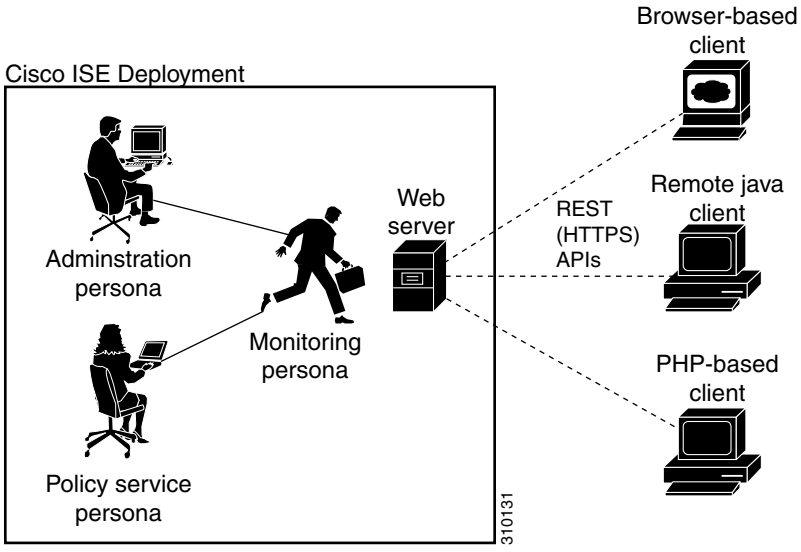
Introduction to the Monitoring REST API

The Monitoring REST API allows allow you to gather session and node-specific information by using Monitoring nodes in your network. A session is defined as the duration between when you access a desired node and complete the operations needed to gather information.

Monitoring REST API calls allow you to locate, monitor, and accumulate important real-time, session-based information stored in individual endpoints in a network. You can access this information through a Monitoring node.

The real-time, session-based information that you gather can help understand Cisco ISE operations and assist in diagnosing conditions or issues. It can also be used to troubleshoot error conditions or an activity or behavior that may be affecting monitoring operations. As shown in Figure 1-1, the Monitoring REST API calls are used to access the Monitoring node and retrieve important session-based information that is stored in the Cisco ISE deployment endpoints.

Figure 1-1 Monitoring REST API Calls in a Distributed Deployment



To perform operations using the Monitoring REST APIs, the users must be assigned to one of the following Admin Groups and must be authenticated against the credentials stored in the Cisco ISE internal database (internal admin users):

- Super Admin
- System Admin
- MnT Admin

The following Monitoring REST API categories are supported:

- Session Management
- Troubleshooting
- Change of Authorization (CoA)

You can use these APIs to gather information about endpoints being monitored by the Monitoring persona. For the remainder of this guide, “Monitoring node” will be used to describe the Monitoring persona of a Cisco ISE node.

Any attempt to use these categories to gather information about the Policy Service persona of a Cisco ISE appliance will result in an error. For more information about Cisco ISE nodes and personas, see *Cisco Identity Services Engine Admin Guide*.

Verifying a Monitoring Node

Before you Begin

Before you can successfully invoke the API calls on a Monitoring node, you need to verify that the node you want to monitor is valid.

Note To be able to use a public Monitoring REST API, you must first authenticate with Cisco ISE using valid credentials.

Step 1 Enter valid login credentials (Username and Password) in the Cisco ISE Login window, and click **Login**.

The Cisco ISE dashboard and user interface appears.

Step 2 Choose **Authorization > System > Deployment**.

The Deployment Nodes page appears, which lists all configured nodes that are deployed.

Step 3 In the Roles column of the Deployment Nodes page, verify that the role for the target node that you want to monitor is listed as a Monitoring node.

Supported API Calls

The following tables describe the different types of API calls and provide an example of the API call format:

- Table 1-1 on page 1-3—defines API calls for session management.
- Table 1-2 on page 1-6—defines API calls for troubleshooting.
- Table 1-3 on page 1-7—defines CoA API calls.

If you intend to use a generic programmatic interface to authenticate with the Monitoring REST API supported by Cisco ISE, you need to first create a REST-based client that bridges Cisco ISE and the specific tool you use. You then use this REST client to authenticate with the Cisco ISE Monitoring REST APIs, marshal and submit the API requests to the Monitoring nodes, and then unmarshal the API responses and pass them on to the specified tool.

Table 1-1 Cisco ISE Session Management API Calls

API Call Category	Description and Example
Session Counters	
ActiveCount	Lists the number of active sessions. https://<ISEhost>/admin/API/mnt/Session/ActiveCount **Note** You must add the HTTP authorization header with the authorization credentials to view the number of active sessions.
PostureCount	Lists the number of Postured endpoints. https://<ISEhost>/admin/API/mnt/Session/PostureCount **Note** Posture is a service that aids in checking the state (or posture) for all the endpoints that connect to a Cisco ISE network. Cisco ISE utilizes NAC Agent for checking the posture compliance of a device.
ProfilerCount	Lists the number of active Profiler service sessions. https://<ISEhost>/admin/API/mnt/Session/ProfilerCount **Note** Profiler is a service that aids in identifying, locating, and determining the capabilities of all attached endpoints on a Cisco ISE network.

Table 1-1 Cisco ISE Session Management API Calls (continued)

API Call Category	Description and Example
Session List **Note** A session list includes the MAC address, network access device (NAD) IP address, username, and session ID information associated with a session.	
ActiveList	Lists all active sessions. https://<ISEhost>/admin/API/mnt/Session/ActiveList **Note** In this release of Cisco ISE, the maximum number of active authenticated endpoint sessions that can be displayed is limited to 250,000.
AuthList	Lists all currently active authenticated sessions. https://<ISEhost>/admin/API/mnt/Session/AuthList/<parameteroptions> You can specify the following parameter options that will return different values: • null/null—Lists all active authenticated sessions. • null/endtime—Lists all active authenticated sessions after the specified end time. • starttime/null—Lists all active authenticated sessions before the specified start time. • starttime/endtime—Lists all active authenticated sessions between the specified start time and end time. Enter the date and time for the start time and end time in the following format: YYYY-MM-DD hh:mm:ss.s where: • YYYY—four-digit year • MM—two-digit month (01=January, and so on) • DD—two-digit day of the month (01 through 31) • hh—two-digit hour (00 through 23) (a.m. and p.m. are not allowed) • mm—two-digit minute (00 through 59) • ss—two-digit second (00 through 59) • s—one or more digits representing a decimal fraction of a second **Note** Every Cisco ISE node is configured with a time zone. Recommended time zone is UTC. See Sample Data Returned from the AuthList API Call with the null/null Option, page 2-9, for samples that show all four parameter options.

Table 1-1 Cisco ISE Session Management API Calls (continued)

API Call Category	Description and Example
Session Attributes **Note** This is a timestamp-based search for the latest session that contains the specified search attribute.	
MACAddress	Searches the database for the latest session that contains the specified MAC address. https://<ISEhost>/admin/API/mnt/Session/MACAddress/<macaddress> **Note** XX:XX:XX:XX:XX:XX is the MAC address format and is not case sensitive (for example, 0a:0B:0c:0D:0e:0F). **Note** The MAC address serves as the only unique key to finding the correct session you want to monitor. Use the ActiveList API call to list all active sessions and their MAC addresses, from which you can base your MAC address search.
UserName	Searches the database for the latest session that contains the specified username. https://<ISEhost>/admin/API/mnt/Session/UserName/<username> **Note** Usernames must conform to the same Cisco ISE password policy used for network usernames. The only invalid character for the Monitoring REST APIs is the backslash (\) character. For details, see "User Password Policy" in *Cisco Identity Services Engine User Guide, Release 1.1*.
IPAddress	Searches the database for the latest session that contains the specified NAS IP address (IPv4 or IPv6 address). https://<ISEhost>/admin/API/mnt/Session/IPAddress/<nasipaddress> **Note** xxx.xxx.xxx.xxx is the NAS IP address format (for example, 10.10.10.10) or https://<ISEhost>/admin/API/mnt/Session/IPAddress/<nasipv6address> **Note** xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx is the NAS IPv6 address format (for example, 2001:cdba:0:0:0:0:3247:9651)
Audit Session ID	Searches the database for the latest session that contains the specified audit session ID. https://<ISEhost>/admin/API/mnt/Session/Active/SessionID/<audit-session-id>/0 **Note** Use the ActiveList API call to list all active sessions and their audit session IDs, from which you can base your audit session ID search. Alternatively, you can obtain the audit session ID from the Live Sessions page in the Admin portal.

For specific details about Cisco ISE API calls for session management, see Chapter 2, "Session Management Query APIs".

Table 1-2 Cisco ISE Troubleshooting API Calls - Troubleshooting

API Call	Description and Example
Version	Lists the node version and type. https://<ISEhost>/admin/API/mnt/Version Node type can be any of the following values (0-3): 0—STAND_ALONE_MNT_NODE 1—ACTIVE_MNT_NODE 2—STAND_BY_MNT_NODE 3—NOT_AN_MNT_NODE **Note** STAND_ALONE_MNT_NODE means it is a Monitoring node that does not function in any distributed deployment. ACTIVE_MNT_NODE means it is a primary node in a primary-secondary relationship in a distributed deployment. STAND_BY_MNT_NODE means it is a secondary node in a primary-secondary pair in a distributed deployment. NOT_AN_MNT_NODE means it is not a Monitoring node. See *Cisco Identity Services Engine User Guide, Release 1.1* for details about the supported ISE nodes and personas.
FailureReasons	Lists the reasons for failure. https://<ISEhost>/admin/API/mnt/FailureReasons Each failure reason displays an error code (failureReason id), a brief description (code), a failure reason (cause), and a possible response (resolution), as shown in the following example: <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. **Note** The FailureReasons API call to be called only once to gather the information from the Monitoring node. You should store the contents of any returned failure reasons into your own file system or database. The returned contents of these API calls are intended to be used for reference purposes. If you experience any issues during authentication, you should compare the failure reason code provided in the authentication response with the list of failure reasons that you have stored in your own file system or database. For a complete list of Cisco ISE failure reasons, see Appendix A, "Cisco ISE Failure Reasons Report".

Table 1-2 Cisco ISE Troubleshooting API Calls - Troubleshooting (continued)

API Call	Description and Example
AuthStatus	Lists the authentication status for all sessions. https://<ISEhost>/admin/API/mnt/AuthStatus/MACAddress/<macaddress>/<numberofseconds>/<numberofrecordspermacaddress>/All **Note** The seconds parameter <numberofseconds> is user-configurable, the range is from 0 to 432000 seconds (5 days).
Get Session Accounting Status	
AcctStatus	Lists the accounting status of all sessions within a specific period of time. https://<ISEhost>/admin/API/mnt/AcctStatusTT/MACAddress/<macaddress>/<numberof seconds> **Note** The seconds parameter <numberofseconds> is user-configurable, with the range is from 0 to 432000 seconds (5 days).

For specific details about Cisco ISE API calls for troubleshooting, see Chapter 2, "Session Management Query APIs".

Table 1-3 Cisco ISE Change of Authorization API Calls

API Call	Description and Example
Reauth	Sends a session reauthentication command and type. https://<ISEhost>/admin/API/mnt/CoA/Reauth/<serverhostname>/<macaddress>/<reauthtype>/<nasipaddress>/<destinationipaddress> Where <*ISEhost*> denotes the ip address of the ISE host, <*serverhostname*> denotes the name of the ISE server, <*nasipaddress*> denotes the identifying ip address of NAS, and <*destinationipaddress*> denotes the ip address of the destination. Reauth type can be any of the following values (0-2): 0—REAUTH_TYPE_DEFAULT 1—REAUTH_TYPE_LAST 2—REAUTH_TYPE_RERUN **Note** If you do not know the NAS IP address, you can enter the required values up to that point and the API will use these values in its search query. However, you must know the MAC address to perform this API call, but you can leave other parameters starting from NAS IP address as null. If the NAS IP address is provided then it's necessary to also provide the Destination IP address. This API call can only be executed on a Monitoring ISE node, which submits the requests to perform CoA remotely. The Administration ISE node is not involved or required to execute these CoA API calls.

Table 1-3 Cisco ISE Change of Authorization API Calls (continued)

API Call	Description and Example
Session Disconnect	
Disconnect	Sends a session disconnect command and port option type. https://<ISEhost>/admin/API/mnt/CoA/Disconnect/<serverhostname>/<macaddress>/<disconnecttype>/<nasipaddress>/<destinationipaddress> Port option type can be any of the following values (0-2): 0—DYNAMIC_AUTHZ_PORT_DEFAULT 1—DYNAMIC_AUTHZ_PORT_BOUNCE 2—DYNAMIC_AUTHZ_PORT_SHUTDOWN **Note** If you do not know the NAS IP address, enter the required values up to that point and the API will use these values in its search query. However, you must know the MAC address to perform this API call, but you can leave other parameters as null.

For details about Cisco ISE Change of Authorization API calls, see Chapter 4, "Change of Authorization REST APIs".

HTTP PUT API Calls

Similar to AuthStatus API call in Table 1-2, there is an HTTP PUT version of an API call that allows clients to retrieve account status. The Monitoring REST API supports both HTTP PUT and HTTP GET calls, with the examples in this guide documenting HTTP GET calls. HTTP PUT addresses the need for calls that require parameter inputs. The following schema file example is a request for account status:

```
<?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>
```