

fn_google_cloud_dlp

Table of Contents

- [Release Notes](#)
 - [Overview](#)
 - [Key Features](#)
 - [Requirements](#)
 - [Resilient platform](#)
 - [Cloud Pak for Security](#)
 - [Authenticating to Google Cloud](#)
 - [Proxy Server](#)
 - [Python Environment](#)
 - [Installation](#)
 - [Install](#)
 - [App Configuration](#)
 - [Function - Google Cloud DLP: De-Identify Content](#)
 - [Function - Google Cloud DLP: Inspect Content](#)
 - [Rules](#)
 - [Troubleshooting & Support](#)
-

Release Notes

| Version | Date | Notes |
|---------|---------|------------------------|
| 1.0.0 | 06/2019 | Initial Release |
| 1.1.0 | 09/2021 | Added App Host Support |

Overview

Resilient Circuits Components for 'fn_google_cloud_dlp'

Customization Settings

Layouts Rules Scripts **Workflows** Functions Message Destinations Phases & Tasks Incident Types Breach Artifacts

Workflows

New Workflow

Search...

| Workflow Name | Description | Object Type | Rules |
|--|--|-------------|--|
| Example: Google Cloud DLP - De-Identify Artifact | An example workflow ran at an attachment level that sends the artifact data to Google Cloud's DLP Service and aims to de-identify the types of personal information specified. By default 14 types are selected out of 50+ types. The return result is a new artifact with the PII removed. | Artifact | Example: Google Cloud - Remove PII from String |
| Example: Google Cloud DLP - De-Identify Attachment | An example workflow ran at an attachment level that sends the attachment data to Google Cloud's DLP Service and aims to de-identify the types of personal information specified. By default 14 types are selected out of 50+ types. The return result is a new attachment with the PII removed. | Attachment | Example: Google Cloud - Remove PII from Attachment |
| Example: Google Cloud DLP - Inspect Attachment for PII | An example workflow ran at an attachment level that sends the attachment data to Google Cloud's DLP Service and aims to de-identify the types of personal information specified. By default 14 types are selected out of 50+ types. Returned results include a list of findings generated from the input including the finding itself, what type of info it was matched against and the likelihood that the match is accurate. | Attachment | Example: Google Cloud - Inspect Attachment for PII |

The Resilient Integration with Google Cloud DLP provides tools to integrate into your Incident Response Plan. The integration brings Automation and Orchestration capabilities for either identifying, redacting or de-identifying Personally identifiable information (PII) in a body of text.

Key Features

- Inspect a text-based attachment for Personal Identifiable Information
- Search for and redact Personal Identifiable Information from an attachment or artifact

Requirements

This app supports the IBM Resilient SOAR Platform and the IBM Cloud Pak for Security.

Resilient platform

The Resilient platform supports two app deployment mechanisms, App Host and integration server.

If deploying to a Resilient platform with an App Host, the requirements are:

- Resilient platform \geq 40.0.6554.
- The app is in a container-based format (available from the AppExchange as a zip file).

If deploying to a Resilient platform with an integration server, the requirements are:

- Resilient platform \geq 40.0.6554.
- The app is in the older integration format (available from the AppExchange as a zip file which contains a tar.gz file).
- Integration server is running resilient_circuits \geq 30.0.0.
- If using an API key account, make sure the account provides the following minimum permissions:

| Name | Permissions |
|----------|-------------|
| Org Data | Read |

| Name | Permissions |
|----------|-------------|
| Function | Read |

The following Resilient platform guides provide additional information:

- *App Host Deployment Guide*: provides installation, configuration, and troubleshooting information, including proxy server settings.
- *Integration Server Guide*: provides installation, configuration, and troubleshooting information, including proxy server settings.
- *System Administrator Guide*: provides the procedure to install, configure and deploy apps.

The above guides are available on the IBM Knowledge Center at ibm.biz/resilient-docs. On this web page, select your Resilient platform version. On the follow-on page, you can find the *App Host Deployment Guide* or *Integration Server Guide* by expanding **Resilient Apps** in the Table of Contents pane. The System Administrator Guide is available by expanding **System Administrator**.

Cloud Pak for Security

If you are deploying to IBM Cloud Pak for Security, the requirements are:

- IBM Cloud Pak for Security >= 1.4.
- Cloud Pak is configured with an App Host.
- The app is in a container-based format (available from the AppExchange as a [zip](#) file).

The following Cloud Pak guides provide additional information:

- *App Host Deployment Guide*: provides installation, configuration, and troubleshooting information, including proxy server settings. From the Table of Contents, select Case Management and Orchestration & Automation > **Orchestration and Automation Apps**.
- *System Administrator Guide*: provides information to install, configure, and deploy apps. From the IBM Cloud Pak for Security Knowledge Center table of contents, select Case Management and Orchestration & Automation > **System administrator**.

These guides are available on the IBM Knowledge Center at ibm.biz/cp4s-docs. From this web page, select your IBM Cloud Pak for Security version. From the version-specific Knowledge Center page, select Case Management and Orchestration & Automation.

Authenticating to Google Cloud

Google Cloud requires an environment variable named `GOOGLE_APPLICATION_CREDENTIALS` in order to authenticate. To get the contents of this variable, you need to create a new service account with the **DLP User** permission in the service accounts tab. Then, under the actions column of the service accounts, select the Manage keys option and create a new key that is a JSON type.

Using an Integration Server: You will need to use the export command in terminal using the path to the json file and with whatever name you'd like.

```
export
GOOGLE_APPLICATION_CREDENTIALS="/Path/to/json/whatever_name_you_want.json"
```

Using App Host: When configuring the app after installing, you must create a new file in the "Configuration" tab. Name the file "service_account_key.json" and have the path be "/var/rescircuits". Paste in the contents of the json file here.

App Settings / service_account_key.json

Edit the settings in the file below. File Path specifies a directory path starting at root. If changing location, the system creates the directory if it does not exist. When done, click Save and Push Changes to implement your changes and restart the app.

Cancel

Save and Push Changes

Created Date: 09/29/2021 15:29

Last Modified Date: 09/29/2021 15:29

File Name

service_account_key.json

File Path

/var/rescircuits

File Description

Purpose of the file.

Show more

File Content

Text or code as appropriate.

Theme dark

File Type Plain Text

```

1 {
2   "type": "service_account",
3   "project_id": "ibm-test-325618",
4   "private_key_id": "*****",
5   "private_key": "-----BEGIN PRIVATE KEY-----*****",
6   "client_email": "ibm-test@ibm-test-325618.iam.gserviceaccount.com",
7   "client_id": "117904021218483152186",
8   "auth_uri": "https://accounts.google.com/o/oauth2/auth",
9   "token_uri": "https://oauth2.googleapis.com/token",
10  "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
11  "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/x509/ibm-test%40ibm-test-325618.iam.gserviceaccount.com"
12 }

```

Proxy Server

The app does support a proxy server.

Python Environment

Both Python 2.7 and Python 3.6 are supported. Additional package dependencies may exist for each of these packages:

- defusedxml
- google-cloud-dlp>=0.10.0,<2.0.0
- PyPDF2>=1.26.0
- python-docx>=0.8.10
- resilient-lib>=32.0.140
- resilient_circuits>=30.0.0

Installation

Install

- To install or uninstall an App or Integration on the *Resilient platform*, see the documentation at ibm.biz/resilient-docs.

- To install or uninstall an App on *IBM Cloud Pak for Security*, see the documentation at ibm.biz/cp4s-docs and follow the instructions above to navigate to Orchestration and Automation.

App Configuration

The following table provides the settings you need to configure the app. These settings are made in the app.config file. See the documentation discussed in the Requirements section for the procedure.

| Config | Required | Example | Description |
|-----------------------------|----------|--------------------------|---|
| gcp_project | Yes | <YOUR_GOOGLE_PROJECT_ID> | Enter a description of the config here. |
| gcp_dlp_masking_char | Yes | # | Enter a description of the config here. |

Function - Google Cloud DLP: De-Identify Content

None

The screenshot shows the Resilient Orchestration and Automation console. The top navigation bar includes 'Dashboards', 'Simulations', 'Incidents', and a 'Create' button. The user is logged in as Alfred Pennyworth from WayneCorp. The 'Customization Settings' section is active, with tabs for 'Layouts', 'Rules', 'Scripts', 'Workflows', 'Functions', 'Message Destinations', 'Phases & Tasks', 'Incident Types', 'Breach', and 'Artifacts'. The 'Workflows' tab is selected, showing a workflow named 'Example: Google Cloud DLP - De-Identify Attachment'. The workflow details include:

- Name:** Example: Google Cloud DLP - De-Identify Attachment
- API Name:** gcp_dlp_deidentify_attachment
- Description:** An example workflow ran at an attachment level that sends the attachment data to Google Cloud's DLP Service and aims to de-identify the types of personal information specified. By default 14 types are selected out of 50+.
- Object Type:** Attachment
- Creator:** Alfred Pennyworth
- Last Modified:** 17/06/2019 11:34
- Last Modified By:** Alfred Pennyworth
- Associated Rules:** Example: Google Cloud - Remove PII from Attachment

The workflow diagram shows a sequence of steps: a start node (circle with a plus sign) labeled 'Start your workflow here', followed by a function node labeled 'Google Cloud DLP: De-Identify Con...', and ending with a final node (circle with a checkmark).

► Inputs:

| Name | Type | Required | Example | Tooltip |
|---------------|--------|----------|---------|---------|
| artifact_id | number | No | — | — |
| attachment_id | number | No | — | — |

| Name | Type | Required | Example | Tooltip |
|---------------------------------|--------------------------|----------|---------|---|
| <code>gcp_artifact_input</code> | <code>text</code> | No | — | A optional input to be used when the function is ran from an artifact and is used to capture the artifacts value. |
| <code>gcp_dlp_info_types</code> | <code>multiselect</code> | No | — | Which types of PII do you want to de-identify. |
| <code>incident_id</code> | <code>number</code> | Yes | — | — |
| <code>task_id</code> | <code>number</code> | No | — | — |

► Outputs:

```
{'version': '1.0', 'success': True, 'reason': None, 'content': {'de_identified_text':
'\uffeffSSN,gender,birthdat...#####'}, 'raw': '{"de_identified_text...#####}', 'inputs':
{'gcp_dlp_info_types': [...], 'incident_id': 2114, 'attachment_id': 22}, 'metrics': {'version': '1.0', 'package':
'fn-google-cloud-dlp', 'package_version': '1.1.0', 'host': [hostname], 'execution_time_ms': 2740086,
'timestamp': '2021-09-28 17:34:07'}}
```

► Example Pre-Process Script:

```
inputs.incident_id = incident.id
if artifact.type == "string":
    inputs.gcp_artifact_input = artifact.value
else:
    inputs.artifact_id = artifact.id
```

► Example Post-Process Script:

```
"""
If the integration was successful in operation, upload a new artifact
containing the now de-identified text.
"""
if results.success:
    incident.addNote(u""""De-Identified using Google Cloud DLP<b>
{}"""".format(results.content["de_identified_text"]))
```

Function - Google Cloud DLP: Inspect Content

None

Resilient Dashboards Simulations Incidents **Create** Alfred Penny... WayneCorp

Customization Settings

Layouts Rules Scripts **Workflows** Functions Message Destinations Phases & Tasks Incident Types Breach Artifacts

[Workflows](#) / Example: Google Cloud DLP - Inspect Attachment for PII Cancel Save & Close Save

Name * Example: Google Cloud DLP - Inspect Attachment for PII

API Name * gcp_dlp_inspect_attachment

Description An example workflow ran at an attachment level that sends the attachment data to Google Cloud's DLP Service and aims to de-identify the types of personal information specified. By default 14 types are selected out of 50+.

Object Type * Attachment

Creator Alfred Pennyworth
Last Modified 17/06/2019 11:48
Last Modified By Alfred Pennyworth
Associated Rules Example: Google Cloud - Inspect Attachment for PII

► Inputs:

| Name | Type | Required | Example | Tooltip |
|--------------------|-------------|----------|---------|---|
| artifact_id | number | No | — | — |
| attachment_id | number | No | — | — |
| gcp_artifact_input | text | No | — | A optional input to be used when the function is ran from an artifact and is used to capture the artifacts value. |
| gcp_dlp_info_types | multiselect | No | — | Which types of PII do you want to de-identify. |
| incident_id | number | Yes | — | — |
| task_id | number | No | — | — |

► Outputs:

```
{'version': '1.0', 'success': True, 'reason': None, 'content': {'findings': [...], 'attachment_name': '[PII Removed]sample-...ta.csv.txt'}, 'raw': '{"findings": [], "at....csv.txt"}', 'inputs': {'gcp_dlp_info_types': [...], 'incident_id': 2114, 'attachment_id': 38}, 'metrics': {'version': '1.0', 'package': 'fn-google-cloud-dlp', 'package_version': '1.1.0', 'host': [hostname], 'execution_time_ms': 11913, 'timestamp': '2021-09-28 19:03:18'}}
```

► Example Pre-Process Script:

```

inputs.incident_id = incident.id

# If this workflow has the task_id available, gather it incase we need it.
if task:
    inputs.task_id = task.id
# If this workflow has the attachment_id available, gather it incase we
need it.
if attachment:
    inputs.attachment_id = attachment.id

# If this workflow has the artifact_id available, gather it incase we need
it.
try:
    if artifact:
        inputs.artifact_id = artifact.id
except:
    pass

```

► Example Post-Process Script:

```

if results.success:
    """Print all the findings as a richtext note. This note may be very long
if you run the integration on a large file with lots of PII. In these
cases you may want to limit how many findings are put into the note."""
    if results.content['findings'] != None:
        note_text = u"""Findings were found from attachment <b>{}</b><br><br>
Findings: <br>""".format(results.content["attachment_name"])
        for finding in results.content['findings']:
            note_text += u"""Text Quote: <b>{}</b>
                        <br> Information Type Suspected: <b>{}</b>
                        <br> Likelihood / Confidence: <b>{}</b><br>
<br>""".format(finding["quote"], finding["info_type"], finding["likelihood"])

        incident.addNote(helper.createRichText(note_text))

```

Rules

| Rule Name | Object | Workflow Triggered |
|--|------------|---|
| Example: Google Cloud - Remove PII from String | artifact | gcp_dlp_deidentify_artifact |
| Example: Google Cloud - Inspect Attachment for PII | attachment | gcp_dlp_inspect_attachment |
| Example: Google Cloud - Remove PII from Attachment | attachment | gcp_dlp_deidentify_attachment |

Troubleshooting & Support

Refer to the documentation listed in the Requirements section for troubleshooting information.

For Support

This is a IBM Community provided App. Please search the Community <https://ibm.biz/resilientcommunity> for assistance.