CylanceOPTICS™

API Guide

CYLANCE

Product: CylanceOPTICS API

Document: CylanceOPTICS API Guide. This guide is a succinct resource for analysts, administrators, and customers who are reviewing or evaluating the product.

Document Release Date: v2.0 rev 3, December 2018

About Cylance: Cylance is the first company to apply artificial intelligence, algorithmic science, and machine learning to cybersecurity and improve the way companies, governments, and end-users proactively solve the world's most difficult security problems. Using a breakthrough predictive analysis process, Cylance® quickly and accurately identifies what is safe and what is a threat, not just what is in a blacklist or whitelist. By coupling sophisticated math and machine learning with a unique understanding of an attacker's mentality, Cylance provides the technology and services to be truly predictive and preventive against advanced threats. For more information, visit cylance.com.

Global Headquarters

400 Spectrum Center Drive, Irvine, CA 92618

Consulting Services Hotline

+1-877-97DEFEND ▪ +1-877-973-3336

Corporate Contact

+1-914-CYLANCE ▪ +1-914-295-2623

Email

partnersupport@cylance.com

Website

https://www.cylance.com

To Open a Support Ticket

https://support.cylance.com - Click on **Submit a Ticket**

To View Knowledge Base and Announcements

Login to https://support.cylance.com

To Request a Callback from Cylance Support

+1-866-699-9689

Contents

Overview

Cylance offers CylanceOPTICS APIs as an alternative way of interacting with the system.

Note: This guide covers the CylanceOPTICS API calls only. For information about configuring custom applications in the Cylance Console, see the Cylance User API guide.

RESTful API

About Device ID

When attempting to query a CylanceOPTICS API call that utilizes a Device ID value, please be aware of the following:

The format for the CylanceOPTICS API Device ID value should be:

CylanceOPTICS Example:

45E07F34E76B4A9EB167D6D0C510D6BA (upper case without dashes)

Passing the Device ID value as the CylancePROTECT format will return an HTTP 200 Status, as if the call was successful, but you will receive an incorrect response.

CylancePROTECT Example:

45e07f34-e76b-4a9e-b167-d6d0c510d6ba (lower case without dashes)

To obtain the Device ID, you must query the CylancePROTECT API and then format the ID to match the CylanceOPTICS format mentioned above.

This query can be found in the CylancePROTECT User API Guide under the section titled "Device API". Use the "Get Devices" and "Get Device" queries from the guide. The Device ID value is the field called: "id".

CylanceOPTICS Detection

The CylanceOPTICS Detection API allows users to interact with Detection Events triggered by the CylanceOPTICS Context Analysis Engine (CAE). CAE allows users to take automated response actions against malicious or suspicious behavior detected on endpoints utilizing both machine learning models and static behavior-based rules.

The CylanceOPTICS Detection API enables further automation of analyzing, triaging, and responding to malicious or suspicious activity prevented or detected by CylanceOPTICS. The workflows currently available through this API include:

- Gathering a summary Detection Events that have occurred in a tenant including a Detection Event's ID, severity, description, occurrence time, associated device, and status.
- Gathering the specific Detection Details of Detection Events that have occurred in a tenant including the artifacts associated with a Detection Event, the status of automated response actions that have been taken against a Detection Event, and other granular details that compose the Detection Event.
- Deleting a single or multiple Detection Events from a tenant.
- Updating a Detection Event's status and comments in a tenant.

Get Detections

Allows a caller to request a page with a list of detections belonging to a tenant, sorted in descending order (most recent detection listed first). The page number and page size parameters are optional. When the values are not specified, the default values are 1 and 20 respectively.

Service Endpoint	North America: https://protectapi.cylance.com/detections/v2?page=m&page_size=n US Government: https://protectapi.us.cylance.com/detections/v2?page=m&page_size=n All Other Regions: https://protectapi-{region-code}.cylance.com/detections/v2?page=m&page_size=n
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsdetect:list scope encoded.
Request	Append the following optional query string parameters: • start: The start date-time of the query range. • end: The end date-time of the query range. • severity: The detection severity filter. Supports OR filters via multiple queries. Possible values are: ○ Informational ○ Low ○ Medium ○ High • detection_type: The detection type filter. Supports OR filters via multiple queries. • detected_on: The detection on filter. Supports OR filters via multiple queries. • event_number: The event number filter. Supports OR filters via multiple queries. • device: The device name filter. Supports OR filters via multiple queries. • status: Possible values are: ○ New ○ In Progress ○ Follow Up

    - Reviewed
    - Done
    - False Positive
- page: The page number to request. Defaults to 1.
- page_size: The number of detection records to retrieve per page. Defaults to 20.
- sort: Sort by the following fields (adding “-“ in front of a value denotes descending order):
    - Severity
    - OccurrenceTime
    - Status
    - Device
    - PhoneticId
    - Description

Response

200 OK

Get Detections Response Schema

```
{
  "page_number": 0,
  "page_size": 0,
  "total_pages": 0,
  "total_number_of_items": 0,
  "page_items": [
   {
    "Id": "string",
    "PhoneticId": "string",
    "Severity": "string",
    "DetectionDescription": "string",
    "OccurrenceTime": "2018-06-14T03:05:45.866Z",
    "Device": {
     "name": "string",
     "CylanceId": "string"
    },
    "Status": "string"
   }
  ]
}
```

400 BadRequest – Returned for the following reasons:

- The tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The page number or page size specified are less than or equal to zero.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

	404 NotFound – The detection resources page requested does not exist. 500 InternalServerError – An unforeseeable error has occurred. 503 Server Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
page_items	The list of detections belonging to the requested page, each displaying the following information: • DetectionDescription: The description of the detection. • Device: ○ CylanceId: The ID for the device. ○ Device: The name of the device. • Id: The unique ID for the detection. • OccurrenceTime: The time at which the detection occurred. • PhoneticId: The easy-to-read version of the ID that is probabilistically unique. • Severity: The criticality of an observance of a detection. • Status: The status of the detection workflow.
page_number	The page number requested.
page_size	The page size requested.
total_number_of_items	The total number of resources.
total_pages	The total number of pages that can be retrieved based on the page size specified.

Update Detection

Allows a caller to update the status and comment fields for an existing detection for a tenant.

Service Endpoint	North America: https://protectapi.cylance.com/detections/v2 US Government: https://protectapi.us.cylance.com/detections/v2 All Other Regions: https://protectapi-{region-code}.cylance.com/detections/v2
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsdetect:update scope encoded.

Request	Post Detection Request Schema
	[
	{
	"detection_id": "string",
	"field_to_update": {
	"status": "string",
	"comment": "string"
	}
	}
	]

Response	200 OK – Successful update
	Post Detection Response Schema
	[
	{
	"detection_id": "string",
	"response": "string"
	}
	]
	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.
	403 Forbidden – The JWT token did not contain the proper scope to perform this action.
	404 Not Found – No such detection.

	500 InternalServerError – An unforeseeable error has occurred.

The request JSON schema contains the following fields:

Field Name	Description
comment	The comment on the detection.
status	The status of the detection. • Done: All actions are complete for this detection. • False Positive: The detection is considered a false positive. • Follow Up: This detection requires someone to follow-up on it. • In Progress: The detection is currently being reviewed and worked on. • New: The detection is new. • Reviewed: The detection has been reviewed, but no actions have been taken.

Get Detection

Allows a caller to request a specific detection resource belonging to a tenant. Use Get Detections to obtain the unique detection ID.

Service Endpoint	North America: https://protectapi.cylance.com/detections/v2/{unique_detection_id}/details US Government: https://protectapi.us.cylance.com/detections/v2/{unique_detection_id}/details All Other Regions: https://protectapi-{region-code}.cylance.com/detections/v2/{unique_detection_id}/details
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsdetect:read scope encoded.
Request	None

Response	200 OK
	Get Detection Response Schema
	{
	  "Id": "string",
	  "ActivationTime": "2018-06-14T21:38:11.543Z",
	  "AppliedExceptions": "Unknown Type: list",
	  "AssociatedArtifacts": [
	    {}
	  ],

```
"Comment": "string",
"DetectionRule": {},
"Detector": {},
"Device": {
  "name": "string",
  "id": "string"
},
"Name": "string",
"OccurrenceTime": "2018-06-14T21:38:11.543Z",
"Product": {},
"PhoneticId": "string",
"ReceivedTime": "2018-06-14T21:38:11.543Z",
"SchemaVersion": 0,
"Severity": "string",
"SeveritySortLevel": 0,
"Status": "string",
"StatusSortLevel": 0,
"TenantId": "string",
"ZoneIds": [
  "string"
],
"Trace": {},
"Context": {},
"InvolvedArtifacts": {},
"Responses": [
  {
    "Status": "New",
    "Comment" : "string",
    "TenantId" : "string",
    "PhoneticId" : "string",
    "DetectionId" : "string",
    "OccurrenceTime": "2018-06-14T21:38:11.543Z",
    "ActionResults": {
      "addtionalProp1": {
        "HandlingResponderVersion": 0,
        "HandlingResponderName": "string",
        "Results": [
          {
            "Status": {
              "Message": "string",
              "Code": {
                "Ordinal": 0,
                "Reason": "string",
                "Name": "string"
              }
            }
          }
        ]
      },
      "additonalProp2": {
        "HandlingResponderVersion": 0,
        "HandlingResponderName": "string",
        "Results": [
          {
            "Status": {
              "Message": "string",
              "Code": {
                "Ordinal": 0,
                "Reason": "string",
                "Name": "string"
              }
            }
          }
        ]
      },
      "additonalProp3": {
        "HandlingResponderVersion": 0,
        "HandlingResponderName": "string",
        "Results": [
          {
            "Status": {
              "Message": "string",
              "Code": {
                "Ordinal": 0,
                "Reason": "string",
                "Name": "string"
              }
            }
          }
        ]
      }
    },
    "AssociatedArtifacts": [
      {}
    ],
    "ResponseRuleId": "string",
    "SchemaVersion": 0,
    "ResponseRuleVersion": 0,
    "ReceivedTime": "2018-06-14T21:38:11.543Z",
    "ObjectType": "string"
```

```
    }
  ]
}
```

400 BadRequest – Returned for the following reasons:

- The Tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The unique identifier for the detection is not valid.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The detection resource requested does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
ActivationTime	The time at which this particular detection first started occurring.
AppliedExceptions	The exceptions that were applied to the detection.
ArtifactsOfInterest	The artifact associated with the rule that triggered the exception. This is a dynamic object.
AssociatedArtifacts	List of artifacts that were involved in this detection. These are dynamic objects.
Comment	The comment on the detection.
Context	The context of the detection.
DetectionRule	The description of the rule from which this detection originated. • Category: The category of the rule. • Description: The description of the rule. • Id: The ID of the rule. • Name: The name of the rule. • Version: The version of the rule.
Detector	The description of the plugin that originated the detection. • Name: The name of the detector. • Version: The version of the detector.
Device	A capture of the current state of the device.

	• CylanceId: The ID for the device. • Name: The name of the device.
Id	The detection’s unique identifier.
InvolvedArtifacts	The artifacts involved in this detection.
Name	The name of the detection.
ObjectType	The object type for the detection.
OccurrenceTime	The time at which the detection occurred.
PhoneticId	The easy-to-read version of the ID that is probabilistically unique.
Product	The description of the Cylance product that originated the detection. • Name: The name of the product. • Version: The version of the product.
ReceivedTime	The time when the detection was received.
Responses	The reponses to the detection. • Status: The status of the response. • Comment: The comment on the response. • TenantId: The tenant ID the response belongs to. • PhoneticId: The easy-to-read version of the ID that is probabilistically unique. • DetectionId: The identifier for the detection event that warranted the response. • OccurrenceTime: The time at which the response actions were taken. • ActionResults: ○ HandlingResponderVersion: The version of the Responder plugin that performed this response. ○ HandlingResponderName: The name of the Responder plugin that performed this response. ○ Results: ▪ Status: The status of the result. • Message: The message of the result. • Code: ○ Ordinal: The indicator code for the success of the action. ○ Reason: The detailed description explaining the indicator code. ○ Name: The friendly name of the status code. • AssociatedArtifacts: The artifacts upon which the action occurred. • ResponseRuleId: The ID of the response rule that triggered this response. • SchemaVersion: The version of the schema to which this object conforms. • ResponseRuleVersion: The version of the response rule. • ReceivedTime: The time the response was received.

	• ObjectType: The type of the object for the response.
SchemaVersion	The version of the schema to which this object conforms.
Severity	The criticality of an observance of this detection.
SeveritySortLevel	The sort level for this severity.
Status	The status of the detection in the workflow.
StatusSortLevel	The sort level for the status.
Trace	The trace information. • Event: The CylanceOPTICS Event that triggered the state. • StateName: The name of a state that was traversed.
TenantId	The ID for the tenant.
ZoneIds	A list of IDs for the zones associated with the detection.

Get Detection Recent

Allows a caller to request a count of recent CylanceOPTICS detection resources belonging to a tenant.

Service Endpoint	North America: https://protectapi.cylance.com/detections/v2/recent US Government: https://protectapi.us.cylance.com/detections/v2/recent All Other Regions: https://protectapi-{region-code}.cylance.com/detections/v2/recent
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsdetect:list scope encoded.
Request	Append the following optional query string parameter: • since: The date-time of the recent detections to retrieve.

Response	200 OK
	Get Detection Recent Response Schema
	{
	"num_after": 0,
	"num_unaddressed": 0
	}
	400 BadRequest – Returned for the following reasons:
	• The tenant ID could not be retrieved from the JWT token specified in the Authorization header.

	• The unique identifier for the execution is not valid. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 NotFound – The execution resource requested does not exist. 500 InternalServerError – An unforeseeable error has occurred. 503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
num_after	The number of detections after the "since" date-time.
num_unaddressed	The number of unaddressed detections after the "since" date-time.

Get Detections CSV

Allows a caller to request a CSV export of the list of CylanceOPTICS detection resources belonging to a tenant.

Service Endpoint	North America: https://protectapi.cylance.com/detections/v2/csv US Government: https://protectapi.us.cylance.com/detections/v2/csv All Other Regions: https://protectapi-{region-code}.cylance.com/detections/v2/csv
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsdetect:list scope encoded.
Request	Append the following optional query string parameters: • start: The start date-time of the query range. • end: The end date-time of the query range. • severity: The detection severity filter. Supports OR filters via multiple queries. Possible values are: o Informational o Low o Medium o High • detection_type: The detection type filter. Supports OR filters via multiple queries. • detected_on: The detection on filter. Supports OR filters via multiple queries. • event_number: The event number filter. Supports OR filters via multiple queries. • device: The device name filter. Supports OR filters via multiple queries. • status: Possible values are: o New o In Progress

	○ Follow Up ○ Reviewed ○ Done ○ False Positive • page: The page number to request. Defaults to 1. • page_size: The number of detection records to retrieve per page. Defaults to 20. • sort: Sort by the following fields (adding "-" in front of a value denotes descending order): ○ Severity ○ OccurrenceTime ○ Status ○ Device ○ PhoneticId ○ Description
Response	204 No Content – Detection deleted 400 BadRequest – Returned for the following reasons: • The Tenant ID cannot be retrieved from the JWT token. • The unique identifier for the detection is not valid. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 NotFound – Returned if the detection resource to update does not exist. 500 InternalServerError – An unforeseeable error has occurred. 503 Service Unavailable – Unable to respond at this time, please retry later.

Export detections for a tenant in CSV format. Any provided filters will be applied, but limit/offset parameters will not. All detections for the tenant will be exported.

Delete Detection

Allows a caller to soft delete a specific CylanceOPTICS detection resource belonging to a tenant. Use Get Detections to obtain the unique detection ID.

Service Endpoint	North America: https://protectapi.cylance.com/detections/v2/{unique_detection_id}
	US Government: https://protectapi.us.cylance.com/detections/v2/{unique_detection_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/detections/v2/{unique_detection_id}

Method	HTTP/1.1 DELETE
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsdetect:delete scope encoded.
Request	None

Response	200 OK – deleted
	Delete Detection Response Schema
	{
	  "id": "string",
	  "success": true
	}
	400 BadRequest – Returned for the following reasons:
	• The Tenant ID cannot be retrieved from the JWT token.
	• The detection's unique identifier is not valid.
	401 Unauthorized – The JWT token was not specified, has expired, or is otherwise invalid.
	403 Forbidden:
	• The JWT token did not contain the proper scope to perform this action.
	• Cannot delete CylanceOPTICS detecion.
	404 NotFound – Returned if the detecion resource to update doesn't exist.
	500 InternalServerError – An unforeseeable error has occurred.
	503 Service Unavailable – Unable to respond at this time, please retry later

Delete Detections

Allows a caller to delete CylanceOPTICS detection resources for a specific tenant.

Service Endpoint	North America: https://protectapi.cylance.com/detections/v2/ US Government: https://protectapi.us.cylance.com/detections/v2/ All Other Regions: https://protectapi-{region-code}.cylance.com/detections/v2/
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsdetect:delete scope encoded.
Request	None

Response	200 OK
	Delete Detections Response Schema
	[
	{
	"id": "string",
	"success": true
	}
	]
	400 BadRequest – Malformed request.
	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.
	403 Forbidden – The JWT token did not contain the proper scope to perform this action.
	500 InternalServerError – An unforeseeable error has occurred.

CylanceOPTICS Package Deployment

CylanceOPTICS users can now interact with a hardened Python interpreter that is present locally on each endpoint that is running CylanceOPTICS v2.3.1000 or higher. This new feature allows users to interact with their endpoints in an efficient and technical manner to accomplish tasks on endpoints in an automated fashion. By default, Cylance is supporting 5 capabilities to collect different forensic artifacts from targeted endpoints. These capabilities include:

- Collecting Master File Table (MFT) artifacts from NTFS volumes.
- Collecting entire Windows Registry Hives from endpoints.
- Collecting entire Windows Event Log files from endpoints.
- Collecting Web Browser History Databases from Chrome, Firefox, Internet Explorer, Edge, Opera, and Safari.
- Collecting common Application Execution Records, including Amcache, Prefetch, and Shimcache.

Users can also configure and deploy Custom Packages to conduct custom, scripted actions against endpoints. This allows customers to upload in-house or third-party scripts and applications to Cylance's cloud services and deploy them to endpoints. This scripting is done via interacting with the local Python interpreter built into CylanceOPTICS, allowing for an easily extensible set of capabilities.

After packages have been deployed and executed on endpoints, users can automatically upload the resulting data to SMB shares or SFTP servers for centralized collection and analysis by other forensic or incident response tools. Users can also configure packages to store the results locally on the endpoints for retrieval at a later time.

The CylanceOPTICS Package Deployment supports up to 20 packages for your organization. Each package has a maximum file size of 15MB.

These capabilities and workflows around the Package Deployment feature are exposed via Cylance's API.

Note: This addendum only covers CylanceOPTICS Package Deployment API information. Read the Cylance User API guide for configuration information.

Get Packages

Allows a caller to request a page with a list of packages belonging to a tenant, sorted by the uploaded date, in descending order (most recent uploaded package listed first). The page number and page size parameters are optional. When the values are not specified, the default values are 1 and 20 respectively.

Service Endpoint	North America: https://protectapi.cylance.com/packages/v2?page=m&page_size=n US Government: https://protectapi.us.cylance.com/packages/v2?page=m&page_size=n All Other Regions: https://protectapi-{region-code}.cylance.com/packages/v2?page=m&page_size=n
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticspkgconfig:list scope encoded.
Request	Append the following optional query string parameters: • page: The page number to request. Defaults to 1. • page_size: The number of packages records to retrieve per page. Defaults to 20. • sort: Sort by the following fields (adding "-" in front of value denotes descending order).

  - packageId
  - uploadedOn
  - uploadedBy.id
  - uploadedBy.login
  - size
  - status - Possible values are "started", "success, "failed", "timeout"
  - timeout
  - packageDescriptor.name
- packageId: Filter by packageId.
- uploadedOn: Filter by uploaded timestamp in UTC.
- uploadedBy.Id - Filter by user id the user who uploaded the package.
- updatedBy.Login - Filter by email fo the user who uploaded the package.
- size - Filter by size of the package in bytes.
- status - Filter by status of the package upload process. Possible values are "started", "success, "failed", "timeout".
- timeout - Filter by the amount of time in seconds for package to upload before status changes to "timeout".
- packageDescriptor.name
- category - Filter by package's category. Possible values are "custom", "cylance".

Response

200 OK

Get Packages Response Schema

```
{
  "page_number": 0,
  "page_size": 0,
  "total_pages": 0,
  "total_number_of_items": 0,
  "page_items": [
   {
     "packageId": "string",
     "uploadedOn": "2018-04-30T23:25:22.788Z",
     "uploadedBy": {
      "id": "string",
      "login": "string"
     },
     "size": 0,
     "status": "started",
     "timeout": 0,
     "downloadUrl": "string",
     "packageDescriptor": {
      "name": "string",
      "description": "string",
      "examples": [
       {
        "invocationString": "string",
          "description": "string"
        }
      ],
      "packageInfo": {
        "fileType": "python",
        "fileName": "string",
        "entryPoint": "main for Python packages."
      },
      "version": 0,
      "packageId": "string"
    },
    "category": "custom"
  }
 ]
}
```

400 BadRequest – Returned for the following reasons:

- The tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The page number or page size specified are less than or equal to zero.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The package resources page requested does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Server Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
page_items	The list of packages belonging to the requested page, each displaying the following information: • category: The category of the package. Values are **custom** or **cylance**. • downloadUrl: The URL to download the package from. • packageDescriptor: The package metadata, provided by the user. ○ description: The description of the package. ○ examples: A list of examples of how to use the package. ▪ invocationString: An example of how to invoke the package. ▪ description: A description of what the example does. ○ name: The name of the package. ○ packageId: The unique identifier for the package. ○ packageInfo: Package level documentation / annotation.

	▪ fileType: The file type of the package. Only Python is supported. ▪ fileName: The name of the package file. ▪ entryPoint: The point of execution for the package. ○ version: The version of the package. • packageId: The unique identifier for the package. • size: The size of the package, in bytes. • status: The status of the package in the upload process. • timeout: The amount of time (seconds) for a package upload before the status changes to timeout. • uploadedBy: The unique identifiers of the user who uploaded the package. ○ id: The unique ID for the user. ○ login: The email address of the user. • uploadedOn: The date and time (UTC) when the package was uploaded.
page_number	The page number requested.
page_size	The page size requested.
total_number_of_items	The total number of resources.
total_pages	The total number of pages that can be retrieved based on the page size specified.

Create Package

Allows a caller to create a new package resource for a tenant.

Service Endpoint	North America: https://protectapi.cylance.com/packages/v2 US Government: https://protectapi.us.cylance.com/packages/v2 All Other Regions: https://protectapi-{region-code}.cylance.com/packages/v2
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticspkgconfig:create scope encoded.
Request	Post Package Request Schema

```
{
  "checksum": "string",
  "packageDescriptor": {
    "name": "string",
    "description": "string",
    "examples": [
      {
        "invocationString": "string",
        "description": "string"
      }
    ],
    "packageInfo": {
      "fileType": "python",
      "fileName": "string",
      "entryPoint": "main"
    },
    "version": 0
  }
}
```

Response

202 Accepted – Successful request

Post Package Response Schema

```
{
  "packageId": "string",
  "uploadTo": "string",
  "packageUrl": "string"
}
```

400 BadRequest – Returned for the following reasons:

- The uploaded Package exceeds the 15MB size limit per package.
- The maximum number of 20 packages exceeded.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

500 InternalServerError – An unforeseeable error has occurred.

The request JSON schema contains the following fields:

Field Name	Description
checksum	The SHA-256 hash of the package.
packageDescriptor	Package metadata provided by the user. • description: The description of the package. • examples: A list of examples of how to use the package. ○ description: A description of what the example does. ○ invocationString: An example of how to invoke the package. • name: The name of the package.

	• packageInfo: Package level documentation / annotation. ○ entryPoint: The point of execution for the package. ○ fileType: The file type of the package. Note: Only Python is supported. ○ fileName: The name of the package file. • version: The version of the package.

The response JSON schema contains the following fields:

Field Name	Description
packageId	The ID of the packaged uploaded.
packageUrl	The URL to retrieve the package in the future, after the actual package is uploaded.
uploadTo	The URL to which the package is uploaded.

Get Package

Allows a caller to request a specific package resource belonging to a tenant. Use Get Packages to obtain the unique package ID.

Service Endpoint	North America: https://protectapi.cylance.com/packages/v2/{unique_package_id}
	US Government: https://protectapi.us.cylance.com/packages/v2/{unique_package_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/packages/v2/{unique_package_id}

Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticspkgconfig:read scope encoded.
Request	None
Response	200 OK Get Package Response Schema

```
{
  "packageId": "string",
  "uploadedOn": "2018-05-01T21:27:59.029Z",
  "uploadedBy": {
    "id": "string",
    "login": "string"
  },
  "size": 0,
  "status": "started",
  "timeout": 0,
  "downloadUrl": "string",
  "packageDescriptor": {
    "name": "string",
    "description": "string",
    "examples": [
     {
      "invocationString": "string",
      "description": "string"
     }
    ],
    "packageInfo": {
     "fileType": "python",
     "fileName": "string",
     "entryPoint": "string"
    },
    "version": 0,
    "packageId": "string"
  },
  "category": "custom"
}
```

400 BadRequest – Returned for the following reasons:

- The Tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The unique identifier for the package is not valid.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The package resource requested does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
category	The category of the package. Values are **custom** or **cylance**.
downloadUrl	The URL to download the package from.
packageDescriptor	The package metadata, provided by the user. • description: The description of the package. • examples: A list of examples of how to use the package. ○ description: A description of what the example does.

	○ invocationString: An example of how to invoke the package. • name: The name of the package. • packageInfo: Package level documentation / annotation. ○ entryPoint: The point of execution for the package. ○ fileName: The name of the package file. ○ fileType: The file type of the package. Only Python is supported. • packageId: The unique identifier for the package. • version: The version of the package.
packageId	The unique identifier for the package.
size	The size of the package, in bytes.
status	The status of the package in the upload process. Statuses are **started**, **success**, **failed**, and **timeout**.
timeout	The amount of time (seconds) for a package upload before the status changes to timeout.
uploadedBy	The unique identifiers of the user who uploaded the package. • id: The unique ID for the user. • login: The email address of the user.
uploadedOn	The date and time (UTC) when the package was uploaded.

Delete Package

Allows a caller to delete a specific package resource belonging to a tenant. Use Get Packages to obtain the unique package ID.

Service Endpoint	North America: https://protectapi.cylance.com/packages/v2/{unique_package_id}
	US Government: https://protectapi.us.cylance.com/packages/v2/{unique_package_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/packages/v2/{unique_package_id}

Method	HTTP/1.1 DELETE
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticspkgconfig:delete scope encoded.
Request	None
Response	204 No Content – Package deleted 400 BadRequest – Returned for the following reasons: • The Tenant ID cannot be retrieved from the JWT token. • The unique identifier for the package is not valid. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden

	• The JWT token did not contain the proper scope to perform this action. • Cannot delete the CylanceOPTICS package. 404 NotFound – Returned if the package resource to update does not exist. 500 InternalServerError – An unforeseeable error has occurred. 503 Service Unavailable – Unable to respond at this time, please retry later.

Create Package Execution

Allows a caller to create (add) a new CylanceOPTICS package execution resource for a specific tenant, which triggers a package to execute on the device or on devices in a specific zone.

Service Endpoint	North America: https://protectapi.cylance.com/packages/v2/executions US Government: https://protectapi.us.cylance.com/packages/v2/executions All Other Regions: https://protectapi-{region-code}.cylance.com/packages/v2/executions
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticspkgdeploy:create scope encoded.
Request	Post Package Execution Request Schema

```
{
  "execution": {
    "name": "string",
    "target": {
      "devices": [
        "string"
      ],
      "zones": [
        "string"
      ]
    },
    "destination": "string",
    "packageExecutions": [
      {
        "arguments": [
          "string"
        ],
        "package": "string"
      }
    ],
    "keepResultsLocally": false
  }
}
```

Response

202 Accepted – Package is being executed

Post Package Execution Request Schema

```
{
  "name": "string",
  "target": {
    "devices": [
      "string"
    ],
    "zones": [
      "string"
    ]
  },
  "destination": "string",
  "packageExecutions": [
    {
      "arguments": [
        "string"
      ],
      "package": "string"
    }
  ],
  "keepResultsLocally": false,
  "id": "string",
  "createdAt": "2018-05-02T00:33:12.092Z",
  "createdBy": {
    "id": "string",
    "login": "string"
  },
  "deviceStatuses": {
    "acked": 0,
    "succeeded": 0,
    "failed": 0
  },
  "deviceCount": 0
}
```

400 BadRequest – Malformed request.

	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 500 InternalServerError – An unforeseeable error has occurred.

The request JSON schema contains the following fields:

Field Name	Description
execution	The execution object to be created. • destination: The FTP, SFTP, or SAMBA URL for saving the results. • keepResultsLocally: The setting to save the results to the local disk drive. If true, the results are saved to file://[Cylance Data Directory]/Optics. • name: The name of the execution. • packageExecutions: The list of packages to execute. o arguments: The list of arguments for the package. See examples from packageDescriptor. o package: The URL to download the package resource from. • target: The devices and/or zones to execute the packages against. o devices: The list of device IDs to execute the packages against. o zones: The list of zone IDs to execute the packages against.

The response JSON schema contains the following fields:

Field Name	Description
createdAt	The date and time (UTC) when the execution was requested.
createdBy	The user who requested the execution. • id: The unique ID for the user. • login: The email address of the user.
destination	The FTP, SFTP, or SAMBA URL for saving the results.
deviceCount	The number of online devices at the moment the package execution request was made.
deviceStatuses	The statuses of the package executions on the devices. • acked (acknowledged): The number of devices that received the package execution command but have not yet responded. • failed: The number of devices that failed to execute the packages. • succeeded: The number of devices that have successfully executed the packages.
id	The ID of the execution resource.

keepResultsLocally	The setting to save the results to the local disk drive. If true, the results are saved to file://[Cylance Data Directory]/Optics.
name	The name of the execution.
packageExecutions	The list of packages to execute. • arguments: The list of arguments for the package. See examples from packageDescriptor. • package: The URL to download the package resource from.
target	The devices and/or zones to execute the packages against. • devices: The list of device IDs to execute the packages against. • zones: The list of zone IDs to execute the packages against.

Get Package Executions

Allows a caller to request a page with a list of package executions belonging to a tenant, sorted by the uploaded date, in descending order (most recent uploaded package execution listed first). The page number and page size parameters are optional. When the values are not specified, the default values are 1 and 20 respectively.

Service Endpoint	North America: https://protectapi.cylance.com/packages/v2/executions?page=m&page_size=n US Government: https://protectapi.us.cylance.com/packages/v2/executions?page=m&page_size=n All Other Regions: https://protectapi-{region-code}.cylance.com/packages/v2/executions?page=m&page_size=n
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticspkgdeploy:list scope encoded.
Request	Append the following optional query string parameters: • page: The page number to request. Defaults to 1. • page_size: The number of packages records to retrieve per page. Defaults to 20. • sort: Sort by the following fields (adding "-" in front of value denotes descending order). o id o name o createdAt o createdBy.id o createdBy.login o deviceCount • id: Filter by ID of execution. • name: Filter by name of execution. • createAt: Filter by date and time of when the execution was requested in UTC. • createdBy.Id: Filter by ID of the user who requested the execution. • createdBy.Login: Filter by the email address of the user who requested the execution.

- deviceCount: Filter by the number of online devices at the moment the package execution request was made.

Response

200 OK

Get Package Executions Response Schema

```
{
  "page_number": 0,
  "page_size": 0,
  "total_pages": 0,
  "total_number_of_items": 0,
  "page_items": [
    {
      "name": "string",
      "target": {
        "devices": [
          "string"
        ],
        "zones": [
          "string"
        ]
      },
      "destination": "string",
      "packageExecutions": [
        {
          "arguments": [
            "string"
          ],
          "package": "string"
        }
      ],
      "keepResultsLocally": false,
      "id": "string",
      "createdAt": "2018-05-02T23:18:08.719Z",
      "createdBy": {
        "id": "string",
        "login": "string"
      },
      "deviceStatuses": {
        "acked": 0,
        "succeeded": 0,
        "failed": 0
      },
      "deviceCount": 0
    }
  ]
}
```

400 BadRequest – Returned for the following reasons:

- The tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The page number or page size specified are less than or equal to zero.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The package resources page requested does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
page_items	The list of executions belonging to the requested page, each displaying the following information: • createdAt: The date and time (UTC) when the execution was requested. • createdBy: The user who requested the execution. ○ id: The ID of the user who requested the execution. ○ login: The email address of the user who requested the execution. • destination: The FTP, SFTP, or SAMBA URL for saving the results. • deviceCount: The number of online devices at the moment the package execution request was made. • deviceStatuses: The statuses of the package executions on the devices. ○ acked (acknowledged): The number of devices that received the package execution command but have not yet responded. ○ failed: The number of devices that failed to execute the packages. ○ succeeded: The number of devices that have successfully executed the packages. • id: The ID of the execution resource. • keepResultsLocally: The setting to save the results to the local disk drive. If true, the results are saved to file://[Cylance Data Directory]/Optics. • name: The name of the execution. • packageExecutions: The list of packages to execute. ○ arguments: The list of arguments for the package. See examples from packageDescriptor. ○ package: The URL to download the package resource from.

	- target: The devices and/or zones to execute the packages against. ○ devices: The list of device IDs to execute the packages against. ○ zones: The list of zone IDs to execute the packages against.
page_number	The page number requested.
page_size	The page size requested.
total_number_of_items	The total number of resources.
total_pages	The total number of pages that can be retrieved based on the page size specified.

Get Package Execution

Allows a caller to request a specific package execution resource belonging to a tenant. Use Get Package Executions to obtain the unique package execution ID.

Service Endpoint	North America: https://protectapi.cylance.com/packages/v2/executions/{unique_exeuction_id}
	US Government: https://protectapi.us.cylance.com/packages/v2/executions/{unique_exeuction_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/packages/v2/executions/{unique_exeuction_id}

Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticspkgdeploy:read scope encoded.
Request	None
Response	200 OK Get Package Execution Response Schema

```
{
 "name": "string",
 "target": {
  "devices": [
   "string"
  ],
  "zones": [
   "string"
  ]
 },
 "destination": "string",
 "packageExecutions": [
  {
   "arguments": [
    "string"
```

```
      ],
      "package": "string"
    }
  ],
  "keepResultsLocally": false,
  "id": "string",
  "createdAt": "2018-05-02T23:18:08.719Z",
  "createdBy": {
    "id": "string",
    "login": "string"
  },
  "deviceStatuses": {
    "acked": 0,
    "succeeded": 0,
    "failed": 0
  },
  "deviceCount": 0
}
```

400 BadRequest – Returned for the following reasons:

- The tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The unique identifier for the execution is not valid.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The execution resource requested does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
createdAt	The date and time (UTC) when the execution was requested.
createdBy	The user who requested the execution. • id: The unique ID for the user. • login: The email address of the user.
destination	The FTP, SFTP, or SAMBA URL for saving the results.
deviceCount	The number of online devices at the moment the package execution request was made.

deviceStatuses	The statuses of the package executions on the devices. • acked (acknowledged): The number of devices that received the package execution command but have not yet responded. • failed: The number of devices that failed to execute the packages. • succeeded: The number of devices that have successfully executed the packages.
id	The ID of the execution resource.
keepResultsLocally	The setting to save the results to the local disk drive. If true, the results are saved to file://[Cylance Data Directory]/Optics.
name	The name of the execution.
packageExecutions	The list of packages to execute. • arguments: The list of arguments for the package. See examples from packageDescriptor. • package: The URL to download the package resource from.
target	The devices and/or zones to execute the packages against. • devices: The list of device IDs to execute the packages against. • zones: The list of zone IDs to execute the packages against.

Delete Package Execution

Allows a caller to delete a specific package resource belonging to a tenant. Use Get Package Executions to obtain the unique package ID.

Service Endpoint	North America: https://protectapi.cylance.com/packages/v2/execution/{unique_exeuction_id}
	US Government: https://protectapi.us.cylance.com/packages/v2/execution/{unique_exeuction_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/packages/v2/execution/{unique_exeuction_id}

Method	HTTP/1.1 DELETE
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticspkgdeploy:delete scope encoded.
Request	None

Response	204 No Content – Package deleted
	Delete Package Execution Response Schema
	{
	"name": "string",
	"target": {
	"devices": [
	"string"

```
  ],
  "zones": [
   "string"
  ]
 },
 "destination": "string",
 "packageExecutions": [
  {
   "arguments": [
    "string"
   ],
   "package": "string"
  }
 ],
 "keepResultsLocally": false,
 "id": "string",
 "createdAt": "2018-05-02T23:18:08.719Z",
 "createdBy": {
  "id": "string",
  "login": "string"
 },
 "deviceStatuses": {
  "acked": 0,
  "succeeded": 0,
  "failed": 0
 },
 "deviceCount": 0
}
```

400 BadRequest – Returned for the following reasons:

- The Tenant ID cannot be retrieved from the JWT token.
- The unique identifier for the package is not valid.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden

- The JWT token did not contain the proper scope to perform this action.
- Cannot delete the CylanceOPTICS package.

404 NotFound – Returned if the package execution resource to update does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
createdAt	The date and time (UTC) when the execution was requested.
createdBy	The user who requested the execution. • id: The unique ID for the user. • login: The email address of the user.
destination	The FTP, SFTP, or SAMBA URL for saving the results.
deviceCount	The number of online devices at the moment the package execution request was made.
deviceStatuses	The statuses of the package executions on the devices. • acked (acknowledged): The number of devices that received the package execution command but have not yet responded. • failed: The number of devices that failed to execute the packages. • succeeded: The number of devices that have successfully executed the packages.
id	The ID of the execution resource.
keepResultsLocally	The setting to save the results to the local disk drive. If true, the results are saved to file://[Cylance Data Directory]/Optics.
name	The name of the execution.
packageExecutions	The list of packages to execute. • arguments: The list of arguments for the package. See examples from packageDescriptor. • package: The URL to download the package resource from.
target	The devices and/or zones to execute the packages against. • devices: The list of device IDs to execute the packages against. • zones: The list of zone IDs to execute the packages against.

CylanceOPTICS Detection Rules

The CylanceOPTICS Detection Rules API allows users to create or update rules to help monitor an organization for security threats or anomalous behavior. The flexibility of Detection Rules allows users to monitor for broad behavior characteristics (for example, files being created with certain naming patterns) or search for a targeted series of events (for example, a process with a certain file signature thumbprint that then creates files and initiates network connections).

The CylanceOPTICS Detection Rules API includes:

- Getting the content of a Detection Rule
- Getting a list of Detection Rules for a tenant
- Getting a list of Detection Rules as a CSV file
- Validating a Detection Rule
- Creating a Detection Rule
- Updating a Detection Rule
- Deactivating (or soft deleting) a Detection Rule
- Getting a natural language representation of a Detection Rule
- Getting a count of how many Detection Rules exist in a tenant

Note: This addendum only covers CylanceOPTICS Detection Rules API information. Read the Cylance User API guide for configuration information.

Get Detection Rule Content

Allows a caller to retrieve the content of a Detection Rule in its native JSON structure. The structure of CylanceOPTICS Detection Rules is beyond the scope of this document. Refer to the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article for a more detailed explanation. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rules/v2/{rule_id}
	US Government: https://protectapi.us.cylance.com/rules/v2/{rule_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/rules/v2/{rule_id}

Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsrule:read scope encoded.
Request	None
Response	200 OK Get Detection Rule Content Response Schema { "Id": "631015e5...", "Name": "My Detection Rule",

```
"Description": "My Detection Rule Description",
"ObjectType": "DetectionRule",
"OperatingSystems": [
 {
  "Name": "Windows"
 }
],
"Plugin": {
 "Name": "OpticsDetector"
},
"Product": {
 "Name": "CylanceOPTICS"
},
"SchemaVersion": 1,
"States": [
 {
  "Name": "MaliciousApp",
  "Scope": "Global",
  "Function": "Function",
  "FieldOperators": {
   "Function": {
    "Type": "EqualsAny",
    "Operands": [
     {
      "Source": "LiteralSet",
      "Data": "badapp.exe"
     }
    ],
    "OperandType": "string",
    "Options": {
     "IgnoreCase": true
    }
   }
  },
  "Actions": [
   {
    "Type": "AOI",
    "ItemName": "InstigatingProcess",
    "Postiion": "PostActivation"
   }
  ]
 }
],
"Tags": [
 "CylanceOPTICS"
```

	], "Version": 1 } 400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such detection rule found. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

The response JSON schema contains the entirety of the Detection Rule logic. This schema is further explained in the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article.

Get Detection Rule List

Allows a caller to retrieve a list of Detection rules available in a tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rules/v2?page=m&page_size=n US Government: https://protectapi.us.cylance.com/rules/v2?page=m&page_size=n All Other Regions: https://protectapi-{region-code}.cylance.com/rules/v2?page=m&page_size=n
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsrule:list scope encoded.
Request	None
Response	200 OK Get Detection Rule List Response Schema { “page_size”: 10, “total_pages”: 1, “page_items”: ["Id": "631015e5…", "Version": 1,

```
    "Name": "My Detection Rule",
    "Description": "My Detection Rule Description",
    "LastModified": "2018-11-26T23:36:22.954Z",
    "ModifiedBy": {
      "id": "efad5148...",
      "login": "user@test.com"
    },
    "Severity": "Medium",
    "DeviceCount": 0,
    "RulesetCount": 0,
    "Category": "Cylance Rules",
    "OperatingSystems": [
      {
        "Name": "Windows"
      }
    ]
  ],
  “total_number_of_items”: 10,
  “page_number”: 1
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
page_size	The number of items on the page.
total_pages	The total number of pages of this size.
total_number_of_items	The total number of Detection Rules in the tenant.
page_number	The current page number of results.
page_items	A list of Exception objects that are available in the tenant that will contain the following fields.
Id	The unique ID of the Detection Rule.

Version	The version of the Detection Rule.
Name	The name of the Detection Rule.
Description	The description of the Detection Rule.
LastModified	The timestamp (in UTC) of the last time that the Detection Rule was modified.
ModifiedBy	An object detailing the last user to modify the Detection Rule. It includes the following fields: • id: The unique ID of the user who modified the Detection Rule. • login: The email address of the user who modified the Detection Rule.
Severity	The severity assigned to the Detection Rule. Possible values are: • High • Medium • Low • Informational
DeviceCount	The number of devices that have the Detection Rule applied.
RulesetCount	The number of Detection Rule Sets that have the Detection Rule enabled.
Category	The category or rule grouping that the Detection Rule belongs to. Possible values include: • Custom: Custom rules that users have uploaded to a tenant. • Cylance Rules: Cylance-official rules. • Cylance Experimental: Cylance rules that are deemed to be experimental in their nature.
OperatingSystems	An object detailing the operating systems that the Detection Rule can be applied to. It will include the “name” field. This can consist of: • “Windows” • “MacOS”

Get Detection Rule CSV List

Allows a caller to retrieve a CSV where every line represents a Detection Rule available in the tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rules/v2/csv US Government: https://protectapi.us.cylance.com/rules/v2/csv All Other Regions: https://protectapi-{region-code}.cylance.com/rules/v2/csv
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsrule:list scope encoded.

Request	None
Response	200 OK – Will initiate a CSV file download. 400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such resource found. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

The response CSV contains the following fields:

Field Name	Description
Name	The name of the Detection Rule.
Id	The unique ID of the Detection Rule.
Version	The version of the Detection Rule.
Description	The description of the Detection Rule.
Severity	The severity of the Detection Rule.
Category	The category that the Detection Rule belongs to.
Last Modified	The timestamp (in UTC) of the last time that the Detection Rule was modified.
Modified By	The email address of the user who last modified the Detection Rule.
Device Count	The number of devices that have the Detection Rule applied.
Ruleset Count	The number of Detection Rule Sets that have the Detection Rule enabled.

Validate Detection Rule

Allows a caller to validate a Detection Rule's JSON by sending the native JSON structure of a Detection Rule to a validation service. The structure of CylanceOPTICS Detection Rules is beyond the scope of this document. Refer to the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article for a more detailed explanation. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rules/v2/validate US Government: https://protectapi.us.cylance.com/rules/v2/validate All Other Regions: https://protectapi-{region-code}.cylance.com/rules/v2/validate
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsrule:read scope encoded.
Request	Post Detection Rule Validation Request Schema

```
{
  "Name": "My Detection Rule",
  "Description": "My Detection Rule Description",
  “Severity”: “Medium”,
  "ObjectType": "DetectionRule",
  "OperatingSystems": [
    {
      "Name": "Windows"
    }
  ],
  "Plugin": {
    "Name": "OpticsDetector"
  },
  "Product": {
    "Name": "CylanceOPTICS"
  },
  "SchemaVersion": 1,
  "States": [
    {
      "Name": "MaliciousApp",
      "Scope": "Global",
      "Function": "Function",
      "FieldOperators": {
        "Function": {
          "Type": "EqualsAny",
          "Operands": [
            {
              "Source": "LiteralSet",
              "Data": "badapp.exe"
            }
          ],
          "OperandType": "string",
          "Options": {
            "IgnoreCase": true
          }
        }
      },
      "Actions": [
        {
          "Type": "AOI",
          "ItemName": "InstigatingProcess",
          "Position": "PostActivation"
        }
      ],
      "Filters": [
        {
          "Type": "Event",
          "Data": {
            "Category": "Process",
            "SubCategory": "",
            "Type": "*"
          }
        }
      ]
    }
  ],
  "Tags": [
    "CylanceOPTICS"
  ]
}
```

Response

200 OK

Post Detection Rule Validation Response Schema

```
{
  "valid": true,
  "warnings": [
    "string"
  ],
  "errors": [
    "string"
  ]
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource found.

	500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
valid	Returns 'true' if the Detection Rule passes validation. Returns 'false' if the Detection Rule does not pass validation.
warnings	A list of warning message strings that may impact the performance or validity of the Detection Rule.
errors	A list of error message that will prevent the Detection Rule from validating and operating correctly.

Create Detection Rule

Allows a caller to create a new Detection Rule by sending the native JSON structure of a Detection Rule. The structure of CylanceOPTICS Detection Rules is beyond the scope of this document. Refer to the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article for a more detailed explanation. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rules/v2 US Government: https://protectapi.us.cylance.com/rules/v2 All Other Regions: https://protectapi-{region-code}.cylance.com/rules/v2
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsrule:create scope encoded.

Request	Post Detection Rule Content Request Schema
	{
	  "Name": "My Detection Rule",
	  "Description": "My Detection Rule Description",
	  “Severity”: “Medium”,
	  "ObjectType": "DetectionRule",
	  "OperatingSystems": [
	   {
	     "Name": "Windows"
	   }

```
],
"Plugin": {
 "Name": "OpticsDetector"
},
"Product": {
 "Name": "CylanceOPTICS"
},
"SchemaVersion": 1,
"States": [
 {
  "Name": "MaliciousApp",
  "Scope": "Global",
  "Function": "Function",
  "FieldOperators": {
   "Function": {
    "Type": "EqualsAny",
    "Operands": [
     {
      "Source": "LiteralSet",
      "Data": "badapp.exe"
     }
    ],
    "OperandType": "string",
    "Options": {
     "IgnoreCase": true
    }
   }
  },
  "Actions": [
   {
    "Type": "AOI",
    "ItemName": "InstigatingProcess",
    "Position": "PostActivation"
   }
  ],
  "Filters": [
   {
    "Type": "Event",
    "Data": {
     "Category": "Process",
     "SubCategory": "",
     "Type": "*"
    }
   }
  ]
```

```
    }
  ],
  "Tags": [
    "CylanceOPTICS"
  ]
}
```

Response

202 OK – Rule created successfully.

Post Detection Rule Content Response Schema

```
{
    "Name": "My Detection Rule",
    "Description": "My Detection Rule Description",
    "Severity": "Medium",
    "ObjectType": "DetectionRule",
    "OperatingSystems": [
        {
            "Name": "Windows"
        }
    ],
    "Plugin": {
        "Name": "OpticsDetector"
    },
    "Product": {
        "Name": "CylanceOPTICS"
    },
    "SchemaVersion": 1,
    "States": [
        {
            "Name": "MaliciousApp",
            "Scope": "Global",
            "Function": "Function",
            "FieldOperators": {
                "Function": {
                    "Type": "EqualsAny",
                    "Operands": [
                        {
                            "Source": "LiteralSet",
                            "Data": "badapp.exe"
                        }
                    ],
                    "OperandType": "string",
                    "Options": {
                        "IgnoreCase": true
                    }
                }
```

```
      },
      "Actions": [
        {
          "Type": "AOI",
          "ItemName": "InstigatingProcess",
          "Position": "PostActivation"
        }
      ],
      "Filters": [
        {
          "Type": "Event",
          "Data": {
            "Category": "Process",
            "SubCategory": "",
            "Type": "*"
          }
        }
      ]
    }
  ],
  "Tags": [
    "CylanceOPTICS"
  ],
  "RuleSourceGrouping": "Custom Rule",
  "Id": "b25285ba-8633-4a13-bf3c-21f0ddbd1569",
  "Version": 1
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the entirety of the Detection Rule logic. This schema is further explained in the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article.

Note that the ‘id’ and ‘version’ fields are automatically populated when the request is submitted.

Update Detection Rule

Allows a caller to update a Detection Rule by sending a new JSON structure. The structure of CylanceOPTICS Detection Rules is beyond the scope of this document. Refer to the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article for a more detailed explanation. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rules/v2/{rule_id}
	US Government: https://protectapi.us.cylance.com/rules/v2/{rule_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/rules/v2/{rule_id}

Method	HTTP/1.1 PUT
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsrule:update scope encoded.
Request	Update Detection Rule Content Request Schema

```
{
  "Name": "My Detection Rule",
  "Description": "My New Detection Rule Description",
  “Severity”: “Medium”,
  "ObjectType": "DetectionRule",
  "OperatingSystems": [
    {
      "Name": "Windows"
    }
  ],
  "Plugin": {
    "Name": "OpticsDetector"
  },
  "Product": {
    "Name": "CylanceOPTICS"
  },
  "SchemaVersion": 1,
  "States": [
    {
      "Name": "MaliciousApp",
      "Scope": "Global",
      "Function": "Function",
      "FieldOperators": {
        "Function": {
          "Type": "EqualsAny",
          "Operands": [
            {
              "Source": "LiteralSet",
              "Data": "badapp_12312.exe"
            }
          ],
          "OperandType": "string",
          "Options": {
            "IgnoreCase": true
          }
        }
      },
      "Actions": [
        {
          "Type": "AOI",
          "ItemName": "InstigatingProcess",
          "Position": "PostActivation"
        }
      ],
      "Filters": [
        {
          "Type": "Event",
          "Data": {
            "Category": "Process",
            "SubCategory": "",
            "Type": "*"
          }
        }
      ]
    }
  ],
  "Tags": [
    "CylanceOPTICS"
  ]
}
```

Response

202 OK – Rule created successfully.

Update Detection Rule Content Response Schema

```
{
  "Name": "My Detection Rule",
  "Description": "My Detection Rule Description",
  “Severity”: “Medium”,
  "ObjectType": "DetectionRule",
  "OperatingSystems": [
    {
      "Name": "Windows"
    }
  ],
  "Plugin": {
```

```
 "Name": "OpticsDetector"
},
"Product": {
 "Name": "CylanceOPTICS"
},
"SchemaVersion": 1,
"States": [
 {
  "Name": "MaliciousApp",
  "Scope": "Global",
  "Function": "Function",
  "FieldOperators": {
   "Function": {
    "Type": "EqualsAny",
    "Operands": [
     {
      "Source": "LiteralSet",
      "Data": " badapp_12312.exe"
     }
    ],
    "OperandType": "string",
    "Options": {
     "IgnoreCase": true
    }
   }
  },
  "Actions": [
   {
    "Type": "AOI",
    "ItemName": "InstigatingProcess",
         "Position": "PostActivation"
       }
     ],
     "Filters": [
       {
         "Type": "Event",
         "Data": {
           "Category": "Process",
           "SubCategory": "",
           "Type": "*"
         }
       }
     ]
   }
 ],
```

	"Tags": ["CylanceOPTICS"], "RuleSourceGrouping": "Custom Rule", "Id": "b25285ba-8633-4a13-bf3c-21f0ddbd1569", "Version": 2 } 400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such Detection Rule found. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

The response JSON schema contains the entirety of the Detection Rule logic. This schema is further explained in the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article.

Note that the 'id' and 'version' fields are automatically populated when the request is submitted.

Deactivate / Delete Detection Rule

Allows a caller to 'soft delete' a Detection Rule and remove it from Detection Rule Sets. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rules/v2/{rule_id}/deactivate US Government: https://protectapi.us.cylance.com/rules/v2/{rule_id}/deactivate All Other Regions: https://protectapi-{region-code}.cylance.com/rules/v2/{rule_id}/deactivate
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsrule:update scope encoded.
Request	None
Response	200 OK 400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

	403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such resource found. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

Note that Detection Rule Sets are not automatically communicated to all endpoints when updates to Detection Rules are made. To ensure that the latest logic is applied to endpoints in the quickest manner, re-save any affected Detection Rule Sets (either via the UI or API.)

Get Detection Rule Natural Language Representation

Allows a caller to retrieve the ‘natural language’ representation of a rule. This process converts the Detection Rule logic into a series of ‘AND’s, ‘OR’s, and ‘NOT’s to describe what the Detection Rule looks for. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rules/v2/{rule_id}/natlang US Government: https://protectapi.us.cylance.com/rules/v2/{rule_id}/natlang All Other Regions: https://protectapi-{region-code}.cylance.com/rules/v2/{rule_id}/natlang
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsrule:read scope encoded.
Request	None

Response	200 OK
	Update Detection Rule Natural Language Response Schema
	{
	"Name": "My Detection Rule",
	"Paths": [
	[
	{“Instigating Process Name is ‘badapp_12312.exe’.}
	]
	]
	}
	400 Bad Request – Malformed request.
	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.
	403 Forbidden – The JWT token did not contain the proper scope to perform this action.
	404 Not Found – No such resource found.
	500 InternalServerError – An unforeseeable error has occurred.
	503 Service unavailable – Please try again later.

Get Detection Rule Counts

Allows a caller to retrieve counts of how many devices, Detection Rule Sets, and policies that have a particular Detection Rule applied. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rules/v2/{rule_id}/counts US Government: https://protectapi.us.cylance.com/rules/v2/{rule_id}/counts All Other Regions: https://protectapi-{region-code}.cylance.com/rules/v2/{rule_id}/counts
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsrule:read scope encoded.
Request	None

Response	200 OK
	Get Detection Rule Count Response Schema
	{
	  "DeviceCount": 0,
	  "RulesetCount": 0,
	  "PolicyCount": 0
	}
	400 Bad Request – Malformed request.
	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.
	403 Forbidden – The JWT token did not contain the proper scope to perform this action.
	404 Not Found – No such Detection Rule found.
	500 InternalServerError – An unforeseeable error has occurred.
	503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
DeviceCount	The number of devices that have the requested Detection Rule applied.
RulesetCount	The number of Detection Rule Sets that have the requested Detection Rule enabled.
PolicyCount	The number of Device Policies that have the requested Detection Rule applied.

CylanceOPTICS Detection Rule Sets

The CylanceOPTICS Detection Rule Set API allows users to create a set of rules and apply that set to Device Policies.

The CylanceOPTICS Detection Rule Set API includes:

- Getting content for a Detection Rule Set
- Getting a list of Detection Rule Sets
- Creating a Detection Rule Set
- Retrieving a Default Detection Rule Set (retrieving a default template)
- Updating a Detection Rule Set
- Deleting a Detection Rule Set
- Deleting multiple Detection Rule Sets
- Getting a list of Detection Rule Sets as a CSV file

Note: This addendum only covers CylanceOPTICS Detection Rule Set API information. Read the Cylance User API guide for configuration information.

Get Detection Rule Set List

Allows a caller to retrieve a list of Detection Rule Sets available in a tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule Set resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rulesets/v2?page=m&page_size=n US Government: https://protectapi.us.cylance.com/ rulesets/v2?page=m&page_size=n All Other Regions: https://protectapi-{region-code}.cylance.com/ rulesets/v2?page=m&page_size=n
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsruleset:list scope encoded.
Request	Append the following parameters to the request to sort and filter the results: • description: Case-insensitive query parameter to filter or sort by the Description field. • last_modified: Case-insensitive query parameter to filter or sort by the Last Modified field. • modified_by.id: Case-insensitive query parameter to filter or sort by a user’s unique ID. • modified_by.login: Case-insensitive query parameter to filter or sort by a user’s email address. • device_count: Filter or sort the list by the number of applied devices. • sort: Sort by field (adding ‘-‘ in front of value denotes descending order.)
Response	200 OK Get Detection Rule Set List Response Schema { “page_size”: 10, “total_pages” 1, “page_items”: [

```
    "name": "My Detection Rule Set",
    "description": "My Detection Rule Set Description",
    "notification_message": "My Detection Rule Set Notification",
    "id": "ae53bc38…",
    "last_modified": "2018-11-26T23:36:22.810Z",
    "modified_by": {
      "id": "6572fe36…",
      "login": "user@test.com"
    },
    "policies": [
      "6521ffee…"
    ],
    "device_count": 0,
    "category": "Cylance"
  ],
  “total_number_of_items”: 10,
  “page_number”: 1
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
page_size	The number of items on the page.
total_pages	The total number of pages of this size.
total_number_of_items	The total number of Exceptions in the tenant.
page_number	The current page number of results.
page_items	A list of Exception objects that are available in the tenant that will contain the following fields.
name	The name of the Detection Rule Set
description	The description of the Detection Rule Set.
id	The unique ID of the Detection Rule Set.

last_modified	The timestamp (in UTC) of the last time that the Detection Rule Set was modified.
modified_by	An object detailing the last user to modify the Detection Rule Set. It includes the following fields: • id: The unique ID of the user who modified the Detection Rule Set. • login: The email address of the user who modified the Detection Rule Set.
policies	A list of policy IDs that a Detection Rule Set is applied to.
Device_count	The number of devices that have the Detection Rule Set applied.
category	The category or rule grouping that the Detection Rule Set belongs to. Possible values include: • Custom: Custom Detection Rule Sets that have been created by a user. • Cylance: Detection Rule Sets that have been created by Cylance.

Get Detection Rule Set Content

Allows a caller to retrieve the content of a Detection Rule Set including Detection Rules, Response Actions, Detection Exceptions, Package Playbooks, and the Polices where the Detection Rule Set is applied. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule Set resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rulesets/v2/{ruleset_id}
	US Government: https://protectapi.us.cylance.com/ rulesets/v2/{ruleset_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/rulesets/v2/{ruleset_id}

Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsruleset:read scope encoded.
Request	None

Response	200 OK
	Get Detection Rule Set Content Response Schema
	{
	  "name": "My Detection Rule Set",
	  "description": "My Detection Rule Set Description",
	  "notification_message": "My Detection Rule Set Notification",
	  "category": "Cylance",
	  "id": "decd5e3a…",
	  "last_modified": "2018-11-26T23:36:22.839Z",
	  "modified_by": {
	    "id": "54ea98ae…",

```
  "login": "user@test.com"
 },
 "rules": [
  {
   "detection_rule_id": "4f722b34... ",
   "detection_rule_version": 1,
   "detection_name": "Dropper/Downloader",
   "detection_description": "A process has created a new executable file and then executed that file",
   "category": "Cylance Rules",
   "severity": "Medium",
   "operating_systems": [
    {
     "Name": "Windows"
    }
   ],
   "date_added": "2018-11-26T23:36:22.839Z",
   "enabled": true,
   "notification_enabled": true,
   "responses": [
    {
     "template_id": "4f722b34...",
     "response_rule_id": "6eb12c34... ",
     "response_rule_version": 1,
     "description": "Delete file",
     "value": {},
     "enabled": true,
     "created": "2018-11-26T23:36:22.839Z"
    }
   ],
   "exceptions": [
    {
     "exception_id": "d2cf6cc7...",
     "enabled": true,
     "name": "My Detection Exception"
    }
   ],
   "playbooks": [
    "d5e4ffcc..."
   ]
  }
 ]
}
```

400 Bad Request – Malformed request.

	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such Detection Rule Set found. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
name	The name of the Detection Rule Set.
description	The description of the Detection Rule Set.
notification_message	The message to display on the endpoint when a Detection Rule is triggered.
id	The unique ID of the Detection Rule Set
last_modified	The timestamp (in UTC) of the last time that the Detection Rule Set was modified.
modified_by	An object detailing the last user to modify the Detection Rule Set. It includes the following fields: • id: The unique ID of the user who modified the Detection Rule Set. • login: The email address of the user who modified the Detection Rule Set.
rules	A list of Detection Rule objects and their associated Response Actions, Detection Exceptions, and Package Playbooks. Each object in the list will contain the following fields.
detection_rule_id	The unique ID of the Detection Rule.
detection_rule_version	The version of the Detection Rule.
detection_name	The name of the Detection Rule.
detection_description	The description of the Detection Rule Set.
category	The category of the Detection Rule.
severity	The severity assigned to the Detection Rule. Possible values are: • High • Medium • Low • Informational
operating_systems	An object detailing the operating systems that the Detection Rule can be applied to. It will include the "name" field. This can consist of: • "Windows"

	• “MacOS”
date_added	The timestamp (in UTC) when the Detection Rule was added to the tenant.
enabled	Determines whether or not a Detection Rule is enabled in the Detection Rule Set. When viewing the content of a Detection Rule Set, this should always be set to ‘true.’
notification_enabled	Determines whether or not the message defined in the ‘notification_message’ field should display on the device when the Detection Rule is triggered.
responses	A list of response objects for each Response Action enabled for a particular Detection Rule. Each object will include the following fields: • template_id: The ID of the response template to use (this is provided by Cylance.) • response_rule_id: The ID of the response rule to enable (these are provided by Cylance.) • response_rule_version: The version of the response rule to enable (this is provided by Cylance.) • description: The description / name of the response rule. • value: A currently unused field. • enabled: This will always be ‘true’ when viewing a Detection Rule Set • created: The date that the Response Rule was added to the tenant.
exceptions	A list of Exception Rule objects that should be applied to the Detection Rule. Each object will include the following fields: • exception_id: The unique ID of the Exception Rule. • enabled: This will always be ‘true’ when viewing a Detection Rule Set. • name: The name of the Exception Rule.
playbooks	A list of Package Playbook unique IDs that will be executed when the Detection Rule is triggered on the device.

Create Detection Rule Set

Allows a caller to create a new Detection Rule Set. Detection Rule Sets can require a large number of fields and unique IDs to function properly. It is recommended that callers make a GET request to ‘*/rulesets/v2/default*’ to obtain a properly formatted template prior to submitting a POST request described below. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule Set resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rulesets/v2 US Government: https://protectapi.us.cylance.com/rulesets/v2 All Other Regions: https://protectapi-{region-code}.cylance.com/rulesets/v2
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsruleset:create scope encoded.

Request

Post Detection Rule Set Content Request Schema

```
{
 "name": "My Detection Rule Set",
 "description": "My Detection Rule Set Description",
 "notification_message": "My Detection Rule Set Notification",
 "category": "Custom",
 "rules": [
  {
   "detection_rule_id": "4f722b34... ",
   "detection_rule_version": 0,
   "detection_name": "Dropper/Downloader",
   "detection_description": "A process has created a new executable file and then executed that file",
   "category": "Cylance Rules",
   "severity": "Medium",
   "operating_systems": [
    {
     "Name": "Windows"
    }
   ],
   "date_added": "2018-11-26T23:36:22.828Z",
   "enabled": true,
   "notification_enabled": true,
   "responses": [
    {
     "template_id": "4f722b34... ",
     "response_rule_id": "6eb12c34...",
     "response_rule_version": 0,
     "description": "Delete file",
     "value": {},
     "enabled": true,
     "created": "2018-11-26T23:36:22.828Z"
    }
   ],
   "exceptions": [
    {
     "exception_id": "d2cf6cc7... ",
     "enabled": true,
     "name": "My Detection Exception"
    }
   ],
   "playbooks": [
    "eefd51ea..."
   ]
  }
  ]
}
```

Response

202 OK – Ruleset created.

Post Detection Rule Set Content Response Schema

```
{
  "name": "My Detection Rule Set",
  "description": "My Detection Rule Set Description",
  "notification_message": "My Detection Rule Set Notification",
  "category": "Custom",
  “id”: “4568e4ac…
  "last_modified": "2018-11-26T23:36:22.828Z",
  "modified_by": {
    "id": "string",
    "login": "string"
  },
  "rules": [
    {
      "detection_rule_id": "4f722b34… ",
      "detection_rule_version": 0,
      "detection_name": "Dropper/Downloader",
      "detection_description": "A process has created a new executable file and then executed that file",
      "category": "Cylance Rules",
      "severity": "Medium",
      "operating_systems": [
        {
          "Name": "Windows"
        }
      ],
      "date_added": "2018-11-26T23:36:22.828Z",
      "enabled": true,
      "notification_enabled": true,
      "responses": [
        {
          "template_id": "4f722b34… ",
          "response_rule_id": "6eb12c34…",
          "response_rule_version": 0,
          "description": "Delete file",
          "value": {},
          "enabled": true,
          "created": "2018-11-26T23:36:22.828Z"
        }
      ],
   "exceptions": [
    {
     "exception_id": "d2cf6cc7... ",
     "enabled": true,
     "name": "My Detection Exception"
    }
   ],
   "playbooks": [
    "eefd51ea..."
   ]
  }
 ]
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
name	The name of the Detection Rule Set.
description	The description of the Detection Rule Set.
notification_message	The message to be displayed on the endpoint when a Detection Rule is triggered.
id	The unique ID of the Detection Rule Set.
last_modified	The timestamp (in UTC) of the last time that the Detection Rule Set was modified.
modified_by	An object detailing the last user to modify the Detection Rule Set. It includes the following fields: • id: The unique ID of the user who modified the Detection Rule Set. • login: The email address of the user who modified the Detection Rule Set.
rules	A list of Detection Rule objects and their associated Response Actions, Detection Exceptions, and Package Playbooks. Each object in the list will contain the following fields.
detection_rule_id	The unique ID of the Detection Rule.

detection_rule_version	The version of the Detection Rule.
detection_name	The name of the Detection Rule.
detection_description	The description of the Detection Rule Set.
category	The category of the Detection Rule.
severity	The severity assigned to the Detection Rule. Possible values are: • High • Medium • Low • Informational
operating_systems	An object detailing the operating systems that the Detection Rule can be applied to. It will include the “name” field. This can consist of: • “Windows” • “MacOS”
date_added	The timestamp (in UTC) that the Detection Rule was added to the tenant.
enabled	Determines whether or not a Detection Rule is enabled in the Detection Rule Set. When viewing the content of a Detection Rule Set, this should always be set to ‘true.’
notification_enabled	Determines whether or not the message defined in the ‘notification_message’ field should display on the device when the Detection Rule is triggered.
responses	A list of response objects for each Response Action enabled for a particular Detection Rule. Each object will include the following fields: • template_id: The ID of the response template to use (this is provided by Cylance.) • response_rule_id: The ID of the response rule to enable (these are provided by Cylance.) • response_rule_version: The version of the response rule to enable (this is provided by Cylance.) • description: The description / name of the response rule. • value: A currently unused field. • enabled: This will always be ‘true’ when viewing a Detection Rule Set. • created: The date that the Response Rule was added to the tenant.
exceptions	A list of Exception Rule objects that should be applied to the Detection Rule. Each object will include the following fields: • exception_id: The unique ID of the Exception Rule. • enabled: This will always be ‘true’ when viewing a Detection Rule Set. • name: The name of the Exception Rule.
playbooks	A list of Package Playbook unique IDs that will be executed when the Detection Rule is triggered on the device.

Due to the complexity and dependency on correct unique IDs for Detection Rules, Playbooks, Exceptions, and Response Actions. It is recommended that the caller make a GET request to *'/rulesets/v2/default'* to obtain a properly formatted default Detection Rule Set template which can then be modified and submitted in a POST request. The 'default' template is covered in the next section.

Retrieve Default Detection Rule Set

Allows a caller to retrieve a properly formatted default Detection Rule Set template that includes all of the Detection Rules, Exceptions, Playbooks, and Response Actions available in a tenant. The output of this request can be modified and submitted as a POST request to *'rulesets/v2'* to create a new Detection Rule Set. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule Set resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rulesets/v2/default US Government: https://protectapi.us.cylance.com/rulesets/v2/default All Other Regions: https://protectapi-{region-code}.cylance.com/rulesets/v2/default
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsruleset:read scope encoded.
Request	None
Response	200 OK Get Default Detection Rule Set Content Response Schema

```
{
  "name": "",
  "description": "",
  "notification_message": "",
  "category": "Custom",
  "rules": [
    {
      "detection_rule_id": "4f722b34... ",
      "detection_rule_version": 0,
      "detection_name": "Dropper/Downloader",
      "detection_description": "A process has created a new executable file and then executed that file",
      "category": "Cylance Rules",
      "severity": "Medium",
      "operating_systems": [
        {
          "Name": "Windows"
        }
      ],
      "date_added": "2018-11-26T23:36:22.828Z",
      "enabled": true,
      "notification_enabled": true,
      "responses": [
        {
          "template_id": "4f722b34… ",
          "response_rule_id": "6eb12c34…",
          "response_rule_version": 0,
          "description": "Delete file",
          "value": {},
          "enabled": true,
          "created": "2018-11-26T23:36:22.828Z"
        }
      ],
      "exceptions": [
        {
          "exception_id": "d2cf6cc7… ",
          "enabled": true,
          "name": "My Detection Exception"
        }
      ],
      "playbooks": [
        "eefd51ea…"
      ]
    }
  ]
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
name	The name of the Detection Rule Set.

description	The description of the Detection Rule Set.
notification_message	The message to be displayed on the endpoint when a Detection Rule is triggered.
rules	A list of Detection Rule objects and their associated Response Actions, Detection Exceptions, and Package Playbooks. Each object in the list will contain the following fields.
detection_rule_id	The unique ID of the Detection Rule.
detection_rule_version	The version of the Detection Rule.
detection_name	The name of the Detection Rule.
detection_description	The description of the Detection Rule Set.
category	The category of the Detection Rule
severity	The severity assigned to the Detection Rule. Possible values are: • High • Medium • Low • Informational
operating_systems	An object detailing the operating systems that the Detection Rule can be applied to. It will include the "name" field. This can consist of: • "Windows" • "MacOS"
date_added	The timestamp (in UTC) that the Detection Rule was added to the tenant.
enabled	Determines whether or not a Detection Rule is enabled in the Detection Rule Set. When viewing the content of a Detection Rule Set, this should always be set to 'true.'
notification_enabled	Determines whether or not the message defined in the 'notification_message' field should be displayed on the device when the Detection Rule is triggered.
responses	A list of response objects for each Response Action enabled for a particular Detection Rule. Each object will include the following fields: • template_id: The ID of the response template to use (this is provided by Cylance.) • response_rule_id: The ID of the response rule to enable (these are provided by Cylance.) • response_rule_version: The version of the response rule to enable (this is provided by Cylance.) • description: The description / name of the response rule. • value: A currently unused field. • enabled: This will always be 'true' when viewing a Detection Rule Set • created: The date that the Response Rule was added to the tenant.

exceptions	A list of Exception Rule objects that should be applied to the Detection Rule. Each object will include the following fields: • exception_id: The unique ID of the Exception Rule. • enabled: This will always be 'true' when viewing a Detection Rule Set. • name: The name of the Exception Rule.
playbooks	A list of Package Playbook unique IDs that will be executed when the Detection Rule is triggered on the device.

Update Detection Rule Set

Allows a caller to update a Detection Rule Set by sending a new JSON structure. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule Set resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rulesets/v2/{ruleset_id}
	US Government: https://protectapi.us.cylance.com/rulesets/v2/{ruleset_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/rulesets/v2/{ruleset_id}

Method	HTTP/1.1 PUT
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsruleset:update scope encoded.
Request	Update Detection Rule Content Request Schema

```
{
 "name": "My Detection Rule Set",
 "description": "My New Detection Rule Set Description",
 "notification_message": "My New Detection Rule Set Notification",
 "category": "Custom",
 "rules": [
  {
   "detection_rule_id": "adceb34… ",
   "detection_rule_version": 1,
   "detection_name": "Powershell Download",
   "detection_description": "Powershell was detected attempting to download a file from a remote location.",
   "category": "Cylance Rules",
   "severity": "Medium",
   "operating_systems": [
    {
     "Name": "Windows"
    }
   ],
   "date_added": "2018-11-26T23:36:22.828Z",
      "enabled": true,
      "notification_enabled": true,
      "responses": [
        {
          "template_id": "4f722b34… ",
          "response_rule_id": "6eb12c34…",
          "response_rule_version": 0,
          "description": "Delete file",
          "value": {},
          "enabled": true,
          "created": "2018-11-26T23:36:22.828Z"
        }
      ],
      "exceptions": [
        {
          "exception_id": "d2cf6cc7… ",
          "enabled": true,
          "name": "My Detection Exception"
        }
      ],
      "playbooks": []
    }
  ]
}
```

Response

202 OK – Ruleset updated.

Update Detection Rule Content Response Schema

```
{
  "name": "My Detection Rule Set",
  "description": "My New Detection Rule Set Description",
  "notification_message": "My New Detection Rule Set Notification",
  "category": "Custom",
  "rules": [
    {
      "detection_rule_id": "adceb34… ",
      "detection_rule_version": 1,
      "detection_name": "Powershell Download",
      "detection_description": "Powershell was detected attempting to download a file from a remote location.",
      "category": "Cylance Rules",
      "severity": "Medium",
      "operating_systems": [
        {
          "Name": "Windows"
        }
      ],
      "date_added": "2018-11-26T23:36:22.828Z",
      "enabled": true,
      "notification_enabled": true,
      "responses": [
       {
        "template_id": "4f722b34… ",
        "response_rule_id": "6eb12c34…",
        "response_rule_version": 0,
        "description": "Delete file",
        "value": {},
        "enabled": true,
        "created": "2018-11-26T23:36:22.828Z"
       }
      ],
      "exceptions": [
       {
        "exception_id": "d2cf6cc7… ",
        "enabled": true,
        "name": "My Detection Exception"
       }
      ],
      "playbooks": []
    }
  ]
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such Detection Rule found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
name	The name of the Detection Rule Set.
description	The description of the Detection Rule Set.

notification_message	The message to be displayed on the endpoint when a Detection Rule is triggered.
rules	A list of Detection Rule objects and their associated Response Actions, Detection Exceptions, and Package Playbooks. Each object in the list will contain the following fields.
detection_rule_id	The unique ID of the Detection Rule.
detection_rule_version	The version of the Detection Rule.
detection_name	The name of the Detection Rule.
detection_description	The description of the Detection Rule Set.
category	The category of the Detection Rule
severity	The severity assigned to the Detection Rule. Possible values are: • High • Medium • Low • Informational
operating_systems	An object detailing the operating systems that the Detection Rule can be applied to. It will include the "name" field. This can consist of: • "Windows" • "MacOS"
date_added	The timestamp (in UTC) that the Detection Rule was added to the tenant.
enabled	Determines whether or not a Detection Rule is enabled in the Detection Rule Set. When viewing the content of a Detection Rule Set, this should always be set to 'true.'
notification_enabled	Determines whether or not the message defined in the 'notification_message' field should be displayed on the device when the Detection Rule is triggered.
responses	A list of response objects for each Response Action enabled for a particular Detection Rule. Each object will include the following fields: • template_id: The ID of the response template to use (this is provided by Cylance.) • response_rule_id: The ID of the response rule to enable (these are provided by Cylance.) • response_rule_version: The version of the response rule to enable (this is provided by Cylance.) • description: The description / name of the response rule. • value: A currently unused field. • enabled: This will always be 'true' when viewing a Detection Rule Set • created: The date that the Response Rule was added to the tenant.
exceptions	A list of Exception Rule objects that should be applied to the Detection Rule. Each object will include the following fields:

	- exception_id: The unique ID of the Exception Rule. - enabled: This will always be 'true' when viewing a Detection Rule Set. - name: The name of the Exception Rule.
playbooks	A list of Package Playbook unique IDs that will be executed when the Detection Rule is triggered on the device.

Delete Detection Rule Set

Allows a caller to delete a Detection Rule Set. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule Set resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rulesets/v2/{ruleset_id}
	US Government: https://protectapi.us.cylance.com/rulesets/v2/{ruleset_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/rulesets/v2/{ruleset_id}

Method	HTTP/1.1 DELETE
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsruleset:delete scope encoded.
Request	None
Response	204 OK – Ruleset deleted 400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such resource found. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

Delete Multiple Detection Rule Sets

Allows a caller to delete multiple Detection Rule Sets in a single request. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule Set resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rulesets/v2 US Government: https://protectapi.us.cylance.com/rulesets/v2 All Other Regions: https://protectapi-{region-code}.cylance.com/rulesets/v2
Method	HTTP/1.1 DELETE

Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsruleset:delete scope encoded.

Request	Delete Detection Rule Sets Content Request Schema
	{
	  “ids”: [
	    “eefa15ea…”
	  ]
	}

Response	200 OK – Rulesets deleted
	Delete Detection Rule Sets Content Request Schema
	[
	  {
	    "id": " eefa15ea…",
	    "success": true,
	    "message": "string"
	  }
	]
	400 Bad Request – Malformed request.
	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.
	403 Forbidden – The JWT token did not contain the proper scope to perform this action.
	404 Not Found – No such resource found.
	500 InternalServerError – An unforeseeable error has occurred.
	503 Service unavailable – Please try again later.

The request JSON schema contains the following fields:

Field Name	Description
ids	A list of Detection Rule Set IDs to be deleted.

The response JSON schema contains the following fields:

Field Name	Description
id	A Detection Rule Set ID that was attempted to be deleted.

success	A Boolean field denoting whether or not the Detection Rule Set ID was deleted.
message	A string containing any error, success, or warning messages.

Get Detection Rule Set CSV List

Allows a caller to retrieve a CSV where every line represents a Detection Rule Set available in the tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Detection Rule Set resource with.

Service Endpoint	North America: https://protectapi.cylance.com/rulesets/v2/csv US Government: https://protectapi.us.cylance.com/rulesets/v2/csv All Other Regions: https://protectapi-{region-code}.cylance.com/rulesets/v2/csv
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsruleset:list scope encoded.
Request	None
Response	200 OK – Will initiate a CSV file download. 400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such resource found. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

The response CSV contains the following fields:

Field Name	Description
Id	The unique ID of the Exception.
Last Modified	The timestamp (in UTC) of the last time that the Detection Rule Set was modified.
Modified By	The email address of the user who last modified the Detection Rule Set.
Name	The name of the Detection Rule Set.
Description	The description of the Detection Rule Set.

Notification	The Notification Message to display on a device if the Detection Rule triggers.
Category	The category of the Detection Rule Set.
Device Count	The number of devices that have the Detection Rule Set applied.

CylanceOPTICS Detection Exceptions

The CylanceOPTICS Detection Exceptions API allows users to add exceptions to their detection rules. Users can create a Detection Exception from a false positive detection, from the Detection Summary page, and from the Detection Details page.

The CylanceOPTICS Detection Exceptions API includes:

- Getting the content for a Detection Exception
- Getting a list of Detection Exceptions for a tenant
- Getting a list of Detection Exceptions as a CSV file
- Creating a Detection Exception
- Updating a Detection Exception
- Deactivating (or soft deleting) a Detection Exception

Note: This addendum only covers CylanceOPTICS Detection Exceptions API information. Read the Cylance User API guide for configuration information.

Get Detection Exceptions List

Allows a caller to retrieve a list of Exception rules available in a tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Exception resource with.

Service Endpoint	North America: https://protectapi.cylance.com/exceptions/v2 US Government: https://protectapi.us.cylance.com/exceptions/v2 All Other Regions: https://protectapi-{region-code}.cylance.com/exceptions/v2
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsexception:list scope encoded.
Request	None

Response	200 OK
	Get Detection Exception List Response Schema
	{
	  “page_size”: 10,
	  “total_pages”: 1
	  "page_items": [

```
    {
      "Id": "631015e5...",
      "Name": "My Exception",
      "Description": "My Exception Description",
      "DeviceCount": 0,
      "LastModified": "2018-11-26T23:36:23.000Z",
      "ModifiedBy": {
        "id": "382eabd4...",
        "login": "user@test.com"
      },
      "OperatingSystems": [
        {
          "Name": "Windows"
        }
      ],
      "PolicyCount": 0,
      "RulesetCount": 0,
      "Version": 1
    }
  ],
  “total_number_of_items”: 10
“page_number”: 1
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
page_size	The number of items on the page.
total_pages	The total number of pages of this size.
total_number_of_items	The total number of Exceptions in the tenant.
page_number	The current page number of results.
page_items	A list of Exception objects that are available in the tenant that will contain the following fields.

Id	The unique ID of the Exception.
Name	The name of the Exception.
description	The description of the Exception.
DeviceCount	The number of devices that have the Exception applied.
LastModified	The timestamp (in UTC) of the last time that the Exception was modified.
ModifiedBy	An object detailing the last user to modify the Exception. It includes the following fields: • id: The unique ID of the user who modified the Exception. • login: The email address of the user who modified the Exception.
OperatingSystems	An object detailing the operating systems that the Exception can be applied to. It will include the "name" field. This can consist of: • "Windows" • "MacOS"
PolicyCount	The number of policies that have the Exception applied.
RulesetCount	The number of Detection Rule Sets that have the Exception enabled.
Version	The version of the Exception.

Get Detection Exception Content

Allows a caller to retrieve the content of an Exception in its native JSON structure. The structure of CylanceOPTICS Exception Rules is beyond the scope of this document. Refer to the CylanceOPTICS Context Analysis Engine Custom Rule Guide for a more detailed explanation. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Exception resource with.

Service Endpoint	North America: https://protectapi.cylance.com/exceptions/v2/{exception_id}
	US Government: https://protectapi.us.cylance.com/exceptions/v2/{exception_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/exceptions/v2/{exception_id}

Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsexception:read scope encoded.
Request	None

Response

200 OK

Get Detection Exception Content Response Schema

```
{
 "Id": "631015e5...",
 "Name": "My Exception",
 "Description": "My Exception Description",
 "ObjectType": "ExceptionRule",
 "OperatingSystems": [
  {
   "Name": "Windows"
  }
 ],
 "Plugin": {
  "Name": "OpticsDetector"
 },
 "Product": {
  "Name": "CylanceOPTICS"
 },
 "SchemaVersion": 1,
 "States": [
  {
   "Name": "UnsignedProc",
   "Scope": "Global",
   "Function": "Function",
   "FieldOperators": {
    "Function": {
     "Type": "EqualsAny",
     "Operands": [
      {
       "Source": "LiteralSet",
       "Data": "iexplore.exe"
      }
     ],
     "OperandType": "string",
     "Options": {
      "IgnoreCase": true
     }
    }
   },
   "Actions": [
    {
     "Type": "AOI",
     "ItemName": "InstigatingProcess",
     "Postiion": "PostActivation"
```

```
        }
      ]
    }
  ],
  "Tags": [
    "CylanceOPTICS, Exception"
  ],
  "Version": 1
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such exception found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the entirety of the Exception logic. This schema is similar to that of the CylanceOPTICS Context Analysis Engine rules which are further explained in the 'CylanceOPTICS Context Analysis Engine Custom Rules Guide' KB Article.

Get Detection Exceptions CSV List

Allows a caller to retrieve a CSV where every line represents an Exception rule available in the tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Exception resource with.

Service Endpoint	North America: https://protectapi.cylance.com/exceptions/v2/csv US Government: https://protectapi.us.cylance.com/exceptions/v2/csv All Other Regions: https://protectapi-{region-code}.cylance.com/exceptions/v2/csv
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsexception:list scope encoded.
Request	None

Response	200 OK – Will initiate a CSV file download. 400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such resource found. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

The response CSV contains the following fields:

Field Name	Description
Name	The name of the Exception.
Id	The unique ID of the Exception.
Version	The version of the Exception.
Description	The description of the Exception.
Last Modified	The timestamp (in UTC) of the last time that the Exception was modified.
Modified By	The email address of the user who last modified the Exception.
Device Count	The number of devices that have the Exception applied.
Ruleset Count	The number of Detection Rule Sets that have the Exception enabled.

Create Detection Exception

Allows a caller to create a new Detection Exception by sending the native JSON structure of a Detection Exception. The structure of CylanceOPTICS Exception Rules is beyond the scope of this document. Refer to the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article for a more detailed explanation. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Exception resource with.

Service Endpoint	North America: https://protectapi.cylance.com/exceptions/v2 US Government: https://protectapi.us.cylance.com/exceptions/v2 All Other Regions: https://protectapi-{region-code}.cylance.com/exceptions/v2
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsexception:create scope encoded.

Request

Post Detection Exception Content Request Schema

```
{
  "Name": "My Exception",
  "Description": "My Exception Description",
  "ObjectType": "ExceptionRule",
  "OperatingSystems": [
    {
      "Name": "Windows"
    }
  ],
  "Plugin": {
    "Name": "OpticsDetector"
  },
  "Product": {
    "Name": "CylanceOPTICS"
  },
  "SchemaVersion": 1,
  "States": [
    {
      "Name": "UnsignedProc",
      "Scope": "Global",
      "Function": "Function",
      "FieldOperators": {
        "Function": {
          "Type": "EqualsAny",
          "Operands": [
            {
              "Source": "LiteralSet",
              "Data": "iexplore.exe"
            }
          ],
          "OperandType": "string",
          "Options": {
            "IgnoreCase": true
          }
        }
      },
      "Actions": [
        {
          "Type": "AOI",
          "ItemName": "InstigatingProcess",
          "Postiion": "PostActivation"
        }
      ]
    }
  ],
  "Tags": [
    "CylanceOPTICS, Exception"
  ]
}
```

Response

202 OK – ExceptionRule created successfully.

Post Detection Exception Content Response Schema

```
{
  "Id": "631015e5…",
  "Name": "My Exception",
  "Description": "My Exception Description",
  "ObjectType": "ExceptionRule",
  "OperatingSystems": [
    {
      "Name": "Windows"
    }
  ],
  "Plugin": {
    "Name": "OpticsDetector"
  },
  "Product": {
    "Name": "CylanceOPTICS"
  },
  "SchemaVersion": 1,
  "States": [
    {
      "Name": "UnsignedProc",
      "Scope": "Global",
      "Function": "Function",
      "FieldOperators": {
        "Function": {
          "Type": "EqualsAny",
          "Operands": [
            {
              "Source": "LiteralSet",
              "Data": "iexplore.exe"
            }
          ],
          "OperandType": "string",
          "Options": {
            "IgnoreCase": true
          }
        }
    },
    "Actions": [
     {
       "Type": "AOI",
       "ItemName": "InstigatingProcess",
       "Postiion": "PostActivation"
     }
    ]
  }
 ],
 "Tags": [
   "CylanceOPTICS, Exception"
 ],
 "Version": 1
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the entirety of the Exception logic. This schema is similar to that of the CylanceOPTICS Context Analysis Engine rules which are further explained in the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article.

Note that the 'id' and 'version' fields are automatically populated when the request is submitted.

Update Detection Exception

Allows a caller to update a Detection Exception by sending a new JSON structure. The structure of CylanceOPTICS Exception Rules is beyond the scope of this document. Refer to the CylanceOPTICS Context Analysis Engine Custom Rule Guide for a more detailed explanation. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Exception resource with.

Service Endpoint	North America: https://protectapi.cylance.com/exceptions/v2/{exception_id}
	US Government: https://protectapi.us.cylance.com/exceptions/v2/{exception_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/exceptions/v2/{exception_id}

Method	HTTP/1.1 PUT

Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsexception:update scope encoded.

Request

Update Detection Exception Content Request Schema

```
{
 "Name": "My Exception",
 "Description": "My Exception Description",
 "ObjectType": "ExceptionRule",
 "OperatingSystems": [
  {
   "Name": "Windows"
  }
 ],
 "Plugin": {
  "Name": "OpticsDetector"
 },
 "Product": {
  "Name": "CylanceOPTICS"
 },
 "SchemaVersion": 1,
 "States": [
  {
   "Name": "UnsignedProc",
   "Scope": "Global",
   "Function": "Function",
   "FieldOperators": {
    "Function": {
     "Type": "EqualsAny",
     "Operands": [
      {
       "Source": "LiteralSet",
       "Data": "my_application.exe"
      }
     ],
     "OperandType": "string",
     "Options": {
      "IgnoreCase": true
     }
    }
   },
   "Actions": [
    {
     "Type": "AOI",
     "ItemName": "InstigatingProcess",
        "Postiion": "PostActivation"
      }
    ]
  }
 ],
 "Tags": [
  "CylanceOPTICS, Exception"
 ]
}
```

Response

202 OK – ExceptionRule created successfully.

Update Detection Exception Content Response Schema

```
{
 "Id": "631015e5...",
 "Name": "My Exception",
 "Description": "My Exception Description",
 "ObjectType": "ExceptionRule",
 "OperatingSystems": [
  {
   "Name": "Windows"
  }
 ],
 "Plugin": {
  "Name": "OpticsDetector"
 },
 "Product": {
  "Name": "CylanceOPTICS"
 },
 "SchemaVersion": 1,
 "States": [
  {
   "Name": "UnsignedProc",
   "Scope": "Global",
   "Function": "Function",
   "FieldOperators": {
    "Function": {
     "Type": "EqualsAny",
     "Operands": [
      {
       "Source": "LiteralSet",
       "Data": "my_application.exe"
      }
     ],
     "OperandType": "string",
     "Options": {
      "IgnoreCase": true
     }
    }
   },
   "Actions": [
    {
     "Type": "AOI",
     "ItemName": "InstigatingProcess",
     "Postiion": "PostActivation"
    }
   ]
  }
 ],
 "Tags": [
  "CylanceOPTICS, Exception"
 ],
 "Version": 2
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such Exception found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the entirety of the Exception logic. This schema is similar to that of the CylanceOPTICS Context Analysis Engine rules which are further explained in the CylanceOPTICS Context Analysis Engine Custom Rule Guide knowledge base article.

Note that the 'id' and 'version' fields are automatically populated when the request is submitted.

Deactivate / Delete Detection Exception

Allows a caller to 'soft delete' a Detection Exception and remove it from Detection Rule Sets. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Exception resource with.

Service Endpoint	North America: https://protectapi.cylance.com/exceptions/v2/{exception_id}/deactivate US Government: https://protectapi.us.cylance.com/exceptions/v2/{exception_id}/deactivate All Other Regions: https://protectapi-{region-code}.cylance.com/exceptions/v2/{exception_id}/deactivate

Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsexception:update scope encoded.
Request	None
Response	200 OK 400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such resource found. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

Note that Detection Rule Sets are not automatically communicated to all endpoints when updates to Detection Exceptions are made. To ensure that the latest logic is applied to endpoints in the quickest manner, re-save any affected Detection Rule Sets (either via the UI or API.)

CylanceOPTICS Device Commands

The CylanceOPTICS Device Commands API allows users to perform actions on the endpoint. For example, locking down an endpoint or retrieving a file from an endpoint.

The CylanceOPTICS Device Commands API includes:

- Locking down an endpoint
- Getting Device Lockdown History for a tenant
- Requesting a file retrieval from an endpoint
- Checking the file retrieval status for an endpoint
- Getting the retrieved file results

Note: This addendum only covers CylanceOPTICS Device Commands API information. Read the Cylance User API guide for configuration information.

Lockdown Device Command

Allows a caller to create a CylanceOPTICS device lockdown command resource for a specific device. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the device lockdown command resource with.

Service Endpoint	North America: https://protectapi.cylance.com/devicecommands/v2/{deviceID}/lockdown?value=true&expires=d US Government: https://protectapi.us.cylance.com/devicecommands/v2/{deviceID}/lockdown?value=true&expires=d All Other Regions: https://protectapi-{region-code}.cylance.com/devicecommands/v2/{deviceID}/lockdown?value=true&expires=d
Method	HTTP/1.1 PUT
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticscommand:create scope encoded.
Request	Append the following optional query string parameters: • value: Whether to lockdown or not. Default to true. • expires: Duration of the lockdown. Format: ‘d:hh:mm’. o Maximum: 3 days. o Minimum: 5 minutes.
Response	201 OK – Command created Put Lockdown Command Response Schema

```
{
  "id": "DEVICE_ID",
  "hostname": "TEST_HOST",
  "tenant_id": "TENANT_ID",
  "connection_status": "connected",
  "optics_device_version": "2.1.1000.478",
  "password": "foo",
  "lockdown_expiration": "2016-01-01T00:00:00Z",
  "lockdown_initiated": "2016-01-01T00:00:00Z",
  "lockdown_history": [
   {
     "user_id": "USER_ID",
     "timestamp": "2016-01-01T00:00:00Z",
     "command": "ClearLockdown"
   }
  ]
}
```

	400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such device. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
id	The unique device ID that the lockdown command was issued to. Note: See About Device ID for device ID formatting.
hostname	The hostname of the device that the lockdown command was issued to.
tenant_id	The unique tenant ID of the tenant that the device belongs to.
connection_status	Displays whether or not the device is connected to Cylance's cloud services.
optics_device_version	Returns the numerical version of CylanceOPTICS that the device is running.
password	The password required to unlock the device.
lockdown_expiration	The timestamp (in UTC) of when the current device lockdown is set to expire.
lockdown_initiated	The timestamp (in UTC) of when the current device lockdown was initiated.
lockdown_history	A list of historical device lockdown commands issued to a particular device.
user_id	The unique ID of the user who locked down the device.
timestamp	The timestamp (in UTC) of when the command was initiated.
command	The command that was executed.

Get Device Lockdown History

Allows a caller to request the current lockdown state and lockdown history for a specific device. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the device lockdown command resource with.

Service Endpoint	North America: https://protectapi.cylance.com/devicecommands/v2/{deviceID}/lockdown US Government: https://protectapi.us.cylance.com/devicecommands/v2/{deviceID}/lockdown All Other Regions: https://protectapi-{region-code}.cylance.com/devicecommands/v2/{deviceID}/lockdown

Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticscommand:read scope encoded.
Request	None

Response	200 OK
	Get Lockdown Status and History Response Schema
	{
	"id": "DEVICE_ID",
	"hostname": "TEST_HOST",
	"tenant_id": "TENANT_ID",
	"connection_status": "connected",
	"optics_device_version": "2.1.1000.478",
	"password": "foo",
	"lockdown_expiration": "2016-01-01T00:00:00Z",
	"lockdown_initiated": "2016-01-01T00:00:00Z",
	"lockdown_history": [
	{
	"user_id": "USER_ID",
	"timestamp": "2016-01-01T00:00:00Z",
	"command": "ClearLockdown"
	}
	]
	}
	400 Bad Request – Malformed request.
	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.
	403 Forbidden – The JWT token did not contain the proper scope to perform this action.
	404 Not Found – No such device.
	500 InternalServerError – An unforeseeable error has occurred.
	503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
id	The unique device ID that the lockdown command was issued to. Note: See About Device ID for device ID formatting.
hostname	The hostname of the device that the lockdown command was issued to.
tenant_id	The unique tenant ID of the tenant that the device belongs to.
connection_status	Displays whether or not the device is connected to Cylance's cloud services.
optics_device_version	Returns the numerical version of CylanceOPTICS that the device is running.
password	The password required to unlock the device.
lockdown_expiration	The timestamp (in UTC) of when the current device lockdown is set to expire.
lockdown_initiated	The timestamp (in UTC) of when the current device lockdown was initiated.
lockdown_history	A list of historical device lockdown commands issued to a particular device.
user_id	The unique ID of the user who locked down the device.
timestamp	The timestamp (in UTC) of when the command was initiated.
command	The command that was executed.

Request File Retrieval from Device

Allows a caller to request that the specified file be retrieved from a specified device and stored in Cylance's cloud console for later analysis. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the file retrieval command resource with.

Service Endpoint	North America: https://protectapi.cylance.com/devicecommands/v2/{deviceID}/getfile US Government: https://protectapi.us.cylance.com/devicecommands/v2/{deviceID}/getfile All Other Regions: https://protectapi-{region-code}.cylance.com/devicecommands/v2/{deviceID}/getfile
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticscommand:read scope encoded.

Request	Get Request File Retrieval Request Schema
	{
	  “file_path”: “C:\path\to\file.txt”
	}

Response

200 OK

Get Request File Retrieval Response Schema

```
{
  "data": {
    "tenant_id": "TENANT_ID",
    "user_id": "USER_ID",
    "device_id": "DEVICE_ID",
    "created_at": "2017-01-01T00:00:00Z",
    "filepath": "REQUESTED_FILE_PATH",
    "download_url": "UNIQUE_URL”
    "file_status": "PENDING",
    "file_status_description": "Too Large | Does Not Exist",
    "password": "foo",
    "md5": "d41d8cd98f00b204e9800998ecf8427e",
    "sha1": "da39a3ee5e6b4b0d3255bfef95601890afd80709",
    "sha256": "e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",
    "correlation_id": "00000000000000000000000000000000",
    "user_login": "test@cylance.com",
    "hostname": "Test-PC"
  }
}
```

400 BadRequest – Returned for the following reasons:

- The tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The page number or page size specified are less than or equal to zero.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The requested device does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Server Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
data	An object containing the various fields associated with the file retrieval request.
tenant_id	The unique ID of the tenant associated with the file retrieval request.
user_id	The unique ID of the user who requested the file retrieval.

device_id	The unique ID of the device that the file retrieval was requested on. Note: See About Device ID for device ID formatting.
created_at	The timestamp (in UTC) of when the file retrieval was requested.
filepath	The file path of the requested file.
download_url	The unique URL and parameters required to download the retrieved file.
file_status	The status of the file retrieval. This will always be “PENDING” for newly created file retrievals.
file_status_description	Displays any errors or status messages associated with the retrieval request.
password	The password required to decrypt the retrieved file.
md5	The MD5 hash of the retrieved file.
sha1	The SHA1 hash of the retrieved fie.
sha256	The SHA256 hash of the retrieved file.
correlation_id	The correlation ID associated with this action.
user_login	The email address of the user who initiated the file retrieval request
hostname	The hostname of the device that the file retrieval was requested on.

Check File Retrieval Status from Device

Allows a caller to check the status of a previously requested file retrieval operation. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the file retrieval command resource with.

Service Endpoint	North America: https://protectapi.cylance.com/devicecommands/v2/{deviceID}/getfile:get US Government: https://protectapi.us.cylance.com/devicecommands/v2/{deviceID}/getfile:get All Other Regions: https://protectapi-{region-code}.cylance.com/devicecommands/v2/{deviceID}/getfile:get
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticscommand:read scope encoded.

Request	Get File Retrieval Status Request Schema
	{
	  “file_path”: “C:\path\to\file.txt”
	}

Response

200 OK

Get File Retrieval Status Response Schema

```
{
  "data": {
    "tenant_id": "TENANT_ID",
    "user_id": "USER_ID",
    "device_id": "DEVICE_ID",
    "created_at": "2017-01-01T00:00:00Z",
    "filepath": "REQUESTED_FILE_PATH",
    "download_url": " UNIQUE_URL”
    "file_status": "REQUEST | RETRY_REQUEST | PENDING | AVAILABLE | UNAVAILABLE | DOES_NOT_EXIST",
    "file_status_description": "Too Large | Does Not Exist",
    "password": "foo",
    "md5": "d41d8cd98f00b204e9800998ecf8427e",
    "sha1": "da39a3ee5e6b4b0d3255bfef95601890afd80709",
    "sha256": "e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",
    "correlation_id": "00000000000000000000000000000000",
    "user_login": "test@cylance.com",
    "hostname": "Test-PC"
  }
}
```

400 BadRequest – Returned for the following reasons:

- The Tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The unique identifier for the detection is not valid.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The InstaQuery resource does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
data	An object containing the various fields associated with the file retrieval request.
tenant_id	The unique ID of the tenant associated with the file retrieval request.
user_id	The unique ID of the user who requested the file retrieval.

device_id	The unique ID of the device that the file retrieval was requested on. Note: See About Device ID for device ID formatting.
created_at	The timestamp (in UTC) of when the file retrieval was requested.
filepath	The file path of the requested file.
download_url	The unique URL and parameters required to download the retrieved file.
file_status	The status of the file retrieval. Possible values are: • REQUEST: The file retrieval has not been requested, but the user may issue a request for it. • RETRY_REQUEST: The file retrieval has been requested previously but no results were received. It can be requested again. • PENDING: The file retrieval has been requested but has not yet been completed. • DOES_NOT_EXIST: The file retrieval has been requested but is not present on the device. • AVAILABLE: The file is available for download. A download link (found in the download_url field) is generated and valid for the next 10 minutes. • UNAVAILABLE: The file is not available. This status may indicate that the requested device is not online, or the requested device failed to upload the file. This status will become RETRY_REQUEST after an hour.
file_status_description	Displays any errors or status messages associated with the retrieval request.
password	The password required to decrypt the retrieved file.
md5	The MD5 hash of the retrieved file.
sha1	The SHA1 hash of the retrieved fie.
sha256	The SHA256 hash of the retrieved file.
correlation_id	The correlation ID associated with this action.
user_login	The email address of the user who initiated the file retrieval request
hostname	The hostname of the device that the file retrieval was requested on.

Get Retrieved Files Results

Allows a caller to obtain a history of file retrieval requests for all devices in the tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the file retrieval command resource with.

Service Endpoint	North America: https://protectapi.cylance.com/devicecommands/v2/retrieved_files?page=m&page_size=n US Government: https://protectapi.us.cylance.com/devicecommands/v2/ retrieved_files?page=m&page_size=n All Other Regions: https://protectapi-{region-code}.cylance.com/devicecommands/v2 retrieved_files?page=m&page_size=n
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticscommand:read scope encoded.
Request	Append the following query string parameters: • q: Case-insensitive search term. • page: The page number to request. Defaults to 1. • page_size: The number of file retrieval records to retrieve per page. Defaults to 20. • sort: Sort by field (adding ‘-‘ in front of the value denotes descending order.)

Response	200 OK
	Get Retrieved File Results Response Schema
	[
	  {
	    "tenant_id": "TENANT_ID",
	    "user_id": "USER_ID",
	    "device_id": "DEVICE_ID",
	    "created_at": "2017-01-01T00:00:00Z",
	    "filepath": "REQUESTED_FILE_PATH”,
	    "download_url": "UNIQUE_URL”
	    "file_status": "REQUEST | RETRY_REQUEST | PENDING | AVAILABLE | UNAVAILABLE | DOES_NOT_EXIST",
	    "file_status_description": "Too Large | Does Not Exist",
	    "password": "foo",
	    "md5": "d41d8cd98f00b204e9800998ecf8427e",
	    "sha1": "da39a3ee5e6b4b0d3255bfef95601890afd80709",
	    "sha256": "e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",
	    "correlation_id": "00000000000000000000000000000000",
	    "user_login": "test@cylance.com",
	    "hostname": "Test-PC"
	  }
	]

	400 BadRequest – Returned for the following reasons: • The tenant ID could not be retrieved from the JWT token specified in the Authorization header. • The unique identifier for the execution is not valid. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 NotFound – The InstaQuery resource does not exist. 500 InternalServerError – An unforeseeable error has occurred. 503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
data	An object containing the various fields associated with the file retrieval request.
tenant_id	The unique ID of the tenant associated with the file retrieval request.
user_id	The unique ID of the user who requested the file retrieval.
device_id	The unique ID of the device that the file retrieval was requested on. Note: See About Device ID for device ID formatting.
created_at	The timestamp (in UTC) of when the file retrieval was requested.
filepath	The file path of the requested file.
download_url	The unique URL and parameters required to download the retrieved file.
file_status	The status of the file retrieval. Possible values are: • REQUEST: The file retrieval has not been requested, but the user may issue a request for it. • RETRY_REQUEST: The file retrieval has been requested previously but no results were received. It can be requested again. • PENDING: The file retrieval has been requested but has not yet been completed. • DOES_NOT_EXIST: The file retrieval has been requested but is not present on the device. • AVAILABLE: The file is available for download. A download link (found in the download_url field) is generated and valid for the next 10 minutes. UNAVAILABLE: The file is not available. This status may indicate that the requested device is not online, or the requested device failed to upload the file. This status will become RETRY_REQUEST after an hour.

file_status_description	Displays any errors or status messages associated with the retrieval request.
password	The password required to decrypt the retrieved file.
md5	The MD5 hash of the retrieved file.
sha1	The SHA1 hash of the retrieved fie.
sha256	The SHA256 hash of the retrieved file.
correlation_id	The correlation ID associated with this action.
user_login	The email address of the user who initiated the file retrieval request
hostname	The hostname of the device that the file retrieval was requested on.

CylanceOPTICS Focus View

The CylanceOPTICS Focus View API allows users to retrieve an information trail starting with the first event related to an artifact from an InstaQuery result or a CylancePROTECT event.

The CylanceOPTICS Focus View API includes:

- Searching for Focus View results
- Generating a Focus View
- Getting a summary of a Focus View
- Getting the results of a Focus View
- Getting a list of Focus Views that have been made in a tenant

Note: This addendum only covers CylanceOPTICS Focus View API information. Read the Cylance User API guide for configuration information.

Get Focus View List

Allows a caller to retrieve a list of Focus Views that have been made in the tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Focus View resource with.

Service Endpoint	North America: https://protectapi.cylance.com/foci/v2?page=m&page_size=n US Government: https://protectapi.us.cylance.com/foci/v2?page=m&page_size=n All Other Regions: https://protectapi-{region-code}.cylance.com/foci/v2?page=m&page_size=n
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsfocus:read scope encoded.
Request	Append the following optional query string parameters: • q: Case-insensitive search term • page: The page number to request. Defaults to 1. • page_size: The number of detections records to retrieve per page. Defaults to 20. • sort: Sort by field (adding "-" in front of value denotes descending order).
Response	200 OK

Get Focus View List Response Schema

```
{
 “page_size”: 10,
 “total_pages”: 5,
 "page_items": [
  {
   "device_id": "D1411D976…",
   "artifact_type": "Process",
   "artifact_subtype": "Uid",
   "value": "CFA3681…",
   "threat_type": "",
   "description": "My Focus View Example",
   "id": "4D61A68…",
   "tenant_id": "F3C2A47…",
   "created_at": "2016-12-15T01:17:44Z",
   "hostname": "JSMITH2-WIN",
   "status": "AVAILABLE",
   "relations": [
    {
     "object": "/survey/survey_id",
     "relationship": "origin-of "
    }
   ]
  }
 ]
 "total_number_of_items": 47,
 "page_number": 1
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
page_size	The number of items per page.
total_pages	The total number of pages of this page size.
total_number_of_items	The total number of Focus Views available to the tenant.
page_number	The current page number.
page_items	A list of Focus View objects which will contain the following fields.
device_id	The unique ID of the Device that the Focus View was requested from. Note: See About Device ID for device ID formatting.
artifact_type	The type of Artifact that the Focus View was requested for. Possible values are: • Protect: Request a Focus View for a CylancePROTECT-generated event. • Process: Request a Focus View for a Process artifact to visualize how a process interacts with the device. (This is the most common option.) • File: Request a Focus View for a File artifact to visualize how the file has been interacted with. • NetworkConnection: Request a Focus View for a Network artifact to visualize communications associated with an IP address. RegistryKey: Request a Focus View for a Registry artifact to visualize how the registry key or path has been interacted with.
artifact_subtype	This field should always be "Uid" at this time.

value	The UID of the Artifact used to gather the Focus View.
threat_type	An option field to use with a "Protect" artifact_type to denote the type of threat that a Focus View is being generated for.
description	The human-readable description of the Focus View.
id	The unique ID of the Focus View.
tenant_id	The unique ID of the tenant associated with the Focus View.
created_at	The timestamp (in UTC) of when the Focus View was created.
hostname	The hostname of the device that the Focus View was requested from.
status	The status of the Focus View result or request. Possible values are: • AVAILABLE: A Focus View has been generated and is available for viewing. • PENDING: The Focus View has been requested. • REQUEST: The Focus View has not been generated, but it can be requested. • RETRY_REQUEST: The Focus View has not been generated. It was previously requested but no results were received. It can be requested again. • DOES_NOT_EXIST: The Focus View requested on the device cannot be completed because the requested parameters do not exist on the device. • UNAVAILABLE: The Focus View is not available, and the associated device is not online to fulfil the request. It can be requested at a later time. UNKNOWN_DEVICE: The Focus View is not available, and the associated device is no longer known.
relations	A list of objects that are related to this Focus View. The following fields can be contained: • Object: The URL of a Focus View, InstaQuery, or Detection Event that is linked to this Focus View. • Relationship: How the relationship was established.

Search for Focus View Results

Allows a caller to search for Focus Views by a list of Device ID and CylancePROTECT Event ID pairs, up to 200 at a time. The request requires both a CylancePROTECT Event ID and Device ID to determine whether or not a Focus View *can* be created. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Focus View resource with.

Service Endpoint	North America: https://protectapi.cylance.com/foci/v2/search US Government: https://protectapi.us.cylance.com/foci/v2/search All Other Regions: https://protectapi-{region-code}.cylance.com/foci/v2/search
Method	HTTP/1.1 POST

Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsfocus:list scope encoded.

Request	Post Search Focus View Results Request Schema
	[
	{
	"uid": "string",
	"device_id": "string",
	}
	]

Response	200 OK
	Post Search Focus View Results Response Schema
	[
	{
	"uid": "string",
	"device_id": "string",
	"status": "string",
	"focus_id": "string"
	}
	]
	400 Bad Request – Malformed request.
	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.
	403 Forbidden – The JWT token did not contain the proper scope to perform this action.
	404 Not Found – No such resource found.
	500 InternalServerError – An unforeseeable error has occurred.
	503 Service unavailable – Please try again later.

The request JSON schema contains the following fields:

Field Name	Description
uid	The unique ID of a CylancePROTECT event.
device_id	The unique ID of the Device that the CylancePROTECT event occurred on. Note: See About Device ID for device ID formatting.

The response JSON schema contains the following fields:

Field Name	Description
uid	The unique ID of a CylancePROTECT event.
device_id	The unique ID of the Device that the CylancePROTECT event occurred on. Note: See About Device ID for device ID formatting.
status	The status of a Focus View result or request. Possible values are: • AVAILABLE: A Focus View has been generated and is available for viewing. • PENDING: The Focus View has been requested. • REQUEST: The Focus View has not been generated, but it can be requested. • RETRY_REQUEST: The Focus View has not been generated. It was previously requested but no results were received. It can be requested again. • DOES_NOT_EXIST: The Focus View requested on the device cannot be completed because the requested parameters do not exist on the device. • UNAVAILABLE: The Focus View is not available, and the associated device is not online to fulfil the request. It can be requested at a later time. • UNKNOWN_DEVICE: The Focus View is not available, and the associated device is no longer known.
focus_id	The unique ID of the Focus View.

Generate a Focus View

Allows a caller to request a Focus View from a specified device. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Focus View resource with.

Service Endpoint	North America: https://protectapi.cylance.com/foci/v2 US Government: https://protectapi.us.cylance.com/foci/v2 All Other Regions: https://protectapi-{region-code}.cylance.com/foci/v2
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsfocus:create scope encoded.

Request	Post Generate Focus View Request Schema
	{
	"device_id": " D1411D976...",
	"artifact_type": "Process",
	"artifact_subtype": "Uid",
	"value": " CFA3681...",
	"threat_type": "",
	"description": "My Focus View Example”
	}

Response

201 OK – Focus Created

Post Generate Focus View Response Schema

```
{
  "device_id": " D1411D976...",
  "artifact_type": "Process",
  "artifact_subtype": "Uid",
  "value": " CFA3681...",
  "threat_type": "",
  "description": "My Focus View Example",
  "id": "4D61A68...”
  "tenant_id": " F3C2A47...",
  "created_at": "2016-12-15T01:17:44Z",
  "hostname": "JSMITH2-WIN",
  "status": "AVAILABLE",
  "relations": [
   {
     "object": "/focus/focus_id",
     "relationship": "originated-from"
   }
  ]
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such resource found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The request JSON schema contains the following fields:

Field Name	**Description**
device_id	The unique ID of the Device to request the Focus View from. Note: See About Device ID for device ID formatting.
artifact_type	The type of Artifact to request the Focus View for. Possible values are: • Protect: Request a Focus View for a CylancePROTECT-generated event.

	• Process: Request a Focus View for a Process artifact to visualize how a process interacts with the device. (This is the most common option.) • File: Request a Focus View for a File artifact to visualize how the file has been interacted with. • NetworkConnection: Request a Focus View for a Network artifact to visualize communications associated with an IP address. • RegistryKey: Request a Focus View for a Registry artifact to visualize how the registry key or path has been interacted with.
artifact_subtype	This field should always be "Uid" at this time.
value	The UID of the Artifact to gather a Focus View about. This can be obtained from InstaQuery results, another Focus View, the details/associated artifacts of a Detection Event, or anywhere else an Artifact is referenced.
threat_type	An option field to use with a "Protect" artifact_type to denote the type of threat that a Focus View is being generated for.
description	A human-readable description of the Focus View.

The response JSON schema contains the following fields:

Field Name	Description
device_id	The unique ID of the Device that the Focus View was requested from. Note: See About Device ID for device ID formatting.
artifact_type	The type of Artifact that the Focus View was requested for. Possible values are: • Protect: Request a Focus View for a CylancePROTECT-generated event. • Process: Request a Focus View for a Process artifact to visualize how a process interacts with the device. (This is the most common option.) • File: Request a Focus View for a File artifact to visualize how the file has been interacted with. • NetworkConnection: Request a Focus View for a Network artifact to visualize communications associated with an IP address. • RegistryKey: Request a Focus View for a Registry artifact to visualize how the registry key or path has been interacted with.
artifact_subtype	This field should always be "Uid" at this time.
value	The UID of the Artifact used to gather the Focus View.
threat_type	An option field to use with a "Protect" artifact_type to denote the type of threat that a Focus View is being generated for.
description	The human-readable description of the Focus View.

id	The unique ID of the Focus View.
tenant_id	The unique ID of the tenant associated with the Focus View.
created_at	The timestamp (in UTC) of when the Focus View was created.
hostname	The hostname of the device that the Focus View was requested from.
status	The status of the Focus View result or request. Possible values are: • AVAILABLE: A Focus View has been generated and is available for viewing. • PENDING: The Focus View has been requested. • REQUEST: The Focus View has not been generated, but it can be requested. • RETRY_REQUEST: The Focus View has not been generated. It was previously requested but no results were received. It can be requested again. • DOES_NOT_EXIST: The Focus View requested on the device cannot be completed because the requested parameters do not exist on the device. • UNAVAILABLE: The Focus View is not available, and the associated device is not online to fulfil the request. It can be requested at a later time. • UNKNOWN_DEVICE: The Focus View is not available, and the associated device is no longer known.
relations	A list of objects that are related to this Focus View. The following fields can be contained: • Object: The URL of a Focus View, InstaQuery, or Detection Event that is linked to this Focus View. • Relationship: How the relationship was established.

Get a Focus View Summary

Allows a caller to get the summary of an existing Focus View. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Focus View resource with.

Service Endpoint	North America: https://protectapi.cylance.com/foci/v2/{focus_id}
	US Government: https://protectapi.us.cylance.com/foci/v2/{focus_id}
	All Other Regions: https://protectapi-{region-code}.cylance.com/foci/v2/{focus_id}

Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsfocus:read scope encoded.
Request	None

Response

200 OK

Get Focus View Summary Response Schema

```
{
  "device_id": " D1411D976...",
  "artifact_type": "Process",
  "artifact_subtype": "Uid",
  "value": " CFA3681...",
  "threat_type": "",
  "description": "My Focus View Example",
  "id": "4D61A68...”
  "tenant_id": " F3C2A47...",
  "created_at": "2016-12-15T01:17:44Z",
  "hostname": "JSMITH2-WIN",
  "status": "AVAILABLE",
  "relations": [
   {
     "object": "/focus/focus_id",
     "relationship": "originated-from"
   }
  ]
}
```

400 Bad Request – Malformed request.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 Not Found – No such Focus View found.

500 InternalServerError – An unforeseeable error has occurred.

503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
device_id	The unique ID of the Device that the Focus View was requested from. Note: See About Device ID for device ID formatting.
artifact_type	The type of Artifact that the Focus View was requested for. Possible values are: • Protect: Request a Focus View for a CylancePROTECT-generated event.

	• Process: Request a Focus View for a Process artifact to visualize how a process interacts with the device. (This is the most common option.) • File: Request a Focus View for a File artifact to visualize how the file has been interacted with. • NetworkConnection: Request a Focus View for a Network artifact to visualize communications associated with an IP address. • RegistryKey: Request a Focus View for a Registry artifact to visualize how the registry key or path has been interacted with.
artifact_subtype	This field should always be "Uid" at this time.
value	The UID of the Artifact used to gather the Focus View.
threat_type	An option field to use with a "Protect" artifact_type to denote the type of threat that a Focus View is being generated for.
description	The human-readable description of the Focus View.
id	The unique ID of the Focus View.
tenant_id	The unique ID of the tenant associated with the Focus View.
created_at	The timestamp (in UTC) of when the Focus View was created.
hostname	The hostname of the device that the Focus View was requested from.
status	The status of the Focus View result or request. Possible values are: • AVAILABLE: A Focus View has been generated and is available for viewing. • PENDING: The Focus View has been requested. • REQUEST: The Focus View has not been generated, but it can be requested. • RETRY_REQUEST: The Focus View has not been generated. It was previously requested but no results were received. It can be requested again. • DOES_NOT_EXIST: The Focus View requested on the device cannot be completed because the requested parameters do not exist on the device. • UNAVAILABLE: The Focus View is not available, and the associated device is not online to fulfil the request. It can be requested at a later time. • UNKNOWN_DEVICE: The Focus View is not available, and the associated device is no longer known.
relations	A list of objects that are related to this Focus View. The following fields can be contained: • Object: The URL of a Focus View, InstaQuery, or Detection Event that is linked to this Focus View. • Relationship: How the relationship was established.

Get Focus View Results

Allows a caller to get the details of an existing Focus View that is used to generate the chart and table in the UI. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the Focus View resource with.

Service Endpoint	North America: https://protectapi.cylance.com/foci/v2/{focus_id}/results US Government: https://protectapi.us.cylance.com/foci/v2/{focus_id}/results All Other Regions: https://protectapi-{region-code}.cylance.com/foci/v2/{focus_id}/results
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticsfocus:read scope encoded.
Request	None

Response	200 OK
	Get Focus View Results Response Schema
	{
	  “id”: “focus_id”
	  "status”: “AVAILALE”
	  “result”:{}
	}
	400 Bad Request – Malformed request.
	401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.
	403 Forbidden – The JWT token did not contain the proper scope to perform this action.
	404 Not Found – No such Focus View found.
	500 InternalServerError – An unforeseeable error has occurred.
	503 Service unavailable – Please try again later.

The response JSON schema contains the following fields:

Field Name	Description
id	The unique ID of the requested Focus View.
status	The status of the Focus View result or request. Possible values are: • AVAILABLE: A Focus View has been generated and is available for viewing.

	• DONE: Synonymous with AVAILABLE. • PENDING: The Focus View has been requested. • REQUEST: The Focus View has not been generated, but it can be requested. • RETRY_REQUEST: The Focus View has not been generated. It was previously requested but no results were received. It can be requested again. • DOES_NOT_EXIST: The Focus View requested on the device cannot be completed because the requested parameters do not exist on the device. • UNAVAILABLE: The Focus View is not available, and the associated device is not online to fulfil the request. It can be requested at a later time. UNKNOWN_DEVICE: The Focus View is not available, and the associated device is no longer known.
result	The large structure of data that is used to generate the Focus View chart and table in the UI. This field will only be populated if the *status* field is *AVAILABLE*. Parsing this data is beyond the scope of this article.

CylanceOPTICS InstaQuery

The CylanceOPTICS InstaQuery API allows users to search for system artifacts stored locally by CylanceOPTICS – files, registry key persistence points, processes, etc. Users can investigate incidents, or hunt for potential threats, and then take appropriate remediation actions.

InstaQuery searches are zone based; unzoned endpoints cannot be searched via InstaQuery.

The CylanceOPTICS InstaQuery API includes:

- Creating an InstaQuery
- Getting a list of InstaQueries in a tenant
- Getting a specific InstaQuery
- Getting the results of an InstaQuery
- Archiving an InstaQuery

Note: This addendum only covers CylanceOPTICS InstaQuery API information. Read the Cylance User API guide for configuration information.

Get InstaQueries

Allows a caller to request a page with a list of CylanceOPTICS InstaQuery resources belonging to a tenant, sorted by occurrence time, in descending order (most recent occurred InstaQuery listed first). The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to retrieve the list of InstaQueries for. The page number and page size parameters are optional, when the values are not specified, they default to 1 and 20 respectively.

Service Endpoint	North America: https://protectapi.cylance.com/instaqueries/v2?page=m&page_size=n US Government: https://protectapi.us.cylance.com/instaqueries/v2?page=m&page_size=n All Other Regions: https://protectapi-{region-code}.cylance.com/instaqueries/v2?page=m&page_size=n

Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticssurvey:list scope encoded.
Request	Append the following optional query string parameters: • q: Case-insensitive search term (e.g. name, zones, artifact). • archived: Include archived surveys. • originated-from: Limit by the relationship. • page: The page number to request. Defaults to 1. • page_size: The number of detections records to retrieve per page. Defaults to 20. • sort: Sort by field (adding "-" in front of value denotes descending order).
Response	200 OK Get InstaQueries Response Schema

```
{
  "page_number": 0,
  "page_size": 0,
  "total_pages": 0,
  "total_number_of_items": 0,
  "page_items": [
   {
     "name": "My InstaQuery Name",
     "description": "My InstaQuery Description",
     "artifact": "File",
     "match_value_type": "Path",
     "match_values": [
      "exe"
     ],
     "case_sensitive": true,
     "match_type": "Fuzzy",
     "zones": [
      "ZONE ID"
     ],
     "filters": [
      {
       "aspect": "OS",
       "value": "Windows"
      }
     ],
     "relations": [
      {
       "object": "/focus/focus_id",
       "relationship": "originated-from"
      }
     ],
     "id": "AF593F38EDC1B743BDC0A6FCC53A03CD",
     "archived": "2016-12-08T21:45:58-08:00",
     "results_available": true,
     "progress": {
      "queried": 500,
      "responded": 450
     }
   }
  ]
}
```

400 BadRequest – Returned for the following reasons:

- The tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The page number or page size specified are less than or equal to zero.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The InstaQuery resources page cannot be found.

500 InternalServerError – An unforeseeable error has occurred.

503 Server Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
page_items	The list of detections belonging to the requested page, each displaying the following information: • name: The name of the InstaQuery. • description: The description of the InstaQuery. • artifact: The Artifact type that was queried. • match_value_type: The type (or Facet) of the Artifact that was queried. • match_values: The list of values that were queried for. • case_sensitive: Whether or not the InstaQuery should take case into account. • match_type: The match type configured for the query, either “fuzzy” or “exact.” • zones: The list of Zones queried. • filters: The list of filters applied to the InstaQuery. • relations: The list of objects (e.g.: Focus Views) that the InstaQuery is related to. • id: The unique ID of the InstaQuery.

	• archived: The timestamp of when the InstaQuery was archived. • results_available: Determines if the InstaQuery has returned any results. • progress: Provides the number of devices queried and the number of devices that have responded.
page_number	The page number requested.
page_size	The page size requested.
total_number_of_items	The total number of resources.
total_pages	The total number of pages that can be retrieved based on the page size specified.

Create InstaQuery

Allows a caller to update CylanceOPTICS InstaQuery resources for a specific tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to associate the InstaQuery resource with.

Service Endpoint	North America: https://protectapi.cylance.com/instaqueries/v2 US Government: https://protectapi.us.cylance.com/instaqueries/v2 All Other Regions: https://protectapi-{region-code}.cylance.com/instaqueries/v2
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticssurvey:create scope encoded.
Request	Post InstaQuery Request Schema

```
{
  "name": "My InstaQuery Name",
  "description": "My InstaQuery Description",
  "artifact": "File",
  "match_value_type": "Path",
  "match_values": [
    "exe"
  ],
  "case_sensitive": true,
  "match_type": "Fuzzy",
  "zones": [
    "ZONE ID"
  ],
  "filters": [
    {
      "aspect": "OS",
      "value": "Windows"
    }
  ],
  "relations": [
    {
      "object": "/focus/focus_id",
      "relationship": "originated-from"
    }
  ]
}
```

Response

201 OK – Survey created

Post InstaQuery Response Schema

```
{
  "name": " My InstaQuery Name ",
  "description": "My InstaQuery Description ",
  "artifact": "File",
  "match_value_type": "Path",
  "match_values": [
    "exe"
  ],
  "case_sensitive": true,
  "match_type": "Fuzzy",
  "zones": [
    "E780CA65878745B6A4579029802D2436"
  ],
  "filters": [
    {
      "aspect": "OS",
      "value": "Windows"
    }
  ],
  "relations": [
    {
      "object": "/focus/focus_id",
      "relationship": "originated-from"
    }
  ],
  "id": "AF593F38EDC1B743BDC0A6FCC53A03CD",
  "created_at": "2016-12-08T21:45:58-08:00",
  "progress": {}
}
```

	400 Bad Request – Malformed request. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 Not Found – No such InstaQuery. 500 InternalServerError – An unforeseeable error has occurred. 503 Service unavailable – Please try again later.

The request JSON schema contains the following fields:

Field Name	Description
name	The name of the InstaQuery.
description	The description of the InstaQuery.
artifact	The type of Artifact to search. Possible values are “File”, “Process”, “NetworkConnection”, “RegistryKey”.
match_value_type	The type of value (also known as a Facet) to search. Possible values are dependent on the selected artifact type. Valid selections for each are as follows: • File ○ Path ○ MD5 ○ SHA256 ○ Owner ○ CreationDateTime • Process ○ Name ○ CommandLine ○ PrimaryImagePath ○ PrimaryImageMd5 ○ StartDateTime • NetworkConnection ○ DestAddr ○ DestPort • RegistryKey ○ ProcessName ○ ProcessPrimaryImagePath ○ ValueName ○ FilePath ○ FileMd5

	○ IsPersistencePoint
match_values	A list of strings to be matched against for the InstaQuery.
case_sensitivity	Determines whether to consider case sensitivity when matching values.
match_type	Determines whether or not to use an exact or ‘fuzzy’ match. The default behavior of InstaQuery is to use a ‘fuzzy’ match. Possible values are: • Fuzzy • Exact
zones	A list of Zone IDs to perform the InstaQuery against.
filters	A list of filters when performing the InstaQuery.
aspect	The aspect (or type) of filters. (E.g.: “OS”)
value	The value to filter for. (E.g.: “Windows”)
relations	A list of objects (E.g.: Focus View URLs) that are related to the InstaQuery. This is similar to the ‘Pivot Query’ functionality in the console.
object	The URL of the Focus View that the InstaQuery relates to.
relationship	How the InstaQuery relates to the URL. This should almost always be “originated-from”.

The response JSON schema contains the following fields:

Field Name	Description
name	The name of the InstaQuery.
description	The description of the InstaQuery.
artifact	The type of Artifact that was searched for. Possible values are “File”, “Process”, “NetworkConnection”, “RegistryKey”.
match_value_type	The type of value (also known as a Facet) that was searched for. Possible values are dependent on the selected artifact type. Valid selections for each are as follows: • File ○ Path ○ MD5 ○ SHA256 ○ Owner ○ CreationDateTime • Process ○ Name

	○ CommandLine ○ PrimaryImagePath ○ PrimaryImageMd5 ○ StartDateTime • NetworkConnection ○ DestAddr ○ DestPort • RegistryKey ○ ProcessName ○ ProcessPrimaryImagePath ○ ValueName ○ FilePath ○ FileMd5 ○ IsPersistencePoint
match_values	A list of strings to be matched against for the InstaQuery.
case_sensitivity	Determines whether to consider case sensitivity when matching values.
match_type	Determines whether or not to use an exact or 'fuzzy' match. The default behavior of InstaQuery is to use a 'fuzzy' match. Possible values are: • Fuzzy • Exact
zones	A list of Zone IDs to perform the InstaQuery against.
filters	A list of filters when performing the InstaQuery.
aspect	The aspect (or type) of filters. (E.g.: "OS")
value	The value to filter for. (E.g.: "Windows")
relations	A list of objects (E.g.: Focus View URLs) that are related to the InstaQuery. This is similar to the 'Pivot Query' functionality in the console.
object	The URL of the Focus View that the InstaQuery relates to.
relationship	How the InstaQuery relates to the URL. This should almost always be "originated-from".
id	The unique identifier of the created InstaQuery.
created_at	The Date and Time that the InstaQuery was created.
progress	The progress of the InstaQuery.

Get InstaQuery

Allows a caller to request a specific InstaQuery resource belonging to a tenant. The underlying logic extracts from the JWT the tenant's unique identifier to retrieve the detection resource for.

Service Endpoint	North America: https://protectapi.cylance.com/instaqueries/v2/{queryID}
	US Government: https://protectapi.us.cylance.com/instaqueries/v2/{queryID}
	All Other Regions: https://protectapi-{region-code}.cylance.com/instaqueries/v2/{queryID}

Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticssurvey:read scope encoded.
Request	None
Response	200 OK Get InstaQuery Response Schema

```
{
  "name": "My InstaQuery Name",
  "description": "My InstaQuery Description",
  "artifact": "File",
  "match_value_type": "Path",
  "match_values": [
    "exe"
  ],
  "case_sensitive": true,
  "match_type": "Fuzzy",
  "zones": [
    "Zone ID"
  ],
  "filters": [
    {
      "aspect": "OS",
      "value": "Windows"
    }
  ],
  "relations": [
    {
      "object": "/focus/focus_id",
      "relationship": "originated-from"
    }
  ],
  "id": "AF593F38EDC1B743BDC0A6FCC53A03CD",
  "archived": "2016-12-08T21:45:58-08:00",
  "results_available": true,
  "progress": {
    "queried": 500,
    "responded": 450
  }
}
```

400 BadRequest – Returned for the following reasons:

- The Tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The unique identifier for the detection is not valid.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The InstaQuery resource does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
name	The name of the InstaQuery.
description	The description of the InstaQuery.
artifact	The type of Artifact that was searched for. Possible values are “File”, “Process”, “NetworkConnection”, “RegistryKey”.
match_value_type	The type of value (also known as a Facet) that was searched for. Possible values are dependent on the selected artifact type. Valid selections for each are as follows: • File ○ Path ○ MD5 ○ SHA256 ○ Owner ○ CreationDateTime • Process ○ Name ○ CommandLine ○ PrimaryImagePath ○ PrimaryImageMd5 ○ StartDateTime

	• NetworkConnection ○ DestAddr ○ DestPort • RegistryKey ○ ProcessName ○ ProcessPrimaryImagePath ○ ValueName ○ FilePath ○ FileMd5 IsPersistencePoint
match_values	A list of strings to be matched against for the InstaQuery.
case_sensitivity	Determines whether to consider case sensitivity when matching values.
match_type	Determines whether or not to use an exact or 'fuzzy' match. The default behavior of InstaQuery is to use a 'fuzzy' match. Possible values are: • Fuzzy • Exact
zones	A list of Zone IDs to perform the InstaQuery against.
filters	A list of filters when performing the InstaQuery.
aspect	The aspect (or type) of filters. (E.g.: "OS")
value	The value to filter for. (E.g.: "Windows")
relations	A list of objects (E.g.: Focus View URLs) that are related to the InstaQuery. This is similar to the 'Pivot Query' functionality in the console.
object	The URL of the Focus View that the InstaQuery relates to.
relationship	How the InstaQuery relates to the URL. This should almost always be "originated-from".
id	The unique identifier of the created InstaQuery.
created_at	The Date and Time that the InstaQuery was created.
progress	The progress of the InstaQuery.
name	The name of the InstaQuery.
description	The description of the InstaQuery.
artifact	The type of Artifact that was searched for. Possible values are "File", "Process", "NetworkConnection", "RegistryKey".

match_value_type	The type of value (also known as a Facet) that was searched for. Possible values are dependent on the selected artifact type. Valid selections for each are as follows: • File o Path o MD5 o SHA256 o Owner o CreationDateTime • Process o Name o CommandLine o PrimaryImagePath o PrimaryImageMd5 o StartDateTime • NetworkConnection o DestAddr o DestPort • RegistryKey o ProcessName o ProcessPrimaryImagePath o ValueName o FilePath o FileMd5 o IsPersistencePoint
match_values	A list of strings to be matched against for the InstaQuery.
case_sensitivity	Determines whether to consider case sensitivity when matching values.
match_type	Determines whether or not to use an exact or 'fuzzy' match. The default behavior of InstaQuery is to use a 'fuzzy' match. Possible values are: • Fuzzy • Exact
zones	A list of Zone IDs to perform the InstaQuery against.
filters	A list of filters when performing the InstaQuery.
aspect	The aspect (or type) of filters. (E.g.: "OS")
value	The value to filter for. (E.g.: "Windows")
relations	A list of objects (E.g.: Focus View URLs) that are related to the InstaQuery. This is similar to the 'Pivot Query' functionality in the console.
object	The URL of the Focus View that the InstaQuery relates to.

relationship	How the InstaQuery relates to the URL. This should almost always be “originated-from”.
id	The unique identifier of the created InstaQuery.
created_at	The Date and Time that the InstaQuery was created.
progress	The progress of the InstaQuery.

Get InstaQuery Results

Allows a caller to request a CylanceOPTICS InstaQuery resource results belonging to a tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to retrieve the results of an InstaQuery for.

Service Endpoint	North America: https://protectapi.cylance.com/instaqueries/v2/{queryID}/results US Government: https://protectapi.us.cylance.com/instaqueries/v2/{queryID}/results All Other Regions: https://protectapi-{region-code}.cylance.com/instaqueries/v2/{queryID}/results
Method	HTTP/1.1 GET
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticssurvey:read scope encoded.
Request	None
Response	200 OK Get InstaQuery Results Response Schema

```
{
  "id": "AF593F38EDC1B743BDC0A6FCC53A03CD",
  "status": "done",
  "result": [
    {
      "@timestamp": 1492623352.23335,
      "HostName": "TEST_DEVICE",
      "DeviceId": "F6D1BFBDA10240E69680EF3082CA01F9",
      "@version": 1,
      "CorrelationId": "AF593F38EDC1B743BDC0A6FCC53A03CD",
      "Result": {
        "FirstObserved": "2017-04-06T21:41:16.396Z",
        "LastObservedTime": "2017-04-19T09:34:29.225Z",
        "Type": "File",
        "Uid": "GsA3xJuhMg4h5FHxLoStLg==",
        "Properties": {
          "OwnerUid": "1/HmPw5+I/8C7acx9bUtKA==",
          "SuspectedFileType": "Executable/PE",
          "Size": 27136,
          "Owner": "BUILTIN\\Administrators",
          "Path": "c:\\windows\\system32\\svchost.exe",
          "Sha256": "93B2ED4004ED5F7F3039DD7ECBD22C7E4E24B6373B4D9EF8D6E45A179B13A5E8",
          "CreationDateTime": "2009-07-13T23:31:13.886Z",
          "Md5": "C78655BC80301D76ED4FEF1C1EA40A7D"
        }
      }
    }
  ]
}
```

400 BadRequest – Returned for the following reasons:

- The tenant ID could not be retrieved from the JWT token specified in the Authorization header.
- The unique identifier for the execution is not valid.

401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid.

403 Forbidden – The JWT token did not contain the proper scope to perform this action.

404 NotFound – The InstaQuery resource does not exist.

500 InternalServerError – An unforeseeable error has occurred.

503 Service Unavailable – Unable to respond at this time, please retry later.

The response JSON schema contains the following fields:

Field Name	Description
Id	The unique ID of the InstaQuery
Status	The status of the InstaQuery.
Result	The list of responses to the InstaQuery.
@timestamp	The timestamp that the result was reported in Unix epoch time.
HostName	The hostname of the device that returned the result.
DeviceID	The unique ID of the device that returned the result.
@version	The version format of the result.
CorrelationID	The unique correlation ID of the result object.
Result	The object containing response data.

FirstObservedTime	The timestamp that the result was first observed on the system. (E.g.: when a File was first observed on the system as in a file being created.)
LastObservedTime	The timestamp that the result was last observed on the system. (E.g.: when a File was last observed as in the last time a file was interacted with.) Note: This value will be the same as the FirstObservedTimestamp for NetworkConnection and Process Artifacts.
Uid	The unique ID of the result.
Type	The type of Artifact that the result's 'properties' contain.
Properties	The object containing the individual elements of the result. This will vary depending on the Artifact and Type that was queried. • File ○ Path: The full path of the file. ○ CreationDateTime: The timestamp (in UTC) of when the file was created on the responding system. ○ Md5: The MD5 hash of the file result (where applicable.) ○ Sha256: The SHA256 hash of the file result (where applicable.) ○ Owner: The owner of the file. ○ SuspectedFileType: The suspected file type of the file object (where applicable.) ○ FileSignature: A set of information derived about the file's signature status. ○ Size: The size of the file object (in bytes.) ○ OwnerUid: The unique ID of the owner of the file. • Process ○ Name: The name of the process. ○ CommandLine: The command line arguments that the process was executed with. ○ StartDateTime: The timestamp (in UTC) of when the process was executed on the responding system. ○ PrimaryImagePath: The image file path of the process. ○ PrimaryImageMd5: The MD5 hash of the image file of the process. ○ PrimaryImageSha256: The SHA256 hash o the image file of the process. ○ PirmaryImageUid: The unique ID of the image file of the process. ○ Owner: The user who owns the process. ○ OwnerUid: The unique ID of the user who owns the process. ○ SuspectedFileType: The suspected file type of the image file of the process. ○ FileSignature: A set of information derived about the image file's signature status. ○ IsBeingDebugged: A Boolean value to determine if the process has a debugger attached to it. • Network ○ DestinationAddress: The IP address that the connection was destined to. ○ DestinationPort: The port associated with the remote IP address. ○ ProcessName: The process name that was associated with the connection.

- ProccessPrimaryImageUid: The unique ID of the process associated with the connection.
- ProcessPrimaryImagePath: The image file path of the process associated with the connection.
- ProcessImageMd5: The MD5 hash of the image file of the process associated with the connection.
- ProcessImageSha2: The SHA256 hash of the image file of the process associated with the connection.
- SuspectedFileType: The suspected file type of the image file of the process associated with the connection.

- Registry
  - IsPersistencePoint: A binary value (1 or 0) to determine if the resulting Registry item is a common persistence location.
  - ValueName: The name of the Registry Value that was interacted with.
  - Path: The full path of the Registry Key.
  - FilePath: The full path of the file referenced in the Registry Value (where applicable.)
  - FileMd5: The MD5 hash of the file referenced in the Registry Value (where applicable.)
  - FileSha256: The SHA256 hash of the file referenced in the Registry Value (where applicable.)
  - FileUid: The unique ID of the file referenced in the Registry Value (where applicable.)
  - SuspectedFileType: The suspected file type of the file referenced in the Registry Value (where applicable.)
  - FileSignature: A set of information derived about the file’s signature status that is referenced in the Registry Value (where applicable.)

Archive InstaQuery

Allows a caller to archive a CylanceOPTICS InstaQuery resource belonging to a tenant. The underlying logic extracts from the JWT specified as the Bearer value in the Authorization request header the tenant's unique identifier to archive an InstaQuery for. Surveys are archived instead of deleted so that user activity history can be maintained.

Service Endpoint	North America: https://protectapi.cylance.com/instaqueries/v2/{queryID}/archive US Government: https://protectapi.us.cylance.com/instaqueries/v2/{queryID}/archive All Other Regions: https://protectapi-{region-code}.cylance.com/instaqueries/v2/{queryID}/archive
Method	HTTP/1.1 POST
Request Headers	Accept: application/json Authorization: Bearer <JWT Token returned by Auth API> with the opticssurvey:update scope encoded.
Request	None

Response	200 OK- Survey archived. 400 BadRequest – Returned for the following reasons: • The Tenant ID cannot be retrieved from the JWT token. • The unique identifier for the detection is not valid. 401 Unauthorized – The JWT token is not specified, has expired, or is otherwise invalid. 403 Forbidden – The JWT token did not contain the proper scope to perform this action. 404 NotFound – The InstaQuery resource does not exist. 500 InternalServerError – An unforeseeable error has occurred. 503 Service Unavailable – Unable to respond at this time, please retry later.