

SFTP Integration usage instructions

SFTP Connection info:

You can use this public SFTP endpoint to do a basic test:

<https://www.sftp.net/public-online-sftp-servers>

Or create an SFTP connection to your own SFTP server.

- 1) Create the SFTP Connection
- 2) Create two new integrations
 - a) Create another one called sftp-get-file
 - b) Upload the .json that has the SFTP integration sample from:
 - i) sftp-get-file-v8.json
- 3) Create a second new integration
 - a) Create one called sftp-poll-path-for-file
 - b) Upload the .json that has the SFTP integration sample from:
 - i) sftp-poll-path-for-file-v14.json
- 4) Configure the Connector task to use the connection in your project
 - a) In the “**sftp-poll-path-for-file**” integration:
 - b) Click on the Connector task called : “SFTP Connector List Files”
 - c) Click on “Configure Task”
 - d) Click on the Reset button
 - e) Select the region where your SFTP Connection is running
 - f) Select the SFTP connection from the list
 - g) Select Entities for the Type
 - h) Select ROOT
 - i) Select LIST for the Operation
 - j) Click Done

Configure connector task

Choose the connection and settings for your task.

Select a region
us-east1

Required before choosing a connection

Connection Add a filter	Type	Entity Add a filter	Operation
pubsub-company-data ⓘ	Entities	Root	List
pubsub-api-hub-approval ⓘ	Actions	pub	Get
pubsub ⓘ			Create
cloudsql-mysql ⓘ			Delete
bq-public-covid19 ⓘ			Update

- 5) Adjust the Filter variable mapping in the task before this one
 - a) Set List Files ConnectorTask filterClause variable - data mapper task
 - b) Use this variable to specify the path on the SFTP server and any filename pattern to scan for new files.

- c) Change the filter expression to work for your use case using an SQL Where clause syntax. See the sample LIST output below. You can use any of the fields from that output in your filter criteria.

← sftp-poll-path-for-file Version 14 Database Persistence Edited 1 hour ago by shaaland@google.com Region: us-east1

Data Mapping Editor

[Learn more](#) about Data Mapping. For a detailed guide to transform functions, see the [functions reference](#).

Variables

Inputs

> STR RemoteFileName

Outputs

Input

STR "Filepath LIKE '/pub/example/%'"

Output

STR filterClause (SFTP Connector List Files)

- 6) Adjust the Connector Tasks in the **"sftp-get-file"** sub integration
- In the Task : SFTP Connector Read File
 - Change to your region / connection
 - Choose Actions / Download

Configure connector task

Choose the connection and settings for your task.

Select a region
us-east1

Required before choosing a connection

Connection Add a filter	Type	Action Add a filter
sftp-dynolab ⓘ	Entities	Download
	Actions	

- In the Task : Get File Details
 - Change to your region / connection
 - Choose Entities / ROOT / GET

Configure connector task

Choose the connection and settings for your task.

Select a region
us-east1

Required before choosing a connection

Connection <a>Add a filter	Type	Entity <a>Add a filter	Operation
sftp-dynolab ⓘ	Entities	Root	List
	Actions		Get
			Create
			Delete
			Update

- 7) Test the integration - it will prompt to publish, etc...
- a) The only file downloadable on the rebex public server is "readme.txt", but if you are connected to a different server, try one of the other file names on that server
 - i) Only Text files are supported at this time. Binary files will fail to download. This is on the roadmap.
 - ii) If you examine the log files for sftp-poll-path-for-file integration, you will see that it loops through all the files in that folder that you provided as input, and it will attempt to download each one. The contents of the file are put into the "Contents" property in the ConnectorOutputPayload.

Viewing: loopMetadata
















```
{
  "Current Element": "/pub/example/WinFormClientSmall.png",
  "Current Iteration Count": 16,
  "Sub-integration execution IDs": [
    "ab4ffd5f-8635-44ed-89d6-cfe66b266222",
    "3ea03d69-17fd-4f6c-b509-572c40886c76",
    "e505dacb-f62b-4062-9ab3-1e7590747714",
    "caa32121-ccb8-4d3c-99de-cd18f0309003",
    "4666f3af-a4a0-46f0-a5c0-93327d5e4925",
    "489ce981-5d66-44d6-a922-40487a78f175",
    "27f9b141-fc36-4639-8dd7-5a5d7cb83c10",
    "d2995015-894f-473f-b8c7-471d3d552fb9",
    "5115cd69-30e8-4b65-9a12-791407985c66",
    "b108fc5f-c1ab-415e-8c05-953bfc7bd35a",
    "f79d5a37-945f-4a34-b7f1-f7304956aca6",
    "9bfd382a-ef04-48d9-ba7a-8cfdfb15c7c7",
    "cfa00990-a296-48ab-8a60-6a9148c5179f",
    "84e29a9b-446f-4f23-93f9-3ee9fb8073c5",
    "e74bf4e0-c909-44f9-9819-95aab49fe161",
    "1d3e3242-f61b-4529-9af7-c0fa9fb54818"
  ]
}
```

CLOSE

- iii) You can see the individual file processing in the sftp-get-file logs. Note that these all had timestamps in the 1:55 PM range:

Integration name = sftp-get-file								Add a filter	
Integration Name	Execution ID	Status	Start time	Duration	Integration Version	Trigger ID			
> sftp-get-file	1d3e3242-f61b-4529-9af7-c0fa9fb54818	✔ Succeeded	Aug 24, 2022, 1:55:50 PM	2.182 s	8	api_trigger/sftp-get-file_API_1			
> sftp-get-file	e74bf4e0-c909-49f7-9819-95aab49fe161	✔ Succeeded	Aug 24, 2022, 1:55:47 PM	2.601 s	8	api_trigger/sftp-get-file_API_1			
> sftp-get-file	84e29a9b-446f-4f23-93f9-3ee9fb8073c5	✔ Succeeded	Aug 24, 2022, 1:55:45 PM	2.079 s	8	api_trigger/sftp-get-file_API_1			
> sftp-get-file	cfa00990-a296-48ab-8a60-6a9148c5179f	✔ Succeeded	Aug 24, 2022, 1:55:42 PM	2.329 s	8	api_trigger/sftp-get-file_API_1			
> sftp-get-file	9bfd382a-ef04-48d9-ba7a-8cfdfb15c7c7	✔ Succeeded	Aug 24, 2022, 1:55:40 PM	2.027 s	8	api_trigger/sftp-get-file_API_1			
> sftp-get-file	f79d5a37-945f-4a34-b7f1-f7304956aca6	✔ Succeeded	Aug 24, 2022, 1:55:32 PM	7.071 s	8	api_trigger/sftp-get-file_API_1			
> sftp-get-file	b108fc5f-c1ab-415e-8c05-953bfc7bd35a	✔ Succeeded	Aug 24, 2022, 1:55:30 PM	2.023 s	8	api_trigger/sftp-get-file_API_1			
> sftp-get-file	5115cd69-30e8-4b65-9a12-791407985c66	✔ Succeeded	Aug 24, 2022, 1:55:28 PM	2.329 s	8	api_trigger/sftp-get-file_API_1			
> sftp-get-file	d2995015-894f-473f-b8c7-471d3d552fb9	✔ Succeeded	Aug 24, 2022, 1:55:25 PM	2.151 s	8	api_trigger/sftp-get-file_API_1			
> sftp-get-file	27f9b141-fc36-4639-8dd7-5a5d7cb83c10	✔ Succeeded	Aug 24, 2022, 1:55:22 PM	2.515 s	8	api_trigger/sftp-get-file_API_1			

v) And here is the one text file that succeeded:

>	 Connectors (Task ID: 1) - Attempt 1	 Succeeded
>	 Data Mapping (Task ID: 3) - Attempt 1	 Succeeded
>	 Data Mapping (Task ID: 5) - Attempt 1	 Succeeded
▼	 Connectors (Task ID: 6) - Attempt 1	 Succeeded
	entityId Task 6: GenericConnectorTask	/pub/example/readme.txt
	FilePathSplit	pub example readme.txt
	FileContents	Welcome. You are connected to an FTP or SFTP server used for. ...
	connectorOutputPayload Task 6: GenericConnectorTask	{ "FilePath": "/pub/example/readme.txt", "Filename": "readme.txt", "FileSize": 405.0, "LastModified": "2021-12-17 10:58: ...
	RemoteFileName	/pub/example/readme.txt
	EndResult	{ "EndResult": "true", "RemoteFilename": "/pub/example/readme.txt" }
	Success	true
>	Response Parameters	

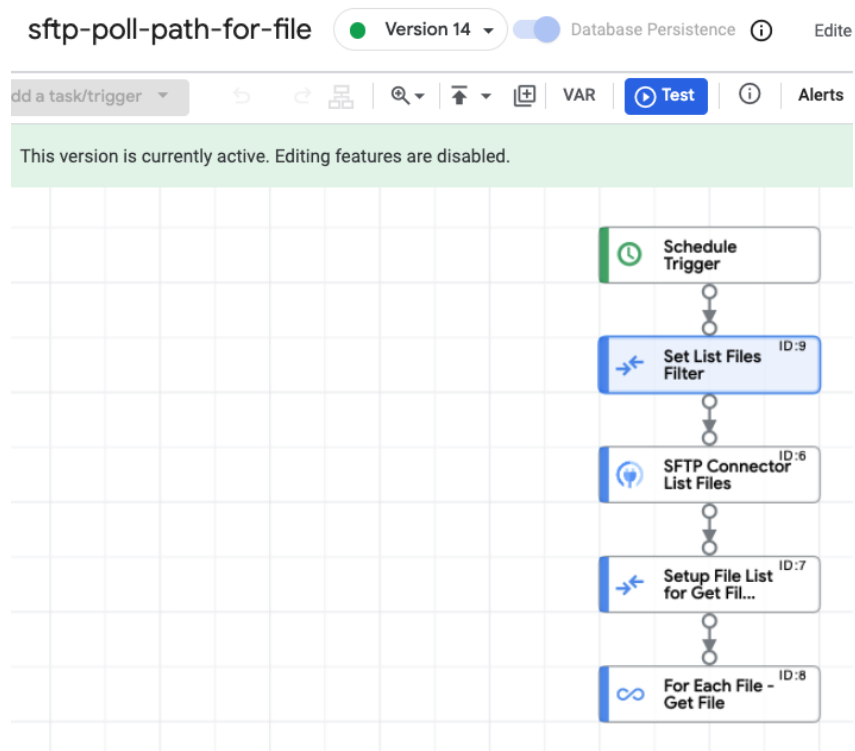
And here are the contents from the connectorOutputPayload:

Viewing: connectorOutputPayload

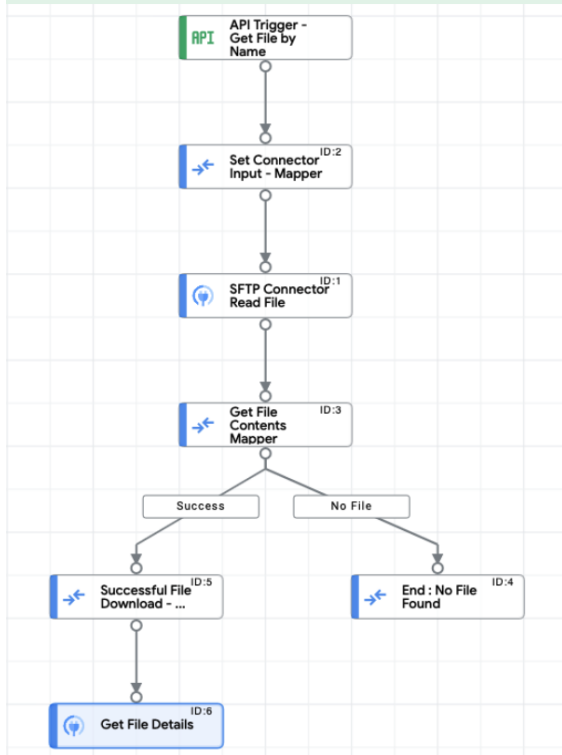
```
[{
  "Success": "true"
}, {
  "Content": "Welcome,\n\nYou are connected to an FTP or SFTP server used for testing purposes by Rebex FTP/SSL or Rebex SFTP sample code.\n\nOnly read access is allowed and the FTP download speed is limited to 16KBps.\n\nFor information about Rebex FTP/SSL, Rebex SFTP and other Rebex .NET components, please visit our website at https://www.rebex.net/\n\nFor feedback and support, contact support@rebex.net\n\nThanks!\n"
}]
```

CLOSE

Here is what the main integration (sftp-poll-path-for-file) looks like:



This is the sub integration (sftp-get-file) which is called in the For Each task:



SFTP Connector - reference:

LIST. Do a file list so that you can see all the files in a directory. Choose the Entity for the directory you want (like ROOT), and then set the Filter input variable for the connector task to put in a search criteria. The filter syntax is the same as an SQL Where clause. Set the Filter variable to something like this:

```
FilePath LIKE '/path/subpath/%.csv' AND FileSize > 0
```

This is what an output from the LIST might look like :

```
[{
  "FilePath": "/pub/example/imap-console-client.png",
  "Filename": "imap-console-client.png",
  "FileSize": 19156.0,
  "LastModified": "2007-02-16 17:00:28.0",
  "IsDirectory": false,
  "Permissions": "-rw-----",
  "Owner": "demo",
  "OwnerId": "",
  "Group": "users",
  "GroupId": "",
  "LocalFile": null
}, {
  "FilePath": "/pub/example/KeyGenerator.png",
  "Filename": "KeyGenerator.png",
  "FileSize": 36672.0,
  "LastModified": "2007-03-19 19:52:58.0",
  "IsDirectory": false,
  "Permissions": "-rw-----",
  "Owner": "demo",
  "OwnerId": "",
  "Group": "users",
  "GroupId": "",
  "LocalFile": null
}, {
  "FilePath": "/pub/example/KeyGeneratorSmall.png",
  ...
  continued...
  ...
```



```
  ]}
```

GET. Choose the Entity option, and then do a GET Operation. The entityID is the FilePath including the file name. This is the ConnectorOutputPayload when I do a GET for entityID: "pub/example/readme.txt"

```
{
  "FilePath": "/pub/example/readme.txt",
  "Filename": "readme.txt",
  "FileSize": 405.0,
  "LastModified": "2021-12-17 10:58:48.0",
  "IsDirectory": false,
  "Permissions": "-rw-----",
  "Owner": "demo",
  "OwnerId": "",
  "Group": "users",
  "GroupId": "",
  "LocalFile": null
}
```

Download a File. Select Action in the Connector Task, and then Download. Pass in the full path file name as input. The first element that is returned is a Success status, and the second element is the contents of the file. *NOTE: Only text files are supported for now. Binary files will throw an exception.*

Set the ConnectorInputPayload to set the JSON property RemoteFile:

```
{
  "RemoteFile": "pub/example/readme.txt"
}
```

This is the ConnectorOutputPayload for the download action for a readme file :

```
[{
  "Success": "true"
}, {
  "Content": "Welcome,\r\n\r\nYou are connected to an FTP or SFTP server  
used for testing purposes by Rebex FTP/SSL or Rebex SFTP sample  
code.\r\nOnly read access is allowed and the FTP download speed is  
limited to 16KBps.\r\n\r\nFor information about Rebex FTP/SSL, Rebex SFTP  
and other Rebex .NET components, please visit our website at  
https://www.rebex.net/\r\n\r\nFor feedback and support, contact  
support@rebex.net\r\n\r\nThanks!\r\n"
}]
```

Upload a File. Select Action in the Connector Task, and then Upload. Pass in Content and RemoteFile in the JSON for the ConnectorInputPayload variable.

Set the ConnectorInputPayload variable to something like this using the Data mapper:

```
[{
  "Content": "Welcome,\r\n\r\nYou are connected to an FTP or SFTP server
used for testing purposes by Rebex FTP/SSL or Rebex SFTP sample
code.\r\nOnly read access is allowed and the FTP download speed is
limited to 16KBps.\r\n\r\nFor information about Rebex FTP/SSL, Rebex SFTP
and other Rebex .NET components, please visit our website at
https://www.rebex.net/\r\n\r\nFor feedback and support, contact
support@rebex.net\r\n\r\nThanks!\r\n"
}, {
  "RemoteFile": "pub/example/myNewFile.csv"
}]
```