

Service Now -Jira Integration

In this scenario, We are going to integrate Service now with Jira application. Once Incident ticket created in Service Now , generate corresponding Jira Bug in Jira application. Here, we don't want to create each and every incident as Bug in Jira, for this we will use specific criteria. Like If Category as a "Jira Issue" and Sub Category as a "Bug" in service now then only we will create that as a Bug in Jira application.

Once Bug created in Jira we will capture Jira issue id and store on incident form in new custom field for other actions like adding comments and modifying status of bug based on Service Now incident status.

A) Create Jira bug once incident created in Service Now with specific criteria:

For this we will follow below steps:

1)Create Rest Message:

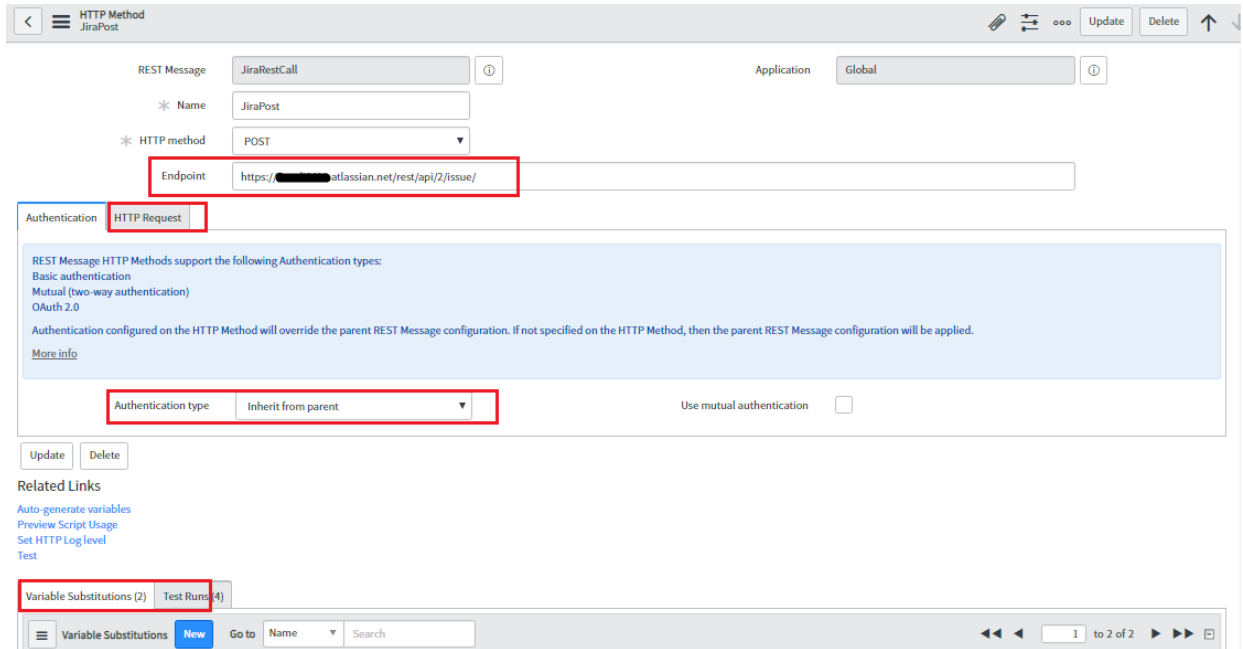
As shown in below, provide endpoint, Authentication type as basic and configure user and password in auth profile.

The screenshot shows the configuration for a REST Message named 'JiraRestCall'. The endpoint is 'https://[redacted].atlassian.net/rest/api/2/issue/'. The authentication type is set to 'Basic' and the basic auth profile is 'Authentication'. The 'HTTP Request' tab is active, showing a list of HTTP methods: 'Default GET' and 'JiraPost'.

Name	HTTP method	Endpoint
Default GET	GET	https://sonali0418.atlassian.net/rest/ap...
JiraPost	POST	https://sonali0418.atlassian.net/rest/ap...

To create bug in Jira need to configure POST method and to get existing bug details from Jira use Get Method. Here we are going to create new bug so that, we will use POST Method

As shown in below, provide endpoint, authentication type and in HTTP request pass the required headers and in content section pass the body for post method



To validate it, Click on “Test” link. If everything was correct then it gives 200 as status code and it will show output based on your HTTP method. Here we are using POST so that, it shows Jira bug id and other information in JSON format.

To make input as dynamic then refer to below screenshot

```
Content
{
  "fields": {
    "project": {
      "key": "FP"
    },
    "summary": "${description}",
    "description": "${short_description}",
    "issuetype": {
      "name": "Bug"
    }
  }
}
```

2) Create Business Rule:

Create new business rule which runs on Incident table and it triggers whenever record inserted in Incident table with defined criteria as (“Category”= “Jira Issue” AND “SubCategory”=“Bug”) and in advance section need to right script. Here we don’t need to right script just go to REST Message which was created in above step and in POST method click on Preview Script Usage and it shows script. Copy it and past it in business rule’s advance section. Just modify parameters i.e ‘param’ to current.incidentfieldname which used as dynamic input to the REST message.

The screenshot displays the configuration for a Business Rule named 'JiraRestRule'. Key settings include:

- Name:** JiraRestRule
- Table:** Incident [incident]
- Priority:** 100
- Application:** Global
- Active:**
- Advanced:** (highlighted with a red box)
- Web Services:**

 The 'Advanced' tab is active, showing:

- When:** async (highlighted with a red box)
- Order:** 100
- Insert:** (highlighted with a red box)
- Update:**
- Delete:**
- Query:**
- Filter Conditions:** Add Filter Condition (highlighted with a red box), Add *OR* Clause
- Role conditions:**

To test create incident ticket by selecting category and sub category as defined criteria once it successfully executed Jira issue id populated in custom field on incident form based on mapping in script which is written in business rule.

Note: here, REST response is in json formate you need to parse it by using JSON parser logic and store jira id in one of the field on incident form.

B) Add comments in Jira, If work notes added in incidents:

Here we will use, Jira issue id which was already populated in one of the field on incident form.

1)REST Message:

Similarly we will create REST message for adding comment in Jira. We will use POST method to create comments in related jira bug.

For this we will use API [https://JIRAInstance/rest/api/2/issue/\\${issueId}/comment](https://JIRAInstance/rest/api/2/issue/${issueId}/comment)

2)Create Business Rule:

Similarly we will create one more Business rule which will trigger when incident record is updated and based on input mapping in POST method of Rest Message corresponding comment will be created for Jira bug.

c) Modify status of Jira bug as Done once incident is closed in service now:

1) Rest Message:

To achieve this we will use Jira issue id which is already present on incident form. We will use POST method to modify status of Jira bug

For this we will use API [https://JIRAInstance/rest/api/2/issue/\\${issueid}/transitions](https://JIRAInstance/rest/api/2/issue/${issueid}/transitions)

2)Create Business Rule:

Similarly we will create one more Business rule which will trigger when incident record is updated and based on input mapping in POST method of Rest Message corresponding Jira bug status will be changed to Done.

If you have any queries or concern or you need help on this then contact us on contact@svrbeliever.com