

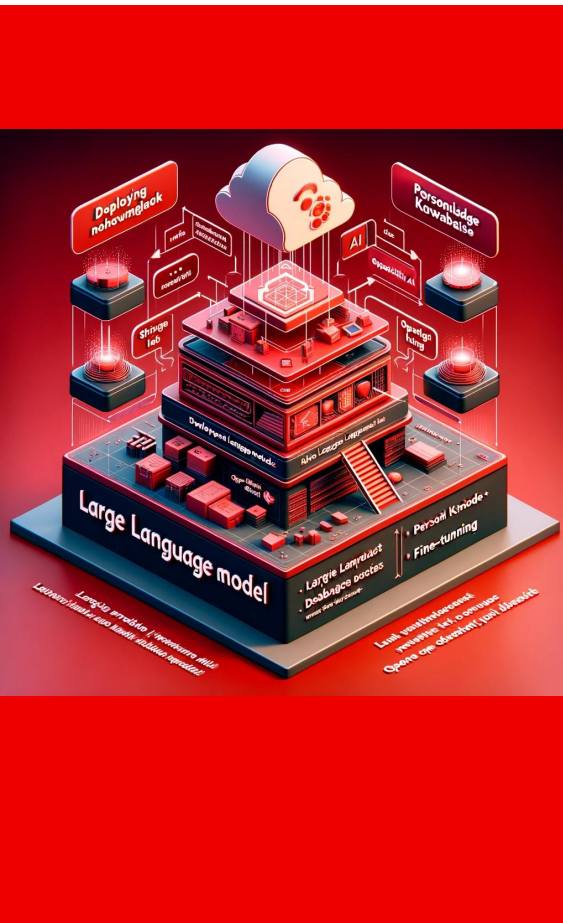
Machine Config Operator Certificate/Key Rotation Logic Comparing

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

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:

```
Feb 26 07:21:11 [host-name: root@15/10] machine-config-daemon[2000]: Starting update from
rendered-worker-4 50cga5d1fed39bf4141958m8K5 to
rendered-worker-4 77ad43403a745218c0c594b54b1874: K8sObjType: fake-kernel-fake-firmware
password-fake-file-fake-uninstall-kernel-type-fake-extension-fake
Feb 26 07:21:11 [host-name: kube-apiserver[5515]: I0226/07:21:11.244000-0000 dynamic_cert
rotation.go:27] "Failed to remove file which it may have been created
"/etc/kubernetes/kubeadm-config.toml" as have not existed finally watch for /etc/ke
bomeres/kubeadm-config"
Feb 26 07:21:11 [host-name: kube-apiserver[5515]: I0226/07:21:11.221100-0000 dynamic_cert
rotation.go:100] "Loaded a new CA Bundle and Verifier
name="client-ca-bundle"/etc/kubernetes/kubeadm-config"
Feb 26 07:21:11 [host-name: systemd[1]: Reloading.
Feb 26 07:21:15 [host-name: systemd[1]: Reloading
Feb 26 07:21:45 [host-name: systemd[1]: Reloading
Feb 26 07:21:12 [host-name: systemd[1]: Reloading
Feb 26 07:21:15 [host-name: logger[15735]: rendered-worker-
4 77ad43403a745218c0c594b54b1874
Feb 26 07:21:45 [host-name: logger[15736]: rendered-worker-
4 77ad43403a745218c0c594b54b1874
Feb 26 07:21:12 [host-name: root@15/10] machine-config-daemon[2000]: Node has Desired Config
rendered-worker-4 77ad43403a745218c0c594b54b1874, skipping reload
--
Feb 26 07:21:12 [host-name: root@15/10] machine-config-daemon[2000]:
Update completed for config rendered-worker-4 77ad43403a745218c0c594b54b1874 and node has
been successfully configured.
```

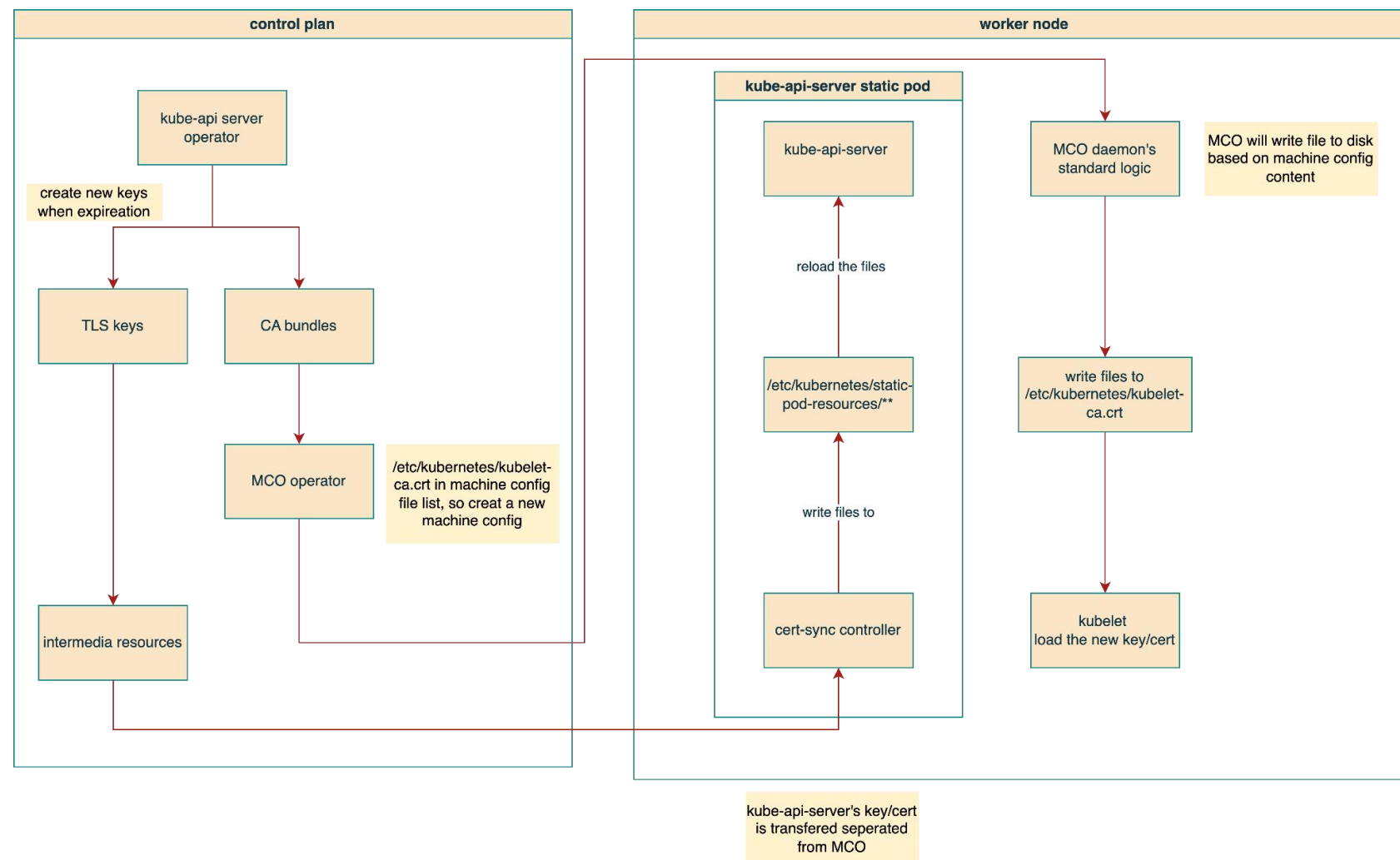
VS

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
```

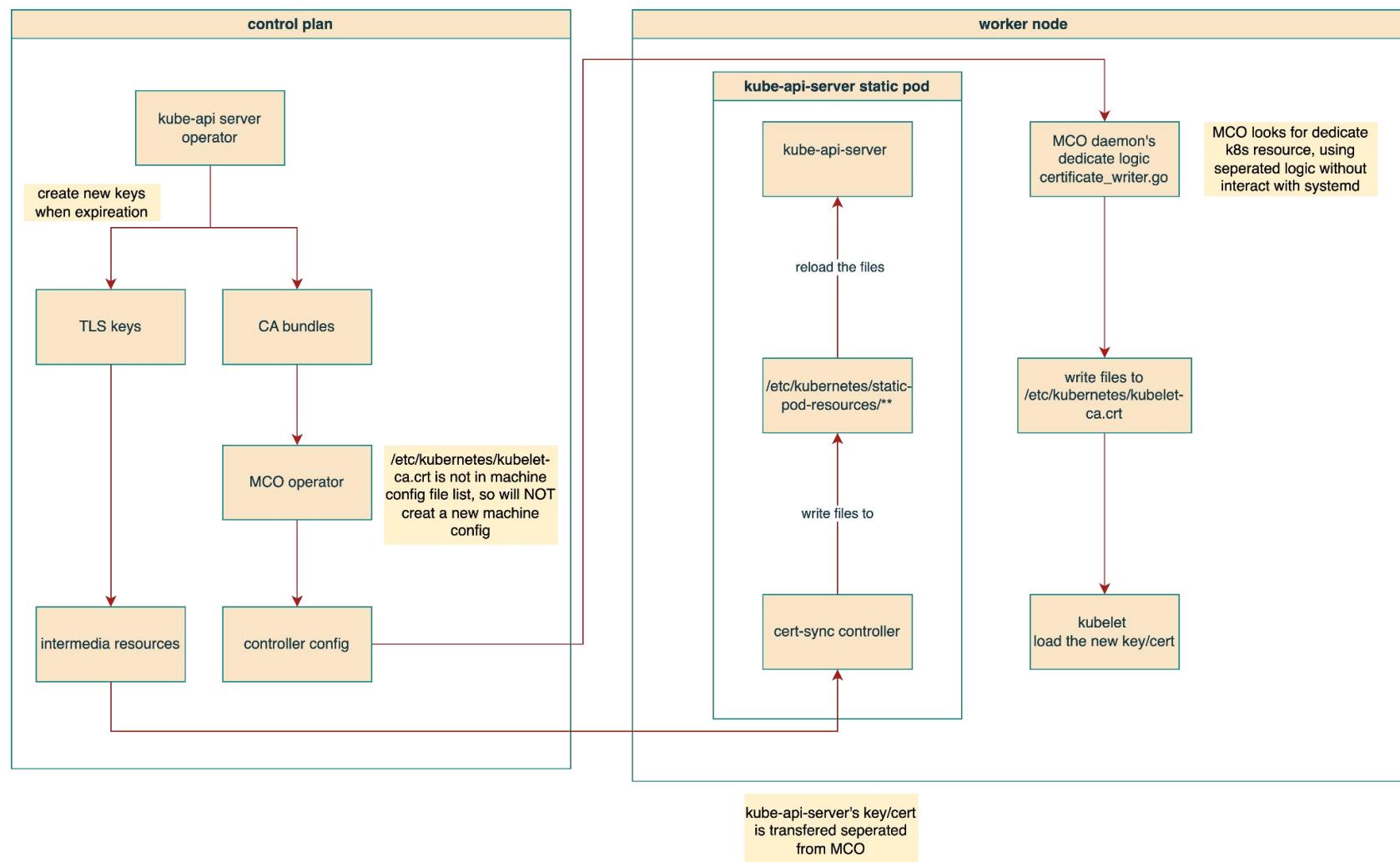
How things works for ocp 4.12

- 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.'`



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.



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.

The screenshot shows the Red Hat OpenShift console interface. On the left, the 'Secrets' menu item is highlighted. The main panel displays the details of the 'kube-apiserver-to-kubelet-signer' secret in the 'openshift-kube-apiserver-operator' namespace. The 'YAML' tab is selected, showing the following content:

```

1 kind: Secret
2 apiVersion: v1
3 metadata:
4   name: kube-apiserver-to-kubelet-signer
5   namespace: openshift-kube-apiserver-operator
6   uid: e3d5961a-4063-4eef-bff7-617b92674f9c
7   resourceVersion: '69046'
8   creationTimestamp: '2025-04-15T03:09:47Z'
9   labels:
10    auth.openshift.io/managed-certificate-type: signer
11   annotations:
12    auth.openshift.io/certificate-issuer: openshift-kube-apiserver-operator_kube-apiserver-to-kubelet-signer@1744702791
13    auth.openshift.io/certificate-not-after: '2026-04-15T07:39:51Z'
14    auth.openshift.io/certificate-not-before: '2025-04-15T07:39:50Z'
15    openshift.io/owning-component: kube-apiserver
16   managedFields: ...
46 data:
47   tls.crt: LS0tLS1CRUdJTiBDRVJUSUZZQ0FURS0tLS0tCk1JSURsVENDQW4yZ0F3SUJBZ0lJR...
48   tls.key: LS0tLS1CRUdJTiB0eG9uZGVkFURSB...
49 type: kubernetes.io/tls
  
```

Red arrows in the image point to the following elements:

- The 'openshift-kube-apiserver-operator' namespace in the breadcrumb.
- The 'kube-apiserver-to-kubelet-signer' secret name.
- The 'YAML' tab.
- The 'auth.openshift.io/certificate-issuer' annotation value.

/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

In ocp 4.16, you can not find the file

```
MachineConfigs > MachineConfig details
MC rendered-master-21dede7020b6377f2db47835adf5761e
Details YAML Events
mode: 493
  overwrite: true
  path: /usr/local/sbin/dynamic-system-reserved-calc.sh
  contents:
    source: >--
    data: ,-----BEGIN%20CERTIFICATE-----%0AMIIDMCCAhiGAWIBAgIIfs2%2BgYg5bJAwdQYJKoZIhvcNAQELBQAwNjE
mode: 420
  overwrite: true
  path: /etc/kubernetes/kubelet-ca.crt
  contents:
    source: >--
    data: ,%23%20Turning%20on%20Accounting%20helps%20track%20down%20performance%20issues.%0A%5BManag
mode: 420
  overwrite: true
  path: /etc/systemd/system.conf.d/kubelet-cgroups.conf
  contents:
    source: 'data:,%5BService%5D%0AEnvironment%3D%22KUBELET_LOG_LEVEL%3D2%22%0A'
```

VS

```
MachineConfigs > MachineConfig details
MC rendered-master-3679972adcb22f1d8563a1d297750282
Details YAML Events
1 apiVersion: machineconfiguration.openshift.io/v1
2 kind: MachineConfig
3 metadata:
4   annotations:
5     machineconfiguration.openshift.io/generated-by-controller-version: 186b988e0db3232dc793fc515573906226b177
6     machineconfiguration.openshift.io/release-image-version: 4.16.38
7   creationTimestamp: '2025-04-15T03:13:41Z'
8   generation: 1
9   managedFields: --
46 name: rendered-master-3679972adcb22f1d8563a1d297750282
47 ownerReferences:
48   - apiVersion: machineconfiguration.openshift.io/v1
49     blockOwnerDeletion: true
50     controller: true
51     kind: MachineConfigPool
```

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

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.

```
1  mode: 0644
2  path: "/etc/kubernetes/kubelet-ca.crt"
3  contents:
4    inline: |
5    {{.KubeAPIServerServingCAData | toString | indent 4}}
6
```


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.

```
1 func calculatePostConfigChangeActionFromMCDiffs(diffFileSet []string) (actions []string) {
2     filesPostConfigChangeActionNone := []string{
3         caBundleFilePath,
4         "/var/lib/kubelet/config.json",
5     }
6     directoriesPostConfigChangeActionNone := []string{
7         constants.OpenShiftNMStateConfigDir,
8     }
9     filesPostConfigChangeActionReloadCrio := []string{
10        constants.ContainerRegistryConfPath,
11        GPGNoRebootPath,
12        "/etc/containers/policy.json",
13    }
14    filesPostConfigChangeActionRestartCrio := []string{
15        "/etc/pki/ca-trust/source/anchors/openshift-config-user-ca-bundle.crt",
16    }
```

Key Code Logic

- mco use command systemctl to interact with systemd

```
1 func restartService(name string) error {  
2     return runCmdSync("systemctl", "restart", name)  
3 }  
4  
5 func reloadService(name string) error {  
6     return runCmdSync("systemctl", "reload", name)  
7 }  
8  
9 func reloadDaemon() error {  
10    return runCmdSync("systemctl", constants.DaemonReloadCommand)  
11 }
```

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](https://www.linkedin.com/company/red-hat)

 [facebook.com/redhatinc](https://www.facebook.com/redhatinc)

 [youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)

 twitter.com/RedHat

