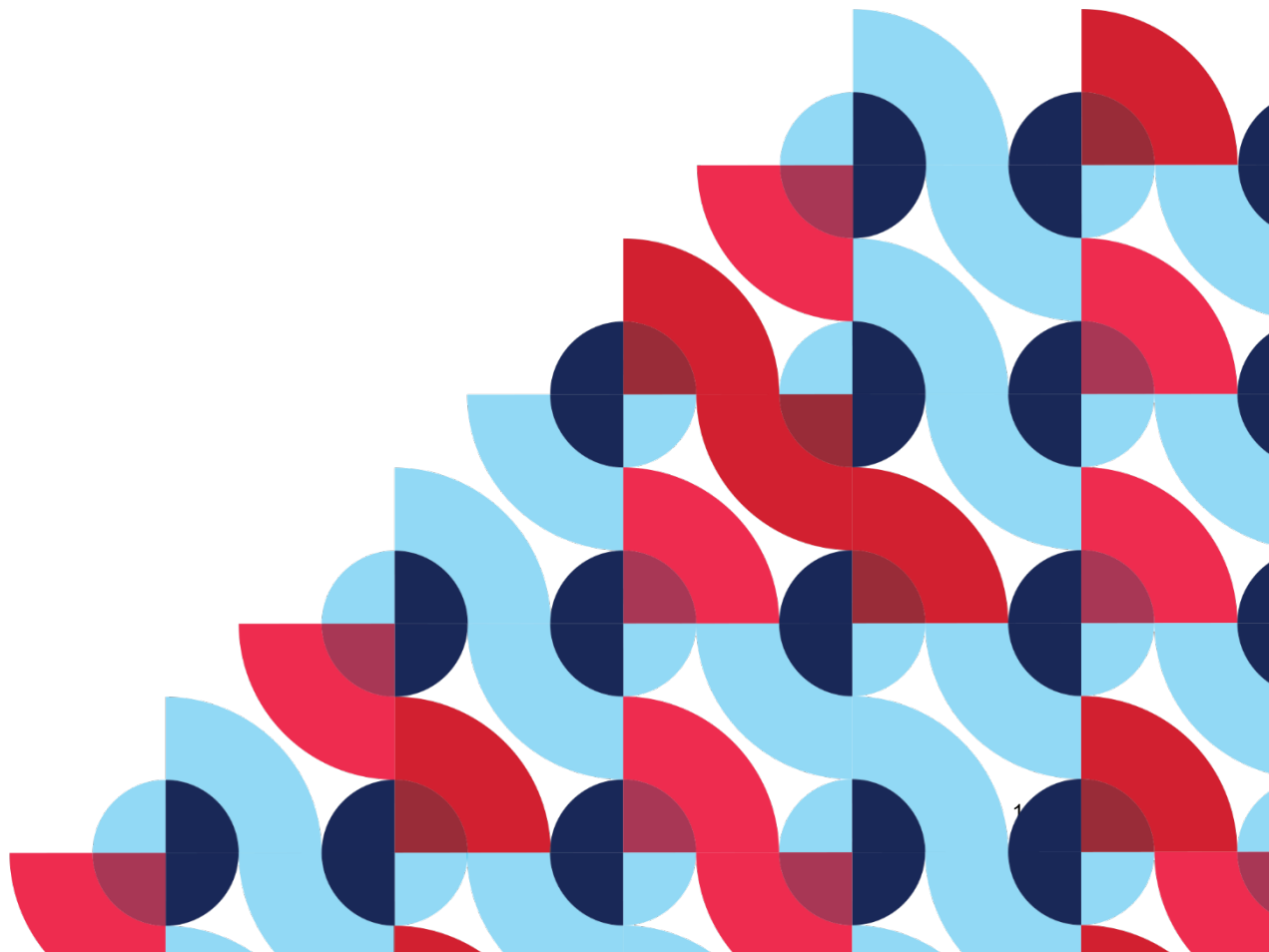United States
Consumer Product Safety Commission

eFiling Application Programming Interface (API) Specifications

December 2025

Version 3.0

Version/Revision Log

Date	Version No.	Description	Page #	Author
9/18/2023	1.0	Initial version	ALL	B. Holsey
8/1/2024	2.0	• Added changelog and contents • Minor text updates throughout • Guidance on null responses • Added "archive" where applicable • Added character limits to field tables	2-3 ALL 4 10, 17 18-20	B. Holsey
12/17/2025	3.0	• New endpoints added • Updated placeholder/sample data • Updated /import with details of certifying during data upload • Updated valid searchTerm values for use with /export • Updated /getTradeParties description to reflect privacy settings • Improved field definitions including updated character lengths and addition of component identifier fields	Multiple Multiple 7 10 15 21-23	B. Holsey

Contents

Overview

The eFiling API allows developers to integrate with CPSC's Product Registry so that product certificate data can be provided from third-party software, to better facilitate the collection of product certificate data to ensure compliance with applicable laws.

Software developers handling integration with the eFiling API will be required to create accounts within the Product Registry to facilitate integration testing. Software development accounts will enable users to establish development/test product collections, add placeholder product certificate information, send collaboration invitations to development team members, and access an API Portal to assist with development. These accounts should *not* be used to store real certificate of compliance data.

Access

All eFiling APIs require an API token to be supplied for a valid eFiling user. To retrieve the token, the user will login to the Product Registry portal, where they will be able to generate an API token that will permit interaction with any Product Collections that the user is a member of. Please note that this token provides the same access rights as the user account for which it was generated. Any changes made through the API will directly affect the specified resources that the user has access to. All development activities should be done against Software Developer accounts and NOT actual importer Business Accounts.

When development of APIs is complete, the integration will allow normal users to provide their API tokens and secrets, which will then be passed through the API request headers as described below. This token will allow the integrated system to act on behalf of that user in accessing and updating data that the user has previously been granted access to. These tokens expire regularly, and the user will be required to generate a new token for continued access to eFiling resources through the API. An API endpoint is provided to check that the current token is still valid and return its expiration date.

The eFiling API will reside under the efiling.saferproducts.gov domain at the path /efiling/api, followed by the specific endpoint being called.

Errors

All APIs will return 401 or 403 access errors for invalid/expired tokens and user requests to access/update data for which they have not been granted access.

If, during development, you receive a "null" response error, it most likely means that there was an issue with your call for which proper error handling has not yet been configured. It is important to note that null errors are *not* always an indication that no data was ingested by the Product Registry, and as such, subsequent attempts to pass the same data via the API may result in Version ID error responses. In such instances, please reach out to the eFiling support team at EFilingSupport@cpsc.gov and provide a copy of the call/payload that was sent.

A more detailed list of error response messages that will be returned from the API under various circumstances will be added to a future iteration of this document.

Endpoints

• **getCollections (GET)** • **import (POST)** • **getImportStatus (GET)** • **getImportLog (GET)**	• **export (GET)** • **exportAsync (GET)** • **exportAsyncStatus (GET)** • **exportAsyncData (GET)**	• **getCertificates (POST)** • **getTradeParties (GET)** • **getTokenExpiration (GET)**

All endpoints will only be accessible to authorized users. Each call must pass both a JSON Web Token (JWT) and its corresponding Secret in order to be validated. The JWT will be passed as a "Authorization: <JWT>" header and the secret will be passed as a "x-api-secret: <secret>" header.

Data Format

The JSON data that gets sent to and from the eFiling Product Registry API has been organized into "modules" to minimize the amount of structural modification necessary for the different endpoint calls.

The largest module, coreProduct, is sent/returned inside a productList object and contains all of the data fields that represent a single, valid product certificate.

The table below details the parameter and module requirements for each endpoint. Formatting examples for modules can be found in the Data Formatting Samples section of this document. The correct ordering of the modules can be found in the sections detailing each individual endpoint.

Endpoint	Parameters	JSON Modules in Request Body	JSON Modules in Response
getCollections	None	--	businessAccountList
import*	• Certifier ID • Collection ID	coreProduct, directives	--
getImportStatus	• Import ID	--	--
getImportLog*	• Import ID • "Errors Only" flag (optional)	--	coreProduct, directives, validations
export*	• Certifier ID • Collection ID • Filters (optional)	--	coreProduct, productMetadata
exportAsync	• Certifier ID • Collection ID • Filters (optional)	--	--
exportAsyncStatus	• Export ID	--	--
exportAsyncData*	• Export ID	--	coreProduct, productMetadata
getCertificates*	• Certifier ID • Collection ID	Array of Product ID/Version tuples	coreProduct, productMetadata
getTradeParties	• Certifier ID • Trade party type (optional)	--	tradePartyList
getTokenExpiration	None	--	--

*Data modules sent and received from these endpoints will be contained within the "productList" JSON object (see example at the end of the document).

Endpoint Details

getCollections (GET)

Description: Returns a list of Product Collections that the provided token can access.

Headers
- Authorization: <JWT>
- x-api-secret: <secret>

Parameters
- No parameters required; response is based on provided JWT

Response
- JSON list of collection information (will include Business Account information as well)
  - Status Code
  - Status Message
  - Business Account List, including accessible collection names and user role(s)

Response Sample

```
{
  "statusCode": "200",
  "statusMessage": "OK",
  "businessAccountList": [{
      "certifierId": "example-company",
      "companyName": "Example Company",
      "collectionList": [{
          "collectionId": "11111111-2222-3333-4444-999999999999",
          "collectionName": "API Demo",
          "userRole": ["Collection Editor"]}, {
          "collectionId": "22222222-3333-4444-5555-999999999999",
           "collectionName": "Another Collection",
           "userRole": ["Collection Administrator", "Certifier"]
      }],
    },{
      "certifierId": "company-two",
      "companyName": "Company Two",
      "collectionList": [{
          "collectionId": "33333333-4444-5555-6666-999999999999",
          "collectionName": "Furniture",
          "userRole": ["Collection Viewer"]
      }]
  }]
}
```

import (POST)

Description: Used to import Certificate of Compliance data from the accompanying JSON payload into the specified eFiling Product Collection. This import process can be used to add new certificates or update existing certificates. Updating existing certificate data or adding new trade parties to a Business Account is accomplished through the "directives" module that follows the coreProduct module. If the importing user is authorized to certify within the destination Product Collection, they have the option of setting the doCertify value to true; if set, all valid certificates will also be certified upon import. If not set, or set to false, imported certificates will be in "Awaiting Certification" status. **Setting "doCertify" to true for a user that is not authorized to certify will return an error message. The presence of the "Certifier" role on a collection can be checked using the getCollections endpoint.** A sample JSON payload is available at the end of this document.

Headers
- Authorization: <JWT>
- x-api-secret: <secret>

Parameters
- Certifier ID (String – Required), Key: certifierId
- Collection ID (String – Required), Key: collectionId
- Certify Valid Indicator (Boolean – Optional), Key: doCertify

Request Body
- Product certificate set (JSON)
    - List of coreProduct/directive module pairs within a productList object (full example here)

Response
- Status Code
- Status Message
- Import ID
- Import Status ("Attempting to upload" if successful; "Upload failed" if unsuccessful)

Response Sample
```
{
  "statusCode": "200",
  "statusMessage": "OK",
  "importId": "12345678-1234-4321-1234-123456789098",
  "importStatus": "Attempting to upload."
}
```

getImportStatus (GET)

Description: Depending on the quantity of data being uploaded, calls to /import can take substantial time to process. Because of this, the /getImportStatus endpoint can be used to check the progress of an upload, returning both a percentage complete as well as a status message.

Headers
- Authorization: <JWT>
- Secret: <secret>

Parameters
- Import ID (String – Required), Key: importId

Response
- Status Code
- Status Message
- Percentage Complete
- State ("Uploading…" if still in progress; "Complete" if finished; "Failed" if finished unsuccessfully)

Response Sample
```
{
 "statusCode": "200",
 "statusMessage": "OK",
 "percentComplete": "44",
 "importStatus": "Uploading…"
}
```

getImportLog (GET)

Description: Used to retrieve a detailed summary of a specific import event. It is recommended that users poll getImportStatus until a status of "Complete" is returned before attempting to retrieve an import log from this endpoint.

Headers
- Authorization: <JWT>
- x-api-secret: <secret>

Parameters
- Import ID (String – Required), Key: importId
- Errors Only flag (Boolean – Optional) (true for only error rows or false for all rows; defaults to false if not provided), Key: errorsOnly

Response
- Status Code
- Status Message
- Response data will be wrapped in a productList object, containing:
  - All data provided via coreProduct modules during upload will be returned
  - All data provided via directives modules during upload will be returned
  - Additional validations module will be returned that includes any error messages generated by the upload

Response Sample
```
{
"statusCode": "200",
"statusMessage": "OK",
    "productList": [{
      "coreProduct": {
            Module data },
      "directives": {
            Module data },
      "validations": {
            Module data }
    },{
      "coreProduct": {
            Module data },
      "directives": {
            Module data },
      "validations": {
            Module data }
    }
   ]
}
```

export (GET)

Description: A synchronous call to retrieve certificate data from a specific Product Collection. Parameters can be passed with the request to filter the certificates returned to various subsets. This call can only retrieve a maximum of 10,000 product certificates at one time and is provided as a convenience. For large collections of products, we recommend using exportAsync, exportAsyncStatus and exportAsyncData.

Headers
- Authorization: <JWT>
- x-api-secret: <secret>

Parameters
- Certifier ID (String – Required), Key: certifierId
- Collection ID (String – Required), Key: collectionId
- Optional Filters:
    - Type of Date (String of "Last Test Date", "Manufacture Date", "Production Start Date", or "Create Date" – Optional), Key: typeOfDate
    - Date range start (String in date format MM/DD/CCYY – Optional unless "Type of Date" provided), Key: startDate
    - Date range end (String in date format MM/DD/CCYY – Optional unless "Type of Date" provided), Key: endDate
    - Status filter (String of "Complete", "Incomplete" or "Awaiting Certification – Optional; if not provided, defaults to "All"), Key: status
    - Visibility filter (String of "Visible", "Archived" or "Both" – Optional; if not provided, defaults to "Visible"), Key: visibility
    - Search term (String – Optional), Key: searchTerm
    - Search on field (String of "all", "productId", "versionId", "productName", "trade/BrandName", "manufacturerName", "lotNumber", "cpscLabId", or "labCitationCode")

Response
- Status Code
- Status Message
- List of coreProduct/productMetadata module pairs inside a productList object that match specified filter criteria (if provided).
    - Most current data for the certificate will be returned via coreProduct module
    - Certificate metadata will be returned via productMetadata module

Response Sample
```
{
"statusCode": "200",
"statusMessage": "OK",
    "productList": [{
      "coreProduct": {
            Module data },
      "productMetadata": {
            Module data }
    }]
}
```

exportAsync (GET)

Description: Initiates an asynchronous process for retrieving certificate data from a specific Product Collection. Response will be an export ID that can subsequently be used with the /exportAsyncStatus and /exportAsyncData endpoints. This call can retrieve a maximum of 50,000 product certificates at one time.

Headers
- Authorization: <JWT>
- x-api-secret: <secret>

Parameters
- Certifier ID (String – Required), Key: certifierId
- Collection ID (String – Required), Key: collectionId
- Optional Filters:
  - Type of Date (String of "Last Test Date", "Manufacture Date", "Production Start Date", or "Create Date" – Optional), Key: typeOfDate
  - Date range start (String in date format MM/DD/CCYY – Optional unless "Type of Date" provided), Key: startDate
  - Date range end (String in date format MM/DD/CCYY – Optional unless "Type of Date" provided), Key: endDate
  - Status filter (String of "Complete", "Incomplete" or "Awaiting Certification – Optional; if not provided, defaults to "All"), Key: status
  - Visibility filter (String of "Visible", "Archived" or "Both" – Optional; if not provided, defaults to "Visible"), Key: visibility
  - Search term (String – Optional), Key: searchTerm
  - Search on field (String of "all", "productId", "versionId", "productName", "trade/BrandName", "manufacturerName", "lotNumber", "cpscLabId", or "labCitationCode")

Response
- Export Status Message
- Export ID
- Status Message
- Status Code

Response Sample
```
{
"exportStatus": "Attempting to export.",
"exportId": "12345678-1234-1234-1234-123456789012",
"statusMessage": "OK",
"statusCode": "200"
}
```

exportAsyncStatus (GET)

Description: Depending on the quantity of data being exported, calls to /exportAsync can take substantial time to finish processing. The /exportAsyncStatus endpoint can be used to check the progress of an asynchronous export, returning both a percentage complete as well as a status message.

Headers
- Authorization: <JWT>
- x-api-secret: <secret>

Parameters
- Export ID (String – Required), Key: exportId

Response
- Percent Complete
- Export Status Message (Possible values are: "Started", "Completed", "Downloaded", "Failed: The result set is too large", or "Failed: Transient GraphQL error")
- Status Message
- Status Code

Response Sample
```
{
"percentageComplete": "50",
"exportStatus": "started",
"statusMessage": "OK",
"statusCode": "200"
}
```

exportAsyncData (GET)

Description: Used to retrieve the data from a specific asynchronous export. It is recommended that users poll exportAsyncStatus until a status of "Completed" is returned before attempting to retrieve an export from this endpoint.

Headers
- Authorization: <JWT>
- x-api-secret: <secret>

Parameters
- Export ID (String – Required), Key: exportId

Response
- Status Code
- Status Message
- Response data will be wrapped in a productList object, containing:
  - Individual certificate data stored in coreProduct modules
  - Associated metadata for each certificate stored in productMetadata modules

Response Sample

```
{
"statusMessage": "OK",
"productList": [{
      "coreProduct": {
             Module data },
      "productMetadata": {
             Module data },
    },{
      "coreProduct": {
             Module data },
      "productMetadata": {
             Module data },
    }
   ],
"statusCode": "200",
}
```

getCertificates (POST)

Description: Used to retrieve the data associated with one or more specific product certificates. If no version ID is provided, all completed certificate versions for a provided Product ID will be returned along with the most recent, or "latest", version (even if that version is not complete yet).

    Headers
- Authorization: <JWT>
- x-api-secret: <secret>

    Parameters
- Certifier ID (String – Required), Key: certifierId
- Collection ID (String – Required), Key: collectionId
- List of Product ID, Version tuples (JSON String in request body)
  - Example:
    ```
    [{
       "productId": "1234-56-789",
       "version": ""
     },{
       "productId": "1111-22-333",
       "version": "V2" }]
    ```

    Response
- Status Code
- Status Message
- List of coreProduct/productMetadata module pairs inside a productList object.
  - Data for each certificate based on specified version (if version ID is provided) returned via list of coreProduct/productMetadata modules inside productList object
  - Data for all completed versions of certificates for a specified Product ID will be returned if no version is specified
  - Along with all completed versions, the most recent, or "latest" certificate associated with a Product ID will be returned. The status of this version will be identified in its corresponding productMetadata module.

    Response Sample
```
{
"statusCode": "200",
"statusMessage": "OK",
    "productList": [{
      "coreProduct": {
            Module data },
      "productMetadata": {
            Module data }
    }, {
      "coreProduct": {
            Module data },
      "productMetadata": {
            Module data }
    }
    ]
}
```

getTradeParties (GET)

Description: Returns a list of the trade parties associated to a specific Product Collection if a Collection ID is provided, or a Business Account's "shared" trade-party list. However, the user token provided with the call must have access to the specific collection requested, or one or more collections that use the specified Business Account's "shared" trade party list for this call to succeed. If the privacy settings of the destination Product Collection are not known, a Collection ID should always be provided.

Headers
- Authorization: <JWT>
- x-api-secret: <secret>

Parameters
- Certifier ID (String – Required), Key: certifierId
- Collection ID (String – Contingent), Key: collectionId
- Trade Party Type (String of "All", "Manufacturer", "Laboratory" or "Point of Contact" – Optional), Key: tradePartyType

Response
- Status Code
- Status Message
- JSON list of trade parties and their data

Response Sample

```
{ "tradePartyList": [
    "tradePartyType": "MANUFACTURER",
    "tradePartyName": "Example Manufacturer LLC",
    "division": null,
    "gln": "",
    "alternateId": "Example-Manufacturer",
    "email": "support@example-manufacturer.xy",
    "website": null,
    "addressLine1": "123 Address Hwy",
    "addressLine2": "",
    "city": "Example City",
    "stateProvince": "TX",
    "postalCode": "55555",
    "country": "USA",
    "phone": "555-123-6543",
    "smallBatchManufacturerId": "",
    "aptNumber": "404" }
    ]
}
```

getTokenExpiration (GET)

Description: Returns the expiration date and time of the API token that made the call.

Headers
- Authorization: <JWT>
- x-api-secret: <secret>

Parameters
- No parameters required

Response
- Status Code
- Status Message
- A single key-value pair containing a date-time string

Response Sample
```
{
  "statusCode": "200",
  "statusMessage": "OK",
  "tokenExpiration": "2023-12-31 23:59:59.1234567 +00:00"
}
```

Data Formatting Samples

Below are JSON samples representative of how the eFiling API will accept and return the different data elements required to interact with the Product Registry via the API. Descriptions of each field can be found in the definitions section further down in this document.

coreProduct Module

This module contains all of the mandatory, contingent, and optional fields that can comprise a single, valid product certificate.

```
"coreProduct": {
    "versionId": "V1",
    "primaryProductId": "1234-5678-9",
    "primaryProductIdType": "Model #",
    "identifiers": [{
        "identifier": "",
        "identType": "GTIN"}, {
        "identifier": "",
        "identType": "UPC"}, {
        "identifier": "",
        "identType": "SKU"}, {
        "identifier": "",
        "identType": "Model #"}, {
        "identifier": "",
        "identType": "Serial #"}, {
        "identifier": "",
        "identType": "Registered #"}, {
        "identifier": "",
        "identType": "Alternate ID"}],
    "certificateType": "GCC",
    "name": "Coffee table",
    "tradeBrandName": "FurnitureCo",
    "description": "Black coffee table, square",
    "color": "Black",
    "style": "Modern",
    "manufacturer": {
        "gln":"",
        "alternateId": "example-manufacturer",
        "sbmId":"",
        "name": "Example Manufacturer Ltd.",
        "addressLine1": "123 Example Rd.",
        "addressLine2": "",
        "aptNumber": "222",
        "city": "Example City",
        "stateProvince": "MD",
        "country": "USA",
        "postalCode": "11111",
```

```
          "phone": "555-234-5678",
          "email": "support@examplemanufacturer.xy"},
      "manufactureDate": "03/2020",
      "productionStartDate": "03/20/2020",
      "productionEndDate": "03/22/2020",
      "lotNumber": "12345",
      "lotNumberAssignedBy": "MANUFACTURER",
      "lastTestDate": "04/27/2020",
      "labs": [{
        "type": "ITL",
        "cpscId": "9999",
        "gln": "",
        "alternateId": "",
        "name": "",
        "addressLine1": "",
        "addressLine2": "",
        "aptNumber": "",
        "city": "",
        "stateProvince": "",
        "country": "",
        "postalCode": "",
        "phone": "",
        "email": "",
        "citationCodes": ["1215", "1216", "1217"],
        "testReportId": "1234-55567",
        "testURL": "www.lab1.com/testreports/0187652",
        "testReportAccessKey": "",
        "isComponent": false,
        "componentDescription": ""},{
        "type": "LAB",
        "cpscId": "",
        "gln": "",
        "alternateId": "lab-two",
        "name": "Lab Two, Inc.",
        "addressLine1": "456 Placeholder Lane",
        "addressLine2": "",
        "aptNumber": "Ste. 42",
        "city": "Los Angeles",
        "stateProvince": "CA",
        "country": "USA",
        "postalCode": "90001",
        "phone": "555-555-5555",
        "email": "support@labtwo.net",
        "citationCodes": ["1201", "1202", "1207"],
        "testReportId": "1234-4567",
        "testURL": "www.lab-two.com/test-reports/1234567",
        "testReportAccessKey": "password!",
```

```
      "isComponent": true,
      "componentDescription": "Table legs"}],
  "exemptions": ["1500.91(d)(5)", "1610.1(d)(1)"],
  "poc": {
    "type": "Importer",
    "gln": "",
    "alternateId": "",
    "name": "",
    "addressLine1": " ",
    "addressLine2": "",
    "aptNumber": "",
    "city": "",
    "stateProvince": "",
    "country": "",
    "postalCode": "",
    "phone": "",
    "email": ""}
}
```

Directives Module

This module must follow each coreProduct module sent to the /import endpoint and will also be returned from calls to /getImportLog. It is used to indicate when a product certificate is being updated—as opposed to being added for the first time—and is also how a user indicates that a trade party (manufacturer, laboratory or point of contact) is being used for the first time and should be added to the destination Product Collection.

```
"directives": {
  "productUpdate": "Y",
  "versionIdToUpdate": "V1",
  "manufacturer": {
    "isNew": "Y",
    "gln": "",
    "alternateId": "example-manufacturer"},
  "labs": [{
    "isNew": "N",
    "gln": "",
    "alternateId": "lab-two"}],
  "poc": {
    "isNew": "Y",
    "gln": "",
    "alternateId": "example-poc"}
  }
```

Product Metadata Module

This module will never be passed by users but will be part of the response when the export endpoints are called. It contains information about the most recent version of a given Product ID's certificate, as well as a list of all Version IDs associated with previously completed certificates for that Product ID.

```
"productMetadata": {
    "isLatestVersion": "Y",
    "status": "Incomplete",
    "isArchived": "N",
    "completedVersions": ["V1", "V2", "V3"]
  }
```

Validations Module

This module is returned to users as part of the response sent back from calls to the /getImportLog endpoint. It contains information about error messages that may have been triggered during an attempt to upload certificate data. (**Note:** the error codes and messages in the sample below are for demonstration purposes only. Currently, the API only returns error messages; error codes will be added in a future update.)

```
"validations": {
  "importedInd": "N",
  "errorDetectedInd": "Y",
  "errors": [{
        "errorCode": "600",
        "errorMessage": "The Phone Number for the Manufacturer is invalid."}, {
        "errorCode": "610",
        "errorMessage": "The City for the Manufacturer is invalid."}
          ]
        }
```

Business Account List & Trade Party List Modules

These modules are the least complicated, and examples of the data they return can be found in the sample responses displayed in the /getCollections and /getTradeParties endpoint sections above. They store one or more JSON objects inside of businessAccountList and tradePartyList objects, respectively.

Definitions of User-Provided Fields

The tables below provide descriptions of all the different data fields that can be sent to provide certificate data by way of the eFiling API.

All Fields in coreProduct

Field Name	Description	Max Character Length	Mandatory (M) Conditional (C) Optional (O)
versionId	The version identifier currently associated with a certificate, or if data is being sent to the API, the desired version identifier.	19	M
primaryProductId	The unique identifier for the product.	19	M
primaryProductIdType	Identifies the type of the identifier used for the primaryProductId. Acceptable values: GTIN, UPC, SKU, Model #, Serial #, Registered #, Alternate ID	N/A	M
identifiers	Additional fields to provide additional product identifiers; do not modify the "identType" fields.	50	O
certificateType	The type of certificate being provided. Acceptable values: "GCC" or "CPC"	N/A	M
name	The name of the product.	250	M
tradeBrandName	The trade/brand name associated with the product.	50	O
description	A brief description of the product.	250	O
color	The color of the product.	50	O
style	The style of the product.	50	O
manufacturer			
• gln	Either a Global Location Number (GLN) or a custom Alternate ID are required when providing a new manufacturer or identifying an existing one.	50	M
• alternateId		50	
• sbmId	The Small Batch Manufacturer ID number, if applicable.	50	O
• name	The name of the manufacturer.	100	C
• addressLine1	The full address of the manufacturer is required if it is being added to the Business Account for the first time. Adding a new manufacturer also requires that "isNew" has a value of "Y" in the corresponding directives module. If "isNew" is left blank or set to "N", the system will attempt to find the manufacturer referred to by GLN/Alternate ID in the Business Account being uploaded to.	100	C
• addressLine2		100	
• aptNumber		10	
• city		50	
• stateProvince		100	
• country		50	
• postalCode		10	
• phone	Phone number of the manufacturer.	20	C
• email	Email address of the manufacturer.	50	C
manufactureDate	The month and year the product was manufactured, in the format "MM/CCYY".	N/A	M
productionStartDate	The start date of the product batch, in the format "MM/DD/CCYY".	N/A	O
productionEndDate	The end date of the product batch, in the format "MM/DD/CCYY".	N/A	O
lotNumber		20	C

lotNumberAssignedBy	These fields are optional; however, if one is provided, the other becomes mandatory. Valid values for lotNumberAssignedBy are "Manufacturer" and "Seller".	N/A	
lastTestDate	The most recent date of testing for the product, in the format "MM/DD/CCYY" (Note: Products that require no testing should not be uploaded to the Product Registry, and should instead be filed using a Disclaim message set).	N/A	M
Labs			
• **type**	The type of lab. Acceptable values are: "ITL" and "LAB".	N/A	M
• **cpscId**	The CPSC-assigned ID of a laboratory; only mandatory if the lab is of type "ITL".	N/A	C
• **gln**	Either a Global Location Number (GLN) or a custom Alternate ID are required when providing a new laboratory, or identifying an existing one, if it is of type "LAB".	50	C
• **alternateId**		50	
• **name**	The name of the laboratory.	100	C
• **addressLine1**	The full address of the laboratory is required if it is being added to the Business Account for the first time. Adding a new laboratory also requires that "isNew" has a value of "Y" in the corresponding directives module. If "isNew" is left blank or set to "N", the system will attempt to find the laboratory referred to by GLN/Alternate ID in the Business Account being uploaded to.	100	C
• **addressLine2**		100	
• **aptNumber**		10	
• **city**		50	
• **stateProvince**		100	
• **country**		50	
• **postalCode**		10	
• **phone**	Phone number of the laboratory.	20	C
• **email**	Email address of the laboratory.	50	C
• **citationCodes**	A valid product certificate requires at least one citation code; should be provided as an array.	N/A	M
• **testReportId**	The test report ID for the provided citations. If providing multiple values, separate them with a semi-colon.	400	O
• **testURL**	The URL where the test report can be viewed. If providing multiple values, separate them with a semi-colon.	400	O
• **testReportAccessKey**	The key/password required to view the test report. If providing multiple values, separate them with a semi-colon.	400	O
• **isComponent**	A Boolean field indicating whether the test information provided applies to a component part. Accepts "true"/"false" values as well as "Yes"/"No".	N/A	O
• **componentDescription**	A description of the component can be provided if isComponent is set to "true" or "yes".	250	O
exemptions	Any testing exemptions being claimed for the product; should be added as an array.	N/A	O
poc			
• **type**	Type of entity that maintains records of test results. Acceptable values: "Importer", "Manufacturer", "Laboratory", "Broker" or "Other". Use "Other" and provide address/contact information if Point of Contact is	N/A	M

		an individual or entity that does not fall into any of the provided classifications.		
•	**gln**	Either a Global Location Number (GLN) or a custom Alternate ID are required when providing a new point of contact of type "Other", or identifying an existing one.	50	C
•	**alternateId**		50	
•	**name**	The name of the point of contact.	100	C
•	**addressLine1**	The full address of a point of contact (POC) of type "Other" is required if it is being added to the Business Account for the first time. Adding a new POC also requires that "isNew" has a value of "Y" in the corresponding directives module. If "isNew" is left blank or set to "N", the system will attempt to find the point of contact referred to by GLN/Alternate ID in the Business Account being uploaded to. Do not provide address information if POC "type" is set to "Importer", "Manufacturer", "Laboratory" or "Broker".	100	C
•	**addressLine2**		100	
•	**aptNumber**		10	
•	**city**		50	
•	**stateProvince**		100	
•	**country**		50	
•	**postalCode**		10	
•	**phone**	The phone number of the point of contact. Only required if adding a POC of type "Other" for the first time.	20	C
•	**email**	The email address of the point of contact. Only required if adding a POC of type "Other" for the first time.	50	C

All Fields in Directives

Field Name	Description	Mandatory (M) / Conditional (C) / Optional (O)
productUpdate	Indicates whether the product identified in the preceding coreProduct module is an existing product being updated. Acceptable values are "Y" (for yes) or "N" (for no).	C
versionIdToUpdate	The version ID of the certificate being updated. Required if the value of productUpdate is "Y".	C
manufacturer		
• **isNew**	Used to indicate whether the manufacturer identified in the preceding coreProduct module is new to the destination Business Account. If left blank, value will be assumed to be "N".	C
• **gln**	Global Location Number (GLN) of the manufacturer; only mandatory if "isNew" has a value of "Y".	C
• **alternateId**	A unique Alternate Identifier for the manufacturer; only mandatory if "isNew" has a value of "Y".	C
labs		
• **isNew**	Used to indicate whether the laboratory identified in the preceding coreProduct module is new to the destination Business Account. If left blank, value will be assumed to be "N".	C
• **gln**	Global Location Number (GLN) of the laboratory; only mandatory if "isNew" has a value of "Y".	C
• **alternateId**	A unique Alternate Identifier for the laboratory; only mandatory if "isNew" has a value of "Y".	C
poc		
• **isNew**	Used to indicate whether the point of contact identified in the preceding coreProduct module is new to the destination Business Account. If left blank, value will be assumed to be "N".	C
• **gln**	Global Location Number (GLN) of the point of contact; only mandatory if "isNew" has a value of "Y".	C
• **alternateId**	A unique Alternate Identifier for the point of contact; only mandatory if "isNew" has a value of "Y".	C

eFiling API Specifications

Sample JSON Payload for Upload to the /import Endpoint

The JSON structure below includes all of the fields required to upload a valid certificate of compliance to a Product Collection using the eFiling Product Registry's API. However, please note that attempting to upload the below information exactly as it appears will result in an error message. This is because a laboratory of the type "ITL" has been included—indicating that the lab is a CPSC-Accepted laboratory—but a valid, 4-digit CPSC-ID has not been provided.

If you wish to test the API using the payload below, you will need to either remove the reference to that invalid laboratory or update its 4-digit CPSC-ID to a valid number. An up-to-date list of CPSC-Accepted laboratories, along with their corresponding IDs, can be found at https://cpsc.gov/cgi-bin/labsearch.

```
{
      "productList": [
        {
          "coreProduct": {
            "versionId": "V1",
            "primaryProductId": "SAMPLE-API-PRODUCT",
            "primaryProductIdType": "Alternate ID",
            "identifiers": [
              {
                "identifier": "123-GTIN-456",
                "identType": "GTIN"
              },
              {
                "identifier": "123-UPC-456",
                "identType": "UPC"
              },
              {
                "identifier": "123-SKU-456",
                "identType": "SKU"
              },
              {
                "identifier": "123-MODEL-456",
                "identType": "Model #"
              },
              {
                "identifier": "123-SERIAL-456",
                "identType": "Serial #"
              },
              {
                "identifier": "123-REGISTERED-456",
                "identType": "Registered #"
              },
              {
                "identifier": "",
                "identType": "Alternate ID"
              }
            ],
            "certificateType": "GCC",
            "name": "EXAMPLE NAME",
            "tradeBrandName": "EXAMPLE BRAND",
            "description": "EXAMPLE DESCRIPTION",
            "color": "EXAMPLE COLOR",
            "style": "EXAMPLE STYLE",
            "manufacturer": {
```

```json
    "gln": "",
    "alternateId": "EXAMPLE-MANUFACTURER",
    "sbmId": "",
    "name": "MANUFACTURER NAME",
    "addressLine1": "MANUFACTURER ADDRESS 1",
    "addressLine2": "MANUFACTURER ADDRESS 2",
    "aptNumber": "#123",
    "city": "CITY",
    "stateProvince": "STATE",
    "country": "USA",
    "postalCode": "ZIPCODE",
    "phone": "555-555-5555",
    "email": "EXAMPLE@MANUFACTURER.XY"
  },
  "manufactureDate": "01/2000",
  "productionStartDate": "01/01/2000",
  "productionEndDate": "01/02/2000",
  "lotNumber": "LOT-NUM",
  "lotNumberAssignedBy": "SELLER",
  "lastTestDate": "01/10/2000",
  "labs": [
   {
     "type": "ITL",
     "cpscId": "9999",
     "alternateId": "",
     "name": "",
     "addressLine1": "",
     "addressLine2": "",
     "aptNumber": "",
     "city": "",
     "stateProvince": "",
     "country": "",
     "postalCode": "",
     "phone": "",
     "email": "",
     "citationCodes": [
      "1215",
      "1216",
      "1217"
     ],
     "testReportId": "EXAMPLE-REPORT-ID",
     "testURL": "EXAMPLE-URL",
     "testReportAccessKey": "EXAMPLE-REPORT-ACCESS-KEY",
     "isComponent": true,
     "componentDescription": "EXAMPLE COMPONENT DESCRIPTION"
   },
   {
     "type": "LAB",
     "cpscId": "",
     "alternateId": "EXAMPLE-LAB",
     "name": "EXAMPLE LAB",
     "addressLine1": "EXAMPLE ADDRESS 1",
     "addressLine2": "EXAMPLE ADDRESS 2",
     "aptNumber": "SUITE",
     "city": "CITY",
     "stateProvince": "STATE",
     "country": "USA",
     "postalCode": "ZIP CODE",
```

```
          "phone": "555-555-5555",
          "email": "EXAMPLE@LAB.XY",
          "citationCodes": [
            "1201",
            "1202",
            "1207"
          ],
          "testReportId": "EXAMPLE-REPORT-ID",
          "testURL": "EXAMPLE-URL",
          "testReportAccessKey": "EXAMPLE-REPORT-ACCESS-KEY",
          "isComponent": "NO",
          "componentDescription": ""
        }
      ],
      "exemptions": [
        "1500.91(d)(5)",
        "1610.1(d)(1)"
      ],
      "poc": {
        "type": "IMPORTER",
        "gln": "",
        "alternateId": "",
        "name": "",
        "addressLine1": " ",
        "addressLine2": "",
        "aptNumber": "",
        "city": "",
        "stateProvince": "",
        "country": "",
        "postalCode": "",
        "phone": "",
        "email": ""
      }
    },
    "directives": {
      "productUpdate": "N",
      "versionIdToUpdate": "",
      "manufacturer": {
        "isNew": "Y",
        "gln": "",
        "alternateId": "EXAMPLE-MANUFACTURER"
      },
      "labs": [
        {
          "isNew": "Y",
          "gln": "",
          "alternateId": "EXAMPLE-LAB"
        }
      ],
      "poc": {
        "isNew": "N",
        "gln": "",
        "alternateId": ""
      }
    }
  }
 ]
}
```