# Octane Integration Guide for Test Evidence Upload

This document outlines the steps and APIs involved in connecting to Octane, uploading test results, organizing them using application modules and suites, and linking them to JIRA.

## 1. Prerequisites

JIRA ID and Octane Mapping:

* Each SPK in Octane is associated with a `space\_id` and `workspace\_id`.
* These are used to form the API URL.
* In QZRelease2, the `space\_id` and `workspace\_id` for a given JIRA are derived using the JIRA API via a custom field.

Test Evidence Preparation:

In QZRelease2, test evidence is generated using the following Python code:  
 ReleaseTrain.configure('prod')  
 b = ReleaseTrain['core.devtools'].aliases['prod.new'].stages['int'].builds[0]  
 results = generate(b, regressionJiras, False, [], True)

## 2. Authentication to Octane Client

* Credentials are stored in Sandra Password Manager: /prod/core/qzrelease/octane/S-{space\_id}
* Use `initConfigWithAuthOnly()` in OctaneAPI class to configure authentication.
* Can be reused for any SPK by adjusting to its JIRA and credential path.

## 3. Application Module Hierarchy

### 3.1 Root Application Module

* Octane organizes evidence under Application Modules.
* QZRelease2 uses: "QzRelease-2 Octane Test Evidences" as the root.
* API: getOrCreateRootApplicationModuleID()

### 3.2 Sub-Application Module

* Created under the root using Train name.
* API: createApplicationModule()

## 4. Test Suite Structure

* Test Suites are named in YYYYQX format based on train version timestamp.
* Refer to: \_calculateEvidencePath() in OctaneClient.
* API: createTestSuite()

## 5. Suite Runs

* Suite run naming: {TrainName}-{TrainVersion}-{StageName}-{BuildVersion}
* Follow similar naming for platforms like QzPlatform.

## 6. Uploading Test Evidence

* Required inputs: rootApplicationModuleID, testSuitePath, suiteRunName, results, jira
* Upload is done via XML.
* API: upload() in OctaneClient
* Returns task ID for tracking/reporting

## 7. Linking JIRA to Uploaded Tests

* Step 1: getAutoRuns() in OctaneClient
* Step 2: updateAutoRuns() in OctaneClient

## 8. De-Linking JIRA from Uploaded Tests

* Test evidence cannot be deleted, but can be de-linked from JIRA.
* Steps:  
   1. getAutoRuns()  
   2. updateAutoRuns(delink=True)
* Note: delink is False by default, must be set True