

Get Hubspot contact information

On this page

[Prerequisites](#) | [Configuration steps](#) | [Action](#) | [Name](#) | [Method](#) | [Default connection](#) | [REST path](#) | [Action body](#) | [Update issue fields](#) | [Related examples](#)

Use case

Get the **contact information** about the reporter of the issue from your **Hubspot** account and copy it into a Text Field (multi-line).

Prerequisites

Hubspot private app

1

Create a **Hubspot private app** in your Hubspot account.

Visit <https://developers.hubspot.com/docs/api/private-apps> for more information.

2

Select the following **scope** for the app.

```
crm.objects.contacts.read
```

3

Store the **access token** for future usage in the *Execute remote action* post function.

Connection with Hubspot

1

In Jira administration > Manage apps > Remote actions > Connections, **create a new connection** and name it.

Set the following **base URL**.

2

`https://api.hubapi.com`

3

Select the **Bearer authentication method** and enter the **access token** previously stored.

Configuration steps

1

Add the [Execute remote action](#) **post function** to the desired transition.

2

Action

In **Expert mode**, create a **new action** with the following parameters:

1

Name

Enter **Get Hubspot contact information** as the name of the new action.

2

Method

Select the **POST** method.

3

Default connection

Select the **connection with Hubspot**.

4

REST path.

Use the following **REST path**.

`/crm/v3/objects/contacts/search`

3

Action body

Enter an **action body** like the following one.

This action body requests the **company, address, city, phone number, mobile phone number and email** from the Hubspot contact information of the reporter but it can be modified to satisfy your requirements.

Note that the reporter should have **set the visibility of his email address to public** to be able to obtain it.

```
{
  "filterGroups": [
    {
      "filters": [
        {
          "values": [
            "emailmaria@hubspot.com",
            "%{issue.reporter.email}"
          ],
          "propertyName": "email",
          "operator": "IN"
        }
      ]
    }
  ],
  "properties": [
    "company",
    "address",
    "city",
    "phone",
    "mobilephone",
    "email"
  ],
  "limit": "10",
  "after": "1"
}
```

4

Update issue fields

1

Update a **Text Field (multi-line)**.

2

Select the option **Set field value manually (parser expression)**.

3

Enter the following expression with the expression editor in **General** mode.

```

%{"*Contact information*" + "







"+
"
" +
(%{action.response.results[].properties[].company} != "" ? "|*Company*" + %{
action.response.results[].properties[].company} + "|" + "
" : "") +
(%{action.response.results[].properties[].address} != "" ? "|*Address*" + %{
action.response.results[].properties[].address} + "|" + "
" : "") +
(%{action.response.results[].properties[].city} != "" ? "|*City*" + %{action.
response.results[].properties[].city} + "|" + "
" : "") +
(%{action.response.results[].properties[].phone} != "" ? "|*Telephone
number*" + %{action.response.results[].properties[].phone} + "|" + "
" : "") +
(%{action.response.results[].properties[].mobilephone} != "" ? "|*Mobile*" + %{
action.response.results[].properties[].mobilephone} + "|" + "
" : "") +
(%{action.response.results[].properties[].email} != "" ? "|*Email*" + %{
action.response.results[].properties[].email} + "|" + "
" : "")}













```

If the action body was edited, this expression should also be adjusted to the new request.



Related examples

Use case	JWT feature	Workflow function	Parser functions	Complexity
Automatically create a version when starting the release		Execute remote action		
Automatically link an issue to an external one		Execute remote action		
Automatically log work on a Jira issue		Execute remote action	dateTimeToString()	
Create a comment on an external Jira ticket		Execute remote action		
Create a new employee account during an onboarding process		Execute remote action		
Create an external project for a new employee during an onboarding process		Execute remote action		

Create an overview page for a software release		Execute remote action	
Create a personal space for a new employee		Execute remote action	
Get Hubspot contact information		Execute remote action	
Link a Jira issue with the corresponding release ticket		Execute remote action	
Link issue to issue mentioned in its description		Execute remote action	findPattern() first()
Notify the reporter of an issue about its status by a Telegram message		Execute remote action	
Read the information from a Trello card		Execute remote action	
Retrieve the assets of an issue in Jira cloud		Execute remote action	
Set the assignee of an external issue same as the transitioned issue		Execute remote action	
Set User Picker field with users from group		Execute remote action	
Transition an external Jira ticket based on the linked internal one		Execute remote action	
Translate the description		Execute remote action	findReplaceAll()

If you still have questions, feel free to refer to our [support team](#).