

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ ИТМО

Дисциплина: Бек-энд разработка

Отчет

Практическая работа 6: Автодеплой приложения на  
удалённый сервер с использованием Github Actions

Выполнил:

Скороходова Елена

Группа  
K33392

Проверил:

Добряков Д. И.

Санкт-Петербург

2024 г.

## Задачи

Необходимо настроить автодеплой (с триггером на обновление кода в вашем репозитории, на определенной ветке) для вашего приложения на удалённый сервер с использованием Github Actions или Gitlab CI (любая другая CI-система также может быть использована).

## Ход работы

Создаем ssh-ключи с помощью команды `ssh-keygen -t ed25519` в терминале Windows. У нас создались приватный и публичный ключи.

```
C:\Users\1255974>ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (C:\Users\1255974\.ssh/id_ed25519):
Created directory 'C:\Users\1255974\.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\1255974\.ssh/id_ed25519.
Your public key has been saved in C:\Users\1255974\.ssh/id_ed25519.pub.
The key fingerprint is:
SHA256:bLf9I+JFvbAoh9lVtCkYxvtf1wNio1bK2EMv600Walg 1255974@LAPTOP-UF8HCHPA
The key's randomart image is:
+---[ED25519 256]--+
|      .o  .      |
|     ..o . o     |
|      ... +      |
|    o o.o =      |
|   * S +.= o .   |
|  E @ * *.o o+   |
| + + B = +...o   |
| . + o o....o    |
| . o. ... ..    |
+-----[SHA256]-----+

C:\Users\1255974>
```

Создаем Ubuntu вм в Yandex Cloud и подключаем по публичному ssh-ключу.

backend-hw  
Виртуальная машина

Обзор

Диски

Файловые хранилища

Резервные копии NEW

Права доступа

Операции

Мониторинг

Обзор

Идентификатор.....fv4iik9r9rjbqca4bkp1

Статус.....Running

Имя.....backend-hw

Дата создания.....04.06.2024, в 13:07

Внутренний FQDN.....backend-hw.ru-central1.internal

Зона доступности.....ru-central1-d

Доступ

Доступ через OS Login ?.....✕ Выключен

Но перед созданием надо в настройках разрешить доступ при помощи ssh-ключей.

## Настройки безопасности

### Доступ к VM при помощи OS Login

**i** Yandex OS Login используется для предоставления пользователям доступа по SSH к VM при помощи IAM. Для доступа к VM, у которых включена опция OS Login, пользователи должны обладать ролью `compute.osLogin` или `compute.osAdminLogin`. Используйте образы с предустановленным агентом OS Login из нашего маркетплейса либо установите агента самостоятельно.

Подробнее читайте в [документации](#).

#### Доступ по OS Login при помощи SSH-сертификатов (рекомендуется)



SSH-сертификаты это наиболее безопасный способ доступа, так как используются короткоживущие учетные данные вместо статических SSH-ключей, которые требуют периодической ручной ротации.

#### Доступ по OS Login при помощи SSH-ключей



Используйте доступ через OS Login с помощью SSH-ключей, если ваша инфраструктура требует использования внешних инструментов для подключения по SSH, или если ключи нужны для соблюдения политик безопасности вашей организации.











Далее в гитхабе в разделе Settings -> Settings and variables -> Action

создаем секреты

MY\_HOST - публичный IPv4 вМ

MY\_PRIVATE\_KEY - приватный ssh-ключ

MY\_USERNAME - username вМ

Repository secrets			New repository secret	
Name 	Last updated			
 MY_HOST	now			
 MY_PRIVATE_KEY	2 minutes ago			
 MY_USERNAME	2 minutes ago			

На машину устанавливаем все нужные зависимости (docker, docker compose и т.д.)

Подключаемся к вМ через терминал

ssh <имя\_пользователя>@<публичный\_IP-адрес\_ВМ>

```
updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

helbroone@backend-hw:~$
```

Далее настраиваем сценарий для деплоя deploy.yml

```
name: Deploy to Yandex.Cloud

on:
  push:
    branches:
      - '*'

jobs:
  deploy:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
      - uses: actions/setup-node@v1
        with:
          node-version: 20.x
      - name: Preparation
        run: |
          sudo apt-get update
          sudo apt-get install rsync -y
          mkdir "$HOME/.ssh"
          echo "${{ secrets.MY_PRIVATE_KEY }}" > "$HOME/.ssh/key"
          chmod 600 "$HOME/.ssh/key"
      - name: Sync
        run: |
          rsync -e "ssh -p 22 -i $HOME/.ssh/key -o StrictHostKeyChecking=no" -avz --exclude '.git' --chown=github:www-data --chmod=Dg=rwx,Fg=rwx --a
      - name: Build
        run: |
          ssh -p 22 -i $HOME/.ssh/key ${{ secrets.MY_USERNAME }}@${{ secrets.MY_HOST }} 'cd /home/helbroone/deploy/labs/K33392/Skorokhodova_Elena/la
```

И пушим на GitHub.

## Вывод

В домашнем задании номер шесть удалось настроить автодеплой на удалённый сервер с использованием Github Actions и Yandex Cloud.