

# Развертывание и настройка Jenkins

--

Для начала работ вам понадобится:

- *docker*
- *k3d*
- *helm*
- *kubect1*

Подробная установка и настройка всех необходимых инструментов описана в методических материалах к предыдущим урокам

--

## Часть 1. Установка jenkins

Запустим k3d-cluster:

```
$ k3d cluster create mycluster
```

```
filipp@filipp-notebook:~/Desktop$ k3d cluster create mycluster
INFO[0000] Prep: Network
INFO[0000] Created network 'k3d-mycluster'
INFO[0000] Created volume 'k3d-mycluster-images'
INFO[0000] Starting new tools node...
INFO[0000] Starting Node 'k3d-mycluster-tools'
INFO[0001] Creating node 'k3d-mycluster-server-0'
INFO[0001] Creating LoadBalancer 'k3d-mycluster-serverlb'
INFO[0001] Using the k3d-tools node to gather environment information
INFO[0001] HostIP: using network gateway 172.29.0.1 address
INFO[0001] Starting cluster 'mycluster'
INFO[0001] Starting servers...
INFO[0001] Starting Node 'k3d-mycluster-server-0'
INFO[0006] All agents already running.
INFO[0006] Starting helpers...
INFO[0006] Starting Node 'k3d-mycluster-serverlb'
INFO[0012] Injecting '172.29.0.1 host.k3d.internal' into /etc/hosts of all nodes
INFO[0012] Injecting records for host.k3d.internal and for 2 network members
INFO[0013] Cluster 'mycluster' created successfully!
INFO[0013] You can now use it like this:
kubect1 cluster-info
```

Добавим helm-repo "jenkins" в helm:

```
$ helm repo add jenkins https://charts.jenkins.io
```

```
filipp@filipp-notebook:~/Desktop$ helm repo add jenkins https://charts.jenkins.io
"jenkins" has been added to your repositories
```

Обновим helm-repos:

```
$ helm repo update
```

```
filipp@filipp-notebook:~/Desktop$ helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "jenkins" chart repository
...Successfully got an update from the "istio" chart repository
...Successfully got an update from the "gitlab" chart repository
Update Complete. ✨Happy Helming!✨
```

Создадим неймспейс для инструмента jenkins:

```
$ kubectl create ns jenkins
```

```
filipp@filipp-notebook:~/Desktop$ kubectl create ns jenkins
namespace/jenkins created
```

Произведем инсталляцию jenkins с помощью helm-chart (в нашем случае используется модифицированный вариант инсталляции с отключением механизма `health-check`, будьте внимательны при установке в `production-ready` среде!):

```
$ helm upgrade --install jenkins jenkins/jenkins --namespace
jenkins --set controller.healthProbes=false --set
controller.serviceType=NodePort
```

```
filipp@filipp-notebook:~/Desktop$ helm upgrade --install jenkins jenkins/jenkins --namespace jenkins --
set controller.healthProbes=false
Release "jenkins" does not exist. Installing it now.
NAME: jenkins
LAST DEPLOYED: Thu Feb  9 15:44:04 2023
NAMESPACE: jenkins
STATUS: deployed
REVISION: 1
NOTES:
1. Get your 'admin' user password by running:
  kubectl exec --namespace jenkins -it svc/jenkins -c jenkins -- /bin/cat /run/secrets/additional/chart
-admin-password && echo
2. Get the Jenkins URL to visit by running these commands in the same shell:
  echo http://127.0.0.1:8080
  kubectl --namespace jenkins port-forward svc/jenkins 8080:8080
3. Login with the password from step 1 and the username: admin
4. Configure security realm and authorization strategy
5. Use Jenkins Configuration as Code by specifying configScripts in your values.yaml file, see document
ation: http://127.0.0.1:8080/configuration-as-code and examples: https://github.com/jenkinsci/configura
tion-as-code-plugin/tree/master/demos

For more information on running Jenkins on Kubernetes, visit:
https://cloud.google.com/solutions/jenkins-on-container-engine

For more information about Jenkins Configuration as Code, visit:
https://jenkins.io/projects/jcasc/

NOTE: Consider using a custom image with pre-installed plugins
```

Выполним команду просмотра списка имеющихся pods в неймспейсе jenkins:

```
$ kubectl get pods -n jenkins
```

```
filipp@filipp-notebook:~/Desktop$ kubectl get pods -n jenkins
NAME          READY   STATUS    RESTARTS   AGE
jenkins-0     2/2     Running   0           57s
```

Jenkins успешно запущен.

Запросим службу jenkins, чтобы узнать выделенный NodePort:

```
$ kubectl get svc jenkins -n jenkins
```

```
filipp@filipp-notebook:~/Desktop$ kubectl get svc jenkins -n jenkins
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
jenkins      NodePort    10.43.71.62   <none>        8080:31879/TCP   110s
```

Узнаем external-ip службы traefik, выполняющую роль ingress-controller (обработчик входящих соединений) в кластере:

```
$ kubectl get svc traefik -n kube-system
```

```
filipp@filipp-notebook:~/Desktop$ kubectl get svc traefik -n kube-system
NAME         TYPE           CLUSTER-IP   EXTERNAL-IP   PORT(S)                                     AGE
traefik      LoadBalancer   10.43.236.138 172.29.0.3    80:32529/TCP,443:31779/TCP                18m
```

Вызовем команду просмотра пароля от пользователя admin в псевдотерминале контейнера jenkins внутри pod:

```
$ kubectl exec jenkins-0 -n jenkins -c jenkins -- cat
/run/secrets/additional/chart-admin-password
```


```
filipp@filipp-notebook:~/Desktop$ kubectl exec jenkins-0 -n jenkins -c jenkins -- cat /run/secrets/additional/chart-admin-password
r3b0fCpzoFNea1ajmD0wXyfilipp@filipp-notebook:~/Desktop$
```

Используя браузер, переходим в веб-интерфейс jenkins:

```
browser: http://<trafik-external-ip>:<jenkins-service-nodeport>
```

← → ↻ ⚠ Not secure | 172.29.0.3:31879/login?from=%2F

de-demo merion google drive



## Welcome to Jenkins!


☐ Keep me signed in

[Sign in](#)

Авторизуйтесь, используя логин "admin" и пароль, полученный ранее:

← → ↻ ⚠ Not secure | 172.29.0.3:31879

de-demo merion google drive

**Jenkins**

поиск (CTRL+K) ? 🔔 1

Dashboard >

+ Создать Item

👤 Пользователи

📁 История сборок

⚙ Настроить Jenkins

📌 My Views

Очередь сборки ▾

Очередь сборки пуста

Состояние сборщиков ▾

## Добро пожаловать в Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can builds or start building a software project.

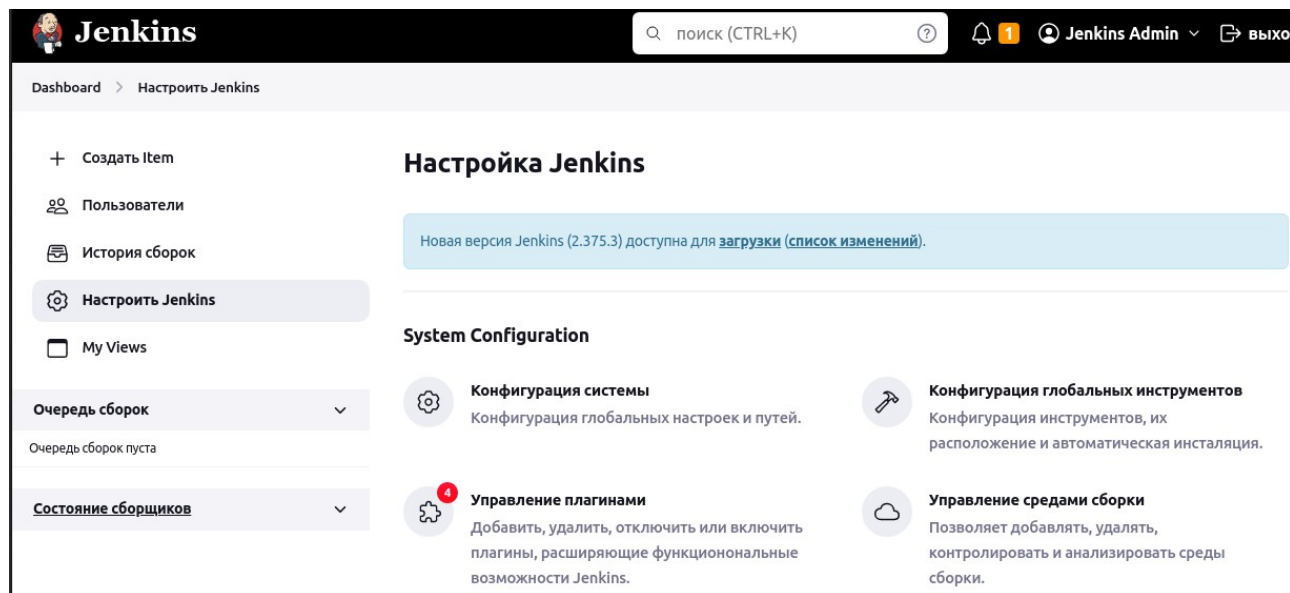
### Start building your software project

Create a job

--

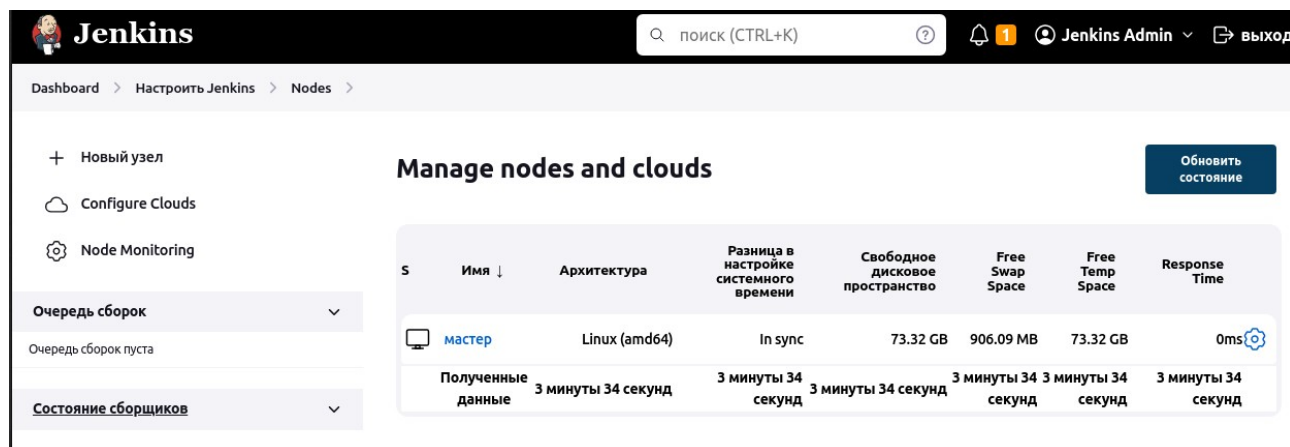
## Часть 2. Настройка среды сборки

В веб-интерфейсе jenkins переходим на вкладку "Настроить Jenkins", далее "Управление средами сборки":



The screenshot shows the Jenkins 'Configure Jenkins' page. The left sidebar contains navigation links: 'Создать Item', 'Пользователи', 'История сборок', 'Настроить Jenkins' (selected), and 'My Views'. Below these are sections for 'Очередь сборки' (Build Queue) and 'Состояние сборщиков' (Builders). The main content area is titled 'Настройка Jenkins' and features a notification about a new version (2.375.3) being available. Under the 'System Configuration' heading, there are four configuration options: 'Конфигурация системы' (System Configuration), 'Конфигурация глобальных инструментов' (Global Tool Configuration), 'Управление плагинами' (Plugin Management), and 'Управление средами сборки' (Build Environment Management). The 'Управление средами сборки' option is highlighted with a red circle and a number 4, indicating it is the next step in the tutorial.


Имеется один узел, выполняющий роль управляющего узла. Сделаем его одновременно и управляющим, и рабочим (*не подходит для production-ready решений!*). Нажимаем на символ шестеренки:



The screenshot shows the Jenkins 'Manage nodes and clouds' page. The left sidebar contains navigation links: 'Новый узел', 'Configure Clouds', and 'Node Monitoring'. Below these are sections for 'Очередь сборки' (Build Queue) and 'Состояние сборщиков' (Builders). The main content area is titled 'Manage nodes and clouds' and features a table of nodes. The table has columns for 'Имя' (Name), 'Архитектура' (Architecture), 'Разница в настройке системного времени' (System Time Difference), 'Свободное дисковое пространство' (Free Disk Space), 'Free Swap Space', 'Free Temp Space', and 'Response Time'. The table lists two nodes: 'мастер' (master) and 'Полученные данные' (Received data). The 'Полученные данные' node is highlighted with a red circle and a number 4, indicating it is the next step in the tutorial.

S	Имя ↓	Архитектура	Разница в настройке системного времени	Свободное дисковое пространство	Free Swap Space	Free Temp Space	Response Time
	мастер	Linux (amd64)	In sync	73.32 GB	906.09 MB	73.32 GB	0ms
	Полученные данные	3 минуты 34 секунд	3 минуты 34 секунд	3 минуты 34 секунд	3 минуты 34 секунд	3 минуты 34 секунд	3 минуты 34 секунд

В открывшемся окне меняем значение "Количество процессов-исполнителей" с "0" на "1", нажимаем "Сохранить":

**Jenkins**

поиск (CTRL+K)

Dashboard > Настроить Jenkins > Nodes > мастер

Статус

**Настроить**

История сборок

Статистика использования

Консоль сценариев

**Состояние сборщиков**

1 В ожидании

Количество процессов-исполнителей ?

1

Метки ?

Использование ?

Use this node as much as possible

Настройки узла

☐ Environment variables

Сохранить

Настройка среды сборки успешно завершена

--

### Часть 3. Проверка работоспособности CI/CD

Находясь в главном меню Jenkins, нажимаем на "Создать Item". Вводим произвольное наименование проекта и выбираем опцию "Создать задачу с произвольной конфигурацией". Нажимаем "ОК":



## Введите имя Item'a

» Обязательное поле



### Создать задачу со свободной конфигурацией

Это - основной и наиболее универсальный тип задач в Jenkins. Jenkins будет работать с любой сборочной системой. Данный тип проектов может использоваться



### Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for and/or organizing complex activities that do not easily fit in free-style job type



### Folder

Creates a container that stores nested items in it. Useful for grouping things to separate namespace, so you can have multiple things of the same name as long as they are in different folders



### Multibranch Pipeline

Creates a set of Pipeline projects according to detected branches in one SCM repository

Пролистываем страницу вниз до категории "Шаги сборки". Нажимаем "Добавить шаг сборки" и в выпадающем списке выбираем "Выполнить команду shell". Повторяем действие еще раз:

## Шаги сборки

☰ Выполнить команду shell ?

Команда

См. [список переменных окружения](#)

Расширенные...

☰ Выполнить команду shell ?

Команда

См. [список переменных окружения](#)

В первом шаге указываем произвольную команду shell (цель - проверка работоспособности сборочного конвейера):

```
$ echo 'hello'
```

Подобное выполняем и для второго шага:

```
$ cat /etc/hosts
```



## Шаги сборки

☰

Выполнить команду shell ?

✕

Команда

См. [список переменных окружения](#)

```
echo 'hello'
```

Расширенные...

☰

Выполнить команду shell ?

✕

Команда

См. [список переменных окружения](#)

```
cat /etc/hosts
```

Расширенные...

Нажимаем "Сохранить"

Находясь на странице проекта, нажимаем кнопку "Собрать сейчас":

Dashboard > myproject >

Статус

Изменения

Сборочная директория

Собрать сейчас

Настройки

Удалить Проект

Rename

История сборок

тренд

Filter builds...

#1 9 февр. 2023 г., 18:50

Atom feed для всех

Atom feed для неудачных

Проект myproject

Постоянные ссылки

В истории сборок нажимаем на сборку "#1", переходим в категорию "Вывод консоли":

Dashboard > myproject > #1

Статус

Изменения

Вывод консоли

Просмотреть как неформатированный текст

Редактировать информацию сборки

Delete build '#1'

Timings

Вывод на консоль

Started by user Jenkins Admin  
Running as SYSTEM  
Building in workspace /var/jenkins\_home/workspace/myproject  
[myproject] \$ /bin/sh -xe /tmp/jenkins15547472666430184576.sh  
+ echo hello  
hello  
[myproject] \$ /bin/sh -xe /tmp/jenkins18368022607943186329.sh  
+ cat /etc/hosts  
# Kubernetes-managed hosts file.  
127.0.0.1 localhost  
::1 localhost ip6-localhost ip6-loopback  
fe00::0 ip6-localnet  
fe00::0 ip6-mcastprefix  
fe00::1 ip6-allnodes  
fe00::2 ip6-allrouters  
10.42.0.16 jenkins-0.jenkins.jenkins.svc.cluster.local jenkins-0  
Finished: SUCCESS

Установка и настройка Jenkins, а также процессов CI/CD завершена успешно

--