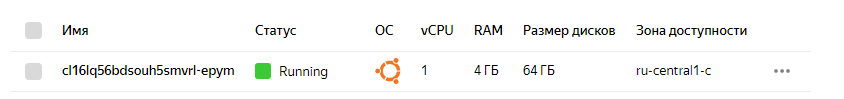
1. Регистрация в облаке
2. Зарегистрироваться в Яндекс-облаке. Прикрепить банковскую карту (даже для демо-режима). Создать каталог. Создать сервисный аккаунт. Создать кластер.
   1. Скачать CLI  
      curl https://storage.yandexcloud.net/yandexcloud-yc/install.sh | bash
   2. Сгенерировать Oath токен  
      <https://oauth.yandex.ru/>
   3. Инициализировать CLI. Добавить токен. Указать зону доступности облака (ту, где есть ip, если куплен (???):  
      yc init
3. Скачать kubectl
   1. <https://kubernetes.io/docs/tasks/tools/install-kubectl/>
   2. Создать папку   
      %userprofile%/.kube
   3. Добавить конфиги в kubectl:  
       yc managed-kubernetes cluster get-credentials sample-service-cluster --external
4. Создать группу узлов (минимум 64 Гб диска на узел):

yc managed-kubernetes node-group create --name sample-service-nodes-1 --cluster-name sample-service-cluster --fixed-size 1 --location zone=ru-central1-c --public-ip --disk-size 64 --cores 1

Ответ такого вида:

|  |
| --- |
| id: catr4tlol07k57n0m4on  cluster\_id: catcte2s00ujha7d5b3v  created\_at: "2019-06-04T17:34:47Z"  name: sample-service-nodes-1  status: RUNNING  node\_template:  platform\_id: standard-v1  resources\_spec:  memory: "4294967296"  cores: "1"  core\_fraction: "100"  boot\_disk\_spec:  disk\_type\_id: network-hdd  disk\_size: "68719476736"  v4\_address\_spec:  one\_to\_one\_nat\_spec:  ip\_version: IPV4  scale\_policy:  fixed\_scale:  size: "1"  allocation\_policy:  locations:  - zone\_id: ru-central1-c  subnet\_id: b0cv0tkfd32lotr4ehkm  instance\_group\_id: cl16lq56bdsouh5smvrl  node\_version: 1.13.3 |

Для каждой ноды ВМ:  


1. Тестовый сервис  
   kubectl.exe run hello-kubernetes --image=k8s.gcr.io/echoserver:1.4 --port=8080  
   Ответ:

|  |
| --- |
| kubectl run --generator=deployment/apps.v1beta1 is DEPRECATED and will be removed in a future version. Use kubectl create instead.  deployment.apps/hello-kubernetes created |

Проверки  
kubectl.exe describe deployment  
kubectl.exe describe service  
Сервис не создан.  
kubectl.exe expose deployment hello-kubernetes --type=NodePort  
Проверить сервис – сервис создан.  
Ответ сервиса (IP внешний, купленный у яндекса, порт у казан при запросе описания сервиса)  
<http://84.201.147.39:30613/>

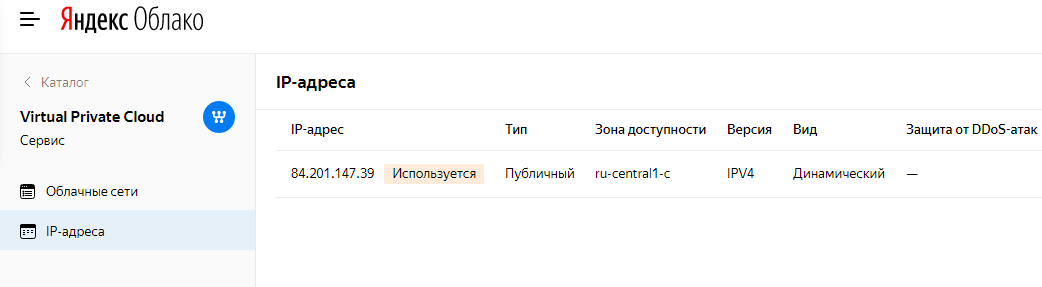
1. Проверить неймспейсы

|  |
| --- |
| $ kubectl.exe get namespaces  NAME STATUS AGE  default Active 23d  kube-public Active 23d  kube-system Active 23d |

1. Получить кластер и пользователя.   
   kubectl config view  
   Создать новый контекст под неймспейс kube-system.

По умолчанию яндекс дает контекст на три неймспейса. При установке деплоймента это может помешать, нужен 1 неймспейс.  
kubectl config set-context yandex-system --namespace=kube-system --cluster=yc-managed-k8s-aaabbbcccddd –user =yc-managed-k8s-aaabbbcccdddd

Переключиться на него  
kubectl config use-context yandex-system  
Справка: <https://kubernetes.io/docs/tasks/administer-cluster/namespaces-walkthrough/>

1. Развертывание деплоймента из файла  
   kubectl apply -f /c/work/git/openSource/dilvish13/OpenSourcePods/reactive-sandbox/reactive-sandbox\_deployment.json  
   Если не указывать имя файла будут загружены объекты из всей папки.  
   В ингрессе указать хост (статический ip от яндекса)  
     
   Проверка  
   kubectl.exe describe pods re
2. Настройка CI GitLab  
   Разрешить девопс на гитлабе для своего проекта  
   <https://gitlab.com/dilvish13/reactive-sandbox/-/settings/ci_cd#autodevops-settings>  
   Настроить подключение к кластеру куба на гитлабе.  
   Документация: <https://gitlab.com/help/user/project/clusters/index#adding-an-existing-kubernetes-cluster>  
   Добавить кластер – кнопка «Add Kubernetes cluster» 

Дальше- «Add existing cluster».  
В качестве «**API URL**» указать публичный IP. Его также можно узнать так:  
kubectl cluster-info | grep 'Kubernetes master' | awk '/http/ {print $NF}'

Указать сертификат CA, который узнать можно так:  
$ kubectl get secret $(kubectl get secrets | grep -o "^default-token-[[:alnum:]]\*") -o jsonpath="{['data']['ca\.crt']}

" | base64 –decode  
  
Используется поиск первого попавшегося секрета с именем default-token-XXXX  
Через kubectl загрузить в текущий неймспейс файл gitlab-admin-service-account.yml с содержимым

|  |
| --- |
| apiVersion: v1  kind: ServiceAccount  metadata:  name: gitlab-admin  namespace: kube-system  ---  apiVersion: rbac.authorization.k8s.io/v1beta1  kind: ClusterRoleBinding  metadata:  name: gitlab-admin  roleRef:  apiGroup: rbac.authorization.k8s.io  kind: ClusterRole  name: cluster-admin  subjects:  - kind: ServiceAccount  name: gitlab-admin  namespace: kube-system |

Загружать:  
kubectl apply –f путь-к-файлу  
Получить токен от этого аккаунта  
kubectl -n kube-system describe secret $(kubectl -n kube-system get secret | grep gitlab-admin | awk '{print $1}')| grep -o "^token:\s\*.\*$"| | grep -o "[^[:space:]]\*$"  
  
Использовать без пробелов и переносов строк.  
Сохранить изменения.

1. Гитлаб предложит установить в куб приложения.  
   Например, Prometheus позволяет из гитлаба просматривать установленные приложения.  
   GitLab Runner позволит выполнять CI/CD скрипты.  
     
   [Прометей](https://prometheus.io/docs/introduction/first_steps/) доступен по адресу  
    <http://localhost:9090/graph>   
   Для этого пробросить порты  
   kubectl port-forward services/prometheus-prometheus-server 9090:80