CA User Activity Reporting Module

Virtual Automation API ガイド

リリース 12.5.03

CA Technologies 製品リファレンス

このマニュアルが参照している CA Technologies の製品は以下のとおりです。

- CA Access Control
- CA Audit
- CA ACF2™
- CA Directory
- CA Embedded Entitlements Manager（CA EEM）
- CA Enterprise Log Manager
- CA Identity Manager
- 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®

CA への連絡先

テクニカル サポートの詳細については、弊社テクニカル サポートの Web サイト（http://www.ca.com/jp/support/）をご覧ください。

目次

第 1 章: 本書の内容

「CA User Activity Reporting Module Virtual Automation API ガイド」は、仮想マシンからのログ収集をセットアップするために REST アーキテクチャの Virtual Automation API を使用するための手順について説明します。

このガイドは、基本的な API 構造と使用方法、CA User Activity Reporting Module クエリとイベント精製に関する知識を持った管理者または Web デザイナを対象としています。このガイドを利用するには、CA User Activity Reporting Module および他の必要なサードパーティ製品または CA 製品に対する管理者アクセス権が必要です。

REST サービスは、すべての通信において HTTP プロトコルを使用します。HTTP プロトコルおよび REST（Representational State Transfer）アーキテクチャの両方についての理解も必須です。

第 2 章: Virtual Automation API について

Virtual Automation API は、CA User Activity Reporting Module を使用して仮想マシン用のイベント収集を展開できるようにします。この API を使用して、事前設定された収集プロファイルをトリガできます。プロファイルには、イベント収集を開始するために必要な情報がすべて含まれています。

また、イベント収集用のアクセス認証情報の設定、使用可能なリソースの特定、および他の関連する機能にこの API を使用することもできます。

詳細情報:

Virtual Automation API 概要 (P. 10)
Virtual Automation API の構造 (P. 11)

Virtual Automation API 概要

Virtual Automation API を使用するには、リソースに対して HTTP メソッドを起動します。それぞれに独自の URI があります。API は以下の HTTP メソッドを使用します。

- POST - メッセージ本文でリソース パラメータを提供して、リソースを作成します。このメソッドを使用して仮想マシン用のイベント収集を展開することができます。
- GET - リソースの現在の状態を取得します。このメソッドを使用してテナントのリストまたは展開に関する情報を取得することができます。
- PUT - リソースを更新し、現在のリソースの状態をメッセージ本文で提供するもので置換します。このメソッドを使用して既存のイベント ソース認証情報を変更することができます。
- DELETE - リソースを削除します。このメソッドを使用してイベント収集を停止することができます。

各 API コールで、有効な CA User Activity Reporting Module ユーザとパスワード、または証明書の名前とパスワードを提供します。これは HTTP 基本認証（Authorization ヘッダ）を使用して行います。

たとえば、利用可能なメソッドを使用して、イベント収集を以下のように展開および制御できます。

1. POST を固定リソース “/deploycollection” に対して使用し、コネクタを展開して仮想マシン上でイベント収集を開始します。POST は、対象のイベント ソースを表すリソースを作成します。

   このメソッドは、新しいリソースの URI を返します。
2. リソース URI に対して GET を使用し、イベント ソースのステータスを確認します。
3. 同じ URI に対して DELETE を使用し、必要に応じてイベント ソースを削除します。

リソースには複数の HTTP メソッドをサポートするものと、1 つのメソッドのみをサポートするものがあります。サポートされているメソッドはそれぞれのドキュメントで確認できます。

Virtual Automation API の構造

以下の例に示すとおり、Virtual Automation API のすべてのリソース URI には定義された構造があります。

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

URI の最初の部分はターゲット サーバを特定します。"hostname" の部分をアクセスする CA User Activity Reporting Module サーバの名前で置き換えます。

URI の 2 番目の部分 "/rest/am/1" は同じサーバ上ですべてのリソースに共通です。"1" は、アクセスする API のバージョンを指定します。

3 番目のエレメントは、アクセスするリソース(この場合は “/collectionprofiles”)を定義します。

データは XML または JSON のいずれかの形式で返したり送信したりできます。データが返される形式を指定するには、HTTP Accept ヘッダに値を含め、使用する形式を指定します。

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

PUT または POST を使用して送信されるデータの形式を指定するには、HTTP Content-Type ヘッダを使用します。

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

注: このガイドの API 例はすべて cURL コマンド ライン HTTP クライアントを使用して示されています。

第 3 章: Virtual Automation API の例

このセクションには、以下のトピックが含まれています。

テナントの一覧表示

仮想 CA User Activity Reporting Module 環境内のテナントを一覧表示し、イベント収集の展開に使用可能なテナントを特定することができます。

サポートされているメソッド: GET

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

戻り値:

```
<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>
```

収集プロファイルの一覧表示（/collectionprofiles）

利用可能なイベント収集プロファイルのリストを返すことができます。各プロファイルには、特定のイベント ソースでイベント収集を設定するために必要な情報が含まれます。

注: イベント収集プロファイルは CA User Activity Reporting Module ユーザ インターフェースから設定されます。イベント収集プロファイルの設定の詳細については、CA User Activity Reporting Module オンライン ヘルプを参照してください。

サポートされているメソッド: GET

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

戻り値:

```
<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>
```

"credentialsRequired" エレメントは、展開中にイベント ソースのユーザ ID とパスワードをサブミットする必要があるかどうかを示します。

- この値は、WinRM コネクタなど、情報を収集するためにイベント ソースをポーリングするアクティブな(またはプルの)収集の場合は true になります。
- この値は、たとえば、CA User Activity Reporting Module にデータを直接送信する syslog サーバなど、パッシブな(またはプッシュの)収集の場合は false になります。

収集の展開(/deploycollection)

この API を使用して仮想マシンでイベント収集を展開することができます。使用するイベント プロファイルを指定するメッセージ本文を含めます。

注: イベント収集プロファイルは CA User Activity Reporting Module ユーザ インターフェースから設定されます。イベント収集プロファイルの設定の詳細については、CA User Activity Reporting Module オンライン ヘルプを参照してください。

以下の手順は、cURL ユーティリティを使用して収集を展開する方法について説明しています。

次の手順に従ってください:

1. deploy.txt というテキスト ファイルを作成し、展開パラメータを含めます。

```
<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>
```

以下のパラメータが使用可能です。

<tenant>

イベント収集を展開する仮想テナントを指定します。/tenants を使用して利用可能なテナントのリストを取得できます。

<profile>

使用するイベント収集プロファイルを指定します。 /collectionprofiles を使用して利用可能なプロファイルのリストを取得できます。

<host>

イベントを収集する対象のイベント ソースを指定します。

<ip>

イベントを収集する対象のイベント ソースの IP アドレスを指定します。

<credentials>

イベント ソースにアクセスするためのユーザ名とパスワードを提供するエレメントが含まれます。このエレメントは、認証情報を必要とするよう設定されている接続プロファイルにのみ必要です。

2. コマンド ライン ウィンドウを開き、テキスト ファイルを保存したディレクトリへ移動します。
3. 以下のコマンドを発行します。

```
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"
```

"-d@deploy.txt" エレメントは、リクエストの本文にテキスト ファイルのコンテンツを提供します。

展開が成功した場合、以下の HTTP 201 （CREATED）メッセージを受信します。

```
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>
```

このレスポンスは、展開されたリソースの URI を示し、続いて場所（"Location"）を示しています。

この情報を使用して展開を変更または削除することができます。 前述の例では、展開されたリソースはパッシブ コネクタであるため、"eventTarget" エレメントが表示されていました。EventTarget は、コネクタのポートおよび IP アドレス情報を表示し、イベントを適切なターゲットに送信するためにイベント ソースを設定できるようにします。

選択されたエージェント グループに十分な利用可能容量がない場合、エラー メッセージ（HTTP 507）が表示されます。

ソース ID コール（/<sourceid>）

<sourceid> リソースは CA User Activity Reporting Module イベント ソースを表します。リソースに関する情報を返したり、リソースを削除したりできます。削除すると、対応するイベント ソースからのイベント収集が停止します。

サポートされているメソッド：GET、DELETE

詳細情報：

リソースの特定 (P. 19)
リソースの削除 (P. 20)

リソースの特定

イベント ソースを表すリソースを特定し、GET を使用して、それらに関する情報を取得できます。このコールは、指定された URI パスのソースに関する情報を返します。このパスは /deploycollection コールの結果に由来します。

```
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>
```

実際の環境では、サンプル URI パス "/agentgroups/<groupid>/agents/<agentid>/connectors/<connid>/sources/<sourceid>" を、対象のリソースのパスで置き換えます。

このコールの戻り値

```
<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>
```

deploymentPending の値が "true" である場合、エージェントが再設定中で、現在多くの操作で利用不可能であることを意味します。

リソースの削除

DELETE を使用して、イベント ソースを表すリソースを削除できます。このコールは、指定されたリソースを削除してイベント収集を停止します。URI パスは /deploycollection コールの結果に由来します。

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

実際の環境では、サンプル URI パス "/agentgroups/<groupid>/agents/<agentid>/connectors/<connid>/sources/<sourceid>" を、対象のリソースのパスで置き換えます。

削除が完了すると、そのコールは確認メッセージ（HTTP 200）を返します。

認証情報コール（/credentials）

/credentials リソースは、イベント ソースにアクセスするためにコネクタによって使用されるユーザ名およびパスワードを表します。認証情報に関する情報を取得するか、またはそれらを更新できます。

サポートされているメソッド：GET、PUT

詳細情報：

認証情報の一覧表示 (P. 21)
認証情報の置換 (P. 21)

認証情報の一覧表示

展開されたコネクタがイベント ソースにアクセスするために使用する認証情報を取得できます。応答にはユーザ名とパスワードが表示されます。このコールはアクティブなコネクタにのみ有効です。パッシブ コネクタに対しては HTTP 404 エラーが表示されます。

```
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
```

実際の環境では、サンプル URI パス "/agentgroups/<groupid>/agents/<agentid>/connectors/<connid>/sources/<sourceid>" を、対象のリソースのパスで置き換えます。

このコールの戻り値

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

オプションのドメインの値は Windows 認証情報にのみ使用されます。

認証情報の置換

既存の認証情報を置換できます。このコールはアクティブなコネクタにのみ有効です。パッシブ コネクタに対しては HTTP 404 エラーが表示されます。

```
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</d
omain></credentials>
"https://hostname:8443/rest/am/1/agentgroups/<groupid>/agents/<agentid>/connector
s/<connid>/sources/<sourceid>/credentials
```

実際の環境では、サンプル URI パス "/agentgroups/<groupid>/agents/<agentid>/connectors/<connid>/sources/<sourceid>" を、対象のリソースのパスで置き換えます。

この場合、"-d" オプションによって、リソースの新しい表現をコマンド ラインに直接指定します。

注: この例にはドメイン値が含まれています。これは Windows 認証情報にのみ必要です。