Integration of CAST Action Plan into Atlassian JIRA

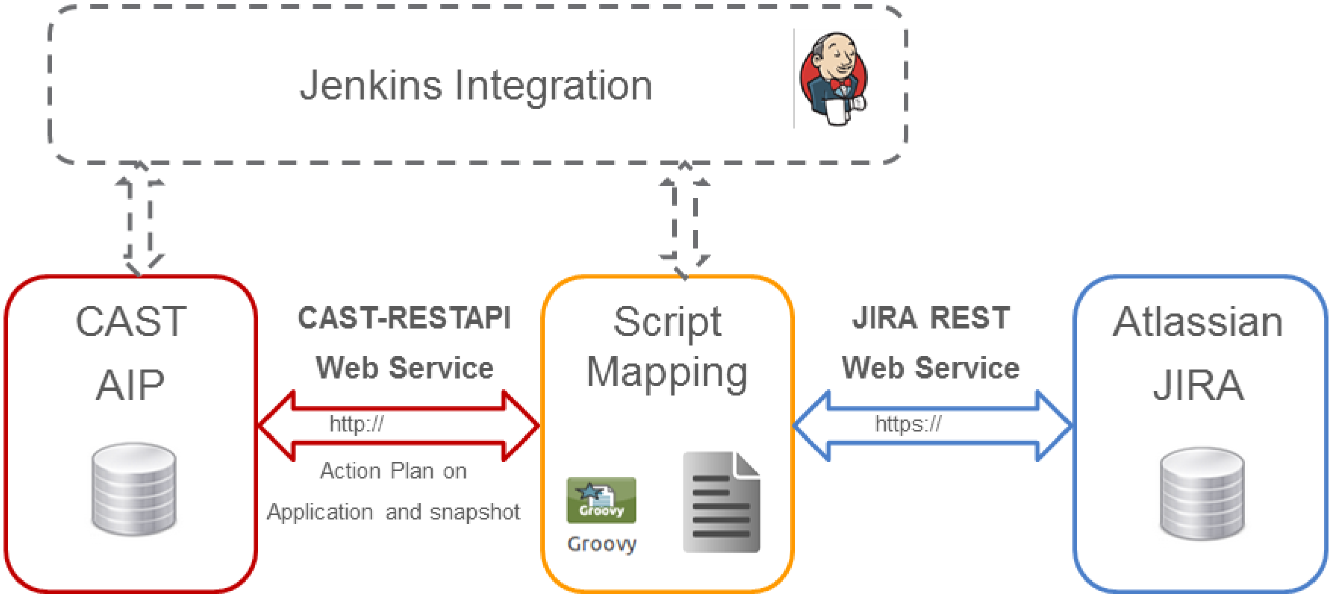
Summary: This document provides an example configuration for the Atlassian JIRA Web Service to exploit the results produced by CAST AIP via the **CAST RestAPI**. Some existing scripts has been produced to create the interaction between APIs. They need to be adapted to the context of your ownproject.

Prerequisites

✓	Access to the CAST RestAPI from the machine on which you are using the scripts.
✓	Access to Atlassian JIRA REST WebService from the machine on which you are using the scripts.
✓	Groovy should be installed on the machine on which you are using the scripts.
✓	Java security should be lowered to avoid SSL issues. Note: JAVA_HOME key in environment variables can lead to “peer not authenticated” issue.
✓	You need to have: • at least one CAST AIP snapshot stored in the CAST Dashboard Service schema (accessed via the CAST RestAPI) • an Action Plan configured with actions either via the Engineering Dashboard or via the legacy CAST Engineering Dashboard

Process Implementation

The objective is to create a script mapping between the CAST AIP environment andthe Atlassian JIRA environment. If you want to automate the process, you can potentially exploit Jenkins, however, this is beyond the scope of this document.



Here is the list of steps involved in the process of pushing violations from the CAST Action Plan into the Atlassian JIRA bug tracking system:

1. Connection to the CAST Rest API
2. Query CAST Rest API for application and snapshot selection
3. Load Action Plan

4. Query list of violations and associated information
5. Connection to Atlassian JIRA REST Web Service
6. Push violations into the JIRA system by creating a ticket

CAST Rest API Configuration

Some information is needed to be able to connect to the REST API - example information is provided below which must be adapted to your own environment:

- CASTWebServiceURL = "http://localhost:8080/CAST-RESTAPI/rest"

- CASTWebServiceLogin = "Architect"
- CASTWebServicePassword = "cast"
- CASTDomainName = "AED"
- CASTApplicationName = "Webgoat“

Atlassian JIRA Configuration

Some information is needed to be able to connect to Atlassian JIRA and to create tickets - example information is provided below which must be adapted to your own environment:

- JIRAWebServiceURL = "https://jira-company.com/rest"
- JIRAWebServiceLogin = "login"
- JIRAWebServicePassword = "password"
- JIRAProjectKey = "MYPROJECT"
- JIRAIssueType = "Bug"
- JIRAVersionId = "41940"

Groovy scripts

All scripts are available in our Github project here: https://github.com/CAST-projects/Jira-plugin. They have been tested in a custom environment but MUST be adapted to your own environment:

- **C2Jconf.groovy**: contains properties for CAST-RestAPI and Atlassian JIRA connections and information needed for requesting Action Plans and creating issues
- **C2Jmain.groovy**: contains steps to get Action Plan and create bugs in JIRA
- **C2JCASTUtils.groovy**: contains methods that get data from the CAST-RestAPI
- **C2JJIRAUtils.groovy**: contains methods that get issues from Atlassian JIRA and pushes Actions into bugs

Running the Groovy scripts

- Update **C2Jconf.groovy** with your own parameters
- Run the **C2Jmain.groovy** script