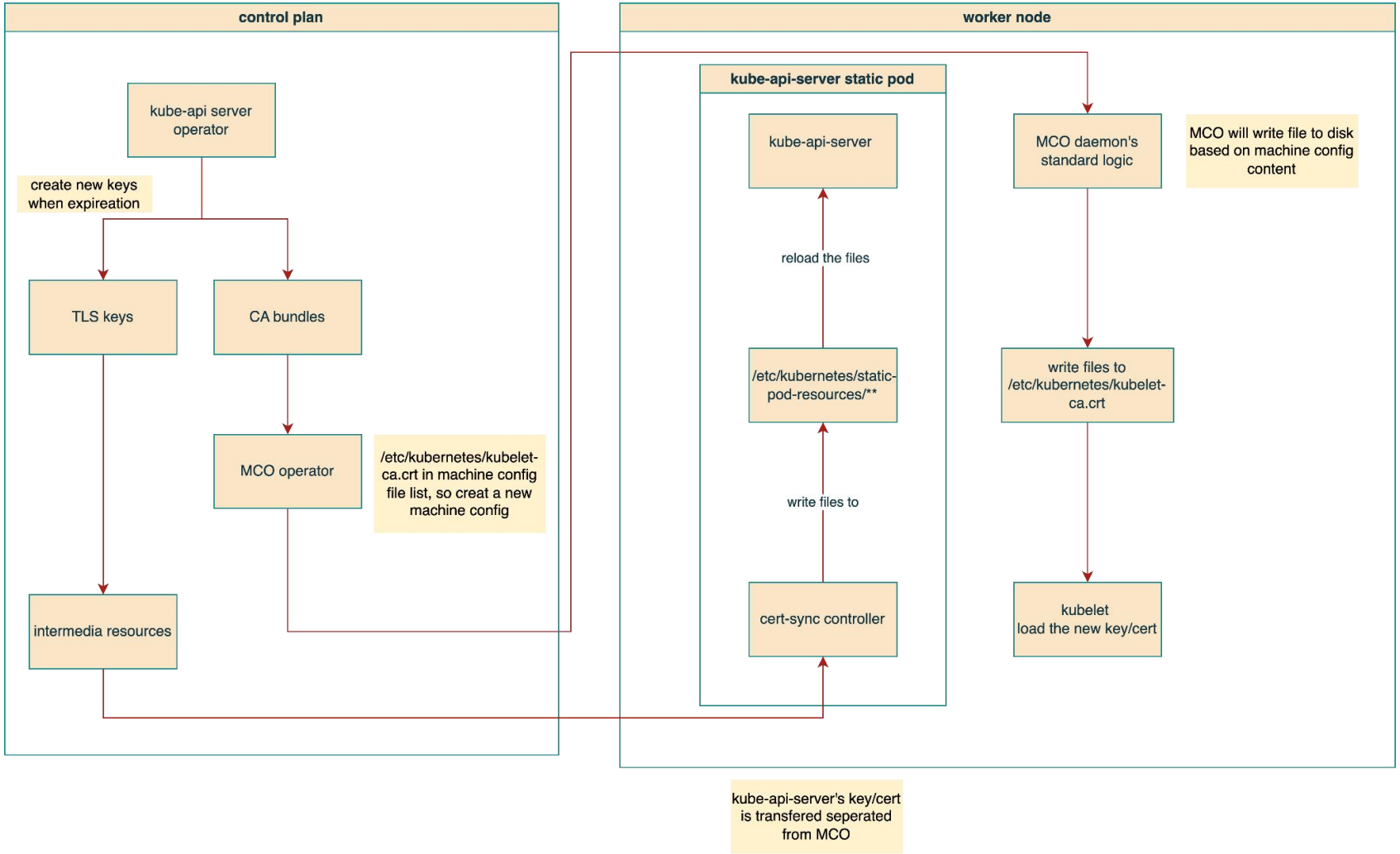
VS

when cert rotation on ocp 4.16, the mco logs looks like:

```
I0410 15:59:14.239434   31662 certificate_writer.go:303] Certificate was synced from controllerconfig resourceVersion 23344
I0410 15:59:15.621037   31662 certificate_writer.go:303] Certificate was synced from controllerconfig resourceVersion 23346
I0410 15:59:15.950768   31662 certificate_writer.go:303] Certificate was synced from controllerconfig resourceVersion 23347
I0410 15:59:17.005671   31662 certificate_writer.go:303] Certificate was synced from controllerconfig resourceVersion 23359
I0410 15:59:24.486742   31662 certificate_writer.go:303] Certificate was synced from controllerconfig resourceVersion 23399
I0410 15:59:24.776106   31662 certificate_writer.go:303] Certificate was synced from controllerconfig resourceVersion 23400
I0410 15:59:25.127322   31662 certificate_writer.go:303] Certificate was synced from controllerconfig resourceVersion 23402
I0410 15:59:25.760395   31662 certificate_writer.go:303] Certificate was synced from controllerconfig resourceVersion 23410
```

How things works for ocp 4.12

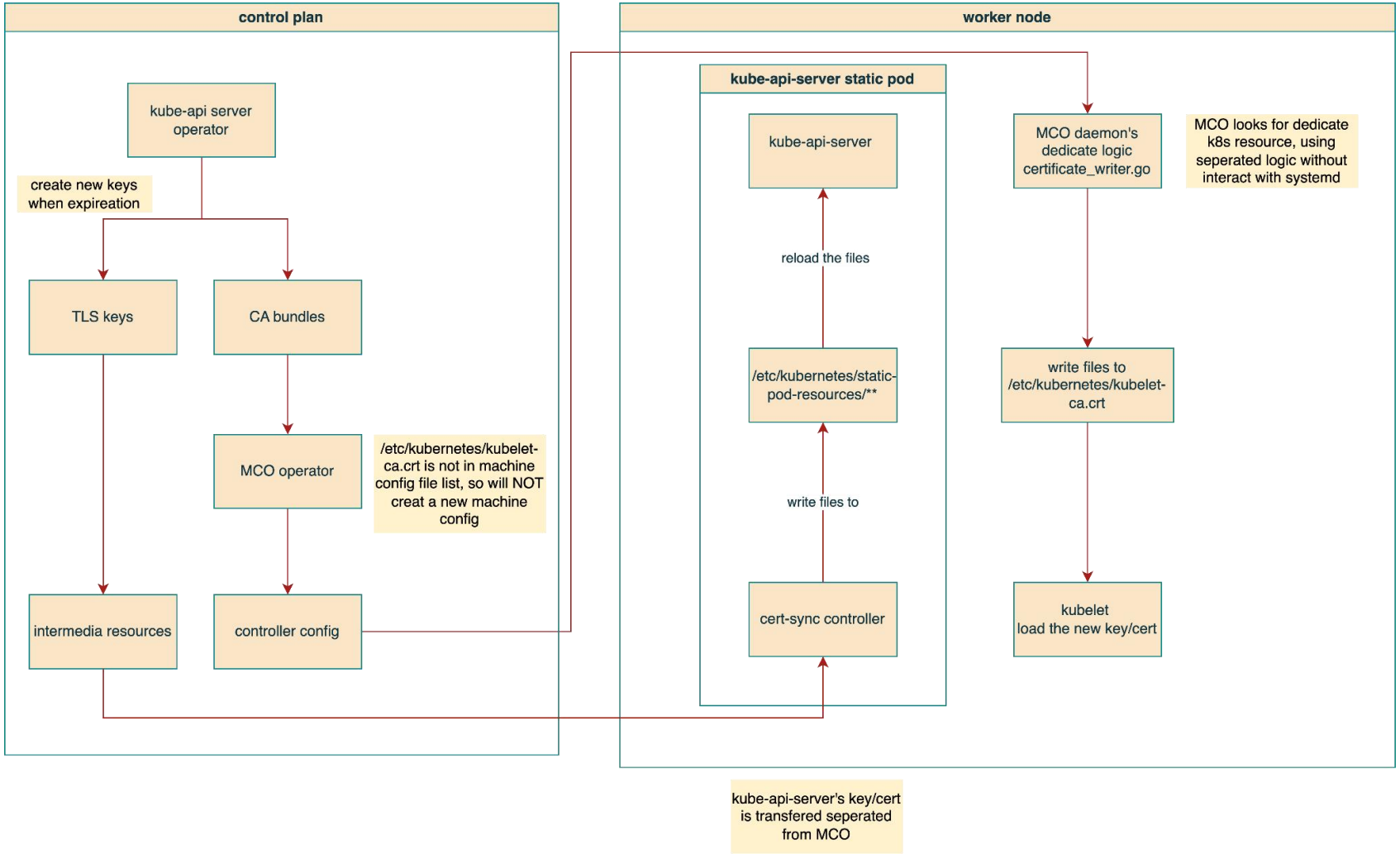
- new keys generate on control plan
- deliver to nodes by using **new machine config version**
- MCO write /etc/kubernetes/kubelet-ca.crt to disk
- kube-api and kubelet re-load the keys directly
- `systemctl enable` called by MCO, this is why you can see 'systemd[1]: Reloading.'



https://github.com/wangzheng422/docker_env/blob/dev/redhat/ocp4/4.12/2025.04.mco.cert.rotation.en.md

How things works for ocp 4.16

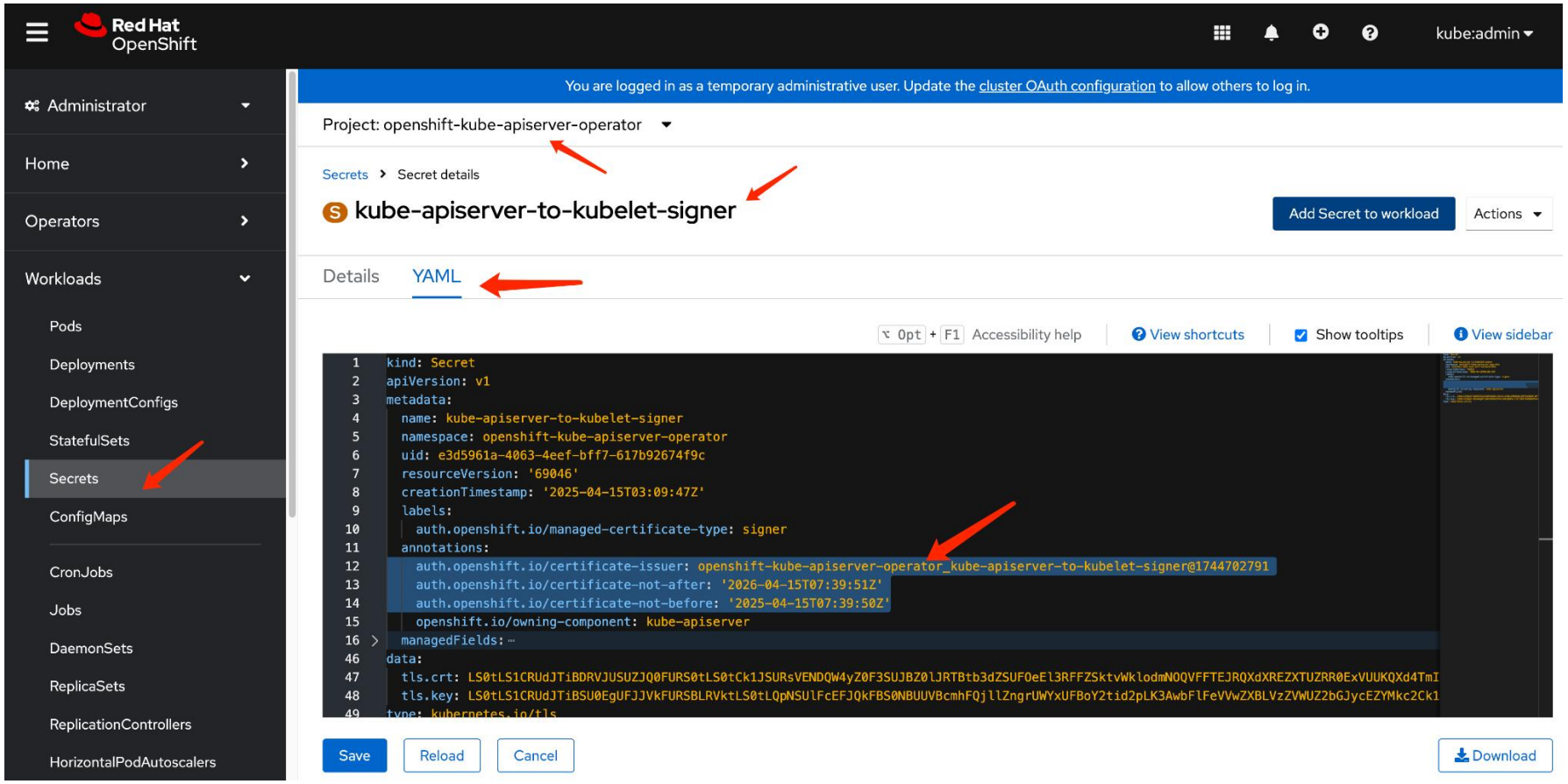
- new keys generate on control plan
- **do not trigger new machine config**, no new render machine config
- deliver to nodes in 2 ways, both using k8s api watch mechanism
- using dedicate logic in MCO, so no systemctl involved.



https://github.com/wangzheng422/docker_env/blob/dev/redhat/ocp4/4.16/2025.04.apiserver.mco.cert.rotation.en.md

How to trigger the cert rotation for testing

- go to namespace openshift-kube-apiserver-operator
- open secret kube-apiserver-to-kubelet-signer
- remove annotation and save
- This will trigger the kube-api-server certs rotation. That is all.

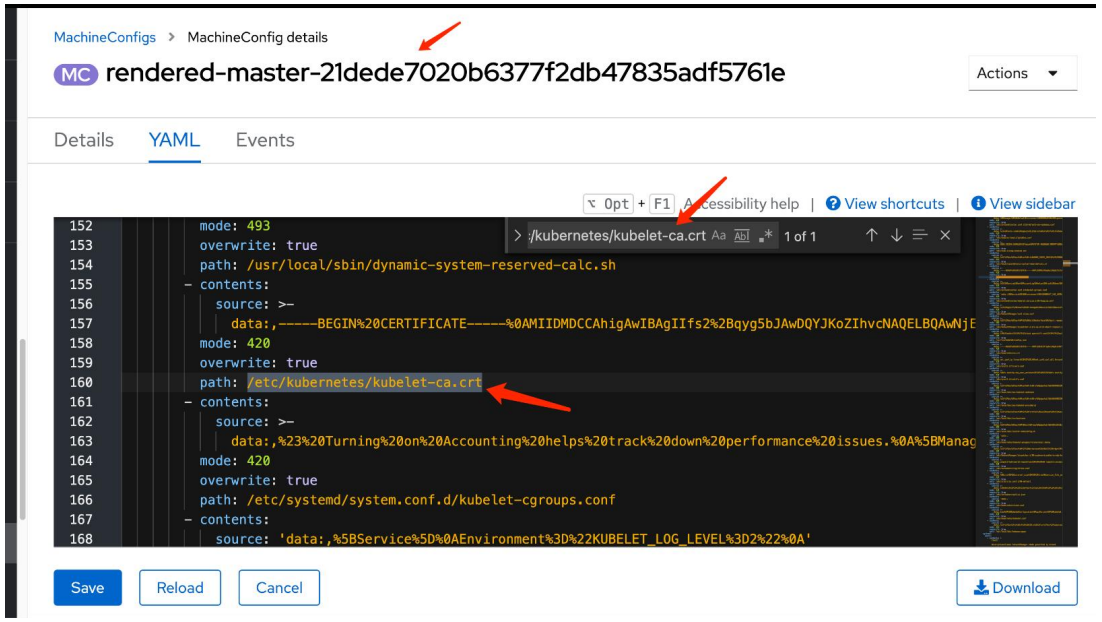


https://github.com/wangzheng422/docker_env/blob/dev/redhat/ocp4/4.16/2025.04.apiserver.mco.cert.rotation.en.md

background

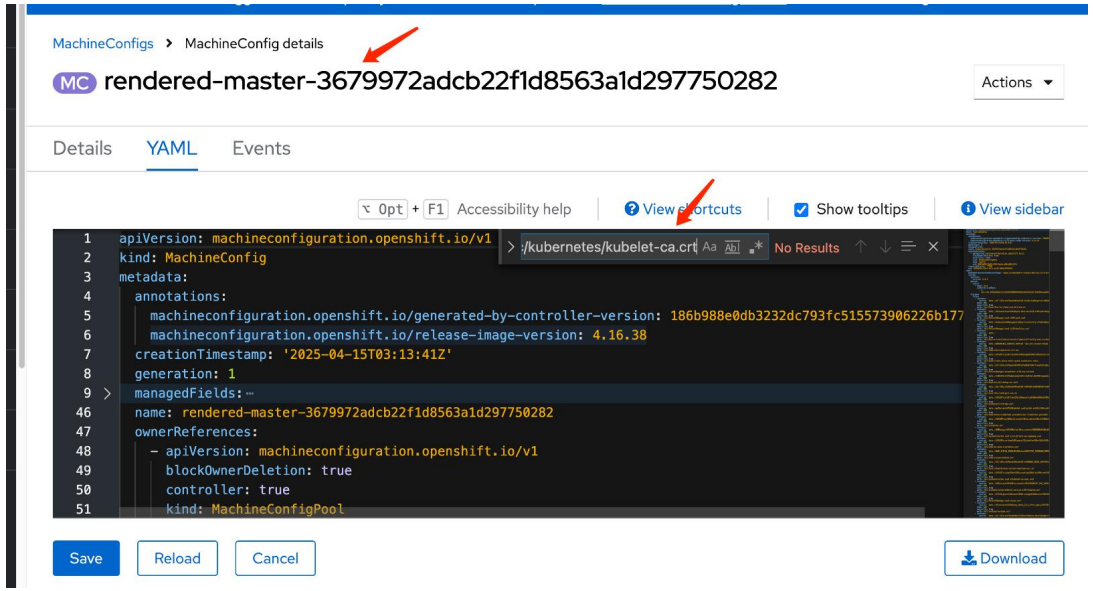
/etc/kubernetes/kubelet-ca.crt is removed from mco operator in ocp 4.16

In ocp4.12, you can find the file in machine configuration



VS

In ocp 4.16, you can not find the file



So mco will not monitor /etc/kubernetes/kubelet-ca.crt 's update, and will not trigger new machine config.

Red Hat

This code removed in ocp 4.16

- templates/common/_base/files/kubelet-ca.yaml
- This file is removed in ocp 4.16, so mco will not monitor the kubelet-ca.crt updating.

```
mode: 0644
path: "/etc/kubernetes/kubelet-ca.crt"
contents:
  inline: |
{{.KubeAPIServerServingCAData | toString | indent 4}}

```


https://github.com/openshift/machine-config-operator/blob/release-4.12/templates/common/_base/files/kubelet-ca.yaml

Key Code Logic

- by default, mco will reboot the node after new machine config applied.
- there are white-list hard coded
- when mco update such files, node reboot skipped.

```go
func calculatePostConfigChangeActionFromMCDiffs(diffFileSet []string) (actions []string) {
    filesPostConfigChangeActionNone := []string{
        caBundleFilePath,
        "/var/lib/kubelet/config.json",
    }
    directoriesPostConfigChangeActionNone := []string{
        constants.OpenShiftNMStateConfigDir,
    }
    filesPostConfigChangeActionReloadCrio := []string{
        constants.ContainerRegistryConfPath,
        GPGNoRebootPath,
        "/etc/containers/policy.json",
    }
    filesPostConfigChangeActionRestartCrio := []string{
        "/etc/pki/ca-trust/source/anchors/openshift-config-user-ca-bundle.crt",
    }
```

https://github.com/openshift/machine-config-operator/blob/release-4.16/pkg/daemon/update.go#L569

Key Code Logic

- mco use command systemctl to interact with systemd

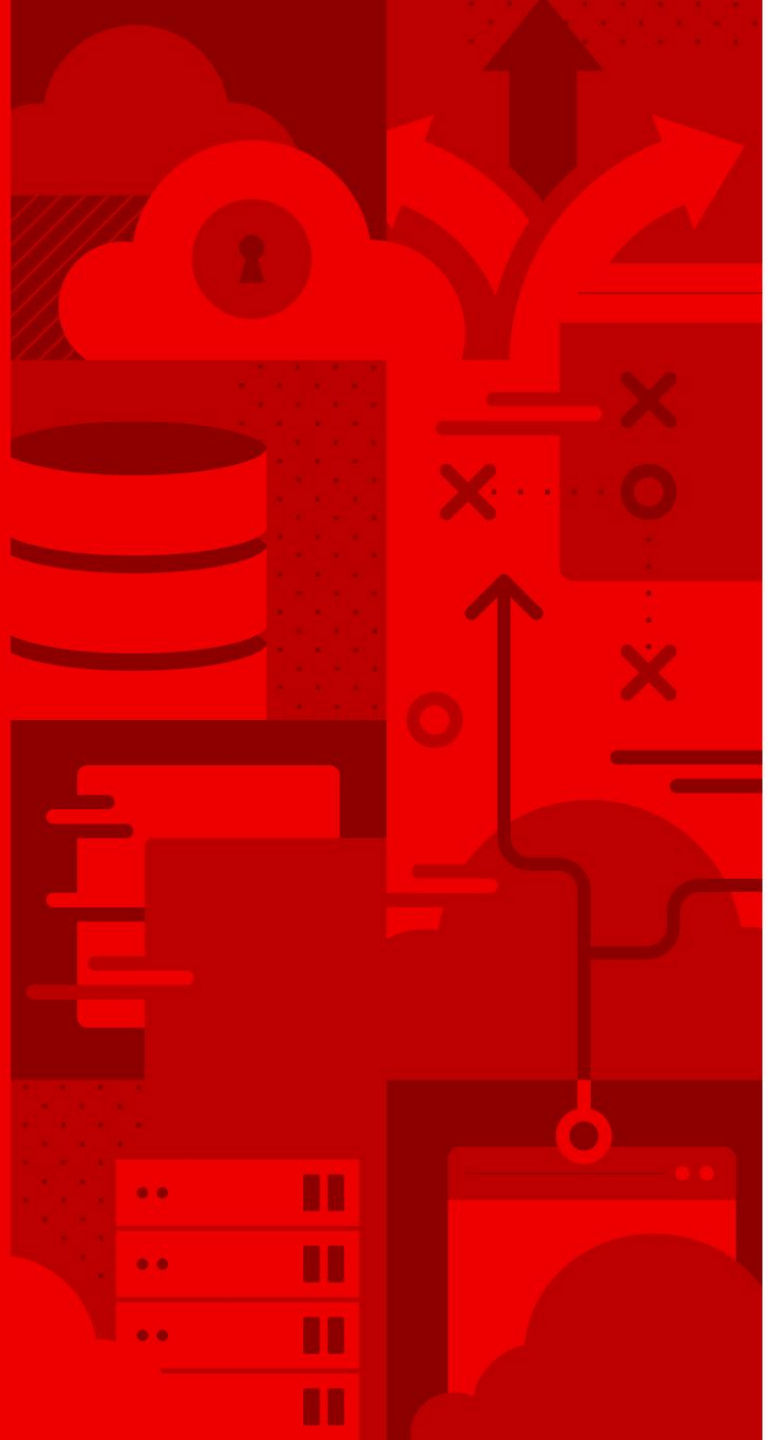
```go
func restartService(name string) error {
    return runCmdSync("systemctl", "restart", name)
}

func reloadService(name string) error {
    return runCmdSync("systemctl", "reload", name)
}

func reloadDaemon() error {
    return runCmdSync("systemctl", constants.DaemonReloadCommand)
}
```

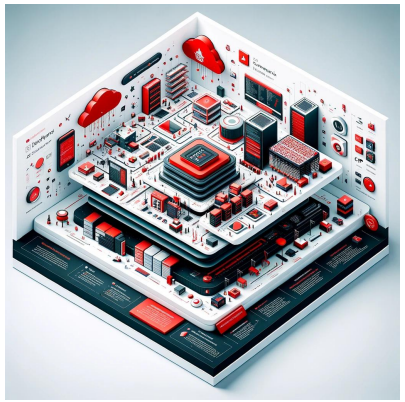
https://github.com/openshift/machine-config-operator/blob/release-4.16/pkg/daemon/update.go#L84

The logic changed between ocp 4.12 and 4.16

- Generally, changing files on ocp node needs to go through machine config operator (new machine config version), and reboot
  a. involving systemd actions
  b. white-list some file without reboot

- In ocp 4.16, /etc/kubernetes/kubelet-ca.crt rotation logic separated from traditional machine config operator
  a. no new machine config version
  b. no systemd actions.





Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat