CMDP-LIMS INTERFACE CONTROL DOCUMENT

Prepared for: WILL BOWMAN
Product Owner
U.S. EPA
OFFICE OF WATER

Prepared by: ATTAIN, LLC

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

INTRODUCTION

ABOUT THIS DOCUMENT

The Compliance Monitoring Data Portal (CMDP) Web Services Laboratory Information Management System (LIMS) Interface Control Document (ICD) is intended for Labs and public water systems (PWS) using LIMS to connect their existing LIMS to CMDP and submit sampling data electronically. This document identifies and describes the technical aspects needed to connect the CMDP LIMS Web Services in order to electronically report sample results to CMDP. Once the connection between LIMS and CMDP is created and sampling data are sent, all future actions (review, certify, submit) are performed via the CMDP user interface (UI). *See the CMDP User Manual for more information on submitting sampling data via CMDP LIMS Web Services.*

A.1.1 Intended Audience

The intended audiences of the CMDP Web Services LIMS ICD are laboratories and public water systems (PWS) using LIMS, as well as their vendors.

Laboratory, State Laboratory, and PWS with Certifier and Administrator roles can submit data to CMDP. *See the CMDP User Manual for more information on CMDP roles.*

A.1.2 Acronyms and Definitions

Table 1 – List of Acronyms and Definitions Commonly Used throughout the Document

Acronym	Definition
EPA	Environmental Protection Agency
CMDP	Compliance Monitoring Data Portal
CROMERR	Cross-Media Electronic Reporting Rule
LIMS	Laboratory Information Management System
NPDWRs	National Primary Drinking Water Regulations
PWS	Public Water System
REST	Representational State Transfer
SDWA	Safe Drinking Water Act
SCS	Shared CROMERR Services
UI	User Interface

SCOPE OF THIS DOCUMENT

The scope of this document is strictly limited to the description of a solution aimed at providing electronic reporting of compliance sample results from laboratories and public water systems using web services.

The CMDP can receive and accept Sampling Data using the following methods:

- Online web-based user interface

- Microsoft Excel-based data entry templates that can be downloaded from the CMDP home page, updated locally, and loaded into CMDP using the CMDP web application
- CMDP web services

This document focuses on the CMDP web service methods for reporting sample results.

None of the contents of this document supersede the statutory requirements of the Safe Drinking Water Act (SDWA); the regulatory reporting requirements of the Code of Federal Regulations (CFR) Parts 141 and 142, including public notification requirements; state[1] regulatory requirements; or the existing reporting relationships between the U.S. Environmental Protection Agency, primacy agencies, public water systems, and laboratories. CMDP is information technology that facilitates the electronic reporting of compliance sample results from laboratories and public water systems to primacy agencies.

SYSTEM OVERVIEW

The Compliance Monitoring Data Portal (CMDP) is a web-based application, relational database, and suite of web services aimed at providing the following:

- Electronic reporting of chemical, radiological, microbiological and operational sample results from laboratories and public water systems;
- Compliance with EPA's CROMERR; and
- Primacy agency access to view and download sample results to support compliance determinations using their state compliance databases.

A.1.3 Supporting Systems and Services

The following systems and services are necessary for LIMS-based reporting to CMDP:

1) **Shared CROMERR Services (SCS)** – A suite of CROMERR-compliant web services that support centralized access to the following:
    a) Registration and Account Management
    b) Identity Management
    c) Signature Device
    d) Signature and Copy of Record Generation

    For more information about SCS and interactions with CMDP, please see the CMDP to SCS JMS document and the SCS portion of the User Manual.
2) **Laboratory Information Management System (LIMS)** – Many laboratories, or water utilities with laboratories, have acquired and are using LIMS. Laboratories can use their LIMS to report data to CMDP electronically using web services or data exchange templates.

1 For the purposes of this CMDP Web Services document, the term "state" represents a government agency with primary enforcement authority (primacy) for National Primary Drinking Water Regulations (NPDWRs) and any state-specific drinking water regulations. A "state," therefore, includes a state government entity, Tribal entity, or county government entity.

CMDP WEB SERVICES

CMDP WEB SERVICE OVERVIEW

The CMDP Representational State Transfer (REST) based web services are intended to be invoked from an external application like a LIMS to upload sample results into the CMDP system. The details of the process are provided in this section.

Beyond this document, the CMDP Team will provide the following additional supportive information to support LIMS integration. All Documentation and supportive information is available on the CMDP Helpdesk site: https://cmdp.zendesk.com

- XML Schema Definition (XSD) files that can be used to validate XML submissions before submitting to CMDP.
- Example XML files that can be used to test the integration of LIMS with CMDP. This will allow LIMS developers to, initially, test integration with CMDP without worrying about formatting data.
- The Web Application Description Language (WADL) is a machine-readable XML description of HTTP-based web applications (typically REST web services). WADL models the resources provided by a service and the relationships between them. WADL is intended to simplify the reuse of web services that are based on the existing HTTP architecture of the Web. It is platform- and language-independent and aims to promote reuse of applications beyond basic use in a web browser.

The purpose of CMDP Pre-Production / Training environment is to provide a place to train users on the usage of the CMDP web application and CMDP integration with LIMS. Once the connection is confirmed, LIMS can be connected to the CMDP Production environment at NCC.

CMDP WEB SERVICE SECURITY

A.1.4 Authentication

A.1.4.1 DESCRIPTION

Authentication occurs as part of a single web service request, not as a separate request. The Basic Authentication mechanism provided by Spring Security was implemented for CMDP Web Services. As part of the web service request, Security credentials are passed from client to server via the **HTTP request header that contains 'userId:password' encoded in base64.**

Users will obtain their credentials through SCS registration and it will be the same userId:password used to login to CMDP. *For more information about SCS and interactions with CMDP, please see the SCS portion of the CMDP User Manual.*

In order to perform any operations using CMDP web services, all requests must contain the user's security credentials in the request header. Only after authentication is successful will the

web request be passed to the REST API and processed. Table 5, below, illustrates the REST API Authentication process.

Note: The security credentials (required) are part of a request, not a separate request.

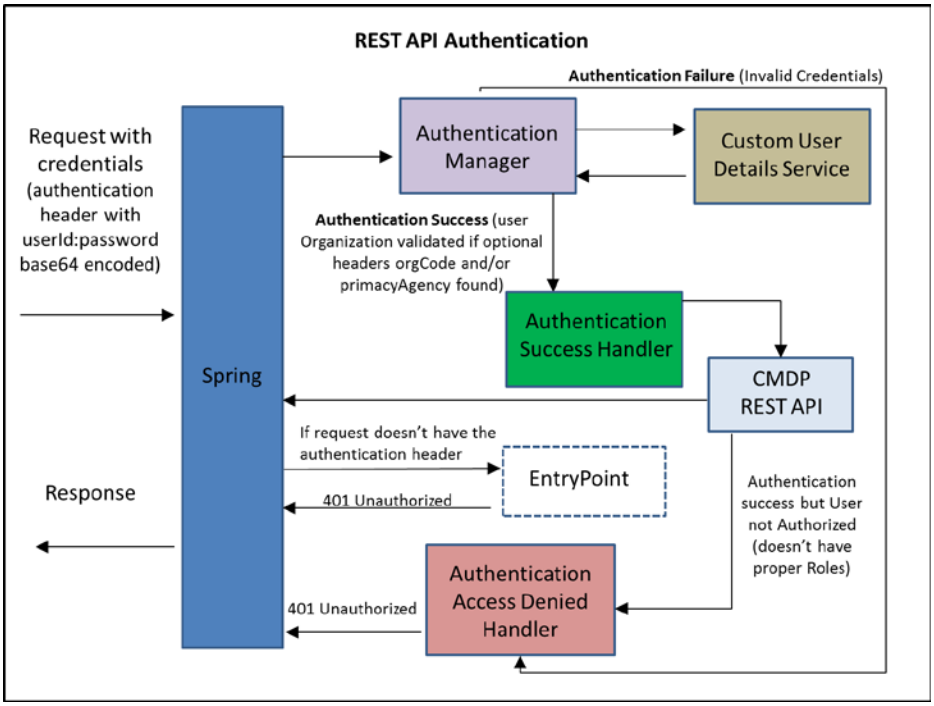

Figure 1 – REST API Authentication

Note: The following are not scenarios for a specific web service endpoint (method)—they are presented to illustrate the responses from the authentication process that occurs as part of a request.

A.1.4.2 REQUEST

Table 2 – Authorization Request Header

Name	Required	Definition / Constraints
Authorization	Yes	SCS User ID and Password encoded in base64

Example 1 – Authorization Request Header (for User ID "userA" with Password "password")

```
Authorization: Basic dXNlckE6cGFzc3dvcmQ=
                     userId:password encoded in base64
```

A.1.4.3 RESPONSE

If authentication is successful, the request will be processed and a response returned. If authentication fails, then the response will raise an exception and return an error message.

Table 3 – Unsuccessful Authentication Scenarios

Scenario	HTTP Response Status Code	Response
No security credentials provided	401 Unauthorized	`<?xml version="1.0" encoding="UTF-8" standalone="yes"?><response><endRow>0</endRow><errorMessage>Full authentication is required to access this resource</errorMessage><queueStatus>0</queueStatus><startRow>0</startRow><status>401</status><totalRows>0</totalRows></response>`
Invalid Security Credentials (i.e., User ID not valid)	401 Unauthorized	`<?xml version="1.0" encoding="UTF-8" standalone="yes"?><serverResponse><response><endRow>0</endRow><errorMessage>Invalid User: userA</errorMessage><queueStatus>0</queueStatus><startRow>0</startRow><status>401</status><totalRows>0</totalRows></response></serverResponse>`
Valid security credentials, but User Not Authorized (based on roles)	401 Unauthorized	`<?xml version="1.0" encoding="UTF-8" standalone="yes" ?><serverResponse><response><endRow>0</endRow><errorMessage>Requested resource 'https://toBeDeterminedServer/cmdp-webservice/api/submissions/sampleData' is not available for this user: stadmin. Applicable Roles for This Resource: [ROLE_WS_MODE, ROLE_LB_MODE, ROLE_LS_MODE]</errorMessage><queueStatus>0</queueStatus>`

```
                                        <startRow>0</startRow>
                                        <status>401</status>
                                        <totalRows>0</totalRows>
                                    </response>
                                </serverResponse>
```

A.1.5 User Organization

A.1.5.1 DESCRIPTION

Users may be associated with multiple organizations in SCS. Optional request headers can be used to specify the user organization to be associated with the web service request for CMDP. If optional headers are not specified, the user's default Organization is used. If specified, the user organization will be validated as part of the authentication process. Only after successful validation of the user organization will the web service request be processed.

The optional request headers for User Organization are detailed in Table 4.

A.1.5.2 REQUEST

Table 4 – User Organization Request Header (optional)

Name	Required	Definition / Constraints
orgCode	No	Organization Code
primacyAgency	No	Primacy Agency

Example 2 - Optional Request Header to Specify User Organization

```
Authorization: Basic dXNlckE6cGFzc3dvcmQ=
orgCode: TX9000001
primacyAgency: TX
```

Section A.1.6 details the web service method to retrieve the list of organizations that the user is associated with in SCS.

A.1.5.3 RESPONSE

If valid, the user role and organization are updated accordingly and the request is processed. If not valid, the web service request will not be processed and the response will raise an exception and return a standard error message.

Note: The following are not scenarios for a specific web service endpoint (method)—they are presented to illustrate the error responses from the authentication process for the user-specified Organization that occurs as part of a request.

Example 3 – Unsuccessful User Organization Scenarios

Scenario	HTTP Response Status Code	Response

Invalid Organization Code and/or Primacy Agency specified for user	401 Unauthorized	<?xml version="1.0" encoding="UTF-8" standalone="yes"?> <serverResponse> <response> <endRow>0</endRow> <errorMessage>Invalid Primacy Agency/Org Code: TestPA/TX9000000 for user: wsadmin</errorMessage> <queueStatus>0</queueStatus> <startRow>0</startRow> <status>401</status> <totalRows>0</totalRows> </response> </serverResponse>

CMDP WEB SERVICE ENDPOINTS (METHODS)

This section details the web service endpoints (methods).

A.1.6 Get User Organizations

A.1.6.1 DESCRIPTION

Retrieves the list of organizations the user is associated with in SCS for CMDP.

A.1.6.2 REQUEST

Pre-Production URL: https://cmdpprep.epa.gov/cmdp-webservice/api/user/userOrganizations
Production URL: https://cmdpservice.epa.gov/cmdp-webservice/api/user/userOrganizations

HTTP Method: GET

Request Header:

```
Authorization: Basic d3NhZG1pbjpwYXNzd29yZA==
```

A.1.6.3 RESPONSE

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<serverResponse>
  <response>
          <data>
                  <userOrganizationRef>
                          <orgCode>TX9000000</orgCode>
                          <orgId>126749</orgId>
                          <orgName>Janak1111</orgName>
                          <orgState>TX</orgState>
                          <orgType>WS</orgType>
                          <username>wsadmin</username>
                  </userOrganizationRef>
                  <userOrganizationRef>
                          <orgCode>TX9000001</orgCode>
                          <orgId>126750</orgId>
```

```
                        <orgName>Janaksaaa</orgName>
                        <orgState>TX</orgState>
                        <orgType>WS</orgType>
                        <username>wsadmin</username>
                    </userOrganizationRef>
            </data>
            <endRow>1</endRow>
            <errorMessage>SUCCESS:null</errorMessage>
            <queueStatus>0</queueStatus>
            <startRow>0</startRow>
            <status>0</status>
            <totalRows>2</totalRows>
  </response>
</serverResponse>
```

A.1.7 Submit Sample Results to CMDP System Using Web Services

A.1.7.1 DESCRIPTION

Submits sample and operational results to CMDP for an organization. The request payload will contain a string representation of the XML data to be submitted.

See the CMDP Web Services Samples Data Dictionary document for details related to the file structure of sample results data. These details include attribute names, data types, field validations, and descriptions. Also, example XML files are available in separate files (opDataExample.xml and SampleDataExample.xml)
Note: XML file generated will fail to load using web services if it is not formatted properly or does not map to the XSD.

A.1.7.2 REQUEST

Pre-Production URL: https://cmdpprep.epa.gov/cmdp-webservice/api/submissions/sampleData
Production URL: https://cmdpservice.epa.gov/cmdp-webservice/api/submissions/sampleData

HTTP Method: POST

Request Header:

```
Authorization: Basic d3NhZG1pbjpwYXNzd29yZA==
orgCode: TX9000001
primacyAgency: PA
```

Request Body (Payload):

```
<?xml version="1.0" encoding="UTF-8"?>
<samples>
  <sample>
            <wsId>CT0010011</wsId>
            <facilityName>DISTRIBUTION SYSTEM - WELL A</facilityName>
            <samplingPointId>MW08-A1</samplingPointId>
            <samplingLocation>slDemo</samplingLocation>
            <sampleCd>AAB1</sampleCd>
```

```
          <collectionDate>2015-09-29</collectionDate>
          <collectionTime>10:00</collectionTime>
          <legalEntityName>PH-0509</        legalEntityName>
          <sampleTypeName>SP</sampleTypeName>
          <sampleVolume>100</sampleVolume>
          <comments>Demo XML Test Microbial</comments>
          <sampleCategoryName>Microbial</sampleCategoryName>
          <sampleResultMicro>
                  <analyteName>3014</analyteName>
                  <methodName>310.3</methodName>
                  <analysisStartDt>2015-07-19</analysisStartDt>
                  <analysisStartTime>12:00:00</analysisStartTime>
                  <analysisComplDt>2015-07-19</analysisComplDt>
                  <analysisComplTime>12:30:00</analysisComplTime>
                  <name>PH-0466</name>
                  <comments>Demo Micro Test</comments>
                  <volumeAssayed>500</volumeAssayed>
                  <apName>P</apName>
                  <count>1234</count>
                  <typeName>Tubes</typeName>
                  <resultVolume>1</resultVolume>
                  <interferenceName>CNFG</interferenceName>
                  <sourceTypeName>Lake</sourceTypeName>
          </sampleResultMicro>
          <sampleResultField>
                  <analyteName>0100</analyteName>
                  <methodName>150.1</methodName>
                  <comments>Demo Micro Field Test</comments>
                  <result>222</result>
                  <uomName>pH</uomName>
          </       sampleResultField>
  </sample>
</samples>
```

A.1.7.3 RESPONSE

Unsuccessful (malformed XML such as missing end tag or failed XSD validation)

HTTP Status 400 Bad Request:

```
<serverResponse>
 <response>
  <data>
   <job>
    <fieldValidationErrors>
     <Error1> Error at LINE: 32, COLUMN: 5 ERROR: The element type
&quot;sourceTypeName&quot; must be terminated by the matching end-tag
&quot;&lt;/sourceTypeName&gt;&quot;.</Error1>
     <Error2> Error at LINE: 32, COLUMN: 5 ERROR: The element type
&quot;sourceTypeName&quot; must be terminated by the matching end-tag
&quot;&lt;/sourceTypeName&gt;&quot;.</Error2>
    </fieldValidationErrors>
    <jobId>0</jobId>
   </job>
  </data>
  <endRow>0</endRow>
  <errorMessage>FAILED_BAD_INPUT_REQUEST:XML validate XSD Failed</errorMessage>
  <queueStatus>1</queueStatus>
  <startRow>0</startRow>
  <status>105</status>
  <totalRows>1</totalRows>
```

```
 </response>
</serverResponse>
```

Successful

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<serverResponse>
  <response>
          <data>
                  <job>
                          <jobId>1117</jobId>
                  </job>
          </data>
          <endRow>0</endRow>
          <errorMessage>SUCCESS:XML Submission Accepted</errorMessage>
          <queueStatus>0</queueStatus>
          <startRow>0</startRow>
          <status>0</status>
          <totalRows>1</totalRows>
  </response>
</serverResponse>
```