flexera™

Software Vulnerability Research
API User Guide

Legal Information

Book Name: Software Vulnerability Research API User Guide

Part Number: SVR-JUNE2024-API00

Product Release Date: June 2024

Copyright Notice

Intellectual Property

Restricted Rights Legend

Contents

Contents

1

Software Vulnerability Research API Help Library

This Software Vulnerability Manager API User Guide provides the API information for Flexera's Software Vulnerability Research

Table 1-1 • Software Vulnerability Research API Help Library

Topic	Content
API Introduction	This section describes how to access the API information.
Vulnerability Manager Module API Information	This section provides Vulnerability Manager module API information. *Note • The Vulnerability Manager module is not available for Software Vulnerability Research - Assessment Only.*
Research Module API Information	This section provides Research module API information. *Note • The Research module is not available for Software Vulnerability Research - Assessment Only.*
Assessment Module API Information	This section provides Assessment module API information. *Note • The Assessment module is not available for Software Vulnerability Research.*
Patching Module API Information	This section provides Patching module API information. *Note • The Patching module is not available for Software Vulnerability Research.*

Table 1-1 • Software Vulnerability Research API Help Library (cont.)

Topic	Content
Settings Module API Information	This section provides Settings module API information.
Appendix A - HTTP Status Codes	This section provides HTTP Status Codes.

Product Support Resources

The following resources are available to assist you with using this product:

- Flexera Product Documentation

- Flexera Community

- Flexera Learning Center

- Flexera Support

Flexera Product Documentation

You can find documentation for all Flexera products on the Flexera Product Documentation site:

https://docs.flexera.com

Flexera Community

On the Flexera Community site, you can quickly find answers to your questions by searching content from other customers, product experts, and thought leaders. You can also post questions on discussion forums for experts to answer. For each of Flexera's product solutions, you can access forums, blog posts, and knowledge base articles.

https://community.flexera.com

Flexera Learning Center

Flexera offers a variety of training courses—both instructor-led and online—to help you understand how to quickly get the most out of your Flexera products. The Flexera Learning Center offers free, self-guided, online training classes. You can also choose to participate in structured classroom training delivered as public classes. You can find a complete list of both online content and public instructor-led training in the Learning Center.

https://learn.flexera.com

Flexera Support

For customers who have purchased a maintenance contract for their product(s), you can submit a support case or check the status of an existing case by making selections on the **Get Support** menu of the Flexera Community.

https://community.flexera.com

Product Feedback

You can submit feedback about Software Vulnerability Manager in the Flexera Customer Community Forum. You can also submit feedback through the Software Vulnerability Manager user interface by clicking the feedback icon in the upper-right-hand corner of each module.
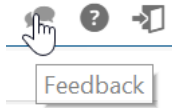
Contact Us

Flexera is headquartered in Itasca, Illinois, and has offices worldwide. To contact us or to learn more about our products, visit our website at:

http://www.flexera.com

You can also follow us on social media:

- Twitter

- Facebook

- LinkedIn

- YouTube

- Instagram

2

API Introduction

This section provides an overview of the following API topics:

- API Explorer

- API Menu Options

- Tokens

- Examples - Calling the API

- Using Windows PowerShell

- API Notes

- API Access - Roles

- XML Feeds

- External API Services (Service Providers)

- API Application Account Lockout

API Explorer

You can explore the API endpoint using a browsable interface at https://api.app.secunia.com/api/ that you can login to using the same credentials used to authenticate your account. The interface is a fully functional API client and any operations performed through the browser will be reflected in the Application.
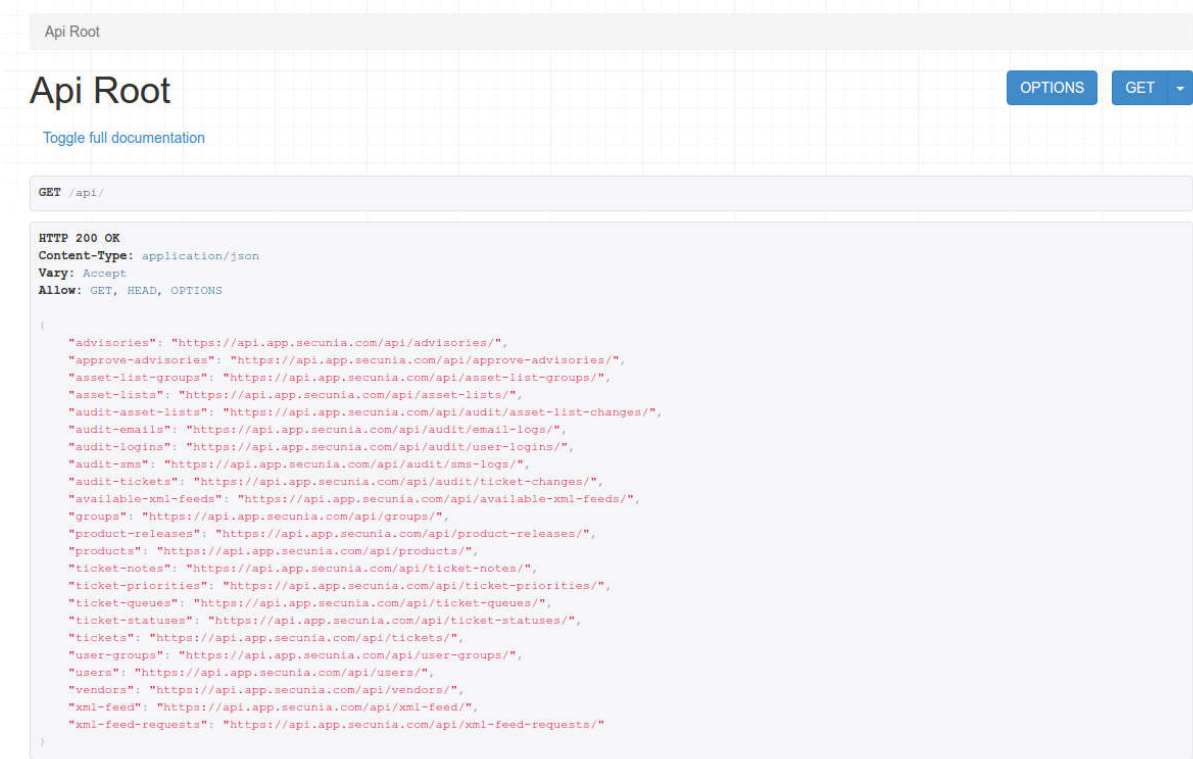
Figure 2-1: API Explorer Page Example

Click Toggle full documentation to access the documentation for each endpoint.

API Menu Options

Use the **Settings > API** pages to work with the Tokens, XML Feeds, and External API Services (Service Providers) associated with your account.

You can use the Token management handling system when accessing the built-in API to add an extra security layer when utilizing the API.
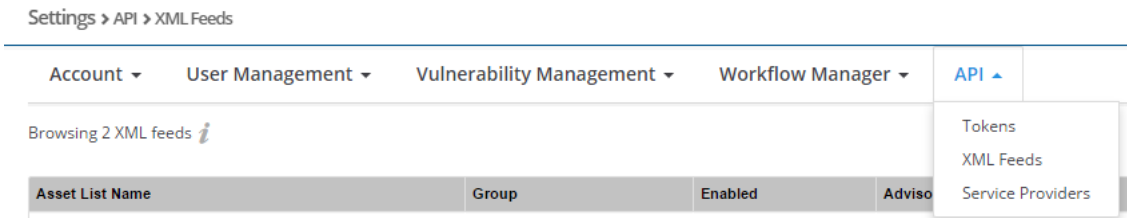
An authenticated and license restricted access HTTP API is provided and follows the REST pattern using the JSON format. Access to the different resources (Watch Lists, Advisories, and so on) is made through specific endpoints, for example https://app.flexerasoftware.com/api/asset-lists/. For further details, see Settings Module API Information.

The HTTP verbs used are as follows:

Table 2-1 • HTTP Verbs

HTTP Verb	Description
GET	For read.
POST	For create.
PATCH / PUT	For update.
DELETE	For delete token.

Tokens

The **Settings > API > Tokens** page displays the user name, Token ID and creation date for all API Access Tokens that have been generated. Every scripted API call requires authorization using an API Token. Every user has a pre-generated token.

For developer convenience, the API is also accessible with cookie based authentication, made available to present the API root and documentation. However, it is forbidden to code API calls using cookie based user and password authentication and Token based authentication is required in this case (each request will also be processed faster this way).

Task   ***Working with Tokens:***

1. When you open the Tokens page, the Token is truncated.

2. To expand the Token, click the ellipsis.

Figure 2-2: Truncated Token

Figure 2-3: Expanded Token

3. Click a Token in the grid to delete the Token.
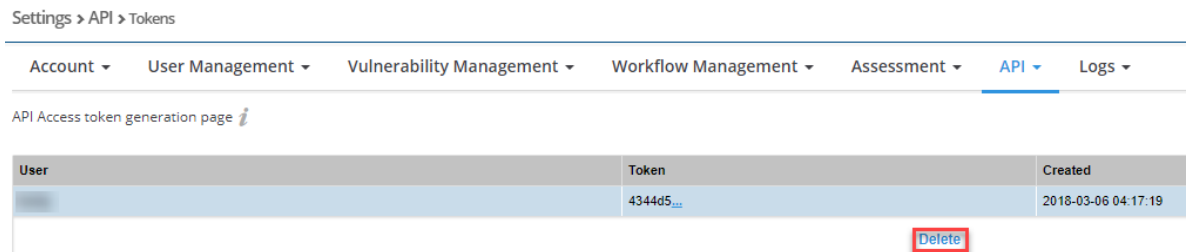
Figure 2-4: Delete a Token

4.   Click ➕ to add a new token.

The token must be specified using the HTTP "Authorization" header. For example:

```
Authorization: Token 8f82bd5574a425bdf867b243917a24d16fbf0079
```

A full example using the "curl" program is shown below:

```
curl -H "Authorization: Token 8f82bd5574a425bdf867b243917a24d16fbf0079" -H "Content-Type: application/
json" https://api.app.secunia.com/api/xml-feed/?feed_type=asset_list&asset_list_id=4&days=1
```

This example will get you the last 24 hours advisory information for Watch list 4. You can find all possible combinations on the XML Feeds settings page.

```
curl -H "Authorization: Token 8f82bd5574a425bdf867b243917a24d16fbf0079" -H "Content-Type: application/
json" https://api.app.secunia.com/api/tickets/
```

However, Flexera recommends calling full collection GET only once and then doing differences where the API allows. Please refer to for further information.

Note • You must use the authorization token for requests made programatically.

While browsing the interface, the request works because cookie based authentication has been enabled for developer convenience. However, the usage of cookie based authentication for your own scripts is forbidden. Please use token based authentication instead.

Examples - Calling the API

Important • All of the examples given below are implemented using curl. These are examples only and have been used for live-testing while coding the API. You will use your own development language to query the API over HTTPS.

- Getting a List of First 20 Advisories

- Getting a Custom List of Advisories

- Getting a Specific Advisory by Integer ID

Getting a List of First 20 Advisories

Use the following code to get a list of the first 20 advisories (first page):

```
curl -H "Authorization: Token REPLACE_WITH_YOUR_TOKEN" -H "Content-Type: application/json" https://
api.app.secunia.com/api/advisories/
```

You can use the "`count: n`" result to know the exact size of your results and then use queries such as `/api/advisories/?page=2&page_size=10` to paginate the results.

> **Note •** *The maximum page size supported is 100 and you cannot get all of the endpoint results in one massive request (which would also not be recommended for performance reasons). To get all the results you will need to script several requests over the total count of results. Please refer to API Throttling for further information.*

Getting a Custom List of Advisories

Use the following code to get a custom list of advisories:

```
curl -H "Authorization: Token REPLACE_WITH_YOUR_TOKEN" -H "Content-Type: application/json" https://
api.app.secunia.com/api/advisories/
?released__gte=1435698000&released__lt=1438376400&criticality=1&criticality=2
```

In this example, the advisory set has been restricted to the released date being greater than or equal (`gte`) to a Unix based date and less than (`lt`) another date, and filtered based on the criticality levels (1 and 2 in this example).

Getting a Specific Advisory by Integer ID

Use the following code to get a specific advisory by integer id (not guaranteed to be consecutive):

```
curl -H "Authorization: Token REPLACE_WITH_YOUR_TOKEN" -H "Content-Type: application/json" https://
api.app.secunia.com/api/advisories/175867/
```

... or, by a unique Secunia Identifier:

```
curl -H "Authorization: Token REPLACE_WITH_YOUR_TOKEN" -H "Content-Type: application/json" https://
api.app.secunia.com/api/advisories/SA69295/
```

This example queries only for a specific advisory based on its "id" (or its unique identifier - SAID) taken from the list of advisories on a previous JSON result.

> **Note •** *The content of an individual response is different than the list offered on the root of the endpoint as there is more information available on an individual level.*

You can also make `POST` requests for the endpoints that support it (you have a request builder on the browsable interface). For example, you can use `POST` on the `/api/tickets/` endpoint to create or update new tickets.

All endpoints have documentation text built-in on each page that you can view by clicking Toggle full documentation, where you can find all the filters and parameters you can use to build your queries.

You can also find this information under the appropriate section in this API User Guide:

- Research Module API Information

- Assessment Module API Information

- Patching Module API Information

- Settings Module API Information

Using Windows PowerShell

Note • The API PowerShell example shown below requires Windows PowerShell version 4.0 or greater. Windows PowerShell 4.0 is bundled with Windows 8.1 or newer Windows operating systems or the Windows 7 operating system with the Windows Management Framework 4.0 installed.

The following PowerShell command can be used to determine which version of Windows PowerShell you are using:

```
$PSVersionTable.PSVersion
```

The following example was created using Windows PowerShell version 5.0:

```
$url = "https://api.app.secunia.com/api/advisories/"
$headers = @{}
$headers.Add("Authorization","Token REPLACE_WITH_YOUR_TOKEN")
$headers.Add("Content-Type","application/json")
Invoke-RestMethod -Method GET -Uri $url -Headers $headers -Verbose -Debug
```

API Notes

The following sections provide additional API information:

- API Versions and Parsing

- API Throttling

- CVSSv4 Score

- CVSSv3 Score

- CPE v2.3

API Versions and Parsing

Periodically, Flexera will make changes to the existing APIs. All of the latest changes will be made available on the path:

```
~/api/
```

If you don't want to risk any breaking changes affecting your scripts, Flexera recommends that you hardcode the API version in the coded requests, for example, all requests to go to:

```
~api/v1/
```

To avoid any breaking changes introduced to the API, Flexera will offer all future changes as a new version (v2, v3, v4 and so on), while keeping the old functionality working for at least one year from the moment a new version is released.

As a rule of thumb, Flexera will NOT change the API version for small fixes where more data is added to existing calls, and it is strongly recommend that you code your JSON parsing in such way that it doesn't expect exactly the same tags in the same order and at the same number of characters from key/tag X; use a good parsing library instead that offers dictionaries/lists for data querying.

Flexera strongly discourages any usage of pseudo-code similar to `foo=j.substring(j.indexof("Foo:"), 5)` or any similar variations of non-true JSON parsing (such as crude guess-reads) as these are error prone and will likely fail in the future.

The same recommendation applies for XML Feeds, where XML parsing is recommended as opposed to string matching over the full document (for example using regexes or any guess patterns).

Important • Flexera accepts no responsibility for any breaking changes introduced by using bad coding practices over the scripts you write.

API Throttling

API uses throttling based on burst, sustained and scoped policies.

- Burst policies restrict more than 250 calls per minute for paid accounts and more than 60 calls per minute for trial accounts.

- Sustained policies are not restricted for paid accounts and restrict more than 1000 calls per day for trial accounts.

- Scoped policies are not restricted for paid accounts and restrict downloading more than 30 advisories per day for trial accounts. However, tickets information or other non-proprietary information is not affected.

Note • Please use timeouts between requests to meet the above restrictions, otherwise the Flexera infrastructure might interpret your attempts as malicious activity and throttle down/reject your calls. Also, you should ensure that you query only for differences (use modified/released/created fields along with `__gte` or `__lt` modifiers) so you don't need to re-query for the entire set of data each day.

When you have reached the thresholds of calls, you will receive the status HTTP_429_TOO_MANY_REQUESTS and a message informing you when (in seconds) your request will be let through.

```
HTTP 429 Too Many Requests
Content-Type: application/json
Retry-After: 35
Allow: GET, HEAD, OPTIONS
Vary: Accept

{
    "detail": "Request was throttled. Expected available in 35 seconds."
}
```

Figure 2-5: HTTP_429_TOO_MANY_REQUESTS message

CVSSv4 Score

SVR will now support entering all new CVSS scores using the v4 standard. After a CVSS v4 score is entered, the score appears in the User Interface (UI), API, XML, email notifications, and PDF reports.

For details, see CVSSv4 Score.

CVSSv3 Score

On May 18, 2018 Flexera's Secunia Research began entering all new CVSS scores using the v3 standard. After a CVSSv3 score is entered, the score appears in the User Interface (UI), API, XML, email notifications, and PDF reports. For details, see CVSSv3 Score.

CPE v2.3

Effective December 15, 2023, NVD will retire the CPE v2.2 legacy feed. Consequently, current products in SVR will be visible in both CPE v2.2 and CPE v2.3 naming formats during this transitional period. However, it's important to note that any new products added to SVR in the future will be displayed in the CPE v2.3 naming format, and the CPE v2.2 format only if available.

API	Description
/cpe23-products/	This API will return the entire list of CPEs v2.3 and when available, its associated product in SVR.
/advisories/	This API now returns a new property named cpes23. This new property contains the products in the CPE v2.3 naming format that are affected by the selected advisory.
/product-releases/	This API now returns a new property named cpes23. For the selected product, when available, the cpes23 property contains the CPE v2.3 naming format of the product.

API Access - Roles

This section describes the API-supported endpoint actions for Roles:

- Available API Access for Roles

- No API Access for Roles

Available API Access for Roles

The following are available API Access for Roles.

Table 2-2 • Available API Access

Roles	API
API Advisories search	/advisories/
API Limited advisories search	/advisories/
API Product database access	/cpe-products/
	/cpe23-products/
	/products/
	/product-fullversion/
	/product-releases/
	/vendors/
API Ticket management	/audit/ticket-changes/
	/ticket-notes/
	/ticket-priorities/
	/ticket-queues/
	/ticket-statuses/
	/tickets/
API User management	/groups/
	/user-groups/
	/users/
API Watch List management	/approve-advisories/
	/asset-list-groups/
	/asset-lists/
	/asset-lists-subscriptions/
	/audit/asset-list-changes/
	/product-fullversion/
	/product-releases/
	/products/
	/vendors/
Super Administrator	Accessible to all APIs

No API Access for Roles

The following Roles do not have access to the API.

Table 2-3 • No API Access

Roles	API
Advisory Manager	No access to API
Advisory PDF attachments	No access to API
Advisory Reader	No access to API
Analytics Manager	No access to API
Analytics Reader	No access to API
API developer	No access to API
Auditor	No access to API
Deny auto-approval	No access to API
Device Database Reader	No access to API
Historical Data Assessment	No access to API
Patch Manager	No access to API
Policy Manager	No access to API
Policy Reader	No access to API
Product Database Reader	No access to API
Receive all advisories	No access to API
Rejected Advisories	No access to API
Scan Manager	No access to API
SCIM	No access to API
Ticket Manager	No access to API
Ticket Reader	No access to API

Table 2-3 • No API Access

Roles	API
Ticket User	No access to API
User Manager	No access to API
Watch List Manager	No access to API
Watch List Manager Local	No access to API
Watch List Reader	No access to API
Watch List Subscription Manager Local	No access to API
Watch List Subscription Manager	No access to API
XML Emails	No access to API
XML Feeds	No access to API

XML Feeds

The **Settings > API > Tokens** page displays the available XML intelligence feeds based on your configured Watch Lists.

The Dynamic feeds show new feeds only, for example anything new since the last time you viewed the feeds. The time-specific feeds display advisories from the last 24, 48 and 72 hours.

Click on **Watch List Name** to view the Watch List.

Figure 2-6: XML Feeds Page

Note • *The feeds do not include advisories released before the time the Watch List was created.*

External API Services (Service Providers)

You have the option to call external API services when certain actions occur.

📄

Note • *The supported external services are ServiceNow and BMC Remedy. Other generic APIs can be called. However, the integration has not been tested by Flexera.*

The recommended scenarios are to call the API when:

● A new advisory is released for an Watch List

● An advisory is updated for an Watch List

● A ticket is created

See Integration with the External API Service Provider to call external API services.

Integration with the External API Service Provider

📋

Task   **To select and configure the integration:**

1. Define the external API to be called, named from now on a "service provider". Go to Settings > API > Service providers.

2. Click + and choose to create a predefined recipe for ServiceNow or BMC Remedy or create your own external API.

3. Change the API endpoint and authentication credentials. The other options are automatically configured.

After selecting and configuring the service provider, set up the following with the service provider:

● Service Provider Fields

● Service Provider Methods

● Service Provider Test Connection

● Create Rules to Call the Service Provider

Service Provider Fields

The service provider contains the following fields:

Table 2-4 • Service Provider Fields

Field	Description
type	One of the following values: ● **Custom**—Custom API defined by the customer. ● **ServiceNow**—ServiceNow specific calls, a REST recipe is offered. ● **BMC Remedy**—BMC Remedy calls, a SOAP recipe is offered, with basic authentication. 📄 **Note •** *The **Custom** value is not tested by Flexera.*

Chapter 2   API Introduction
External API Services (Service Providers)

Table 2-4 • Service Provider Fields

Field	Description
name	Identifies the service providers in selection forms.
url	The public accessible API endpoint, the root endpoint. The final URL is constructed based on the root url, plus the partial one from the method.
protocol type	One of the following values: • REST • SOAP
authentication type	One of the following values: • **None**—The authentication details will be set otherwise, for example in the headers for token based authentication, or in the request body for SOAP Basic access • **Basic authentication**—The authentication details will be set in the default authentication header for REST or, for BMC Remedy, in the custom soap header.
headers	(Optional) Any custom headers that need to be sent, for example, the authentication through an accessible token.

Service Provider Methods

The service provider has methods, the actual endpoints that will be created. For the newly created service provider, you need to create the methods that will be called. A method is identified by the following:

Table 2-5 • Service Provider Methods

Method	Description
service provider	The service provider it belongs to.
name	Identifies the endpoint in the selection forms.
url	Partial url, that will be appended to the public API.
method	The method that will be called: • **For REST protocol**, the method is one of the HTTP method calls: GET, POST, PUT etc. • **For SOAP protocol**, the method represents the SOAP method called on the service
headers	(Optional) Any custom headers that need to be sent with the request.
query params	(Optional) Any custom query strings that need to added to the URL.

Table 2-5 • Service Provider Methods

Method	Description
content	The data part of the request: • **For REST protocol**, the content must be a JSON object with the entire content • **For SOAP protocol**, the content may be a JSON object or the entire XML body. The JSON object is used to dynamically construct the request. It's an easier way to enter the values for the request than the raw XML.
retrieve entity id description	After each call the system makes, it will try to extract the unique identifier for the external object that was created/updated, to be able to make change requests on the same object when the corresponding entity changes in Software Vulnerability Research. For instance, if an incident is created in ServiceNow when an advisory is released, the system is able to update the same incident if the advisory is updated. The expression under "retrieve entity id" is used to extract the object id from the response.

The available options for Service Method and BMC Remedy each creates three methods: create, get and update entities. The methods can be customized to send more information in the existing fields and/or other fields.

The content and urls contain placeholders that are replaced before the request with the appropriate information. The placeholders are marked by the characters #$. The information that can be used in the placeholders is related to advisories, tickets, and the referenced object id. On the service methods page you can get full examples of the information available. Some examples are:

Table 2-6 • Placeholder Examples

Placeholders	Description
`#$advisory.advisory_identifier#$`	The unique advisory identifier released by Secunia.
`#$advisory.title#$`	The advisory title.
`#$advisory.products.name#$`	Affected products.
`#$asset_list.name#$`	Watch List name for which the advisory was released.

Figure 2-7: ServiceNow Service Provider Example

Figure 2-8: ServiceNow Methods Example

Edit method Incident create - HelpDesk_Submit_Service   ✕

Service Provider

BMC Remedy

Name

Incident create

Url

HPD_IncidentInterface_Create_WS

Method

HelpDesk_Submit_Service

Header	Value	Action
Header	Value	ADD +

Query Param	Value	Action
Query Param	Value	ADD +

Content

{'Action': 'CREATE', 'Status': 'New', 'Summary': 'Advisory #$advisory.advisory_identifier#$ for asset list #$asset_list.name#$ was released. Ticket #$ticket.pretty_id#$. `#$advisory.title#$` affects products: #$advisory.products.name#$', 'Service_Type': 'User Service Request', 'Impact': '4-Minor/Localized', 'Reported_Source': 'Other', 'Last_Name': 'Allbrook', 'Urgency': '4-Low', 'First_Name': 'Allen'}

Retrieve entity id expression

result.Incident_Number

Cancel   Save

Figure 2-9: Create BMC Remedy Example:

Method	Url	Retrieve entity id expressio
HelpDesk_Submit_Service	HPD_IncidentInterface_Create_WS	result.Incident_Number
HelpDesk_QueryList_Service	HPD_IncidentInterface_WS	result.Incident_Number
HelpDesk_Modify_Service	HPD_IncidentInterface_WS	result.Incident_Number

Figure 2-10: BMC Ready Methods Example

Service Provider Test Connection

After you create the methods for the service providers, it is advisable to test the connection. The test option exists on each method. The system performs a call with the shown parameters and returns the response from the external API. For example, if a create call is successful a new entity will be created in the external system.

All service calls and the response from them are recorded under **Auditor > Service Calls**.

Figure 2-11: Test ServiceNow Create Incident Call Example

Create Rules to Call the Service Provider

After the service providers and methods are correctly configured, you can create rules to tell the system when to call the external API method.

Under **Settings > Workflow Manager > Rules** you can add the action **Call service provider** to existing rules or create a new rule according to your requirements.
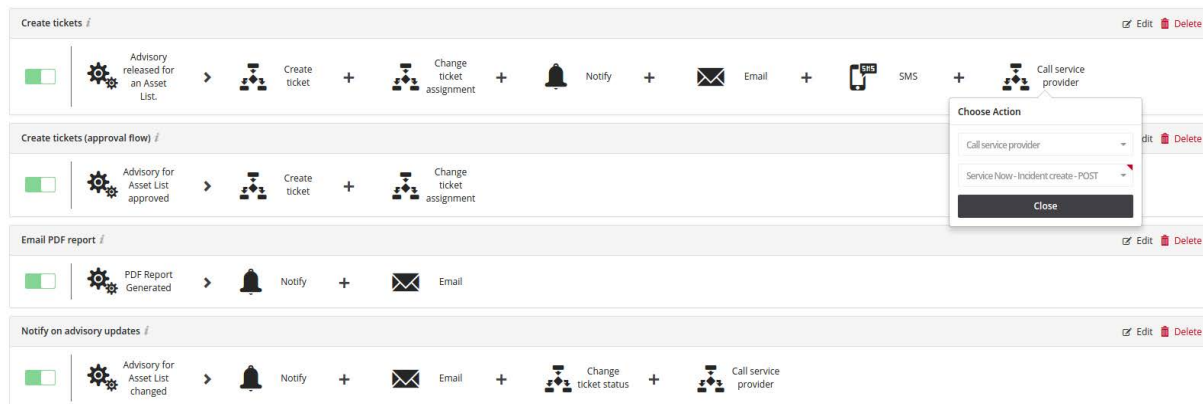
Figure 2-12: Rules with Tickets with "Call service provider" Example

Note • *The system checks if, on the request, all placeholders in the content and/or url can be replaced. The system knows of the following placeholders: advisory, Watch list, ticket, ref_object_id, when each entity makes sense. If the trigger is a generic trigger "advisory released for an Watch list", means that the system knows about the "advisory" and "Watch list", but no ticket yet exists. The ticket will be present after the action "create ticket".*

It is assumed there are at least the standard methods for create/get/update for the external object, for example "incident":

● For a rule "Create tickets", at the end, add "Call service provider", select the "Incident create" method and save. The "Incident create" will be called for each new advisory released on all Watch Lists.

● For a rule "Notify on advisory updates", at the end, add "Call service provider", select "Incident update" method and save. You can also choose to create a new incident for updates.

Task   **To disable the creation of tickets:**

1. Disable existing rules.

2. Create two new rules:

● **Advisory released**: Trigger "Advisory released for an Watch List", for any Watch List, action "Call service provider" with the create method, optional email and notification action.

● **Advisory updated**: Trigger "Advisory for Watch List changed", for any Watch List, action "Call service provider" with the update or create method.

Note • *For this example, no mention to the ticket should exist in the content for the method, or the system will not allow the save of the rule.*
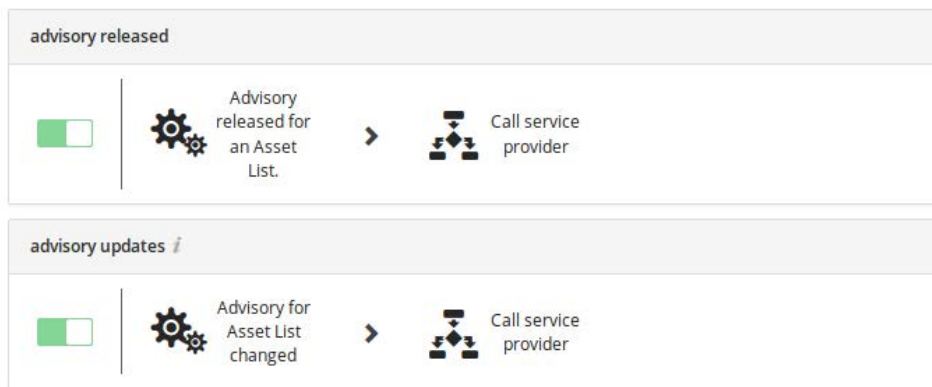
Figure 2-13: Rules without Tickets Example

API Application Account Lockout

An account lockout policy has been implement to prevent multiple failed login attempts. Five failed login attempts for accessing SVR APIs via browser will block the account. Only an Administrator has the rights to unblock such accounts.

3

Vulnerability Manager Module API Information

Edition • *The Vulnerability Manager module is not available for Software Vulnerability Research - Assessment Only.*

Important • *The following information has been taken from the individual links in the API Root screen and becomes available when you press* ***Toggle full documentation****. The information can become obsolete and you should* ***always*** *check the API information inside the portal.*

The links to the various portals displayed in your API Root screen are the ones you have access to based on your subscription and user groups. These may not match the links given below.

This section includes the following API information for the Assessment module.

- Watch List Advisory List
- Watch List Group List
- Watch List List
- Watch List Changes
- PowerShell Script to Download Watch Lists to a CSV File

Watch List Advisory List

For information on the Watch List Advisory List, see the following URL:

https://api.app.secunia.com/api/approve-advisories/

A list of advisories per Watch List awaiting approval before tickets are created.

When creating an Watch List, if you enable the option "Advisories need approval", the Watch List creator will be notified when an advisory is released and needs approval. If the advisory is approved, a ticket is then created if the advisory criticality is greater than the ticket threshold criticality and emails/SMS are sent if the threshold conditions apply.

If the advisory is dismissed, it disappears from the initial list. You can delete the dismissed advisories and you can permanently delete or approve them in case the dismissal was done by mistake.

API Supported Endpoint Actions and Available Methods for Watch List Advisory List APIs include:

- Available Methods for Watch List Advisory List

- Available Filters on Watch List Advisory List

- Approve Method for Watch List Advisory List

- Dismiss Method for Watch List Advisory List

Available Methods for Watch List Advisory List

The following methods are available for the Watch List Advisory List:

Table 3-1 • Methods for Watch List Advisory List

Method	Example
get list	GET <URL>
approve instance	POST <URL><id>/approve/
dismiss instance	POST <URL><id>/dismiss/
delete instance	DELETE <URL><id>/

Available Filters on Watch List Advisory List

The following filters are available for the Watch List Advisory List:

Table 3-2 • Filters on Watch List Advisory List

Filter	Description
asset_list_id (int)	The Watch list id for which the advisory matches
identifier (string)	Unique advisory identifier
title (string)	Case insensitive term search in the advisory title
criticality (int)	Advisories with a certain criticality. (See criticality filter options on advisories page.)
solution_status (int)	Advisories with a certain solution status. (See solution status filter options on advisories page.)
released__gte (int)	Unix timestamp for the release date of the advisory, filter type greater than or equal (seconds)
released__lt (int)	Unix timestamp for the release date of the advisory, filter type less than (seconds)

Table 3-2 • Filters on Watch List Advisory List

Filter	Description
`dismissed (bool)`	Filters advisories that were previously dismissed and can now be permanently deleted or approved.

Approve Method for Watch List Advisory List

Approved advisories for an Watch list. Then, if the threshold conditions pass, a ticket is created and notifications are sent.

Dismiss Method for Watch List Advisory List

The advisory is dismissed and removed from the list.

Watch List Group List

For information on the Watch List Group List, see the following URL:

https://api.app.secunia.com/api/asset-list-groups/

Watch List Groups are used to visually group together Watch Lists, for example "Windows" Watch Lists, "QA Products Watch List" and so on.

API Supported Endpoint Actions and Available Methods for Watch List Group List APIs include:

- Available Methods for Watch List Group List
- Available Filters on Watch List Group List
- Watch List Group List Fields for Create/Edit

Available Methods for Watch List Group List

The following methods are available for the Watch List Group List.

Table 3-3 • Methods for Watch List Group List

Method	Example
get list	`GET <URL>`
get instance details	`GET <URL><id>/`
create instance	`POST <URL>`
edit instance	`PUT <URL><id>/`
delete instance	`DELETE <URL><id>/`

Available Filters on Watch List Group List

The following filters are available for the Watch List Group List

Table 3-4 • Filters on Watch List Group List

Filter	Description
`name (string)`	Invariant case search by term in name.
`created_gte`	Unix timestamp for the asset list group create date, filter type greater than or equal (seconds).
`created_lt`	Unix timestamp for the asset list group create date, filter type less than (seconds).

Watch List Group List Fields for Create/Edit

The following are Watch List Group List fields for Create/Edit.

Table 3-5 • Watch List Group List Fields for Create/Edit

Filter	Description
`name (string)`	The group name visible in the interface

Watch List List

For information on the Watch List List, see the following URL:

https://api.app.secunia.com/api/asset-lists/

Watch Lists represent a combination of vendors, products and product versions that you want to track advisories for. Disabled Watch Lists are not taken into consideration by the rule system.

API Supported Endpoint Actions and Available Methods for Watch List List APIs include:

- Available Methods for Watch List List

- Available Filters on Watch List List

- Watch List List Fields for Create/Edit

- Watch List List Threshold Choices

Available Methods for Watch List List

The following methods are available for the Watch List List.

Table 3-6 • Methods for Watch List List

Method	Description
get list	`GET <URL>`

Table 3-6 • Methods for Watch List List

Method	Description
get instance details	`GET <URL><id>/`
create instance	`POST <URL>`
edit instance	`PUT <URL><id>/`
delete instance	`DELETE <URL><id>/`
vendors	Gets the paginated list of vendors for an watch list: `GET <URL><id>/vendors/`
products	Gets the paginated list of products for an watch list: `GET <URL><id>/products/`
product-releases	Gets the paginated list of product releases/versions for an watch list: `GET <URL><id>/product-releases/`

Available Filters on Watch List List

The following filters are available for the Watch List List:

Table 3-7 • Filters on Watch List List

Filter	Description
`name (string)`	Invariant case search by term in name.
`group__name (string)`	Invariant case search by term in name.
`group_id (int)`	Exact search for watch lists in group.
`enabled (bool)`	Searched for enabled /disabled Watch lists.
`created_by_id (int)`	Owner.
`created_gte`	Unix timestamp for the watch list create date, filter type greater than or equal (seconds).
`created_lt`	Unix timestamp for the watch list create date, filter type less than (seconds).

Watch List List Fields for Create/Edit

The following are Watch List List List fields for Create/Edit:

Table 3-8 • Watch List List List Fields for Create/Edit

File	Description
name (string)	The Watch list name visible in the interface
group (id)	The group id in which the Watch list should be included
group_name (string)	The group name if the group does not exist; the group will be created and the Watch list will be assigned to that group
advisories_need_approval (bool)	Means that the matched advisories for the Watch list generate only some alerts for the user. If those advisories are approved, they transform into tickets. Otherwise, they are dismissed by the system. This gives you an extra method to filter only advisories relevant to your organizational needs.
enabled (bool)	If the Watch list is disabled, new advisories released will not be matched against it
vendors (list of int)	Vendor ids list that you want to track, the ids can be taken from the vendors api
products (list of int)	Products ids list that you want to track, the ids can be taken from the products api
product_releases (list of int)	Product specific versions ids list that you want tracked, the ids can be taken from the product versions api
ticket_notification_threshold (int)	Used in generating tickets / alerts for approval. If an advisory has the criticality below this threshold, the advisory is dismissed for the Watch list and no notifications are generated (notification, emails, sms). **Note •** *See Watch List List Threshold Choices for integer choices.*
notification_level_email (int)	Used for sending emails. If the ticket is generated, you will be notified only if the advisory criticality level is over the "notification_level_email". **Note •** *See Watch List List Threshold Choices for integer choices.*

Table 3-8 • Watch List List List Fields for Create/Edit

File	Description
`notification_level_sms (int)`	Used for sending sms when an advisory is released that matches your Watch list, the ticket was created and the advisory criticality is over this threshold. We highly recommend a value of "Extremely critical" for this value. 📄 ***Note •** See Watch List List Threshold Choices for integer choices.*

Watch List List Threshold Choices

The following threshold choices are available.

Table 3-9 • Watch List List Threshold Choices

Integer	Description
`0`	None (not available for `ticket_notification_threshold`)
`1`	Extremely critical
`2`	Highly critical and above
`3`	Moderately critical and above
`4`	Less critical and above
`5`	Not critical and above
`"custom_cr" (string)`	Confidentiality Requirement 📄 ***Note •** See Custom Requirements.*
`"custom_ir" (string)`	Integrity Requirement 📄 ***Note •** See Custom Requirements.*
`"custom_ar" (string)`	Availability Requirement 📄 ***Note •** See Custom Requirements.*

Custom Requirements

The custom requirements are used to override the environmental metrics of the CVSS vector for the advisories. They may have one of the following values or be left undefined:

- **ND**—Not defined

- **L**—Low

- **M**—Medium

- **H**—High

If you choose to set these values, the CVSS vector and Score for the advisories that match the Watch list will take into consideration the defined values.

Watch List Changes

For information on the Watch List Changes, see the following URL:

https://api.app.secunia.com/api/audit/asset-list-changes/

Available Filters for Watch List Changes

The following are the available filters for Watch List changes.

Table 3-10 • Filters for Watch List Changes

Filter	Description
start (int)	Unix timestamp for the start date
end (int)	Unix timestamp for the start date
asc (bool)	Sorting order, ascending (True) or descending (False)
page_size (int)	Page size.
ref (guid)	"Next" value from a paginated response
object_id (int)	The Watch list id for which the changes were made

PowerShell Script to Download Watch Lists to a CSV File

Below is a sample PowerShell script to download watch lists to a CSV file:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$OutputFile = 'c:\code\script\My.CSV'
$WebServiceHeader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$WebServiceHeader.Add("Content-Type", 'application/json')
```

```
$WebServiceHeader.Add("Authorization", "Token YOUR_TOKEN_HERE" )
$URL = "https://api.app.secunia.com/api/asset-lists/export-assets/
?asset_list=32&asset_list=1&asset_list=33&asset_list=34&asset_list=35&asset_list=36&asset_list=37&asset
_list=38&asset_list=39&asset_list=40&asset_list=41&asset_type=product_release&format=json&export=csv&fi
lename=export_20171010_152115"
(Invoke-RestMethod ($URL) -Method Get -Headers $WebServiceHeader) | Out-File $OutputFile
```

4

Research Module API Information

Edition • *The Research module is not available for Software Vulnerability Research - Assessment Only.*

This section includes the API information involved with the Research module. For details, see:

- PowerShell Script to Pull Advisory Information
- PowerShell Script to List All Devices and Their System Scores
- PowerShell Script to Save All Advisories Within a Date Range to CSV
- PowerShell Script to Query Historic Advisories by Product and Version

PowerShell Script to Pull Advisory Information

Below is a sample PowerShell script to pull advisory information:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
#Max number of advistories to pull
$global:QueryLimit = 20
function QueryData ($URL, $Header)
{
    # Get First Page of results (20 items)
    $result = @()
    $results = @()
    try
    {
        $result = Invoke-RestMethod ($URL) -Method Get -Headers $Header
        $results = $result.results
        if ($result.results)
        {
            $results = $result.results
        }
        else
        {
```

```
                    $results = $result
            }
        }
        catch
        {
            Write-host ("Error QueryData1 " + $URL + " " +  $_.Exception.Message + " " +
$_.Exception.ItemName) -ForegroundColor Red
        }
        #Get the next pages of results, if any
        while (![string]::IsNullOrWhiteSpace($result.next))
        {
            try
            {
                $result = Invoke-RestMethod $result.next -Method Get -Headers $Header
                $results += $result.results
                if ($results.count -gt $global:QueryLimit)
                {
                    break;
                }
            }
            catch
            {
                Write-host ("Error QueryData2 " + $URL + $result.next + " " +  $_.Exception.Message + " " +
$_.Exception.ItemName) -ForegroundColor Red
                return $results
            }
        }
        return $results
}

function CallAPI ($URL, $Header)
{
    $Collection = QueryData $URL $Header
    foreach ($Advistory in $Collection)
    {
        #$Advistory
        $advisoryDetails = QueryData ("https://api.app.flexerasoftware.com/api/advisories/" +
$Advistory.id +"/") $Header
        $advisoryDetails

        #Remove this and it will loop over the first $global:QueryLimit advistories and stop
        break;
    }

}
$WebServiceHeader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$WebServiceHeader.Add("Content-Type", 'application/json')
$WebServiceHeader.Add("Authorization", "Token YOURTOKENHERE" )
CallAPI  "https://api.app.flexerasoftware.com/api/advisories/" $WebServiceHeader
```
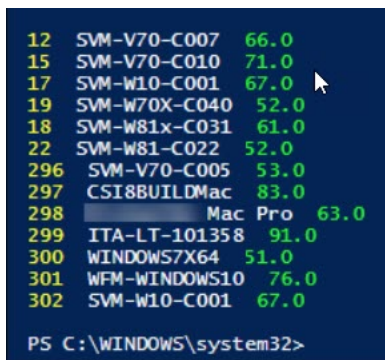
PowerShell Script to List All Devices and Their System Scores

Below is a sample PowerShell script to list all devices and their system scores:

```
$global:WebServiceHeader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$global:WebServiceHeader.Add("Content-Type", 'application/json')
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$global:WebServiceHeader.Add("Authorization", 'Token YOURTOKENHERE')
$global:WebServiceURLSecunia = "https://api.app.secunia.com/api/"
# Get First Page of results (20 items)
$result = Invoke-RestMethod ($global:WebServiceURLSecunia + "inventory/hosts/") -Method Get -Headers
$global:WebServiceHeader
$results = $result.results
#Get the next pages of results, if any
while ($result.next)
{
    $result = Invoke-RestMethod $result.next -Method Get -Headers $global:WebServiceHeader
    $results += $result.results
}
#Simple Dump the data ID then Name
foreach  ($item in $results)
{
    Write-Host $item.id -ForegroundColor Yellow -NoNewline
    Write-Host " " $item.name -ForegroundColor White -NoNewline
    Write-Host " " $item.stat.system_score -ForegroundColor Green
}
```

Below is the sample output:

PowerShell Script to Save All Advisories Within a Date Range to CSV

Below is a sample PowerShell script to save all advisories within a date range to a CSV file:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
#Max number of advistories to pull
$global:QueryLimit = 1500
#FileName
```

```
$FileName = "c:\api_advisories.csv"
function QueryData ($URL, $Header)
{
# Get First Page of results (20 items)
$result = @()
$results = @()
try
{
$result = Invoke-RestMethod ($URL) -Method Get -Headers $Header
$results = $result.results
if ($result.results)
{
$results = $result.results
}
else
{
$results = $result
}
}
catch
{
Write-host ("Error QueryData1 " + $URL + " " + $_.Exception.Message + " " + $_.Exception.ItemName) -
ForegroundColor Red
}
#Get the next pages of results, if any
while (![string]::IsNullOrWhiteSpace($result.next))
{
try
{
$result = Invoke-RestMethod $result.next -Method Get -Headers $Header
$results += $result.results
if ($results.count -gt $global:QueryLimit)
{
break;
}
}
catch
{
Write-host ("Error QueryData2 " + $URL + $result.next + " " + $_.Exception.Message + " " +
$_.Exception.ItemName) -ForegroundColor Red
return $results
}
}
return $results
}
function CallAPI ($URL, $Header)
{
$Collection = QueryData $URL $Header
$CustomCollection = @()
foreach ($Advistory in $Collection)
{
#$Advistory
$advisoryDetails = QueryData ("https://api.app.secunia.com/api/advisories/" + $Advistory.id +"/")
$Header
$products = ""
foreach ($product in $advisoryDetails.products)
```

```
{
$Productdata = QueryData ("https://api.app.flexerasoftware.com/api/product-releases/" + $product.id +"/
") $Header
$products += $Productdata.name + ","
}
$Data = New-Object System.Object
$Data | Add-Member -MemberType NoteProperty -Name "id" -Value ($advisoryDetails.id -replace "\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "said" -Value ($advisoryDetails.advisory_identifier -
replace "\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "criticality" -Value ($advisoryDetails.criticality -
replace "\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "criticality_description" -Value
($advisoryDetails.criticality_description -replace "\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "title" -Value ($advisoryDetails.title -replace
"\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "description" -Value ($advisoryDetails.description -
replace "\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "solution" -Value ($advisoryDetails.solution -replace
"\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "cvss_score" -Value ($advisoryDetails.cvss_score -
replace "\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "cvss3_score" -Value ($advisoryDetails.cvss3_score -
replace "\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "products" -Value ($products -replace "\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "cve_str_list" -Value ($advisoryDetails.cve_str_list -
replace "\r\n", " ")
$Data | Add-Member -MemberType NoteProperty -Name "released" -Value ($advisoryDetails.released -replace
"\r\n", " ")
$refs = ""
foreach ($ref in $advisoryDetails.references)
{
$refs += $ref.url + ","
}
$Data | Add-Member -MemberType NoteProperty -Name "Refs" -Value $refs
#$Data | Add-Member -MemberType NoteProperty -Name "references" -Value ($advisoryDetails.references -
replace "\r\n", " ")
$CustomCollection += $Data
}
return $CustomCollection
}
$WebServiceHeader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$WebServiceHeader.Add("Content-Type", 'application/json')
$WebServiceHeader.Add("Authorization", "Token YOURTOKENHERE" )
$CustomCollection = CallAPI "https://api.app.secunia.com/api/advisories/
?released__gte=1529038800000&released__lt=1530421199000" $WebServiceHeader
$CustomCollection | Export-Csv -path $FileName -NoTypeInformation
$CustomCollection
```

PowerShell Script to Query Historic Advisories by Product and Version

The following is a PowerShell script to query historic advisories by product and version.

```
##################################################################################################
########################################
$Site = ( "Account",      "https://api.app.flexerasoftware.com/api/"     ,"Token
00000000000000000000000000000000000000")
###############################################################################^^^^^^^^^^^^^YOUR  TOKEN
HERE^^^^^^^^^^^^########################
$global:QueryLimit = 10000
$global:WebServiceHeader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$global:WebServiceHeader.Add("Content-Type", 'application/json')
$global:WebServiceHeader.Add("Authorization", $Site[2])
$global:WebServiceURLSecunia = $Site[1]
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$global:ErrorArray = @()
function Display-Errors ()
{
    if ($global:ErrorArray.Count -eq 0)
    {
         #Write-Host (" All Good "  + (Write-Header)) $false
    }
    else
    {
        Write-Host  (" Errors: ") -ForegroundColor Yellow
        foreach ($item in $global:ErrorArray)
        {
            Write-Host  ("     " + $item ) -ForegroundColor Green
        }
    }
}
function DeleteData ($BaseURL, $Header, $URL)
{
    try
    {
        $result = Invoke-RestMethod ($BaseURL + $URL) -Method Delete -Headers $Header
    }
    catch
    {
        $global:ErrorArray += ("Error QueryData " + $BaseURL + $URL + " " +  $_.Exception.Message + " "
+ $_.Exception.ItemName)
    }
}
function QueryData ($BaseURL, $Header, $URL)
{
    # Get First Page of results (20 items)
    $result = @()
    $results = @()
    try
    {
        $result = Invoke-RestMethod ($BaseURL + $URL) -Method Get -Headers $Header
        if ($result.results)
        {
            $results = $result.results
        }
        else
        {
            $results = $result
        }
```

```
        }
        catch
        {
            $global:ErrorArray += ("Error QueryData1 " + $BaseURL + $URL + " " +  $_.Exception.Message + " "
+ $_.Exception.ItemName)
        }
        #Get the next pages of results, if any
        while (![string]::IsNullOrWhiteSpace($result.next))
        {
            try
            {
                $result = Invoke-RestMethod $result.next -Method Get -Headers $Header
                $results += $result.results
                if ($results.count -gt $global:QueryLimit)
                {
                    break;
                }
            }
            catch
            {
                $global:ErrorArray += ("Error QueryData2 " + $result.next + " " +  $_.Exception.Message + "
" + $_.Exception.ItemName)
                return $results
            }
        }
        return $results
}
function FindAssetList ($URL, $match)
{
    $Hosts = QueryData $global:WebServiceURLSecunia $global:WebServiceHeader $URL
    foreach  ($item in $Hosts)
    {
        if ($item.name -like $match)
        {
            Write-Host "Match Found" $item.id $item.name
            return $item.id
        }
        else
        {
            Write-Host "Match Not Found" $item.id $item.name
        }
    }
    return 0
}
function FindItem ($URL, $match)
{
    $items = QueryData $global:WebServiceURLSecunia $global:WebServiceHeader $URL
    foreach  ($item in $items)
    {
        if ($item.name -like $match)
        {
            Write-Host "Match Found" $item.id $item.name
            return $item.id
        }
        else
        {
```

```
                    # Write-Host "Match Not Found" $item.id $item.name
            }
        }
        return 0
    }
    function DisplayRelatedData ($URL)
    {
        $items = QueryData $global:WebServiceURLSecunia $global:WebServiceHeader $URL
        foreach  ($item in $items)
        {
            Write-Host "    " "Related Products:" -ForegroundColor Yellow
            foreach  ($product in $items.products)
            {
                Write-Host "                   " $product.name
            }
        }
    }
    function DisplaySAIDData ($URL)
    {
        $items = QueryData $global:WebServiceURLSecunia $global:WebServiceHeader $URL
        foreach  ($item in $items)
        {
            Write-Host $item.advisory_identifier $item.title
            DisplayRelatedData ("advisories/" + $item.id + "/")
        }
    }
    $AssetListName = "CHrome"
    $AssetListID = FindItem "asset-lists/" $AssetListName
    if ($AssetListID -ne 0)
    {
        DisplaySAIDData ("historic-advisories/?asset_list=" + $AssetListID)
    }
    Display-Errors
```

5

Assessment Module API Information

Edition • *The Assessment module is not available for Software Vulnerability Research.*

Important • *The following information has been taken from the individual links in the API Root screen and becomes available when you press* ***Toggle full documentation****. The information can become obsolete and you should* ***always*** *check the API information inside the portal.*

The links to the various portals displayed in your API Root screen are the ones you have access to based on your subscription and user groups. These may not match the links given below.

This section includes the following API information for the Assessment module.

- Device Groups

- Devices

- Overview of the Major Product Versions Detected on Devices

- Major Product Versions Detected on Devices for Device Groups

- Advisories Detected on Devices for Device Groups

- Advisories Detected on Devices

- PowerShell Script to Look at Device Data

- PowerShell Script to Look at Product Data

- PowerShell Script to Look at Hosts and Their Advisories Since a Specific Date

- Query Assessment Data Based on Smart Groups

Device Groups

API Supported Endpoint Actions and Available Methods for Device Group APIs include:

- Available Methods for Device Groups

- Available Filters on Device Groups List

Available Methods for Device Groups

The following are available methods for Device Groups.

Table 5-1 • Methods for Device Groups

Method	Description
get list	GET <URL>
get instance details	GET <URL><id>/
create instance	POST <URL>

Available Filters on Device Groups List

The following are available filters on Device Groups List.

Table 5-2 • Filters on Device Groups Lists

Filter	Description
id (int)	Exact match on the id of Device group
name (string)	Name of the Device group
path (string)	Tath for Device group
source (int)	Source types: - **0**—Active Directory - **1**—Smart Group

Table 5-2 • Filters on Device Groups Lists

Filter	Description
`type (int)`	External types for Device group: • **0**—None • **1**—Domain Component • **2**—Common Name • **3**—Organizational Unit • **4**—Organization Name • **5**—Street Address • **6**—Locality • **7**—State or Province • **8**—Country • **9**—Userid

Example

The following is a filter to filter out a Device group, where group name is `some-lan-group`:

`api/inventory/host-groups/?name=some-lan-group`

Devices

API Supported Endpoint Actions and Available Methods for Device APIs include:

• Available Methods for Devices

• Available Filters on Devices List

Available Methods for Devices

The following are available methods for Devices:

Table 5-3 • Methods for Devices

Method	Description
get list	`GET <URL>`
get instance details	`GET <URL><id>/`

Available Filters on Devices List

The following are available filters on Device Lists:

Table 5-4 • Filters on Devices List

Filter	Description
name(string)	Name of Device
last_scan_date__gte (int)	Unix timestamp for the last scan date of the Device, filter type greater than or equal (seconds)
last_scan_date__lt (int)	Unix timestamp for the last scan date of the Device, filter type greater than or equal (seconds)
system_score__gte (int)	Unix timestamp for the system score date of the Device, filter type greater than or equal (seconds)
system_score__lt (int)	Unix timestamp for the system score date of the Device, filter type less than or equal (seconds)
is_insecure (bool)	Filters the insecure Device
is_secure (bool)	Filters the secure Device that is secure
secure_type (int)	Secure type of Device: • **0**—Insecure • **1**—Secure
platform (int)	Platform / Operating system for Device: • **0**—All • **1**—Windows • **2**—Mac • **3**—Red Hat • **4**—Android • **5**—iOS • **6**—Debian
max_criticality (int)	Maximum criticality for Device
max_where (int)	Maximum where for Device
max_solution_status (int)	Maximum solution status for Device
system_score_ranges (int)	Maximum score range for Device

Table 5-4 • Filters on Devices List

Filter	Description
last_scan_status (int)	Last scan status for Device

Example

The following is an example of filter out a Device, where Device name is some-Device-name:

```
api/inventory/hosts/?name=some-Device-name
```

Overview of the Major Product Versions Detected on Devices

This section describes the API-supported endpoint actions and available methods for overview of the major product versions detected on the device APIs.

- Available Methods for Overview of the Major Product Versions Detected on Devices

- Available Filters on Overview of the Major Product Versions Detected on the Devices List

Available Methods for Overview of the Major Product Versions Detected on Devices

The following are the available methods for overview of the major product versions detected on Devices:

Table 5-5 • Methods for Overview of the Major Product Versions Detected on Devices

Method	Description
get list	GET <URL>

Available Filters on Overview of the Major Product Versions Detected on the Devices List

The following are available filters on overview of the major product versions detected on the Devices List:

Table 5-6 • Filters on Overview of the Major Product Versions Detected on the Devices List

Filter	Description
product__name (string)	Name of the product
product__name_startswith (string)	Name of the product starts with
product__version (string)	Version of the product
is_insecure (bool)	Filters the insecure products
is_eol(bool)	Filters product that is end of life or not

Table 5-6 • Filters on Overview of the Major Product Versions Detected on the Devices List

Filter	Description
is_secure (bool)	Filters the secure products
vendor__name (string)	Name of the vendor
max_criticality (int)	Maximum criticality for product
max_where (int)	Maximum where for product
max_solution_status (int)	Maximum solution status for product

Example

The following is an example of a filter that filters out a product that is end of life.

api/inventory/products/?is_eol=true

Major Product Versions Detected on Devices for Device Groups

This section describes the API-supported endpoint actions and available methods for major product versions detected on devices for Device Group APIs.

- Available Methods for Major Product Versions Detected on Devices for Device Groups
- Available Filters on Major Product Versions Detected on Devices for the Device Groups List

Available Methods for Major Product Versions Detected on Devices for Device Groups

The following are the available methods for major product versions detected on Devices for Device Groups.

Table 5-7 • Methods for Major Product Versions Detected on Devices for Device Groups

Method	Description
get list	GET <URL>

Available Filters on Major Product Versions Detected on Devices for the Device Groups List

The following are available filters on major product versions detected on the Devices for the Device Groups

Table 5-8 • Filters on Major Product Versions Detected on the Devices for the Device Group

Filter	Description
product__name (string)	Name of the product

Table 5-8 • Filters on Major Product Versions Detected on the Devices for the Device Group

Filter	Description
product__name_startswith (string)	Name of the product starts with
product__version (string)	Version of the product
is_insecure (bool)	Filters the insecure products
is_eol(bool)	Filters product that is end of life or not
is_secure (bool)	Filters the secure products
vendor__name (string)	Name of the vendor
max_criticality (int)	Maximum criticality for product
max_where (int)	Maximum where for product
max_solution_status (int)	Maximum solution status for product

Example

The following is an example of a filter that filters out a product that is end of life:

```
api/inventory/products-stats/?is_eol=true
```

Advisories Detected on Devices for Device Groups

This section described the API-supported endpoint actions and available methods for advisories detected on the Devices for Device Group APIs.

- Available Methods for Advisories Detected on Devices for Device Groups
- Available Filters on Advisories Detected on Devices for the Device Groups List

Available Methods for Advisories Detected on Devices for Device Groups

The following are the available methods for advisories detected on Devices for Device Groups..

Table 5-9 • Methods for Advisories Detected on Devices for Device Groups

Method	Description
get list	GET <URL>
get advisory details	GET <URL><id / advisory_identifier>/

Examples

```
/api/inventory/advisories-stats/178453/
```

```
/api/inventory/advisories-stats/SA66828/
```

📄

Note • *The advisory identifier represents a unique identifier for the Secunia advisories visible on the site, while the ID is uncorrelated and represents an internal ID.*

Available Filters on Advisories Detected on Devices for the Device Groups List

The following are available filters on advisories detected on Devices for the Device Groups List:

Table 5-10 • Filters on Advisories Detected on Devices for the Device Groups List

Filter	Description
identifier (string)	Exact match on the advisory main identifier (e.g. SA65472)
title (string)	Case insensitive search in the title of the advisory
criticality (int / list of int)	Criticality type: • **0**—Rejected • **1**—Extremely critical • **2**—Highly critical • **3**—Moderately critical • **4**—Less critical • **5**—Not critical
where (int / list of int)	Where type: • **0**—None • **1**—From remote • **2**—From local network • **3**—Local system

Table 5-10 • Filters on Advisories Detected on Devices for the Device Groups List

Filter	Description
`impact (int / list of int)`	Impact type: • **1**—System access • **2**—DoS • **3**—Privilege escalation • **4**—Exposure of sensitive information • **5**—Exposure of system information • **6**—Brute force • **7**—Manipulation of data • **8**—Spoofing • **9**—Cross-site Scripting • **10**—Security Bypass • **11**—Hijacking • **12**—Unknown
`solution_status (int)`	Solution type: • **0**—None • **1**—No Fix • **2**—Vendor Patched • **3**—Vendor Workaround • **4**—Partial Fix
`released__gte (int)`	Unix timestamp for the release date of the advisory, filter type greater than or equal (seconds)
`released__lt (int)`	Unix timestamp for the release date of the advisory, filter type less than (seconds)
`modified__gte (int)`	Unix timestamp for the last modified date of the advisory, filter type greater than or equal (seconds)
`modified__lt (int)`	Unix timestamp for the last modified date of the advisory, filter type less than (seconds)
`product_release_id (int)`	Product Version (Release) ID filter, filters the advisories released for a specific product release
`product_id (int)`	Product ID filter, filters the advisories released for a specific product

Table 5-10 • Filters on Advisories Detected on Devices for the Device Groups List

Filter	Description
vendor_id (int)	Product ID filter, filters the advisories released for a specific product
is_zero_day (bool)	Filters the zero day advisories
CVE (string)	Filters the advisories with a specific CVE. Example: CVE-2015-0286
cvss_score__gte (decimal)	CVSS Score greater than or equal filter. Example: 8.5
cvss_score__lte (decimal)	CVSS Score less than or equal filter. Example: 9.5
type (int)	Available based on licensing, it offers the possibility to search the rejected advisories: • **0**—Secunia advisory • **1**—Secunia Rejected Advisory

Example

The following is an example of a filter to display advisories released in July 2015 that are highly and extremely critical:

```
/api/inventory/advisories-stats/
?released__gte=1435698000&released__lt=1438376400&criticality=1&criticality=2
```

Advisories Detected on Devices

This section describes the API-supported endpoint actions and available methods for advisories detected on device APIs:

• Available Methods for Advisories Detected on Devices

• Available Filters on Advisories Detected on Devices List

Available Methods for Advisories Detected on Devices

The following are the available methods for advisories detected on Devices.

Table 5-11 • Methods for Advisories Detected on Devices

Method	Description
get list	GET <URL>
get advisory details	GET <URL><id / advisory_identifier>/

Examples

`/api/inventory/advisories/178453/`

`/api/inventory/advisories/SA66828/`

📄

Note • The advisory identifier represents a unique identifier for the Secunia advisories visible on the site, while the ID is uncorrelated and represents an internal ID.

Available Filters on Advisories Detected on Devices List

The following are available filters on advisories detected on Devices List:

Table 5-12 • Filters on Advisories Detected on Devices List

Filter	Description
`identifier (string)`	Exact match on the advisory main identifier (e.g. SA65472)
`title (string)`	Case insensitive search in the title of the advisory
`criticality (int / list of int)`	Criticality type: ● **0**—Rejected ● **1**—Extremely critical ● **2**—Highly critical ● **3**—Moderately critical ● **4**—Less critical ● **5**—Not critical
`where (int / list of int)`	Where type: ● **0**—None ● **1**—From remote ● **2**—From local network ● **3**—Local system

Table 5-12 • Filters on Advisories Detected on Devices List

Filter	Description
`impact (int / list of int)`	Impact type: • **1**—System access • **2**—DoS • **3**—Privilege escalation • **4**—Exposure of sensitive information • **5**—Exposure of system information • **6**—Brute force • **7**—Manipulation of data • **8**—Spoofing • **9**—Cross-site Scripting • **10**—Security Bypass • **11**—Hijacking • **12**—Unknown
`solution_status (int)`	Solution type: • **0**—None • **1**—No Fix • **2**—Vendor Patched • **3**—Vendor Workaround • **4**—Partial Fix
`released__gte (int)`	Unix timestamp for the release date of the advisory, filter type greater than or equal (seconds)
`released__lt (int)`	Unix timestamp for the release date of the advisory, filter type less than (seconds)
`modified__gte (int)`	Unix timestamp for the last modified date of the advisory, filter type greater than or equal (seconds)
`modified__lt (int)`	Unix timestamp for the last modified date of the advisory, filter type less than (seconds)
`product_release_id (int)`	Product Version (Release) ID filter, filters the advisories released for a specific product release
`product_id (int)`	Product ID filter, filters the advisories released for a specific product

Table 5-12 • Filters on Advisories Detected on Devices List

Filter	Description
vendor_id (int)	Product ID filter, filters the advisories released for a specific product
is_zero_day (bool)	Filters the zero day advisories
CVE (string)	Filters the advisories with a specific CVE. Example: CVE-2015-0286
cvss_score__gte (decimal)	CVSS Score greater than or equal filter. Example: 8.5
cvss_score__lte (decimal)	CVSS Score less than or equal filter. Example: 9.5
type (int)	Available based on licensing, it offers the possibility to search the rejected advisories: • **0**—Secunia advisory • **1**—Secunia Rejected Advisory

Example

The following is an example of a filter to display advisories released in July 2015 that are highly and extremely critical:

/api/inventory/advisories/?released__gte=1435698000&released__lt=1438376400&criticality=1&criticality=2

PowerShell Script to Look at Device Data

The end point to look at device (host) data is:

https://api.app.flexerasoftware.com/api/inventory/hosts/

To get the Device Data List, use:

GET /api/inventory/hosts/

Below is a sample PowerShell script to look at device data:

```
$global:WebServiceHeader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$global:WebServiceHeader.Add("Content-Type", 'application/json')
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$global:WebServiceHeader.Add("Authorization", 'Token YOURTOKENHERE')
$global:WebServiceURLSecunia = "https://api.app.secunia.com/api/"
# Get First Page of results (20 items)
$result = Invoke-RestMethod ($global:WebServiceURLSecunia + "inventory/hosts/") -Method Get -Headers
$global:WebServiceHeader
$results = $result.results
#Get the next pages of results, if any
while ($result.next)
{
    $result = Invoke-RestMethod $result.next -Method Get -Headers $global:WebServiceHeader
    $results += $result.results
}
#Simple Dump the data ID then Name
```

```
foreach  ($item in $results)
{
    Write-Host $item.id $item.name
}
#Data that you can get from each item
$results[0]
```

PowerShell Script to Look at Product Data

The end point to look at product data is:

 https://api.app.flexerasoftware.com/api/inventory/products/

To get the product data list, use the following:

```
GET /api/inventory/products/
```

Below is a sample PowerShell script to look at product data:

```
$global:WebServiceHeader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$global:WebServiceHeader.Add("Content-Type", 'application/json')
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$global:WebServiceHeader.Add("Authorization", 'Token YOURTOKENHERE')
$global:WebServiceURLSecunia = "https://api.app.secunia.com/api/
#Get First Page of results (20 items)
$result = Invoke-RestMethod ($global:WebServiceURLSecunia + "inventory/products/") -Method Get -Headers
$global:WebServiceHeader
$results = $result.results
#Get the next pages of results, if any
while ($result.next)
{
  $result = Invoke-RestMethod $result.next -Method Get -Headers $global:WebServiceHeader
  $results += $result.results
}
#Simple Dump the data ID then Name
foreach  ($item in $results)
{
    Write-Host $item.product.name "Installed" $item.stat.hosts "Insecure" $item.stat.insecure_hosts
}
#Data that you can get from each item
$result.results[0]
```

PowerShell Script to Look at Hosts and Their Advisories Since a Specific Date

The end point to look at hosts and their advisory data is:

https://api.app.flexerasoftware.com/api/inventory/hosts/510/advisories/

To get the host and their advisories list, use the following:

```
GET /api/inventory/hosts/510/advisories/
```

Below is a sample PowerShell script to look at hosts and their advisories since a specific date:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$global:ErrorArray = @()
$global:QueryLimit = 2000  #<- Increase to max number of hosts you want...
################################################################################################
#########################################################################
#
Name                    URL                                             Token
    $Sites = ( "Flexera SVM",    "https://api.app.flexerasoftware.com/api/"      ,"Token YOUR TOKEN
HERE")
$Header = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$Header.Add("Content-Type", 'application/json')
$Header.Add("Authorization", $Sites[2] )
function Display-Errors ()
{
    if ($global:ErrorArray.Count -eq 0)
    {
         #Write-Message ((Write-Spacing) + " All Good "  + (Write-Header)) $false
    }
    else
    {
        Write-Message (" Errors: ") $true
        foreach ($item in $global:ErrorArray)
        {
            Write-Message ("      " + $item + " " + (Write-Header)) $true
        }
    }
}
function QueryData ($BaseURL, $Header, $URL)
{
    # Get First Page of results (20 items)
    $result = @()
    $results = @()
    try
    {
        $result = Invoke-RestMethod ($BaseURL + $URL) -Method Get -Headers $Header
        if ($result.results)
        {
            $results = $result.results
        }
        else
        {
            $results = $result
        }
    }
    catch
    {
        $global:ErrorArray += ("Error QueryData1 " + $BaseURL + $URL + " " +  $_.Exception.Message + " "
+ $_.Exception.ItemName)
    }
    #Get the next pages of results, if any
    while (![string]::IsNullOrWhiteSpace($result.next))
    {
        try
        {
            $result = Invoke-RestMethod $result.next -Method Get -Headers $Header
            $results += $result.results
```

```
                    if ($results.count -gt $global:QueryLimit)
                    {
                        break;
                    }
                }
            catch
            {
                $global:ErrorArray += ("Error QueryData2 " + $URL + $result.next + " "
+  $_.Exception.Message + " " + $_.Exception.ItemName)
                return $results
            }
        }
    return $results
}
function ShowHostData ($BaseURL, $Header, $StartDate, $Date)
{
    $Hosts = QueryData $BaseURL $Header "inventory/hosts/"
    foreach  ($hostItem in $Hosts)
    {
        Write-Host $hostItem.Name -ForegroundColor Green
        $Advisories = QueryData $BaseURL $Header ("inventory/hosts/" + $hostItem.id + "/advisories/
?modified__gte=" + $Date)
        if ($Advisories.count -eq 0)
        {
             Write-Host "   " "No Advisories Since " $StartDate
        }
        else
        {
            foreach  ($item in $Advisories)
            {
                Write-Host "   "  $item.advisory_identifier $item.title $item.modified_date
            }
        }
    }
}
####################################
# Get Advisories Data since this date
$StartDate = "9/1/2018"
####################################
$date1 = Get-Date -Date "01/01/1970"
$date2 = Get-Date -Date $StartDate
$UnixDate = (New-TimeSpan -Start $date1 -End $date2).TotalSeconds
ShowHostData $Sites[1] $Header $StartDate $UnixDate
Display-Errors
```

Query Assessment Data Based on Smart Groups

To query assessment data based on Smart Groups, perform the following steps.

Task   ***To query assessment data based on Smart Groups;***

1. Use the following URL: https://app.flexerasoftware.com/api/inventory/host-groups/top_custom/

   ```
   [{"id":122,"name":"Server
   SVM","path":null,"level":0,"children_count":0,"reprocess":true,"source":1,"priority":1}]
   ```

2. Pull the ID from you smart group you wish to query (122 is the ID in the example above).

3. Insert the Smart Group ID in API calls (using 122 for our example):

   ● https://api.app.flexerasoftware.com/api/inventory/host-groups/122/

   ● https://api.app.flexerasoftware.com/api/inventory/host-groups/122/advisories/

   ● https://api.app.flexerasoftware.com/api/inventory/host-groups/122/hosts/

   ● https://api.app.flexerasoftware.com/api/inventory/host-groups/122/products/

6

Patching Module API Information

Edition • *The Patching module is not available for Software Vulnerability Research.*

Important • *The following information has been taken from the individual links in the API Root screen and becomes available when you press* **Toggle full documentation***. The information can become obsolete and you should* **always** *check the API information inside the portal.*

The links to the various portals displayed in your API Root screen are the ones you have access to based on your subscription and user groups. These may not match the links given below.

This section includes the following API information for the Patching module.

- Daemon Lists
- Server Details
- Server Group Details
- Customer Patch Template Name Details
- Customer Patch Template Created by Details
- Patchable Product Details
- Patch Package Details
- Customer's Patch Package Publishing Details
- Patch Tasks
- Patches Available
- Available Patches Grouped
- Patch Language
- Publish Patch List

- Patch Package List

- Product Release Instance

- PowerShell Script to Delete Data

Daemon Lists

This section describes the API-supported endpoint actions and available methods for Daemon List APIs:

- Available Methods for Daemon Lists

- Available Filters on Daemon Lists

Available Methods for Daemon Lists

The following are available methods for Daemon Lists.

Table 6-1 • Methods for Daemon Lists

Method	Description
get list	GET <URL>

Available Filters on Daemon Lists

The following are available filters on Daemon Lists.

Table 6-2 • Filters on Device Groups Lists

Filter	Description
last_connection_date__gte(int)	Unix timestamp for the last connection date of the daemon, filter type greater than or equal (seconds)
last_connection_date__lt(int)	Unix timestamp for the last connection date of the daemon, filter type less than or equal (seconds)

Example

The following is an example of a filter that filters out a daemon whose last connection date is July 2015:

api/patch/daemons/?last_connection_date__gte=1435698000&last_connection_date__lt=1438376400

Server Details

This section describes the API-supported endpoint actions and available methods for Server Detail APIs.

- Available Methods for Server Details

- Available Filters on Server Detail Lists

Available Methods for Server Details

The following are available methods for Server Details.

Table 6-3 • Methods for Server Details

Method	Description
get list	GET <URL>

Available Filters on Server Detail Lists

The following are available filters on Server Detail Lists.

Table 6-4 • Filters on Server Detail Lists

Filter	Description
name (string)	Name of the server
external_id (string)	External id of the server

Example

The following is an example of a filter to filter out a server whose name is "my-server".

```
api/patch/servers/?name=my-server
```

Server Group Details

The following section describes the API-supported endpoint actions and available methods for Server Group Detail APIs.

- Available Methods for Server Group Details
- Available Filters on Server Group Detail Lists

Available Methods for Server Group Details

The following are available methods for Server Group Details.

Table 6-5 • Methods for Server Group Details

Method	Description
get list	GET <URL>

Available Filters on Server Group Detail Lists

The following are available filters on Server Group Detail Lists.

Table 6-6 • Filters on Server Group Detail Lists

Filter	Description
name (string)	Name of the server group
external_id (string)	External id of the server
server_id (int)	Server identifier for a server group

Example

The following is an example of a filter that filters out a server group whose name is "my-server":

```
api/patch/groups/?name=my-server
```

Customer Patch Template Name Details

The following section describes the API-supported endpoint actions and available methods for Customer Patch Template Name APIs.

- Available Methods for Customer Patch Template Name Details
- Available Filters on Customer Patch Template Name Detail Lists

Available Methods for Customer Patch Template Name Details

The following are available methods for Customer Patch Template Name Details.

Table 6-7 • Methods for Customer Patch Template Name Details

Method	Description
get list	GET <URL>
create instance	POST <URL>

Available Filters on Customer Patch Template Name Detail Lists

The following are available filters on Customer Patch Template Name Detail Lists.

Table 6-8 • Filters on Customer Patch Template Name Detail Lists

Filter	Description
name (string)	Name of the patch template
has_customer_template (bool)	Patch has customer template or not

Example

The following filter filters out patch templates whose name is "xyz".

```
api/patch/patch-templates/?name=xyz
```

Customer Patch Template Created by Details

The following section describes the API-supported endpoint actions and available methods for Customer Patch Template Created by APIs.

- Available Methods for Customer Patch Template Created by Details

- Available Filters on Customer Patch Template Created by Lists

Available Methods for Customer Patch Template Created by Details

The following are available methods for Customer Patch Template Created by Details.

Table 6-9 • Methods for Customer Patch Template Created by Details

Method	Description
get list	GET <URL>
create instance	POST <URL>

Available Filters on Customer Patch Template Created by Lists

The following are available filters on Customer Patch Template Created by Lists.

Table 6-10 • Filters on Customer Patch Template Created by Lists

Filter	Description
`name (string)`	Name of the patch template
`description (string)`	Description about patch template
`patch_template_id (int)`	Identifier for patch template
`created_by_id (int)`	Identifier for created by
`for_architecture (int)`	Architecture type for patch template. Architecture types are: • **0**—32-bit/64-bit • **1**—32-bit • **2**—64-bit
`for_languages (list)`	Languages `iso_code`, `language_display`

Table 6-10 • Filters on Customer Patch Template Created by Lists

Filter	Description
product_id (int)	Product identifier for patch template
edition (string)	Edition for patch template

Example

The following is an example of a filter that filters out patch templates created by user id 3.

api/patch/customer-patch-templates/?created_by_id=3

Patchable Product Details

The following section describes the API-supported endpoint actions and available methods for Patchable Product APIs.

● Available Methods for Patchable Product Details

● Available Filters on Patchable Product Lists

Available Methods for Patchable Product Details

The following are available methods for Patchable Product Details.

Table 6-11 • Methods for Patchable Product Details

Method	Description
get list	GET <URL>

Available Filters on Patchable Product Lists

The following are available filters on Patchable Product Lists.

Table 6-12 • Filters on Patchable Product Lists

Filter	Description
patch_template_id (int)	Identifier for patch template
product_release_id (int)	Identifier for product release
architecture (int)	Architecture type: ● **0**—32-bit/64-bit ● **1**—32-bit ● **2**—64-bit

Table 6-12 • Filters on Patchable Product Lists

Filter	Description
`platform (int)`	Operating system: • **0**—All • **1**—Windows • **2**—Mac • **3**—Red Hat • **4**—Android • **5**—iOS • **6**—Debian
`edition (string)`	Edition or version for product
`product_name (string)`	Product name
`vendor_name (string)`	Vendor name
`created_gte`	Unix timestamp for the patchable product create date, filter type greater than or equal (seconds)
`created_lt`	Unix timestamp for the patchable product create date, filter type less than (seconds)

Example

The following is an example of a filter that filters out patchable product whose product name id "java".

```
api/patch/customer-patch-templates/?product_name=java
```

Patch Package Details

The following section describes the API-supported endpoint actions and available methods for Package APIs.

• Available Methods for Patch Package Details

• Available Filters on Patch Package Lists

Available Methods for Patch Package Details

The following are available methods for Patch Package Details.

Table 6-13 • Methods for Patch Package Details

Method	Description
get list	`GET <URL>`

Table 6-13 • Methods for Patch Package Details

Method	Description
get package details	GET <URL><id>

Available Filters on Patch Package Lists

The following are available filters on Patch Package Lists.

Table 6-14 • Filters on Patch Package Lists

Filter	Description
id (int)	Identifier for package
customer_patch_template_id (int)	Identifier for customer package template
name (string)	Name of the package
type (int)	Package type: • **0**—Install/Update • **1**—Uninstall • **2**—Install/Update/Uninstall • **3**—Custom • **4**—agent_deployment
product_release_id (int)	Identifier for product release
product_name (string)	Name of the product
vendor_name (string)	Vendor name
status (int)	Identifier for status. Status: • **0**—Not Ready • **1**—Building • **2**—Ready • **3**—Error building it
solution_id (int)	Solution ID: • **0**—"default" - from old sr_product_secure • **1**—"language" - from old sr_solution_download table, with language options • **2**—"custom" - from old solution, special because it contains parameters, special patching, exclusive

Table 6-14 • Filters on Patch Package Lists

Filter	Description
platform (int)	Operating system: ● **0**—All ● **1**—Windows ● **2**—Mac ● **3**—Red Hat ● **4**—Android ● **5**—iOS ● **6**—Debian
architecture (int)	Architecture: ● **0**—32-bit/64-bit ● **1**—32-bit ● **2**—64-bit
iso_code (string)	ISO code for package
created_gte	Unix timestamp for the package create date, filter type greater than or equal (seconds)
created_lt	Unix timestamp for the package create date, filter type less than (seconds)

Example

The following is an example of a filter that filters package whose customer patch template identifier is 1.

```
api/patch/packages/?customer_patch_template_id=1
```

Customer's Patch Package Publishing Details

The following section describes the API-supported endpoint actions and available methods for Customer's Patch Package Publishing APIs.

● Available Methods for Customer's Patch Package Publishing Details

● Available Filters on Customer's Patch Package Publishing Lists

Available Methods for Customer's Patch Package Publishing Details

The following are available methods for Customer's Patch Package Publishing Details.

Table 6-15 • Methods for Customer's Patch Package Publishing Details

Method	Description
get list	GET <URL>

Available Filters on Customer's Patch Package Publishing Lists

The following are available filters on Customer's Patch Package Publishing Lists.

Table 6-16 • Filters on Customer's Patch Package Publishing Lists

Filter	Description
id (int)	identifier for publish
package_id (int)	Identifier about patch package
package_ids (list)	Identifiers for patch package
server_id (int)	Identifier for server
state (int)	State of the published / publishing packages • **0**—Pending • **1**—Loaded • **2**—Completed • **3**—Failed • **4**—Pending Delete • **5**—Deleted • **6**—Waiting for signature
last_updated__gte (int)	Unix timestamp for the last updated date of publish, filter type greater than or equal (seconds)
last_updated__lt (int)	Unix timestamp for the last updated date of publish, filter type less than or equal (seconds)
product_name (string)	Package product name
vendor_name (string)	Package vendor name
name (string)	Package name

Example

The following is an example of a filter that filters out publish instance, where package vendor name is `java`.

`api/patch/publishes/?vendor_name=java`

Patch Tasks

The following section describes the API-supported endpoint actions and available methods for Patch Task APIs.

- Available Methods for Patch Task Details

- Available Filters on Patch Task Lists

Available Methods for Patch Task Details

The following are available methods for Patch Task Details.

Table 6-17 • Methods for Patch Task Details

Method	Description
get list	GET <URL>

Available Filters on Patch Task Lists

The following are available filters on Patch Task Lists.

Table 6-18 • Filters on Patch Task Lists

Filter	Description
`daemon_id (int)`	Daemon ID
`publish_id (int)`	Publish ID
`type (int)`	Task type:

For `type (int)`:

- **3**—Push package to Patch Server
- **6**—Approve package in Patch Server
- **7**—Unapproves package in Patch Server
- **9**—Fetches info about package from Daemon and Patch Server
- **10**—Fetches info about all packages
- **15**—Agent update
- **16**—Delete the Package
- **17**—Request package be signed for later deployment

Table 6-18 • Filters on Patch Task Lists

Filter	Description
`result (int)`	Task type result:

- **0**—New
- **1**—Queued
- **2**—Processing
- **3**—Done
- **4**—Success
- **5**—Failed
- **6**—Cancelled
- **7**—Unsupported
- **8**—Aborted
- **9**—Completed

Example

The following is an example of a filter that filters out a task whose publish id is "1234"

```
api/patch/tasks/?publish_id=1234
```

Patches Available

The following section describes the API-supported endpoint actions and available methods for Patches Available APIs.

- Available Methods for Patches Available
- Available Filters on Patches Available Lists

Available Methods for Patches Available

The following are available methods for Patches Available.

Table 6-19 • Methods for Patches Available

Method	Description
get list	`GET <URL>`
get instance details	`GET <URL><id>/`

Available Filters on Patches Available Lists

The following are available filters on Patches Available Lists.

Table 6-20 • Filters on Patches Available Lists

Filter	Description
product_release_id (int)	Release id of a product

Example

The following is an example of a filter that filters out a product whose product release id is "111":

api/patch/available-patches/?product_release_id=111

Available Patches Grouped

The end point to look at the available patches group list is: https://api.app.flexerasoftware.com/api/patch/available-patches-grouped/

The following section describes the API-supported endpoint actions and available methods for Available Patches Grouped APIs.

- Available Methods for Available Patches Grouped
- Available Filters on Available Patches Grouped Lists

Available Methods for Available Patches Grouped

The following are available methods for Available Patches Grouped.

Table 6-21 • Methods for Available Patches Grouped

Method	Description
get list	GET <URL> For example: GET /api/patch/available-patches-grouped/
get instance details	GET <URL><id>/

Available Filters on Available Patches Grouped Lists

The following are available filters on Available Patches Grouped Lists.

Table 6-22 • Filters on Available Patches Grouped Lists

Filter	Description
product_release_id (int)	Product release identifier

Table 6-22 • Filters on Available Patches Grouped Lists

Filter	Description
product_id (int)	Product identifier
product_name (string)	Product name
vendor_name (string)	Vendor name
secure_version (string)	Secure version
said (string)	Secunia Advisory ID of a product
cve (string)	Common vulnerability score of a product
has_customer_template (bool)	Product has customer template or not
has_package(bool)	Product has package or not
my_environment (bool)	Affecting my environment or not
fullver (int)	Full version of the product

Example

The following is an example of a filter that filters out a product whose product relase id is "111"

```
api/patch/available-patches-grouped/?product_release_id=111
```

Patch Language

The following section describes the API-supported endpoint actions and available methods for Patch Language APIs.

- Available Methods for Patch Language

- Available Filters on Patch Language Lists

Available Methods for Patch Language

The following are available methods for Patch Language.

Table 6-23 • Methods for Patch Language

Method	Description
get list	GET <URL>

Available Filters on Patch Language Lists

The following are available filters on Patch Language Lists.

Table 6-24 • Filters on Patch LanguageLists

Filter	Description
iso_code (string)	ISO code for language
language_display (string)	Language for display

Example

The following is an example of a filter that filters out a language whose ISO code is English US

```
api/patch/languages/?iso_code=en_US
```

Publish Patch List

The end point to look at a published patch list is:

https://api.app.flexerasoftware.com/api/patch/publishes/

To get the Publish Patch List, use:

```
GET /api/patch/publishes/
```

Patch Package List

The end point to look at a patch package list is:

https://api.app.flexerasoftware.com/api/patch/packages/

To get the Patch Package List, use:

```
GET /api/patch/packages/
```

Product Release Instance

The end point to look at a product release instance is:

https://api.app.flexerasoftware.com/api/product-releases/

To get a specific product release instance, use:

```
GET /api/product-releases/
```

PowerShell Script to Delete Data

Below is a sample PowerShell script to delete data from a Software Vulnerability Research system via automation.

⚠️

Caution • *Use extreme caution when running this script as THERE IS NO OPTION TO RESTORE DELETED DATA. The line*
#DeleteData ($URL + $item.id + "/") is commented out in the script by default. If you want the script to actually delete
data, you need to uncomment this line.

```
$global:WebServiceHeader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$global:WebServiceHeader.Add("Content-Type", 'application/json')
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$LogFile = (Join-Path $PSScriptRoot  "CleanUp.txt")
$global:ErrorArray = @()
#PROD
$global:WebServiceHeader.Add("Authorization", 'Token YOURTOKENTOKEN')
$global:WebServiceURLSecunia = "https://api.app.flexerasoftware.com/api/"
function Write-Message ($Message, $Error)
{
    $Header = $Message
    if ($Error)
    {
        Write-Host $Header -ForegroundColor Yellow
    }
    else
    {
        Write-Host $Header -ForegroundColor Green
    }
    $Header | Out-File $LogFile -Append
}
function Display-Errors ()
{
    if ($global:ErrorArray.Count -eq 0)
    {
        Write-Message (" All Good "  + (Write-Header)) $false
    }
    else
    {
        Write-Message (" Errors: ") $true
        foreach ($item in $global:ErrorArray)
        {
            Write-Message ("     " + $item + (Write-Header)) $true
        }
    }
}
function DeleteData ($URL)
{
    try
    {
        $result = Invoke-RestMethod ($global:WebServiceURLSecunia + $URL) -Method Delete -Headers
$global:WebServiceHeader
    }
    catch
    {
        $global:ErrorArray += ("Error QueryData " + $global:WebServiceURLSecunia + $URL + " "
+  $_.Exception.Message + " " + $_.Exception.ItemName)
    }
}
function QueryData ($URL)
```

```
{
    # Get First Page of results (20 items)
    $result = @()
    $results = @()
    try
    {
        $result = Invoke-RestMethod ($global:WebServiceURLSecunia + $URL) -Method Get -Headers
$global:WebServiceHeader
        $results = $result.results
    }
    catch
    {
        $global:ErrorArray += ("Error QueryData1 " + $global:WebServiceURLSecunia + $URL + " "
+ $_.Exception.Message + " " + $_.Exception.ItemName)
    }
    #Get the next pages of results, if any
    while (![string]::IsNullOrWhiteSpace($result.next))
    {
        try
        {
            $result = Invoke-RestMethod $result.next -Method Get -Headers $global:WebServiceHeader
            $results += $result.results
        }
        catch
        {
            $global:ErrorArray += ("Error QueryData2 " + $URL + $result.next + " "
+ $_.Exception.Message + " " + $_.Exception.ItemName)
            return $results
        }
    }
    return $results
}
function RemoveData ($URL, $match)
{
    $Hosts = QueryData $URL
    foreach  ($item in $Hosts)
    {
        if ($item.name -like $match)
        {
            Write-Message ('Deleting ' + $item.id + ' ' + $item.name) $true
            #DeleteData ($URL + $item.id + "/")
        }
        else
        {
            Write-Message ('Not Deleting ' +  $item.id + " " + $item.name) $false
        }
    }
}
RemoveData "inventory/hosts/" "*"
RemoveData "patch/customer-patch-templates/" "*"
RemoveData "patch/packages/" "*"
Display-Errors
```

7

Settings Module API Information

📄

Important • *The following information has been taken from the individual links in the API Root screen and becomes available when you press **Toggle full documentation**. The information can become obsolete and you should **always** check the API information inside the portal.*

The links to the various portals displayed in your API Root screen are the ones you have access to based on your subscription and user groups. These may not match the links given below.

This section includes the Settings API information for the following Settings module tabs.

- User Management

- Workflow Management

- API

User Management

The APIs for User Management include:

- Authenticated User List

- User Group List

- User Logins

- Email Logs

- SMS Logs

- Group List (Roles)

Authenticated User List

For information on the Authenticated User List APIs, see the following URL:

https://api.app.secunia.com/api/users/

The Authenticated User List is a list of users for your account; users that have access to the system per your license agreement. The number of active users represents the number of used licenses.

API-supported endpoint actions and available methods for authenticated user list APIs include:

- Available Methods for Authenticated User List

- Authenticated User List Fields for Create/Edit

Available Methods for Authenticated User List

The following are available methods for Authenticated User List.

Table 7-1 • Methods for Authenticated User List

Method	Description
get list	GET <URL>
get instance details	GET <URL><id>/
create instance	POST <URL>
edit instance	PUT <URL><id>/

Authenticated User List Fields for Create/Edit

The following are authenticated user list fields for Create/Edit.

Table 7-2 • Authenticated User List Fields for Create/Edit

Fields	Description
username (string)	Read only after create
first_name (string)	User first name
last_name (string)	User last name
job_title (string)	Job title
title (string)	
email (string)	The user's email address, mandatory and unique field
phone_number (string)	Phone number for two factor authentication and for SMS alerts must be in an international format, e.g. +1 201 555 1234
is_active (bool)	Determines if the user is still valid, can log in, receive alerts etc. The active status of an user can only be enabled after creation by the user through clicking the link from the email activation that is sent by the system.

Table 7-2 • Authenticated User List Fields for Create/Edit

Fields	Description
country (string)	The user's country
language (string)	The user's preferred language
timezone (string)	The user's preferred timezone
user_groups (list of int)	The user groups the user is included in, the permissions will be determined based on the user group affiliation

User Group List

For information on the User Group List APIs, see the following URL:

https://api.app.secunia.com/api/user-groups/

User Groups are a grouping of roles to the system and can be assigned to users. Including a user into User Groups means granting the user access to all the roles contained within those User Groups.

You have full access to the User Groups and the system offers you a list of predefined User Groups that you can edit, delete, alter and grant as you see fit.

API supported endpoint actions and available methods for User Group List APIs include:

- Available Methods for User Group List
- User Group Fields for Create/Edit

Available Methods for User Group List

The following are available methods for User Group List.

Table 7-3 • Available Methods for User Group List

Method	Description
get list	GET <URL>
get instance details	GET <URL><id>/
create instance	POST <URL>
edit instance	PUT <URL><id>/
delete instance	DELETE <URL><id>/

User Group Fields for Create/Edit

The following are user group fields for Create/Edit.

Table 7-4 • User Group Fields for Create/Edit

Field	Description
name (string)	The user group name visible in the interface
description (string)	Further information about the user group
groups (list of int)	List of system groups / roles that the user group is composed of

User Logins

For information on User Logins APIs, see the following URL:

https://api.app.secunia.com/api/audit/user-logins/

The User Logins API gives you a list of user logins.

Available Filters for User Logins

The following are available filters for User Logins.

Table 7-5 • Available Filters for User Logins

Filter	Description
start (int)	Unix timestamp for the start date
end (int)	Unix timestamp for the start date
asc (bool)	Sorting order, ascending (True) or descending (False)
page_size (int)	Page size
ref (guid)	"Next" value from a paginated response

Email Logs

For information on Email Logs APIs, see the following URL:

https://api.app.secunia.com/api/audit/email-logs/

The Email Logs API gives you a list of emails sent to your users.

Available Filters for Email Logs

The following are available filters for Email Logs.

Table 7-6 • Available Filters for Email Logs

Filter	Description
start (int)	Unix timestamp for the start date
end (int)	Unix timestamp for the start date
asc (bool)	Sorting order, ascending (True) or descending (False)
page_size (int)	Page size
ref (guid)	"Next" value from a paginated response

SMS Logs

For information on SMS Logs APIs, see the following URL:

https://api.app.secunia.com/api/audit/sms-logs/

The SMS Logs API gives you a list of SMS sent to your users.

Available Filters for SMS Logs

The following are available filters for SMS Logs.

Table 7-7 • Available Filters for SMS Logs

Filter	Description
start (int)	Unix timestamp for the start date
end (int)	Unix timestamp for the start date
asc (bool)	Sorting order, ascending (True) or descending (False)
page_size (int)	Page size
ref (guid)	"Next" value from a paginated response

Group List (Roles)

For information on the Group List (Roles) APIs, see the following URL:

https://api.app.secunia.com/api/groups/

Groups or roles are used for determining the rights a user should have to the system. The entire permission system is centered on the notion of roles and user groups.

The list of roles is predefined and can grant access and rights to different parts of the system and is determined by the purchased license.

Grouping the available roles into User Groups gives you control over who can access what.

Note • *Administrator and API User Management users can access complete data of the API.*

Available Methods for Group List

The following are available methods for Group List.

Table 7-8 • Available Methods for Group List

Method	Description
get list	GET <URL>
get instance details	GET <URL><id>/

Workflow Management

The APIs for Ticket Management include:

- Ticket List

- Ticket Queue List

- Ticket Status List

- Ticket Priority List

- Ticket Changes

- Ticket Note List

- PowerShell Script to Close Tickets Using a Certain Date

Ticket List

For information on the Ticket List APIs, see the following URL:

https://api.app.secunia.com/api/tickets/

Tickets help you keep track and resolve vulnerabilities identified for your Watch Lists.

API Supported Endpoint Actions and Available Methods for Ticket List APIs include:

- Available Methods for Ticket List

- Available Filters on Ticket List

- Create Method Fields for Ticket Lists

- Edit Method Fields for Ticket Lists

Available Methods for Ticket List

The following are available methods for Ticket List.

Table 7-9 • Available Methods for Ticket List

Method	Description
get list	GET <URL>
create instance(s)	POST <URL> 📄 **Note** • *Can create multiple tickets, one per advisory - Watch list pair.*
edit instance(s)	POST <URL>edit/ 📄 **Note** • *Can edit multiple tickets.*

Available Filters on Ticket List

The following are available filters for Ticket List.

Table 7-10 • Available Filters on Ticket List

Filter	Description
assigned_to_id (int)	Tickets assigned to a specific users, id-username list available at /api/users/kvlist/
status_id (int)	Tickets with a certain status; list available at /api/ticket-statuses/
priority_id (int)	Tickets with a certain priority; list available at /api/ticket-priorities/
queue_id (int)	Tickets on a certain queue; list available at /api/ticket-queues/
asset_list_id (int)	Tickets created for a certain Watch list; list available at /api/Watch-lists/
criticality (int)	Tickets for advisories with a certain criticality. (See criticality filter options on advisories page.)
created__gte (int)	Unix timestamp for the ticket create date, filter type greater than or equal (seconds)
created__lt (int)	Unix timestamp for the ticket create date, filter type less than (seconds)

Table 7-10 • Available Filters on Ticket List

Filter	Description
solution_status (int)	Solution type for the advisory associated with the ticket (See solution_status filter options on advisories page.)
cvss_score__gte (decimal)	CVSS Score of the advisory greater than or equal filter, e.g. 8.5
cvss_score__lte (decimal)	CVSS Score of the advisory less than or equal filter, e.g. 9.5
last_updated__gte (int)	Unix timestamp for the ticket last change date, filter type greater than or equal (seconds)
last_updated__lt (int)	Unix timestamp for the ticket last change date, filter type less than (seconds)

Create Method Fields for Ticket Lists

The following are available create method fields for Ticket Lists.

Table 7-11 • Create Method Fields for Ticket Lists

Field	Description
advisory (list of int, optional)	List of advisory ids for which the tickets should be created. A ticket will be created for each advisory id
advisory_identifier (string, ignored if advisory)	Unique advisory identifier for which the ticket should be created, Used when the advisory ids list is not present.
status_id (int, optional)	The status id for the new tickets. Default "Open"
priority_id (int, optional)	The priority id for the new tickets. Default calculated on advisory criticality
queue_id (int, optional)	The queue id for the new tickets. Default "Default"
assigned_to_id (int, optional)	To whom to assign the ticket; id-username list available at /api/users/kvlist/
asset_list (list of int, optional)	On which Watch list ids the advisory is matched. A ticket is created for each unique combination of Watch list id, advisory
comment (string, optional)	Ticket note that should be assigned to the ticket

Edit Method Fields for Ticket Lists

Allows you to edit multiple tickets (if a field does not exit, the value for that ticket doesn't change):

Table 7-12 • Edit Method Fields for Ticket Lists

Field	Description
`ticket (list of int)`	The list of ticket ids that need to be changed
`status (int, optional)`	The status id for the new tickets
`priority (int)`	The priority id for the new tickets
`queue (int)`	The queue id for the new tickets
`assigned_to (int, optional)`	To whom to assign the ticket; id-username list available at `/api/users/kvlist/`
`comment (string, optional)`	Ticket note that should be assigned to the ticket

Ticket Queue List

For information on the Ticket Queue List APIs, see the following URL:

https://api.app.secunia.com/api/ticket-queues/

Ticket queues are used to visually group together tickets, for example "EMEA Support", "Asia QA" and so on.

In the case of multiple teams with multiple Watch Lists that monitor different products, you can grant rights on ticket queues to avoid cluttering the main ticket page for a normal user.

API Supported Endpoint Actions and Available Methods for Ticket Queue List APIs include:

- Available Methods for Ticket Queue List
- Available Filters on Ticket Queue List
- Ticket Queue List Fields for Create/Edit

Available Methods for Ticket Queue List

The following are available methods for Ticket Queue List.

Table 7-13 • Available Methods for Ticket Queue List

Method	Description
get list	`GET <URL>`
get instance details	`GET <URL><id>/`
create instance	`POST <URL>`

Table 7-13 • Available Methods for Ticket Queue List

Method	Description
edit instance	`PUT <URL><id>/`
delete instance	`DELETE <URL><id>/`

Available Filters on Ticket Queue List

The following are available filters for Ticket Queue List.

Table 7-14 • Available Filters on Ticket Queue List

Filter	Description
`name (string)`	invariant case search by term in name

Ticket Queue List Fields for Create/Edit

The following are available filters for Ticket Queue List fields for Create/Edit.

Table 7-15 • Ticket Queue List Fields for Create/Edit

Field	Description
`name (string)`	The group name visible in the interface
`visible_for_account (bool)`	True if all users should see tickets from this queue
`user_groups (list of int)`	A list of user groups ids in which a specific user must be part of in order to see the tickets from the queue. Administrators see all tickets.

Ticket Status List

For information on the Ticket Status List APIs, see the following URL:

https://api.app.secunia.com/api/ticket-statuses/

Ticket statuses are used to indicate in what state the ticket currently is, e.g. "in progress", "handled".

You have control over the number of statuses you have in your workflow and an open status determines the initial state of the ticket. The default ticket statuses are used in reports and compliance policies.

API Supported Endpoint Actions and Available Methods for Ticket Queue List APIs include:

- Available Methods for Ticket Status List

- Available Filters on Ticket Status List

- Ticket Status List Fields for Create/Edit

Available Methods for Ticket Status List

The following are available methods for Ticket Status List.

Table 7-16 • Available Methods for Ticket Status List

Method	Description
get list	GET <URL>
get instance details	GET <URL><id>/
create instance	POST <URL>
edit instance	PUT <URL><id>/
delete instance	DELETE <URL><id>/

Available Filters on Ticket Status List

The following are available filters for Ticket Status List.

Table 7-17 •

Filter	Description
name (string)	Invariant case search by term in name

Ticket Status List Fields for Create/Edit

The following are ticket status list fields for Create/Edit.

Table 7-18 • Ticket Status List Fields for Create/Edit

Field	Description
name (string)	The group name visible in the interface
default ticket status (int)	The default ticket status in our system for reports and compliance policies: • **0**—Open • **1**—Waiting (or in progress) • **2**—Handled (or closed) • **3**—Irrelevant

Ticket Priority List

For information on the Ticket Priority List APIs, see the following URL:

https://api.app.secunia.com/api/ticket-priorities/

Ticket priorities help your workflow by indicating which tickets should be handled before others.

By default, the ticket priority is determined from the advisory criticality. Extremely critical advisories generate urgent tickets, highly critical advisories generate a high priority, moderately critical advisories generate medium priorities and less or not critical advisories generate low priority tickets.

API Supported Endpoint Actions and Available Methods for Ticket Priority List APIs include:

- Available Methods for Ticket Priority List
- Available Filters on Ticket Priority List
- Ticket Priority List Fields for Create/Edit

Available Methods for Ticket Priority List

The following are available methods for Ticket Priority List.

Table 7-19 • Available Methods for Ticket Priority List

Method	Description
get list	GET <URL>
get instance details	GET <URL><id>/
create instance	POST <URL>
edit instance	PUT <URL><id>/
delete instance	DELETE <URL><id>/

Available Filters on Ticket Priority List

The following are available filters for Ticket Priority List.

Table 7-20 • Available Filters on Ticket Priority List

Filter	Description
name (string)	Invariant case search by term in name

Ticket Priority List Fields for Create/Edit

The following are ticket priority list fields for Create/Edit.

Table 7-21 • Ticket Priority List Fields for Create/Edit

Field	Description
name (string)	The group name visible in the interface

Table 7-21 • Ticket Priority List Fields for Create/Edit

Field	Description
`default ticket priority (int)`	The default ticket priority in our system • **0**—Low • **1**—Medium • **2**—High • **3**—Urgent

Ticket Changes

For information on Ticket Changes APIs, see the following URL:

https://api.app.secunia.com/api/audit/ticket-changes/

List of ticket changes.

Available Filters for Ticket Changes

The following are available filters for Ticket Changes.

Table 7-22 • Available Filters for Ticket Changes

Filter	Description
`start (int)`	Unix timestamp for the start date
`end (int)`	Unix timestamp for the start date
`asc (bool)`	Sorting order, ascending (True) or descending (False)
`page_size (int)`	Page size.
`ref (guid)`	"Next" value from a paginated response
`object_id (int)`	The ticket id for which the changes were made

Ticket Note List

For information on the Ticket Note List APIs, see the following URL:

https://api.app.secunia.com/api/ticket-notes/

At any point you can make notes and comments on the ticket. For security purposes, the comments are encrypted in our database. As a direct consequence of this, ticket notes can't be searched and we can't offer free text search functionality on the notes.

API supported endpoint actions and available methods for Ticket Note List APIs include:

- Available Methods for Ticket Note List

- Available Filters on Ticket Note List

- Ticket Note List Fields for Create/Edit

Available Methods for Ticket Note List

The following are available methods for Ticket Note List.

Table 7-23 • Available Methods for Ticket Note List

Method	Description
get list	GET <URL>
get instance details	GET <URL><id>/
create instance	POST <URL>
edit instance	PUT <URL><id>/
delete instance	DELETE <URL><id>/

Available Filters on Ticket Note List

The following are available filters for Ticket Note List.

Table 7-24 • Available Filters on Ticket Note List

Filter	Description
ticket_id (int)	The parent ticket id

Ticket Note List Fields for Create/Edit

The following are ticket note list fields for Create/Edit.

Table 7-25 • Ticket Note List Fields for Create/Edit

Field	Description
ticket_id (int)	The parent ticket id on which the comment is added
comment (string)	The new comment

PowerShell Script to Close Tickets Using a Certain Date

Below is a sample PowerShell script to close tickets using a certain date:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
#Max number of advistories to pull
```

```
$global:QueryLimit = 20
function QueryData ($URL, $Header)
{
    # Get First Page of results (20 items)
    $result = @()
    $results = @()
    try
    {
        $result = Invoke-RestMethod ($URL) -Method Get -Headers $Header
        $results = $result.results
        if ($result.results)
        {
            $results = $result.results
        }
        else
        {
            $results = $result
        }
    }
    catch
    {
        Write-host ("Error QueryData1 " + $URL + " " +  $_.Exception.Message + " " +
$_.Exception.ItemName) -ForegroundColor Red
    }
    #Get the next pages of results, if any
    while (![string]::IsNullOrWhiteSpace($result.next))
    {
        try
        {
            $result = Invoke-RestMethod $result.next -Method Get -Headers $Header
            $results += $result.results
            if ($results.count -gt $global:QueryLimit)
            {
                break;
            }
        }
        catch
        {
            Write-host ("Error QueryData2 " + $URL + $result.next + " " +  $_.Exception.Message + " " +
$_.Exception.ItemName) -ForegroundColor Red
            return $results
        }
    }
    return $results
}
function PostData ($URL, $Header, $Body)
{
    try
    {
        $result = Invoke-RestMethod $URL -Method Post -Headers $Header -Body $Body
    }
    catch
    {
        Write-host ("Error PostData " + $URL + " " +  $_.Exception.Message + " " +
$_.Exception.ItemName) -ForegroundColor Red
    }
```

```
      }
      function ChangeTicketStatuses ($URL, $Header)
      {
          $Collection = QueryData $URL $Header
          foreach ($Ticket in $Collection)
          {
              [datetime] $TicketDate = $Ticket.created
              [datetime] $CompareDate = Get-Date "9/13/2017 12:00 AM"
              if ($TicketDate -lt $CompareDate)
              {
                  Write-Host "Changing status of Ticket" $Ticket.id "to 3" -ForegroundColor Red
                  $Ticket

                  # Change Status to 3 (Closed)
                  $Body = '{"priority":null,"queue":null,"assigned_to":null,"comment":null,"ticket":[' +
$Ticket.id + '],"status":3}'
                  PostData ("https://api.app.flexerasoftware.com/api/tickets/edit/") $WebServiceHeader $Body
              }
              else
              {
                  Write-Host "Leaving Ticket" $Ticket.id "Alone" $Ticket.created -ForegroundColor Green
              }
          }

      }
      $WebServiceHeader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
      $WebServiceHeader.Add("Content-Type", 'application/json')
      $WebServiceHeader.Add("Authorization", "Token YOURTOKENHERE" )
      ChangeTicketStatuses  "https://api.app.flexerasoftware.com/api/tickets/" $WebServiceHeader
```

API

Following is the API information for options listed under **Settings > API**.

- XML Feed List

- XML Feed Request List

XML Feed List

For information on the XML Feed List APIs, see the following URL:

https://api.app.secunia.com/api/available-xml-feeds/

List of available XML Feed serializers.

Available Methods for XML Feed List

The following are available methods for XML Feed List.

Table 7-26 • Available Methods for XML Feed List

Method	Description
get list	GET <URL>
get instance details	GET <URL><id>/

XML Feed Request List

For information on the XML Feed Request List APIs, see the following URL:

https://api.app.secunia.com/api/xml-feed-requests/

This provides a list of XML Feed requests. It logs the dynamic requests to the XML Feeds to track changes since the last request.

Available Methods for XML Feed Request List

The following are available methods for XML Feed Request List.

Table 7-27 • Available Methods for XML Feed Request List

Method	Description
get list	GET <URL>
get instance details	GET <URL><id>/
load-all	GET <URL>/load-all/ *Note* • *Loads all the XML Feeds request made; the response is not paginated as a normal GET*

A

Appendix A - HTTP Status Codes

The status codes outlined below are taken from the Django REST framework API Guide Status Codes page, which you can access here.

- Informational - 1xx
- Successful - 2xx
- Redirection - 3xx
- Client Error - 4xx
- Server Error - 5xx
- Helper Functions

For more information on the proper usage of HTTP status codes, refer to RFC 2616 and RFC 6585.

Informational - 1xx

This class of status code indicates a provisional response. There are no 1xx status codes used in REST framework by default.

```
HTTP_100_CONTINUE
HTTP_101_SWITCHING_PROTOCOLS
```

Successful - 2xx

This class of status code indicates that the client's request was successfully received, understood, and accepted.

```
HTTP_200_OK
HTTP_201_CREATED
HTTP_202_ACCEPTED
HTTP_203_NON_AUTHORITATIVE_INFORMATION
HTTP_204_NO_CONTENT
HTTP_205_RESET_CONTENT
HTTP_206_PARTIAL_CONTENT
```

Redirection - 3xx

This class of status code indicates that further action needs to be taken by the user agent in order to fulfill the request.

```
HTTP_300_MULTIPLE_CHOICES
HTTP_301_MOVED_PERMANENTLY
HTTP_302_FOUND
HTTP_303_SEE_OTHER
HTTP_304_NOT_MODIFIED
HTTP_305_USE_PROXY
HTTP_306_RESERVED
HTTP_307_TEMPORARY_REDIRECT
```

Client Error - 4xx

The 4xx class of status code is intended for cases in which the client seems to have erred. Except when responding to a HEAD request, the server SHOULD include an entity containing an explanation of the error situation, and whether it is a temporary or permanent condition.

```
HTTP_400_BAD_REQUEST
HTTP_401_UNAUTHORIZED
HTTP_402_PAYMENT_REQUIRED
HTTP_403_FORBIDDEN
HTTP_404_NOT_FOUND
HTTP_405_METHOD_NOT_ALLOWED
HTTP_406_NOT_ACCEPTABLE
HTTP_407_PROXY_AUTHENTICATION_REQUIRED
HTTP_408_REQUEST_TIMEOUT
HTTP_409_CONFLICT
HTTP_410_GONE
HTTP_411_LENGTH_REQUIRED
HTTP_412_PRECONDITION_FAILED
HTTP_413_REQUEST_ENTITY_TOO_LARGE
HTTP_414_REQUEST_URI_TOO_LONG
HTTP_415_UNSUPPORTED_MEDIA_TYPE
HTTP_416_REQUESTED_RANGE_NOT_SATISFIABLE
HTTP_417_EXPECTATION_FAILED
HTTP_428_PRECONDITION_REQUIRED
HTTP_429_TOO_MANY_REQUESTS
HTTP_431_REQUEST_HEADER_FIELDS_TOO_LARGE
HTTP_451_UNAVAILABLE_FOR_LEGAL_REASONS
```

Server Error - 5xx

Response status codes beginning with the digit "5" indicate cases in which the server is aware that it has erred or is incapable of performing the request. Except when responding to a HEAD request, the server SHOULD include an entity containing an explanation of the error situation, and whether it is a temporary or permanent condition.

```
HTTP_500_INTERNAL_SERVER_ERROR
HTTP_501_NOT_IMPLEMENTED
HTTP_502_BAD_GATEWAY
HTTP_503_SERVICE_UNAVAILABLE
HTTP_504_GATEWAY_TIMEOUT
HTTP_505_HTTP_VERSION_NOT_SUPPORTED
HTTP_511_NETWORK_AUTHENTICATION_REQUIRED
```

Helper Functions

The following helper functions are available for identifying the category of the response code.

```
is_informational()  # 1xx
is_success()        # 2xx
is_redirect()       # 3xx
is_client_error()   # 4xx
is_server_error()   # 5xx
```