



CHAPTER 1

Introduction to the Cisco ANM Web Services API

This chapter describes the Cisco ANM Web Services application programming interface (API), which provides a programmable interface for system developers to integrate with customized or third-party management applications.

This chapter contains the following sections:

- [ANM Web Services API Overview, page 1-1](#)
- [Web Services Description Language, page 1-2](#)
- [Using the ANM Web Services API, page 1-3](#)

ANM Web Services API Overview

ANM Web Services provides APIs that support the operations for Application Control Engine (ACE) modules and appliances and Cisco Content Services Switch (CSS) and Cisco Content Switching Module (CSM) devices (for restrictive APIs) with all available releases. This API allows configuration functionalities.

ANM is a Java application that uses Java SE 6. Java SE 6 is a standard with Web Services support in the form of JSR 224, Java API for XML-Based Web Services (JAX-WS) 2.0. ANM makes use of Java SE 6 Web Services support to implement its Web Services-based NBI. Java SE 6 also supports JSR 181, Web Services Metadata for the Java Platform. JSR 181 makes it possible to develop web services. Additionally, Java SE 6 makes it extremely easy to publish a Java Web Service by using `javax.xml.ws.Endpoint` of JAX-WS.

The ANM Web Services API is a Web Service implementation. Web Service is defined by the W3C standard as a software system designed to support interoperable machine-to-machine (client and server) interaction over the network. The client and server communication follows the Simple Object Access Protocol or Service Oriented Architecture Protocol (SOAP) standard.

SOAP, which exchanges XML-based messages over the network using HTTP or HTTPS, is the foundation layer of the Web Service stack. It provides a basic messaging framework that allows more abstract layers to build on. SOAP encoding wraps XML headers and tags in a SOAP envelope.

To call a service, you connect to a particular ANM through a web browser by using a service URL that contains the IP address or hostname of the ANM and the name of the particular service (such as `OperationManager`). For example, `https://<host/ip>:8443/anm/OperationManager?wsdl` is the service URL for the Operation Manager Service. By default, ANM 4.3 supports up to 10 concurrent sessions for web services (see the [“Modifying the Number of Concurrent Sessions Allowed”](#) section on page 4-3).

Next, you must post a SOAP request written in a XML format to retrieve the information. The request calls for a particular action (such as Login) and the input parameter content when required. The ANM responds with a SOAP envelope that contains the answer in an XML format. The response contains the output values for this action.

The following example shows an XML-formatted SOAP request to perform the [Login](#) action. There are no input parameters for this particular action.

Example Request

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Body>
    <login
      xmlns="http://anm.cisco.com">
      <user
        xmlns="">admin</user>
      <password xmlns="">admin</password>
    </login>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

The following example shows the XML response that contains the output values for this action, such as the SessionToken.

Example Response

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope
  xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:loginResponse
      xmlns:ns2="http://anm.cisco.com">
      <SessionToken xmlns="">
        <sid>5546099606143734326</sid>
      </SessionToken>
    </ns2:loginResponse>
  </S:Body>
</S:Envelope>
```

Enabling the Web Services Interface

After you have installed the ANM, you can reconfigure ANM configuration properties, such as enabling HTTP(S) for Web Services. For information about enabling Web Services, see the “[Changing ANM Configuration Property Values](#)” section in the *User Guide for the Cisco Application Networking Manager 4.3*.

Web Services Description Language

In the ANM Web Services API, Web Services Description Language (WSDL) is used with SOAP and XML schemas to provide Web Services. WSDL is an XML-based service that describes the functionality offered by the Web Service and defines the actions, parameter names, input parameter data types, and

return data types for the Web Service. When you connect to a Web Service through a web browser, you can read the WSDL file to determine which functions are available on the server. Any special data types that are used are embedded in the WSDL file in an XML schema. You can then call one of the functions listed in the WSDL file by sending a SOAP request message.

To obtain the WSDL file defined for a particular service in the ANM Web Services API implementation, submit a URL to the service with a **?wsdl** suffix. For example, to retrieve the WSDL for the OperationManager service running on `https://localhost:8443/anm/OperationManager`, call the WSDL file by using the following URL: `https://<host/ip>:8443/anm/OperationManager?wsdl`.

Using the ANM Web Services API

This section describes how to use the ANM Web Services API. It contains the following topics:

- [Required Software, Web Standards, and Supported Hardware, page 1-3](#)
- [Generating the Client Code to Invoke a Web Service, page 1-3](#)

Required Software, Web Standards, and Supported Hardware

The Web Services API is supported in ANM 4.1 and later. The API examples in this document require the following development environment:

- Java SE 6



Note Java SE 6 is applicable only if the client code is java-based.

- WSDL Support: 1.1
- SOAP 1.1

The ANM Web Services API is supported on the following product types:

- Cisco Application Control Engine (ACE) module with ANM 4.1 and later.
- Cisco Application Control Engine (ACE) appliance with ANM 4.1 and later.
- Cisco Content Services Switch (CSS) with ANM 4.1 and later.
- Cisco Content Switching Module (CSM) and Cisco Content Switching Module with SSL (CSM-S) with ANM 4.1 and later.
- Cisco Global Site Selector (GSS) with ANM 4.3 and later.

Generating the Client Code to Invoke a Web Service

To generate the client source code, follow these steps:

-
- Step 1** Download the WSDL file from the server using the following WSDL URL format:
- ```
https://<host/ip>:8443/anm/OperationManager?wsdl
```
- or
- ```
http://<host/ip>:8080/anm/OperationManager?wsdl
```

where the *host/ip* value is the hostname or IP address of the ANM that has the service running.

Step 2 Save the file as `OperationManager.wsdl`.

Step 3 Use the `wsimport` command to generate and compile the source code:

wsimport -d *<specify where to place generated output files>* **-s** *<specify where to place generated source files>* **OperationManager.wsdl**

The above command generates the java source code from the wsdl file and then compiles it and puts the class file in the same location where the source file `.wsimport` file is located in the dir `$JAVA_HOME/bin`.



Note This step is applicable only if the client code is a java-based code.
