**原创；微信公众号：千里行走；**

**受限图片大小限制，有些图片不是很清晰，可以到微信公众号查看；**

提供helm/yaml容器化nginx-ingress的方式；

如果可能，尽量用公有云提供的ingress-nginx支持，方便快捷易维护，且可以直接通过便宜方式挂载到公有云的SLB上。

**目录**

**(1).容器化组件**

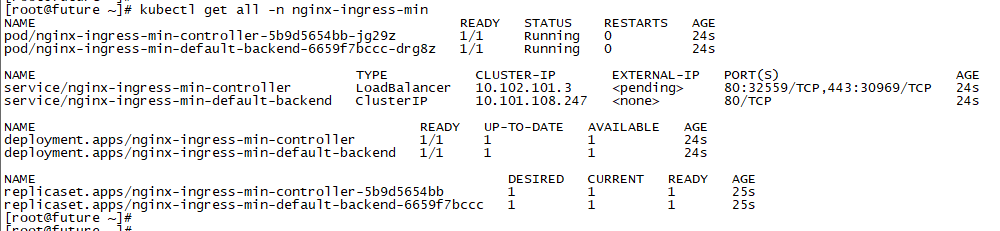
**(2).helm容器化方式**

**(3).yaml容器化方式**

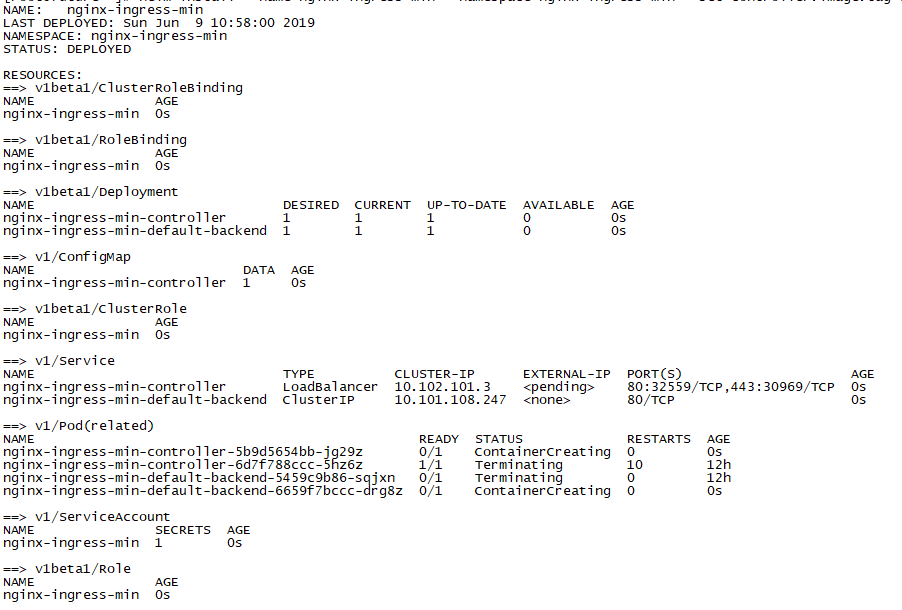
**正文**

**(1).容器化组件**

容器化成功后的组件：



所有相关组件：



**(2).helm容器化方式**

由于helm容器化nginx-ingress使用的镜像地址是k8s，网很慢/不通，所以需要从hub.docker.com下载后对镜像重命名：

(注意尽量从大公司的镜像下载，否则可能有不可知/很奇怪的错误，公司用最好从阿里云官方/k8s官方镜像下载，然后push到公司私有镜像仓库)

下载并重命名nginx-ingress-controller镜像：

docker pull ibmcom/nginx-ingress-controller:0.23.0

重命名镜像为：quay.io/kubernetes-ingress-controller/nginx-ingress-controller:0.21.0

docker images |grep nginx-ingress-controller |awk '{print "docker tag ",$1":"$2,$1":"$2}' |sed -e 's#ibmcom\/nginx-ingress-controller#quay\.io\/kubernetes-ingress-controller\/nginx-ingress-controller#2' |sh -x

下载并重命名defaultbackend-amd64镜像：

docker pull googlecontainer/defaultbackend:1.4

重命名镜像为：k8s.gcr.io/defaultbackend

docker images |grep defaultbackend-amd64 |awk '{print "docker tag ",$1":"$2,$1":"$2}' |sed -e 's#googlecontainer\/defaultbackend-amd64#k8s\.gcr\.io\/defaultbackend#2' |sh -x

执行helm容器化命令：

helm install --name nginx-ingress-min --namespace nginx-ingress-min --set controller.image.tag=0.23.0 stable/nginx-ingress

--name：容器中helm组件命名；

--namespace：指定部署到的命名空间；

--set：指定详细配置参数；

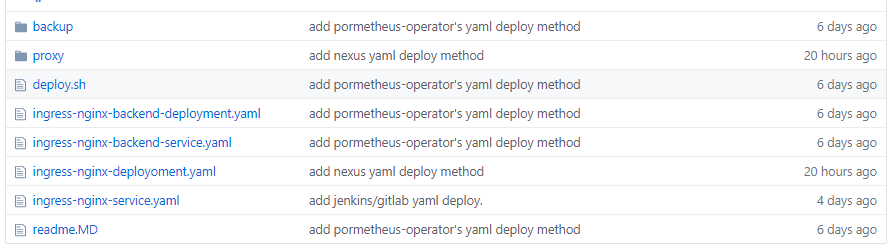
controller.image.tag：指定controller镜像版本；

**(3).yaml容器化方式**

ingress-nginx主要包含两个组件，ingress-nginx-controller和ingress-nginx-backend，前者提供配置功能，后者提供代理服务。

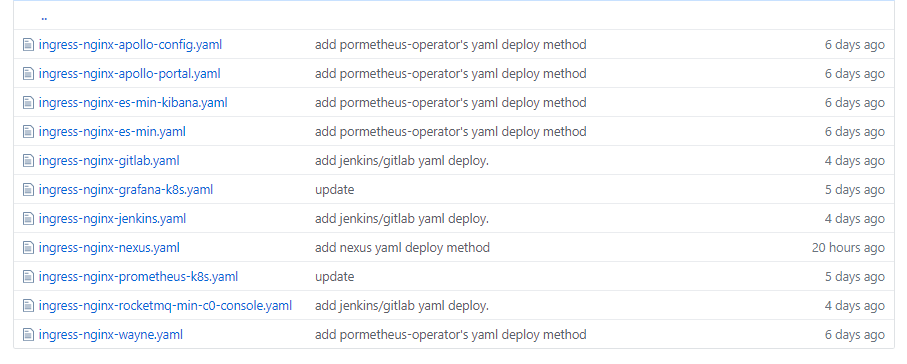
笔者提供了yaml方式的配置文件，位于：

<https://github.com/hepyu/k8s-app-config/tree/master/yaml/min-cluster-allinone/ingress-nginx-min>



提供了shell脚本直接部署：sh deploy.sh

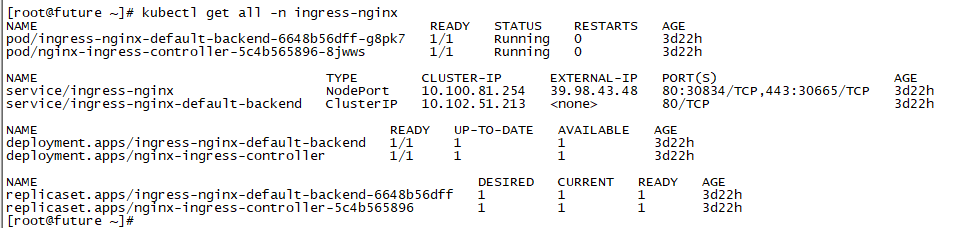
另外，在此目录下的proxy中，提供了代理各个基础中间件和辅助组件的yaml配置文件：



进入此目录，直接执行如下命令即可部署：

kubectl apply -f .

组件如下，与helm方式一样：



**(4).ingress-nginx配置参数**

在ingress-nginx-deployoment.yaml中有一个configmap的组件，在这里配置所有的nginx参数，如下图，配置了proxy-body-size：”600m”，表示ingress-nginx支持的请求数据的上限是600m。

其他参数请参见ingress-nginx官网查询，不同版本有可能配置方式不同。

