

Jenkins -> k8s

[Видео по настройке взаимодействия](#)

[Описание плагина Kubernetes](#)

[Описание плагина Credentials](#)

[Детальное описание Pod](#)

[Шпаргалка по k8s](#)

- Есть Jenkins вне кластера. Дополнительно установить плагин Kubernetes. Остальные нужные плагины подтянутся сами (Kubernetes client API, ...).

Kubernetes plugin 4313.va_9b_4fe2a_0e34

This plugin integrates Jenkins with Kubernetes

[Report an issue with this plugin](#)

- Есть кластер k8s с доступным для Jenkins внешним адресом. Внутри кластера есть одна нода с доступом в Интернет. Доступ с ноды для приложенного pipeline критичен из-за скачивания исполняемых файлов, если эти файлы есть внутри кластера - все еще лучше.
- На кластере создаем namespace, пользователя jenkins, токен

```
kubectl create namespace jenkins
```

```
kubectl create sa jenkins -n jenkins
```

```
kubectl create token jenkins -n jenkins --duration=8760h
```

#здесь отобразится токен, его нужно сохранить, затем добавим в настройки Cloud

```
kubectl create rolebinding jenkins-admin-binding --clusterrole=admin --
```

```
serviceaccount=jenkins:jenkins --namespace=jenkins
```

- //еще не полностью понял// Под термином Cloud в Jenkins понимается поддерживаемые типы виртуализации/контейнеризации для создания динамических агентов. Т е авторизация и т д, настраиваемая в Clouds, используется только для создания динамического агента в соответствующем типе виртуализации и затем удаления его. Все. Агент создается пустой, все остальное нужно доделывать. При установке соответствующих плагинов (Docker, Kubernetes, Virtualbox) в списке появляются соответствующие типы облаков.

- Для использования в динамическом агенте настроим авторизацию. Настроить Jenkins - Credentials - System
- Выбрать тип Secret text, в обычно одинаковый.

New credentials

ый токен, ID и Description

Kind

Secret text

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Secret

ID ?

forkuber

Description ?

forkuber

Create

- Добавляем Cloud. Name это имя, любое. Kubernetes URL - адрес кластера, по которому Jenkins может подключиться. Credentials - созданный на предыдущем шаге.

Name ?

Kuber

Kubernetes URL ?

https://192.168.1.185:6443

☐ Use Jenkins Proxy ?

Kubernetes server certificate key ?

☒ Disable https certificate check ?

Kubernetes Namespace

jenkins

Agent Docker Registry ?

☐ Inject restricted PSS security context in agent container

Credentials

kubesci

+ Add

- Справа есть кнопка Test connection Обязательно проверить соединение.
- Добавить стоит только адрес Jenkins сервера и Web Socket. Все остальное по умолчанию.

☒ WebSocket ?

☐ Direct Connection ?

Jenkins URL ?

http://192.168.1.194:8080/

Jenkins tunnel ?

Connection Timeout ?

5

- Настройка основы для динамического Jenkins клиента завершена.

Для работы с kubernetes нужен файл авторизации. Формат файла и настройки админа (для примера) можно посмотреть на кластере Kubernetes в

```
/etc/kubernetes/admin.conf
```

Сохраняем его в Credentials Jenkins и тип - файл. В данном случае ID textauth.



textauth

fuck.txt (textauth)

Secret file

textauth



Важно, что рабочая директория Jenkins может (и скорее всего будет) отличаться от homedir пользователя

Pipeline:

```
pipeline {
  agent {
    kubernetes (kubernetesAgent(name: 'mini'))
  }

  stages {
    stage('Integrate Remote k8s with Jenkins ') {
      steps {
        sh "mkdir ~/.kube"
        withCredentials([file(credentialsId: 'textauth', variable: 'FILE')]) {
          sh 'cp $FILE ~/.kube/config'
        }
        sh "curl -LO 'https://dl.k8s.io/release/v1.32.2/bin/linux/amd64/kubectl'"
        sh "chmod +x kubectl"
        sh "./kubectl get nodes"
      }
    }
  }
}
```

Revision #5

Created 22 February 2025 15:06:44 by Admin

Updated 25 February 2025 17:21:48 by Admin