Query APIs for Troubleshooting

This chapter provides examples and describes how to use the individual Cisco Prime Network Control System (NCS) REST API calls.

Cisco Prime NCS API Calls

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.

Troubleshooting Cisco ISE using the Query API Calls

Cisco Prime NCS troubleshooting API calls send status requests to the target Cisco Monitoring ISE node 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

Step 1   Enter the Cisco ISE URL in the address bar of your browser (for example, *https://<ise hostname or ip address>/admin/*).

Step 2   Enter the username and case-sensitive password, that was specified and configured during the initial Cisco ISE setup.

Step 3   Click **Login** or press **Enter**.

If your login is unsuccessful, click the **Problem logging in?** link in the Login page and follow the instructions in Step 2.

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

```
https://acme123/admin/API/mnt/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 5   Press **Enter** to issue the API call.

Related Topics

- Verifying a Monitoring Node, page 1-2

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 Cisco ISE Failure Reasons Report, 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

Step 1   Enter the Cisco ISE URL in the address bar of your browser (for example, *https://<ise hostname or ip address>/admin/*).

Step 2   Enter the username and case-sensitive password, that was specified and configured during the initial Cisco ISE setup.

Step 3   Click **Login** or press **Enter**.

If your login is unsuccessful, click the **Problem logging in?** link in the Login page and follow the instructions in Step 2.

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

```
https://acme123/admin/API/mnt/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 5   Press **Enter** to issue the API call.

Related Topics

• Verifying a Monitoring Node, page 1-2

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>
```

Related Topics

-
-

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).

Here is an example of the generic form of a URL with the Duration and Records attributes:

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

- 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 3-12).

    - 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" or name="failed"	Authentication status results: • Passed • Failed
name="user_name"	User name
name="nas_ip_address"	IP address/hostname for the network access device
name="nas_ipv6_address"	IPv6 address/hostname for the network access device
name="failure_reason"	Reason for session authentication failure
name="calling_station_id"	Source IP address
name="nas_port"	Network access server port
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
name="framed_ipv6_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

Table 3-2   Authentication Status Table Attributes (continued)

Attribute	Description
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: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>
```

Invoking the AuthStatus API Call

Step 1 Enter the Cisco ISE URL in the address bar of your browser (for example, *https://<ise hostname or ip address>/admin/*).

Step 2 Enter the username and case-sensitive password, that was specified and configured during the initial Cisco ISE setup.

Step 3 Click **Login** or press **Enter**.

If your login is unsuccessful, click the **Problem logging in?** link in the Login page and follow the instructions in Step 2.

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

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

✎

Note The REST API calls are case-sensitive. The use of "mnt" in the API call convention represents a Cisco Monitoring ISE node.

Step 5 Press **Enter** to issue the API call.

Related Topics

- Verifying a Monitoring Node, page 1-2

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="00:0C: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=Defau
ltGuestPortal,
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>
```

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). For example,

  – 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

Step 1   Enter the Cisco ISE URL in the address bar of your browser (for example, *https://<ise hostname or ip address>/admin/*).

Step 2   Enter the username and case-sensitive password, that was specified and configured during the initial Cisco ISE setup.

Step 3   Click **Login** or press **Enter**.

If your login is unsuccessful, click the **Problem logging in?** link in the Login page and follow the instructions in Step 2.

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

```
https://acme123/admin/API/mnt/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 5 Press **Enter** to issue the API call.

Related Topics

- Verifying a Monitoring Node, page 1-2

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>
```