

作者：李晓辉

联系方式：

1. 微信：Lxh_Chat

2. 邮箱：939958092@qq.com

OpenShift容器平台4通过红帽企业Linux CoreOS加了好多新功能，还搞了个新的软件分发系统，专门用来更新集群和底层操作系统，这升级路径简直完美！重点是，OpenShift集群现在能搞空中更新（OTA）啦，太酷了！

这个OTA软件分发系统，能搞定控制器清单、集群角色，还有更新到特定版本需要的所有资源。有了它，集群能无缝运行最新版本，新功能、漏洞修复和安全补丁都能第一时间用上，升级停机时间也少得可怜，简直不要太爽！

openshift更新通道

Candidate通道

candidate通道是专门用来测试下一版本OpenShift容器平台的功能的。不过，这些release candidate版本还得再检查检查，要是符合质量标准了，才会升级到**fast**或**stable**通道。不过，红帽可不支持只在**candidate**通道里的更新哦。

Fast通道

一旦红帽宣布某个版本正式发布，**fast**通道就会提供更新。红帽是支持这个通道的更新的，它最适合开发和QA环境。

Stable通道

红帽支持团队和站点可靠性工程（SRE）团队会先在**fast**通道里监控集群的运行情况。要是这些集群通过了额外的测试和验证，**fast**通道里的更新就会被启用到**stable**通道。红帽也支持这个通道的更新，它最适合生产环境。

扩展更新支持通道

从OpenShift容器平台4.8开始，红帽把所有偶数编号的次要版本（比如4.8、4.10和4.12）都标记为扩展更新支持（EUS）版本。

对了，为了保证集群的稳定性和正确的支持级别，你可别在**stable**和**fast**通道之间来回切换哦！

暂停健康检测

升级集群的时候可得注意点，节点可能会暂时“掉线”，尤其是worker节点。计算机健康检查可能会误以为这些节点“生病”了，然后就重启它们。为了避免这种情况，更新集群之前，记得先把所有计算机健康检查资源给暂停掉。

你可以看看最多能有多少机器“不健康”。这里有几点需要注意：

- **NAME**：就是资源的名字，比如这个例子中的 `machine-api-termination-handler`。
- **MAXUNHEALTHY**：这个很重要，它表示在任何时间点上，允许有多少机器“不健康”。在这个例子中是 `100%`，也就是说，理论上所有机器都可以“不健康”，也不会触发自动处理。
- **EXPECTEDMACHINES**：这是预计存在的机器数，不过在输出里没显示具体数量。
- **CURRENTHEALTHY**：当前健康的机器数，这个在输出里也没显示具体数量。

```
[student@workstation ~]$ oc get machinehealthcheck -n openshift-machine-api
NAME                                     MAXUNHEALTHY   EXPECTEDMACHINES   CURRENTHEALTHY
machine-api-termination-handler        100%
```

将 `cluster.x-k8s.io/paused` 注释添加到计算机健康检查资源，以在更新集群之前将其暂停。

```
[student@workstation ~]$ oc annotate machinehealthcheck -n openshift-machine-api \
machine-api-termination-handler cluster.x-k8s.io/paused=""
```

在集群更新后，删除注释。

```
[student@workstation ~]$ oc annotate machinehealthcheck -n openshift-machine-api \
machine-api-termination-handler cluster.x-k8s.io/paused-
```

更新界面展示

 openshift-upgrade

更新注意事项

更新 OpenShift 集群之前，管理员得先手动点个头，才能把集群从 4.11 版本升级到 4.12 版本。这主要是为了避免升级后出现各种问题，比如工作负载、工具或者其他和集群打交道的组件还在用已经被删除的 API。

管理员得先检查集群里有没有工作负载在用那些被删除的 API，要是有的话，就得赶紧把这些受影响的组件迁移到新的 API 版本上。搞定之后，就可以手动确认了。

执行下面的命令后，`admin-acks` ConfigMap 就会更新，这就表示管理员已经确认了和 Kubernetes 1.25 版本相关的 API 移除通知，准备妥当，可以升级到 OpenShift 4.12 版本了。

```
oc patch configmap admin-acks -n openshift-config --type=merge \
  --patch '{"data":{"ack-4.11-kube-1.25-api-removals-in-4.12":"true"}}'
```

Kubernetes API 弃用策略

Kubernetes 的 API 版本是按照功能的成熟度来分类的，就像给不同的功能贴上了“实验中”“快上市”和“已经很稳”这样的标签。

- **v1alpha1**：这是实验阶段的功能，还在摸着石头过河呢。
- **v1beta1**：这是快上市的功能，已经有点模样了，但可能还会有点小问题。
- **v1**：这是已经很稳的功能，可以放心大胆地用。

API 版本	类别	描述
v1alpha1	Alpha	experimental 功能
v1beta1	beta	pre-release 功能
v1	Stable	stable 功能，全面可用

举例来说，我们看看cronjob这个资源的API版本，它的版本就是v1，说明是稳定版

```
[student@workstation ~]$ oc api-resources | egrep '^NAME|cronjobs'
```

NAME	SHORTNAMES	APIVERSION
cronjobs	cj	batch/v1

发布功能的 stable 版本时，beta 版本将标记为已弃用，并在三个 Kubernetes 版本后删除。


列出我们正在使用，但是将要弃用的API

```
[student@workstation ~]$ oc get apirequestcounts | awk '{if(NF==4){print $0}}'
```

NAME	REMOVEDINRELEASE
flowschemas.v1beta1.flowcontrol.apiserver.k8s.io	1.26
prioritylevelconfigurations.v1beta1.flowcontrol.apiserver.k8s.io	1.26

使用 OLM 更新operator

这个在界面上，如果operator有更新会提示，我们只需要在确认更新时，点击批准更新即可，对于自动审批的operator而言，在有更新时，会自动更新，无需管理员参与。

 openshift-olm

本文档在线版本：<https://www.linuxcenter.cn>