

Integrations

CSVEnrichmentSource For Director Console API

V5.2.0

CONTENTS

1	CSV Enrichment Source	1
2	Installing CSV Enrichment Source	2
2.1	Uploading CSV Enrichment Source to the Fabric Storage	2
2.2	Installing CSV Enrichment Source	3
3	Configuring CSV Enrichment Source	5
3.1	Uploading a CSV File in the Fabric Storage	5
3.2	Configuring CSV Enrichment Source Using a CSV File	6
3.3	Configuring CSV Enrichment Source Using a URL	9
3.4	Fetching CSV Enrichment Source Sample Data	13
3.5	Listing the CSV Enrichment Source Configurations	15
3.6	Getting CSV Enrichment Source Configurations by ID	17
3.7	Removing CSV Enrichment Source Configurations by ID	18
3.8	Editing the CSV Enrichment Source Configuration Using a CSV File	19
3.9	Editing the CSV Enrichment Source Configuration Using a URL	21
3.10	Deleting the CSV Enrichment Source Configuration	24
3.11	Deleting the CSV File in the Fabric Storage	25
4	Uninstalling CSV Enrichment Source	27
5	CSV Enrichment Source - RefreshList	28
5.1	Refreshing the list for CSV Enrichment Source Collections	28
6	CSV Enrichment Source - Trash	29
6.1	Deleting CSV Enrichment Source from the Fabric Storage	29

CSV ENRICHMENT SOURCE

CSV Enrichment Source enables you to use a CSV file as an enrichment source in Logpoint. It fetches data feeds from a CSV file and enriches search results with the data. You can either upload a CSV file or provide a URL link to a CSV file.

It supports the following types of CSV files:

1. Tab-separated
2. Comma-separated
3. Space-separated
4. Semicolon-separated

INSTALLING CSV ENRICHMENT SOURCE

2.1 Uploading CSV Enrichment Source to the Fabric Storage

2.1.1 Private Location

You can upload CSV Enrichment Source to a private location in the Fabric Storage using the **Upload - Upload** API.

Endpoint URL:

`https://api-server-host-name/configapi/Uploads/{pool_UUID}/`

Method:

`POST`

Header:

FIELD	DESCRIPTION
file_name	Name of the CSV Enrichment Source pak file.
Content-Type	Content type of the CSV Enrichment Source pak file. It must be <i>(application/octet-stream)</i> .

Parameters:

FIELD	TYPE	DESCRIPTION	REQUIRED
file	[Object]	The CSV Enrichment Source pak file to be uploaded.	Mandatory

Success Response:

```
{
  "status": "Success",
  "message": "CSVEnrichmentSource_5.2.0.pak successfully uploaded in private storage."
}
```

2.1.2 Public Location

You can upload CSV Enrichment Source to a public location in the Fabric Storage using the **Upload - UploadPublic files** API.

Endpoint URL:

```
https://api-server-host-name/configapi/Uploads/PublicUpload
```

Method:

```
POST
```

Header:

FIELD	DESCRIPTION
file_name	Name of the CSV Enrichment Source pak file.
Content-Type	Content type of the CSV Enrichment Source pak file. It must be <i>(application/octet-stream)</i> .

Parameters:

FIELD	TYPE	DESCRIPTION	REQUIRED
file	[Object]	The CSV Enrichment Source pak file to be uploaded.	Mandatory

Success Response:

```
{
  "status": "Success",
  "message": "CSVEnrichmentSource_5.2.0.pak successfully uploaded in public storage."
}
```

2.2 Installing CSV Enrichment Source

You can install CSV Enrichment Source in a Fabric-enabled LogPoint using the **Upload - Install** API. You must upload the package to the Fabric Storage before the installation.

Endpoint URL:

`https://api-server-host-name/configapi/Uploads/{pool_UUID}/{logpoint_identifier}/install`

Method:

`POST`

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
application_type	-	String	Application type of CSV Enrichment Source. The value must be <i>Application</i> .	Mandatory
file_name	-	String	Name of the file. Obtain the name of the file uploaded to the public location using the Upload - ListPublic API and the private location using the Upload - List API.	Mandatory
file_location	-	String	Location of the file. Can be either <i>private</i> or <i>public</i> .	Mandatory

Request Example:

```
{
  "data": {
    "application_type": "Application",
    "file_name": "CSVEnrichmentSource_5.2.0.pak",
    "file_location": "private"
  }
}
```

Success Response:

```
{
  "status": "Success",
  "message": "monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"
}
```

CONFIGURING CSV ENRICHMENT SOURCE

3.1 Uploading a CSV File in the Fabric Storage

3.1.1 Private Storage

You can upload a CSV file to private storage in the Fabric storage using the **PluginConfiguration-Upload** API.

Endpoint URL:

```
https://api-server-host-name/configapi/{pool_UUID}/PluginConfiguration/  
↪ CSVEnrichmentSource/upload
```

Method:

```
POST
```

Header:

FIELD	LABEL IN UI	DESCRIPTION
Content-Type	–	Content type of the CSV file. It must be <i>Multipart/Form-data</i> .

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
file_name	–	Object	File to be uploaded.	Mandatory

Success Response:

```
{  
  "status": "Success",  
}
```

(continues on next page)

(continued from previous page)

```
"message": "enrichmentsource.csv successfully uploaded in private storage."
}
```

3.1.2 Public Storage

You can upload a CSV file to public storage in the Fabric storage using the **PluginConfiguration-UploadPublic** API.

Endpoint URL:

```
https://api-server-host-name/configapi/PluginConfiguration/CSVEnrichmentSource/
↪publicupload
```

Method:

POST

Header:

FIELD	LABEL IN UI	DESCRIPTION
Content-Type	–	Content type of the CSV file. It must be <i>Multipart/Form-data</i> .

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
file_name	–	Object	File to be uploaded.	Mandatory

Success Response:

```
{
  "status": "Success",
  "message": "enrichmentsource.csv successfully uploaded in public storage."
}
```

3.2 Configuring CSV Enrichment Source Using a CSV File

You can configure CSV Enrichment Source in a Fabric-enabled LogPoint using the **PluginConfiguration - Create** API.

Endpoint URL:

`https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/
↪ CSVEnrichmentSource`

Method:

`POST`

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
charset	Charset	String	Existing LogPoint charset. Obtain the value of the required charset using the Charsets - List API.	Mandatory
csv_mode	Upload Options	String	Option to provide the CSV file. It must be file .	Mandatory
delimiter	Delimiter	String	Type of data included in the CSV file.	Mandatory
source_fields	Source Fields	JSON	The field headers of the CSV file along with the sample data: field: Names of the fields in the CSV. type: Type of the fields in the CSV.	Mandatory
source_name	Name	String	Name for the configuration.	Mandatory
file_location	–	String	Location of the CSV file. Can be either <i>private</i> or <i>public</i> .	Mandatory
files	CSV file	String	The CSV file to enrich search results: csvfileupload: Name of the uploaded CSV file.	Mandatory

Request Example:

```
{
  "data": {
    "charset": "utf_8",
```

(continues on next page)

(continued from previous page)

```

    "csv_mode": "file",
    "delimiter": ",",
    "include_headers": true,
    "source_fields": [
      {
        "field": "user",
        "type": "string",
      }
    ],
    "source_name": "one_field",
    "file_location": "public",
    "files": {
      "csvfileupload": "enrichmentsource.csv"
    }
  }
}

```

Success Response:

```

{
  "status": "Success",
  "message": "/monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"
}

```

3.3 Configuring CSV Enrichment Source Using a URL

You can configure CSV Enrichment Source in a Fabric-enabled LogPoint using the **PluginConfiguration - Create API**.

Endpoint URL:

```

https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/
↪ CSVEnrichmentSource

```

Method:

POST

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
charset	Charset	String	Existing LogPoint charset. Obtain the value of the required charset using the Charsets - List API.	Mandatory
csv_mode	Upload Options	String	Option to provide the CSV file. It must be url .	Mandatory
delimiter	Delimiter	String	Type of data included in the CSV file.	Mandatory
include_headers	CSV file includes header	Boolean	Option to select if the CSV file contains field headers.	Optional
source_fields	Source Fields	JSON	The field headers of the CSV file along with the sample data: field: Names of the fields in the CSV. type: Type of the fields in the CSV.	Mandatory
source_name	Name	String	Name for the configuration.	Mandatory
auth_mode	Authentication	String	Authentication method of the provided CSV URL. It can be <i>none</i> or <i>basic</i> or <i>key</i> .	Mandatory
none	None	String	Authentication mode.	Optional

Continued on next page

Table 2 – continued from previous page

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
basic	Basic	String	Authentication mode: Username: Parameter to access the URL. Password: Parameter to access the URL.	Mandatory
key	Key	String	Authentication mode: Key: API Key to access the CSV URL.	Mandatory
csvfileurl	CSV URL	String	URL that links to the CSV file.	Mandatory
interval_day	Update Interval	Integer	Time interval in day to update the CSV URL.	Mandatory
interval_hour	Update Interval	Integer	Time interval in hour to update the CSV URL.	Mandatory
interval_minute	Update Interval	Integer	Time interval in minute to update the CSV URL.	Mandatory
verify	Verify	boolean	Parameter to to ensure a secure connection.	Optional
upload_cert_file	Upload Certificate File	Object	Option to upload a self-signed SSL certificate.	Optional
file_location	–	String	Location of the file.	Mandatory

Continued on next page

Table 2 – continued from previous page

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
files	Files	JSON	Self-signed SSL certificate: cert_file: Self-signed SSL certificate.	Mandatory

Request Example:

```
{
  "data":{
    "charset": "utf_8",
    "csv_mode":"url",
    "delimiter": ",",
    "include_headers": true,
    "source_fields":
      [
        {
          "field": "ip",
          "type": "IP"
        },
        {
          "field": "category",
          "type": "string"
        },
        {
          "field": "score",
          "type": "integer"
        },
        {
          "field": "first_seen",
          "type": "string"
        },
        {
          "field": "last_seen",
          "type": "string"
        },
        {
          "field": "ports",
          "type": "integer"
        }
      ]
  },
}
```

(continues on next page)

(continued from previous page)

```

"source_name": "working_url",
"auth_mode": "none",
"csvfileurl": "https://csvfile.com/spreadsheets/d/enrichmentsource.csv",
"interval_day": 30,
"interval_hour": 23,
"interval_minute": 30,
"verify": true,
"upload_cert_file": true,
"file_location": "public",
"files": {
  "cert_file": "cert.pem"
}
}

```

Success Response:

```

{
  "status": "Success",
  "message": "/monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"
}

```

3.4 Fetching CSV Enrichment Source Sample Data

You can fetch CSV Enrichment Source sample data using the **PluginConfiguration - Create** API.

Endpoint URL:

```

https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/
↳ CSVEnrichmentSource/

```

Method:

POST

Request Example:

```

{
  "data": {
    "charset": "utf_8",
    "csv_mode": "url",

```

(continues on next page)

(continued from previous page)

```

    "delimiter": ";",
    "include_headers": true,
    "source_name": "working_url",
    "auth_mode": "none",
    "csvfileurl": "https://csvfile.com/spreadsheets/d/enrichmentsource.csv",
    "interval_day": 30,
    "interval_hour": 23,
    "interval_minute": 30,
    "action": "fetch_sample_csv_data",
    "verify": true,
    "upload_cert_file": false,

  }
}

```

Success Response:

```

{
  "request": {},
  "logpoint_identifier": "4789cebbe0b14bc28ea898118c9185c0",
  "pool_uuid": "dccd20bc72e847d4b01db4b500888da3",
  "response": {
    "success": true,
    "data": {
      "data": [
        [
          "ip",
          "category",
          "score",
          "not_seen",
          "data"

```

(continues on next page)

(continued from previous page)

```

    ],
    [
      "1.1.1.1",
      "DDOS",
      "50",
      "2/10/71",
      "1234"
    ]
  ],
  "message": ""
}
}

```

3.5 Listing the CSV Enrichment Source Configurations

You can list the CSV Enrichment Source configurations using the **PluginConfiguration - List** API.

Endpoint URL:

```
https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/
↪ CSVEnrichmentSource
```

Method:

```
GET
```

Success Response:

```
{
  "delete_status": "",
  "source_name": "fileP",
  "source_type": "CSV",
  "plugin_info": {
    "includes_header": true,
    "charset": "utf_8",
    "delimiter": ";",
    "source_name": "fileP",
    "csv_file": "/opt/immune/app_store/norm/enrichment/csv/enrichmentsource.csv",
    "source_fields": [
      {
        "field": "ip",
        "type": "string",
        "name": "ip"
      },
      {
        "field": "score",
        "type": "string",
        "name": "score"
      },
      {
        "field": "category",
        "type": "string",
        "name": "category"
      },
      {
        "field": "port",
        "type": "string",
        "name": "port"
      }
    ],
    "csv_mode": "file"
  },
  "tid": "",
  "source_info": {
    "id": "6befbfbfd70364e64efbfbfd6fefbfbfd6c0f292a5349efbfbfd",
    "source_name": "CSV"
  },
  "active": true,
  "result": "Updated",
  "reason": null,
  "last_updated": 1599717155,
  "id": "5f59bf1f3d73545286029239"
}
```

3.6 Getting CSV Enrichment Source Configurations by ID

You can fetch CSV Enrichment Source configurations with the given ID using the **PluginConfiguration - Get** API.

Endpoint URL:

`https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/
↪ CSVEnrichmentSource/id`

Method:

GET

Parameters:

FIELD	TYPE	DESCRIPTION	Required
id	String	The ID of the existing CSV Enrichment Source configurations to fetch. Obtain the value of the ID using the EnrichmentSource - List API.	Mandatory

Success Response:

```
{
  "delete_status": "",
  "source_name": "fileP",
  "source_type": "CSV",
  "plugin_info": {
    "includes_header": true,
    "charset": "utf_8",
    "delimiter": ";",
    "source_name": "fileP",
    "csv_file": "/opt/immune/app_store/norm/enrichment/csv/enrichmentsource.csv",
    "source_fields": [
      {
        "field": "ip",
        "type": "string",
        "name": "ip"
      },
      {
        "field": "score",
```

(continues on next page)

(continued from previous page)

```

    "type": "string",
    "name": "score"
  },
  {
    "field": "category",
    "type": "string",
    "name": "category"
  },
  {
    "field": "port",
    "type": "string",
    "name": "port"
  }
],
"csv_mode": "file"
},
{
  "tid": "",
  "source_info": {
    "id": "6befbfbfd70364e64efbfbfd6fefbfbfd6c0f292a5349efbfbfd",
    "source_name": "CSV"
  },
  "active": true,
  "result": "Updated",
  "reason": null,
  "last_updated": 1599717155,
  "id": "5f59bf1f3d73545286029239"
}

```

3.7 Removing CSV Enrichment Source Configurations by ID

You can delete CSV Enrichment Source configurations using the **PluginConfiguration - Trash** API.

Endpoint URL:

https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/CSVEnrichmentSource/{id}

Method:

DELETE

Parameters:

FIELD	TYPE	DESCRIPTION	Required
id	String	The ID of the existing CSV Enrichment Source configurations to delete. Obtain the value of the ID using the EnrichmentSource - List API.	Mandatory

Success Response:

```
{
  "status": "Success",
  "message": "/monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"
}
```

3.8 Editing the CSV Enrichment Source Configuration Using a CSV File

You can edit a CSV Enrichment Source configuration in a Fabric-enabled LogPoint using the **PluginConfiguration - Edit** API.

Endpoint URL:

```
https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/
↪ CSVEnrichmentSource/{id}
```

Method:

PUT

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
charset	Charset	String	Existing LogPoint charset. Obtain the value of the required charset using the Charsets - List API.	Mandatory

Continued on next page

Table 3 – continued from previous page

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
csv_mode	Upload Options	String	Option to provide the CSV file. It must be file .	Mandatory
delimiter	Delimiter	String	Type of data included in the CSV file.	Mandatory
source_fields	Source Fields	JSON	The field headers of the CSV file along with the sample data: field : Names of the fields in the CSV. type : Type of the fields in the CSV.	Mandatory
source_name	Name	String	Name for the configuration.	Mandatory
file_location	–	String	Location of the CSV file. Can be either <i>private</i> or <i>public</i> .	Mandatory
files	CSV file	String	The CSV file to enrich search results: csvfileupload : Name of the uploaded CSV file.	Mandatory
id	–	String	Existing CSV Enrichment Source ID. Obtain the value of the ID using the EnrichmentSource - List API.	Mandatory

Request Example:

```
{
  "data": {
    "charset": "utf_8",
```

(continues on next page)

(continued from previous page)

```

    "csv_mode": "file",
    "delimiter": ",",
    "include_headers": true,
    "source_fields": [
      {
        "field": "user",
        "type": "string",
      }
    ],
    "source_name": "one_field",
    "file_location": "public",
    "files": {
      "csvfileupload": "enrichmentsource.csv"
    }
  }
}

```

Success Response:

```

{
  "status": "Success",
  "message": "/monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"
}

```

3.9 Editing the CSV Enrichment Source Configuration Using a URL

You can edit a CSV Enrichment Source configuration in a Fabric-enabled LogPoint using the **PluginConfiguration - Edit** API.

Endpoint URL:

```

https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/
↪ CSVEnrichmentSource/{id}

```

Method:

PUT

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
charset	Charset	String	Existing LogPoint charset. Obtain the value of the required charset using the Charsets - List API.	Mandatory
csv_mode	Upload Options	String	Option to provide the CSV file. It must be url .	Mandatory
delimiter	Delimiter	String	Type of data included in the CSV file.	Mandatory
include_headers	CSV file includes header	Boolean	Option to select if the CSV file contains field headers.	Optional
source_fields	Source Fields	JSON	The field headers of the CSV file along with the sample data: field: Names of the fields in the CSV. type: Type of the fields in the CSV.	Mandatory
auth_mode	Authentication	String	Authentication method of the provided CSV URL. It can be <i>none</i> or <i>basic</i> or <i>key</i> .	Mandatory
none	None	String	Authentication mode.	Optional
basic	Basic	String	Authentication mode: Username: Parameter to access the URL. Password: Parameter to access the URL.	Mandatory
key	Key	String	Authentication mode: Key: API Key to access the CSV URL.	Mandatory
csvfileurl	CSV URL	String	URL that links to the CSV file.	Mandatory
interval_day	Update Interval	Integer	Time interval in days to update the CSV URL.	Mandatory

Continued on next page

Table 4 – continued from previous page

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
interval_hour	Update Interval	Integer	Time interval in hours to update the CSV URL.	Mandatory
interval_minute	Update Interval	Integer	Time interval in minutes to update the CSV URL.	Mandatory
verify	Verify	boolean	Parameter to to ensure a secure connection.	Optional
upload_cert_file	Upload Certificate File	Object	Option to upload a self-signed SSL certificate.	Optional
file_location	–	String	Location of the file.	Mandatory
files	Files	JSON	Self-signed SSL certificate: cert_file : Self-signed SSL certificate.	Mandatory
id	–	String	Existing CSV Enrichment Source ID . Obtain the value of the ID using the EnrichmentSource - List API.	Mandatory

Request Example:

```
{
  "data":{
    "charset": "utf_8",
    "csv_mode": "url",
    "delimiter": ",",
    "include_headers": true,
    "source_fields":
      [
        {
          "field": "ip",
          "type": "IP"
        },
        {
          "field": "category",
          "type": "string"
        },
        {
          "field": "score",
          "type": "integer"
        }
      ]
  }
}
```

(continues on next page)

(continued from previous page)

```

    },
    {
      "field": "first_seen",
      "type": "string"
    },
    {
      "field": "last_seen",
      "type": "string"
    },
    {
      "field": "ports",
      "type": "integer"
    }
  ],

  "source_name": "working_url",
  "auth_mode": "none",
  "csvfileurl": "https://csvfile.com/spreadsheets/d/enrichmentsource.csv",
  "interval_day": 30,
  "interval_hour": 23,
  "interval_minute": 30
}
}

```

Success Response:

```

{
  "status": "Success",
  "message": "/monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"
}

```

3.10 Deleting the CSV Enrichment Source Configuration

You can delete a CSV Enrichment Source configuration in a Fabric-enabled LogPoint using the **PluginConfiguration - Trash** API.

Endpoint URL:

```

https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/
↪ CSVEnrichmentSource/{id}

```

Method:

Delete

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
id	–	String	Existing CSVEnrichmentSource id. Obtain the value of the ID using the EnrichmentSource - List API.	Mandatory

Success Response:

```
{
  "status": "Success",
  "message": "/monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"
}
```

3.11 Deleting the CSV File in the Fabric Storage

3.11.1 Trash Private Uploads

You can delete the uploaded CSV files in private storage in the Fabric storage using the **PluginConfiguration-TrashPrivateUploads** API.

Endpoint URL:

[https://api-server-host-name/configapi/{pool_UUID}/PluginConfiguration/
→ CSVEnrichmentSource/{file_name}](https://api-server-host-name/configapi/{pool_UUID}/PluginConfiguration/CSVEnrichmentSource/{file_name})

Method:

DELETE

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
file_name	–	String	File to be deleted.	Mandatory

Success Response:

```
{
  "status": "Success",
  "message": "enrichmentsource.csv successfully deleted."
}
```

3.11.2 Trash Public Uploads

You can delete the uploaded CSV files in public storage in the Fabric storage using the **PluginConfiguration-TrashPublicUploads** API.

Endpoint URL:

```
https://api-server-host-name/configapi/PluginConfiguration/CSVEnrichmentSource/{file_name}
```

Method:

DELETE

Header:

FIELD	LABEL IN UI	DESCRIPTION
Content-Type	–	Content type of the CSV file. It must be <i>Multipart/Form-data</i> .

Parameters:

FIELD	LABEL IN UI	TYPE	DESCRIPTION	REQUIRED
file_name	–	String	File to be deleted.	Mandatory

Success Response:

```
{
  "status": "Success",
  "message": "enrichmentsource.csv successfully deleted."
}
```

UNINSTALLING CSV ENRICHMENT SOURCE

You can uninstall CSV Enrichment Source in a Fabric-enabled LogPoint using the **Plugins - Uninstall** API.

Endpoint URL:

```
https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/Plugins/{id}
```

Method:

```
DELETE
```

Parameters:

FIELD	TYPE	DESCRIPTION	REQUIRED
id	String	The ID of the existing CSV Enrichment Source. Obtain the value of the ID using the Plugins - List API.	Mandatory

Success Response:

```
{  
  "status": "Success",  
  "message": "/monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"  
}
```

CSV ENRICHMENT SOURCE - REFRESHLIST

5.1 Refreshing the list for CSV Enrichment Source Collections

You can refresh the updated collections of CSV Enrichment Source using the **PluginConfiguration - RefreshList** API.

Endpoint URL:

```
https://api-server-host-name/configapi/{pool_UUID}/{logpoint_identifier}/PluginConfiguration/  
↪ CSVEnrichmentSource/refreshlist
```

Method:

```
POST
```

Request Example:

```
{  
  "data": {  
  }  
}
```

Success Response:

```
{  
  "status": "Success",  
  "message": "/monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"  
}
```

CSV ENRICHMENT SOURCE - TRASH

6.1 Deleting CSV Enrichment Source from the Fabric Storage

6.1.1 Private Uploads

You can delete CSV Enrichment Source package from private storage in the Fabric Storage using the **Upload - TrashPrivate** API.

Endpoint URL:

```
https://api-server-host-name/configapi/Uploads/{pool_UUID}/{file_name}
```

Method:

```
DELETE
```

Parameters:

FIELD	TYPE	DESCRIPTION	REQUIRED
id	String	CSV Enrichment Source application ID. Obtain the ID of CSV Enrichment Source using the Plugins - List API.	Mandatory

Success Response:

```
{
  "status": "Success",
  "message": "monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"
}
```

6.1.2 Public Uploads

You can delete CSV Enrichment Source package from public storage in the Fabric Storage using the **Upload - TrashPrivate** API.

Endpoint URL:

```
https://api-server-host-name/configapi/Uploads/{file_name}
```

Method:

```
DELETE
```

Parameters:

FIELD	TYPE	DESCRIPTION	REQUIRED
id	String	CSV Enrichment Source application ID. Obtain the ID of CSV Enrichment Source using the Plugins - List API.	Mandatory

Success Response:

```
{
  "status": "Success",
  "message": "monitorapi/{pool_UUID}/{logpoint_identifier}/orders/{request_id}"
}
```