


# Deploy build artifacts to Bitbucket Downloads

Bitbucket app passwords and the Bitbucket REST API allow you to deploy an artifact that has been produced by your pipeline to Bitbucket Downloads.

See also [Test with databases in Bitbucket Pipelines](#).

## Step 1: Create an App password for the repository owner

Log in to Bitbucket > select the **Settings** cog  on the top navigation bar > select **Personal Bitbucket settings** from the **Settings** dropdown menu > select **App Passwords** under **Access management**.

Create a new [app password](#) with **write permissions** to your repositories, and take note of the generated password that pops up. The name of the password is only for your reference, so use "Pipelines" or any other label you like.

### Outcome

You should now have the app password to authenticate your pipe.

## Step 2: Create a Pipelines variable with the authentication token

In order to upload files to Bitbucket Downloads you will need to configure the following variables:

Parameter	Value
BITBUCKET_USERNAME	Bitbucket username of the repository owner (and also the user who will upload the artifacts)
BITBUCKET_APP_PASSWORD	App password as generated by bitbucket

You can define these variables for a specific deployment environment from the repository's settings. Select **Repository settings** > select **Repository variables** under **Pipelines**.

### Outcome

You should now have these two variables configured in your repository which means they are ready to be used from a pipeline.

## Step 3a: Deploy your artifacts using the bitbucket-upload-file pipe

Use [bitbucket-upload-file](#) pipe to upload your artifact to Bitbucket Downloads. You just need to reference the file you want to upload from your build output.

Below is a simple example from an Android project built with Gradle. You can make your debug Android Package (APK) available in Bitbucket Downloads.

### Bitbucket-pipelines.yml

```
1 image: maven:3.3.3
2
3 pipelines:
4   default:
5     - step:
6       script:
7         - pipe: atlassian/bitbucket-upload-file:0.1.2
8           variables:
9             BITBUCKET_USERNAME: $BITBUCKET_USERNAME
10            BITBUCKET_APP_PASSWORD: $BITBUCKET_APP_PASSWORD
11            FILENAME: "app/build/outputs/apk/debug/application-debug.apk"
```

### Outcome

You should have a bitbucket\_pipeline.yml file with the pipe configuration. When the step completes, your artifact will be uploaded to the Downloads section in your repository.

## Step 3b: Deploy your artifacts using curl and the Bitbucket REST API

With the variable and app password in place, you can now use `curl` in your build script to deploy artifacts to Bitbucket via the REST API:

```
1 curl -X POST "https://${BITBUCKET_USERNAME}:${BITBUCKET_APP_PASSWORD}@api.bitbucket.or
```

Below, there is a simple example from an Android project built with Gradle. You can make your debug Android Package (APK) available in Bitbucket Downloads.

### Bitbucket-pipelines.yml

```
1 image: maven:3.3.3
2
3 pipelines:
4   default:
5     - step:
6       script:
7         - mvn -B clean install
8         - curl -X POST --user "${BB_AUTH_STRING}" "https://api.bitbucket.org/2.0/rep
```

### Outcome

You should have a bitbucket\_pipeline.yml file with the pipe configuration. When the step completes, your artifact will be uploaded to the Downloads section in your repository.

You can check your bitbucket-pipelines.yml file with our [online validator](#).

Was this helpful?

Yes

No

[Provide feedback about this article](#)

**Still need help?**

The Atlassian Community is here for you.

[Ask the Community](#)

### Access Pipelines deployment guides

Show more

[Deploy with pull requests](#)

[Deploy using SCP](#)

- [Deploy build artifacts to Bitbucket Downloads](#)

[Publish and link your build artifacts](#)

[Build and push a Docker image to a container registry](#)

### On this page

[Step 1: Create an App password for the repository owner](#)

[Step 2: Create a Pipelines variable with the authentication token](#)

[Step 3a: Deploy your artifacts using the bitbucket-upload-file pipe](#)

[Step 3b: Deploy your artifacts using curl and the Bitbucket REST API](#)

### Community

[Questions, discussions, and articles](#)