Symantec™ Data Loss Prevention Detection REST API 2.0 Reference Guide

15.8

Last updated: April 13, 2021

Table of Contents

Introducing the Symantec Data Loss Prevention Detection REST API 2.0

About the Symantec Data Loss Prevention Detection REST API

The Symantec Data Loss Prevention Detection REST API enables you to develop REST clients that can connect to, and inspect the content of, specific applications that you identify for security monitoring through the Symantec Data Loss Prevention Cloud Detection Service or Symantec Data Loss Prevention API Detection for Developer Apps virtual appliance. Your REST API client sends sensitive data to Symantec Data Loss Prevention for inspection by way of the Cloud Detection Service or the API Detection for Developer Apps virtual appliance. Symantec Data Loss Prevention inspects the data and creates incidents where applicable. Symantec Data Loss Prevention also returns response action recommendations in the REST detection request response.

The Symantec Data Loss Prevention API Detection for Developer Apps virtual appliance provides detection capabilities deployed on-premises that enables you to monitor content and identify information policy violations in applications using a custom REST client.

> **NOTE**
>
> Hereafter in this guide, the "API Detection for Developer Apps virtual appliance" will be referred to as the "REST API Appliance."

The Symantec Data Loss Prevention Cloud Detection Service is a Symantec Data Loss Prevention detection service deployed in the cloud that enables you to monitor content and identify information policy violations in cloud applications. The Cloud Detection Service can inspect both network traffic ("data in motion," or DIM) and data stored in a repository ("data at rest," or DAR). You can specify which type of data you are posting for inspection using the `common.dataType` context entry in your detection request.

About the Detection REST API 2.0 Guide

This guide is a reference document intended for software developers creating REST clients for in-house or third-party cloud applications for use with Symantec Data Loss Prevention. It does not assume deep knowledge of Symantec Data Loss Prevention, though familiarity with the concepts of Symantec Data Loss Prevention may be helpful. Consult with your Symantec Data Loss Prevention administrator to learn more about Symantec Data Loss Prevention and how it processes the data you post.

The latest version of this guide is available at the Related Documents section of 15.8 Help.

Overview of the Symantec Data Loss Prevention Detection REST API 2.0

The following table describes some general features of the REST API 2.0.

Table 1: General features of the Symantec Data Loss Prevention Detection REST API 2.0

Feature	Description
Version	You specify the API version using the path prefix in the base URL. The base URL path for this version is `/v2.0/`.
HTTP methods	Version 2.0 supports the `POST` method only.

Feature	Description
Message serialization	Version 2.0 supports JSON-formatted messages only.
Transport security	Version 2.0 uses SSL/TLS to secure all connections.
Authentication	Version 2.0 uses client certificate authentication. Symantec provides you with a certificate to use with your REST client.
Connection longevity	The Cloud Detection Service maintains long-lasting connections to REST clients, letting you submit multiple detection requests using the same connection.
Caching	Version 2.0 uses standard HTTP caching for detection responses. You cannot cache detection requests.
Internationalization	Some response rules include user messages, such as the **Bounce Message to Sender** option of the **Network Prevent: Block SMTP Message** response rule action. The policy author configures these messages. They cannot be localized for each detection request. The policy author is responsible for providing a localized message in such cases.
Throttling	When your Cloud Detection Service is overloaded, it returns the HTTP error status code `503 Service Unavailable`.
Detection limits	When you exceed the detection request size limit, Symantec Data Loss Prevention returns the appropriate error in the response. The Symantec Data Loss Prevention administrator is responsible for specifying detection size limits in the Enforce Server administrative console.
Batch requests	Version 2.0 does not support batch requests.
Asynchronous requests	Version 2.0 does not support asynchronous requests.
Results paging	Version 2.0 does not support results paging.

Symantec DLP Detection REST API 2.0 Reference

Detection requests

This section describes the structure and usage of detection requests in the Symantec Data Loss Prevention Detection REST API 2.0.

URL

The detection request URL is `/v1.0/DetectionRequests`.

HTTP method

The detection request HTTP method is `POST`.

HTTP request headers

HTTP request headers describes the HTTP request headers for detection requests.

Table 2: HTTP request headers

HTTP header	Required?	Description
`Accept`	No	Specifies the accepted request format: `application/json`
`X-SYMC-DLP-CustomerID`	No	Specifies the Symantec Data Loss Prevention customer. This header is used when third parties post detection requests on behalf of a Symantec Data Loss Prevention customer.
`X-SYMC-DLP-DetectorID`	Yes	Specifies the target Cloud Detection Service cloud detector or or REST API Appliance. You may have several Cloud Detection Services deployed, each with different policies applied to them. This header lets you direct your detection request to the Cloud Detection Service with the appropriate policy.

HTTP body

The HTTP body consists of a single detection request.

Detection request format and definitions

This section describes and defines the parameters of a single detection request. The detection request specifies one field of the `ContextEntry` type and multiple fields of the `ContentBlock` type.

Context entries send attribute information to Symantec Data Loss Prevention. These attributes may affect which policies or response rules apply to the data you submit for inspection.

Content blocks send the data you want to inspect in a format appropriate for Symantec Data Loss Prevention policy detection. Each Symantec Data Loss Prevention policy may treat these content blocks differently, so your REST client must distinguish where the content appeared in the network traffic you send for inspection. For general document inspection, use the `attachments` content block.

```
DetectionRequest ::= SEQUENCE {
    context          SEQUENCE OF ContextEntry,
    subject          ContentBlock,
    body             ContentBlock,
    attachments      SEQUENCE of ContentBlock
}
```

Table 3: Detection request fields

Field	Required or optional?	Description
`context`	Optional	A list of context attributes submitted with the detection request. Context entries
`subject`	Required (at least one of `subject`, `body`, or `attachments`)	A content block submitted with the detection request. Content blocks
`body`	Optional (at least one of `subject`, `body`, or `attachments`)	A content block submitted with the detection request. Content blocks
`attachments`	Optional (at least one of `subject`, `body`, or `attachments`)	A series of content blocks submitted with the detection request. Content blocks

Context entries

This section describes the context attributes that you can submit with your detection request.

```
ContextEntry ::= SEQUENCE {
    name          ENUMERATED UTF8String,
    value         SEQUENCE OF UTF8String
}
```

Table 4: Context entry fields

Field	Required or optional?	Description
`name`	Required	The name of the context parameter. For a list of the currently defined context parameter names, see Context entry parameter names.
`value`	Required	The value of the context parameter. Values with known mappings to Symantec Data Loss Prevention message attributes are automatically mapped to those attributes. Values for which no known mapping exists are added to the message envelope as key/value pairs. Standard text-based detection rules can then be used to match on those keys and their values.

Table 5: Context entry parameter names

Name	Description
`client.domain`	A string representing the domain of the REST client making a detection request.
`client.user.id`	A string representing the identifier for the user within the client domain making the detection request.
`common.application`	Required. A string representing the application name that is specific to Gatelets and Securlets. For example, "securlet.box' or "gatelet.box."
`common.application.reportName`	A string representing the application name as it appears in reports, such as "Box."
`common.authrecipient`	A string representing the authenticated recipient of the message, if known.
`common.authsender`	A string representing the authenticated sender of the message. For example, an endpoint user, an authenticated HTTP proxy user, an authenticated SMTP user, and so on.
`common.created`	A string representing the ISO 8601 timestamp when the file was created. For example, 2015-10-13T10:11:06.419Z. Used in DAR requests only.
`common.dataType`	Required. Specifies the type of data in the data request. This must be one of two values: `DAR` ("data at rest") or `DIM` ("data in motion").
`common.description`	A string representing the description field of the file. Used in DAR requests only.
`common.doc.activityCount`	A long integer representing the number of user actions on the document.
`common.doc.creatorId`	A string representing the unique identifier of the document creator.

Name	Description
common.doc.exposed	A Boolean value indicating if the document is shared or accessible. The document is considered exposed if it is shared with or accessible to everyone within your organization, or shared with or accessible to anyone outside of your organization. If the document is only shared with certain members of your organization, it is not considered an exposed document.
common.doc.exposures.allInternal	A Boolean value indicating if the document is shared with or accessible to everyone within your organization.
common.doc.exposures.externalCollaborators	A list of email addresses of people outside your organization that have access to the document.
common.doc.exposures.internalCollaborators	A list of email addresses of people inside your organization that have access to the document.
common.doc.exposures.public	A Boolean value indicating if the document is shared with or accessible to everyone outside your organization. Such documents are available to everyone on the internet.
common.doc.id	A string representing the unique identifier for the document in the SaaS application.
common.doc.isInternal	A Boolean value indicating if the document is "internal." A document is considered internal if it was created by a member of your organization.
common.doc.parentFolderId	A string representing the unique identifier of the parent folder containing the document.
common.doc.type	A string representing the type of document, such as "file" or "folder."
common.expectActionsAck	A Boolean value indicating if the Enforce server should expect an action acknowledgement for any responses.
common.filter	Required. A list of filter identifiers associated with a given scan.
common.folder	A string representing the name of the folder containing the files or attachments.
common.job.id	A string representing the identifier of an on-demand scan.
common.lastAccessed	A string representing the ISO 8601 timestamp when the file was last accessed. For example, 2015-10-13T10:11:06.419Z. Used in DAR requests only.
common.lastModified	A string representing the ISO 8601 timestamp when the file was last modified. For example, 2015-10-13T10:11:06.419Z. Used in DAR requests only.
common.log.id	A string representing the unique identifier for the log.
common.messageSource	A string that specifies the source of a message when application names overlap. For example, this parameter would specify whether a Box incident came from the Box Gatelet or Box Securlet.
common.owner	A string representing the user identification of the data owner. Used in DAR requests only.
common.service.classification	A string representing the Shadow IT service classification, such as "Sanctioned."
common.service.score	An integer representing the Shadow IT score for the service associated with the detection request. For example, Box has a Shadow IT service score of 80.

Name	Description
common.sharedWith	An array of user IDs for all users the file is shared with. Used in DAR requests only.
common.sharepoint	A string representing the SharePoint site name.
common.sharingUrl	A string representing the URL used to share the document. Used in DAR requests only.
common.tag	A string representing the tag field of the file. Used in DAR requests only.
common.transactionId	The transaction identifier used to link back to the incident on an external console, such as CloudSOC.
common.user.activityType	A string representing the activity performed by the user, such as "add" or "delete."
common.user.docsExposedCount	A long integer representing the number of documents exposed to the user.
common.user.groupMembership	A list of groups in the SaaS application that include the user.
common.user.id	A string representing the unique identifier of the user.
common.user.isInternal	A Boolean value indicating if the user is a member of your organization.
common.user.name	A string representing the name of the user as displayed in reports.
common.user.threatScore	A long integer representing the threat score associated with the user or event.
custom	A custom context attribute.
device.isCompliant	A Boolean value indicating if the device is compliant with your mobile device management policy.
device.isPersonal	A Boolean value indicating if the device is owned and managed by the user.
device.isTrustedDevice	A Boolean value indicating if the detection request came from a trusted device.
device.isUnmanaged	A Boolean value indicating if the device is not managed by your mobile device management system.
device.os	A string specifying the device's operating system.
device.type	A string specifying the type of device.
email.envelope.recipient	The envelope recipient for an email message.
email.envelope.sender	The envelope sender for an email message.
email.header.recipient	The header recipient for an email message.
email.header.sender	The header sender for an email message.
http.browser	A string representing the name of the web browser, as determined by the user agent.
http.cookies	HTTP request cookies.
http.method	Specifies the method used in the HTTP traffic submitted for inspection.
http.siteClassification	Specifies the classification of the site, such as "Social Media."
http.siteRiskScore	A numerical value indicating the risk level of the target site.
http.url	The target URL of the HTTP traffic submitted for inspection.

Name	Description
`http.userAgent`	The user agent supplied by the user's device or browser for the HTTP traffic submitted for inspection.
`link.doc.exposure`	A string representing a link to the CloudSOC console **Exposures** panel for the document.
`link.incident`	A string representing a link to the CloudSOC **Policy Alert** corresponding to the incident.
`link.service.application`	A string representing a link to the CloudSOC **Service Visibility** panel for that user.
`link.service.file.activity`	A string representing a link to the application specific activities list in CloudSOC, filtered by the document name.
`link.uba`	A string representing a link to the CloudSOC **Investigate** panel. The **Investigate** panel displays a list of the activities performed by the user for the last seven days, filtered by application and username.
`link.user.exposures`	A string representing a link to the CloudSOC **Securlets** console. The **Securlets** console displays the **Exposures** panel for the user.
`link.user.threatTree`	A string representing a link to the CloudSOC **Threat Tree** for the user.
`location.coords.latitude`	The geographic latitude of the device.
`location.coords.longitude`	The geographic longitude of the device.
`location.isInsideOffice`	Boolean value indicating if the data originated from or resides on a device inside your office.
`location.region`	A string representing the location from which the action was performed, in the format "City(Country)."
`location.region.country`	A string representing the country where the activity was performed.
`network.direction`	Specifies if the DIM detection request is for content upload or download. Must be one of two values: `Upload` or `Download`. Used in DIM requests only.
`network.protocol`	Specifies the OSI Level 7 network protocol for the detection request. For example, SMTP, HTTP, FTP, and so on. Used in DIM requests only.
`network.recipient.ip`	The IP address of the message recipient.
`network.recipient.port`	The network port of the message recipient.
`network.sender.ip`	The IP address of the message sender.
`network.sender.port`	The network port of the message sender.

Obtaining the `common.filter` identifier

The `common.filter` context entry parameter identifies the Application Detection configuration associated with your detection request. For detailed information on Application Detection configurations, see the *Symantec Data Loss Prevention Administration Guide* or the online Help.

You can obtain the `common.filter` identifier from the **Manage > Application Detection > Configuration > Edit Configuration** page in the Enforce Server administrative console.

To obtain the `common.filter` identifier

1. In the Enforce Server administrative console, navigate to **Manage > Application Detection > Configuration**.
2. Optional: If you have not already done so, create and save a **Cloud Detection API Service** Application Detection configuration for your application.
3. Click the name of the appropriate Application Detection configuration on the **Manage > Application Detection > Configuration** list page.

   The **Manage > Application Detection > Configuration > Edit Configuration** page appears.
4. Copy the value in the **ID** field. This value is the `common.filter` identifier.

Content blocks

This section describes the fields of the content blocks that you can submit with your detection request.

```
ContentBlock ::= SEQUENCE {
    contentBlockId     UTF8String,
    mimeType           UTF8String,
    characterEncoding  UTF8String,
    extracted          Boolean,
    fileName           UTF8String,
    fileType           UTF8String,
    originalSize       Integer,
    data               Base64 encoded data
}
```

Table 6: Content block fields

Field	Required or optional?	Description
`contentBlockId`	Required	A client-defined ID that uniquely identifies the content block within the scope of the detection request. This ID is used for response action targeting.
`mimeType`	Required	Specifies the MIME type of the base64-decoded content block.
`characterEncoding`	Optional	Specifies the character set encoding used by the base64-decoded content block.
`extracted`	Optional	Boolean value indicating that the data in the content block has already had its content extracted. In such a case, the Cloud Detection Service or REST API Appliance will not perform content extraction. If this field is omitted, the Cloud Detection Service or REST API Appliance performs content extraction.
`fileName`	Optional	Specifies the name of the file in the content block.
`fileType`	Optional	Specifies the file type of the file in the content block. Use this field only for content that you have already extracted. Supported file types

Field	Required or optional?	Description
`originalSize`	Optional	An integer specifying the original size of the file included in the content block, in bytes. Use this field only for content that you have already extracted.
`data`	Required	The base64-encoded file content you want to submit for detection.

Sample request

This is a sample detection request:

```
POST /v2.0/DetectionRequests HTTP/1.1
User-Agent: curl/7.35.0
Host: 10.1.2.3
Content-Type: application/json
Accept: application/json

{
 "context" : [
  {"name": "common.dataType",              "value": ["DIM"]},
  {"name": "common.application",           "value": ["securlet.googledrive"]},
  {"name": "email.envelope.sender",        "value": ["slava@myco.com"]},
  {"name": "email.envelope.recipient",     "value": ["joe@example.com",
 "bob@example.com"]},
  {"name": "email.header.sender",          "value": ["slava@myco.com"]},
  {"name": "email.header.recipient",       "value": ["joe@example.com",
 "bob@example.com"]},
  {"name": "location.region",              "value": ["United States"]},
  {"name": "location.region.country",      "value": ["US"]},
  {"name": "http.url",                     "value": ["http://google.com"]},
  {"name": "link.user.exposures",          "value": ["http://google.com/
userexposures"]},
  {"name": "link.uba",                     "value": ["http://google.com/uba"]},
  {"name": "link.doc.exposure",            "value": ["http://google.com/docexposures"]},
  {"name": "link.service.file.activity",   "value": ["http://google.com/
servicefileactivity"]},
  {"name": "link.incident",                "value": ["http://google.com/incident"]},
  {"name": "link.service.application",     "value": ["http://google.com/
serviceapplication"]},
  {"name": "common.transactionId",         "value": ["a32cc030-9776-45ce-
ba55-84f9f5afe009"]},
  {"name": "common.user.name",             "value": ["My Favorite User"]},
  {"name": "common.doc.exposed",           "value": ["True"]},
  {"name": "common.doc.exposures.public",  "value": ["True"]},
  {"name": "common.user.threatScore",      "value": ["99"]},
  {"name": "common.doc.type",              "value": ["folder"]},
  {"name": "common.user.docsExposedCount", "value": ["3"]},
  {"name": "common.doc.creatorId",         "value": ["321"]},
  {"name": "common.doc.parentFolderId",    "value": ["123"]},
  {"name": "http.method",                  "value": ["GET"]},
  {"name": "http.cookies",                 "value": ["G123213ET"]},
```

```
   {"name": "device.type",                  "value": ["mobile"]},
   {"name": "http.siteRiskScore",           "value": ["66"]},
   {"name": "common.user.activityType ",    "value": ["create"]},
   {"name": "http.browser ",                "value": ["IE"]},
   {"name": "common.filter",                "value": ["69132E5E-732B-42AB-89C5-
C18B4A82434D"]},
   {"name": "common.expectActionsAck",      "value": ["true"]}
    ],

  "subject":
  {"contentBlockId":  "block1",
      "mimeType": "text/plain",
      "data": "c2VjcmV0"}
}
```

Input validation

Detection requests are validated before the Cloud Detection Service or REST API Appliance submits them to Symantec Data Loss Prevention for detection. Some validation errors are fatal. Detection requests with fatal validation errors are not submitted to Symantec Data Loss Prevention for detection. The Cloud Detection Service or REST API Appliance will return an HTTP error with an error message body describing the problem.

If your detection request has a minor problem with validation, the Cloud Detection Service or REST API Appliance submits the content to Symantec Data Loss Prevention for detection, and it returns a warning in the detection result.

All context entries are validated against the following constraints:

- Entry has a non-null name. Failure to validate against this constraint results in a fatal error.
- Entry has a non-null value. Failure to validate against this constraint results in a fatal error.
- Values do not exceed a configuration length. Failure to validate against this constraint results in a warning.
- Entry is checked for whether or not it is allowed to have multiple values. Failure to validate against this constraint results in a warning.

In addition to these validations, the following table specifies context entries that are subject to additional validation:

Table 7: Additional context entry validations

Context entry	Multiple values allowed	Match value against list	Range check	Numeric characters only	DIM/DAR specific	Date format check
`common.authrecipient`	Yes					
`common.created`					Yes	Yes (Fatal error)
`common.dataType`		Yes (Fatal error)				
`common.description`					Yes	
`common.lastAccessed`					Yes	Yes (Fatal error)
`common.lastModified`					Yes	Yes (Fatal error)
`common.sharedWithList`	Yes				Yes	
`common.sharingUrl`					Yes	

Context entry	Multiple values allowed	Match value against list	Range check	Numeric characters only	DIM/DAR specific	Date format check
common.tag					Yes	
common.doc.exposures.externalCollaborators	Yes					
common.doc.exposures.internalCollaborators	Yes					
common.user.groupMembership	Yes					
email.envelope.recipient	Yes					
email.header.recipient	Yes					
http.siteRiskScore	Yes			Yes		
network.direction		Yes (Fatal error)				
network.recipient.port	Yes		Yes	Yes		
network.sender.port			Yes	Yes		

Detection results

This section describes the content of detection results sent from the Cloud Detection Service or REST API Appliance back to your REST client.

A detection request may result in zero to many policy violations. Each policy violation may indicate one or more response actions that the Symantec Data Loss Prevention policy indicates that your organization should apply. Response actions are included in the detection result for informational purposes only. Someone in your organization must carry out the indicated response action to comply with your data loss prevention policies.

HTTP response headers

The following table describes the HTTP response headers for a detection result response.

Table 8: HTTP response headers

HTTP header	Description
cache-control	This header is the standard HTTP caching header.
pragma: no-cache	This header is identical to cache-control: no-cache
WWW-Authenticate: Basic realm="realm"	Indicates that basic authentication is required, in compliance with HTTP standards. The Cloud Server Connector returns the realm Enforce.

HTTP response codes

The detection response will include one of these response codes:

Success response code:

- 201 Created

Error response codes:

- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 408 Request timeout
- 503 Service Unavailable

Detection result format and definitions

This section describes and defines the format of a single detection result. The detection result consists of four fields: `requestId`, `violation`, `responseAction`, and `warning`.

```
DetectionResult ::= SEQUENCE {
    requestId           UTF8String,
    violation           SEQUENCE OF Policy,
    responseAction      SEQUENCE OF ResponseAction,
    warning             SEQUENCE OF Warning
 }
```

Table 9: Detection result fields

Field	Description
`requestId`	A unique identifier for the detection request. The Cloud Service Connector assigns an ID to each detection request. If the detection request results in a policy violation incident, the Cloud Service Connector passes the `requestId` to Symantec Data Loss Prevention. You can use the `requestId` in reports in the Enforce Server administration console to correlate your REST client detection requests and Symantec Data Loss Prevention incidents.
`violation`	An unordered list of violated policies, if applicable.
`responseAction`	A list of response actions indicated by the violated policies, if applicable. Your incident responder should apply these response actions in the indicated priority order.
`warning`	A list of warnings, if applicable.

Policy

This section describes the fields in the list of violated policies that may appear in your detection result.

```
Policy ::= SEQUENCE {
    policyId      UTF8String,
    name          UTF8String
}
```

Table 10: Policy fields

Field	Description
`policyId`	The identifier of the violated policy.
`name`	The descriptive name of the violated policy.

Response action

This section describes the fields in the list of response actions that may appear in your detection result.

```
ResponseAction ::= SEQUENCE {
    action              ENUMERATED UTF8String,
    priority            INTEGER,
    parameter           SEQUENCE OF ResponseActionParameter
}
```

Table 11: Response action fields

Field	Description
`action`	An enumerated list of response actions. Action descriptions
`priority`	The response rule execution priority.
`parameter`	A list of response action parameters. This list varies according to the response action. Response action parameter descriptions

Table 12: Action descriptions

Action	Description
`block`	The policy indicates that you should block the file and display an error to the user. Required parameters: `message`. Optional parameters: `contentBlockId`. Applies to DIM detection requests.
`breaklinks`	The policy indicates that you should break the links in the content. Required parameters: none. Optional parameters: `contentBlockId`, `customResponsePayload`. Applies to DAR detection requests.
`custom`	The policy indicates that a custom response rule should be applied. You must interpret what the custom response rule should be based on the response action parameters. Required parameters: none. Optional parameters: `contentBlockId`, `customResponsePayload`. Applies to both DAR and DIM detection requests.
`delete`	The policy indicates that you should delete the content. Required parameters: none. Optional parameters: `contentBlockId`. Applies to DAR detection requests.
`drm`	The policy indicates that data rights management should be applied to the content. Required parameters: none. Optional parameters: `contentBlockId`, `customResponsePayload`. Applies to both DAR and DIM detection requests.
`encrypt`	The policy indicates that you should encrypt the content. Required parameters: none. Optional parameters: `contentBlockId`, `customResponsePayload`. Applies to both DAR and DIM detection requests.
`quarantine`	The policy indicates that you should quarantine or move the content. Required parameters: none. Optional parameters: `contentBlockId`, `customResponsePayload`. Applies to both DAR and DIM detection requests.

Action	Description
`redact`	The policy indicates that you should replace the content with the specified message. You can choose whether or not to display an error to the user. Required parameters: `message`. Optional parameters: `contentBlockId`. Applies to DIM detection requests.
`tag`	The policy indicates that you should tag the content. Required parameters: none. Optional parameters: `contentBlockId`, `customResponsePayload`. Applies to DAR detection requests.

Response action parameters

This section describes the fields and content of the response action parameters that may appear in the response action section of your detection result.

```
ResponseActionParameter ::= SEQUENCE {
    name          UTF8String,
    value         SEQUENCE of UTF8String
}
```

Table 13: Response action parameter fields

Field	Description
`name`	The name of the response action. Response action parameter descriptions
`value`	A list of values for the response action.

Table 14: Response action parameter descriptions

Response action parameter	Description
`contentBlockId`	The identifier of the content block that violates the policy. This parameter may have multiple values if the response rule applies to multiple content blocks.
`customResponsePayload`	A custom response parameter that may be configured with the response action. This custom parameter has no meaning to the Cloud Service Connector. For example, you might configure a quarantine location for your organization to use for content that triggers the quarantine response action.
`message`	A message that you can display to your users.

Warning

This section describes the fields in the list of warnings that may appear in your detection response. Warnings are returned when there are issues with the content you posted for detection, but the content was submitted for detection regardless.

```
Warning ::= SEQUENCE {
    messageId    UTF8String,
    fieldName    UTF8String,
    message      UTF8String
}
```

Table 15: Warning field descriptions

Warning field	Description
`messageId`	An identifier for the validation warning type.
`fieldName`	The issue in the detection request that triggered the warning.
`message`	A detailed description of the warning.

Error messages

If your detection request could not be submitted for detection, the detection response will include an error message.

```
Error ::= SEQUENCE {
    messageId    UTF8String,
    message      UTF8String
}
```

Table 16: Error message field descriptions

Error message field	Description
`messageId`	An identifier for the error message that uniquely identifies the error condition.
`message`	A detailed description of the error.

Sample response

This is a sample detection response:

```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: 250
Date: Wed, 23 Apr 2014 01:56:05 GMT

{
  "requestId": "e402973a-5254-40ba-a725-84b2af6e58aa",
  "violation": [
       {"policyId": "pid12345689", "name": "Company Confidential Policy"},
       {"policyId": "pid00000099", "name": "PCI Policy"}
   ],
   "responseAction": [
       { "action": "redact",
         "priority":  1,
         "parameter" : [
              { "name": "contentBlockId", "value": ["block2"] },
              { "name": "message",        "value": ["The content was removed due to a
 loss
                                                    prevention policy violation"] }
           ]
       }
   ]
   "warning": [
       {
```

```
            "messageId": "conflicting-fields",
            "fieldName": "attachments",
            "message": "fileType should be specified for extracted data"
        }
    ]
}
```

Action acknowledgement requests

When a client receives a response action, it can optionally provide an action acknowledgement to the Cloud Detection Service or REST API Appliance. The action acknowledgement indicates the action taken, the result of the action, the time the action was performed, and can include an optional payload.

This section describes the structure and usage of action acknowledgements in the Symantec Data Loss Prevention Detection REST API 2.0.

For automated remediation actions, the client should only provide an action acknowledgement once per `requestId` or `transactionId`. For manual remediation actions, the action acknowledgement can be provided more than once per `requestId` or `transactionId`.

URL

The action acknowledgement URL is `/v2.0/ActionsAcknowledge`.

HTTP Method

The action acknowledgement HTTP method is `POST`.

HTTP body

The HTTP body consists of a single detection request.

Action acknowledgement request format and descriptions

This section describes the fields in the client acknowledgement of a response action.

```
ActionsAcknowledgement ::= {
     requestId          UTF8String,
     transactionId      UTF8String OPTIONAL if requestId is supplied,
     actionsTaken       SEQUENCE OF ActionsTaken,
 }
```

Table 17: Action acknowledgement field descriptions

Action acknowledgement field	Description
`requestId`	A unique identifier for the detection request. The Cloud Detection Service or REST API Appliance assigns an ID to each detection request. If the detection request results in a policy violation incident, the Cloud Detection Service passes the `requestId` to Symantec Data Loss Prevention. You can use the `requestId` in reports in the Enforce Server administration console to correlate your REST client detection requests and Symantec Data Loss Prevention incidents.
`transactionId`	A unique identifier for the request transaction. The REST client assigns a transaction identifier to each detection transaction. The `transactionId` field is optional if the `requestId` is included in the action acknowledgement. If the action acknowledgement does not include a `requestId`, it must contain a `transactionId`. The `transactionId` must also be included in the original detection request.
`actionsTaken`	A sequence indicating the action taken, the result of the action, the time the action was performed, and an optional payload.

Actions taken

This section describes the values in the `actionsTaken` field of an action acknowledgement.

```
ActionsTaken::= SEQUENCE {
     action             UTF8String,
     result             UTF8String,
     payload            SEQUENCE of UTF8String OPTIONAL,
     timestamp          UTF8String in ISO8601 format
  }
```

Table 18: Actions taken field descriptions

Field	Description
`action`	A description of the action taken.
`result`	The result of the action.
`payload`	An optional custom payload.
`timestamp`	A timestamp in ISO8601 format.

Sample action acknowledgment request

This is a sample action acknowledgement request (line breaks added for legibility):

```
POST /v2.0/ActionsAcknowledge HTTP/1.1
 User-Agent: curl/7.35.0
 Host: 10.1.2.3
 Content-Type: application/json
 Accept: application/json
   {
   "requestId" : "56c55b54-c4fa-4a38-ad7e-3a106f746d09",
   "transactionId" : "a32cc030-9776-45ce-ba55-84f9f5afe009",
   "actionsTaken": [
     {
       "action": "quarantine",
```

```
      "result": "failure",
      "timestamp": "2015-10-14T10:11:06.419Z"
    },
    {
      "action": "encrypt",
      "result": "success",
      "payload": {
        "key1" : "https://abc.xyz.com/auth/admin/index.html#/documentDetails/
                  058b634e-8918-4440-ba4f-0de62d3017b4/",
        "key2" : "https://somwhere.pgp.com"
        },
      "timestamp": "2015-10-14T10:11:06.419Z"
    }
  ]
}
```

Supported File Types

The following sections list the file types that are supported for detection by the Symantec Data Loss Prevention REST API.

Word processing file types supported for REST API 2.0 detection

- act
- adobe_maker
- aes
- aldus_pagemaker
- amipro
- applix_words
- apple_pages
- ascii
- cdf
- comet
- dca_rft
- display_write
- doc
- docuworks
- folio_flat
- health_level7
- hwp
- html
- macwrite
- mswrite
- multimate
- oasys
- odt
- omni_outliner
- onenote
- rtf
- sgml
- unicode
- word_pro
- wordperfect
- wordstar
- works
- writenow
- xml
- xywrite

Multimedia file types supported for REST API 2.0 detection

- aiff
- ac3_audio
- asf
- macromedia_flash
- macromedia_dir
- midi
- mp3
- mpeg_movie
- qt
- quickdraw
- realaudio
- realmedia
- riff
- video_win
- vrml
- wav
- wma
- wmv

Spreadsheet file types supported for REST API 2.0 detection

- 123
- applix_spread
- apple_numbers
- csv
- mod
- ods
- quattro_pro
- sylk
- works_spread
- xls
- excel_macros

Presentation file types supported for REST API 2.0 detection

- apple_keynote
- corel_pres
- lotus-fg
- odp
- pdf
- ppt
- pr2
- xfdl
- xps

Image file types supported for REST API 2.0 detection

- ami_draw
- app_graph
- bmp
- cad_draw
- cat
- cdd
- cdr
- cgm
- ch3
- dicom
- dwg
- drw
- emf
- enc_ps
- fax_sys
- freehand
- gif
- hpg
- ico
- jpg
- ms_drawing
- nur
- pcx
- pic
- pict
- pm_mf
- png
- pntg
- ps
- sgi_img
- solid_works
- svf
- targa
- tiff
- visio
- wmf
- wpg

Encapsulation file types supported for REST API 2.0 detection

- 7zip
- binhex
- bkf
- bzip2
- cab
- compress
- cpio
- eml
- emx
- encase
- gz
- iso
- lzh
- lotus-dxl
- lotus-nsf
- msg
- onm
- pex
- rar
- scrap
- shar
- stuffit
- tar
- tnef
- uu
- yenc
- zip

Encryption file types supported for REST API 2.0 detection

- encrypted_doc
- encrypted_nero
- encrypted_pdf
- encrypted_ppt
- encrypted_xls
- encrypted_zip
- open-pgp
- pgp
- pgpnetshare

Other file types supported for REST API 2.0 detection

- access
- dbf
- exe
- exe_unix
- fm
- frame
- help
- macbin
- paradox
- pcl
- proj
- publ
- qxpress
- smtp
- wcm
- works_db

Copyright statement


Symantec
A Division of Broadcom