



Cisco Identity Services Engine API, Release 2.x リ ファレンス ガイド

Cisco Systems, Inc.
www.cisco.com

シスコは世界各国 200 箇所にオフィスを開設しています。
所在地、電話番号、FAX 番号
は以下のシスコ Web サイトをご覧ください。
www.cisco.com/go/offices

**【注意】 シスコ製品をご使用になる前に、安全上の注意
(www.cisco.com/jp/go/safety_warning/) をご確認ください。**

本書は、米国シスコシステムズ発行ドキュメントの参考和訳です。
リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動 / 変更されている場合がありますことをご了承ください。
あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。

また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任になります。

対象製品のソフトウェア ライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

ここに記載されている他のいかなる保証によらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコおよびこれら各社は、商品性の保証、特定目的への準拠の保証、および権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコまたはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

CCDE, CCVP, Cisco Eos, Cisco StadiumVision, the Cisco logo, DCE, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn is a service mark; and Access Registrar, Aironet, AsyncOS, Bringing the Meeting To You, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Collaboration Without Limitation, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, IronPort, the IronPort logo, LightStream, Linksys, MediaTone, MeetingPlace, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerPanels, ProConnect, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0801R)

このマニュアルで使用している IP アドレスは、実際のアドレスを示すものではありません。マニュアル内の例、コマンド出力、および図は、説明のみを目的として使用されています。説明の中に実際のアドレスが使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

Cisco Identity Services Engine API, Release 2.x リファレンス ガイド
© 2017 年 Cisco Systems, Inc. All rights reserved.



はじめに	vii	
Cisco Identity Services Engine の概要		vii
目的	viii	
対象読者	viii	
マニュアルの表記法	viii	
関連資料	ix	
プラットフォーム別のマニュアル		iii-ix
マニュアルの入手方法およびテクニカル サポート		x

PART 1

Cisco ISE Monitoring REST API

CHAPTER 1

Monitoring REST API の概要	1-1
モニタリング ノードの確認	1-2
サポートされる API コール	1-2
HTTP PUT API コール	1-8

CHAPTER 2

セッション管理クエリー API	2-1
セッション カウンタ API コール	2-1
アクティブ セッション カウンタ	2-1
ActiveCount API の出力スキーマ	2-1
ActiveCount API コールの呼び出し	2-2
ActiveCount API コールから返されるサンプル データ	2-2
ポストチャ セッション カウンタ	2-3
PostureCount API の出力スキーマ	2-3
PostureCount API コールの呼び出し	2-3
PostureCount API コールから返されるサンプル データ	2-4
プロファイラ セッション カウンタ	2-4
ProfilerCount API の出力スキーマ	2-4
ProfilerCount API コールの呼び出し	2-4
ProfilerCount API コールから返されるサンプル データ	2-5
単純なセッション リスト API コール	2-5
アクティブなセッション リスト	2-5
ActiveList API の出力スキーマ	2-5

ActiveList API コールの呼び出し	2-6
ActiveList API コールから返されるサンプル データ	2-7
認証セッション リスト	2-8
AuthList API の出力スキーマ	2-8
AuthList API コールの呼び出し	2-8
null/null オプションを使用した AuthList API コールから返されるサンプル データ	2-9
endtime/null オプションを使用した AuthList API コールから返されるサンプル データ	2-10
null/starttime オプションを使用した AuthList API コールから返されるサンプル データ	2-11
starttime/endtime オプションを使用した AuthList API コールから返されるサンプル データ	2-12
詳細なセッション属性 API コール	2-12
MAC アドレス セッションの検索	2-13
MACAddress API の出力スキーマ	2-13
MACAddress API コール呼び出し	2-15
MACAddress API コールから返されるサンプル データ	2-15
ユーザ名のセッションの検索	2-17
UserName API の出力スキーマ	2-17
UserName API コール呼び出し	2-19
UserName API コールから返されるサンプル データ	2-20
NAS IP アドレス セッションの検索	2-21
IPAddress API の出力スキーマ	2-21
NAS IPAddress API コール呼び出し	2-23
IPAddress API コールから返されるサンプル データ	2-24
エンドポイントの IP アドレスのセッションの検索	2-25
EndPointIPAddress API の出力スキーマ	2-26
EndPointIPAddress API コール呼び出し	2-27
EndPointIPAddress API コールから返されるサンプル データ	2-28
監査セッション ID の検索	2-30
Audit Session ID API の出力スキーマ	2-30
Audit Session ID API コール呼び出し	2-32
Audit Session ID API コールから返されるサンプル データ	2-32
古いセッション	2-33
古いセッションの削除	2-33
CHAPTER 3	トラブルシューティング用のクエリー API 3-1
Cisco Prime NCS API コール	3-1
クエリー API を使用した Cisco ISE のトラブルシューティング	3-1

ノードのバージョンおよびタイプの API コール	3-1
Version API の出力スキーマ	3-2
Version API コールの呼び出し	3-2
Version API コールから返されるサンプルデータ	3-3
障害理由 API コール	3-3
FailureReasons API の出力スキーマ	3-4
FailureReasons API コールの呼び出し	3-4
FailureReasons API コールから返されるサンプルデータ	3-5
認証ステータス API コール	3-6
AuthStatus API の出力スキーマ	3-8
AuthStatus API コールの呼び出し	3-10
AuthStatus API コールから返されるサンプルデータ	3-11
アカウントステータス API コール	3-12
AcctStatus API の出力スキーマ	3-13
AcctStatus API コールの呼び出し	3-13
AcctStatus API コールから返されるサンプルデータ	3-14

CHAPTER 4

認可変更 REST API 4-1

はじめに	4-1
CoA セッション管理 API コール	4-1
セッション再認証 API コール	4-1
Reauth API の出力スキーマ	4-1
Reauth API コールの呼び出し	4-2
Reauth API コールから返されるサンプルデータ	4-2
セッション切断 API コール	4-3
Disconnect API の出力スキーマ	4-3
Disconnect API コールの呼び出し	4-3
Disconnect API コールから返されるサンプルデータ	4-4

PART 2

Cisco ISE 外部 RESTful サービス API

CHAPTER 5

ERS API の概要 5-1

外部 RESTful サービス API コールを使用するための前提条件	5-1
外部 RESTful サービス SDK	5-1
外部 RESTful サービス API の認証および承認	5-3

APPENDIX A

Cisco ISE 障害理由レポート A-1

はじめに	A-1
障害理由の表示	A-1



はじめに

- [Cisco Identity Services Engine の概要 \(vii ページ\)](#)
- [目的 \(viii ページ\)](#)
- [対象読者 \(viii ページ\)](#)
- [マニュアルの表記法 \(viii ページ\)](#)
- [関連資料 \(ix ページ\)](#)
- [マニュアルの入手方法およびテクニカル サポート \(x ページ\)](#)

Cisco Identity Services Engine の概要

Cisco Identity Services Engine (ISE) は、企業でのコンプライアンスの順守、インフラストラクチャのセキュリティの強化、サービス オペレーションの合理化を実現する、次世代のアイデンティティおよびアクセス コントロール ポリシーのプラットフォームです。Cisco ISE の固有のアーキテクチャにより、企業は、アクセス スイッチ、Wireless LAN Controller (WLC)、バーチャル プライベート ネットワーク (VPN) ゲートウェイ、およびデータセンター スイッチなど、さまざまなネットワーク要素に ID を結びつけることで予防的な管理を決定するために、ネットワーク、ユーザ、およびデバイスからリアルタイムのコンテキスト情報を収集することができるようになります。

Cisco ISE は Cisco Security Group Access Solution のキー コンポーネントです。Cisco ISE は、統合されたポリシーベースのアクセス コントロール ソリューションで以下を実現します。

- 認証、承認、アカウント (AAA)、ポスチャ、プロファイラ、ゲスト管理サービスを 1 つのアプリケーションに結合します。
- 802.1X 環境を含むネットワークにアクセスしているすべてのエンドポイントのデバイス ポスチャをチェックすることでエンドポイント コンプライアンスを徹底します。
- ネットワーク上のエンドポイント デバイスの検出、プロファイリング、ポリシーベースの配置、モニタリングのサポートを提供します。
- 集中型展開および分散型展開においてポリシーの一貫性が維持され、サービスを必要な場所に配信できるようになります。
- Security Group Tags (SGT) および Security Group (SG) Access Control List (ACL) によって Security Group Access (SGA) などの高度な強化機能を使用します。
- 小さな事務所から大企業まで様々な環境の展開シナリオに対応するスケーラビリティをサポートします。

Cisco ISE のアーキテクチャは、集中型ポータルからネットワークを設定して管理できるように、スタンドアロンの導入と分散型の導入をサポートします。Cisco ISE の機能の詳細については、『[Cisco Identity Services Engine Admin Guide](#)』を参照してください。

目的

このアプリケーションプログラミング インターフェイス (API) リファレンス ガイドは、サポート対象の API が提供する機能の概要だけを説明します。この API リファレンス ガイドの目的は、Cisco ISE 展開内で概説された API を使用するための基本的な注意事項を、開発者、システム管理者やネットワーク管理者、またはシステム インテグレータに提供することです。

REST API コールは、次の種類のデータを確認するためにクエリーを使用します。

- アクティブ セッションの数
- アクティブ セッションのタイプ
- アクティブ セッションの認証ステータス
- 使用中の MAC アドレス
- 使用中の NAS の IP アドレス
- ノードのバージョンとタイプ
- ノードのセッション障害の理由

外部 RESTful サービス API および関連 API コールは、Cisco ISE リソースに対して CRUD (作成、読み取り、更新、削除) 操作を実行するために使用できます。外部 RESTful サービスは HTTP プロトコルおよび REST 方法論に基づいています。



(注)

Cisco ISE ネットワークとそのノードおよびペルソナ、動作または用途の概念、Cisco ISE ユーザー インターフェイスの使用法の詳細については、『[Cisco Identity Services Engine Admin Guide](#)』を参照してください。

対象読者

この API リファレンス ガイドは、ネットワーク環境内で Cisco ISE アプライアンスを管理する経験豊富なシステム管理者、API を利用するシステム インテグレータ、Cisco ISE 導入の管理やトラブルシューティングの役割を持つサードパーティ製パートナーを対象としています。この API リファレンス ガイドを使用する前提条件として、トラブルシューティングと診断方法について、API コールの作成および解釈方法について、基礎を理解しておく必要があります。

マニュアルの表記法

ここでは、このマニュアル全体で使用されている表記法について説明します。



注意

「要注意」の意味です。機器の損傷またはデータ損失を予防するための注意事項が記載されています。



(注) 「注釈」です。役立つ情報や、このマニュアル以外の参照資料などを紹介しています。

この API リファレンス ガイドは次の表記法を使用して、指示と情報を示します。

項目	表記法
手順で選択する必要があるコマンド、キーワード、特殊な用語、およびオプション	太字
ユーザが値を指定する変数、および新しい用語や重要な用語	<i>italic</i> フォント
表示されるセッション情報、システム情報、パス、およびファイル名	Screen フォント
ユーザが入力する情報	太字の screen フォント
ユーザが入力する変数	イタリック体の <i>screen</i> フォント
メニュー項目およびボタン名	[]
選択する順序に並べられたメニュー項目	[オプション (Option)] > [ネットワーク設定 (Network Preferences)]

関連資料

ここでは、このリリースのマニュアルと、このプラットフォームのマニュアルの情報を提供します。

Cisco ISE の全般的な製品情報は <http://www.cisco.com/go/ise> で確認できます。エンドユーザ マニュアルは、Cisco.com の http://www.cisco.com/en/US/products/ps11640/tsd_products_support_series_home.html から入手できます。

プラットフォーム別のマニュアル

- Cisco Secure ACS
http://www.cisco.com/en/US/products/ps9911/tsd_products_support_series_home.html
- Cisco NAC Appliance
http://www.cisco.com/en/US/products/ps6128/tsd_products_support_series_home.html
- Cisco NAC Profiler
http://www.cisco.com/en/US/products/ps8464/tsd_products_support_series_home.html
- Cisco NAC Guest Server
http://www.cisco.com/en/US/products/ps10160/tsd_products_support_series_home.html

マニュアルの入手方法およびテクニカルサポート

マニュアルの入手方法、テクニカルサポート、その他の有用な情報について、次の URL で、毎月更新される『*What's New in Cisco Product Documentation*』を参照してください。シスコの新規および改訂版の技術マニュアルの一覧も示されています。

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>

『*What's New in Cisco Product Documentation*』は RSS フィードとして購読できます。また、リーダーアプリケーションを使用してコンテンツがデスクトップに直接配信されるように設定することもできます。RSS フィードは無料のサービスです。シスコは現在、RSS バージョン 2.0 をサポートしています。



PART 1

Cisco ISE Monitoring REST API



Monitoring REST API の概要

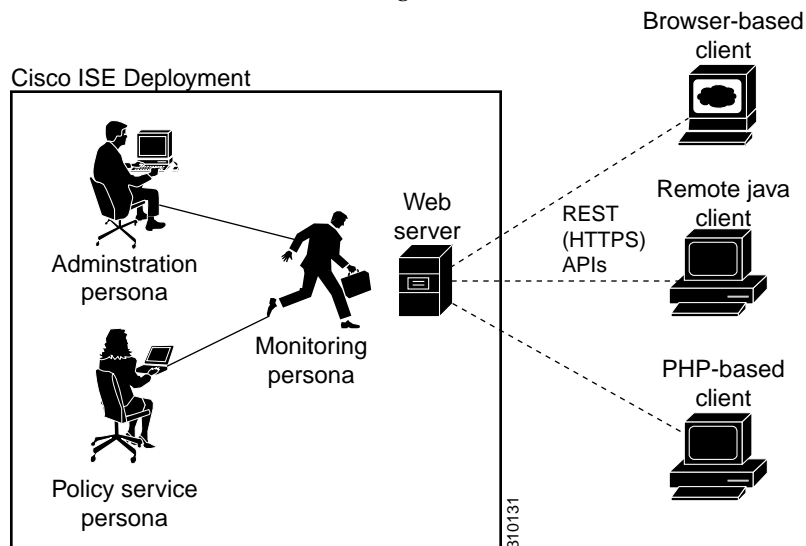
Monitoring REST API では、ネットワークでモニタリング ノードを使用して、セッションおよびノード固有の情報を収集することができます。セッションは、目的のノードにアクセスしてから情報の収集に必要な操作を完了するまでの期間として定義されます。

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



Monitoring REST API を使用して操作を実行するには、次の管理者グループのいずれかにユーザを割り当て、Cisco ISE の内部データベース (内部管理者ユーザ) に保存されているクレデンシャルに対して認証する必要があります。

- スーパー管理者
- システム管理者
- MnT 管理者

次の Monitoring REST API のカテゴリがサポートされています。

- セッション管理
- トラブルシューティング
- 認可変更 (CoA)

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、これらの API を使用できます。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、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 Admin Guide](#)』を参照してください。

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 \(1-3 ページ\)](#): セッション管理用の API コールを定義します。
- [表 1-2 \(1-6 ページ\)](#): 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 \(1-7 ページ\)](#): 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	<p>アクティブなセッションの数をリストします。</p> <p><code>https://<ISEhost>/admin/API/mnt/Session/ActiveCount</code></p> <p>(注) アクティブなセッションの数を表示するには、認証クレデンシャルのある HTTP 認証ヘッダーを追加する必要があります。</p>
PostureCount	<p>ポストチャされたエンドポイントの数をリストします。</p> <p><code>https://<ISEhost>/admin/API/mnt/Session/PostureCount</code></p> <p>(注) ポストチャとは、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(またはポストチャ)の確認を支援するサービスです。Cisco ISE は、デバイスのポスト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。</p>
ProfilerCount	<p>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。</p> <p><code>https://<ISEhost>/admin/API/mnt/Session/ProfilerCount</code></p> <p>(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。</p>
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	<p>すべてのアクティブなセッションをリストします。</p> <p><code>https://<ISEhost>/admin/API/mnt/Session/ActiveList</code></p> <p>(注) このリリースの Cisco ISE では、表示可能な認証されたエンドポイント セッションの最大数が 250,000 に制限されています。</p>

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
AuthList	<p>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。 <a href="https://<ISEhost>/admin/API/mnt/Session/AuthList/<parameteroptions>">https://<ISEhost>/admin/API/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。</p> <ul style="list-style-type: none"> • 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。 <p>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</p> <p>引数の説明</p> <ul style="list-style-type: none"> • YYYY: 4桁の年 • MM: 2桁の月 (01 = 1月など) • DD: 2桁の日 (01 ~ 31) • hh: 2桁の時間 (00 ~ 23) (a.m. および 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 <p>(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。</p> <p>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、null/null オプションを使用した AuthList API コールから返されるサンプル データ (2-9 ページ)を参照してください。</p>
セッション属性	<p>(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。</p>
MACAddress	<p>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。 <a href="https://<ISEhost>/admin/API/mnt/Session/MACAddress/<macaddress>">https://<ISEhost>/admin/API/mnt/Session/MACAddress/<macaddress></p> <p>(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)。</p> <p>(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。</p>

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
UserName	<p>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。</p> <p><code>https://<ISEhost>/admin/API/mnt/Session/UserName/<username></code></p> <p>(注) ユーザ名は、ネットワーク ユーザ名に使用しているのと同じ 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。</p>
IPAddress	<p>指定した NAS IP アドレス (IPv4 または IPv6 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。</p> <p><code>https://<ISEhost>/admin/API/mnt/Session/IPAddress/<nasipaddress></code></p> <p>(注) xxx.xxx.xxx.xxx は NAS IP アドレス形式 (例: 10.10.10.10) です。</p> <p>または</p> <p><code>https://<ISEhost>/admin/API/mnt/Session/IPAddress/<nasipv6address></code></p> <p>(注) xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx は NAS IPv6 アドレス形式です (例: 2001:cdba:0:0:0:0:3247:9651)。</p>
Audit Session ID	<p>指定した監査セッション ID 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。</p> <p><code>https://<ISEhost>/admin/API/mnt/Session/Active/SessionID/<audit-session-id>/0</code></p> <p>(注) 監査セッション ID 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監査セッション ID をリストするには ActiveList API コールを使用します。または、管理者ポータル の [ライブセッション (Live Sessions)] ページから監査セッション ID を取得できます。</p>

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「セッション管理クエリー API」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	<p>ノードのバージョンおよびタイプをリストします。</p> <p><code>https://<ISEhost>/admin/API/mnt/Version</code></p> <p>ノードのタイプは、次の値(0 ~ 3)のいずれかです。</p> <p>0: STAND_ALONE_MNT_NODE</p> <p>1: ACTIVE_MNT_NODE</p> <p>2: STAND_BY_MNT_NODE</p> <p>3: NOT_AN_MNT_NODE</p> <p>(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。</p> <p>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。</p> <p>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。</p> <p>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。</p>
FailureReasons	<p>障害の理由をリストします。</p> <p><code>https://<ISEhost>/admin/API/mnt/FailureReasons</code></p> <p>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。</p> <pre><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.</pre> <p>(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た障害理由のリストを比較する必要があります。</p> <p>Cisco ISE 障害理由の完全なリストについては、付録 A「Cisco ISE 障害理由レポート」を参照してください。</p>

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	<p>すべてのセッションの認証ステータスをリストします。</p> <p>https://<ISEhost>/admin/API/mnt/AuthStatus/MACAddress/<macaddress>/<numberofseconds>/<numberofrecordspermacaddress>/All</p> <p>(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。</p>
セッション アカウンティング ステータスの取得	
AcctStatus	<p>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。</p> <p>https://<ISEhost>/admin/API/mnt/AcctStatusTT/MACAddress/<macaddress>/<numberof seconds></p> <p>(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。</p>

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「セッション管理クエリー API」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	<p>セッション再認証コマンドとタイプを送信します。</p> <p>https://<ISEhost>/admin/API/mnt/CoA/Reauth/<serverhostname>/<macaddress>/<reauthtype>/<nasipaddress>/<destinationipaddress></p> <p>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。</p> <p>再認証タイプは次の値 (0 ~ 2) のいずれかです。</p> <p>0: REAUTH_TYPE_DEFAULT 1: REAUTH_TYPE_LAST 2: REAUTH_TYPE_RERUN</p> <p>(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、NAS IP アドレスで始まる他のパラメータはヌルにしたままにできます。NAS IP アドレスを指定する場合、宛先 IP アドレスも指定する必要があります。</p> <p>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。</p>
セッション切断	

表 1-3 Cisco ISE 認可変更 API コール(続き)

API コール	説明と例
<i>Disconnect</i>	<p>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。</p> <pre>https://<ISEhost>/admin/API/mnt/CoA/Disconnect/<serverhostname>/ <macaddress>/<disconnecttype>/<nasipaddress>/ <destinationipaddress></pre> <p>ポート オプション タイプは次の値(0 ~ 2)のいずれかです。</p> <p>0: DYNAMIC_AUTHZ_PORT_DEFAULT 1: DYNAMIC_AUTHZ_PORT_BOUNCE 2: DYNAMIC_AUTHZ_PORT_SHUTDOWN</p> <p>(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することができます。</p>

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」](#)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>
```



セッション管理クエリー API

この章では、Cisco ISE 展開において、Cisco Monitoring ISE ノード内から重要なセッション関連の情報を取得する手段を提供するセッション管理 API コールについて説明します。

セッションカウンタ API コール

次のセッションカウンタ API コールによって、Cisco ISE 展開におけるターゲット Cisco Monitoring ISE ノードのセッション関連情報の現在のカウントをすぐに収集できるようになります。

- アクティブセッション (ActiveCount) : アクティブセッションは、ネットワークで認証されるセッションの 1 つです。
- ポスチャセッション (PostureCount) : ポスチャが結論付けられる (準拠/非準拠) と、ポスチャ状態がアサートされます。ポスチャはオプションで、IP 電話やプリンタなどはポスチャ状態になりません。ポスチャ後、アカウンティングの開始が設定されると開始済み状態になるため、ポスチャ状態は短期間の一時的な状態です。
- プロファイルセッション (ProfilerCount)

いずれかのフェーズでエンドポイントが停止した場合、これらのさまざまな状態はトラブルシューティングが必要であることを示します。

アクティブセッションカウンタ

現在アクティブなすべてのセッションカウントを取得するために ActiveCount API コールを使用できます。



(注)

アクティブなセッションの数を表示するには、認証クレデンシャルのある HTTP 認証ヘッダーを追加する必要があります。

ActiveCount API の出力スキーマ

このサンプルスキーマファイルは、ISE のノードのターゲット Monitoring ペルソナでアクティブセッションのカウントを取得するための ActiveCount API コールの出力です。


```
<?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="sessionCount" type="activeCount"/>
<xs:complexType name="activeCount">
  <xs:sequence>
    <xs:element name="count" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>

```

ActiveCount API コールの呼び出し

-
- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
`https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash`
- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント (`/admin/API/mnt/<specific-api-call>`) に置き換えて、ターゲット ノードの URL アドレス フィールドに ActiveCount API コールを入力します。
`https://acme123/admin/API/mnt/Session/ActiveCount`
-  **(注)** これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、ターゲット Cisco Monitoring ISE ノードを表します。
-
- ステップ 5** **Enter** キーを押して API コールを発行します。
-

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

ActiveCount API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで ActiveCount API コールを呼び出すときに返されるデータ(アクティブセッション数)を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

-
<sessionCount>
<count>5</count>
</sessionCount>

```

ポスタチャ セッションカウンタ

現在アクティブなすべてのポスタチャセッションの現在のカウントを取得するために PostureCount API コールを使用できます。

PostureCount API の出力スキーマ

このサンプルスキーマファイルは、ターゲット Cisco Monitoring ISE ノードで現在アクティブなポスタチャセッションのカウントを取得するための PostureCount API コールの出力です。

```
<?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="sessionCount" type="postureCount"/>

  <xs:complexType name="postureCount">
    <xs:sequence>
      <xs:element name="count" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

PostureCount API コールの呼び出し

- ステップ 1 Cisco ISE URL をブラウザのアドレスバーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2 ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3 [ログイン(Login)] をクリックするか、**Enter** を押します。
たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
`https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash`
- ステップ 4 「/admin/」コンポーネントを API コールのコンポーネント (`/admin/API/mnt/Session/<specific-api-call>`) に置き換えて、ターゲット ノードの URL アドレス フィールドに PostureCount API コールを入力します。
`https://acme123/admin/API/mnt/Session/PostureCount`



(注) これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、ターゲット Cisco Monitoring ISE ノードを表します。

- ステップ 5 **Enter** キーを押して API コールを発行します。

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

PostureCount API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで PostureCount API コールを呼び出すときに返されるデータ(現在アクティブなポストチャセッション数)を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<sessionCount>
<count>3</count>
</sessionCount>
```

プロファイラセッションカウンタ

現在アクティブなすべてのプロファイラセッションカウントを取得するために ProfilerCount API コールを使用できます。

ProfilerCount API の出力スキーマ

このサンプルスキーマファイルは、ターゲット Cisco Monitoring ISE ノードで現在アクティブなプロファイラセッションのカウントを取得するための ProfilerCount API コールの出力です。

```
<?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="sessionCount" type="profilerCount"/>

  <xs:complexType name="profilerCount">
    <xs:sequence>
      <xs:element name="count" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

ProfilerCount API コールの呼び出し

-
- ステップ 1 Cisco ISE URL をブラウザのアドレスバーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
 - ステップ 2 ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
 - ステップ 3 [ログイン(Login)] をクリックするか、**Enter** を押します。
たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
`https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash`
 - ステップ 4 「/admin/」コンポーネントを API コールのコンポーネント (`/admin/API/mnt/Session/<specific-api-call>`) に置き換えて、ターゲット ノードの URL アドレスフィールドに ProfilerCount API コールを入力します。
`https://acme123/admin/API/mnt/Session/ProfilerCount`



(注) これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。

ステップ 5 Enter キーを押して API コールを発行します。

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

ProfilerCount API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで ProfilerCount API コールを呼び出すときに返されるデータ (現在アクティブなプロファイラ セッション数) を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-  
<sessionCount>  
<count>1</count>  
</sessionCount>
```

単純なセッション リスト API コール

次の単純なセッション リスト API コールによって、Cisco ISE 展開におけるターゲット Cisco Monitoring ISE ノードの現在のアクティブ セッションに関連付けられた 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 ID などのセッション関連の情報をすぐに収集できるようになります。

- アクティブなセッション リスト (ActiveList)
- 認証セッション リスト (AuthList)

アクティブなセッション リスト

現在アクティブなすべてのセッションをリストするには ActiveList API 呼び出しを使用できます。



(注) 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、100,000 に制限されています。

ActiveList API の出力スキーマ

このサンプル スキーマ ファイルは、ターゲット Cisco Monitoring ISE ノードで現在アクティブなセッション (およびセッション関連情報) のリストを取得するための ActiveList API コールの出力です。

```
<?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="activeSessionList" type="simpleActiveSessionList"/>


<xs:complexType name="simpleActiveSessionList">
  <xs:sequence>
    <xs:element name="activeSession" type="simpleActiveSession" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="noOfActiveSession" type="xs:int" use="required"/>
</xs:complexType>

<xs:complexType name="simpleActiveSession">
  <xs:sequence>
    <xs:element name="user_name" type="xs:string" minOccurs="0"/>
    <xs:element name="calling_station_id" type="xs:string" minOccurs="0"/>
    <xs:element name="nas_ip_address" type="xs:string" minOccurs="0"/>
    <xs:element name="acct_session_id" type="xs:string" minOccurs="0"/>
    <xs:element name="audit_session_id" type="xs:string" minOccurs="0"/>
    <xs:element name="server" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<xs:element name="nas_ipv6_address" type="xs:string"/>
  <xs:complexType name="framed_ipv6_address_list">
    <xs:sequence minOccurs="0" maxOccurs="8"><xs:element name="ipv6_address"
type="xs:string" />
    </xs:sequence>
  </xs:complexType>
<xs:element name="framed_ipv6_address" type="framed_ipv6_address_list" minOccurs="1"
maxOccurs="1"/>
</xs:schema>

```

ActiveList API コールの呼び出し

-
- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
`https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash`
- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント (`/admin/API/mnt/Session/<specific-api-call>`) に置き換えて、ターゲットノードの URL アドレスフィールドに ActiveList API コールを入力します。
`https://acme123/admin/API/mnt/Session/ActiveList`
-
-  (注) これらのコールは、大文字小文字を区別するため、ターゲットノードの URL アドレスフィールドに慎重に各 API 呼び出しを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。
-
- ステップ 5** **Enter** キーを押して API コールを発行します。
-

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

ActiveList API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで ActiveList API コールを呼び出すときにアクティブセッションのリストから返されるセッション関連データを示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<activeSessionList noOfActiveSession="5">
-
<activeSession>
<calling_station_id>00:0C:29:FA:EF:0A</calling_station_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<calling_station_id>70:5A:B6:68:F7:CC</calling_station_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>tom_wolfe</user_name>
<calling_station_id>00:14:BF:5A:0C:03</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::3257:9652</nas_ipv6_address>
<acct_session_id>00000032</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>graham_hancock</user_name>
<calling_station_id>00:50:56:8E:28:BD</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::3257:9652</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3257:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3257:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:9652</ipv6_address>
</framed_ipv6_address>
<acct_session_id>0000002C</acct_session_id>
<audit_session_id>0ACB6BA10000002A165FD0C8</audit_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>ipepvpnuser</user_name>
<calling_station_id>172.23.130.89</calling_station_id>
<nas_ip_address>10.203.107.45</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<acct_session_id>A2000070</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
</activeSessionList>
```

認証セッションリスト

現在アクティブなすべての認証セッションのリストを取得するために AuthList API コールを使用できます。



(注) 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、100,000 に制限されています。

AuthList API の出力スキーマ

このサンプル スキーマ ファイルは、ターゲット Cisco Monitoring ISE ノードでの、指定した期間内(または「null/null」パラメータを使用して期間を指定しない場合)現在アクティブなすべての認証セッションのリストを取得するための AuthList API コールの出力です。

```
<?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="activeSessionList" type="simpleActiveSessionList"/>

<xs:complexType name="simpleActiveSessionList">
  <xs:sequence>
    <xs:element name="activeSession" type="simpleActiveSession" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="noOfActiveSession" type="xs:int" use="required"/>
</xs:complexType>

<xs:complexType name="simpleActiveSession">
  <xs:sequence>
    <xs:element name="user_name" type="xs:string" minOccurs="0"/>
    <xs:element name="calling_station_id" type="xs:string" minOccurs="0"/>
    <xs:element name="nas_ip_address" type="xs:string" minOccurs="0"/>
    <xs:element name="acct_session_id" type="xs:string" minOccurs="0"/>
    <xs:element name="audit_session_id" type="xs:string" minOccurs="0"/>
    <xs:element name="server" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

  <xs:element name="nas_ipv6_address" type="xs:string"/>
  <xs:complexType name="framed_ipv6_address_list">
    <xs:sequence minOccurs="0" maxOccurs="8"><xs:element name="ipv6_address"
type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="framed_ipv6_address" type="framed_ipv6_address_list" minOccurs="1"
maxOccurs="1"/>

</xs:schema>
```

AuthList API コールの呼び出し

- ステップ 1 Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2 ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。

ステップ 3 [ログイン(Login)] をクリックするか、**Enter** を押します。

たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。

```
https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash
```

ステップ 4 「/admin/」コンポーネントを API コールのコンポーネント (/admin/API/mnt/Session/<specific-api-call>) に置き換えて、ターゲット ノードの URL アドレス フィールドに AuthList API コールを入力します。



(注) 次の 2 種類の例では、定義済みの開始時刻パラメータおよび null パラメータを使用し、開始時刻以降に認証された現在アクティブなセッションのリストを表示します。2 番目の例は、現在アクティブなすべての認証済みセッションのリストを表示する「null/null」パラメータを使用します。この API コールに対する 4 種類のパラメータ設定の例については、[null/null オプションを使用した AuthList API コールから返されるサンプル データ \(2-9 ページ\)](#) を参照してください。

```
https://acme123/admin/API/mnt/Session/AuthList/2010-12-14 15:33:15/null
```

```
https://acme123/admin/API/mnt/Session/AuthList/null/null
```



(注) これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。

ステップ 5 **Enter** キーを押して API コールを発行します。

関連項目

- [モニタリング ノードの確認 \(1-2 ページ\)](#)

null/null オプションを使用した AuthList API コールから返されるサンプル データ

次に、null/null オプションを使用して AuthList API コールを呼び出した場合に返される現在アクティブな認証済みセッションのリストの例を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<activeSessionList noOfActiveSession="3">
-
<activeSession>
<user_name>ipepwluser</user_name>
<calling_station_id>00:26:82:7B:D2:51</calling_station_id>
<nas_ip_address>10.203.107.10</nas_ip_address>
<nas_ipv6_address>2001:cdba::3257:9652</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3257:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3257:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:9652</ipv6_address>
</framed_ipv6_address>
<audit_session_id>0acb6b0c000000174D07F487</audit_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
```

```

<activeSession>
<user_name>tom_wolfe</user_name>
<calling_station_id>00:50:56:8E:28:BD</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<acct_session_id>00000035</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>graham_hancock</user_name>
<calling_station_id>00:14:BF:5A:0C:03</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<acct_session_id>00000033</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
</activeSessionList>

```

endtime/null オプションを使用した AuthList API コールから返されるサンプルデータ

次に、endtime/null オプションを使用して AuthList API コールを呼び出した場合に返される現在アクティブな認証済みセッションのリストの例を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

-
<activeSessionList noOfActiveSession="3">
-
<activeSession>
<user_name>ipepwluser</user_name>
<calling_station_id>00:26:82:7B:D2:51</calling_station_id>
<nas_ip_address>10.203.107.10</nas_ip_address>
<nas_ipv6_address>2001:cdba::3257:9652</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3257:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3257:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:9652</ipv6_address>
</framed_ipv6_address>
<audit_session_id>0acb6b0c0000001F4D08085A</audit_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>hunter_thompson</user_name>
<calling_station_id>00:50:56:8E:28:BD</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>

```

```

<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<acct_session_id>00000035</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>bob_ludlum</user_name>
<calling_station_id>00:14:BF:5A:0C:03</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<acct_session_id>00000033</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
</activeSessionList>

```

null/starttime オプションを使用した AuthList API コールから返されるサンプルデータ

次に、null/starttime オプションを使用して AuthList API コールを呼び出した場合に返される現在アクティブな認証済みセッションのリストの例を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

-
<activeSessionList noOfActiveSession="3">
-
<activeSession>
<user_name>ipepwluser</user_name>
<calling_station_id>00:26:82:7B:D2:51</calling_station_id>
<nas_ip_address>10.203.107.10</nas_ip_address>
<nas_ipv6_address>2001:cdba::3257:9652</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3257:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3257:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:9652</ipv6_address>
</framed_ipv6_address>
<audit_session_id>0acb6b0c0000001F4D08085A</audit_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>bob_ludlum</user_name>
<calling_station_id>00:50:56:8E:28:BD</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<acct_session_id>00000035</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>tom_wolfe</user_name>

```

```

<calling_station_id>00:14:BF:5A:0C:03</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<acct_session_id>00000033</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
</activeSessionList>

```

starttime/endtime オプションを使用した AuthList API コールから返されるサンプルデータ

次に、starttime/endtime オプションを使用して AuthList API コールを呼び出した場合に返される現在アクティブな認証済みセッションのリストの例を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

-
<activeSessionList noOfActiveSession="3">
-
<activeSession>
<user_name>ipepwluser</user_name>
<calling_station_id>00:26:82:7B:D2:51</calling_station_id>
<nas_ip_address>10.203.107.10</nas_ip_address>
<audit_session_id>0acb6b0c0000001F4D08085A</audit_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>graham_hancock</user_name>
<calling_station_id>00:50:56:8E:28:BD</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<acct_session_id>00000035</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
-
<activeSession>
<user_name>hunter_thompson</user_name>
<calling_station_id>00:14:BF:5A:0C:03</calling_station_id>
<nas_ip_address>10.203.107.161</nas_ip_address>
<acct_session_id>00000033</acct_session_id>
<server>HAREESH-R6-1-PDP2</server>
</activeSession>
</activeSessionList>

```

詳細なセッション属性 API コール

次の詳細なセッション属性 API コールによって、次のようなキー情報の最新のセッションをすぐに検索することができるようになります。

- MAC アドレス セッションの検索 (MACAddress)
- ユーザ名のセッションの検索 (UserName)
- NAS IP アドレス セッションの検索 (ターゲット Monitoring ISE ノードに関連付けられた IP アドレス)

- エンドポイントの IP アドレスのセッションの検索 (EndPointIPAddress)
- 監査セッション ID の検索 (Audit Session ID)

MAC アドレス セッションの検索

現在のアクティブなセッションから指定された MAC アドレスを取得するために MACAddress API コールを使用できます。この API コールは、ノードデータベース テーブルから供給されるさまざまなセッション関連の情報をリストします。

MACAddress API の出力スキーマ

このサンプル スキーマ ファイルは、現在アクティブなセッションから指定された MAC アドレスを取得するための MACAddress API コールの出力です。

```
<?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="sessionParameters" type="restsdStatus"/>

  <xs:complexType name="restsdStatus">
    <xs:sequence>
      <xs:element name="passed" type="xs:anyType" minOccurs="0"/>
      <xs:element name="failed" type="xs:anyType" minOccurs="0"/>
      <xs:element name="user_name" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_ip_address" type="xs:string" minOccurs="0"/>
      <xs:element name="failure_reason" type="xs:string" minOccurs="0"/>
      <xs:element name="calling_station_id" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_port" type="xs:string" minOccurs="0"/>
      <xs:element name="identity_group" type="xs:string" minOccurs="0"/>
      <xs:element name="network_device_name" type="xs:string" minOccurs="0"/>
      <xs:element name="acs_server" type="xs:string" minOccurs="0"/>
      <xs:element name="authen_protocol" type="xs:string" minOccurs="0"/>
      <xs:element name="framed_ip_address" type="xs:string" minOccurs="0"/>
      <xs:element name="network_device_groups" type="xs:string" minOccurs="0"/>
      <xs:element name="access_service" type="xs:string" minOccurs="0"/>
      <xs:element name="auth_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
      <xs:element name="authentication_method" type="xs:string" minOccurs="0"/>
      <xs:element name="execution_steps" type="xs:string" minOccurs="0"/>
      <xs:element name="radius_response" type="xs:string" minOccurs="0"/>
      <xs:element name="audit_session_id" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_identifier" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_port_id" type="xs:string" minOccurs="0"/>
      <xs:element name="nac_policy_compliance" type="xs:string" minOccurs="0"/>
      <xs:element name="auth_id" type="xs:long" minOccurs="0"/>
      <xs:element name="auth_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
      <xs:element name="message_code" type="xs:string" minOccurs="0"/>
      <xs:element name="acs_session_id" type="xs:string" minOccurs="0"/>
      <xs:element name="service_selection_policy" type="xs:string" minOccurs="0"/>
      <xs:element name="authorization_policy" type="xs:string" minOccurs="0"/>
      <xs:element name="identity_store" type="xs:string" minOccurs="0"/>
      <xs:element name="response" type="xs:string" minOccurs="0"/>
      <xs:element name="service_type" type="xs:string" minOccurs="0"/>
      <xs:element name="cts_security_group" type="xs:string" minOccurs="0"/>
      <xs:element name="use_case" type="xs:string" minOccurs="0"/>
      <xs:element name="cisco_av_pair" type="xs:string" minOccurs="0"/>
      <xs:element name="ad_domain" type="xs:string" minOccurs="0"/>
      <xs:element name="acs_username" type="xs:string" minOccurs="0"/>
      <xs:element name="radius_username" type="xs:string" minOccurs="0"/>
      <xs:element name="nac_role" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```

<xs:element name="nac_username" type="xs:string" minOccurs="0"/>
<xs:element name="nac_posture_token" type="xs:string" minOccurs="0"/>
<xs:element name="nac_radius_is_user_auth" type="xs:string" minOccurs="0"/>
<xs:element name="selected_posture_server" type="xs:string" minOccurs="0"/>
<xs:element name="selected_identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="authentication_identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="azn_exp_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="ext_pol_server_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="grp_mapping_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="identity_policy_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port_type" type="xs:string" minOccurs="0"/>
<xs:element name="query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="selected_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="sel_exp_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="selected_query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="eap_tunnel" type="xs:string" minOccurs="0"/>
<xs:element name="tunnel_details" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_ssg_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="other_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="response_time" type="xs:long" minOccurs="0"/>
<xs:element name="nad_failure" type="xs:anyType" minOccurs="0"/>
<xs:element name="destination_ip_address" type="xs:string" minOccurs="0"/>
<xs:element name="acct_id" type="xs:long" minOccurs="0"/>
<xs:element name="acct_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_status_type" type="xs:string" minOccurs="0"/>
<xs:element name="acct_session_time" type="xs:long" minOccurs="0"/>
<xs:element name="acct_input_octets" type="xs:string" minOccurs="0"/>
<xs:element name="acct_output_octets" type="xs:string" minOccurs="0"/>
<xs:element name="acct_input_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_output_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_class" type="xs:string" minOccurs="0"/>
<xs:element name="acct_terminate_cause" type="xs:string" minOccurs="0"/>
<xs:element name="acct_multi_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_authentic" type="xs:string" minOccurs="0"/>
<xs:element name="termination_action" type="xs:string" minOccurs="0"/>
<xs:element name="session_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="idle_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="acct_interim_interval" type="xs:string" minOccurs="0"/>
<xs:element name="acct_delay_time" type="xs:string" minOccurs="0"/>
<xs:element name="event_timestamp" type="xs:string" minOccurs="0"/>
<xs:element name="acct_tunnel_connection" type="xs:string" minOccurs="0"/>
<xs:element name="acct_tunnel_packet_lost" type="xs:string" minOccurs="0"/>
<xs:element name="security_group" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_setup_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_connect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_disconnect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="framed_protocol" type="xs:string" minOccurs="0"/>
<xs:element name="started" type="xs:anyType" minOccurs="0"/>
<xs:element name="stopped" type="xs:anyType" minOccurs="0"/>
<xs:element name="ckpt_id" type="xs:long" minOccurs="0"/>
<xs:element name="type" type="xs:long" minOccurs="0"/>
<xs:element name="nad_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="vlan" type="xs:string" minOccurs="0"/>
<xs:element name="dacl" type="xs:string" minOccurs="0"/>
<xs:element name="authentication_type" type="xs:string" minOccurs="0"/>
<xs:element name="interface_name" type="xs:string" minOccurs="0"/>
<xs:element name="reason" type="xs:string" minOccurs="0"/>
<xs:element name="endpoint_policy" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>



```

```

<xs:element name="nas_ipv6_address" type="xs:string"/>
<xs:complexType name="framed_ipv6_address_list">
  <xs:sequence minOccurs="0" maxOccurs="8"><xs:element name="ipv6_address"
type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:element name="framed_ipv6_address" type="framed_ipv6_address_list" minOccurs="1"
maxOccurs="1"/>
</xs:schema>

```

MACAddress API コールの呼び出し

- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
`https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash`
- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント (/admin/API/mnt/<specific-api-call>/<macaddress>) に置き換えて、ターゲット ノードの URL アドレス フィールドに MACAddress API コールを入力します。
`https://acme123/admin/API/mnt/Session/MACAddress/0A:0B:0C:0D:0E:0F`
-  (注) `XX:XX:XX:XX:XX:XX` 形式を使用して MAC アドレスを指定していることを確認します。
-  (注) これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。
- ステップ 5** **Enter** キーを押して API コールを発行します。

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

MACAddress API コールから返されるサンプルデータ

次に、MACAddress API コールを呼び出すときにアクティブセッションのリストから返されるセッション関連データの例を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

-
<sessionParameters>

```

```

<passed xsi:type="xs:boolean">true</passed>
<failed xsi:type="xs:boolean">>false</failed>
<user_name>hunter_thompson</user_name>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<calling_station_id>00:14:BF:5A:0C:03</calling_station_id>
<nas_port>50115</nas_port>
<identity_group>Profiled</identity_group>
<network_device_name>Core-Switch</network_device_name>
<acs_server>HAREESH-R6-1-PDP2</acs_server>
<authn_protocol>Lookup</authn_protocol>
-
<network_device_groups>
Device Type#All Device Types,Location#All Locations
</network_device_groups>
<access_service>RADIUS</access_service>
<auth_acs_timestamp>2010-12-15T02:11:12.359Z</auth_acs_timestamp>
<authentication_method>mab</authentication_method>
-
<execution_steps>
11001,11017,11027,15008,15048,15004,15041,15004,15013,24209,24211,22037,15036,15048,15048,
15004,15016,11022,11002
</execution_steps>
<audit_session_id>0ACB6BA1000000351BBFBF8B</audit_session_id>
<nas_port_id>GigabitEthernet1/0/15</nas_port_id>
<nac_policy_compliance>Pending</nac_policy_compliance>
<auth_id>1291240762077361</auth_id>
<auth_acsview_timestamp>2010-12-15T02:11:12.360Z</auth_acsview_timestamp>
<message_code>5200</message_code>
<acs_session_id>HAREESH-R6-1-PDP2/81148292/681</acs_session_id>
<service_selection_policy>MAB</service_selection_policy>
<identity_store>Internal Hosts</identity_store>
-
<response>
{UserName=00-14-BF-5A-0C-03; User-Name=00-14-BF-5A-0C-03;
State=ReauthSession:0ACB6BA1000000351BBFBF8B;
Class=CACS:0ACB6BA1000000351BBFBF8B:HAREESH-R6-1-PDP2/81148292/681;
Termination-Action=RADIUS-Request; cisco-av-pair=url-redirect-acl=ACL-WEBAUTH-REDIRECT;
cisco-av-pair=url-redirect=https://HAREESH-R6-1-PDP2.cisco.com:8443/guestportal/gateway?se
ssionId=0ACB6BA1000000351BBFBF8B&action=cwa;
cisco-av-pair=ACS:CiscoSecure-Defined-ACL=#ACSACL#-IP-ACL-DENY-4ced8390; }
</response>
<service_type>Call Check</service_type>
<use_case>Host Lookup</use_case>
<cisco_av_pair>audit-session-id=0ACB6BA1000000351BBFBF8B</cisco_av_pair>
<acs_username>00:14:BF:5A:0C:03</acs_username>
<radius_username>00:14:BF:5A:0C:03</radius_username>
<selected_identity_store>Internal Hosts</selected_identity_store>
<authentication_identity_store>Internal Hosts</authentication_identity_store>
<identity_policy_matched_rule>Default</identity_policy_matched_rule>
<nas_port_type>Ethernet</nas_port_type>
<selected_azn_profiles>CWA</selected_azn_profiles>
-
<other_attributes>
ConfigVersionId=44, DestinationIPAddress=10.203.107.162, DestinationPort=1812, Protocol=Radiu
s, Framed-MTU=1500, EAP-Key-Name=, CPMSessionID=0ACB6BA1000000351BBFBF8B, CPMSessionID=0ACB6BA
1000000351BBFBF8B, EndPointMACAddress=00-14-BF-5A-0C-03, HostIdentityGroup=Endpoint Identity

```

```

Groups:Profiled,Device Type=Device Type#All Device Types,Location=Location#All
Locations,Model Name=Unknown,Software Version=Unknown,Device IP
Address=10.203.107.161,Called-Station-ID=04:FE:7F:7F:C0:8F
</other_attributes>
<response_time>77</response_time>
<acct_id>1291240762077386</acct_id>
<acct_acs_timestamp>2010-12-15T02:12:30.779Z</acct_acs_timestamp>
<acct_acsview_timestamp>2010-12-15T02:12:30.780Z</acct_acsview_timestamp>
<acct_session_id>00000038</acct_session_id>
<acct_status_type>Interim-Update</acct_status_type>
<acct_session_time>78</acct_session_time>
<acct_input_octets>13742</acct_input_octets>
<acct_output_octets>6277</acct_output_octets>
<acct_input_packets>108</acct_input_packets>
<acct_output_packets>66</acct_output_packets>
-
<acct_class>
CACS:0ACB6BA1000000351BBFBF8B:HAREESH-R6-1-PDP2/81148292/681
</acct_class>
<acct_delay_time>0</acct_delay_time>
<started xsi:type="xs:boolean">false</started>
<stopped xsi:type="xs:boolean">false</stopped>
</sessionParameters>

```

ユーザ名のセッションの検索

現在のアクティブなセッションから指定されたユーザ名を取得するために **UserName API** コールを使用できます。この API は、ノードデータベース テーブルから供給されるさまざまなセッション関連の情報をリストします。

UserName API の出力スキーマ

このサンプル スキーマ ファイルは、現在アクティブなセッションから指定されたユーザ名を取得するための **UserName API** コールの出力です。

```

<?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="sessionParameters" type="restsdStatus"/>

  <xs:complexType name="restsdStatus">
    <xs:sequence>
      <xs:element name="passed" type="xs:anyType" minOccurs="0"/>
      <xs:element name="failed" type="xs:anyType" minOccurs="0"/>
      <xs:element name="user_name" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_ip_address" type="xs:string" minOccurs="0"/>
      <xs:element name="failure_reason" type="xs:string" minOccurs="0"/>
      <xs:element name="calling_station_id" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_port" type="xs:string" minOccurs="0"/>
      <xs:element name="identity_group" type="xs:string" minOccurs="0"/>
      <xs:element name="network_device_name" type="xs:string" minOccurs="0"/>
      <xs:element name="acs_server" type="xs:string" minOccurs="0"/>
      <xs:element name="authen_protocol" type="xs:string" minOccurs="0"/>
      <xs:element name="framed_ip_address" type="xs:string" minOccurs="0"/>
      <xs:element name="network_device_groups" type="xs:string" minOccurs="0"/>
      <xs:element name="access_service" type="xs:string" minOccurs="0"/>
      <xs:element name="auth_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
      <xs:element name="authentication_method" type="xs:string" minOccurs="0"/>
      <xs:element name="execution_steps" type="xs:string" minOccurs="0"/>
      <xs:element name="radius_response" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

```

```

<xs:element name="audit_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="nas_identifier" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port_id" type="xs:string" minOccurs="0"/>
<xs:element name="nac_policy_compliance" type="xs:string" minOccurs="0"/>
<xs:element name="auth_id" type="xs:long" minOccurs="0"/>
<xs:element name="auth_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="message_code" type="xs:string" minOccurs="0"/>
<xs:element name="acs_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="service_selection_policy" type="xs:string" minOccurs="0"/>
<xs:element name="authorization_policy" type="xs:string" minOccurs="0"/>
<xs:element name="identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="response" type="xs:string" minOccurs="0"/>
<xs:element name="service_type" type="xs:string" minOccurs="0"/>
<xs:element name="cts_security_group" type="xs:string" minOccurs="0"/>
<xs:element name="use_case" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_av_pair" type="xs:string" minOccurs="0"/>
<xs:element name="ad_domain" type="xs:string" minOccurs="0"/>
<xs:element name="acs_username" type="xs:string" minOccurs="0"/>
<xs:element name="radius_username" type="xs:string" minOccurs="0"/>
<xs:element name="nac_role" type="xs:string" minOccurs="0"/>
<xs:element name="nac_username" type="xs:string" minOccurs="0"/>
<xs:element name="nac_posture_token" type="xs:string" minOccurs="0"/>
<xs:element name="nac_radius_is_user_auth" type="xs:string" minOccurs="0"/>
<xs:element name="selected_posture_server" type="xs:string" minOccurs="0"/>
<xs:element name="selected_identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="authentication_identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="azn_exp_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="ext_pol_server_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="grp_mapping_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="identity_policy_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port_type" type="xs:string" minOccurs="0"/>
<xs:element name="query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="selected_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="sel_exp_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="selected_query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="eap_tunnel" type="xs:string" minOccurs="0"/>
<xs:element name="tunnel_details" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_ssg_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="other_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="response_time" type="xs:long" minOccurs="0"/>
<xs:element name="nad_failure" type="xs:anyType" minOccurs="0"/>
<xs:element name="destination_ip_address" type="xs:string" minOccurs="0"/>
<xs:element name="acct_id" type="xs:long" minOccurs="0"/>
<xs:element name="acct_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_status_type" type="xs:string" minOccurs="0"/>
<xs:element name="acct_session_time" type="xs:long" minOccurs="0"/>
<xs:element name="acct_input_octets" type="xs:string" minOccurs="0"/>
<xs:element name="acct_output_octets" type="xs:string" minOccurs="0"/>
<xs:element name="acct_input_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_output_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_class" type="xs:string" minOccurs="0"/>
<xs:element name="acct_terminate_cause" type="xs:string" minOccurs="0"/>
<xs:element name="acct_multi_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_authentic" type="xs:string" minOccurs="0"/>
<xs:element name="termination_action" type="xs:string" minOccurs="0"/>
<xs:element name="session_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="idle_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="acct_interim_interval" type="xs:string" minOccurs="0"/>
<xs:element name="acct_delay_time" type="xs:string" minOccurs="0"/>
<xs:element name="event_timestamp" type="xs:string" minOccurs="0"/>
<xs:element name="acct_tunnel_connection" type="xs:string" minOccurs="0"/>

```


```

<xs:element name="acct_tunnel_packet_lost" type="xs:string" minOccurs="0"/>
<xs:element name="security_group" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_setup_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_connect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_disconnect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="framed_protocol" type="xs:string" minOccurs="0"/>
<xs:element name="started" type="xs:anyType" minOccurs="0"/>
<xs:element name="stopped" type="xs:anyType" minOccurs="0"/>
<xs:element name="ckpt_id" type="xs:long" minOccurs="0"/>
<xs:element name="type" type="xs:long" minOccurs="0"/>
<xs:element name="nad_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="vlan" type="xs:string" minOccurs="0"/>
<xs:element name="dacl" type="xs:string" minOccurs="0"/>
<xs:element name="authentication_type" type="xs:string" minOccurs="0"/>
<xs:element name="interface_name" type="xs:string" minOccurs="0"/>
<xs:element name="reason" type="xs:string" minOccurs="0"/>
<xs:element name="endpoint_policy" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

<xs:element name="nas_ipv6_address" type="xs:string"/>
<xs:complexType name="framed_ipv6_address_list">
  <xs:sequence minOccurs="0" maxOccurs="8"><xs:element name="ipv6_address"
type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:element name="framed_ipv6_address" type="framed_ipv6_address_list" minOccurs="1"
maxOccurs="1"/>
</xs:schema>

```

UserName API コールの呼び出し

-
- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば *https://<ise hostname or ip address>/admin/*)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
たとえば、ホスト名が *acme123* の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash
- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント (/admin/API/mnt/<specific-api-call>/<username>) に置き換えて、ターゲット ノードの URL アドレス フィールドに UserName API コールを入力します。
https://acme123/admin/API/mnt/Session/UserName/graham_hancock
-  **(注)** これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。
-
- ステップ 5** **Enter** キーを押して API コールを発行します。
-

関連項目

- ・ [モニタリング ノードの確認\(1-2 ページ\)](#)

UserName API コールから返されるサンプルデータ

次に、UserName API コールを呼び出すときにアクティブセッションのリストから返されるセッション関連データの例を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

-
<sessionParameters>
<passed xsi:type="xs:boolean">true</passed>
<failed xsi:type="xs:boolean">false</failed>
<user_name>graham_hancock</user_name>
<nas_ip_address>10.203.107.161</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<calling_station_id>00:14:BF:5A:0C:03</calling_station_id>
<nas_port>50115</nas_port>
<identity_group>Profiled</identity_group>
<network_device_name>Core-Switch</network_device_name>
<acs_server>HAREESH-R6-1-PDP2</acs_server>
<authen_protocol>Lookup</authen_protocol>
-
<network_device_groups>
Device Type#All Device Types,Location#All Locations
</network_device_groups>
<access_service>RADIUS</access_service>
<auth_acs_timestamp>2010-12-15T02:11:12.359Z</auth_acs_timestamp>
<authentication_method>mab</authentication_method>
-
<execution_steps>
11001,11017,11027,15008,15048,15004,15041,15004,15013,24209,24211,22037,15036,15048,15048,
15004,15016,11022,11002
</execution_steps>
<audit_session_id>0ACB6BA1000000351BBFBF8B</audit_session_id>
<nas_port_id>GigabitEthernet1/0/15</nas_port_id>
<nac_policy_compliance>Pending</nac_policy_compliance>
<auth_id>1291240762077361</auth_id>
<auth_acsview_timestamp>2010-12-15T02:11:12.360Z</auth_acsview_timestamp>
<message_code>5200</message_code>
<acs_session_id>HAREESH-R6-1-PDP2/81148292/681</acs_session_id>
<service_selection_policy>MAB</service_selection_policy>
<identity_store>Internal Hosts</identity_store>
-
<response>
{UserName=graham_hancock; User-Name=graham_hancock;
State=ReauthSession:0ACB6BA1000000351BBFBF8B;
Class=CACS:0ACB6BA1000000351BBFBF8B:HAREESH-R6-1-PDP2/81148292/681;
Termination-Action=RADIUS-Request; cisco-av-pair=url-redirect-acl=ACL-WEBAUTH-REDIRECT;
cisco-av-pair=url-redirect=https://HAREESH-R6-1-PDP2.cisco.com:8443/guestportal/gateway?se
ssionId=0ACB6BA1000000351BBFBF8B&action=cwa;
cisco-av-pair=ACS:CiscoSecure-Defined-ACL=#ACSACL#-IP-ACL-DENY-4ced8390; }
</response>
<service_type>Call Check</service_type>
<use_case>Host Lookup</use_case>

```



```

<cisco_av_pair>audit-session-id=0ACB6BA1000000351BBFBF8B</cisco_av_pair>
<acs_username>graham_hancock</acs_username>
<radius_username>00:14:BF:5A:0C:03</radius_username>
<selected_identity_store>Internal Hosts</selected_identity_store>
<authentication_identity_store>Internal Hosts</authentication_identity_store>
<identity_policy_matched_rule>Default</identity_policy_matched_rule>
<nas_port_type>Ethernet</nas_port_type>
<selected_azn_profiles>CWA</selected_azn_profiles>
-
<other_attributes>
ConfigVersionId=44, DestinationIpAddress=10.203.107.162, DestinationPort=1812, Protocol=Radius, Framed-MTU=1500, EAP-Key-Name=, CPMSessionID=0ACB6BA1000000351BBFBF8B, CPMSessionID=0ACB6BA1000000351BBFBF8B, EndPointMACAddress=00-14-BF-5A-0C-03, HostIdentityGroup=Endpoint Identity Groups:Profiled, Device Type=Device Type#All Device Types, Location=Location#All Locations, Model Name=Unknown, Software Version=Unknown, Device IP Address=10.203.107.161, Called-Station-ID=04:FE:7F:7F:C0:8F
</other_attributes>
<response_time>77</response_time>
<acct_id>1291240762077386</acct_id>
<acct_acs_timestamp>2010-12-15T02:12:30.779Z</acct_acs_timestamp>
<acct_acsview_timestamp>2010-12-15T02:12:30.780Z</acct_acsview_timestamp>
<acct_session_id>00000038</acct_session_id>
<acct_status_type>Interim-Update</acct_status_type>
<acct_session_time>78</acct_session_time>
<acct_input_octets>13742</acct_input_octets>
<acct_output_octets>6277</acct_output_octets>
<acct_input_packets>108</acct_input_packets>
<acct_output_packets>66</acct_output_packets>
-
<acct_class>
CACS:0ACB6BA1000000351BBFBF8B:HAREESH-R6-1-PDP2/81148292/681
</acct_class>
<acct_delay_time>0</acct_delay_time>
<started xsi:type="xs:boolean">false</started>
<stopped xsi:type="xs:boolean">false</stopped>
</sessionParameters>

```

NAS IP アドレス セッションの検索

指定した NAS IP アドレス (IPv4 または IPv6 アドレス) のデータを現在のセッションから取得するために IP Address API コールを使用できます。この API は、ノード データベース テーブルから供給されるさまざまなセッション関連の情報をリストします。

IPAddress API の出力スキーマ

このサンプル スキーマ ファイルは、現在アクティブなセッションから指定した NAS IP アドレス (IPv4 または IPv6 アドレス) を取得するための IP Address API コールの出力です。

```

<?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="sessionParameters" type="restsdStatus"/>

  <xs:complexType name="restsdStatus">
    <xs:sequence>
      <xs:element name="passed" type="xs:anyType" minOccurs="0"/>
      <xs:element name="failed" type="xs:anyType" minOccurs="0"/>
      <xs:element name="user_name" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_ip_address" type="xs:string" minOccurs="0"/>
      <xs:element name="failure_reason" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

```

```

<xs:element name="calling_station_id" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port" type="xs:string" minOccurs="0"/>
<xs:element name="identity_group" type="xs:string" minOccurs="0"/>
<xs:element name="network_device_name" type="xs:string" minOccurs="0"/>
<xs:element name="acs_server" type="xs:string" minOccurs="0"/>
<xs:element name="authn_protocol" type="xs:string" minOccurs="0"/>
<xs:element name="framed_ip_address" type="xs:string" minOccurs="0"/>
<xs:element name="network_device_groups" type="xs:string" minOccurs="0"/>
<xs:element name="access_service" type="xs:string" minOccurs="0"/>
<xs:element name="auth_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="authentication_method" type="xs:string" minOccurs="0"/>
<xs:element name="execution_steps" type="xs:string" minOccurs="0"/>
<xs:element name="radius_response" type="xs:string" minOccurs="0"/>
<xs:element name="audit_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="nas_identifer" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port_id" type="xs:string" minOccurs="0"/>
<xs:element name="nac_policy_compliance" type="xs:string" minOccurs="0"/>
<xs:element name="auth_id" type="xs:long" minOccurs="0"/>
<xs:element name="auth_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="message_code" type="xs:string" minOccurs="0"/>
<xs:element name="acs_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="service_selection_policy" type="xs:string" minOccurs="0"/>
<xs:element name="authorization_policy" type="xs:string" minOccurs="0"/>
<xs:element name="identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="response" type="xs:string" minOccurs="0"/>
<xs:element name="service_type" type="xs:string" minOccurs="0"/>
<xs:element name="cts_security_group" type="xs:string" minOccurs="0"/>
<xs:element name="use_case" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_av_pair" type="xs:string" minOccurs="0"/>
<xs:element name="ad_domain" type="xs:string" minOccurs="0"/>
<xs:element name="acs_username" type="xs:string" minOccurs="0"/>
<xs:element name="radius_username" type="xs:string" minOccurs="0"/>
<xs:element name="nac_role" type="xs:string" minOccurs="0"/>
<xs:element name="nac_username" type="xs:string" minOccurs="0"/>
<xs:element name="nac_posture_token" type="xs:string" minOccurs="0"/>
<xs:element name="nac_radius_is_user_auth" type="xs:string" minOccurs="0"/>
<xs:element name="selected_posture_server" type="xs:string" minOccurs="0"/>
<xs:element name="selected_identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="authentication_identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="azn_exp_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="ext_pol_server_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="grp_mapping_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="identity_policy_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port_type" type="xs:string" minOccurs="0"/>
<xs:element name="query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="selected_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="sel_exp_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="selected_query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="eap_tunnel" type="xs:string" minOccurs="0"/>
<xs:element name="tunnel_details" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_ssg_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="other_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="response_time" type="xs:long" minOccurs="0"/>
<xs:element name="nad_failure" type="xs:anyType" minOccurs="0"/>
<xs:element name="destination_ip_address" type="xs:string" minOccurs="0"/>
<xs:element name="acct_id" type="xs:long" minOccurs="0"/>
<xs:element name="acct_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_status_type" type="xs:string" minOccurs="0"/>
<xs:element name="acct_session_time" type="xs:long" minOccurs="0"/>
<xs:element name="acct_input_octets" type="xs:string" minOccurs="0"/>
<xs:element name="acct_output_octets" type="xs:string" minOccurs="0"/>

```

```

<xs:element name="acct_input_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_output_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_class" type="xs:string" minOccurs="0"/>
<xs:element name="acct_terminate_cause" type="xs:string" minOccurs="0"/>
<xs:element name="acct_multi_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_authentic" type="xs:string" minOccurs="0"/>
<xs:element name="termination_action" type="xs:string" minOccurs="0"/>
<xs:element name="session_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="idle_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="acct_interim_interval" type="xs:string" minOccurs="0"/>
<xs:element name="acct_delay_time" type="xs:string" minOccurs="0"/>
<xs:element name="event_timestamp" type="xs:string" minOccurs="0"/>
<xs:element name="acct_tunnel_connection" type="xs:string" minOccurs="0"/>
<xs:element name="acct_tunnel_packet_lost" type="xs:string" minOccurs="0"/>
<xs:element name="security_group" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_setup_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_connect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_disconnect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="framed_protocol" type="xs:string" minOccurs="0"/>
<xs:element name="started" type="xs:anyType" minOccurs="0"/>
<xs:element name="stopped" type="xs:anyType" minOccurs="0"/>
<xs:element name="ckpt_id" type="xs:long" minOccurs="0"/>
<xs:element name="type" type="xs:long" minOccurs="0"/>
<xs:element name="nad_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="vlan" type="xs:string" minOccurs="0"/>
<xs:element name="dacl" type="xs:string" minOccurs="0"/>
<xs:element name="authentication_type" type="xs:string" minOccurs="0"/>
<xs:element name="interface_name" type="xs:string" minOccurs="0"/>
<xs:element name="reason" type="xs:string" minOccurs="0"/>
<xs:element name="endpoint_policy" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

<xs:element name="nas_ipv6_address" type="xs:string"/>
<xs:complexType name="framed_ipv6_address_list">
  <xs:sequence minOccurs="0" maxOccurs="8"><xs:element name="ipv6_address"
type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:element name="framed_ipv6_address" type="framed_ipv6_address_list" minOccurs="1"
maxOccurs="1"/>
</xs:schema>

```

NAS IPAddress API コールの呼び出し

- ステップ 1 Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2 ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3 [ログイン(Login)] をクリックするか、**Enter** を押します。
たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
`https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash`

- ステップ 4 「/admin/」コンポーネントを API コールのコンポーネント (/admin/API/mnt/<specific-api-call>/<nasipaddress>) に置き換えて、ターゲット ノードの URL アドレス フィールドに IPAddress API コールを入力します。

```
https://acme123/admin/API/mnt/Session/IpAddress/10.10.10.10
```



- (注) IPv4 アドレスまたは IPv6 アドレス (NAS IP アドレス) は、それぞれ xxx.xxx.xxx.xxx 形式または圧縮形式を使用して指定してください。



- (注) これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。

- ステップ 5 **Enter** キーを押して API コールを発行します。

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

IPAddress API コールから返されるサンプルデータ

次に、IPAddress API コールを呼び出すときにアクティブ セッションのリストから返されるセッション関連データの例を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<sessionParameters>
<passed xsi:type="xs:boolean">true</passed>
<failed xsi:type="xs:boolean">false</failed>
<user_name>ipepvpnuser</user_name>
<nas_ip_address>10.10.10.10</nas_ip_address>
<nas_ipv6_address>2001:cdba::357:965</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>200:cdba:0000:0000:0000:3157:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3247:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:962</ipv6_address>
</framed_ipv6_address>
<calling_station_id>172.23.130.90</calling_station_id>
<nas_port>1015</nas_port>
<identity_group>iPEP-VPN-Group</identity_group>
<network_device_name>iPEP-HA-Routed</network_device_name>
<acs_server>HAREESH-R6-1-PDP2</acs_server>
<authn_protocol>PAP_ASCII</authn_protocol>
-
<network_device_groups>
Device Type#All Device Types,Location#All Locations
</network_device_groups>
<access_service>RADIUS</access_service>
<auth_acs_timestamp>2010-12-15T19:57:29.885Z</auth_acs_timestamp>
<authentication_method>PAP_ASCII</authentication_method>
-
<execution_steps>
11001,11017,15008,15048,15048,15004,15041,15004,15013,24210,24212,22037,15036,15048,15048,
15004,15016,11002
</execution_steps>
```

```

<audit_session_id>0acb6be400000044D091DA9</audit_session_id>
<nac_policy_compliance>NotApplicable</nac_policy_compliance>
<auth_id>1291240762083580</auth_id>
<auth_acsview_timestamp>2010-12-15T19:57:29.887Z</auth_acsview_timestamp>
<message_code>5200</message_code>
<acs_session_id>HAREESH-R6-1-PDP2/81148292/693</acs_session_id>
<service_selection_policy>iPEP-VPN</service_selection_policy>
<identity_store>Internal Users</identity_store>
-
<response>
{User-Name=ipepvpnuser; State=ReauthSession:0acb6be400000044D091DA9;
Class=CACS:0acb6be400000044D091DA9:HAREESH-R6-1-PDP2/81148292/693;
Termination-Action=RADIUS-Request; }
</response>
<service_type>Framed</service_type>
-
<cisco_av_pair>
audit-session-id=0acb6be400000044D091DA9,ipep-proxy=true
</cisco_av_pair>
<acs_username>ipepvpnuser</acs_username>
<radius_username>ipepvpnuser</radius_username>
<selected_identity_store>Internal Users</selected_identity_store>
<authentication_identity_store>Internal Users</authentication_identity_store>
<identity_policy_matched_rule>Default</identity_policy_matched_rule>
<nas_port_type>Virtual</nas_port_type>
<selected_azn_profiles>iPEP-Unknown-Auth-Profile</selected_azn_profiles>
<tunnel_details>Tunnel-Client-Endpoint=(tag=0) 172.23.130.90</tunnel_details>
-
<other_attributes>
ConfigVersionId=44, DestinationIpAddress=10.203.107.162, DestinationPort=1812, Protocol=RADIUS,
Framed-Protocol=PPP, Proxy-State=Cisco Secure
ACS9e733142-070a-11e0-c000-000000000000-2906094480-3222, CPMSessionID=0acb6be400000044D091
DA9, CPMSessionID=0acb6be400000044D091DA9, Device Type=Device Type#All Device
Types, Location=Location#All Locations, Model Name=Unknown, Software Version=Unknown, Device
IP Address=10.203.107.228, Called-Station-ID=172.23.130.94
</other_attributes>
<response_time>20</response_time>
<acct_id>1291240762083582</acct_id>
<acct_acs_timestamp>2010-12-15T19:57:30.281Z</acct_acs_timestamp>
<acct_acsview_timestamp>2010-12-15T19:57:30.283Z</acct_acsview_timestamp>
<acct_session_id>F1800007</acct_session_id>
<acct_status_type>Start</acct_status_type>
-
<acct_class>
CACS:0acb6be400000044D091DA9:HAREESH-R6-1-PDP2/81148292/693
</acct_class>
<acct_delay_time>0</acct_delay_time>
<framed_protocol>PPP</framed_protocol>
<started xsi:type="xs:boolean">true</started>
<stopped xsi:type="xs:boolean">false</stopped>
</sessionParameters>

```

エンドポイントの IP アドレスのセッションの検索

現在のアクティブなセッションからセッションディレクトリ情報を取得するために EndPointIpAddress API コールを使用できます。ここでは、スキーマファイルの出力例、EndPointIpAddress API コールを呼び出すことより、指定された IP アドレスが含まれる最新のアクティブセッションに対応するノードデータベースを検索する手順、API コールの後に返されたエンドポイント関連データのサンプルについて説明します。この API コールは、ノードデータベース テーブルから供給されるさまざまなセッションディレクトリ情報をリストします。

EndPointIPAddress API の出力スキーマ

このサンプルスキーマファイルは、ターゲット Cisco Monitoring ISE ノードで現在アクティブなセッションから指定されたエンドポイントに関するセッションディレクトリ情報を取得するための EndPointIPAddress API コールの出力です。

```
<?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="sessionParameters" type="restsdStatus"/>
<xs:complexType name="restsdStatus">
<xs:sequence>
<xs:element name="passed" type="xs:anyType" minOccurs="0"/>
<xs:element name="failed" type="xs:anyType" minOccurs="0"/>
<xs:element name="user_name" type="xs:string" minOccurs="0"/>
<xs:element name="nas_ip_address" type="xs:string" minOccurs="0"/>
<xs:element name="failure_reason" type="xs:string" minOccurs="0"/>
<xs:element name="calling_station_id" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port" type="xs:string" minOccurs="0"/>
<xs:element name="identity_group" type="xs:string" minOccurs="0"/>
<xs:element name="network_device_name" type="xs:string" minOccurs="0"/>
<xs:element name="acs_server" type="xs:string" minOccurs="0"/>
<xs:element name="authn_protocol" type="xs:string" minOccurs="0"/>
<xs:element name="framed_ip_address" type="xs:string" minOccurs="0"/>
<xs:element name="network_device_groups" type="xs:string" minOccurs="0"/>
<xs:element name="access_service" type="xs:string" minOccurs="0"/>
<xs:element name="auth_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="authentication_method" type="xs:string" minOccurs="0"/>
<xs:element name="execution_steps" type="xs:string" minOccurs="0"/>
<xs:element name="radius_response" type="xs:string" minOccurs="0"/>
<xs:element name="audit_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="nas_identifier" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port_id" type="xs:string" minOccurs="0"/>
<xs:element name="nac_policy_compliance" type="xs:string" minOccurs="0"/>
<xs:element name="auth_id" type="xs:long" minOccurs="0"/>
<xs:element name="auth_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="message_code" type="xs:string" minOccurs="0"/>
<xs:element name="acs_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="service_selection_policy" type="xs:string" minOccurs="0"/>
<xs:element name="authorization_policy" type="xs:string" minOccurs="0"/>
<xs:element name="identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="response" type="xs:string" minOccurs="0"/>
<xs:element name="service_type" type="xs:string" minOccurs="0"/>
<xs:element name="cts_security_group" type="xs:string" minOccurs="0"/>
<xs:element name="use_case" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_av_pair" type="xs:string" minOccurs="0"/>
<xs:element name="ad_domain" type="xs:string" minOccurs="0"/>
<xs:element name="acs_username" type="xs:string" minOccurs="0"/>
<xs:element name="radius_username" type="xs:string" minOccurs="0"/>
<xs:element name="nac_role" type="xs:string" minOccurs="0"/>
<xs:element name="nac_username" type="xs:string" minOccurs="0"/>
<xs:element name="nac_posture_token" type="xs:string" minOccurs="0"/>
<xs:element name="nac_radius_is_user_auth" type="xs:string" minOccurs="0"/>
<xs:element name="selected_posture_server" type="xs:string" minOccurs="0"/>
<xs:element name="selected_identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="authentication_identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="azn_exp_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="ext_pol_server_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="grp_mapping_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="identity_policy_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port_type" type="xs:string" minOccurs="0"/>
<xs:element name="query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="selected_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="sel_exp_azn_profiles" type="xs:string" minOccurs="0"/>

```

```

<xs:element name="selected_query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="eap_tunnel" type="xs:string" minOccurs="0"/>
<xs:element name="tunnel_details" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_ssg_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="other_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="response_time" type="xs:long" minOccurs="0"/>
<xs:element name="nad_failure" type="xs:anyType" minOccurs="0"/>
<xs:element name="destination_ip_address" type="xs:string" minOccurs="0"/>
<xs:element name="acct_id" type="xs:long" minOccurs="0"/>
<xs:element name="acct_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_status_type" type="xs:string" minOccurs="0"/>
<xs:element name="acct_session_time" type="xs:long" minOccurs="0"/>
<xs:element name="acct_input_octets" type="xs:string" minOccurs="0"/>
<xs:element name="acct_output_octets" type="xs:string" minOccurs="0"/>
<xs:element name="acct_input_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_output_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_class" type="xs:string" minOccurs="0"/>
<xs:element name="acct_terminate_cause" type="xs:string" minOccurs="0"/>
<xs:element name="acct_multi_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_authentic" type="xs:string" minOccurs="0"/>
<xs:element name="termination_action" type="xs:string" minOccurs="0"/>
<xs:element name="session_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="idle_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="acct_interim_interval" type="xs:string" minOccurs="0"/>
<xs:element name="acct_delay_time" type="xs:string" minOccurs="0"/>
<xs:element name="event_timestamp" type="xs:string" minOccurs="0"/>
<xs:element name="acct_tunnel_connection" type="xs:string" minOccurs="0"/>
<xs:element name="acct_tunnel_packet_lost" type="xs:string" minOccurs="0"/>
<xs:element name="security_group" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_setup_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_connect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_disconnect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="framed_protocol" type="xs:string" minOccurs="0"/>
<xs:element name="started" type="xs:anyType" minOccurs="0"/>
<xs:element name="stopped" type="xs:anyType" minOccurs="0"/>
<xs:element name="ckpt_id" type="xs:long" minOccurs="0"/>
<xs:element name="type" type="xs:long" minOccurs="0"/>
<xs:element name="nad_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="vlan" type="xs:string" minOccurs="0"/>
<xs:element name="dacl" type="xs:string" minOccurs="0"/>
<xs:element name="authentication_type" type="xs:string" minOccurs="0"/>
<xs:element name="interface_name" type="xs:string" minOccurs="0"/>
<xs:element name="reason" type="xs:string" minOccurs="0"/>
<xs:element name="endpoint_policy" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
</xs:schema>

```

EndPointIPAddress API コールの呼び出し



(注)

API コールを発行するターゲット ノードが、有効な Cisco Monitoring ISE ノードであることを確認している必要があります。

EndPointIPAddress API コールを発行するには、次の手順を実行します。

- ステップ 1** ターゲット Cisco Monitoring ISE ノードにログインします。
たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
- ```
https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash
```
- ステップ 2** 「/admin/」コンポーネントを API コールのコンポーネント  
(`/ise/mnt/api/Session/EndPointIPAddress/<endpoint_ip>`) に置き換えて、ターゲット ノードの URL アドレス フィールドに EndPointIPAddress API コールを入力します。
- ```
https://acme123/ise/mnt/api/Session/EndPointIPAddress/A.B.C.D
```



(注) これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。

- ステップ 3** **Enter** キーを押して API コールを発行します。

EndPointIPAddress API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで EndPointIPAddress API コールを呼び出すときにアクティブセッションのリストから返されるセッション関連データを示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<sessionParameters>
<passed xsi:type="xs:boolean">true</passed>
<failed xsi:type="xs:boolean">false</failed>
<user_name>00:0C:29:95:A5:C1</user_name>
<nas_ip_address>10.77.152.139</nas_ip_address>
<calling_station_id>00:0C:29:95:A5:C1</calling_station_id>
<nas_port>50109</nas_port>
<identity_group>RegisteredDevices</identity_group>
<network_device_name>switch</network_device_name>
<acs_server>ise248</acs_server>
<authn_protocol>Lookup</authn_protocol>
<framed_ip_address>10.20.40.10</framed_ip_address>
-
<network_device_groups>
Device Type#All Device Types,Location#All Locations
</network_device_groups>
<access_service>RADIUS</access_service>
<auth_acs_timestamp>2012-03-13T17:02:22.169+05:30</auth_acs_timestamp>
<authentication_method>mab</authentication_method>
-
<execution_steps>
11001,11017,11027,15008,15048,15048,15004,15041,15006,15013,24209,24211,22037,15036,15048,
15004,15016,11022,11002
</execution_steps>
<audit_session_id>0A4D988B000000E337B8D983</audit_session_id>
<nas_port_id>GigabitEthernet1/0/9</nas_port_id>
<nac_policy_compliance>Pending</nac_policy_compliance>
<auth_id>1331101769985927</auth_id>
```



```

<auth_acsview_timestamp>2012-03-13T17:02:22.171+05:30</auth_acsview_timestamp>
<message_code>5200</message_code>
<acs_session_id>ise248/120476308/97</acs_session_id>
<service_selection_policy>MAB</service_selection_policy>
<authorization_policy>wired_redirect</authorization_policy>
<identity_store>Internal Endpoints</identity_store>
-
<response>
{UserName=00:0C:29:95:A5:C1; User-Name=00-0C-29-95-A5-C1;
State=ReauthSession:0A4D988B000000E337B8D983;
Class=CACS:0A4D988B000000E337B8D983:ise248/120476308/97;
Termination-Action=RADIUS-Request; Tunnel-Type=(tag=1) VLAN; Tunnel-Medium-Type=(tag=1)
802; Tunnel-Private-Group-ID=(tag=1) 30;
cisco-av-pair=url-redirect-acl=ACL-WEBAUTH-REDIRECT;
cisco-av-pair=url-redirect=https://ise248.cisco.com:8443/guestportal/gateway?sessionId=0A4
D988B000000E337B8D983&action=cwa;
cisco-av-pair=ACS:CiscoSecure-Defined-ACL=#ACSACL#-IP-cwa_wired-4f570619;
cisco-av-pair=profile-name=WindowsXP-Workstation; }
</response>
<service_type>Call Check</service_type>
<use_case>Host Lookup</use_case>
<cisco_av_pair>audit-session-id=0A4D988B000000E337B8D983</cisco_av_pair>
<acs_username>00:0C:29:95:A5:C1</acs_username>
<radius_username>00:0C:29:95:A5:C1</radius_username>
<selected_identity_store>Internal Endpoints</selected_identity_store>
<authentication_identity_store>Internal Endpoints</authentication_identity_store>
<identity_policy_matched_rule>Default</identity_policy_matched_rule>
<nas_port_type>Ethernet</nas_port_type>
<selected_azn_profiles>wired_cwa_redirect</selected_azn_profiles>
<response_time>17</response_time>
<destination_ip_address>10.77.152.248</destination_ip_address>
-
<other_attributes>
ConfigVersionId=15, DestinationPort=1812, Protocol=Radius, Framed-MTU=1500, EAP-Key-Name=, cisc
o-nas-port=GigabitEthernet1/0/9, CPMSessionID=0A4D988B000000E337B8D983, EndPointMACAddress=0
0-0C-29-95-A5-C1, EndPointMatchedProfile=WindowsXP-Workstation, HostIdentityGroup=Endpoint
Identity Groups:RegisteredDevices, Device Type=Device Type#All Device
Types, Location=Location#All Locations, Device IP
Address=10.77.152.139, Called-Station-ID=EC:C8:82:55:2E:09
</other_attributes>
<acct_id>1331101769985928</acct_id>
<acct_acs_timestamp>2012-03-13T17:02:22.365+05:30</acct_acs_timestamp>
<acct_acsview_timestamp>2012-03-13T17:02:22.366+05:30</acct_acsview_timestamp>
<acct_session_id>000000FC</acct_session_id>
<acct_status_type>Interim-Update</acct_status_type>
<acct_session_time>16411</acct_session_time>
<acct_input_octets>3053882</acct_input_octets>
<acct_output_octets>2633472</acct_output_octets>
<acct_input_packets>20166</acct_input_packets>
<acct_output_packets>20297</acct_output_packets>
<acct_class>CACS:0A4D988B000000E337B8D983:ise248/120476308/97</acct_class>
<acct_delay_time>0</acct_delay_time>
<started xsi:type="xs:boolean">false</started>
<stopped xsi:type="xs:boolean">false</stopped>
<vlan>30</vlan>
<dacl>#ACSACL#-IP-cwa_wired-4f570619</dacl>
<endpoint_policy>WindowsXP-Workstation</endpoint_policy>
</sessionParameters>

```

監査セッション ID の検索

現在のアクティブなセッションから指定した監査セッションを取得するために Audit Session ID API コールを使用できます。この API コールは、ノードデータベース テーブルから供給されるさまざまなセッション関連の情報をリストします。

Audit Session ID API の出力スキーマ

このサンプル スキーマ ファイルは、現在アクティブなセッションから指定した監査セッション ID を取得するための Audit Session ID API コールの出力です。

```
<?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="sessionParameters" type="restsdStatus"/>

  <xs:complexType name="restsdStatus">
    <xs:sequence>
      <xs:element name="passed" type="xs:anyType" minOccurs="0"/>
      <xs:element name="failed" type="xs:anyType" minOccurs="0"/>
      <xs:element name="user_name" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_ip_address" type="xs:string" minOccurs="0"/>
      <xs:element name="failure_reason" type="xs:string" minOccurs="0"/>
      <xs:element name="calling_station_id" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_port" type="xs:string" minOccurs="0"/>
      <xs:element name="identity_group" type="xs:string" minOccurs="0"/>
      <xs:element name="network_device_name" type="xs:string" minOccurs="0"/>
      <xs:element name="acs_server" type="xs:string" minOccurs="0"/>
      <xs:element name="authn_protocol" type="xs:string" minOccurs="0"/>
      <xs:element name="framed_ip_address" type="xs:string" minOccurs="0"/>
      <xs:element name="network_device_groups" type="xs:string" minOccurs="0"/>
      <xs:element name="access_service" type="xs:string" minOccurs="0"/>
      <xs:element name="auth_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
      <xs:element name="authentication_method" type="xs:string" minOccurs="0"/>
      <xs:element name="execution_steps" type="xs:string" minOccurs="0"/>
      <xs:element name="radius_response" type="xs:string" minOccurs="0"/>
      <xs:element name="audit_session_id" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_identifer" type="xs:string" minOccurs="0"/>
      <xs:element name="nas_port_id" type="xs:string" minOccurs="0"/>
      <xs:element name="nac_policy_compliance" type="xs:string" minOccurs="0"/>
      <xs:element name="auth_id" type="xs:long" minOccurs="0"/>
      <xs:element name="auth_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
      <xs:element name="message_code" type="xs:string" minOccurs="0"/>
      <xs:element name="acs_session_id" type="xs:string" minOccurs="0"/>
      <xs:element name="service_selection_policy" type="xs:string" minOccurs="0"/>
      <xs:element name="authorization_policy" type="xs:string" minOccurs="0"/>
      <xs:element name="identity_store" type="xs:string" minOccurs="0"/>
      <xs:element name="response" type="xs:string" minOccurs="0"/>
      <xs:element name="service_type" type="xs:string" minOccurs="0"/>
      <xs:element name="cts_security_group" type="xs:string" minOccurs="0"/>
      <xs:element name="use_case" type="xs:string" minOccurs="0"/>
      <xs:element name="cisco_av_pair" type="xs:string" minOccurs="0"/>
      <xs:element name="ad_domain" type="xs:string" minOccurs="0"/>
      <xs:element name="acs_username" type="xs:string" minOccurs="0"/>
      <xs:element name="radius_username" type="xs:string" minOccurs="0"/>
      <xs:element name="nac_role" type="xs:string" minOccurs="0"/>
      <xs:element name="nac_username" type="xs:string" minOccurs="0"/>
      <xs:element name="nac_posture_token" type="xs:string" minOccurs="0"/>
      <xs:element name="nac_radius_is_user_auth" type="xs:string" minOccurs="0"/>
      <xs:element name="selected_posture_server" type="xs:string" minOccurs="0"/>
      <xs:element name="selected_identity_store" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```

<xs:element name="authentication_identity_store" type="xs:string" minOccurs="0"/>
<xs:element name="azn_exp_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="ext_pol_server_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="grp_mapping_pol_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="identity_policy_matched_rule" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port_type" type="xs:string" minOccurs="0"/>
<xs:element name="query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="selected_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="sel_exp_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="selected_query_identity_stores" type="xs:string" minOccurs="0"/>
<xs:element name="eap_tunnel" type="xs:string" minOccurs="0"/>
<xs:element name="tunnel_details" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_ssg_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="other_attributes" type="xs:string" minOccurs="0"/>
<xs:element name="response_time" type="xs:long" minOccurs="0"/>
<xs:element name="nad_failure" type="xs:anyType" minOccurs="0"/>
<xs:element name="destination_ip_address" type="xs:string" minOccurs="0"/>
<xs:element name="acct_id" type="xs:long" minOccurs="0"/>
<xs:element name="acct_acs_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="acct_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_status_type" type="xs:string" minOccurs="0"/>
<xs:element name="acct_session_time" type="xs:long" minOccurs="0"/>
<xs:element name="acct_input_octets" type="xs:string" minOccurs="0"/>
<xs:element name="acct_output_octets" type="xs:string" minOccurs="0"/>
<xs:element name="acct_input_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_output_packets" type="xs:long" minOccurs="0"/>
<xs:element name="acct_class" type="xs:string" minOccurs="0"/>
<xs:element name="acct_terminate_cause" type="xs:string" minOccurs="0"/>
<xs:element name="acct_multi_session_id" type="xs:string" minOccurs="0"/>
<xs:element name="acct_authentic" type="xs:string" minOccurs="0"/>
<xs:element name="termination_action" type="xs:string" minOccurs="0"/>
<xs:element name="session_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="idle_timeout" type="xs:string" minOccurs="0"/>
<xs:element name="acct_interim_interval" type="xs:string" minOccurs="0"/>
<xs:element name="acct_delay_time" type="xs:string" minOccurs="0"/>
<xs:element name="event_timestamp" type="xs:string" minOccurs="0"/>
<xs:element name="acct_tunnel_connection" type="xs:string" minOccurs="0"/>
<xs:element name="acct_tunnel_packet_lost" type="xs:string" minOccurs="0"/>
<xs:element name="security_group" type="xs:string" minOccurs="0"/>
<xs:element name="cisco_h323_setup_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_connect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="cisco_h323_disconnect_time" type="xs:dateTime" minOccurs="0"/>
<xs:element name="framed_protocol" type="xs:string" minOccurs="0"/>
<xs:element name="started" type="xs:anyType" minOccurs="0"/>
<xs:element name="stopped" type="xs:anyType" minOccurs="0"/>
<xs:element name="ckpt_id" type="xs:long" minOccurs="0"/>
<xs:element name="type" type="xs:long" minOccurs="0"/>
<xs:element name="nad_acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
<xs:element name="vlan" type="xs:string" minOccurs="0"/>
<xs:element name="dacl" type="xs:string" minOccurs="0"/>
<xs:element name="authentication_type" type="xs:string" minOccurs="0"/>
<xs:element name="interface_name" type="xs:string" minOccurs="0"/>
<xs:element name="reason" type="xs:string" minOccurs="0"/>
<xs:element name="endpoint_policy" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

<xs:element name="nas_ipv6_address" type="xs:string"/>
<xs:complexType name="framed_ipv6_address_list">

```

```

    <xs:sequence minOccurs="0" maxOccurs="8"><xs:element name="ipv6_address"
type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="framed_ipv6_address" type="framed_ipv6_address_list" minOccurs="1"
maxOccurs="1"/>

</xs:schema>

```

Audit Session ID API コールの呼び出し

-
- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
`https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash`
- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント
(`/admin/API/mnt/Session/Active/SessionID/<audit-session-id>/0`)に置き換えて、ターゲット ノードの URL アドレス フィールドに Audit Session ID API コールを入力します。
`https://acme123/admin/API/mnt/Session/Active/SessionID/0A000A770000006B609A13A9/0`
-
- (注)** これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。
-
- ステップ 5** **Enter** キーを押して API コールを発行します。
-

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

Audit Session ID API コールから返されるサンプルデータ

次に、Audit Session ID API コールを呼び出すときにアクティブ セッションのリストから返されるセッション関連データの例を示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

-<activeSessionList noOfActiveSession="1">
  -<activeSession>
    <calling_station_id>00:50:56:10:13:02</calling_station_id>
    <session_state_bit>0</session_state_bit>
    <session_source>0</session_source>
    <acct_session_time>0</acct_session_time>
    <nas_ip_address>10.0.10.119</nas_ip_address>
    <nas_ipv6_address>2001:cdba::3257:9652</nas_ipv6_address>
    <framed_ipv6_address>
    <ipv6_address>200:cdba:0000:0000:0000:0000:3257:9652</ipv6_address>
  </activeSession>
</activeSessionList>

```

```

<ipv6_address> 2001:cdba:0:0:0:0:3257:9651</ipv6_address>
<ipv6_address>2001:cdba::3257:9652</ipv6_address>
</framed_ipv6_address>
<nas_port_id>GigabitEthernet1/0/15</nas_port_id>
<auth_method>dot1x</auth_method>
<auth_protocol>PEAP (EAP-MSCHAPv2)</auth_protocol>
<posture_status>Compliant</posture_status>
<endpoint_policy>Undetermined</endpoint_policy>
<server>acme123</server>
<paks_in>0</paks_in>
<paks_out>0</paks_out>
<bytes_in>0</bytes_in>
<bytes_out>0</bytes_out>
</activeSession>
</activeSessionList>

```

古いセッション

一部のデバイスでは、Wireless LAN Controller (WLC) など、古いセッションを保持できるようにする場合があります。このような場合、手動で非アクティブなセッションを削除するには、HTTP **DELETE** API コールを使用できます。これを行うには、URL (HTTP、HTTPS) 構文のデータを転送するための無償のサードパーティ製のコマンドライン ツールである **cURL** を使用します。

ISE は、これらのセッションを追跡しません。これは、ISE が長期間ネットワークに接続できなくなり、WLC/NAD から多数のアカウントングを停止できなくなった場合に問題を軽減するためです。この API を使用して ISE からこのような古い情報をクリアすることができます。



(注) HTTP および HTTPS を使用してファイルを取得するための無償ユーティリティである GNU Wget は、HTTP **DELETE** API コールをサポートしません。

古いセッションの削除

- ステップ 1 Cisco ISE URL をブラウザのアドレス バーに入力します (たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2 ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3 [ログイン(Login)] をクリックするか、**Enter** を押します。



(注) API コールは大文字と小文字が区別され、慎重に入力する必要があります。変数 `<mntnode>` は Cisco Monitoring ISE ノードを表します。

- ステップ 4 手動で MAC アドレスの古いセッションを削除するには、コマンドラインで次の API コールを実行します。

```
curl -X DELETE https://<mntnode>/admin/API/mnt/Session/Delete/MACAddress/<madaddress>
```

■ 古いセッション

ステップ 5 手動でセッション ID の古いセッションを削除するには、コマンドラインで次の API コールを発行します。

```
curl -X DELETE https://<mntnode>/admin/API/mnt/Session/Delete/SessionID/<sid#>
```

ステップ 6 手動でモニタリング ノードのすべてのセッションを削除するには、コマンドラインで次の API コールを発行します。

```
curl -X DELETE https://<mntnode>/admin/API/mnt/Session/Delete/All
```

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)



トラブルシューティング用のクエリー API

この章では、個々の Cisco Prime Network Control System (NCS) REST API コールの使用法について例をあげながら説明します。

Cisco Prime NCS API コール

Cisco Prime NCS API コールはノードのバージョンおよびタイプ、障害の理由、認証ステータスとアカウント ステータスを含むターゲット Cisco Monitoring ISE ノードのセッションに関する主要なトラブルシューティング情報を取得するためのメカニズムを提供します。

クエリー API を使用した Cisco ISE のトラブルシューティング

Cisco Prime NCS トラブルシューティング API コールは、Cisco ISE 展開のターゲット Cisco Monitoring ISE ノードにステータス要求を送信し、次の診断関連情報を取得します。

- ノードのバージョンおよびタイプ (Version API コールを使用)
- 障害理由 (FailureReasons API コールを使用)
- 認証ステータス (AuthStatus API コールを使用)
- アカウンティング ステータス (AcctStatus API コールを使用)

ノードのバージョンおよびタイプの API コール

各ノードの REST Programmatic インターフェイス (PI) サービスとクレデンシャルをテストするには Version API コールを使用できます。ここでは、スキーマ ファイルの出力例、この API コールを呼び出すことにより、Cisco ISE ソフトウェアのバージョンおよびノード タイプを要求する手順、この API コール発行後に返されるノードのバージョンとタイプのサンプルについて説明します。

ノード タイプは次のいずれかになります。

- STANDALONE_MNT_NODE = 0
- ACTIVE_MNT_NODE = 1
- BACKUP_MNT_NODE = 2
- NOT_AN_MNT_NODE = 3

Version API の出力スキーマ


このサンプル スキーマ ファイルは、ターゲット Cisco Monitoring ISE ノードへの送信後の、Version API コールの出力です。

```
<?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="product" type="product"/>

  <xs:complexType name="product">
    <xs:sequence>
      <xs:element name="version" type="xs:string" minOccurs="0"/>
      <xs:element name="type_of_node" type="xs:int"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string"/>
  </xs:complexType>
</xs:schema>
```

Version API コールの呼び出し

-
- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
- ログインに失敗した場合は、[ログイン(Login)] ページの [ログインで問題が発生する場合 (Problem logging in?)] リンクをクリックして、[ステップ 2](#) の説明に従ってください。
- たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
- ```
https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash
```
- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント (`/admin/API/mnt/<specific-api-call>`) に置き換えて、ターゲット ノードの URL アドレス フィールドに Version API コールを入力します。
- ```
https://acme123/admin/API/mnt/Version
```
-
-  **(注)** これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。
-
- ステップ 5** **Enter** キーを押して API コールを発行します。
-

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

Version API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで Version API コールを呼び出すときに返されるデータを示します。この API コールでは、ターゲット ノードについて次の 2 種類の値が返されます。

- ノードのバージョン(この例では、1.0.3.032 を表示します)。
- Cisco Monitoring ISE ノードのタイプ(この例では、アクティブな Cisco Monitoring ISE ノードが 1 つであることを意味する「1」を表示します)。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<product name="Cisco Identity Services Engine">
<version>1.0.3.032</version>
<type_of_node>1</type_of_node>
</product>
```

障害理由 API コール

ターゲット ノードで行われた認証ステータスのチェックで返された障害理由のリストを返すために FailureReasons API コールを使用できます。ここでは、スキーマファイルの出力例、この API コールを呼び出すことにより、Cisco Monitoring ISE ノードで記録される障害理由のリストを要求する手順、この API コール発行後に返される障害理由のサンプルについて説明します。返される障害理由は、それぞれ表 3-1 に示す次の要素で構成されます。



(注)

Cisco ISE Failure Reasons Editor を使用して障害理由の完全なリストにアクセスする方法に関する詳細については、[Cisco ISE 障害理由レポート \(A-1 ページ\)](#) を参照してください。

表 3-1 Cisco Identity Services Engine の製品マニュアル

障害理由の要素	例
障害理由 ID	<failureReason id="11011">
コード	<11011 RADIUS listener failed>
原因	<Could not open one or more of the ports used to receive RADIUS requests>
対処法	<Ensure that the ports 1812, 1813, 1645 and 1646 are not being used by another process on the system>



(注)

Cisco ISE ユーザーインターフェイスを使用して([モニタ (Monitor)]>[レポート (Reports)]>[カタログ (Catalog)]>[障害理由 (Failure Reasons)]) をクリックして障害理由レポートがあるかどうかを確認します。障害理由レポートが表示されます。

FailureReasons API の出力スキーマ

このサンプル スキーマ ファイルは、ターゲット Cisco Monitoring ISE ノードへの要求の送信後の、FailureReasons API コールの出力です。

```
<?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="failureReasonList" type="failureReasonList"/>

  <xs:complexType name="failureReasonList">
    <xs:sequence>
      <xs:element name="failureReason" type="failureReason" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="failureReason">
    <xs:sequence>
      <xs:element name="code" type="xs:string" minOccurs="0"/>
      <xs:element name="cause" type="xs:string" minOccurs="0"/>
      <xs:element name="resolution" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"/>
  </xs:complexType>
</xs:schema>
```

FailureReasons API コールの呼び出し

-
- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
- ログインに失敗した場合は、[ログイン(Login)] ページの [ログインで問題が発生する場合 (Problem logging in?)] リンクをクリックして、[ステップ 2](#) の説明に従ってください。
- たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。

```
https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash
```

- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント (`/admin/API/mnt/<specific-api-call>`) に置き換えて、ターゲット ノードの URL アドレス フィールドに FailureReasons API コールを入力します。

```
https://acme123/admin/API/mnt/FailureReasons
```



(注) これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。

- ステップ 5** **Enter** キーを押して API コールを発行します。
-

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

FailureReasons API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで FailureReasons API コールを呼び出すときに返されるデータを示します。この API コールは、ターゲット ノードから障害理由のリストを返します。障害理由は、それぞれ、障害 ID、障害コード、原因、対処法(既知の場合)によって定義されます。



(注) 次の FailureReasons API コールの例は、返されるデータの小規模なサンプルを表示しています。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<failureReasonList>
-
<failureReason id="100001">
-
<code>
100001 AUTHMGR-5-FAIL Authorization failed for client
</code>
<cause>This may or may not be indicating a violation</cause>
-
<resolution>
Please review and resolve according to your organization's policy
</resolution>
</failureReason>
-
<failureReason id="100002">
-
<code>
100002 AUTHMGR-5-SECURITY_VIOLATION Security violation on the interface
</code>
<cause>This may or may not be indicating a violation</cause>
-
<resolution>
Please review and resolve according to your organization's policy
</resolution>
</failureReason>
-
<failureReason id="100003">
-
<code>
100003 AUTHMGR-5-UNAUTHORIZED Interface unauthorized
</code>
<cause>This may or may not be indicating a violation</cause>
-
<resolution>
Please review and resolve according to your organization's policy
</resolution>
</failureReason>
-
<failureReason id="100004">
-
<code>
100004 DOT1X-5-FAIL Authentication failed for client
</code>
<cause>This may or may not be indicating a violation</cause>
```

```

-
<resolution>
Please review and resolve according to your organization's policy
</resolution>
</failureReason>
-
<failureReason id="100005">
<code>100005 MAB-5-FAIL Authentication failed for client</code>
<cause>This may or may not be indicating a violation</cause>
-
<resolution>
Please review and resolve according to your organization's policy
</resolution>
</failureReason>
-
<failureReason id="100006">
-
<code>
100006 RADIUS-4-RADIUS_DEAD RADIUS server is not responding
</code>
<cause>This may or may not be indicating a violation</cause>
-
<resolution>
Please review and resolve according to your organization's policy
</resolution>
</failureReason>
-
<failureReason id="100007">
-
<code>
100007 EPM-6-POLICY_APP_FAILURE Interface ACL not configured
</code>
<cause>This may or may not be indicating a violation</cause>
-
<resolution>
Please review and resolve according to your organization's policy
</resolution>
</failureReason>

```

関連項目

- [モニタリング ノードの確認 \(1-2 ページ\)](#)
- [付録 A「Cisco ISE 障害理由レポート」](#)

認証ステータス API コール

ターゲット ノードのセッションの認証ステータスをチェックするために `AuthStatus` API 呼び出しを使用できます。この API コールに関連付けられたクエリーには、一致の検索対象である MAC アドレスが少なくとも 1 つ必要です。指定の MAC アドレスが返されるように、最新レコードに、ユーザ設定が可能な制限を付けます。

ここでは、スキーマ ファイルの出力例、この API コールを呼び出すことにより、ターゲットのモニタリング モードでセッション認証のステータスを検索する要求を送信する手順、この API コール発行後に返されるデータのサンプルについて説明します。

AuthStatus API コールにより、次の検索関連パラメータを設定できるようになります。

- 期間: 指定された MAC アドレスに関連付けられた認証ステータス レコードの検索と取得が試行される秒数を定義します。ユーザが設定可能な値の有効範囲は 1 ~ 864000 秒(10 日)です。0 秒の値を入力した場合は、デフォルト期間の 10 日を指定します。
- レコード: MAC アドレスごとに検索するセッションのレコード数を定義します。ユーザが設定可能な値の有効範囲は 1 ~ 500 レコードです。0 を入力した場合は、デフォルト設定の 200 レコードを指定します。



(注) 期間およびレコード パラメータの両方に値 0 を指定すると、この API コールは、指定された MAC アドレスに関連付けられている最新の認証セッション レコードのみを返します。

ここに、期間とレコードの属性を指定した URL の一般的な形式の例を示します。

`https://10.10.10.10/admin/API/mnt/AuthStatus/MACAddress/01:23:45:67:89:98/900000/2/All`

- 属性: AuthStatus API コールを使用して認証ステータスの検索で返された認証ステータスのテーブルの属性数を定義します。有効な値は 0(デフォルト)、All、または `user_name+acs_timestamp` です (AuthStatus スキーマの例 ([AcctStatus API の出力スキーマ \(3-13 ページ\)](#)) を参照)。
 - 「0」を入力すると、表 3-2 で定義された属性が返されます。これらは出力スキーマの `restAuthStatus` のセクションに記載されています。
 - 「All」を入力すると、より詳しい属性セットが返されます。これらは出力スキーマの `fullRESTAuthStatus` のセクションに記載されています。
 - `user_name+acs_timestamp` のスキーマに示されている値を入力すると、それらの属性だけが返されます。`user_name` 属性と `acs_timestamp` 属性は、出力スキーマ `restAuthStatus` のセクションに記載されています。

表 3-2 認証ステータス テーブルの属性

属性 (Attribute)	説明
<code>name="passed"</code> または <code>name="failed"</code>	認証ステータスの結果: <ul style="list-style-type: none"> • パス (Passed) • 失敗しました (Failed)
<code>name="user_name"</code>	ユーザ名
<code>name="nas_ip_address"</code>	ネットワーク アクセス デバイスの IP アドレス/ホスト名
<code>name="nas_ipv6_address"</code>	ネットワーク アクセス デバイスの IPv6 アドレス/ホスト名
<code>name="failure_reason"</code>	セッション認証障害の理由
<code>name="calling_station_id"</code>	送信元 IP アドレス
<code>name="nas_port"</code>	ネットワーク アクセス サーバー ポート
<code>name="identity_group"</code>	関連ユーザおよび関連ホストで構成される論理グループ
<code>name="network_device_name"</code>	ネットワーク デバイスの名前
<code>name="acs_server"</code>	Cisco ISE アプライアンスの名前
<code>name="eap_authentication"</code>	認証要求用に使用される拡張認証プロトコル (EAP) メソッド
<code>name="framed_ip_address"</code>	特定ユーザ用に設定されたアドレス

表 3-2 認証ステータス テーブルの属性(続き)

属性 (Attribute)	説明
name="framed_ipv6_address"	特定ユーザ用に設定されたアドレス
network_device_groups"	関連ネットワーク デバイスで構成された論理グループ
name="access_service"	適用されたアクセス サービス
name="acs_timestamp"	Cisco ISE 認証要求に関連付けられたタイム スタンプ
name="authentication_method"	認証に使用されたメソッドを識別します
name="execution_steps"	要求処理の間に記録された、各診断メッセージのメッセージコードのリスト
name="radius_response"	RADIUS 応答のタイプ (VLAN、ACL など)
name="audit_session_id"	認証セッションの ID
name="nas_identifier"	特定リソースに関連付けられたネットワーク アクセス サーバ (NAS)
name="nas_port_id"	使用 NAS ポートの ID
name="nac_policy_compliance"	ポスチャ状態 (準拠または非準拠) を反映します
name="selected_azn_profiles"	認証に使用されたプロファイルを識別します
name="service_type"	フレームド ユーザを示します
name="eap_tunnel"	EAP 認証用に使用されるトンネルまたは外部メソッド
name="message_code"	要求結果処理を定義する監査メッセージの識別子
name="destination_ip_address"	宛先 IP アドレスを識別します

AuthStatus API の出力スキーマ

このサンプル スキーマ ファイルは、ターゲット Cisco Monitoring ISE ノードでの指定されたセッションへの送信後の、AuthStatus API コールの実出力です。

```
<?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="authStatusOutputList" type="fullRESTAuthStatusOutputList"/>

  <xs:complexType name="fullRESTAuthStatusOutputList">
    <xs:sequence>
      <xs:element name="authStatusList" type="fullRESTAuthStatusList" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="fullRESTAuthStatusList">
    <xs:sequence>
      <xs:element name="authStatusElements" type="fullRESTAuthStatus" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="key" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="fullRESTAuthStatus">
    <xs:complexContent>
      <xs:extension base="restAuthStatus">
```

```

<xs:sequence>
  <xs:element name="id" type="xs:long" minOccurs="0"/>
  <xs:element name="acsview_timestamp" type="xs:dateTime" minOccurs="0"/>
  <xs:element name="acs_session_id" type="xs:string" minOccurs="0"/>
  <xs:element name="service_selection_policy" type="xs:string" minOccurs="0"/>
  <xs:element name="authorization_policy" type="xs:string" minOccurs="0"/>
  <xs:element name="identity_store" type="xs:string" minOccurs="0"/>
  <xs:element name="response" type="xs:string" minOccurs="0"/>
  <xs:element name="cts_security_group" type="xs:string" minOccurs="0"/>
  <xs:element name="use_case" type="xs:string" minOccurs="0"/>
  <xs:element name="cisco_av_pair" type="xs:string" minOccurs="0"/>
  <xs:element name="ad_domain" type="xs:string" minOccurs="0"/>
  <xs:element name="acs_username" type="xs:string" minOccurs="0"/>
  <xs:element name="radius_username" type="xs:string" minOccurs="0"/>
  <xs:element name="nac_role" type="xs:string" minOccurs="0"/>
  <xs:element name="nac_username" type="xs:string" minOccurs="0"/>
  <xs:element name="nac_posture_token" type="xs:string" minOccurs="0"/>
  <xs:element name="nac_radius_is_user_auth" type="xs:string" minOccurs="0"/>
  <xs:element name="selected_posture_server" type="xs:string" minOccurs="0"/>
  <xs:element name="selected_identity_store" type="xs:string" minOccurs="0"/>
  <xs:element name="authentication_identity_store" type="xs:string"
minOccurs="0"/>
  <xs:element name="azn_exp_pol_matched_rule" type="xs:string" minOccurs="0"/>
  <xs:element name="ext_pol_server_matched_rule" type="xs:string" minOccurs="0"/>
  <xs:element name="grp_mapping_pol_matched_rule" type="xs:string" minOccurs="0"/>
  <xs:element name="identity_policy_matched_rule" type="xs:string" minOccurs="0"/>
  <xs:element name="nas_port_type" type="xs:string" minOccurs="0"/>
  <xs:element name="query_identity_stores" type="xs:string" minOccurs="0"/>
  <xs:element name="sel_exp_azn_profiles" type="xs:string" minOccurs="0"/>
  <xs:element name="selected_query_identity_stores" type="xs:string"
minOccurs="0"/>
  <xs:element name="tunnel_details" type="xs:string" minOccurs="0"/>
  <xs:element name="cisco_h323_attributes" type="xs:string" minOccurs="0"/>
  <xs:element name="cisco_ssg_attributes" type="xs:string" minOccurs="0"/>
  <xs:element name="other_attributes" type="xs:string" minOccurs="0"/>
  <xs:element name="response_time" type="xs:long" minOccurs="0"/>
  <xs:element name="nad_failure" type="xs:anyType" minOccurs="0"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

<xs:complexType name="restAuthStatus">
  <xs:sequence>
    <xs:element name="passed" type="xs:anyType" minOccurs="0"/>
    <xs:element name="failed" type="xs:anyType" minOccurs="0"/>
    <xs:element name="user_name" type="xs:string" minOccurs="0"/>
    <xs:element name="nas_ip_address" type="xs:string" minOccurs="0"/>
    <xs:element name="failure_reason" type="xs:string" minOccurs="0"/>
    <xs:element name="calling_station_id" type="xs:string" minOccurs="0"/>
    <xs:element name="nas_port" type="xs:string" minOccurs="0"/>
    <xs:element name="identity_group" type="xs:string" minOccurs="0"/>
    <xs:element name="network_device_name" type="xs:string" minOccurs="0"/>
    <xs:element name="acs_server" type="xs:string" minOccurs="0"/>
    <xs:element name="eap_authentication" type="xs:string" minOccurs="0"/>
    <xs:element name="framed_ip_address" type="xs:string" minOccurs="0"/>
    <xs:element name="network_device_groups" type="xs:string" minOccurs="0"/>
    <xs:element name="access_service" type="xs:string" minOccurs="0"/>
    <xs:element name="acs_timestamp" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="authentication_method" type="xs:string" minOccurs="0"/>
    <xs:element name="execution_steps" type="xs:string" minOccurs="0"/>
    <xs:element name="radius_response" type="xs:string" minOccurs="0"/>
    <xs:element name="audit_session_id" type="xs:string" minOccurs="0"/>
  </xs:sequence>

```


```

<xs:element name="nas_identifier" type="xs:string" minOccurs="0"/>
<xs:element name="nas_port_id" type="xs:string" minOccurs="0"/>
<xs:element name="nac_policy_compliance" type="xs:string" minOccurs="0"/>
<xs:element name="selected_azn_profiles" type="xs:string" minOccurs="0"/>
<xs:element name="service_type" type="xs:string" minOccurs="0"/>
<xs:element name="eap_tunnel" type="xs:string" minOccurs="0"/>
<xs:element name="message_code" type="xs:string" minOccurs="0"/>
<xs:element name="destination_ip_address" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>

<xs:element name="nas_ipv6_address" type="xs:string"/>
<xs:complexType name="framed_ipv6_address_list">
  <xs:sequence minOccurs="0" maxOccurs="8"><xs:element name="ipv6_address"
type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:element name="framed_ipv6_address" type="framed_ipv6_address_list" minOccurs="1"
maxOccurs="1"/>
</xs:schema>

```

AuthStatus API コールの呼び出し

-
- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
 ログインに失敗した場合は、[ログイン(Login)] ページの [ログインで問題が発生する場合 (Problem logging in?)] リンクをクリックして、[ステップ 2](#) の説明に従ってください。
 たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
`https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash`
- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント
 (`/admin/API/mnt/<specific-api-call>/MACAddress/<macaddress>/<seconds>/<numberofrecordsper macaddress>/All`) に置き換えて、ターゲット ノードの URL アドレス フィールドに AuthStatus API コールを入力します。
`https://acme123/admin/API/mnt/AuthStatus/MACAddress/00:50:56:10:13:02/120/100/All`
-  **(注)** REST API コールは大文字と小文字を区別します。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。
-
- ステップ 5** **Enter** キーを押して API コールを発行します。
-

関連項目

- ・ [モニタリング ノードの確認\(1-2 ページ\)](#)

AuthStatus API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで AuthStatus API コールを呼び出すときに返されるデータを示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<authStatusOutputList>
-
<authStatusList key="00:0C:29:46:F3:B8"><authStatusElements>
-
<passed xsi:type="xs:boolean">true</passed>
<failed xsi:type="xs:boolean">false</failed>
<user_name>suser77</user_name>
<nas_ip_address>10.77.152.209</nas_ip_address>
<nas_ipv6_address>2001:cdba::3257:9652</nas_ipv6_address>
<calling_station_id>00:0C:29:46:F3:B8</calling_station_id>
<identity_group>User Identity Groups:Guest</identity_group>
<acs_server>guest-240</acs_server>
<acs_timestamp>2012-10-05T10:50:56.515Z</acs_timestamp>
<execution_steps>5231</execution_steps>
<message_code>5231</message_code>
<id>1349422277270561</id>
<acsview_timestamp>2012-10-05T10:50:56.517Z</acsview_timestamp>
<identity_store>Internal Users</identity_store>
<response_time>146</response_time>
<other_attributes>ConfigVersionId=81,EndPointMACAddress=00-0C-29-46-F3-B8,PortalName=DefaultGuestPortal,
CPMSessionID=0A4D98D1000001F26F0C04D9,CiscoAVPair=</other_attributes>
</authStatusElements>
-
<authStatusElements>
<passed xsi:type="xs:boolean">true</passed>
<failed xsi:type="xs:boolean">false</failed>
<user_name>00:0C:29:46:F3:B8</user_name>
<nas_ip_address>10.77.152.209</nas_ip_address>
<nas_ipv6_address>2001:cdba::3257:9652</nas_ipv6_address>
<framed_ipv6_address>
<ipv6_address>2001:cdba:0000:0000:0000:0000:3257:9652</ipv6_address>
<ipv6_address> 2001:cdba:0:0:0:0:3257:9652</ipv6_address>
<ipv6_address>2001:cdba::3257:9652</ipv6_address>
</framed_ipv6_address>
<calling_station_id>00:0C:29:46:F3:B8</calling_station_id>
<identity_group>Guest_IDG</identity_group>
<network_device_name>switch</network_device_name>
<acs_server>guest-240</acs_server>
<authentication_method>mab</authentication_method>
<authentication_protocol>Lookup</authentication_protocol>
<acs_timestamp>2012-10-05T10:49:47.915Z</acs_timestamp>
<execution_steps>11001,11017,11027,15049,15008,15048,15048,15004,15041,15006,15013,24209,2
421
1,22037,15036,15048,15004,15016,11022,11002</execution_steps>
<response>{UserName
=00:0C:29:46:F3:B8; User-Name=00-0C-29-46-F3-B8;
State=ReauthSession:0A4D98D1000001F26F0C04D9;
Class=CACS:0A4D98D1000001F26F0C04D9:guest-240/138796808/76;
Termination-Action=RADIUS-Request; Tunnel-Type=(tag=1) VLAN;
Tunnel-Medium-Type=(tag=1) 802; Tunnel-Private-Group-ID=(tag=1) 2;
cisco-av-pair=url-redirect-acl=ACL-WEBAUTH-REDIRECT;
cisco-av-pair=url-redirect=https://guest-240.cisco.com:8443/guestportal/gateway?
sessionId=0A4D98D1000001F26F0C04D9&action=cwa;
```

```

cisco-av-pair=ACS:CiscoSecure-Defined-ACL=#ACSACL#-IP-pre-posture-506e980a;
cisco-av-pair=profile-name=WindowsXP-Workstation;}</response
><audit_session_id>0A4D98D1000001F26F0C04D9</audit_session_id><nas_po
rt_id>GigabitEthernet1/0/17</nas_port_id><posture_status>Pending</posture_status>
<selected_azn_profiles>CWA_Redirect</selected_azn_profiles>
<service_type>Call Check</service_type>
<message_code>5200</message_code>
<nac_policy_compliance>Pending</nac_policy_compliance>
<id>1349422277270556</id>
<acsview_timestamp>2012-10-05T10:49:47.915Z</acsview_timestamp>
<identity_store>Internal Endpoints</identity_store>
<response_time>13</response_time>
<other_attributes>ConfigVersionId=81, DestinationPort=1812, Protocol=Radius, AuthorizationPol
icyMatchedRule=CWA_Redirect,
NAS-Port=50117, Framed-MTU=1500, NAS-Port-Type=Ethernet, EAP-Key-N
ame=, cisco-nas-port=GigabitEthernet1/0/17, AcsSessionID=guest-240/138796808/76, Us
eCase=Host Lookup, SelectedAuthenticationIdentityStores=Internal
Endpoints, ServiceSelectionMatchedRule=MAB, IdentityPolicyMatchedRule=Default, CPMS
essionID=0A4D98D1000001F26F0C04D9, EndPointMACAddress=00-0C-29-46-F3-B8, EndPointM
atchedProfile=WindowsXP-Workstation, ISEPolicySetName=Default, HostIdentityGroup=E
ndpoint Identity Groups:Guest_IDG, Device Type=Device Type#All Device
Types, Location=Location#All Locations, Device IP
Address=10.77.152.209, Called-Station-ID=00:24:F7:73:9A:91, CiscoAVPair=audit-sess
ion-id=0A4D98D1000001F26F0C04D9</other_attributes>
-
</authStatusElements>
-
</authStatusList>
-
</authStatusOutputList>

```

アカウントステータス API コール

ターゲット ノードの最新のデバイスおよびセッションのアカウント情報を取得するために AcctStatus API コールを使用できます。ここでは、スキーマ ファイルの出力例、この API コールを呼び出すことにより、最新のデバイスおよびセッション情報の要求を送信する手順、この API コール発行後に返されるデータのサンプルについて説明します。AcctStatus API コールにより、時間関連パラメータを設定できるようになります。

- 期間: 指定された MAC アドレスに関連付けられた最新アカウントのデバイス レコードの検索と取得が試行される秒数を定義します。ユーザが設定可能な値の有効範囲は 1 ~ 432000 秒 (5 日) です。次に例を示します。
 - 2400 秒 (40 分) の値を入力した場合は、過去 40 分間に使用可能な指定 MAC アドレスの最新アカウントのデバイス レコードが必要であることを意味します。
 - 0 秒の値を入力した場合は、デフォルト期間の 15 分 (900 秒) を指定します。これは、この時間内に使用可能な指定 MAC アドレスの最新アカウントのデバイス レコードが必要であることを意味します。

AcctList API コールは、API 出力として、次のアカウントステータスのデータ フィールドを提供します (表 3-3 を参照)。

表 3-3 アカウンティングステータスのデータ フィールド

データ フィールド	説明
MAC アドレス	クライアントの MAC アドレス
監査セッション ID	監査セッション ID

表 3-3 アカウンティングステータスのデータ フィールド(続き)

データ フィールド	説明
パケット入力	総受信パケット カウント
パケット出力	総送信パケット カウント
バイト入力	総受信バイト カウント
バイト出力	総送信バイト カウント
セッション時間	現在のセッションの期間

AcctStatus API の出力スキーマ

このサンプル スキーマ ファイルは、ターゲット Cisco Monitoring ISE ノードでの指定されたセッションへの送信後の、AcctStatus API コールの実出力です。

```
<?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="acctStatusOutputList" type="restAcctStatusOutputList"/>

  <xs:complexType name="restAcctStatusOutputList">
    <xs:sequence>
      <xs:element name="acctStatusList" type="restAcctStatusList" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="restAcctStatusList">
    <xs:sequence>
      <xs:element name="acctStatusElements" type="restAcctStatus" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="macAddress" type="xs:string"/>
    <xs:attribute name="username" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="restAcctStatus">
    <xs:sequence>
      <xs:element name="calling_station_id" type="xs:string" minOccurs="0"/>
      <xs:element name="audit_session_id" type="xs:string" minOccurs="0"/>
      <xs:element name="paks_in" type="xs:long" minOccurs="0"/>
      <xs:element name="paks_out" type="xs:long" minOccurs="0"/>
      <xs:element name="bytes_in" type="xs:long" minOccurs="0"/>
      <xs:element name="bytes_out" type="xs:long" minOccurs="0"/>
      <xs:element name="session_time" type="xs:long" minOccurs="0"/>
      <xs:element name="username" type="xs:string" minOccurs="0"/>
      <xs:element name="server" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

AcctStatus API コールの呼び出し

- ステップ 1 Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2 ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。

ステップ 3 [ログイン(Login)] をクリックするか、**Enter** を押します。

ログインに失敗した場合は、[ログイン(Login)] ページの [ログインで問題が発生する場合 (Problem logging in?)] リンクをクリックして、**ステップ 2** の説明に従ってください。

たとえば、ホスト名が acme123 の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。

```
https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash
```

ステップ 4 「/admin/」コンポーネントを API コールのコンポーネント (/admin/API/mnt/<specific-api-call>/MACAddress/<macaddress>/<durationofcurrenttime>) に置き換えて、ターゲット ノードの URL アドレス フィールドに AcctStatus API コールを入力します。

```
https://acme123/admin/API/mnt/AcctStatus/MACAddress/00:26:82:7B:D2:51/1200
```



(注) これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。

ステップ 5 **Enter** キーを押して API コールを発行します。

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

AcctStatus API コールから返されるサンプル データ

次に、ターゲット Cisco Monitoring ISE ノードで AcctStatus API コールを呼び出すときに返されるデータを示します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<acctStatusOutputList>
-
<acctStatusList macAddress="00:25:9C:A3:7D:48">
-
<acctStatusElements>
<calling_station_id>00:25:9C:A3:7D:48</calling_station_id>
<audit_session_id>0acb6b0b000000B4D0C0DBD</audit_session_id>
<paks_in>0</paks_in>
<paks_out>0</paks_out>
<bytes_in>0</bytes_in>
<bytes_out>0</bytes_out>
<session_time>240243</session_time>
<server>HAREESH-R6-1-PDP1</server>
</acctStatusElements>
</acctStatusList>
</acctStatusOutputList>
```



認可変更 REST API

この章では、Cisco Identity Services Engine のこのリリースでサポートされている次の個々の認可変更 (CoA) REST API コールの使用法について例をあげながら説明します。

はじめに

CoA API コールは、Cisco ISE 導入で指定された Cisco Monitoring ISE ノードセッションに認証コマンドおよびセッション切断コマンドを送信する方法を提供します。

CoA セッション管理 API コール

CoA セッション管理 API コールにより、Cisco ISE 導入において、ターゲット Cisco Monitoring ISE ノードの指定セッションに再認証コマンドおよび切断コマンドを送信できるようにします。

- セッション再認証 (Reauth)
- セッション切断 (Disconnect)

セッション再認証 API コール

セッション再認証 API コールは次のタイプを構成します。

- REAUTH_TYPE_DEFAULT = 0
- REAUTH_TYPE_LAST = 1
- REAUTH_TYPE_RERUN = 2

Reauth API の出力スキーマ

このサンプルスキーマ ファイルは、ターゲット Cisco Monitoring ISE ノードで指定セッションへの送信後の Reauth API コールの出力です。

```
<?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="remoteCoA" type="coAResult"/>
<xs:complexType name="coAResult">
  <xs:sequence>
    <xs:element name="results" type="xs:boolean" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>
```

```
<xs:attribute name="requestType" type="xs:string"/>
</xs:complexType>
</xs:schema>
```

Reauth API コールの呼び出し

- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
`https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash`
- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント (`/admin/API/mnt/CoA/<specific-api-call>/<macaddress>/<reauthtype>`) に置き換えて、ターゲットノードの URL アドレス フィールドに Reauth API コールを入力します。
`https://acme123/admin/API/mnt/CoA/Reauth/server12/00:26:82:7B:D2:51/1`



(注) これらのコールは、大文字小文字を区別するため、ターゲットノードの URL アドレスフィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。

- ステップ 5** **Enter** キーを押して API コールを発行します。

関連項目

- ・ [モニタリング ノードの確認\(1-2 ページ\)](#)

Reauth API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで Reauth API コールを呼び出すときに返されるデータを示します。このコマンドの呼び出しから、次の 2 種類の結果が返されます。

- ・ 「True」はコマンドが正常に実行されたことを示します。
- ・ 「False」は(さまざまな条件により)コマンドが実行されなかったことを意味します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-
<remoteCoA requestType="reauth">
<results>true</results>
</remoteCoA>
```

セッション切断 API コール

セッション切断 API コールは、次の接続解除のポート オプション タイプを構成します。

- DYNAMIC_AUTHZ_PORT_DEFAULT = 0
- DYNAMIC_AUTHZ_PORT_BOUNCE = 1
- DYNAMIC_AUTHZ_PORT_SHUTDOWN = 2

Disconnect API の出力スキーマ


このサンプル スキーマ ファイルは、ターゲット Cisco Monitoring ISE ノードで指定セッションへの送信後の Disconnect API コールの出力です。

```
<?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="remoteCoA" type="coAResult"/>

  <xs:complexType name="coAResult">
    <xs:sequence>
      <xs:element name="results" type="xs:boolean" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="requestType" type="xs:string"/>
  </xs:complexType>
</xs:schema>
```

Disconnect API コールの呼び出し

-
- ステップ 1** Cisco ISE URL をブラウザのアドレス バーに入力します(たとえば `https://<ise hostname or ip address>/admin/`)。
- ステップ 2** ユーザ名と、Cisco ISE の初期セットアップで指定および設定した大文字と小文字が区別されるパスワードを入力します。
- ステップ 3** [ログイン(Login)] をクリックするか、**Enter** を押します。
- たとえば、ホスト名が `acme123` の Cisco Monitoring ISE ノードに最初にログインする場合、このノードの URL アドレスが次のように表示されます。
- ```
https://acme123/admin/LoginAction.do#pageId=com_cisco_xmp_web_page_tmpdash
```
- ステップ 4** 「/admin/」コンポーネントを API コールのコンポーネント (/admin/API/mnt/CoA/<Disconnect>/<serverhostname>/<macaddress>/<portoptiontype>/<nasipaddress>/<destinationipaddress>) に置き換えて、ターゲット ノードの URL アドレス フィールドに Disconnect API コールを入力します。
- ```
https://acme123/admin/API/mnt/CoA/Disconnect/server12/00:26:82:7B:D2:51/2/10.10.10.10/192.168.1.1
```
-  **(注)** これらのコールは、大文字小文字を区別するため、ターゲット ノードの URL アドレス フィールドに慎重に各 API コールを入力する必要があります。API コール規則での「mnt」の使用は、Cisco Monitoring ISE ノードを表します。
-
- ステップ 5** **Enter** キーを押して API コールを発行します。
-

関連項目

- [モニタリング ノードの確認\(1-2 ページ\)](#)

Disconnect API コールから返されるサンプルデータ

次に、ターゲット Cisco Monitoring ISE ノードで Disconnect API コールを呼び出すときに返されるデータを示します。このコマンドの呼び出しから、次の2種類の結果が返されます。

- 「True」はコマンドが正常に実行されたことを示します。
- 「False」は(さまざまな条件により)コマンドが実行されなかったことを意味します。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
-  
<remoteCoA requestType="reauth">  
<results>true</results>  
</remoteCoA>
```




PART 2

Cisco ISE 外部 RESTful サービス API



ERS API の概要

外部 RESTful サービス API コールを使用するための前提条件

外部 RESTful サービス API コールを呼び出す前に、次の前提条件を満たす必要があります。

- GUI から外部 RESTful サービスをイネーブ爾にしておく必要があります。
- 外部 RESTful サービスの管理者権限が必要です。

JAVA、curl Linux コマンド、Python などの REST クライアントやその他のクライアントを使用して、外部 RESTful サービス API コールを呼び出すことができます。

外部 RESTful サービス SDK

外部 RESTful サービス SDK を使用して、独自ツールの構築を開始できます。次の URL から外部 RESTful サービス SDK にアクセスできます。<https://<ISE-ADMIN-NODE>:9060/ers/sdk>

外部 RESTful サービス SDK には、外部 RESTful サービス管理ユーザのみがアクセスできます。SDK は、次のコンポーネントで構成されています。

- クイック リファレンス API マニュアル
- すべての利用可能な API 操作の完全なリスト
- ダウンロード可能なスキーマ ファイル
- ダウンロード可能な Java のサンプル アプリケーション
- cURL スクリプト形式の使用例
- python スクリプト形式の使用例
- Chrome POSTMAN の使用方法

次の API が SDK で使用できます。

- 証明書テンプレート API
- 脅威と脆弱性のクリア API
- 出力マトリックスセル API
- エンドポイント API
- エンドポイントの証明書 API
- エンドポイント アイデンティティ グループ API

- ゲスト ロケーション API
- ゲスト SMTP 通知設定 API
- ゲスト SSID API
- ゲスト タイプ API
- ゲスト ユーザ API
- ホットスポット ポータル API
- IP-to-SGT マッピング API
- IP-to-SGT マッピング グループ API
- ISE サービス情報 API
- アイデンティティ グループ API
- アイデンティティ シーケンス API
- 内部ユーザ API
- マイ デバイス ポータル API
- ネイティブ サプリカント プロファイル API
- ネットワーク デバイス API
- ネットワーク デバイス グループ API
- ノードの詳細 API
- RADIUS サービスのある PSN ノードの詳細
- ポータル API
- ポータルのテーマ API
- プロファイラ プロファイル API
- SMS サーバ API
- SXP 接続 API
- SXP ローカル バインディング API
- SXP VPN API
- セキュリティ グループ API
- セキュリティ グループ ACL (SGACL) API
- セルフ登録ポータル API
- スポンサー グループ API
- スポンサー グループ メンバー API
- スポンサー ポータル API
- スポンサー ゲスト ポータル API

外部 RESTful サービス API の認証および承認

外部 RESTful サービス API は HTTPS プロトコルおよび REST 方法論に基づいており、ポート 9060 を使用します。

外部 RESTful サービス API は、基本認証をサポートしています。認証クレデンシャルは、暗号化され、要求ヘッダーの一部となっています。

ISE 管理者は、外部 RESTful サービス API を使用して操作を実行するための特権をユーザに割り当てる必要があります。

外部 RESTful サービス API (ゲスト API を除く) を使用して操作を実行するには、次の管理者グループのいずれかにユーザを割り当て、Cisco ISE の内部データベース (内部管理者ユーザ) に保存されているクレデンシャルに対して認証する必要があります。

- 外部 RESTful サービス管理者: すべての ERS API へのフルアクセス (GET、POST、DELETE、PUT)。このユーザは、ERS API 要求を作成、読み取り、更新、および削除できます。
- 外部 RESTful サービス オペレータ: 読み取り専用アクセス (GET 要求のみ)。

必要な権限がない場合に外部 RESTful サービス API を使用して操作を実行しようとすると、エラー応答を受信します。



Cisco ISE 障害理由レポート

この付録では、Cisco ISE 障害理由レポートにアクセスするための手順を提供します。Cisco ISE 障害理由レポートには、障害理由のリストが示されます。

はじめに

Cisco ISE 障害理由レポートは、検出できる障害理由すべてに関する情報を提供する Cisco ISE ユーザーインターフェイスのオプションです。API を解決する Cisco ISE クエリーを使用すると Get Failure Reason Mapping コールから出力として返されるオプションをチェックする場合があります。

Cisco ISE 障害理由レポートを使用すると、Cisco ISE ソフトウェアによって定義された Cisco Monitoring ISE ノード動作に適用する障害理由の全リストにアクセスできるようになります。次の手順により、定義された障害理由のリストを表示または編集することができます。障害理由を表示し、ここにアクセスするには、宛先 Cisco Monitoring ISE ノードの Cisco ISE ユーザーインターフェイスにログインする必要があります。ログインに関する詳細については、[モニタリング ノードの確認\(1-2 ページ\)](#)を参照してください。

障害理由の表示

- ステップ 1 [操作(Operations)] > [レポート(Reports)] > [認証の要約(Authentication Summary)] レポートを選択します。
- ステップ 2 ナビゲーション パネルの [モニタリング(Monitoring)] を展開し、[障害理由エディタ(Failure Reason Editor)] を選択します。
- ステップ 3 提供されたフィルタのリストから [障害理由(Failure Reasons)] を選択します。
- ステップ 4 探している障害理由を指定します。
- ステップ 5 [実行(Run)] をクリックします。
障害理由のリストが右側のパネルに表示されます。
- ステップ 6 任意の障害理由をクリックして、新しいウィンドウで詳細レポートを取得します。

