

HP EnterpriseView

For Windows

Software Version: 1.5

REST API Developer Guide

Document Release Date: October 2012

Software Release Date: October 2012



Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

The network information used in the examples in this document (including IP addresses and host names) is for illustration purposes only.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 2012 Hewlett-Packard Development Company, L.P.

Follow this link to see a complete statement of copyrights and acknowledgements for all ArcSight products: <http://www.arcsight.com/copyrightnotice>.

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

This document is confidential.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

<http://h20230.www2.hp.com/selfsolve/manuals>

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

<http://h20229.www2.hp.com/passport-registration.html>

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at:

<http://www.hp.com/go/hpsoftwaresupport>

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

<http://h20229.www2.hp.com/passport-registration.html>

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

Contents

REST API Overview	6
HTTP Return Codes	8
Error Messages	9
Example Application	11
Application Context	11
HTTP Client Factory Bean	12
Example	12
Authenticate	14
Resource References	15
Asset Reconciliation	15
Compliance Score	17
P5 Control Maturity Factor Score (Single)	18
P5 Control Maturity Factors Scores (Multiple)	19
Notes	20
Aggregated Risk Score	22
Aggregated Vulnerability Score	22
Aggregated ESM Threat Score	23
Aggregated Compliance Score	24
Aggregated Control Maturity Score	25

REST API Overview

Using the EnterpriseView REST API you can import and export asset data in the following ways:

- Importing assessments on assets from any external security system or tool, eliminating the need to manually assess assets in EnterpriseView.
- Exporting assessments on assets to any external security system or tool.

Note: You can import/export assessment information only for assets of type **IP**; this assessment information is aggregated to top-level assets as in manual assessments

- Exporting asset aggregated scores to any external security system or tool, including:
 - The aggregated risk score of the asset
 - The aggregated compliance score of the asset
 - The aggregated control maturity score of the asset
 - The aggregated asset vulnerability score of the asset
 - The aggregated ESM threat score of the asset

Prerequisites:

Make sure that the following entities and parameters are identical in both EnterpriseView and the Security Automation tool:

- Assets of type IP
- Policy names
- Control paragraph numbers

The following HTTP request methods for importing and exporting asset data are available:

- **GET:** Reads one of the following resources from the EnterpriseView server:
 - Asset ID
 - P5 control maturity scores for multiple (1-5) factors
 - Compliance score of an asset with a control
 - Note on a control that is applied to an asset
 - The aggregated scores of an asset
- **PUT:** Updates/creates the following resources on the EnterpriseView server:
 - P5 control maturity score for a single factor
 - P5 control maturity scores for multiple (1-5) factors

- Compliance score of an asset with a control
- **POST:** Inserts notes on a control that is applied to an asset on the EnterpriseView server.

The first step of a client application is authenticating the user, as described in ["Authenticate" on page 14](#). After a client is authenticated, it can work with EnterpriseView resources.

HTTP Return Codes

Unless otherwise specified, these HTTP return codes are used:

Code	Cause
200	Successful operations
204	No content
400	Bad request
401	Unauthorized
403	Unauthorized operations
404	Resource not found
409	Conflict
500	Internal server error

Error Messages

These are the application error messages returned by the REST API.

Code	Exception	Can happen when
400	Cannot execute your request. Parameters are missing.	Attempting to retrieve an asset ID by the asset's IP address, MAC address or DNS name. Parameters are not passed.
404	Cannot find asset IP Address: {IP_Address} or Cannot find asset MAC Address: {MAC_Address} or Cannot find asset DNS Name: {DNS_Name}	Attempting to retrieve an asset ID by the asset's IP address, MAC address or DNS name.
	Cannot find policy: {Policy_Name}	Attempting to update or retrieve P5 score/compliance score/note for a specific control.
409	Cannot find control: {Control_Paragraph_Number}	Attempting to retrieve an asset ID by the asset's IP address, MAC address or DNS name.

Code	Exception	Can happen when
500	Cannot find asset: {Asset_ID}	Attempting to input a compliance score that is out of range.
	More than one asset matches the request. You can add additional parameters to filter the results.	Attempting to input a P5 control maturity factor score that is out of range.
	Score is out of range. Range should be between 0-100.	Attempting to input a P5 control maturity factor score or a compliance score that is not numeric or "NotAssessed".
	Score is out of range. Range should be between 0-5.	Attempting to input a P5 control maturity factor score or a compliance score that is not an integer.
	Score value must be either numeric or "NotAssessed".	Attempting to input a P5 control maturity factor score or a compliance score for a control that is not applied to an asset.
	Score value must be an integer.	Attempting to input one or more P5 control maturity factor that are not applied to the control.
	Control {Control_Paragraph_Number} is not applied to asset {Asset_ID}. Scores cannot be updated.	Attempting to input P5 control maturity factor scores or a compliance score for a control that is blocked for automatic assessments.

Example Application

The following examples are based on Spring framework version 3.0.

Application Context

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:context="http://www.springframework.org/schema/context"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-
3.0.xsd
http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context-3.0.xsd">

    <context:annotation-config/>

    <context:component-scan base-package="com.hp.redcat.restsample"/>

    <bean id="credentials"
class="org.apache.commons.httpclient.UsernamePasswordCredentials">
        <constructor-arg value="admin"/>
        <constructor-arg value="admin"/>
    </bean>

    <bean id="secureHttpClient" factory-bean="httpClientFactoryBean"
factory-method="getHttpClient"/>

    <bean id="httpClientFactory"
class="org.springframework.http.client.CommonsClientHttpRequestFactory">
        <constructor-arg ref="secureHttpClient"/>
    </bean>

    <bean id="restTemplate"
class="org.springframework.web.client.RestTemplate">
        <constructor-arg ref="httpClientFactory"/>
    </bean>

</beans>
```

HTTP Client Factory Bean

```
package com.hp.redcat.restsample;

import org.apache.commons.httpclient.Credentials;
import org.apache.commons.httpclient.HttpClient;
import org.apache.commons.httpclient.auth.AuthScope;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;

@Component
public class HttpClientFactoryBean {

    private HttpClient httpClient;

    @Autowired
    public HttpClientFactoryBean(Credentials credentials) {
        this.httpClient = new HttpClient();
        this.httpClient.getState().setCredentials(AuthScope.ANY,
credentials);
    }

    public HttpClient getHttpClient() {
        return httpClient;
    }
}
```

Example

```
package com.hp.redcat.restsample;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;
import org.springframework.web.client.RestTemplate;

import java.util.HashMap;
import java.util.Map;

@Component
public class RestSample {

    @Autowired
    private RestTemplate restTemplate;

    /**
     * @param assetIdInEnterpriseView is the asset ID in EnterpriseView.
     * Follow the instructions in "Asset Reconciliation" on page 15 to
     get it.
     */
}
```

```
*  
**/  
  
public void putP5Assessment(String assetIdInEnterpriseView) {  
    Map<String, String> variables = new HashMap<String, String>();  
    variables.put("assetId", assetIdInEnterpriseView);  
    variables.put("policyKey", "Cobit 4.1");  
    variables.put("controlParagraph", "PO1.1");  
    variables.put("p5type", "People");  
  
    final String url = "http://hostname:8080/redcat/rest/" +  
        "assets/{assetId}/policy/audit/{policyKey}/  
{controlParagraph}/p5/{p5type}";  
    restTemplate.put(url, "1", variables);  
}
```

Authenticate

The application must perform HTTP basic authentication.

The user must have the following permissions:

- Login
- Read Policy Assessment
- Edit Policy Assessment
- Read Assets

Resource References

"%20" is used to represent a space in parameters such as Policy Names and control Paragraph Numbers.

This section includes the following resources:

Asset Reconciliation	15
Compliance Score	17
P5 Control Maturity Factor Score (Single)	18
P5 Control Maturity Factors Scores (Multiple)	19
Notes	20
Aggregated Risk Score	22
Aggregated Vulnerability Score	22
Aggregated ESM Threat Score	23
Aggregated Compliance Score	24
Aggregated Control Maturity Score	25

Asset Reconciliation

Description

Reconciliation is the process of identifying and matching entities from different data repositories, for example, in HP Server Automation (SA) and EnterpriseView. This process is designed to assure unique identification of assets in EnterpriseView.

The asset ID is passed as a parameter in all of the other methods, therefore, retrieving the asset ID is always the first action.

Parameters

The asset ID can be retrieved with one or any combination of the following parameters:

- IP Address
- MAC Address
- Host Name

URL

```
http://{host}:{port}/redcat/rest/reconcile/  
asset?ipAddress={ipAddress}
```

```
http://{host}:{port}/redcat/rest/reconcile/  
asset?macAddress={macAddress}
```

```
http://{host}:{port}/redcat/rest/reconcile/  
asset?hostname={hostname}
```

Or any combination of parameters. For example:

```
http://{host}:{port}/redcat/rest/reconcile/  
asset?hostname={hostname}&macAddress=  
{macAddress}&ipAddress={ipAddress}
```

Note: The IP Address, MAC Address and Host Name of an asset can change. In this case, using these parameters may return a different asset ID. If your connector has a unique key for this asset that is made up of the **Connector Asset ID** and the **Connector Name**, then it is recommended that you use these parameters to make sure that the same asset is always returned.

URL

```
http://{host}:{port}/redcat/rest/reconcile/  
asset?connectorName={connectorName}&connectorAssetId=  
{connectorAssetId}&hostname={hostname}
```

```
http://{host}:{port}/redcat/rest/reconcile/  
asset?connectorName={connectorName}&connectorAssetId=  
{connectorAssetId}&ipAddress={ipAddress}
```

```
http://{host}:{port}/redcat/rest/reconcile/  
asset?connectorName={connectorName}&connectorAssetId=  
{connectorAssetId}&macAddress={macAddress}
```

Or any combination of parameters. For example:

```
http://{host}:{port}/redcat/rest/reconcile/  
asset?connectorName=MyConnector&connectorAssetId=17&ipAddress=  
192.168.0.1&hostname=assetHost&macAddress=assetMAC
```

Remark

EnterpriseView can return only one asset ID. If more than one asset is found with this method, then 404 error code is returned. None of the parameters (IP address, MAC address, host name) are mandatory in EnterpriseView. Using your knowledge of EnterpriseView, construct your queries to match a unique asset.

GET	
Action	Returns one asset ID.
Request Body	None

GET	
Returns	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <asset> <id>{asset ID}</id> </asset></pre>
Example	GET http://127.0.0.1:8080/redcat/rest/reconcile/asset?ipAddress=192.168.0.1

Compliance Score

Description

A score between 0 and 100 representing the asset compliance with a specific control.

URL

```
http://{host}:{port}/redcat/rest/assets/{asset_id}/policy/audit/
{policy_name}/{control_paragraph}/compliance
```

Remark

Attempting to create/update a compliance score that is not in the range of 0 and 100, results in a 500 exception.

All parameters are case sensitive.

GET	
Action	Returns the compliance score for a specific control in a specific policy for a specific asset. Note: If the compliance is not assessed, GET returns the parameter "NotAssessed" instead of a score.
Request Body	None
Returns	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <compliance> <complianceScore>{compliance Score}</complianceScore> </compliance></pre>
Example	GET http://127.0.0.1:8080/redcat/rest/assets/189/policy/audit/Cobit%204.1/PO1.1/compliance

PUT	
Action	Creates/updates the compliance score for a specific control in a specific policy for a specific asset. Note: If the compliance that you want to create/update is not assessed, you need to pass the parameter "NotAssessed" instead of a score.
Request Body	The compliance score
Example	PUT http://127.0.0.1:8080/redcat/rest/assets/189/policy/audit/Cobit%204.1/PO1.1/compliance (Request body) 100

P5 Control Maturity Factor Score (Single)

Description

A score between 1-5 representing the control maturity of a single P5 factor in reference to a specific asset.

URL

http://{host}:{port}/redcat/rest/assets/{asset_id}/policy/audit/{policy_name}/{control_paragraph}/p5/{p5_factor}

P5 factors: people, proof, procedure, process, product

Remark

All parameters are case sensitive.

PUT	
Action	Creates/updates the P5 control maturity score for one factor on a specific control in a specific policy for a specific asset. Note: If the P5 factor that you want to create/update is not assessed, pass the parameter "NotAssessed" instead of a score.
Request Body	The P5 control maturity score
Example	PUT http://127.0.0.1:8080/redcat/rest/assets/189/policy/audit/Cobit%204.1/PO1.1/p5/people (Request body) 5

P5 Control Maturity Factors Scores (Multiple)

Description

A score between 1-5 representing the control maturity of one or more (maximum five) P5 factors in reference to a specific asset.

URL

```
http://{host}:{port}/redcat/rest/assets/{asset_id}/policy/audit/  
{policy_name}/{control_paragraph}/p5
```

Remark

All parameters are case sensitive.

GET	
Action	<p>Returns the P5 control maturity score for multiple factors on a specific control in a specific policy for a specific asset. If a specific P5 factor is not applied to the control, then it will not be returned.</p> <div>Note: If a P5 factor is not assessed, GET returns the parameter "NotAssessed" instead of a score.</div>
Request Body	None
Returns	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <p5> <people>{people}</people> <procedure>{procedure}</procedure> <process>{process}</process> <product>{product}</product> <proof>{proof}</proof> </p5></pre>
Example	GET http://127.0.0.1:8080/redcat/rest/assets/189/policy/audit/Cobit%204.1/PO1.1/p5/

PUT	
Action	<p>Creates/updates the P5 control maturity score for multiple factors on a specific control in a specific policy for a specific asset.</p> <div> <p>Note:</p> <ul style="list-style-type: none"> • If the P5 factor that you want to create/update is not assessed, you need to pass the parameter "NotAssessed" instead of a score. • If you try to create/update a P5 factor that is not applied to the control, you get a 500 error code. </div>
Request Header	content-type: application/xml
Request Body	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <p5> <people>{people}</people> <procedure>{procedure}</procedure> <process>{process}</process> <product>{product}</product> <proof>{proof}</proof> </p5></pre>
Example	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <p5> <people>3</people> <procedure>3</procedure> <process>4</process> <product>5</product> <proof>NotAssessed</proof> </p5></pre>

Notes

Description

Any number of notes can be attached to a control in a specific policy for a specific asset.

URL

`http://{host}:{port}/redcat/rest/assets/{asset_id}/policy/audit/{policy_name}/{control_paragraph}/notes`

Remark

POST creates a single note. GET returns all notes for the control.

GET	
Action	Returns all of the notes for a specific control in a specific policy for a specific asset.
Request Body	None
Returns	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <notes> <note> <createTime>{creation date and time} </createTime> <creator> <uniqueId>{user name}</uniqueId> </creator> <message>{note text}</message> </note> ... </notes></pre>
Example	GET <code>http://127.0.0.1:8080/rest/assets/189/policy/audit/Cobit%204.1/PO1.1/notes</code>

POST	
Action	Create a new note for a specific control in a specific policy for a specific asset.
Request Body	The note text
Returns	HTTP code 200 or an error code
Example	<pre>http://127.0.0.1:8080/redcat/rest/assets/189/policy/audit/Cobit%204.1/PO1.1/note {note text}</pre>

Aggregated Risk Score

Description

A score between 0 and 100 representing the aggregated risk of an asset.

URL

`http://{host}:{port}/redcat/rest/risk/score/{asset_id}`

Remark

All parameters are case sensitive.

GET	
Action	Returns the aggregated risk score for a specific asset. Note: If there is no aggregated risk score for the asset, GET returns an empty tag instead of a score.
Request Body	None
Returns	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <RiskScore> <AggregatedScore> {aggregated risk score} </AggregatedScore> </RiskScore></pre>
Example	GET <code>http://127.0.0.1:8080/redcat/rest/risk/score/189</code>

Aggregated Vulnerability Score

Description

A score between 0 and 10 representing the aggregated vulnerability of an asset.

URL

`http://{host}:{port}/redcat/rest/vulnerability/score/{asset_id}`

Remark

All parameters are case sensitive.

GET	
Action	Returns the aggregated risk score for a specific asset. Note: If there is no aggregated vulnerability score for the asset, GET returns "0".
Request Body	None
Returns	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <VulnerabilityScore> <AggregatedScore> {aggregated vulnerability score} </AggregatedScore> </VulnerabilityScore></pre>
Example	GET <code>http://127.0.0.1:8080/redcat/rest/vulnerability/score/189</code>

Aggregated ESM Threat Score

Description

A score between 0 and 10 representing the aggregated ESM threat of an asset.

URL

`http://{host}:{port}/redcat/rest/esm/score/{asset_id}`

Remark

All parameters are case sensitive.

GET	
Action	Returns the aggregated risk score for a specific asset. Note: If there is no aggregated ESM threat score for the asset, GET returns "0".
Request Body	None

GET	
Returns	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <ESMScore> <AggregatedScore> {aggregated esm threat score} </AggregatedScore> </ESMScore></pre>
Example	GET http://127.0.0.1:8080/redcat/rest/esm/score/189

Aggregated Compliance Score

Description

A score between 0 and 100 representing the aggregated compliance of an asset.

URL

http://{host}:{port}/redcat/rest/compliance/score/{asset_id}

Remark

All parameters are case sensitive.

GET	
Action	Returns the aggregated compliance score for a specific asset. Note: If there is no aggregated compliance score for the asset, GET returns an empty tag instead of a score.
Request Body	None
Returns	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <ComplianceScore> <AggregatedScore> {aggregated compliance score} </AggregatedScore> </ComplianceScore></pre>
Example	GET http://127.0.0.1:8080/redcat/rest/compliance/score/189

Aggregated Control Maturity Score

Description

A score between 0 and 5 representing the aggregated maturity of an asset.

URL

`http://{host}:{port}/redcat/rest/maturity/score/{asset_id}`

Remark

All parameters are case sensitive.

GET	
Action	Returns the aggregated maturity score for a specific asset. Note: If there is no aggregated maturity score for the asset, GET returns an empty tag instead of a score.
Request Body	None
Returns	<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <MaturityScore> <AggregatedScore> {aggregated maturity score} </AggregatedScore> </MaturityScore></pre>
Example	GET <code>http://127.0.0.1:8080/redcat/rest/maturity/score/189</code>