

# Установка

## Jenkins server, docker compose

### [Источник](#)

- Создать директорию jenkins\_compose

```
mkdir jenkins_compose && cd jenkins_compose
```

- Создать compose файл

```
nano docker-compose.yaml
```

- Скопировать в файл текст, сохранить и выйти

```
services:
  jenkins:
    image: jenkins/jenkins:lts
    privileged: true
    user: root
    ports:
      - 8080:8080
      - 50000:50000
    container_name: jenkins
    volumes:
      - /home/${myname}/jenkins_compose/jenkins_configuration:/var/jenkins_home
      - /var/run/docker.sock:/var/run/docker.sock
```

- Запустить образ

```
docker compose up
```

- С данной конфигурацией по <http://localhost:8080> будет находиться web интерфейс. При первом старте в консоль будет выдан ключ инициализации, который нужно скопировать в web форму.
- Jenkins для обучения работает!

Для обучения на этом можно остановиться.

## Jenkins agent, docker compose

- В директории jenkins\_compose создаем ключи

```
ssh-keygen -t rsa -f jenkins_agent
```

- Перейти

**Настройка Jenkins**

Есть подозрение, что настройки вашего обратного прокси некорректны.

Выполнять сборки на встроенном узле может быть небезопасно. Рекомендуется настроить распределенные сборки. См. документацию.

**System Configuration**

- System**: Конфигурация глобальных настроек и путей.
- Tools**: Конфигурация инструментов, их расположение и автоматическая установка.
- Plugins**: Добавить, удалить, включить плагины функционала Jenkins.
- Clouds**: Add, remove, and configure cloud instances to provision agents on-demand.
- Appearance**: Configure the look and feel of Jenkins

**Security**

- Security**: Настройка безопасности Jenkins, конфигурация прав для доступа и использования системы.
- Credentials**: Configure credentials (highlighted with a red arrow)
- Credential Prov**: Configure the credential types

- Перейти в системное хранилище - глобальные параметры

## Stores scoped to Jenkins

Store	Domains
System	(global)

## System

Domain

Global credentials (unrestricted)

- Добавить способ авторизации со следующими настройками (ключ ввести вручную)

Scope ?

System (Jenkins and nodes only)

ID ?

jenkins\_agent

Description ?

For agen conn

Username


jenkins

☐ Treat username as secret ?

Private Key

☒ Enter directly

Key

 Concealed for Confidentiality

Passphrase

- Теперь добавить в compose файл образ агента с открытым ключом

```
services:
  jenkins:
    image: jenkins/jenkins:lts
    privileged: true
    user: root
    ports:
      - 8080:8080
      - 50000:50000
    container_name: jenkins
    volumes:
      - /home/${myname}/jenkins_compose/jenkins_configuration:/var/jenkins_home
      - /var/run/docker.sock:/var/run/docker.sock
  agent:
```

```
image: jenkins/ssh-agent
privileged: true
user: root
container_name: agent
expose:
  - 22
environment:
  - JENKINS_AGENT_SSH_PUBKEY=ssh-rsa...
```

- Почти все прошло по инструкции, кроме путей java и версии java. Второе было исправлено в compose (сейчас актуальная версия). А для установки корректного пути к JAVA нужно было найти в контейнере путь java

```
sergey@sergey-VirtualBox:~$ docker exec -it 2b5036c4a35b /bin/bash
root@2b5036c4a35b:/home/jenkins# env | grep JAVA
JAVA_HOME=/opt/java/openjdk
```

И установить следующий путь в настройках агента:

Расширенные ^  Edited

Port ?

22

JavaPath ?

/opt/java/openjdk/bin/java

- Агент поднялся!

---

Revision #7

Created 19 February 2025 17:01:08 by Admin

Updated 21 February 2025 16:28:54 by Admin