CHAPTER 3

Using the Query APIs for Troubleshooting

This chapter provides examples and describes how to use the individual Cisco Prime Network Control System (NCS) REST API calls that are supported in this release. The Cisco Prime NCS API calls provide a mechanism for retrieving key troubleshooting information about the target Cisco Monitoring ISE node sessions that include node version and type, failure reasons, authentication status, and accounting status.

The following sections provide you with troubleshooting information obtained by using Query API calls, and this information is in the form of output schema file examples, procedures for issuing each API call, and a sample of the data returned by each API call:

Troubleshooting Cisco ISE using the Query API Calls

The following sections provide key Cisco Prime NCS troubleshooting API calls that send status requests to the target Cisco Monitoring ISE node that you designated in your Cisco ISE deployment and retrieve the following diagnostic-related information:

- Node version and type (using the Version API call)
- Failure reasons (using the FailureReasons API call)
- Authentication status (using the AuthStatus API call)
- Accounting status (using the AcctStatus API call)

Node Version and Type API Call

You can use the Version API call to test the REST programmatic interface (PI) service and the credentials of each node. This section provides a schema file output example, a procedure for requesting the version of the Cisco ISE software and the node type by invoking this API call, and a sample of the node version and type that is returned after this API call is issued.

The node types can be any of the following:

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

Version API Output Schema

This sample schema file is the output of the Version API call after sending it to the target Cisco Monitoring ISE node:

```
<?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>
```

Invoking the Version API Call

Note Make sure that you have verified that the target node to which you are issuing an API call is a valid Cisco Monitoring ISE node. To verify the persona of a Cisco ISE node, see Verifying a Cisco Monitoring ISE Node, page 1-2.

To issue the Version API call, complete the following steps:

Step 1 Log into the target Cisco Monitoring ISE node.

For example, when you initially log into a Cisco Monitoring ISE node with the hostname of acme123, this would display the following URL Address field for this node:

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

Step 2 Enter the Version API call in the URL Address field of the target node by replacing the “/admin/” component with the API call component (/ise/mnt/api/<specific-api-call>):

```
https://acme123/ise/mnt/api/Version
```

Note You must carefully enter each API call in the URL Address field of a target node because these calls are case-sensitive. The use of "mnt" in the API call convention represents a Cisco Monitoring ISE node.

Step 3 Press **Enter** to issue the API call.

Sample Data Returned from the Version API Call

The following example illustrates the data returned when you invoke a Version API call on a target Cisco Monitoring ISE node. This API call returns the following two values for the target node:

- Node version (this example displays 1.0.3.032).
- Type of Cisco Monitoring ISE node (this example displays a "1", which means an active Cisco Monitoring ISE node).

```
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>
```

Failure Reasons API Call

You can use the FailureReasons API call to return a list of failure reasons returned in the authentication status check done on the target node. This section provides a schema file output example, a procedure for requesting a list of all failure reasons logged by the Cisco Monitoring ISE node by invoking this API call, and a sample of the failure reasons returned after this API call is issued. Each failure reason that is returned consists of the following elements shown in Table 3-1.

Note For details about using the Cisco ISE Failure Reasons Editor to access the complete list of failure reasons, see Using the Cisco ISE Failure Reasons Editor, page A-1.

Table 3-1 Product Documentation for Cisco Identity Services Engine

Failure Reason Elements	Example
Failure reason ID	`<failureReason id="11011">`
Code	`<11011 RADIUS listener failed>`
Cause	`<Could not open one or more of the ports used to receive RADIUS requests>`
Resolution	`<Ensure that the ports 1812, 1813, 1645 and 1646 are not being used by another process on the system>`

Note You can also check for failure reason reports using the Cisco ISE user interface (click **Monitor > Reports > Catalog > Failure Reasons**), which will display failure reason reports.

FailureReasons API Output Schema

This sample schema file is the output of the FailureReasons API call after sending the request to a target Cisco Monitoring ISE node:

```
<?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>
```

Invoking the FailureReasons API Call

Note Make sure that you have verified that the target node to which you are issuing an API call is a valid Cisco Monitoring ISE node. To verify the persona of a Cisco ISE node, see Verifying a Cisco Monitoring ISE Node, page 1-2.

To issue the FailureReasons API call, complete the following steps:

Step 1 Log into the target Cisco Monitoring ISE node.

For example, when you initially log into a Cisco Monitoring ISE node with the hostname of acme123, this would display the following URL Address field for this node:

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

Step 2 Enter the FailureReasons API call in the URL Address field of the target node by replacing the "/admin/" component with the API call component (/ise/mnt/api/<specific-api-call>):

```
https://acme123/ise/mnt/api/FailureReasons
```

Note You must carefully enter each API call in the URL Address field of a target node because these calls are case-sensitive. The use of "mnt" in the API call convention represents a Cisco Monitoring ISE node.

Step 3 Press **Enter** to issue the API call.

Sample Data Returned from the FailureReasons API Call

The following example illustrates the data returned when you invoke a FailureReasons API call on a target Cisco Monitoring ISE node. This API call returns a list of failure reasons from the target node, and each failure reason is defined by a failure ID, a failure code, a cause, and a resolution (if known).

Note The following FailureReasons API call example only displays a small sample of data that can be returned.

```
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>
```


For more information

For more information about the Cisco ISE Failure Reasons Editor, see Appendix A, “Using the Cisco ISE Failure Reasons Editor”.

Authentication Status API Call

You can use the AuthStatus API call to check the authentication status of sessions on the target node. The query associated with this API call requires at least one MAC address to be searched for a match, with a user-configurable limit of the most recent records for the specified MAC address returned.

This section provides a schema file output example, a procedure for sending a a request to search for session authentication status on a target Monitoring mode by invoking this API call, and a sample of the data returned after this API call is issued.

The AuthStatus API call lets you configure the following search-related parameters:

- Duration—Defines the number of seconds in which an attempt is made to search and retrieve the authentication status records associated with the designated MAC address. Valid user-configurable values range from 1 to 864000 seconds (10 days). If you enter a value of 0 seconds, this specifies a default duration of 10 days.
- Records—Defines the number of session records to be searched per MAC address. Valid user-configurable values range from 1 to 500 records. If you enter 0, this specifies a default setting of 200 records.

Note If you specify the value 0 for both the duration and the records parameters, this API call returns only the very latest authentication session record associated with the designated MAC address(es).

- Attributes—Defines the number of attributes in the authentication status table that are returned from an authentication status search using the AuthStatus API call. Valid values include 0 (the default), All, or user_name+acs_timestamp (see the AuthStatus schema example, AcctStatus API Output Schema, page 3-15).
  - If you enter “0”, the attributes defined in Table 3-2 are returned. These are listed in the restAuthStatus section of the output schema.
  - If you enter “All”, a fuller set of attributes are returned. These are listed in the fullRESTAuthStatus section of the output schema.
  - If you enter the values listed in the schema for user_name+acs_timestamp, only those attributes are returned. The user_name and acs_timestamp attributes are listed in the restAuthStatus section of the output schema.

Table 3-2 Authentication Status Table Attributes

Attribute	Description
name=”passed”	One of two possible authentication status results: • Passed
name=”failed”	One of two possible authentication status results: • Failed
name=”user_name”	User name
name=”nas_ip_address”	IP address/hostname for the network access switch
name=”failure_reason”	Reason for session authentication failure
name=”calling_station_id”	Source IP address
name=”nas_port”	Network access server port

Table 3-2 Authentication Status Table Attributes (continued)

Attribute	Description
name=”identity_group”	A logical group consisting of related users and hosts
name=”network_device_name”	Name of the network device
name=”acs_server”	Name of the Cisco ISE appliance
name=”eap_authentication”	Extensible Authentication Protocol (EAP) method used for authentication request
name=”framed_ip_address”	Address configured for a specific user
network_device_groups”	A logical group consisting of related network devices
name=”access_service”	Applied access service
name=”acs_timestamp”	Time stamp that is associated with the Cisco ISE authentication request
name=”authentication_method”	Identifies the method used in authentication
name=”execution_steps”	List of message codes for each diagnostic message logged while processing the request
name=”radius_response”	Type of RADIUS response (for example, VLAN or ACL)
name=”audit_session_id”	ID of the authentication session
name=”nas_identifier”	A network access server (NAS) associated with a specific resource
name=”nas_port_id”	ID of the NAS port used
name=”nac_policy_compliance”	Reflects Posture status (compliant or non-compliant)
name=”selected_azn_profiles”	Identifies the profile used in authorization
name=”service_type”	Indicates a framed user
name=”eap_tunnel”	Tunnel or outer method used for EAP authentication
name=”message_code”	Identifier of the audit message that defines the processed request result
name=”destination_ip_address”	Identifies the destination IP address

AuthStatus API Output Schema

This sample schema file is the output of the AuthStatus API call after sending it to a specified session on the target Cisco Monitoring ISE node:

```
<?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: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:schema>
```

Invoking the AuthStatus API Call

Note Make sure that you have verified that the target node to which you are issuing an API call is a valid Cisco Monitoring ISE node. To verify the persona of a Cisco ISE node, see Verifying a Cisco Monitoring ISE Node, page 1-2.

To issue the AuthStatus API call, complete the following steps:

Step 1 Log into the target Cisco Monitoring ISE node.

For example, when you initially log into a Cisco Monitoring ISE node with the hostname of acme123, this would display the following URL Address field for this node:

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

Step 2 Enter the AuthStatus API call in the URL Address field of the target node by replacing the "/admin/" component with the API call component (/ise/mnt/api/<specific-api-call>/MACAddress/<macaddress>/<seconds>/<numberofrecordspermacaddress>/All):

```
https://acme123/ise/mnt/api/AuthStatus/MACAddress/00:50:56:10:13:02/120/100/All
```

Note You must carefully enter each API call in the URL Address field of a target node because these calls are case-sensitive. The use of "mnt" in the API call convention represents a Cisco Monitoring ISE node.

Step 3 Press **Enter** to issue the API call.

Sample Data Returned from the AuthStatus API Call

The following example illustrates the data returned when you invoke a AuthStatus API call on a target Cisco Monitoring ISE node:

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

-
<authStatusOutputList>
-
<authStatusList key="00:25:9C:A3:7D:48">
-
<authStatusElements>
<passed xsi:type="xs:boolean">true</passed>
<failed xsi:type="xs:boolean">false</failed>
<user_name>hareesh6</user_name>
<nas_ip_address>10.203.107.10</nas_ip_address>
<calling_station_id>00:25:9C:A3:7D:48</calling_station_id>
<nas_port>1</nas_port>
<identity_group>iPEP-WLC-Group</identity_group>
<network_device_name>iPEP3</network_device_name>
<acs_server>HAREESH-R6-1-PDP1</acs_server>
<eap_authentication>EAP-MSCHAPv2</eap_authentication>
-
<network_device_groups>
Device Type#All Device Types#iPEP,Location#All Locations
</network_device_groups>
<access_service>Default Network Access</access_service>
<acs_timestamp>2010-12-20T01:38:49.566Z</acs_timestamp>
<authentication_method>MSCHAPV2</authentication_method>
-
<execution_steps>
11001,11017,15008,15048,15048,15004,11507,12500,11006,11001,11018,12101,12100,11006,11001,
11018,12102,12800,12175,12805,12806,12801,12802,12105,11006,11001,11018,12104,12804,12816,
12132,12125,11806,12105,11006,11001,11018,12104,11808,15041,15006,15013,24210,24212,22037,
11824,12105,11006,11001,11018,12104,11810,11814,11519,12128,12105,11006,11001,11018,12104,
12126,12127,15036,15048,15048,15004,15016,12171,12105,11006,11001,11018,12104,12106,11503,
15036,15048,15048,15004,15016,11002
</execution_steps>
<audit_session_id>0acb6b0b0000000D4D0EB3A9</audit_session_id>
<nas_identifier>Cisco_4d:c0:a0</nas_identifier>
<nac_policy_compliance>NotApplicable</nac_policy_compliance>
<selected_azn_profiles>iPEP-Compliant-Authz-Profile</selected_azn_profiles>
<service_type>Framed</service_type>
<eap_tunnel>EAP-FAST</eap_tunnel>
<message_code>5200</message_code>
<destination_ip_address>10.203.107.150</destination_ip_address>
<id>1292549379215912</id>
<acsview_timestamp>2010-12-20T01:38:49.567Z</acsview_timestamp>
<acs_session_id>HAREESH-R6-1-PDP1/81999140/50</acs_session_id>
<service_selection_policy>iPEP-WLC</service_selection_policy>
<authorization_policy>iPEP-WLC-Compliant-Policy</authorization_policy>
<identity_store>Internal Users</identity_store>
-
<response>
{User-Name=hareesh6; State=ReauthSession:0acb6b0b0000000D4D0EB3A9;
Class=CACS:0acb6b0b0000000D4D0EB3A9:HAREESH-R6-1-PDP1/81999140/50;
Termination-Action=RADIUS-Request;
MS-MPPE-Send-Key=04:11:2d:bf:8b:5f:c1:b0:14:b1:73:ad:48:90:65:e0:c2:a3:f7:66:2d:dc:70:f1:a
b:56:cd:09:c4:b0:b7:ae;
MS-MPPE-Recv-Key=7e:38:94:72:e2:a3:8a:e4:90:18:45:61:91:c0:44:ea:0c:21:39:14:2f:7c:9f:55:d
6:52:af:fd:55:48:3f:34; }
```

```
</response>
-
<cisco_av_pair>
audit-session-id=0acb6b0b0000000D4D0EB3A9,ipep-proxy=true
</cisco_av_pair>
<acs_username>hareesh6</acs_username>
<radius_username>anonymous</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>Wireless - IEEE 802.11</nas_port_type>
-
<tunnel_details>
Tunnel-Type=(tag=0) VLAN,Tunnel-Medium-Type=(tag=0) 802,Tunnel-Private-Group-ID=(tag=0)
208
</tunnel_details>
-
<other_attributes>
ConfigVersionId=18,DestinationPort=1812,Protocol=Radius,Framed-MTU=1300,State=37CPMSession
ID=0acb6b0b0000000D4D0EB3A9;39SessionID=HAREESH-R6-1-PDP1/81999140/50;,Proxy-State=Cisco
Secure
ACS53e5cfac-0a31-11e0-c000-000000000000-2905701264-3372,Airespace-Wlan-Id=2,CPMSessionID=0
acb6b0b0000000D4D0EB3A9,IssuedPacInfo=Issued PAC type=Authorization with expiration time:
Mon Dec 20 02:38:49
2010,CPMSessionID=0acb6b0b0000000D4D0EB3A9,EndPointMACAddress=00-25-9C-A3-7D-48,Device
Type=Device Type#All Device Types#iPEP,Location=Location#All Locations,Model
Name=Unknown,Software Version=Unknown,Device IP
Address=10.203.107.11,Called-Station-ID=00-24-c4-1b-36-70:ipep3
</other_attributes>
<response_time>3</response_time>
</authStatusElements>
-
<authStatusElements>
<passed xsi:type="xs:boolean">true</passed>
<failed xsi:type="xs:boolean">false</failed>
<user_name>hareesh6</user_name>
<nas_ip_address>10.203.107.10</nas_ip_address>
<calling_station_id>00:25:9C:A3:7D:48</calling_station_id>
<nas_port>1</nas_port>
<identity_group>iPEP-WLC-Group</identity_group>
<network_device_name>iPEP3</network_device_name>
<acs_server>HAREESH-R6-1-PDP1</acs_server>
<eap_authentication>EAP-MSCHAPv2</eap_authentication>
-
<network_device_groups>
Device Type#All Device Types#iPEP,Location#All Locations
</network_device_groups>
<access_service>Default Network Access</access_service>
<acs_timestamp>2010-12-19T01:32:39.220Z</acs_timestamp>
<authentication_method>MSCHAPV2</authentication_method>
-
<execution_steps>
11001,11017,15008,15048,15048,15004,11507,12500,11006,11001,11018,12101,12100,11006,11001,
11018,12102,12800,12175,12805,12806,12801,12802,12105,11006,11001,11018,12104,12804,12816,
12132,12125,11806,12105,11006,11001,11018,12104,11808,15041,15006,15013,24210,24212,22037,
11824,12105,11006,11001,11018,12104,11810,11814,11519,12128,12105,11006,11001,11018,12104,
12126,12127,15036,15048,15048,15004,15016,12171,12105,11006,11001,11018,12104,12106,11503,
15036,15048,15048,15004,15016,11002
</execution_steps>
<audit_session_id>0acb6b0b0000000C4D0D60B6</audit_session_id>
<nas_identifier>Cisco_4d:c0:a0</nas_identifier>
<nac_policy_compliance>NotApplicable</nac_policy_compliance>
<selected_azn_profiles>iPEP-Compliant-Authz-Profile</selected_azn_profiles>
<service_type>Framed</service_type>
```

```
<eap_tunnel>EAP-FAST</eap_tunnel>
<message_code>5200</message_code>
<destination_ip_address>10.203.107.150</destination_ip_address>
<id>1292549379206881</id>
<acsview_timestamp>2010-12-19T01:32:39.218Z</acsview_timestamp>
<acs_session_id>HAREESH-R6-1-PDP1/81999140/46</acs_session_id>
<service_selection_policy>iPEP-WLC</service_selection_policy>
<authorization_policy>iPEP-WLC-Compliant-Policy</authorization_policy>
<identity_store>Internal Users</identity_store>
-
<response>
{User-Name=hareesh6; State=ReauthSession:0acb6b0b0000000C4D0D60B6;
Class=CACS:0acb6b0b0000000C4D0D60B6:HAREESH-R6-1-PDP1/81999140/46;
Termination-Action=RADIUS-Request;
MS-MPPE-Send-Key=f0:f4:5d:38:c4:5d:e8:85:51:65:ea:9e:ad:27:9f:c6:50:ae:11:ae:f8:8c:9d:c2:5
c:d3:33:06:36:be:14:79;
MS-MPPE-Recv-Key=d3:4a:2b:e6:6b:f8:31:ef:cc:84:d0:57:96:24:ab:e4:9b:45:3a:43:a7:1a:05:e7:5
d:a0:46:33:02:63:ef:39; }
</response>
-
<cisco_av_pair>
audit-session-id=0acb6b0b0000000C4D0D60B6,ipep-proxy=true
</cisco_av_pair>
<acs_username>hareesh6</acs_username>
<radius_username>anonymous</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>Wireless - IEEE 802.11</nas_port_type>
-
<tunnel_details>
Tunnel-Type=(tag=0) VLAN,Tunnel-Medium-Type=(tag=0) 802,Tunnel-Private-Group-ID=(tag=0)
208
</tunnel_details>
-
<other_attributes>
ConfigVersionId=18,DestinationPort=1812,Protocol=Radius,Framed-MTU=1300,State=37CPMSession
ID=0acb6b0b0000000C4D0D60B6;39SessionID=HAREESH-R6-1-PDP1/81999140/46;,Proxy-State=Cisco
Secure
ACS53e5cfac-0a31-11e0-c000-000000000000-2905701264-3372,Airespace-Wlan-Id=2,CPMSessionID=0
acb6b0b0000000C4D0D60B6,IssuedPacInfo=Issued PAC type=Authorization with expiration time:
Sun Dec 19 02:32:39
2010,CPMSessionID=0acb6b0b0000000C4D0D60B6,EndPointMACAddress=00-25-9C-A3-7D-48,Device
Type=Device Type#All Device Types#iPEP,Location=Location#All Locations,Model
Name=Unknown,Software Version=Unknown,Device IP
Address=10.203.107.11,Called-Station-ID=00-24-c4-1b-36-70:ipep3
</other_attributes>
<response_time>3</response_time>
</authStatusElements>
-
<authStatusElements>
<passed xsi:type="xs:boolean">true</passed>
<failed xsi:type="xs:boolean">false</failed>
<user_name>hareesh6</user_name>
<nas_ip_address>10.203.107.10</nas_ip_address>
<calling_station_id>00:25:9C:A3:7D:48</calling_station_id>
<nas_port>1</nas_port>
<identity_group>iPEP-WLC-Group</identity_group>
<network_device_name>iPEP3</network_device_name>
<acs_server>HAREESH-R6-1-PDP1</acs_server>
<eap_authentication>EAP-MSCHAPv2</eap_authentication>
-
<network_device_groups>
Device Type#All Device Types#iPEP,Location#All Locations
```

```
</network_device_groups>
<access_service>Default Network Access</access_service>
<acs_timestamp>2010-12-18T01:26:22.089Z</acs_timestamp>
<authentication_method>MSCHAPV2</authentication_method>
-
<execution_steps>
11001,11017,15008,15048,15048,15004,11507,12500,11006,11001,11018,12101,12100,11006,11001,
11018,12102,12800,12805,12806,12807,12810,12105,11006,11001,11018,12104,12812,12804,12801,
12802,12816,12149,12105,11006,11001,11018,12104,12125,11521,12105,11006,11001,11018,12104,
11522,11806,12105,11006,11001,11018,12104,11808,15041,15006,15013,24210,24212,22037,11824,
12105,11006,11001,11018,12104,11810,11814,11519,12128,12105,11006,11001,11018,12104,12126,
12127,15036,15048,15048,15004,15016,12169,12105,11006,11001,11018,12104,12651,12107,11503,
15036,15048,15048,15004,15016,11002
</execution_steps>
<audit_session_id>0acb6b0b0000000B4D0C0DBD</audit_session_id>
<nas_identifier>Cisco_4d:c0:a0</nas_identifier>
<nac_policy_compliance>NotApplicable</nac_policy_compliance>
<selected_azn_profiles>iPEP-Compliant-Authz-Profile</selected_azn_profiles>
<service_type>Framed</service_type>
<eap_tunnel>EAP-FAST</eap_tunnel>
<message_code>5200</message_code>
<destination_ip_address>10.203.107.150</destination_ip_address>
<id>1292549379197803</id>
<acsview_timestamp>2010-12-18T01:26:22.042Z</acsview_timestamp>
<acs_session_id>HAREESH-R6-1-PDP1/81999140/30</acs_session_id>
<service_selection_policy>iPEP-WLC</service_selection_policy>
<authorization_policy>iPEP-WLC-Compliant-Policy</authorization_policy>
<identity_store>Internal Users</identity_store>
-
<response>
{User-Name=hareesh6; State=ReauthSession:0acb6b0b0000000B4D0C0DBD;
Class=CACS:0acb6b0b0000000B4D0C0DBD:HAREESH-R6-1-PDP1/81999140/30;
Termination-Action=RADIUS-Request;
MS-MPPE-Send-Key=d3:94:df:2b:fc:18:12:91:ad:4f:3b:09:d1:76:93:83:21:83:33:3a:14:b9:9b:c0:a
0:81:71:96:95:64:2c:ed;
MS-MPPE-Recv-Key=3b:c2:31:58:81:8a:34:24:d4:55:03:cd:a2:91:85:49:7f:16:36:30:d9:8d:24:a7:5
0:ec:3e:df:7a:85:ea:5c; }
</response>
-
<cisco_av_pair>
audit-session-id=0acb6b0b0000000B4D0C0DBD,ipep-proxy=true
</cisco_av_pair>
<acs_username>hareesh6</acs_username>
<radius_username>anonymous</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>Wireless - IEEE 802.11</nas_port_type>
-
<tunnel_details>
Tunnel-Type=(tag=0) VLAN,Tunnel-Medium-Type=(tag=0) 802,Tunnel-Private-Group-ID=(tag=0)
208
</tunnel_details>
-
<other_attributes>
ConfigVersionId=18,DestinationPort=1812,Protocol=Radius,Framed-MTU=1300,State=37CPMSession
ID=0acb6b0b0000000B4D0C0DBD;39SessionID=HAREESH-R6-1-PDP1/81999140/30;,Proxy-State=Cisco
Secure
ACS53e5cfac-0a31-11e0-c000-000000000000-2905701264-3372,Airespace-Wlan-Id=2,CPMSessionID=0
acb6b0b0000000B4D0C0DBD,IssuedPacInfo=Issued PAC type=Tunnel V1 with expiration time: Fri
Mar 18 01:26:22
2011,CPMSessionID=0acb6b0b0000000B4D0C0DBD,EndPointMACAddress=00-25-9C-A3-7D-48,Device
```

```
Type=Device Type#All Device Types#iPEP,Location=Location#All Locations,Model
Name=Unknown,Software Version=Unknown,Device IP
Address=10.203.107.11,Called-Station-ID=00-24-c4-1b-36-70:ipep3
</other_attributes>
<response_time>3</response_time>
</authStatusElements>
</authStatusList>
</authStatusOutputList>
```

Account Status API Call

You can use the AcctStatus API call to retrieve the latest device and session account information on the target node. This section provides a schema file output example, a procedure for sending a request for the latest device and session information by invoking this API call, and a sample of the data returned after this API call is issued. The AcctStatus API call lets you configure a time-related parameter:

- Duration—Defines the number of seconds in which an attempt is made to search and retrieve the latest account device records associated with the designated MAC address. Valid user-configurable values range from 1 to 432000 seconds (5 days).
  - If you enter a value of 2400 seconds (40 minutes), this means that you want the latest account device records for the designated MAC address that are available in the past 40 minutes.
  - If you enter a value of 0 seconds, this specifies a default duration of 15 minutes (900 seconds). This means that you want the latest account device records for the designated MAC address that are available within this time period.

The AcctList API call provides the following account status data fields as API outputs (see Table 3-3):

Table 3-3 Accounting Status Data Fields

Data Field	Description
MAC address	MAC address of the client
audit-session-id	Authentication session ID
Packets in	Packets received count total
Packets out	Packets transmitted count total
Bytes in	Bytes received count total
Bytes out	Bytes transmitted count total
Session time	Duration of current sessions

AcctStatus API Output Schema

This sample schema file is the output of the AcctStatus API call after sending it to a specified session on the target Cisco Monitoring ISE node:

```
<?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>
```

Invoking the AcctStatus API Call

Note Make sure that you have verified that the target node to which you are issuing an API call is a valid Cisco Monitoring ISE node. To verify the persona of a Cisco ISE node, see Verifying a Cisco Monitoring ISE Node, page 1-2.

To issue the AcctStatus API call, complete the following steps:

Step 1 Log into the target Cisco Monitoring ISE node.

For example, when you initially log into a Cisco Monitoring ISE node with the hostname of acme123, this would display the following URL Address field for this node:

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

Step 2 Enter the AcctStatus API call in the URL Address field of the target node by replacing the "/admin/" component with the API call component (/ise/mnt/api/<specific-api-call>/MACAddress/<macaddress>/<durationofcurrenttime>):

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

Note You must carefully enter each API call in the URL Address field of a target node because these calls are case-sensitive. The use of "mnt" in the API call convention represents a Cisco Monitoring ISE node.

Step 3 Press **Enter** to issue the API call.

Sample Data Returned from the AcctStatus API Call

The following example illustrates the data returned when you invoke an AcctStatus API call on a target Cisco Monitoring ISE node:

```
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>0acb6b0b0000000B4D0C0DBD</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>
```