

# Python3 Jira Client

clazzes.org's python3 jira-client is a python Command Line Interface(CLI) tool which can create and read issues for projects. It can be heavily customized with config files and supports both Basic Authentication and OAuth 1.0a.

## Supported Commands:

**Note:** The values needed for the commands are prioritized in this order:

Command line specified > custom config file with -c > user config file > global config file

Command	Required Parameters	Optional Parameters	Description	Requirements
listissues	--project: The project's identifier in Jira		Pretty prints, to stdout, all the issues for a given project name, in order of creation descending.  Could redirect output to save in file.	Browsing Permissions
createissue	--project: The identifier of the project to add the issue to.  --summary: Headline of the issue to be created. Put in quotation marks if it contains spaces.  --desc: Description of the issue to be created.  --issuetype: How the issue should be categorized. Examples are bug, task, feature, subtask,...  --priority: Number from 1-5 where 1 is a Blocker and 5 is Trivial.	--assignee: The username of the requested assignee, if one should be immediately assigned.  --parent: The identifier of the parent issue. Only required if you create a subtask.	Creates an issue for the given project name.	Browsing and Issue Creation Permissions
initconfig	None	--configfile: Absolute path to the file where the config values should be saved/overwritten. If it doesn't exist, attempt to create it there.  Any of username, password, oAuthToken, consumerKey, consumerSecret, oAuthSecretToken, authMethod, privateKeyLocation, baseurl that should be saved in the config file	Initialize or change a config file	None
createoauth	None	--consumerkey: The name of the OAuth public key as configured on the Jira Server. Specify to use a different one than the config file has.  --consumersecret: The OAuth secret as configured on the Jira Server. Specify to use a different one than the config file has.  --pkeyloc: Path to the private to be used for the creation of this token.	Create an OAuth token.  <b>Important:</b>  OAuth Consumer must have been configured server-side by an Administrator. A Tutorial on how to configure that can be found on the official Atlassian Jira Documentation: <a href="https://developer.atlassian.com/server/jira/platform/oauth/">https://developer.atlassian.com/server/jira/platform/oauth/</a> (Only Step 1 is relevant, Step 2 and 3 are handled by JiraClient)	Consumer secret and key already configured on Jira.  Private Key used must correspond to configured public key.

## Example Command Usage

### listissues

List issues for project with "id" attribute EXAMPLEPROJECTNAME

```
jira-client listissues --project EXAMPLEPROJECTNAME
```

Example Output:

```
EXAMPLEPROJECTNAME-2 (Bug):
Priority 3 - Heavy
DB leaks lead to security invulnerabilities
Created by John Doe at Thu, 14.06.2012 17:39:21 (UTC+02:00)
Assigned to Will Dugh
Last updated: Tue, 13.11.2012 17:26:06 (UTC+01:00)
-----
EXAMPLEPROJECTNAME-1 (Feature):
Priority 4 - Minor
Should allow filtering of currently online users
Created by Will Dugh at Wed, 14.06.2012 17:39:21 (UTC+02:00)
Assigned to Nobody
Last updated: Never
-----
```

## createissue

Create an issue for EXAMPLEHOSTINGNAME project which is a Bug with Priority Minor(4):

```
jira-client createissue --project EXAMPLEPROJECTNAME --summary "This is the headline" --desc "This is a multi-
lined\ndescription" --issuetype Bug --priority 4
```

No output when successful.

## initconfig

Initialize the user config file from scratch interactively:

```
> jira-client initconfig

Enter config file path (Alternatively enter 'u' for user or 'g' for global config):

> u

Enter remaining configurable values, leave empty to ignore:

username: > johndoe
password: > notapassword123
oAuthToken:
consumerKey: > notakey987
consumerSecret:
authMethod: > OAUTH
privateKeyLocation: > /home/johndoe/mykeyring/pkey.pem
baseurl: > https://jira.mydomain.com
```

This will set username, password, consumerKey, authMethod, privateKeyLocation and baseurl to the specified values and leave the rest alone.

Initialize a config file to a custom path with all parameters already set:

```
> jira-client initconfig --conffile /home/johndoe/customjira.conf --username johndoe --password notapw123 --
-oauthtoken Jb72NuBYBvT9zfrYfXCzdfF7y1lX6ewC -oauthsecrettoken lxpMs8QI1lXE3Y9kxGu3yakyyqHs5kh3 -auth OAUTH -
-consumersecret notasecret1923 -consumerkey notakey29 --pkeyloc /home/johndoe/mykeyring/pkey.pem --baseurl
https://jira.mydomain.com
```

No output if successful.

## createoauth

Can only be done interactively because OAuth1.0a requires manual user authentication.

Creating the OAuthToken and saving it to user config file:

```
> jira-client createoauth
Request token recieved, please authorize me in the Browser via the following URL:
https://jira.mydomain.com/plugins/servlet/oauth/authorize?oauth_token=60uUqPdN237z3qjOCafLeNWHJHO2g6zw

Have you authorized me? (Y/n)

--(Manual authentication in browser)--

> Y

{'oauth_token': 'yPpXwNyWeLpc923qIpKvMLCabVqztb28', 'oauth_token_secret': 'uDtPg6ePeKwG9JsNZZblubAWfnwbCoN1',
'oauth_expires_in': '157680000', 'oauth_session_handle': 'bmF8KVnnF2AMwUX6AAZ8P5Oc3WtS0b5A',
'oauth_authorization_expires_in': '160272000'}
Save OAuth Token to user or global config, or none (u/g/n)?
Alternatively, enter a custom absolute file path(starting with /):

> u
```

## Installation & configuration tips

### Apt sources

<http://deb.clazzes.org/any/sources.list.d/any/any-xpkg-1.list>

<http://deb.clazzes.org/any/sources.list.d/any/any-pytools-1.list>

### OAuth token creation

See <https://developer.atlassian.com/server/jira/platform/oauth/>

But not all fields in step 3 of "Create an application link" are irrelevant ;-)

### Sample cnf

#### jira-client.cnf example

```
[CLI]
username = jdoe
#password = myPass
baseurl = https://jira.company.test/

privatekeylocation=/etc/jira-client/jira_privatekey.pem
consumerkey=OAuthKey
consumersecret=secret

oauthtoken=Q.....a
auth=OAUTH
oauthsecrettoken=4...Y
```