



# Software Vulnerability Research

API User Guide



# Legal Information

**Book Name:** Software Vulnerability Research API Guide  
**Part Number:** SVR-2019-API00  
**Product Release Date:** May 2019

## Copyright Notice

Copyright © 2019 Flexera

This publication contains proprietary and confidential information and creative works owned by Flexera and its licensors, if any. Any use, copying, publication, distribution, display, modification, or transmission of such publication in whole or in part in any form or by any means without the prior express written permission of Flexera is strictly prohibited. Except where expressly provided by Flexera in writing, possession of this publication shall not be construed to confer any license or rights under any Flexera intellectual property rights, whether by estoppel, implication, or otherwise.

All copies of the technology and related information, if allowed by Flexera, must display this notice of copyright and ownership in full.

## Intellectual Property

For a list of trademarks and patents that are owned by Flexera, see <https://www.flexera.com/producer/company/about/intellectual-property/>. All other brand and product names mentioned in Flexera products, product documentation, and marketing materials are the trademarks and registered trademarks of their respective owners.

## Restricted Rights Legend

The Software is commercial computer software. If the user or licensee of the Software is an agency, department, or other entity of the United States Government, the use, duplication, reproduction, release, modification, disclosure, or transfer of the Software, or any related documentation of any kind, including technical data and manuals, is restricted by a license agreement or by the terms of this Agreement in accordance with Federal Acquisition Regulation 12.212 for civilian purposes and Defense Federal Acquisition Regulation Supplement 227.7202 for military purposes. The Software was developed fully at private expense. All other use is prohibited.

# Contents

- 1 Software Vulnerability Research API Help Library ..... 7**
  - Using Help ..... 8
  - Contact Us..... 8
- 2 API Introduction..... 9**
  - API Explorer ..... 9
  - API Menu Options..... 10
  - Tokens ..... 11
  - Examples - Calling the API ..... 12
    - Getting a list of first 20 advisories (first page): ..... 12
    - Getting a custom list of advisories: ..... 13
    - Getting a specific advisory by integer id (not guaranteed to be consecutive): ..... 13
  - Using Windows PowerShell ..... 13
  - API Notes..... 14
    - API Versions and Parsing ..... 14
    - API Throttling..... 14
    - CVSSv3 Score ..... 15
  - XML Feeds ..... 15
  - External API Services (Service Providers) ..... 16
    - Integration with the External API Service Provider ..... 16
      - Service Provider Fields ..... 16
      - Service Provider Methods..... 17
      - Service Provider Test Connection..... 20
      - Create Rules to Call the Service Provider ..... 21
- 3 Vulnerability Manager Module API Information ..... 23**
  - API Supported Endpoint Actions and Available Methods for Vulnerability Manager APIs..... 23
    - Watch List Advisory List ..... 23

Available Methods for Watch List Advisory List: .....	24
Available Filters on Watch List Advisory List: .....	24
Approve Method for Watch List Advisory List: .....	24
Dismiss Method for Watch List Advisory List: .....	24
Watch List Group List .....	25
Available Methods for Watch List Group List: .....	25
Available Filters on Watch List Group List: .....	25
Watch List Group List Fields for Create/Edit: .....	25
Watch List List .....	25
Available Methods for Watch List List: .....	26
Available Filters on Watch List List: .....	26
Watch List List Fields for Create/Edit: .....	26
Watch List List Threshold choices: .....	27
Watch List Changes .....	27
Available Filters for Watch List Changes: .....	27
PowerShell Script to Download Watch Lists to a CSV File .....	28
<b>4 Research Module API Information .....</b>	<b>29</b>
PowerShell Script to Pull Advisory Information .....	29
PowerShell Script to List All Devices and Their System Scores .....	31
PowerShell Script to Save All Advisories within a Date Range to CSV .....	32
PowerShell Script to Query Historic Advisories by Product and Version .....	34
<b>5 Assessment Module API Information .....</b>	<b>39</b>
API Supported Endpoint Actions and Available Methods for Assessment APIs .....	39
Device Groups .....	40
Available Methods for Device Groups: .....	40
Available Filters on Device Groups List: .....	40
Devices .....	41
Available Methods for Devices: .....	41
Available Filters on Devices List: .....	41
Overview of the Major Product Versions Detected on Devices .....	42
Available Methods for Overview of the Major Product Versions Detected on Devices: .....	42
Available Filters on Overview of the Major Product Versions Detected on the Devices List: .....	42
Major Product Versions Detected on Devices for Device Groups .....	43
Available Methods for Major Product Versions Detected on Devices for Device Groups: .....	43
Available Filters on Major Product Versions Detected on Devices for the Device Groups List: .....	43
Advisories Detected on Devices for Device Groups .....	43
Available Methods for Advisories Detected on Devices for Device Groups: .....	44
Available Filters on Advisories Detected on Devices for the Device Groups List: .....	44
Advisories Detected on Devices .....	45
Available Methods for Advisories Detected on Devices: .....	46
Available Filters on Advisories Detected on Devices List: .....	46
PowerShell Script to Look at Device Data .....	47
PowerShell Script to Look at Product Data .....	48

PowerShell Script to Look at Hosts and Their Advisories Since a Specific Date .....	49
Query Assessment Data Based on Smart Groups .....	51

## 6 Patching Module API Information ..... 53

### API Supported Endpoint Actions and Available Methods for Patching APIs ..... 53

Daemon Lists .....	54
Available Methods for Daemon Lists: .....	54
Available Filters on Daemon Lists: .....	54
Server Details .....	54
Available Methods for Server Details: .....	54
Available Filters on Server Detail Lists: .....	55
Server Group Details .....	55
Available Methods for Server Group Details: .....	55
Available Filters on Server Group Detail Lists: .....	55
Customer Patch Template Name Details .....	55
Available Methods for Customer Patch Template Name Details: .....	55
Available Filters on Customer Patch Template Name Detail Lists: .....	55
Customer Patch Template Created by Details .....	56
Available Methods for Customer Patch Template Created by Details: .....	56
Available Filters on Customer Patch Template Created by Lists: .....	56
Patchable Product Details .....	56
Available Methods for Patchable Product Details: .....	57
Available Filters on Patchable Product Lists: .....	57
Patch Package Details .....	57
Available Methods for Patch Package Details: .....	57
Available Filters on Patch Package Lists: .....	58
Customer's Patch Package Publishing Details .....	59
Available Methods for Customer's Patch Package Publishing Details: .....	59
Available Filters on Customer's Patch Package Publishing Lists: .....	59
Patch Tasks .....	60
Available Methods for Patch Task Details: .....	60
Available Filters on Patch Task Lists: .....	60
Patches Available .....	61
Available Methods for Patches Available: .....	61
Available Filters on Patches Available Lists: .....	61
Available Patches Grouped .....	61
Available Methods for Available Patches Grouped: .....	61
Available Filters on Available Patches Grouped Lists: .....	61
Patch Language .....	62
Available Methods for Patch Language: .....	62
Available Filters on Patch Language Lists: .....	62
Publish Patch List .....	62
Patch Package List .....	63
Product Release Instance .....	63
PowerShell Script to Delete Data .....	63

<b>7 Settings Module API Information .....</b>	<b>67</b>
<b>API Supported Endpoint Actions and Available Methods for Settings APIs .....</b>	<b>67</b>
<b>User Management .....</b>	<b>67</b>
Authenticated User List .....	68
Available Methods for Authenticated User List: .....	68
Authenticated User List Fields for Create/Edit: .....	68
User Group List .....	69
Available Methods for User Group List: .....	69
User Group Fields for Create/Edit: .....	69
User Logins .....	69
Available Filters for User Logins: .....	69
Email Logs .....	70
Available Filters for Email Logs: .....	70
SMS Logs .....	70
Available Filters for SMS Logs: .....	70
Group List (Roles) .....	70
Available Methods for Group List: .....	71
<b>Workflow Management .....</b>	<b>71</b>
Ticket List .....	71
Available Methods for Ticket List: .....	71
Available Filters on Ticket List: .....	71
Create Method Fields for Ticket Lists: .....	72
Edit Method Fields for Ticket Lists: .....	72
Ticket Queue List .....	73
Available Methods for Ticket Queue List: .....	73
Available Filters on Ticket Queue List: .....	73
Ticket Queue List Fields for Create/Edit: .....	73
Ticket Status List .....	73
Available Methods for Ticket Status List: .....	74
Available Filters on Ticket Status List: .....	74
Ticket Status List Fields for Create/Edit: .....	74
Ticket Priority List .....	74
Available Methods for Ticket Priority List: .....	75
Available Filters on Ticket Priority List: .....	75
Ticket Priority List Fields for Create/Edit: .....	75
Ticket Changes .....	75
Available Filters for Ticket Changes: .....	75
Ticket Note List .....	76
Available Methods for Ticket Note List: .....	76
Available Filters on Ticket Note List: .....	76
Ticket Note List Fields for Create/Edit: .....	76
PowerShell Script to Close Tickets Using a Certain Date .....	76
<b>API .....</b>	<b>78</b>
XML Feed List .....	78
Available Methods for XML Feed List: .....	78
XML Feed Request List .....	78

Available Methods for XML Feed Request List:..... 79

**A   Appendix A - HTTP Status Codes ..... 81**

    Informational - 1xx ..... 81

    Successful - 2xx ..... 81

    Redirection - 3xx ..... 82

    Client Error - 4xx..... 82

    Server Error - 5xx ..... 82

    Helper functions ..... 83









# Software Vulnerability Research API Help Library

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

**Table 1-1** • Software Vulnerability Research API Help Library

Topic	Content
<b>API Introduction</b>	This section describes how to access the API information.
<b>Vulnerability Manager Module API Information</b>	<p>This section provides Vulnerability Manager module API information.</p>  <p><b>Edition</b> • The Vulnerability Manager module is not available for Software Vulnerability Research - Assessment Only.</p>
<b>Research Module API Information</b>	<p>This section provides Research module API information.</p>  <p><b>Edition</b> • The Research module is not available for Software Vulnerability Research - Assessment Only.</p>
<b>Assessment Module API Information</b>	<p>This section provides Assessment module API information.</p>  <p><b>Edition</b> • The Assessment module is not available for Software Vulnerability Research.</p>
<b>Patching Module API Information</b>	<p>This section provides Patching module API information.</p>  <p><b>Edition</b> • The Patching module is not available for Software Vulnerability Research.</p>

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

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

## Using Help

Help is available from the Software Vulnerability Manager interface help icon located at the top right of the screen.

### Online Help

For online help, see <https://helpnet.flexerasoftware.com/svm/Default.htm>

### Release Notes

For the latest release notes, see <https://helpnet.flexerasoftware.com/?product=Software%20Vulnerability%20Manager&version=Current>

For earlier release notes, see <https://helpnet.flexerasoftware.com/?product=Software%20Vulnerability%20Manager&version=Previous>

## 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:

<https://www.flexera.com/>

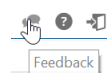
### Product Feedback

Have a suggestion for how we can improve this product? Please come share direct feedback with the product team and vote on ideas submitted by other users in our [Customer Community feedback page for Software Vulnerability Research](#)



**Note •** You will need your Flexera Customer Community credentials to enter feedback.

You can also submit feedback through the Software Vulnerability Research user interface by clicking the feedback icon in the upper-right-hand corner of each module.



# 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](#)
- [XML Feeds](#)
- [External API Services \(Service Providers\)](#)

### 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.

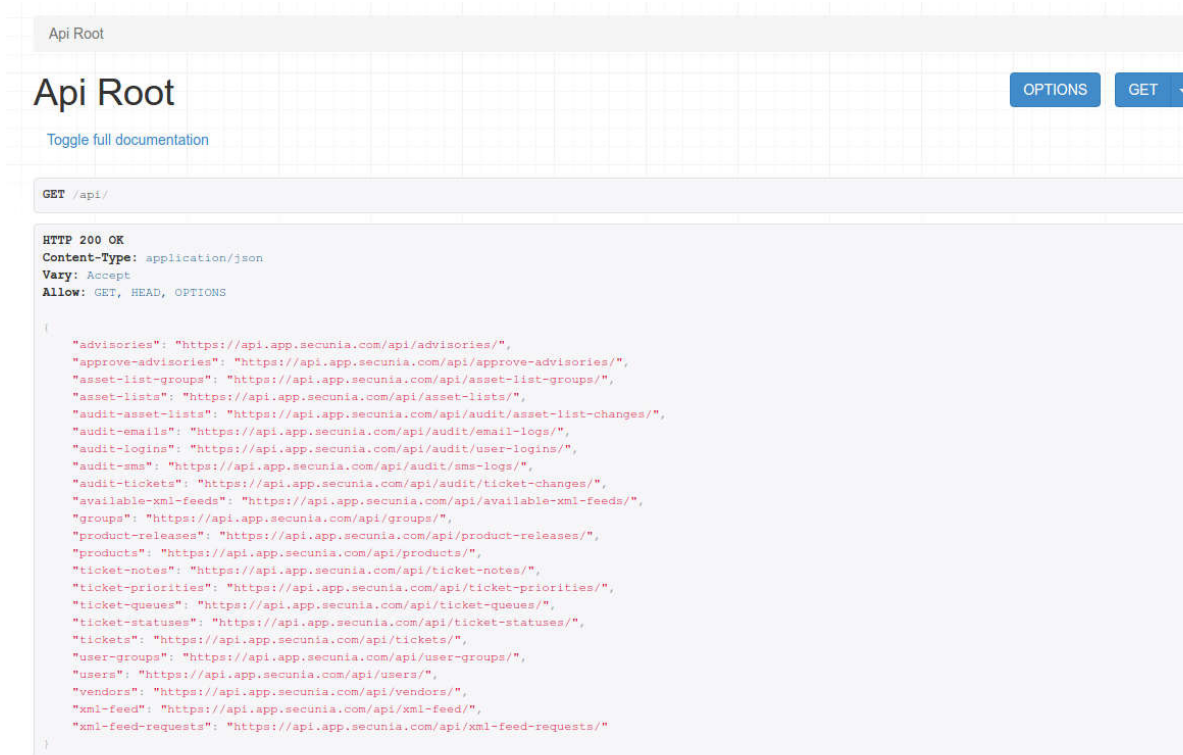


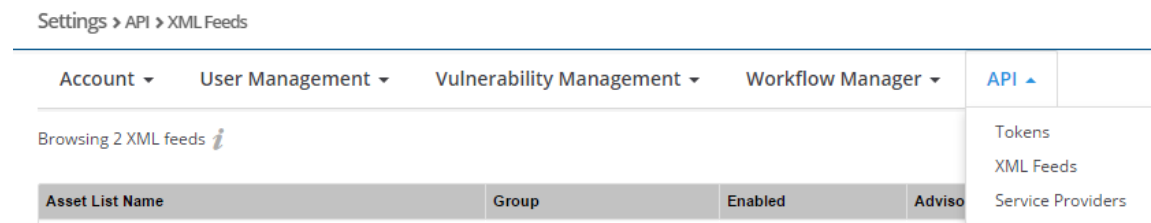
Figure 2-1: API Explorer Page Example

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:

- **GET** - for read

- **POST** - for create
- **PATCH / PUT** - for update
- **DELETE** - for delete tokens

## 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.

Settings > API > Tokens

Account ▾ User Management ▾ Vulnerability Management ▾ Workflow Management ▾ Assessment ▾ API ▾ Logs ▾

API Access token generation page ⓘ

User	Token	Created
	4344d5...	2018-03-06 04:17:19

Figure 2-2: Truncated Token

Settings > API > Tokens

Account ▾ User Management ▾ Vulnerability Management ▾

API Access token generation page ⓘ

User	Token
	4344d541c3fa38fe4359310045a37eb071f61afc

Figure 2-3: Expanded Token

3. Click a Token in the grid to delete the Token.

Settings > API > Tokens


Account ▾ User Management ▾ Vulnerability Management ▾ Workflow Management ▾ Assessment ▾ API ▾ Logs ▾

API Access token generation page ⓘ

User	Token	Created
	4344d5...	2018-03-06 04:17:19

Delete

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 programmatically.

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 (first page):
- Getting a custom list of advisories:
- Getting a specific advisory by integer id (not guaranteed to be consecutive):

### Getting a list of 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:

```
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 (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](#)
- [CVSSv3 Score](#)

## 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

## 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](#).

## 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 an **Watch List Name** to view the Watch List.

Account ▾ User Management ▾ Vulnerability Management ▾ Workflow Manager ▾ <b>API ▾</b>					
Browsing 3 XML feeds <a href="#">?</a>					
Asset List Name	Group	Enabled	Advisories need approval	Last requested	XML
My_Asset_List.csv		Yes	Yes		<a href="#">Dynamic</a>   <a href="#">24-Hour</a>   <a href="#">48-Hour</a>   <a href="#">72-Hour</a>
<a href="#">View asset list</a>					
My_Asset_List.csv - Cloned		Yes	Yes		<a href="#">Dynamic</a>   <a href="#">24-Hour</a>   <a href="#">48-Hour</a>   <a href="#">72-Hour</a>
SCIM Import.csv		Yes	Yes		<a href="#">Dynamic</a>   <a href="#">24-Hour</a>   <a href="#">48-Hour</a>   <a href="#">72-Hour</a>

**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:

type:

- Custom: custom API defined by the customer



**Note** • This is not tested by Flexera.

- ServiceNow: ServiceNow specific calls, a REST recipe is offered
- BMC Remedy: BMC Remedy calls, a SOAP recipe is offered, with basic authentication

**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:

- REST
- SOAP

authentication type:

- 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:

**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.

**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:

#\$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

Edit provider - ServiceNow

Type

ServiceNow

Name

ServiceNow

Url

https://dev009.service-now.com/

Protocol Type

REST

Authentication Type

Basic authentication

Username

admin

Password

\*\*\*\*\*

Header	Value	Action
Header	Value	ADD +

Cancel

Save

Figure 2-7: ServiceNow Service Provider Example

Browsing 3 methods for provider **Service Now**

Name	Method	Url	Retrieve entity id expression
Incident create	POST	/api/now/table/incident	result.sys_id
Incident get	GET	/api/now/table/incident/#\$ref_object_id#\$	result.sys_id
Incident edit	PUT	/api/now/table/incident/#\$ref_object_id#\$	result.sys_id

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 expression
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**.

Test service method Incident create - POST for provider ServiceNow

Partial Url

/api/now/table/incident

HTTP Method

POST

Header	Value	Action
Header	Value	ADD +

Query Param	Value	Action
sysparm_limit	10	DELETE X
Query Param	Value	ADD +

Content (JSON dictionary)

{  
 "short\_description": "[FLEXERA SOFTWARE] Advisory #\${advisory.advisory\_identifier}# for asset list #\${asset\_list.name}# was released.",  
 "description": "Advisory #\${advisory.advisory\_identifier}# for asset list #\${asset\_list.name}# was released. Ticket #\${ticket.pretty\_id}#`#\${advisory.title}#`  
 affects products: #\${advisory.products.name}#`  
}

Retrieve entity id expression

result.sys\_id

Response was successful: True

Response code: 201

Ref object id: 33396b7f130dee004e5050f32244b0b5

Response content:

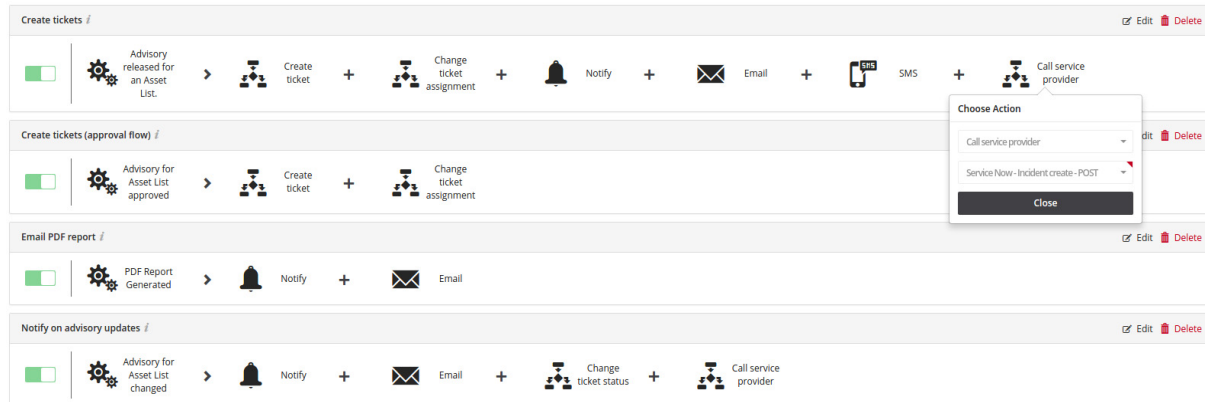
```
{  
  "result": {  
    "location": "",  
    "state": "1",  
    "delivery_task": "",  
    "delivery_plan": "",  
    "comments": "",  
    "correlation_display": "",  
    "sys_id": "33396b7f130dee004e5050f32244b0b5",  
    "problem_id": "",  
    "notify": "1",  
    "reopen_count": "0",  
    "close_code": ""  
  }  
}
```

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.



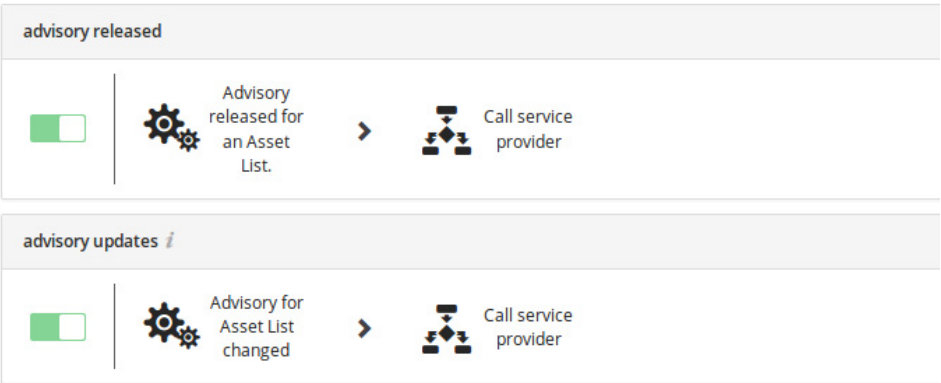
### Task

#### 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.



Rules without Tickets Example



# Vulnerability Manager Module API Information



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

This section describes the [API Supported Endpoint Actions and Available Methods for Vulnerability Manager APIs](#).

## API Supported Endpoint Actions and Available Methods for Vulnerability Manager APIs



**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

<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:

- 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:

- watch\_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)
- 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

<https://api.app.secunia.com/api/Watch-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:

- 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:

- name (string) - invariant case search by term in name

### Watch List Group List Fields for Create/Edit:

- name (string) - the group name visible in the interface

## Watch List List

<https://api.app.secunia.com/api/Watch-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:

- get list - GET <URL>
- 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:

- 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

## Watch List List Fields for Create/Edit:

- 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 - see below for choices) - 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).

- notification\_level\_email (int - see below for choices) - 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".
- notification\_level\_sms (int - see below for choices) - 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.

## Watch List List Threshold choices:

- 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 - see below for choices) - "Confidentiality Requirement"
- "custom\_ir" (string - see below for choices) - "Integrity Requirement"
- "custom\_ar" (string - see below for choices) - "Availability Requirement"

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 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

<https://api.app.secunia.com/api/audit/Watch-list-changes/>

List of Watch List changes.

Following are the [Available Filters for Watch List Changes](#):

## Available Filters for Watch List Changes:

- 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)
- 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
```

# 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 + " " +
$_.Exception.ItemName) -ForegroundColor Red
}
#Get the next pages of results, if any
while (![string]::IsNullOrEmpty($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 $Header
    foreach ($Advisory in $Collection)
    {
        #Advisory
        $advisoryDetails = QueryData ("https://api.app.flexerasoftware.com/api/advisories/" +
$Advisory.id + "/") $Header
        $advisoryDetails

        #Remove this and it will loop over the first $global:QueryLimit advisories 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

```

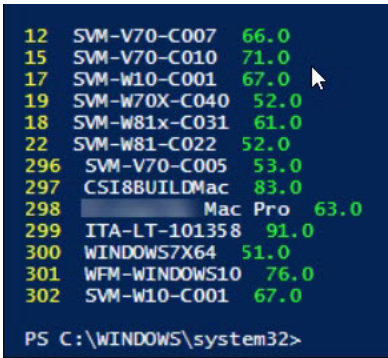


# PowerShell Script to List All Devices and Their System Scores

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:



```
12 SVM-V70-C007 66.0
15 SVM-V70-C010 71.0
17 SVM-W10-C001 67.0
19 SVM-W70X-C040 52.0
18 SVM-W81x-C031 61.0
22 SVM-W81-C022 52.0
296 SVM-V70-C005 53.0
297 CSI8BUILDMac 83.0
298 Mac Pro 63.0
299 ITA-LT-101358 91.0
300 WINDOWS7X64 51.0
301 WFM-WINDOWS10 76.0
302 SVM-W10-C001 67.0

PS C:\WINDOWS\system32>
```

# 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 advisories 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
{
}
}
catch
{
Write-host ("Error QueryData1 " + $URL + " " + $_.Exception.Message + " " + $_.Exception.ItemName) -
ForegroundColor Red
}
#Get the next pages of results, if any
while (![string]::IsNullOrEmpty($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
$CustomCollection = @()
foreach ($Advisory in $Collection)
{
#Advisory
$advisoryDetails = QueryData ("https://api.app.secunia.com/api/advisories/" + $Advisory.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

```

## PowerShell Script to Query Historic Advisories by Product and Version

34

```

    }
    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]::IsNullOrEmpty($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

```





# Assessment Module API Information



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

This section describes the [API Supported Endpoint Actions and Available Methods for Assessment APIs](#).

## API Supported Endpoint Actions and Available Methods for Assessment APIs



**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:

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

### Available Filters on Device Groups List:

- id (int) - exact match on the id of Device group
- name (string) - name of the Device group
- path (string) - path for Device group
- source (int) - SOURCE TYPES:
  - 0 - Active Directory
  - 1 - Smart Group
- 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: 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:

- get list - GET <URL>
- get instance details - GET <URL><id>/

### Available Filters on Devices List:

- 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
- `last_scan_status` (int) - Last scan status for Device

Example: 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

API Supported Endpoint Actions and Available Methods for Overview of the Major Product Versions Detected on the Device APIs include:

- [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:

- `get list` - GET <URL>

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

- `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
- `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: filters out a product that is end of life

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

## Major Product Versions Detected on Devices for Device Groups

API Supported Endpoint Actions and Available Methods for Major Product Versions Detected on the Devices for Device Group APIs include:

- [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:

- get list - GET <URL>

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

- 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) - Product is secure or not
- is\_eol(bool) - Filters product that is end of life or not
- is\_secure (bool) - Filters product that is secure or not
- 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: filters out a product that is end of life

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

## Advisories Detected on Devices for Device Groups

API Supported Endpoint Actions and Available Methods for Advisories Detected on the Devices for Device Group APIs include:

- [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:

- get list - GET <URL>
- get advisory details GET <URL><id / advisory\_identifier>/

Examples: /api/inventory/advisories-stats/178453/ or /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:

- 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
- 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
- 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: 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

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

- [Available Methods for Advisories Detected on Devices:](#)
- [Available Filters on Advisories Detected on Devices for the Device Groups List:](#)

## Available Methods for Advisories Detected on Devices:

- get list - GET <URL>
- get advisory details GET <URL><id / advisory\_identifier>/

Examples: /api/inventory/advisories/178453/ or /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:

- identifier (string) - exact match on the advisory main identifier (Example: 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
- 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
- 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: 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: 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: 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: 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 + " "
+ $_.Exception.ItemName)
}
#Get the next pages of results, if any
while (![string]::IsNullOrEmpty($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



### Task

#### To query assessment data based on Smart Groups

1. Use the following URL: [https://app.flexerasoftware.com/api/inventory/host-groups/top\\_custom/](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/>

# Patching Module API Information



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

This section describes the [API Supported Endpoint Actions and Available Methods for Patching APIs](#).

## API Supported Endpoint Actions and Available Methods for Patching APIs



**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

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

- [Available Methods for Daemon Lists:](#)
- [Available Filters on Daemon Lists:](#)

### Available Methods for Daemon Lists:

- `get list - GET <URL>`

### Available Filters on Daemon Lists:

- `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: 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

API Supported Endpoint Actions and Available Methods for Server Detail APIs include:

- [Available Methods for Server Details:](#)
- [Available Filters on Server Detail Lists:](#)

### Available Methods for Server Details:

- `get list - GET <URL>`



## Available Filters on Server Detail Lists:

- name (string) - name of the server
- external\_id (string) - external id of the server

Example: filters out a server whose name is “my-server”

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

## Server Group Details

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

- [Available Methods for Server Group Details:](#)
- [Available Filters on Server Group Detail Lists:](#)

## Available Methods for Server Group Details:

- get list - GET <URL>

## Available Filters on Server Group Detail Lists:

- 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: filters out a server group whose name is “my-server”

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

## Customer Patch Template Name Details

API Supported Endpoint Actions and Available Methods for Customer Patch Template Name APIs include:

- [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:

- get list - GET <URL>
- create instance - POST <URL>

## Available Filters on Customer Patch Template Name Detail Lists:

- name (string) - name of patch template

- `has_customer_template` (bool) - Patch has customer template or not

Example: filters out patch templates whose name is “xyz”

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

## Customer Patch Template Created by Details

API Supported Endpoint Actions and Available Methods for Customer Patch Template Created by APIs include:

- [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:

- get list - GET <URL>
- create instance - POST <URL>

### Available Filters on Customer Patch Template Created by Lists:

- `name` (string) - name of customer 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:
  - 0 - 32-bit/64-bit
  - 1 - 32-bit
  - 2 - 64-bit
- `for_languages` (list) - languages `iso_code`, `language_display`
- `product_id` (int) - product identifier for patch template
- `edition` (string) - edition for patch template

Example: filters out patch templates created by user id 3

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

## Patchable Product Details

API Supported Endpoint Actions and Available Methods for Patchable Product APIs include:

- [Available Methods for Patchable Product Details:](#)
- [Available Filters on Patchable Product Lists:](#)

## Available Methods for Patchable Product Details:

- get list - GET <URL>

## Available Filters on Patchable Product Lists:

- 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
- 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

Example: filters out patchable product whose product name id "java"

api/patch/customer-patch-templates/?product\_name=java

## Patch Package Details

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

- [Available Methods for Patch Package Details:](#)
- [Available Filters on Patch Package Lists:](#)

## Available Methods for Patch Package Details:

- get list - GET <URL>
- get package details GET <URL><id>

## Available Filters on Patch Package Lists:

- 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
- 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

Example: filters package whose customer patch template identifier is 1

api/patch/packages/?customer\_patch\_template\_id=1

## Customer's Patch Package Publishing Details

API Supported Endpoint Actions and Available Methods for Customer's Patch Package Publishing APIs include:

- [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:

- get list - GET <URL>

### Available Filters on Customer's Patch Package Publishing Lists:

- 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: filters out publish instance, where package vendor name is java

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

## Patch Tasks

API Supported Endpoint Actions and Available Methods for Patch Task APIs include:

- [Available Methods for Patch Task Details:](#)
- [Available Filters on Patch Task Lists:](#)

### Available Methods for Patch Task Details:

- get list - GET <URL>

### Available Filters on Patch Task Lists:

- daemon\_id (int) - Daemon ID
- publish\_id (int) - Publish ID
- type (int) - Task Type:
  - 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
- 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: filters out a task whose publish id is “1234”

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

## Patches Available

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

- [Available Methods for Patches Available:](#)
- [Available Filters on Patches Available Lists:](#)

### Available Methods for Patches Available:

- get list - GET <URL>
- get instance details - GET <URL><id>/

### Available Filters on Patches Available Lists:

- `product_release_id` (int) - release id of a product

Example: 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/>

API Supported Endpoint Actions and Available Methods for Available Patches Grouped APIs include:

- [Available Methods for Available Patches Grouped:](#)
- [Available Filters on Available Patches Grouped Lists:](#)

### Available Methods for Available Patches Grouped:

- get list - GET <URL> Example: GET `/api/patch/available-patches-grouped/`
- get instance details - GET <URL><id>/

### Available Filters on Available Patches Grouped Lists:

- `product_release_id` (int) - Product release identifier

- 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: filters out a product whose product release id is “111”

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

## Patch Language

API Supported Endpoint Actions and Available Methods for Patch Language APIs include:

- [Available Methods for Patch Language:](#)
- [Available Filters on Patch Language Lists:](#)

### Available Methods for Patch Language:

- get list - GET <URL>

### Available Filters on Patch Language Lists:

- iso\_code (string) - ISO code for language
- language\_display (string) - language for display

Example: 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: 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: 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: 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 YOURTOKEN')
$global:WebServiceURLSecunia = "https://api.app.flexerasoftware.com/api/"
function Write-Message ($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 + " " + $_Exception.ItemName)  
}  
#Get the next pages of results, if any  
while (![string]::IsNullOrEmpty($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
```

# Settings Module API Information

This section describes the [API Supported Endpoint Actions and Available Methods for Settings APIs](#).

## API Supported Endpoint Actions and Available Methods for Settings APIs



**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](#)

## Authenticated User List

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

List of users for your account.

List of 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:

- 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:

- 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.
- 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

<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:

- 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:

- 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

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

List of user logins.

Below are the [Available Filters for User Logins](#):

## Available Filters for User Logins:

- 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)
- ref (guid) - "next" value from a paginated response

## Email Logs

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

List of emails sent to your users.

Below are the [Available Filters for Email Logs](#):

### Available Filters for Email Logs:

- 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)
- ref (guid) - "next" value from a paginated response

## SMS Logs

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

List of SMS sent to your users.

Below are the [Available Filters for SMS Logs](#):

### Available Filters for SMS Logs:

- 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)
- ref (guid) - "next" value from a paginated response

## Group List (Roles)

<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.

Below are the [Available Methods for Group List](#):



## Available Methods for Group List:

- 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

<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:

- get list - GET <URL>
- create instance(s) - POST <URL> - can create multiple tickets, one per advisory - Watch list pair
- edit instance(s) - POST <URL>edit/ - can edit multiple tickets

## Available Filters on Ticket List:

- 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)
- `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:

- `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 exist, the value for that ticket doesn't change):

- `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

<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:

- 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 Queue List:

- name (string) - invariant case search by term in name

## Ticket Queue List Fields for Create/Edit:

- 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

<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:

- 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:

- name (string) - invariant case search by term in name

## Ticket Status List Fields for Create/Edit:

- 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

<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:

- 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:

- name (string) - invariant case search by term in name

## Ticket Priority List Fields for Create/Edit:

- name (string) - the group name visible in the interface
- default ticket priority (int) - the default ticket priority in our system
  - 0 = Low
  - 1 = Medium
  - 2 = High
  - 3 = Urgent

## Ticket Changes

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

List of ticket changes.

Below are the [Available Filters for Ticket Changes](#):

## Available Filters for Ticket Changes:

- 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)
- ref (guid) - "next" value from a paginated response
- object\_id (int) - the ticket id for which the changes were made

## Ticket Note List

<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:

- 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:

- ticket\_id (int) - the parent ticket id

### Ticket Note List Fields for Create/Edit:

- 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 + " " +
        $_.Exception.ItemName) -ForegroundColor Red
    }
    #Get the next pages of results, if any
    while (![string]::IsNullOrEmpty($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

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

List of available XML Feed serializers.

Below are the [Available Methods for XML Feed List](#):

### Available Methods for XML Feed List:

- get list - GET <URL>
- get instance details - GET <URL><id>/

## XML Feed Request List

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

List of XML Feed requests.

Logs the dynamic requests to the XML Feeds to track changes since the last request.

Below are the [Available Methods for XML Feed Request List](#):



## Available Methods for XML Feed Request List:

- get list - GET <URL>
- get instance details - GET <URL><id>/
- load-all - GET <URL>/load-all/ - loads all the XML Feeds request made, the response is not paginated as a normal GET





# 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
```

