

Andrei Matveenکو

## 1. Ansible

ansible.Dockerfile:

FROM sbeliakou/centos:7.2

RUN yum install -y python-pip python-devel gcc epel-release yum-plugin-ovl openssl-devel && \

pip install ansible

**Ansible container usage:**

alias ansible-playbook='docker run --rm -v \$(pwd):\$(pwd) -w \$(pwd) ansible ansible-playbook -vv'

ansible-playbook playbook.yml

**Output:**

```
[root@epbyminw2976 docker2]# docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS
3014a26e1636   ansible   "/bin/bash"             21 seconds ago Up 20 seconds
6adf9bffa7500   docker2_deploy "/bin/sh -c 'java ..."  37 minutes ago Up 37 minutes   0.0.0.0:8080->8080/tcp
1d4cb8a62ddd   jenkins   "/bin/tini -- /usr..."  54 minutes ago Up 54 minutes   0.0.0.0:8080->8080/tcp
[root@epbyminw2976 docker2]# alias ansible-playbook='docker run --rm -v $(pwd):$(pwd) -w $(pwd) ansible ansible-playbook -vv'
[root@epbyminw2976 docker2]# ^C
[root@epbyminw2976 docker2]# ansible-playbook tomcat_provision.yml
No config file found; using defaults
[WARNING]: Host file not found: /etc/ansible/hosts
[WARNING]: provided hosts list is empty, only localhost is available

PLAYBOOK: tomcat_provision.yml *****
1 plays in tomcat_provision.yml

PLAY [Tomcat tomcat_provision] *****
skipping: no hosts matched

PLAY RECAP *****
```

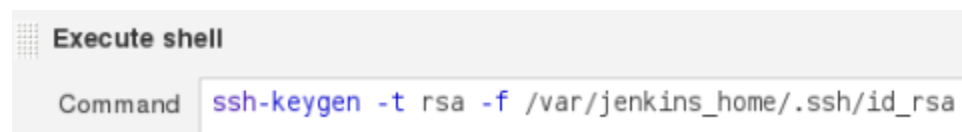
## Jenkins

docker run -d -p 8080:8080 -p 50000:50000 -v /home/student/docker/docker2:/var/jenkins\_home jenkins

**Connect host as a node.**

**Steps:**

- create job 1



- copy public key .ssh/id\_rsa.pub to ~/.ssh/authorized\_keys
- connect host as node

Name

Description

# of executors

Remote root directory

Labels

Usage

Launch method

Host

Credentials

S	Name ↓	Architecture
	<a href="#">master</a>	Linux (amd64)
	<a href="#">slave</a>	Linux (amd64)
Data obtained		33 min

```
[04/29/17 12:05:56] [SSH] Starting slave process: cd
<===[JENKINS REMOTING CAPACITY]====>channel started
Slave.jar version: 3.7
This is a Unix agent
Evacuated stdout
Agent successfully connected and online
```

## Gradle

gradle.Dockerfile:

FROM sbeliakou/centos:7.2

RUN yum install -y yum-plugin-ovl && \

yum install -y which zip unzip java-1.8.0-openjdk java-1.8.0-openjdk-devel && \

curl -s "https://get.sdkman.io" | bash && \

source "/root/.sdkman/bin/sdkman-init.sh" && \

sdk install gradle 3.5

ENV JAVA\_HOME /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.131-2.b11.el7\_3.x86\_64/

ENV PATH /root/.sdkman/candidates/gradle/3.5/bin/:\$PATH

## Spring-boot application

deploy.Dockerfile:

FROM sbeliakou/centos:7.2

ENV APP\_NAME gs-spring-boot-0.1.0.jar

ADD node1/workspace/2/initial/build/libs/\$APP\_NAME /opt/

RUN yum install -y yum-plugin-ovl && \

```
yum -y install java
```

```
CMD java -jar /opt/$APP_NAME
```

```
EXPOSE 8080
```

### **Spring Docker-compose**

```
docker-compose.yml:
```

```
version: '2'
```

```
services:
```

```
  deploy:
```

```
    build:
```

```
      context: .
```

```
      dockerfile: deploy.Dockerfile
```

```
    expose: [ "8080" ]
```

```
    ports: [ "0.0.0.0:8088:8080" ]
```

### **Jenkinsfiles:**

- **Gradle + spring:**

```
node('slave') {
```

```
  stage('Checking out') {
```

```
    git 'https://github.com/spring-guides/gs-spring-boot.git'
```

```
  }
```

```
  stage('Build application'){
```

```
    sh '''ls -la
```

```
    cd initial
```

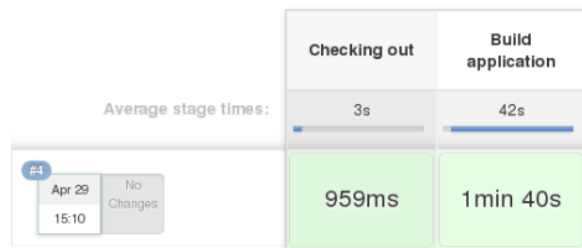
```
    alias gradle='docker run --rm -v $(pwd):$(pwd) -w $(pwd) gradle gradle'
```

```
    gradle build'''
```

```
  }
```

```
}
```

## Stage View



- **Spring compose:**

```
node('slave') {  
  stage('dockercompose'){  
    sh ''' cd /home/student/docker/docker2/  
    docker-compose up -d'''  
  }  
}
```

## Stage View

