CA User Activity Reporting Module

Virtual Automation API Guide

Release 12.6

CA Technologies Product References

This document references the following CA Technologies products:

- CA ControlMinder
- CA Audit
- CA ACF2™
- CA Directory
- CA Embedded Entitlements Manager (CA EEM)
- CA User Activity Reporting Module
- CA IdentityMinder
- CA IT Process Automation Manager (CA IT PAM)
- CA NSM
- CA Security Command Center (CA SCC)
- CA Service Desk
- CA SiteMinder®
- CA Spectrum®
- CA Top Secret®

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: About This Guide 7

Chapter 2: About the Virtual Automation API 7

Virtual Automation API Overview 8
Virtual Automation API Structure 9

Chapter 3: Virtual Automation API Examples 11

List Tenants 12
List Collection Profiles (/collectionprofiles) 13
Deploy Collection (/deploycollection) 15
Source ID Calls (/<sourceid>) 16
    Identify Resource 17
    Delete Resource 17
Credentials Calls (/credentials) 18
    List Credentials 18
    Replace Credentials 19

Chapter 1: About This Guide

The *CA User Activity Reporting Module Virtual Automation API Guide* provides instructions for using the REST-architecture virtual automation API to set up log collection from virtual machines.

The guide is designed for administrators or web designers familiar with basic API structure and usage, CA User Activity Reporting Module queries and event refinement. They need administrator access to CA User Activity Reporting Module and other required third-party or CA products.

REST services use the HTTP protocol for all communication. Familiarity with both the HTTP protocol and REST (Representational State Transfer) architecture is required.

Chapter 2: About the Virtual Automation API

The virtual automation API allows you to deploy event collection for virtual machines using CA User Activity Reporting Module. You can use the API to trigger a preset collection profile, which contains all the information necessary to provision event collection.

You can also use the API to set access credentials for event collection, identify available resources, and other related functions.

More information:

Virtual Automation API Structure (see page 9)
Virtual Automation API Overview (see page 8)

Virtual Automation API Overview

To use the virtual API, you invoke HTTP methods against resources, each of which has its own URI. The API uses the following HTTP methods:

- POST - creates a resource, supplying the resource parameters in a message body. You can use this method to deploy event collection for a virtual machine.
- GET - retrieves the current representation of a resource. You can use this method to get a list of tenants or information about a deployment.
- PUT - updates a resource, replacing the current resource representation with the one you supply in the message body. You can use this method to change existing event source credentials.
- DELETE - deletes a resource. You can use this method to halt event collection.

Provide a valid CA User Activity Reporting Module user and password, or certificate name and password, on each API call. Do this using HTTP Basic Authentication (the Authorization header).

For example, you could use the available methods to deploy and control event collection this way:

1. Deploy a connector and start event collection on a virtual machine by using POST to the fixed resource "/deploycollection". POST creates a resource that represents your event source.

   This method returns a URI for the new resource.
2. Check the status of the event source, by using GET against the resource URI.
3. Remove the event source, if necessary, by using DELETE against the same URI.

Some resources support multiple HTTP methods, others support only one. The documentation for each identifies the supported methods.

Virtual Automation API Structure

All resource URIs for the virtual API have a defined structure, as illustrated in the following example:

```
https://hostname:8443/rest/am/1/collectionprofiles
```

The first part of the URI identifies the target server. Replace "hostname" with the name of the CA User Activity Reporting Module server you want to contact.

The second part of the URI, "/rest/am/1" is common to all the resources on that server. The “1” specifies the version of the API you want to access.

The third element defines the resource you want to access, in this case “/collectionprofiles”.

You can return data or send data in either XML or JSON format. To specify data return format, include values in the HTTP Accept header to specify which you format you want:

- "Accept: application/xml"
- "Accept: application/json"

To specify sent the format of data you send using PUT or POST, use the HTTP Content-Type header:

- "Content-Type: application/xml"
- “Content-Type: application/json”

Note: All API examples in this guide are shown using the cURL command line HTTP client.

Chapter 3: Virtual Automation API Examples

This section contains the following topics:

List Tenants (see page 12)
List Collection Profiles (/collectionprofiles) (see page 13)
Deploy Collection (/deploycollection) (see page 15)
Source ID Calls (/<sourceid>) (see page 16)
Credentials Calls (/credentials) (see page 18)

List Tenants

You can list the tenants in your virtual CA User Activity Reporting Module environment, allowing you to identify the tenants available for event collection deployment.

Supported Methods: GET

```
GET curl -u elm_user:elm_password -k -H "Accept: application/xml"
"https://hostname:8443/rest/am/1/tenants"
```

Returns:

```
<tenants>

    <tenant>

        <name>Default</name>

        <description>The default Tenant</description>

    </tenant>

    <teant>

        <name>Tenant1</name>

        <description>Text description of the first tenant</description>

    </tenant>

    <tenant>

        <name>Tenant 2</name>

        <description>Text description of the second tenant</description>

    </tenant>

</tenants>
```

List Collection Profiles (/collectionprofiles)

You can use this call to return a list of the available event collection profiles. Each profile contains the information required to configure event collection on a specific event source.

Note: Event Collection profiles are configured from the CA User Activity Reporting Module user interface. See the CA User Activity Reporting Module Online Help for more information on configuring Event Collection profiles.

Supported Methods: GET

```
GET curl -u elm_user:elm_password -k -H "Accept: application/xml"
"https://hostname:8443/rest/am/1/collectionprofiles"
```

Returns:

```
<collectionProfiles>

    <collectionProfile>

        <name>Tenant1 - Linux</name>

        <description>Collects Linux syslog events for the first tenant</description>

        <credentialsRequired>false</credentialsRequired>

    </collectionProfile>

    <collectionProfile>

        <name>Tenant1 Windows</name>

        <description>Collects WinRM events for the first tenant</description>

        <credentialsRequired>true</credentialsRequired>

    </collectionProfile>

    <collectionProfile>

        <name>Tenant2 HPUX</name>

        <description>Collects HPUX syslog events for the second tenant</description>

        <credentialsRequired>false</credentialsRequired>

    </collectionProfile>

</collectionProfiles>
```

The “credentialsRequired” element indicates whether you must submit an event source userid and password during deployment:

- This value is true in the case of active (or pull) collection such as a WinRM connector, that polls event sources for information.
- This value is false in the case of passive (or push) collection such a syslog server that sends data directly to CA User Activity Reporting Module.

Deploy Collection (/deploycollection)

You can use the API to deploy event collection on a virtual machine. Include a message body specifying the event profile you want to use.

Note: Event Collection profiles are configured from the CA User Activity Reporting Module user interface. See the CA User Activity Reporting Module Online Help for more information on configuring Event Collection profiles.

The following procedure illustrates how to deploy a collection using the cURL utility.

Follow these steps:

1. Create a text file called deploy.txt containing the deployment parameters:

   ```
   <deploymentRequest>
   <tenant>Default</tenant><profile>syslog
   test</profile><host>syslogsource.ca.com</host><ip>10.0.0.0</ip><credentials><
   user>root</user><password>rootpw</password></credentials></deploymentRequest>
   ```

   The following parameters are available:

   <tenant>

   Names the virtual tenant where you want to deploy event collection. You can get a list of available tenants using /tenants.

   <profile>

   Names the event collection profile you want to use. You can get a list of available profiles using /collectionprofiles.

   <host>

   Names the event source from which you want to collect events.

   <ip>

   Specifies the IP address of the event source from which you want to collect events.

   <credentials>

   Contains the elements that supply the username and password for access to the event source. This element is only required for connection profiles that are set to require credentials.

2. Open a command line window, and navigate to the directory where you saved the text file.

3. Issue the following command:

   ```
   curl -u elm_user:elm_password -k -H "Accept: application/xml" –H "Content-Type:
   application/xml" –X POST –d @deploy.txt
   "https://hostname:8443/rest/am/1/deploycollection"
   ```

   The "-d @deploy.txt" element delivers the content of the text file in the body of the request.

If the deployment is successful, you receive an HTTP 201 (CREATED) message:

```
HTTP/1.1 201 Created
Location:
http://myelmhost:8443/rest/agentgroups/Agents/agents/014589ec-4b97-4179-8778-65b1
671996f8/connectors/1cde5aa8-e11c-4c36-b7cc-712477c9f52f/sources/10.0.0.0
Content-Type: application/xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<eventTarget>
    <host>10.0.0.0</host>
    <tcpPort>1468</tcpPort>
    <udpPort>40514</udpPort>
</eventTarget>
```

The response shows the URI of the deployed resource, following "Location:".

This information can be used to modify or delete the deployment. In the preceding example, the deployed resource is a passive connector, so the "eventTarget" element appears. EventTarget shows the port and IP address information for the connector, allowing you to configure the event source to transmit events to the proper target.

If there is not enough capacity available in the selected agent group, an error message (HTTP 507) appears.

Source ID Calls (/<sourceid>)

The <sourceid> resource represents a CA User Activity Reporting Module event source. You can return information about the resource, or remove it, which halts event collection from the corresponding event source.

Supported Methods, GET, DELETE

More information:

Identify Resource (see page 17)
Delete Resource (see page 17)

Identify Resource

You can identify resources representing event sources and get information about them using GET. This call returns information about the source at the specified URI path. This path is derived from the result of a /deploycollection call.

```
GET curl -u elm_user:elm_password -k -H "Accept: application/xml"
"https://hostname:8443/rest/am/1/agentgroups/<groupid>/agents/<agentid>/connector
s/<connid>/sources/<sourceid>
```

In your environment, replace the sample URI path "/agentgroups/<groupid>/agents/<agentid>/connectors/<connid>/sources/<sourceid>" with the path for the resource you want.

This call returns:

```
<connectorSource>
    <id>e94523c9-65a3-4510-87cb-fc693ffce966</id>
    <integration>Syslog</integration>
    <integrationVersion>12.5.5203.0</integrationVersion>
    <deploymentPending>false</deploymentPending>
    <target>
        <host>calmdev06</host>
        <tcpPort>1468</tcpPort>
        <udpPort>40514</udpPort>
    </target>
</connectorSource>
```

When the deploymentPending value is "true", it means that the agent is reconfiguring and is currently unavailable for many operations.

Delete Resource

You can remove a resource representing an event source using DELETE. This call deletes the specified resource and halts event collection. The URI path is derived from the result of a /deploycollection call.

```
DELETE curl -u elm_user:elm_password -k -H "Accept: application/xml"
"https://hostname:8443/rest/am/1//agentgroups/<groupid>/agents/<agentid>/connecto
rs/<connid>/sources/<sourceid>
```

In your environment, replace the sample URI path "/agentgroups/<groupid>/agents/<agentid>/connectors/<connid>/sources/<sourceid>" with the path for the resource you want.

The call returns a confirmation (HTTP 200) when the deletion is complete.

Credentials Calls (/credentials)

The /credentials resource represents the user name and password used by a connector to access an event source. You can retrieve information about the credentials, or update them.

Supported Methods, GET, PUT

More information:

List Credentials (see page 18)
Replace Credentials (see page 19)

List Credentials

You can retrieve the credentials used by a deployed connector to access an event source. The response displays the username and password. This call is only valid for active connectors. An HTTP 404 error appears for passive connectors.

```
GET curl -u elm_user:elm_password -k -H "Accept: application/xml"
"https://hostname:8443/rest/am/1/agentgroups/<groupid>/agents/<agentid>/connector
s/<connid>/sources/<sourceid>/credentials
```

In your environment, replace the sample URI path "/agentgroups/<groupid>/agents/<agentid>/connectors/<connid>/sources/<sourceid>" with the path for the resource you want.

This call returns:

```
<credentials>
    <user>root</user>
    <password>password</password>
    <domain>domain_name</domain>
</credentials>
```

The optional domain value is only used for Windows credentials.

Replace Credentials

You can replace existing credentials. This call is only valid for active connectors. An HTTP 404 error appears for passive connectors.

```
curl -u elm_user:elm_password -k -H "Accept: application/xml" –H "Content-Type: application/xml" –X PUT –d
<credentials><user>root</user><password>password</password><domain>domain_name</domain></credentials>
"https://hostname:8443/rest/am/1/agentgroups/<groupid>/agents/<agentid>/connectors/<connid>/sources/<sourceid>/credentials
```

In your environment, replace the sample URI path "/agentgroups/<groupid>/agents/<agentid>/connectors/<connid>/sources/<sourceid>" with the path for the resource you want.

In this case, the "–d" option specifies the new representation for the resource directly on the command line.

Note: This example contains the domain value, which is only required for Windows credentials.