



Web Service API for User Security

Introduction

This document describes how to use the MIT Report 1.1 Web Service API to programmatically perform the following actions from outside of the MIT Report user interface:

- Create new MIT Report Users
- Update Existing MIT Report Users

Each of these sections will be presented in the following manner:

1. Description
2. Required Information
3. Request Example
4. Response Example

More information about individual tags can be found under *User Security API Element Definitions* on page 10. A copy of the schema used by this API can be found under the section entitled *User Security API Schema Reference* on page 18.

Posting Method

User Security API requests are sent via HTTP (using the POST method) to the *webservices/userAndGroupSecurity* subdirectory of the MIT Report Server. For example, if MIT Report was installed on the local machine with a port setting of 8080, it would look like this:

<http://localhost:8080/webServices/userAndGroupSecurity>

Authentication Requirements

A MIT Report User account with System Administrator privileges will be needed to use this API. More information on privileges within MIT Report can be found in the *MIT Report Administration Guide*.

Authentication is performed using the base64 encoding provided by *HTTP Basic Authentication*. The following line will need to be included in the header of each request:

Authorizati on: Basic username: password

(where username and password are replaced with their encoded MIT Report User values. For example: bdonis:ilikemits Encoded: YmRvbmlzOmlsaWtlbWl0cw==)

Important Terminology

Report Source ID

This refers to the number MITS Report Server uses to reference a Report Source. See the *MITS Report Web Service API for Report Sources* for more information. In this document, this number is referred to as follows:

sourceId

There are two ways to obtain this number. The first uses the Report Source API. The second method can be accomplished from within MITS Report.

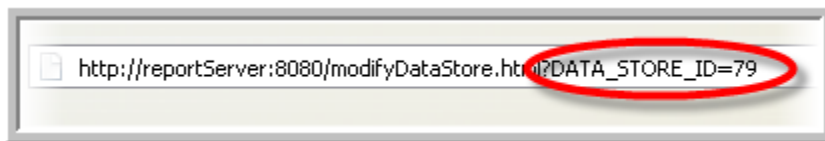
Using the API

To use this method, you must know the absolute path and name of the database file used to create the Report Source. See the section entitled **List Report Sources** in the *Web Service APIs - Report Source API* document for instructions. The Report Source ID will be within the <mitsReportSourceInternalId> tag of the response.

Using the MITS Report Interface

To use this method, you must have **Access** and **Configure** permissions to the appropriate Report Source.

1. From within MITS Report, click the Sources tab.
2. Click the appropriate Source Title
3. The Report Source ID will appear in the browser's address bar, listed as DATA_STORE_ID. In the example below, the Report Source ID is 79.



Group ID

System Groups and Custom Groups within the MIT Report Security are referred to in API calls using their Group ID.

System Groups

System Groups are those User Groups that came prepackaged with MIT Report. Below you will find a list of the System Groups with their associated IDs.

- System Administrator: 1
- Source Administrator: 2
- Source Updater: 3
- Source Viewer: 4
- Library Administrator: 5
- Library Viewer: 6
- Source Operations Creator: 8
- Library Creator: 10
- Source Creator: 11
- Email Sender: 12
- Email Scheduler: 13

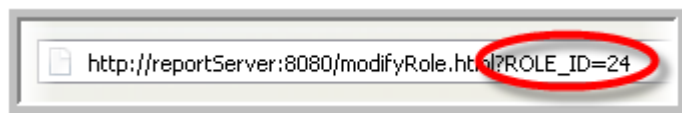
More information on Groups and their permissions can be found in the *MIT Report Administration Guide*.

Custom Groups

Custom Groups are those User Groups created by MIT Report System Administrators. They are assigned an internal ID when they are created.

You must belong MIT Report System Administrator Group to obtain the internal IDs of custom Groups. To find the internal ID of a custom Group:

1. Click on the **Administration** tab
2. Click **Manage Groups** from the menu
3. Click **Modify** next to the appropriate custom Group
4. The custom Group ID will appear in the browser's address bar, listed as `ROLE_ID`. In the example below, the Group ID is 24.



Column ID

MITS Report Server uses a numeric column ID to reference each column within a report.

There are two ways to obtain this number. The first uses the Report Source API. The second method can be accomplished from within MITS Report.

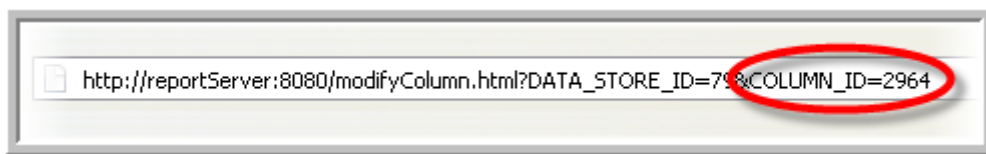
Report Source API method

To use this method, you must know the internal ID of the Report Source associated with the column. (See the document entitled *MITS Report Web Service API for Report Sources*)

MITS Report method

You must have access and configure permissions to the appropriate Report Source to get the internal IDs of columns. To find the internal ID of a column:

1. From within MITS Report, click the **Sources** tab.
2. Click **Configure** under the *Actions* column of the appropriate Report Source
3. Click the appropriate column name under *ID*
4. The column ID will appear in the browser's address bar, listed as COLUMN_ID. In the example below, the Column ID is 2964.



List Users

Description

This will return all MITS Report Users as well as their personal information and associated Groups, as defined on the **Administration** tab of MITS Report.

Required Information

None

Request Example

```
<ListUsers />
```

Response Example

```
<users>
  <user>
    <reportUserId>7</reportUserId>
    <emailAddress>bobd@mits.com</emailAddress>
    <firstName>Bob</firstName>
    <lastName>Donis</lastName>
    <password>*****</password>
    <username>bdonis</username>
    <roles class="List">
      <role>
        <roleId>6</roleId>
        <systemRole>true</systemRole>
        <name>Report Library Reader</name>
      </role>
      <role>
        <roleId>4</roleId>
        <systemRole>true</systemRole>
        <name>Report Source Reader</name>
      </role>
    </roles>
    <enabled>true</enabled>
  </user>
</users>
```

Create a New User

Description

This will create a new MITS Report User.

Required Information

MITS Report login credentials for the User (username and password)

Request Example

```
<userSecurityUpdate>
  <users>
    <user>
      <username>bdonis</username>
      <password>ilikemits</password>
      <firstName>Bob</firstName>
      <lastName>Donis</lastName>
      <emailAddress>bobd@mits.com</emailAddress>
      <roles>
        <roleId>4</roleId>
      </roles>
      <sources>
        <source>
          <sourceId>44</sourceId>
          <sourceRestriction>
            <access>true</access>
            <configuration>false</configuration>
            <updateData>false</updateData>
          </sourceRestriction>
          <columnAccessRestrictions>
            <columnAccessRestriction>1691</columnAccessRestriction>
          </columnAccessRestrictions>
          <columnValueRestrictions>
            <booleanOperator>AND</booleanOperator>
            <columnValueRestriction>
              <booleanOperator>AND</booleanOperator>
              <columnId>1677</columnId>
              <valueRestriction>
                <expression>EQUAL</expression>
                <value>BONI FACE BINGO</value>
              </valueRestriction>
            </columnValueRestriction>
          </columnValueRestrictions>
        </source>
      </sources>
    </user>
  </users>
</userSecurityUpdate>
```

Response Example (Success)

```
<successResponse>  
  <responseTime>2007-06-22 09: 45: 32. 5</responseTime>  
</successResponse>
```

Response Example (Failure)

```
<failureResponse>  
  <failureReason>unknown</failureReason>  
  <errorMessage>j ava. net. SocketException: Read timed out</errorMessage>  
</failureResponse>
```


Modifying a Report User

Description

This will update an existing MITS Report User.

Required Information

MITS Report login credentials for the User (username and password)

Request Example

```
<userSecurityUpdate>
  <users>
    <user>
      <username>bdonis</username>
      <password>ilikemits</password>
      <firstName></firstName>
      <lastName></lastName>
      <emailAddress></emailAddress>
      <roles>
        <roleId>5</roleId>
        <roleId>3</roleId>
      </roles>
      <sources>
        <source>
          <sourceId>44</sourceId>
          <sourceRestriction>
            <access>true</access>
            <configuration>false</configuration>
            <updateData>false</updateData>
          </sourceRestriction>
          <columnAccessRestrictions>
            <columnAccessRestriction>1691</columnAccessRestriction>
          </columnAccessRestrictions>
          <columnValueRestrictions>
            <booleanOperator>AND</booleanOperator>
            <columnValueRestriction>
              <booleanOperator>AND</booleanOperator>
              <columnId>1677</columnId>
              <valueRestriction>
                <expression>EQUAL</expression>
                <value>BONIFACE BINGO</value>
              </valueRestriction>
            </columnValueRestriction>
          </columnValueRestrictions>
        </source>
      </sources>
    </user>
  </users>
</userSecurityUpdate>
```

Response Example (Success)

```
<successResponse>  
  <responseTime>2007-06-22 09: 45: 32. 5</responseTime>  
</successResponse>
```

Response Example (Failure)

```
<failureResponse>  
  <failureReason>unknown</failureReason>  
  <errorMessage>java.net.SocketException: Read timed out</errorMessage>  
</failureResponse>
```

User Security API Element Definitions

access

Definition	Request tag: One of three permissions that can be set to either true or false for each Report Source. Access refers to whether or not a Report Source can be seen by the user. This is a <i>required</i> child of the <sourceRestriction> tag.
Child of	sourceRestriction
Parent of	None
Content Format	Boolean
Example	<access>true</access>

booleanOperator

Definition	Request tag: Links restrictions with AND and OR clauses. AND requires both criteria to be met. OR requires at least one of the criteria to be met. This is a <i>required</i> child of the <columnValueRestrictions> and <columnValueRestriction> tags.
Child of	columnValueRestriction, columnValueRestrictions
Parent of	None
Content Format	Boolean
Example	<access>true</access>

columnAccessRestriction

Definition	Request tag: Contains the column id number of the column to be restricted from the User's view. See the Internal Ids section for more information. This is a <i>required</i> child of the <columnAccessRestrictions> tag.
Child of	columnAccessRestrictions
Parent of	None
Content Format	Integer
Example	<columnAccessRestriction>1675</columnAccessRestriction>

columnAccessRestrictions

Definition	Request tag: Container for one or more <columnAccessRestriction> tags. This <i>requires</i> a <columnAccessRestriction> child tag.
Child of	source
Parent of	columnAccessRestriction
Content Format	Container
Example	<columnAccessRestrictions>

columnId

Definition	Request tag: Contains the column id number of the column whose values are to be restricted. More information can be found in the Internal Ids section, below. This is a <i>*required*</i> child of the <columnValueRestriction> tag.
Child of	columnValueRestriction
Parent of	None
Content Format	Integer
Example	<columnId>1677</columnId>

columnValueRestriction

Definition	Request tag: Contains the column and row information to be restricted. This is a <i>*required*</i> child of the <columnValueRestrictions> tag. This <i>*requires*</i> <columnId>, <booleanOperator>, and <valueRestriction> child tags.
Child of	columnValueRestrictions
Parent of	columnId, valueRestriction
Content Format	Container
Example	<columnValueRestriction>

columnValueRestrictions

Definition	Request tag: Container for one or more <columnValueRestriction> tags. This <i>*requires*</i> <columnValueRestriction> and <booleanOperator> child tags.
Child of	source
Parent of	columnValueRestriction
Content Format	Container
Example	<columnValueRestrictions>

configuration

Definition	Request tag: One of three permissions that can be set to either true or false for each Report Source. Configure refers to whether or not a Report Source can be modified by the user. This is a <i>*required*</i> child of the <sourceRestriction> tag.
Child of	sourceRestriction
Parent of	None
Content Format	Boolean
Example	<configuration>false</configuration>

emailAddress

Definition	Request tag: Email address of the User. This is a *required* child of the <user> tag. This tag can be left blank.
Child of	user
Parent of	None
Content Format	String
Example	<emailAddress>philc@mits.com</emailAddress>

enabled

Definition	Response tag: The status of the User. The contained value will be either true or false.
Child of	user
Parent of	None
Content Format	Boolean
Example	<enabled>true</enabled>

errorMessage

Definition	Response tag: Message containing what the error is and where it occurred.
Child of	failureResponse
Parent of	None
Content Format	String
Example	<errorMessage>java.net.SocketTimeoutException: Read timed out</errorMessage>

expression

Definition	Request tag: Together with the <columnId> tag and the <value> tag, determines which fields are visible in a report. This is a *required* child of the <valueRestriction> tag.
Child of	valueRestriction
Parent of	None
Content Format	String
Example	<expression>EQUAL</expression>

failureReason

Definition	Response tag: Message containing why the error occurred. Possible values: MitsReportSourceNotFound, ColumnNotFound, RoleNotFound, UserAlreadyExists, UserNotFound, MismatchedValueRestrictionType, and Unknown.
Child of	failureResponse
Parent of	None
Content Format	String
Example	<failureReason>unknown</failureReason>

failureResponse

Definition	Response tag: Signifies a failed request attempt.
Child of	None
Parent of	failureReason, errorMessage
Content Format	Container
Example	<failureResponse>

firstName

Definition	Both a Request tag and a Response tag: The first name of the User. This is a <i>*required*</i> child of the <user> tag. This tag can be left blank.
Child of	user
Parent of	None
Content Format	String
Example	<firstName>Phil</firstName>

lastName

Definition	Both a Request tag and a Response tag: The last name of the User. This is a <i>*required*</i> child of the <user> tag. This tag can be left blank.
Child of	user
Parent of	None
Content Format	String
Example	<lastName>Cardigan</lastName>

listUsers

Definition	Request tag: Returns a list of all Users.
Child of	None. This is the Root element.
Parent of	None
Content Format	No content
Example	<listUsers />

name

Definition	Response tag: The Group's name.
Child of	role
Parent of	None
Content Format	String
Example	<name>Report Source Updater</name>

password

Definition	Both a Request tag and a Response tag: The password for the User's login. As a Request tag, this is a <i>*required*</i> child of the <user> tag. As a Response tag, this value will be a series of '*' characters
Child of	user
Parent of	None
Content Format	String
Example	<password>W87de42</password>

responseTime

Definition	Response tag: When updating a Report Source, this tag contains the date and time that a Report Source update initialized. It is given in the following format: YYYY-MM-DD HH: MM: SS
Child of	None
Parent of	mitsReportSourceInternalId
Content Format	Date and Time
Example	<responseTime>2007-06-25 09: 45: 32</responseTime>

role

Definition	Response tag: Contains Group information.
Child of	roles
Parent of	roleID, systemRole, name
Content Format	Container
Example	<role>

roleid

Definition	Request tag: A number used by MIT S Report Server to reference the User's Groups internally. More information can be found in the Internal Ids section, below. This is a <i>*required*</i> child of the <roles> tag.
Child of	roles
Parent of	None
Content Format	Integer
Example	<roleid>7</roleid>

roleID

Definition	Response tag: A number used by MIT S Report Server to reference the User's Groups internally. More information can be found in the Internal Ids section, below.
Child of	role
Parent of	None
Content Format	Integer
Example	<roleID>7</roleID>

roles

Definition	Both a Request tag and a Response tag: As a Request tag, it <i>*requires*</i> one or more <roleId> child tags. As a Response tag, it contains one or more <role> child tags.
Child of	user
Parent of	roleId
Content Format	Container
Example	<roles>

source

Definition	Request tag: Contains the internal id of a Report Source with all associated restrictions. This is a <i>*required*</i> child of the <sources> tag. This <i>*requires*</i> a <sourceId> child.
Child of	sources
Parent of	sourceId, sourceRestriction, columnAccessRestrictions, columnValueRestrictions
Content Format	Container
Example	<source>

sourceId

Definition	Request tag: A number used by MIT Report Server to reference the Report Sources internally. More information can be found in the Internal Ids section, below. This is a <i>*required*</i> child of the <source> tag.
Child of	source
Parent of	None
Content Format	Integer
Example	<sourceId>42</sourceId>

sourceRestriction

Definition	Request tag: Contains the permissions associated with a source. This <i>*requires*</i> <access>, <configuration>, and <updateData> child tags.
Child of	source
Parent of	access, configure, updateData
Content Format	Container
Example	<sourceRestriction>

sources

Definition	Request tag: Contains one or more <source> tags. This <i>*requires*</i> a <source> child.
Child of	user
Parent of	source
Content Format	Container
Example	<sources>

successResponse

Definition	Response tag: Signifies a request was successful.
Child of	None
Parent of	?
Content Format	Container
Example	<successResponse>

systemRole

Definition	Response tag: Determines whether a Group is a System Group or a custom Group.
Child of	role
Parent of	None
Content Format	Boolean
Example	<systemRole>true</SystemRole>

updateData

Definition	Request tag: One of three permissions that can be set to either true or false for each Report Source. UpdateData refers to whether or not a Report Source can be updated by the user. This is a <i>*required*</i> child of the <sourceRestriction> tag.
Child of	sourceRestriction
Parent of	None
Content Format	Boolean
Example	<updateData>false</updateData>

user

Definition	Request tag: Contains a collection of information, permissions, and restrictions for a User. This is a <i>*required*</i> child of the <users> tag. This <i>*requires*</i> <username>, <password>, <firstName>, <lastName>, and <emailAddress> child tags.
Child of	users
Parent of	username, password, firstName, lastname, emailAddress, roles, sources (also a parent of xx as a response tag
Content Format	Container
Example	<user>

username

Definition	Request tag: The username for the User's login. This is a <i>*required*</i> child of the <user> tag.
Child of	user
Parent of	None
Content Format	String
Example	<username>Phil</username>

users

Definition	Request tag: Contains one or more <user> tags. This <i>*requires*</i> a <user> child.
Child of	userSecurityUpdate
Parent of	user
Content Format	Container
Example	<users>

userSecurityUpdate

Definition	Request tag: Root Request tag: Posted to create new and update existing Users.
Child of	None: This is the Root Element
Parent of	users
Content Format	Container
Example	<userSecurityUpdate>

value

Definition	Request tag: Together with the <columnId> tag and the <expression> tag, determines which fields are visible in a report. This is a <i>*required*</i> child of the <columnValueRestriction> tag.
Child of	valueRestriction
Parent of	None
Content Format	String
Example	<value>BONIFACE BINGO</value>

valueRestriction

Definition	Request tag: Container for an <expression> and <value> tag pair. This is a <i>*required*</i> child of the <columnValueRestriction> tag.
Child of	columnValueRestriction
Parent of	expression, value
Content Format	Container
Example	<valueRestriction>

User Security API Schema Reference

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
  XML Schema for User Security API Calls
-->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!--Requests-->
  <!--
    Request xml for updating/creating users and permissions
  -->
  <xs:element name="userSecurityUpdate">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="users">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="user" type="userType" minOccurs="1" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!--
    Request xml to retrieve a list of Report Users (without the permissions)
  -->
  <xs:element name="listUsers" type="xs:element" />

  <!--Response Types-->
  <!--
    Success Response for an update
  -->
  <xs:element name="successResponse" type="xs:string" />

  <!--
    Failure response for an update
  -->
  <xs:element name="failureResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="failureReason" type="failureReasonEnumType"/>
        <xs:element name="errorMessage" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!--Referenced Types-->
  <xs:complexType name="userType">
    <xs:sequence>
      <xs:element name="roles" minOccurs="0" maxOccurs="1">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="role" type="xs:integer" minOccurs="1" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="sources" minOccurs="0" maxOccurs="1">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="source" type="sourceType" minOccurs="1" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="username" type="xs:string" minOccurs="1" maxOccurs="1" />
      <xs:element name="password" type="xs:string" minOccurs="1" maxOccurs="1" />
      <xs:element name="firstName" type="xs:string" minOccurs="1" maxOccurs="1" />
      <xs:element name="lastName" type="xs:string" minOccurs="1" maxOccurs="1" />
      <xs:element name="emailAddress" type="xs:string" minOccurs="1" maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="sourceType">
    <xs:sequence>
      <xs:element name="sourceRestriction" minOccurs="0" maxOccurs="1">

```

```

<xs:complexType>
  <xs:sequence>
    <xs:element name="access" type="xs:boolean" minOccurs="1" maxOccurs="1" />
    <xs:element name="configure" type="xs:boolean" minOccurs="1" maxOccurs="1" />
    <xs:element name="updateData" type="xs:boolean" minOccurs="1" maxOccurs="1" />
  </xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="columnAccessRestrictions" minOccurs="0" maxOccurs="1">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="columnAccessRestriction" minOccurs="1" maxOccurs="unbounded"
type="xs:integer" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="columnValueRestrictions" minOccurs="0" maxOccurs="1">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="columnValueRestriction" type="columnValueRestrictionType" minOccurs="1"
maxOccurs="unbounded" />
      <xs:element name="booleanOperator" type="booleanOperatorEnumType" minOccurs="1"
maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

  <xs:element name="sourceId" type="xs:integer" minOccurs="1" maxOccurs="1" />
</xs:sequence>
</xs:complexType>

<xs:complexType name="columnValueRestrictionType">
  <xs:sequence>
    <xs:element name="valueRestriction" minOccurs="1" maxOccurs="unbounded">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="expression" type="expressionEnumType" minOccurs="1" maxOccurs="1" />
          <xs:element name="value" type="xs:string" minOccurs="1" maxOccurs="1" />
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="booleanOperator" type="booleanOperatorEnumType" minOccurs="1" maxOccurs="1" />
    <xs:element name="columnId" type="xs:integer" minOccurs="1" maxOccurs="1" />
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="booleanOperatorEnumType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="AND" />
    <xs:enumeration value="OR" />
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="expressionEnumType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="EQUAL" />
    <xs:enumeration value="NOT_EQUAL" />
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="failureReasonEnumType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="MitsReportSourceNotFound" />
    <xs:enumeration value="ColumnNotFound" />
    <xs:enumeration value="RoleNotFound" />

    <!-- Occurs when adding a user that already exists -->
    <xs:enumeration value="UserAlreadyExists" />

    <!-- Occurs when updating a user that does not already exist -->
    <xs:enumeration value="UserNotFound" />

    <!-- Occurs when adding a Text value restriction to a Number Column -->
    <xs:enumeration value="MismatchedValueRestrictionType" />
    <xs:enumeration value="Unknown" />
  </xs:restriction>
</xs:simpleType>
</xs:schema>

```

Index

A

API

- authentication 1
- element definitions 10
- posting method 1
- request 1
- schema 18

authentication 1

C

Column ID

- definition 4
- identifying, using the API 4
- identifying, using the interface 4

create user 6

custom group

- definition 3
- ID 3

G

Group ID

- definition 3
- identifying, custom 3
- identifying, system 3

H

HTTP

- authentication 1
- posting method 1

I

identify

- column 4
- group, custom 3
- group, system 3
- Report Source, using the API 2
- Report Source, using the interface 2

internal ID

- column 4
- group, custom 3
- group, system 3
- Report Source 2

L

list users 5

M

modify user 8

P

posting

- authentication 1
- method 1

R

Report Source ID

- definition 2
- identifying, using the API 2
- identifying, using the interface 2

request

- authentication, for post 1
- posting method 1

S

schema

- user security 18

system group

- definition 3
- ID 3

T

tag definitions 10

tags

- access 10
- booleanOperator 10
- columnAccessRestriction 10
- columnAccessRestrictions 10
- columnId 4, 11
- columnValueRestriction 11
- columnValueRestrictions 11
- configuration 11
- emailAddress 12
- enabled 12
- errorMessage 12
- expression 12
- failureReason 12
- failureResponse 13
- firstName 13
- lastName 13
- listUsers 5, 13
- mitsReportSourceInternalId 2
- name 13
- password 14
- responseTime 14
- role 14
- roleId 14
- roleId 3, 14
- roles 15
- source 15
- sourceId 2, 15
- sourceRestriction 15
- sources 15
- successResponse 16
- systemRole 16
- updateData 16
- user 16
- username 16
- users 17
- userSecurityUpdate 6, 8, 17
- value 17
- valueRestriction 17

U

user

- create new 6
- list 5
- modify 8

user security API schema 18

X

XML schema 18