Monitoring REST API の概要

Monitoring REST API では、ネットワークでモニタリング ノードを使用して、セッションおよびノード固有の情報を収集することができます。セッションは、目的のノードにアクセスしてから情報の収集に必要な操作を完了するまでの期間として定義されます。

次の Monitoring REST API カテゴリが Cisco ISE Release 1.2 でサポートされます:

- セッション管理
- トラブルシューティング
- 認可変更（CoA）

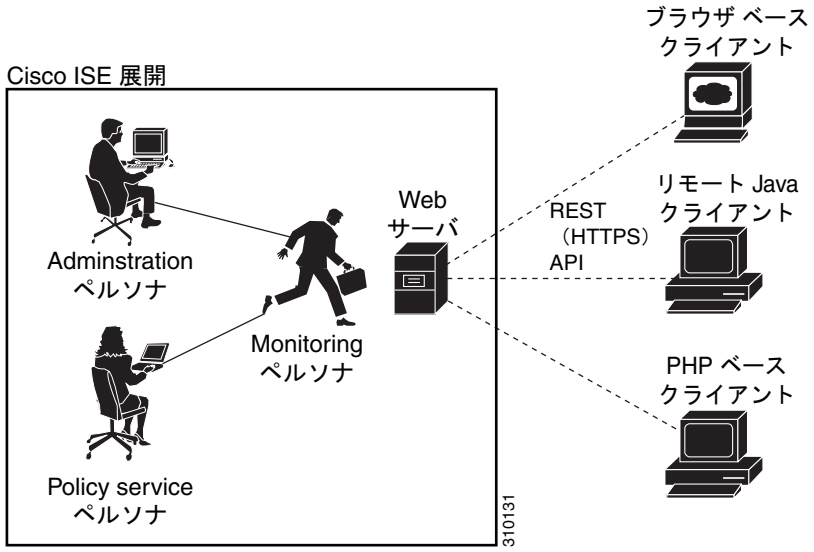
（注） Monitoring ペルソナによって監視されているエンドポイントに関する情報を収集するために、サポート対象のカテゴリだけを使用できます。Monitoring は、ノード タイプが Cisco ISE Release 1.2 の導入で実行できる、サポート対象の 3 つのペルソナの 1 つです。このガイドの残りの部分では、Cisco ISE ノードの Monitoring ペルソナを説明するため、「モニタリング ノード」を使用します。

これらのカテゴリを Cisco ISE アプライアンスの Policy service ペルソナに関する情報の収集に使用しようとすると、エラーが発生します。Cisco ISE ノードおよびペルソナに関する詳細については、『*Cisco Identity Services Engine User Guide, Release 1.2*』を参照してください。

Monitoring REST API コールを使用すると、ネットワークで、個々のエンド ポイントに格納されている重要なリアルタイムのセッション ベースの情報を検索、監視、収集することができます。モニタリング ノードを通じてこの情報にアクセスできます。

収集するリアルタイムのセッション ベースの情報は、Cisco ISE 操作を理解するのに役立ち、状態や問題の診断を支援することができます。また、モニタリング動作に影響を及ぼす可能性のあるエラー条件、またはアクティビティや動作をトラブルシュートするために使用できます。図 1-1 に示すように、Monitoring REST API コールは、モニタリング ノードにアクセスして Cisco ISE 導入のエンド ポイントに格納されている重要なセッション ベースの情報を取得する目的で使用されます。

図 1-1　分散展開での Monitoring REST API コール



モニタリング ノードの確認

はじめる前に

API コールをモニタリング ノードで正常に呼び出す前に、監視するノードが有効なノードであることを確認しておく必要があります。

（注）　パブリック Monitoring REST API を使用できるようにするには、最初に有効なクレデンシャルを使用して Cisco ISE で認証を受ける必要があります。

ステップ 1　有効なログイン クレデンシャル（ユーザ名とパスワード）を [Cisco ISE ログイン（Cisco ISE Login）] ウィンドウに入力し、[ログイン（Login）] をクリックします。

Cisco ISE ダッシュボードとユーザ インターフェイスが表示されます。

ステップ 2　[許可（Authorization）] > [システム（System）] > [展開（Deployment）] の順に選択します。

展開されたすべての設定済みノードがリストされた [展開ノード（Deployment Nodes）] ページが表示されます。

ステップ 3　[展開ノード（Deployment Nodes）] ページの [ロール（Roles）] カラムで、モニタするターゲット ノードのロールがモニタリング ノードとしてリストされていることを確認します。

サポートされる API コール

次の表で、さまざまな種類の API コールを説明し、API コールの形式の例を示します。

- 表 1-1（P.1-3）：セッション管理用の API コールを定義します。
- 表 1-2（P.1-6）：トラブルシューティング用の API コールを定義します。

- 表 1-3（P.1-7）：CoA API コールを定義します。

Cisco ISE でサポートされる Monitoring REST API を使用して認証を受けるため、汎用プログラマチック インターフェイスを使用する計画の場合、Cisco ISE と使用するツールを接続する REST ベースのクライアントを最初に作成する必要があります。次に、この REST クライアントを使用して Cisco ISE Monitoring REST API で認証を受け、API 要求を変換してモニタリング ノードに送信します。そして、API 応答を再変換し、指定されたツールに引き渡します。

表 1-1　Cisco ISE セッション管理 API コール

API コール カテゴリ	説明と例
セッション カウンタ	
ActiveCount	アクティブなセッションの数をリストします。 https://<ISEhost>/ise/mnt/Session/ActiveCount
PostureCount	ポスチャされたエンドポイントの数をリストします。 https://<ISEhost>/ise/mnt/Session/PostureCount **（注）** ポスチャとは、Cisco ISE ネットワークに接続しているすべてのエンドポイントの状態（またはポスチャ）の確認を支援するサービスです。Cisco ISE は、デバイスのポスチャ コンプライアンスを確認するために NAC Agent を使用します。
ProfilerCount	アクティブなプロファイラ サービス セッションの数をリストします。 https://<ISEhost>/ise/mnt/Session/ProfilerCount **（注）** プロファイラとは、Cisco ISE ネットワークにあるすべての接続エンドポイントの機能の識別、検索、確認を支援するサービスです。

表 1-1　Cisco ISE セッション管理 API コール （続き）

API コール カテゴリ	説明と例
セッション リスト （注）　セッション リストには、MAC アドレス、ネットワーク アクセス デバイス（NAD）の IP アドレス、ユーザ名、セッションに関連付けられているセッション ID 情報が含まれます。	
ActiveList	すべてのアクティブなセッションをリストします。 https://<ISEhost>/ise/mnt/Session/ActiveList （注）　Cisco ISE のこのリリースでは、アクティブな認証済みエンドポイント セッションの表示可能な最大数は、250,000 に制限されています。
AuthList	現在アクティブなすべての認証済みセッションをリストします。 https://<ISEhost>/ise/mnt/Session/AuthList/<parameteroptions> 異なる値を返す次のパラメータ オプションを指定できます。 • null/null：すべてのアクティブな認証済みセッションをリストします。 • null/endtime：指定された終了時刻の後にアクティブなすべての認証済みセッションがリストされます。 • starttime/null：指定された開始時刻の前にアクティブなすべての認証済みセッションがリストされます。 • starttime/endtime：指定された開始時刻と終了時刻の間で認証されたすべてのアクティブなセッションがリストされます。 次の形式で、開始時刻と終了時刻の日付と時刻を入力します。 YYYY-MM-DD hh:mm:ss.s 値は次のとおりです。 • YYYY：4 桁の年 • MM：2 桁の月（01 = 1 月など） • DD：2 桁の日（01 ～ 31） • hh：2 桁の時刻（00 ～ 23）（a.m. と p.m. は 使用できません） • mm：2 桁の分（00 ～ 59） • ss：2 桁の秒（00 ～ 59） • s：秒の小数を表す 1 桁以上の値 （注）　すべての Cisco ISE ノードは、タイム ゾーンを使用して設定されます。推奨されるタイム ゾーンは UTC です。 4 つのパラメータ オプションをすべて示すサンプルについては、「AuthList API コール データ」（P.2-8）を参照してください。

表 1-1　Cisco ISE セッション管理 API コール （続き）

API コール カテゴリ	説明と例
セッション属性 **（注）** これは、指定された検索属性を含む最新のセッションのタイムスタンプに基づいた検索です。	
MACAddress	指定した MAC アドレスを含む最新のセッションについてデータベースを検索します。 https://<ISEhost>/ise/mnt/Session/MACAddress/<macaddress> **（注）** XX:XX:XX:XX:XX:XX は MAC アドレス形式です。大文字と小文字は区別されません（例：0a: 0B: 0c: 0D: 0e: 0F）。 **（注）** MAC アドレスは、監視対象の正しいセッションを検索する唯一の一意のキーとして機能します。MAC アドレスの検索のベースとすることが可能なアクティブなすべてのセッションと MAC アドレスをリストするには ActiveList API コールを使用します。
UserName	指定したユーザ名を含む最新のセッションについてデータベースを検索します。 https://<ISEhost>/ise/mnt/Session/UserName/<username> **（注）** ユーザ名は、ネットワーク ユーザ名に使用しているものと同じ Cisco ISE パスワード ポリシーに準拠している必要があります。Monitoring REST API の唯一の無効な文字はバックスラッシュ（\）文字です。詳細については、『*Cisco Identity Services Engine User Guide, Release 1.1*』の「User Password Policy」を参照してください。
IPAddress	指定した NAS IP アドレスを含む最新のセッションについてデータベースを検索します。 https://<ISEhost>/ise/mnt/Session/IPAddress/<nasipaddress> **（注）** xxx.xxx.xxx.xxx は NAS IP アドレス形式（例：10.10.10.10）です。

セッション管理用の Cisco ISE API コールの詳細については、第 2 章「セッション管理用の API コールの使用方法」を参照してください。

表 1-2　Cisco ISE トラブルシューティング API コール - トラブルシューティング

API コール	説明と例
Version	ノードのバージョンおよびタイプをリストします。 https://<ISEhost>/ise/mnt/Version ノードのタイプは、次の値（0 ～ 3）のいずれかです。 0：STAND_ALONE_MNT_NODE 1：ACTIVE_MNT_NODE 2：STAND_BY_MNT_NODE 3：NOT_AN_MNT_NODE （注）　STAND_ALONE_MNT_NODE は、分散展開で機能しないモニタリング ノードであることを意味します。 ACTIVE_MNT_NODE は、分散展開におけるプライマリ - セカンダリ関係のプライマリ ノードであることを意味します。 STAND_BY_MNT_NODE は、分散展開におけるプライマリ - セカンダリ ペアのセカンダリ ノードであることを意味します。 NOT_AN_MNT_NODE は、モニタリング ノードではないことを意味します。サポート対象の ISE ノードおよびペルソナの詳細については、『*Cisco Identity Services Engine User Guide, Release 1.1*』を参照してください。
FailureReasons	障害の理由をリストします。 https://<ISEhost>/ise/mnt/FailureReasons 各障害理由は、次の例に示すように、エラーコード（failureReason id）、簡単な説明（code）、障害理由（cause）、および可能な対処（resolution）を表示します。 <failureReason id="100009"> <code> 100009 WEBAUTH_FAIL <cause> This may or may not be indicating a violation. <resolution> Please review and resolve this issue according to your organization's policy. （注）　FailureReasons API コールは、モニタリング ノードから情報を収集するために一度だけ呼び出されます。使用しているファイル システムまたはデータベースに、返された障害理由の内容を保存する必要があります。これらの API コールの返信内容はあくまでも参照用に使用することを目的としています。認証中に問題が発生した場合、認証応答で提供される障害理由コードと、ユーザのファイル システムまたはデータベースに保存している障害理由のリストと比較する必要があります。 Cisco ISE 障害理由の完全なリストについては、付録 A「障害理由レポートの使用方法」を参照してください。

表 1-2　Cisco ISE トラブルシューティング API コール - トラブルシューティング （続き）

API コール	説明と例
AuthStatus	すべてのセッションの認証ステータスをリストします。 https://<ISEhost>/ise/mnt/AuthStatus/MACAddress/<macaddress>/<numberofseconds>/<numberofrecordspermacaddress>/All （注）　seconds パラメータ <numberofseconds> は、0 秒から 432000 秒（5 日）の範囲でユーザが設定できます。
セッション アカウンティング ステータスの取得	
AcctStatus	特定の期間内のすべてのセッションのアカウンティング ステータスを示します。 https://<ISEhost>/ise/mnt/Session/AcctStatusTT/MACAddress/<macaddress>/<numberof seconds> （注）　seconds パラメータ <numberofseconds> は、0 秒から 432000 秒（5 日）の範囲でユーザが設定できます。

トラブルシューティング用の Cisco ISE API コールの詳細については、第 2 章「セッション管理用の API コールの使用方法」を参照してください。

表 1-3　Cisco ISE 認可変更 API コール

API コール	説明と例
Reauth	セッション再認証コマンドとタイプを送信します。 https://<ISEhost>/ise/mnt/CoA/Reauth/<serverhostname>/<macaddress>/<reauthtype>/<nasipaddress>/<destinationipaddress> ここで、<*ISEhost*> は ISE ホストの IP アドレスを示し、<*serverhostname*> は ISE サーバの名前を示し、<*nasipaddress*> は NAS の識別 IP アドレスを示し、<*destinationipaddress*> は宛先の IP アドレスを示します。 再認証タイプは次の値（0 ～ 2）のいずれかです。 0：REAUTH_TYPE_DEFAULT 1：REAUTH_TYPE_LAST 2：REAUTH_TYPE_RERUN （注）　NAS IP アドレスが不明な場合は、この時点までに必要な値を入力できます。API はこれらの値を検索クエリーに使用します。ただし、この API コールを実行するには、MAC アドレスを知っている必要がありますが、他のパラメータはヌルにしたままにできます。 この API コールは、CoA をリモートで実行する要求を送信する Monitoring ISE ノードでしか実行できません。Administration ISE ノードは、これらの CoA API コールの実行には関係ないか、必要がありません。

表 1-3　Cisco ISE 認可変更 API コール （続き）

API コール	説明と例
セッション切断	
Disconnect	セッション切断コマンドおよびポート オプション タイプを送信します。 https://<ISEhost>/ise/mnt/CoA/Disconnect/<serverhostname>/<macaddress>/<disconnecttype>/<nasipaddress>/<destinationipaddress> ポート オプション タイプは次の値（0 ～ 2）のいずれかです。 0：DYNAMIC_AUTHZ_PORT_DEFAULT 1：DYNAMIC_AUTHZ_PORT_BOUNCE 2：DYNAMIC_AUTHZ_PORT_SHUTDOWN （注）　NAS IP アドレスが不明な場合は、この時点までに必要な値を入力します。API はこれらの値を検索クエリーに使用します。ただし、この API コールを実行するには、MAC アドレスを知っている必要がありますが、他のパラメータはヌルにしたままにできます。

Cisco ISE 認可変更 API コールに関する詳細については、第 4 章「認可変更 REST API の使用方法」を参照してください。

HTTP PUT API コール

表 1-2 の AuthStatus API コールと同様に、クライアントがアカウント ステータスを取得できるようにする API コールの HTTP PUT バージョンがあります。Monitoring REST API は、HTTP GET コールについて記述したこのマニュアルの例で示すように、HTTP PUT と HTTP GET の両方のコールをサポートします。HTTP PUT は、パラメータの入力が必要なコールの必要性に対処します。次のスキーマ ファイルの例は、アカウント ステータスの要求です。

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<xs:schema version="1.0" xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="acctRequest" type="mnTRESTAcctRequest"/>

  <xs:complexType name="mnTRESTAcctRequest">
    <xs:complexContent>
      <xs:extension base="mnTRESTRequest">
        <xs:sequence>
          <xs:element name="duration" type="xs:string" minOccurs="0"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="mnTRESTRequest" abstract="true">
    <xs:sequence>
      <xs:element name="valueList">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="value" type="xs:string" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="searchCriteria" type="xs:string"/>
```

```
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```