

Deploy to Google Cloud

Before you begin:

- You need a Google service account key. For more guidance see [Google's guide to creating service keys](#).
- Ensure that an IAM user is configured with sufficient permissions to perform a deployment of your application using gcloud.
- Enable the right [APIs and services](#) needed for your application.

- All [cloud components](#) are installed.

Steps

- Clone the [Google App Engine example repository](#).
- Create a Google service account key. For more guidance see Google's [guide to creating service keys](#).
- Once you have your key file, open up a terminal and browse to the location of your key file.
- Encode your file in base64 format: `base64 -w 0 <your-key-file>` and copy the output of the command.
Note: for some versions of MacOS the `-w 0` is not necessary.
- In your repo go to **Repository settings**, under *Pipelines*, select **Repository variables** and create a new variable named `KEY_FILE` and paste the encoded service account credentials.
- Configure the pipe and add the variables: `PROJECT` and `KEY_FILE`

Variables list

PROJECT (*): The project that owns the app to deploy.

KEY_FILE (*): base64 encoded Key file for a [Google service account](#).

DEPLOYABLES: List of white space separated yaml files to be passed to gcloud. Default `""`. If left empty, the `app.yaml` from the current directory will be used.

VERSION: The version of the app to be created/replaced.

BUCKET: A google cloud storage bucket to store the files associated with the deployment.

IMAGE: Deploy with a specific GCR docker image.

PROMOTE: If true the deployed version receives all traffic, false to not receive traffic.

STOP_PREVIOUS_VERSION: If true the previous version receiving traffic is stopped, false to not stop the previous version.

EXTRA_ARGS: Extra arguments to be passed to the CLI (see [Google app deploy docs](#) for more details).

DEBUG: Turn on extra debug information. Default `false`.

(*) = required variable.

End to end example

Example

```
1 image: node:10.15.1
2
3 pipelines:
4   default:
5     - step:
6       name: Build and Test
7       script:
8         - npm install
9         - npm test
10    - step:
11      name: Deploy
12      script:
13        - pipe: atlassian/google-app-engine-deploy:0.2.1
14          variables:
15            KEY_FILE: $KEY_FILE
16            PROJECT: 'my-project'
```

Advanced examples

Deploy multiple deployables

Example

```
1 script:
2   - pipe: atlassian/google-app-engine-deploy:0.4.1
3     variables:
4       KEY_FILE: $KEY_FILE
5       PROJECT: 'my-project'
6       DEPLOYABLES: 'app-1.yaml app-2.yaml'
7       VERSION: 'alpha'
8       BUCKET: 'gs://my-bucket'
9       IMAGE: 'gcr.io/my/image'
10      PROMOTE: 'true'
11      STOP_PREVIOUS_VERSION: 'true'
12      EXTRA_ARGS: '--verbosity=debug'
```

Deploy a Python app to gcloud

Example

```
1 script:
2   - pipe: atlassian/google-app-engine-deploy:0.4.1
3     variables:
4       KEY_FILE: $KEY_FILE_PYTHON_APP
5       PROJECT: 'pipes-example-python3-app'
6       DEPLOYABLES: 'hello_world/app.yaml'
7       VERSION: '${BITBUCKET_BUILD_NUMBER}'
8       PROMOTE: 'true'
9       STOP_PREVIOUS_VERSION: 'true'
10      EXTRA_ARGS: '--verbosity=debug --quiet'
```

Deploy a Spring Boot based Java app build with Maven `app.yaml` to gcloud

Example

```
1 script:
2   - pipe: atlassian/google-app-engine-deploy:0.4.1
3     variables:
4       KEY_FILE: $KEY_FILE
5       PROJECT: 'pipes-java-maven-spring-boot'
6       DEPLOYABLES: 'target/appengine-staging/app.yaml'
7       VERSION: '${BITBUCKET_BUILD_NUMBER}'
8       EXTRA_ARGS: '--verbosity=debug'
```

Deploy a Spring Boot based Java app build with Gradle `app.yaml` to gcloud

Example

```
1 script:
2   - pipe: atlassian/google-app-engine-deploy:0.4.1
3     variables:
4       KEY_FILE: $KEY_FILE
5       PROJECT: 'pipes-java-gradle-spring-boot'
6       DEPLOYABLES: 'build/staged-app/app.yaml'
7       # DEPLOYABLES: 'src/main/WEB-INF/appengine-web.xml' # or you can provide web.xml
8       VERSION: '${BITBUCKET_BUILD_NUMBER}'
9       EXTRA_ARGS: '--verbosity=debug'
```

Deploy a Spring Boot based Java app build with Gradle `WEB-INF/appengine-web.xml` to gcloud

Example

```
1 script:
2   - pipe: atlassian/google-app-engine-deploy:0.4.1
3     variables:
4       KEY_FILE: $KEY_FILE
5       PROJECT: 'pipes-java-gradle-spring-boot'
6       DEPLOYABLES: 'src/main/WEB-INF/appengine-web.xml'
7       VERSION: '${BITBUCKET_BUILD_NUMBER}'
8       EXTRA_ARGS: '--verbosity=debug'
```

Deploy a standard Java app build with Maven `app.yaml` to gcloud

Example

```
1 script:
2   - pipe: atlassian/google-app-engine-deploy:0.4.1
3     variables:
4       KEY_FILE: $KEY_FILE
5       PROJECT: 'pipes-example-java-app-maven'
6       DEPLOYABLES: 'target/appengine-staging/app.yaml'
7       VERSION: '${BITBUCKET_BUILD_NUMBER}'
8       PROMOTE: 'true'
9       STOP_PREVIOUS_VERSION: 'true'
10      EXTRA_ARGS: '--verbosity=debug'
```

Deploy a standard Java app build with Maven `WEB-INF/appengine-web.xml` to gcloud

Example

```
1 script:
2   - pipe: atlassian/google-app-engine-deploy:0.4.1
3     variables:
4       KEY_FILE: $KEY_FILE
5       PROJECT: 'pipes-example-java-app-maven'
6       DEPLOYABLES: 'target/appengine-staging/WEB-INF/appengine-web.xml'
7       VERSION: '${BITBUCKET_BUILD_NUMBER}'
8       PROMOTE: 'true'
9       STOP_PREVIOUS_VERSION: 'true'
10      EXTRA_ARGS: '--verbosity=debug'
```

Deploy a standard Java app build with Gradle `WEB-INF/appengine-web.xml` to gcloud

Example

```
1 script:
2   - pipe: atlassian/google-app-engine-deploy:0.4.1
3     variables:
4       KEY_FILE: $KEY_FILE
5       PROJECT: 'pipes-example-java-app-gradle'
6       DEPLOYABLES: 'build/exploded-helloworld/WEB-INF/appengine-web.xml'
7       VERSION: '${BITBUCKET_BUILD_NUMBER}'
8       PROMOTE: 'true'
9       STOP_PREVIOUS_VERSION: 'true'
10      EXTRA_ARGS: '--verbosity=debug'
```

Deploy a standard Java app build with Gradle `app.yaml` to gcloud

Example

```
1 script:
2   - pipe: atlassian/google-app-engine-deploy:0.4.1
3     variables:
4       KEY_FILE: $KEY_FILE
5       PROJECT: 'pipes-example-java-app-gradle'
6       DEPLOYABLES: 'build/staged-app/app.yaml'
7       VERSION: '${BITBUCKET_BUILD_NUMBER}'
8       PROMOTE: 'true'
9       STOP_PREVIOUS_VERSION: 'true'
10      EXTRA_ARGS: '--verbosity=debug'
```

Access Pipelines deployment guides

Show more ▾

[Deploy to Amazon ECS](#)

[Deploy to Firebase](#)

• [Deploy to Google Cloud](#)

[Deploy to Heroku](#)

[Deploy to Kubernetes](#)

Show more ▾

On this page

[Before you begin:](#)

[Steps](#)

[Variables list](#)

Community

[Questions, discussions, and articles](#)

Was this helpful? ☐ Yes ☐ No [Provide feedback about this article](#)

Still need help?

The Atlassian Community is here for you.

[Ask the Community](#)