

## 7Kubernetes系列（八）图形化 UI 管理工具

### Kubernetes系列（八）图形化 UI 管理工具

本文主要介绍我主要用到的一些 kubernetes 管理工具，主要挑选了 Lens 、 k9S 以及 Kui。

#### Lens

Lens 适用于那些对 Kubectl 命令不太熟悉的同学，它将 cluster、service、pod等所有资源整合在一起便于统一访问。同时他还与 Prometheus 集成，可视化查看资源使用情况。同时与 Helm Charts 集成，发现、部署及管理存储库。

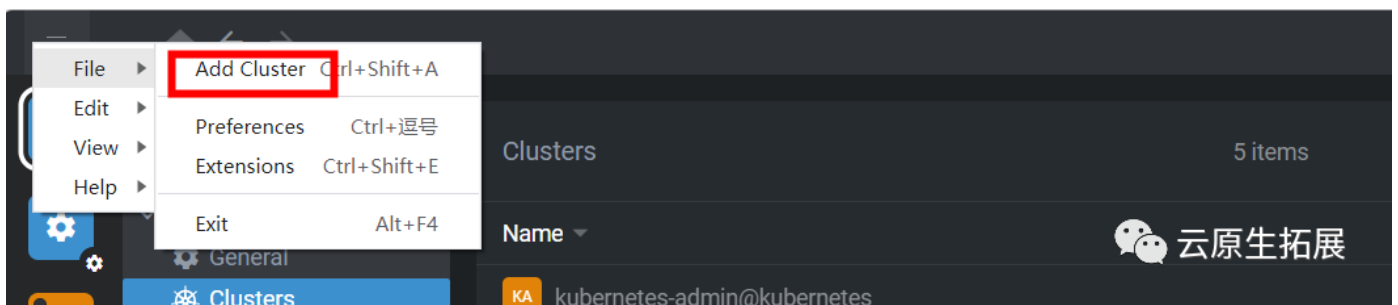
#### 安装

Lens 支持 MacOS、Windows和 Linux 操作系统，下面以 windows 为例进行说明。

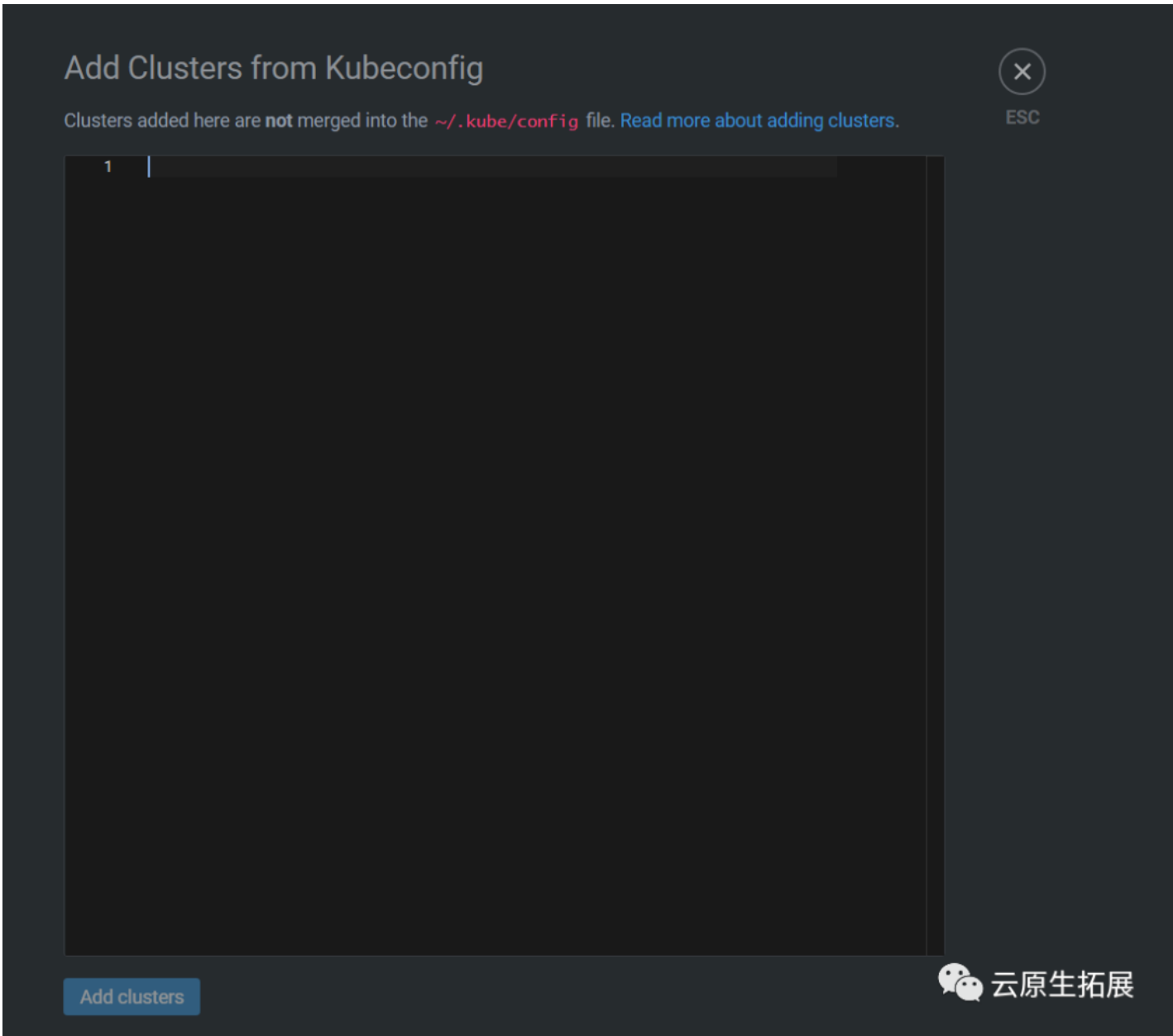
官网：[Lens | The Kubernetes IDE \(k8slens.dev\)](https://k8slens.dev) 下载安装程序，直接打开 `exe` 安装即可。

#### 使用

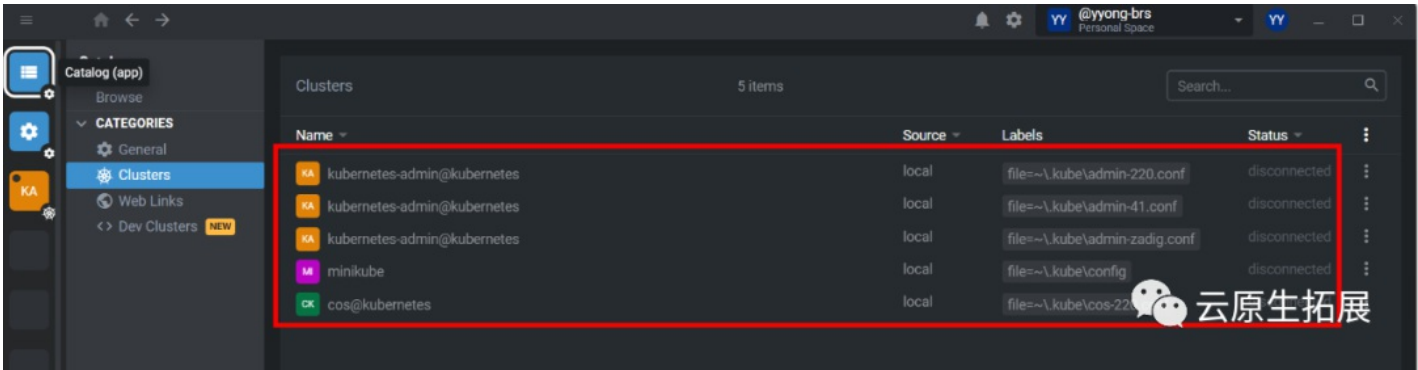
• 添加集群 打开 lens，点击File - Add Cluster;进入添加集群页面



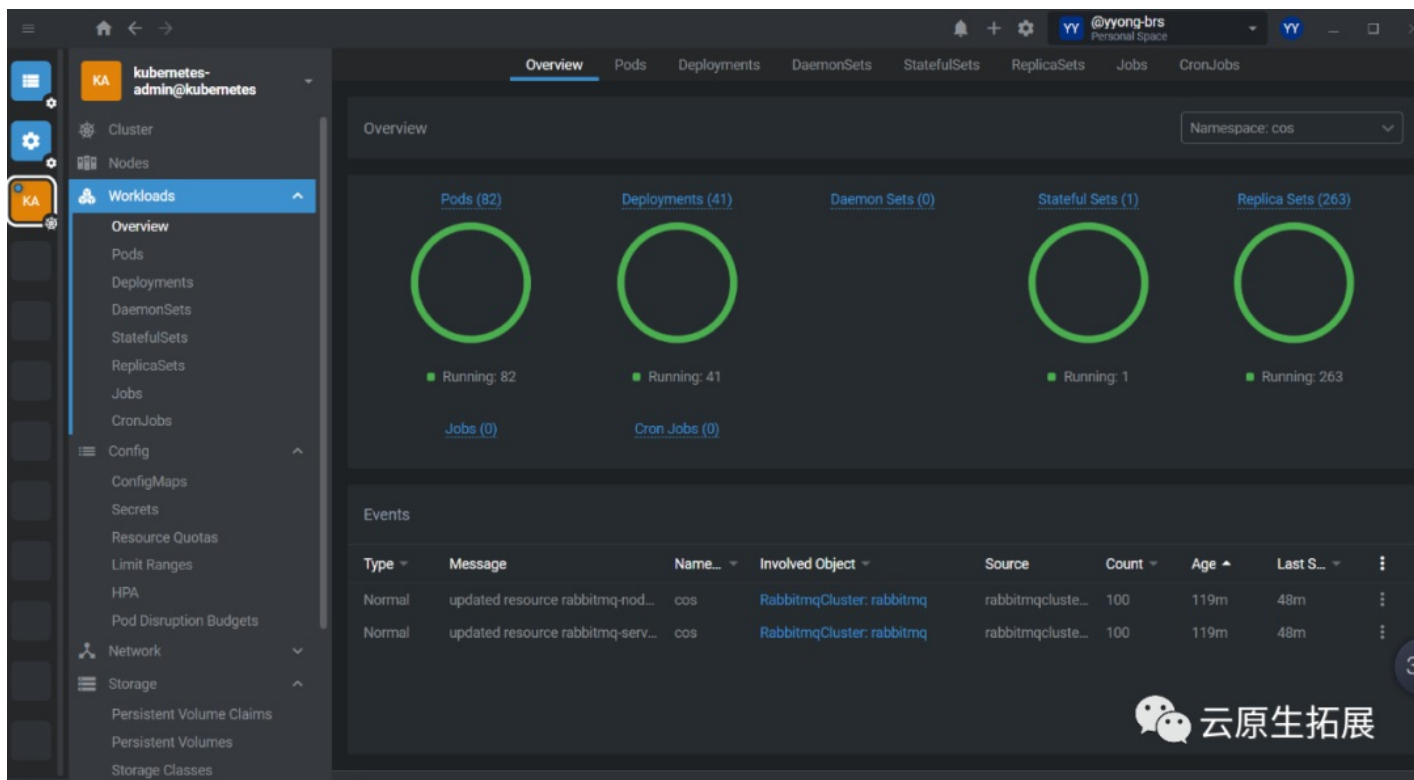
拷贝目标集群 config 文件内容，点击 `Add Clusters` 按钮保存集群



• 查看 Cluster



选择集群，进入集群详细页面：



左侧导航栏可以切换查看对应类型资源信息，包括日常的 pod 日志查看、远程登录等操作

## K9S

k9s 是一个终端UI，同样不需要掌握 kubectl 命令，它的操作方式有点类似 vim，同时会提示相关快捷操作指令。

### 安装

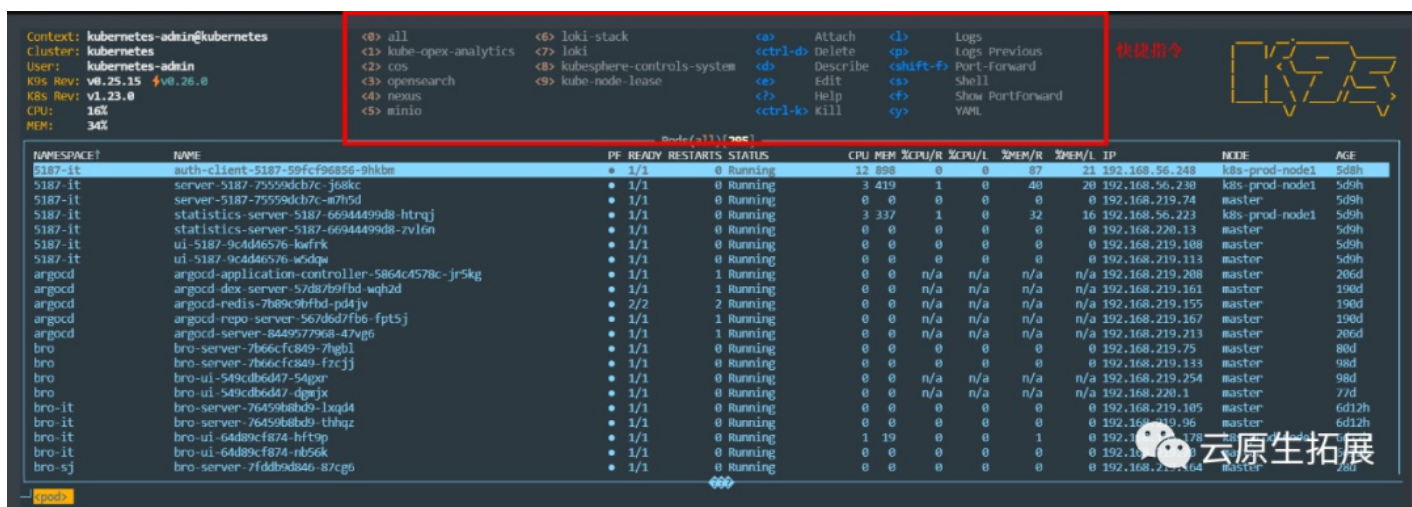
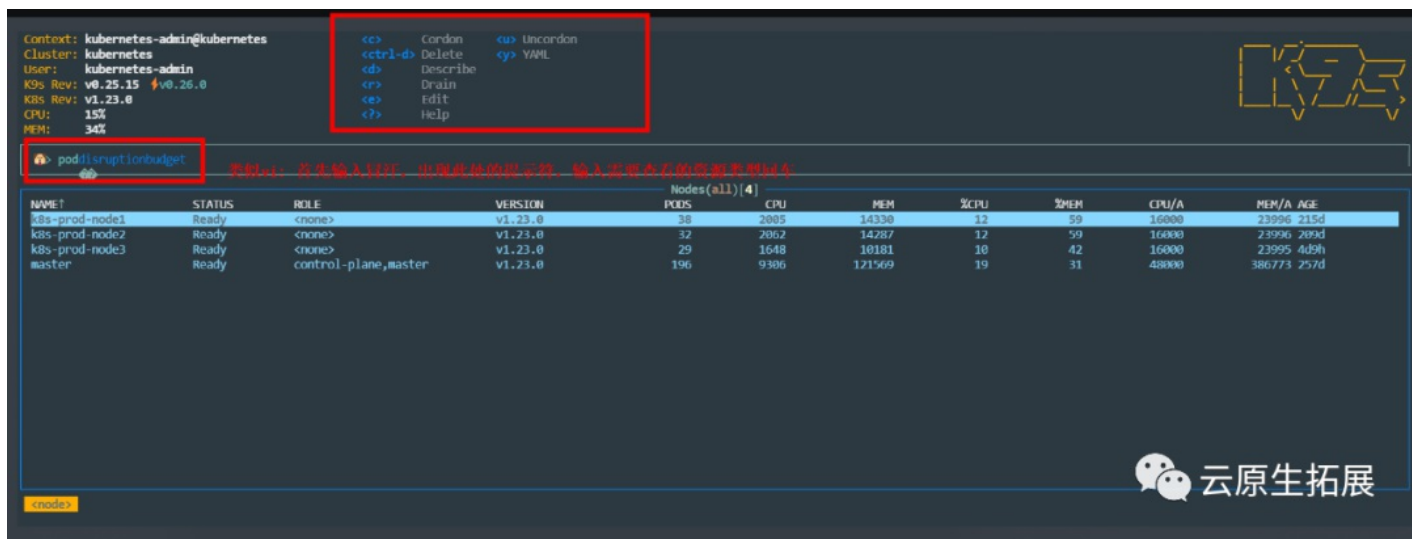
安装参考：[Install \(k9scli.io\)](https://k9scli.io/install/)

### 使用

#### 基本命令

```
# help
k9s help
# 启动
k9s
# 退出 k9s
q / quit
```

#### 运行参考



## Kui

Kui 是由 IBM 开源的云原生开发工具，Kui 提供了一种新的开发体验（原先大多数时候我们是通过 `kubectl` 与 Kubernetes 中的资源进行交互），Kui 结合了原有 CLI 的强大功能，并提供一种可视化的方式，方便我们对 Kubernetes 中 YAML 或者 JSON 格式数据的处理。

### 安装

安装方式其实很简单，项目的 [安装文档](#) 已经非常详细了。我主要推荐两种在 Linux 下的安装方式（当然，Kui 是支持跨平台的 macOS 和 Windows 也都可以用）：

- 下载预构建好的二进制包：

```
(MoeLove) → wget -O kui.zip https://linux-zip.kui-shell.org/
(MoeLove) → unzip kui.zip
(MoeLove) → ./Kui-linux-x64/Kui shell
```

- 通过以下命令可进行验证：

```
(MoeLove) → kui git:(master) kui version
2.33.0
(MoeLove) → kui git:(master) kubectl kui version
2.33.0
```

### 使用

你的机器上需要提前配置好 `KUBECONFIG` 的环境变量或者是将配置文件放到了正确的位置 i.e. `$HOME/.kube/config`

get pods -n kube-system

Last updated Today at 下午3:01:24

NAME	READY	STATUS	RESTARTS	AGE
coredns-fb8b8dccf-292wk	1/1	Running	0	154m
coredns-fb8b8dccf-lbj2g	1/1	Running	0	154m
etcd-moelove-control-plane	1/1	Running	0	153m
kube-apiserver-moelove-control-plane	1/1	Running	0	153m
kube-controller-manager-moelove-control-plane	1/1	Running	0	153m
kube-proxy-jltdj	1/1	Running	0	154m
kube-proxy-thpq5	1/1	Running	0	154m
kube-scheduler-moelove-control-plane	1/1	Running	0	153m
weave-net-dws2l	2/2	Running	0	154m
weave-net-tt82h	2/2	Running	0	154m

(MoeLove) → kui git:(master) kubectl kui get pods -n kube-system

```
NAME
coredns-fb8b8dccf-292wk
coredns-fb8b8dccf-lbj2g
etcd-moelove-control-plane
kube-apiserver-moelove-control-plane
kube-controller-manager-moelove-control-plane
kube-proxy-jltdj
kube-proxy-thpq5
kube-scheduler-moelove-control-plane
weave-net-dws2l
weave-net-tt82h
```

(MoeLove) → kui git:(master) kubectl kui get pods -n kube-system

(MoeLove) → kui git:(master) kubectl kui get pods -n kube-system

云原生拓展

默认情况下，如果我们没有给 `kui` 传递 `--ui` 参数的话，它和正常使用 `kubectl` 的行为一致，仍然在终端下输出内容。

类似这样：

```
(MoeLove) → kui git:(master) kubectl kui get pods -n kube-system
```

NAME	READY	STATUS	RESTARTS	AGE
coredns-fb8b8dccf-292wk	1/1	Running	0	153m
coredns-fb8b8dccf-lbj2g	1/1	Running	0	153m
etcd-moelove-control-plane	1/1	Running	0	152m
kube-apiserver-moelove-control-plane	1/1	Running	0	152m
kube-controller-manager-moelove-control-plane	1/1	Running	0	152m
kube-proxy-jltdj	1/1	Running	0	153m
kube-proxy-thpq5	1/1	Running	0	153m
kube-scheduler-moelove-control-plane	1/1	Running	0	152m
weave-net-dws2l	2/2	Running	0	153m
weave-net-tt82h	2/2	Running	0	153m

但如果我们给它传递了 `--ui` 的参数，则它会打开一个新的窗口，并将内容展示出来。

点击其中的任意资源，便可打开其详细内容：

KUBE-SYSTEM

kube-apiserver-moelove-control-plane

Created on Today at 下午12:27:51

```
1 apiVersion: v1
2 kind: Pod
3 metadata:
4   annotations:
5     kubernetes.io/config.hash: 8afb824c150f8f035d7c269570f6463c
6     kubernetes.io/config.mirror: 8afb824c150f8f035d7c269570f6463c
7     kubernetes.io/config.seen: "2019-05-25T04:26:25.774575123Z"
8     kubernetes.io/config.source: file
9   creationTimestamp: "2019-05-25T04:27:51Z"
10  labels:
11    component: kube-apiserver
12    tier: control-plane
13  name: kube-apiserver-moelove-control-plane
14  namespace: kube-system
15  resourceVersion: "564"
16  selfLink: /api/v1/namespaces/kube-system/pods/kube-apiserver-moelove-control-plane
17  uid: 7584f6fc-7ea5-11e9-94b4-0242ac1e0003
18 spec:
19   containers:
20     - command:
21       - kube-apiserver
22       - --advertise-address=172.30.0.3
23       - --allow-privileged=true
24       - --authorization-mode=Node,RBAC
25       - --client-ca-file=/etc/kubernetes/pki/ca.crt
```

YAML Status Conditions Containers

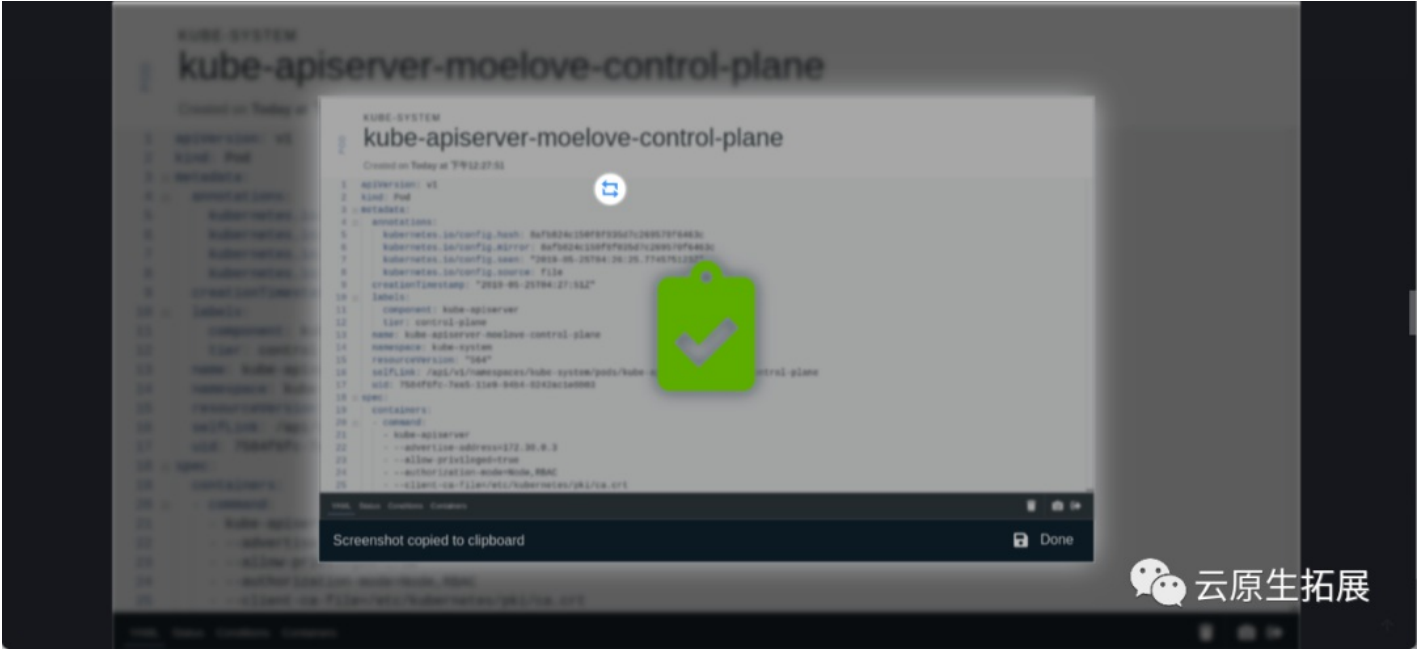
云原生拓展

底部 tab 可进行内容切换，比如可查看它的状态或者它其中的 Container 等。



当然，右下角还有一个删除按钮，可用于删除此资源（没有二次确认的，请慎重）

截图

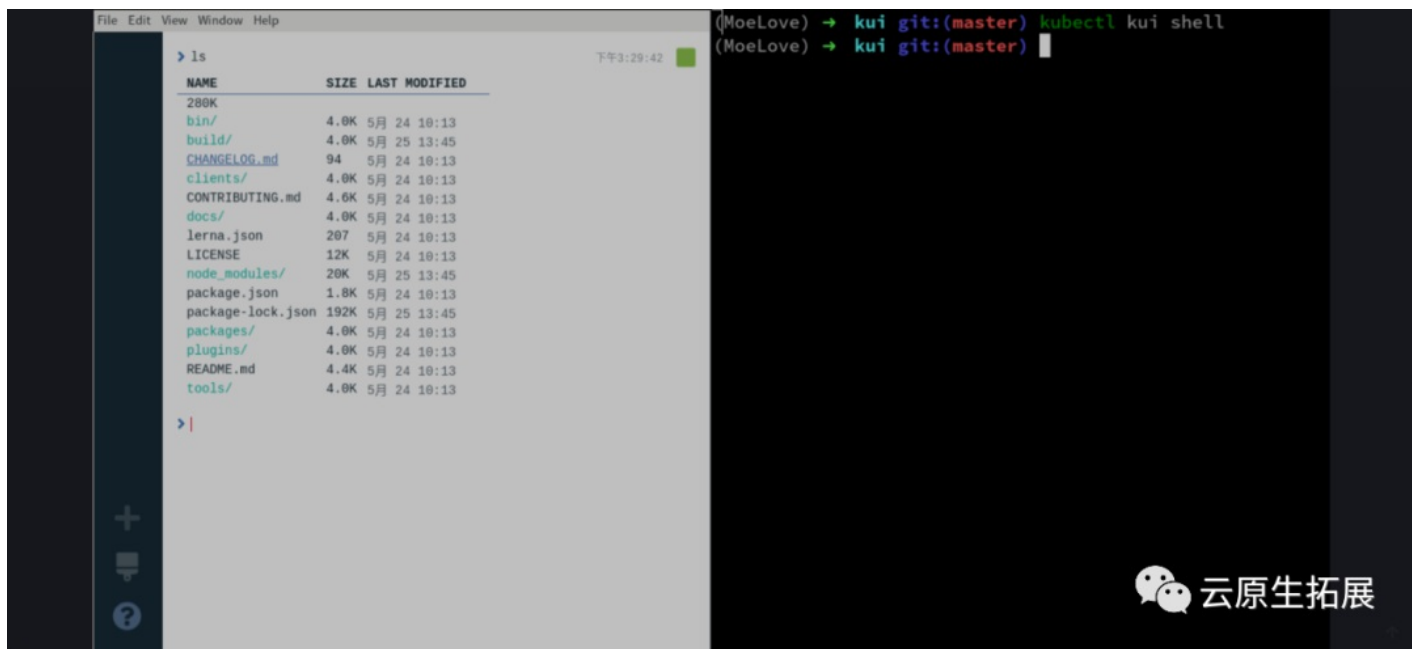


在窗口的右下角有一个截图按钮，使用此按钮可以方便的对当前窗口的图像进行拷贝和保存。

交互式 Shell

使用以下命令可以打开交互式 Shell 窗口：

```
kubectl kui shell
```



在此交互式 Shell 窗口中，你可以正常的执行你日常使用的命令：



当然除了像正常在终端中的命令输出外，如果输出内容是 JSON 或者 YAML 的话，则会自动打开 kui 的一个窗口进行展示。



PODS

NAME	READY	STATUS
coredns-fb8b8dcf-292wk	1/1	Running
coredns-fb8b8dcf-lbj2g	1/1	Running
etcd-moelove-control-plane	1/1	Running
kube-apiserver-moelove-control-plane	1/1	Running
kube-controller-manager-moelove-contr...	1/1	Running
kube-proxy-jltdj	1/1	Running
kube-proxy-thpq5	1/1	Running
kube-scheduler-moelove-control-plane	1/1	Running
weave-net-dws2l	2/2	Running
weave-net-tt8zh	2/2	Running

>

> kubectl -n kube-system get pods -o json

ok

> kubectl -n kube-system edit pod weave-net-tt8zh

Edit cancelled, no changes made.

> kubectl -n kube-system get pods -o json

ok

> |

GET PODS

get

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

```
"apiVersion": "v1",
"items": [
  {
    "apiVersion": "v1",
    "kind": "Pod",
    "metadata": {
      "creationTimestamp": "2019-05-25T04:26:53Z",
      "generateName": "coredns-fb8b8dcf-",
      "labels": {
        "k8s-app": "kube-dns",
        "pod-template-hash": "fb8b8dcf"
      },
      "name": "coredns-fb8b8dcf-292wk",
      "namespace": "kube-system",
      "ownerReferences": [
        {
          "apiVersion": "apps/v1",
          "blockOwnerDeletion": true,
          "controller": true,
          "kind": "ReplicaSet",
          "name": "coredns-fb8b8dcf",
          "uid": "52d81399-7ea5-11e9-94b4-0242ac1e0003"
        }
      ],
      "resourceVersion": "440",
      "selfLink": "/api/v1/namespaces/kube-system/pods/coredns-fb8b8dcf-292wk",
      "uid": "52df37b0-7ea5-11e9-94b4-0242ac1e0003"
    }
  }
],
"resourceVersion": "440",
"selfLink": "/api/v1/namespaces/kube-system/pods/coredns-fb8b8dcf-292wk",
"uid": "52df37b0-7ea5-11e9-94b4-0242ac1e0003"
}
```

JSON Status

云原生拓展

欢迎关注我的公众号“云原生拓展”，原创技术文章第一时间推送。