Jira Integration

A Jira integration is required to create issues on Jira server from JFrog Pipelines steps.

Page Contents

Adding an integration

You can add this integration by following steps on the Managing Pipelines Integrations page.

Here is the information you need to create this integration:

- **Name** -- choose a friendly name for the integration
- **Url** -- Jira URL (for example, `https://jira.mycompany.com`)
- **Username** -- account username/email used for login
- **Token/Password** -- either a Jira API Tokenfor the account with permissions on the Jira REST APIs(for Jira Cloud), or the account password (for Jira Server)

Usage

The Jira integration can be added directly to a step in the `integrations` section.

The send_notification utility function can be used with a Jira integration to create an issue in Jira. For example, if a deployment fails, a ticket can be created in Jira through this integration. For more information, see Creating Jira Issues From Pipelines.

Default Environment Variables

When you add this integration to a step, a set of environment variables is automatically made available.

Environment variable	Description
`int_<integration-name>_url`	Jira API endpoint URL
`int_<integration-name>_username`	Username or email to use to login
`int_<integration-name>_token`	Authorization to use for invoking the APIs

Example

The following example pipeline demonstrates the use of the `send_notification` utility function to create a Jira issue (ticket).

- The YAML for this example is available in this repository in the JFrog GitHub account.
- In addition to the Jira integration, this example uses an Artifactory Integration.
- This example sets environment variables for the project-id and type options.
- This example performs a MvnBuild native step, which produces a log file.
- On failure, this example uses the `send_notification` utility function to create the JIra ticket. The command line:
  - Relies on the environment variables for the project-id and type options
  - Attaches the log file produced by Maven to the Jira issue
  - Specifies the summary and description options using standard environment variables

```
resources:
  - name: my_repo
    type: GitRepo
    configuration:
      gitProvider: MyGithub
      path: myrepo/myproject

pipelines:
  - name: jira_ticket_example
    configuration:
      environmentVariables:
        readOnly:
          NOTIFY_PROJECT_ID: "DEMO"            # Jira project key for all tickets we create
          NOTIFY_TYPE: "Bug"                   # Jira issue type for all tickets we create
    steps:
    ### with jira for log attachment
      - name: BuildSample
        type: MvnBuild
        configuration:
          sourceLocation: artifactory-maven-plugin-example
          configFileLocation: .
          configFileName: config
          mvnCommand: "install -P release --log-file ${step_tmp_dir}/log.txt"
          inputResources:
            - name: my_repo
          integrations:
            - name: myArtifactory
            - name: myJira
        execution:
          onFailure:
            - send_notification myJira --attach-file "$step_tmp_dir/log.txt" --description "Failure occured in $pipeline_name - $step_name" --summary "$step_name has failed"
```