

EMC[®] Unisphere[™] for VMAX[®]

Version 8.4.0

REST API Concepts and Programmer's Guide

Copyright © 2014-2017 EMC Corporation All rights reserved.

Published June 7, 2017

Dell believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS-IS." DELL MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. USE, COPYING, AND DISTRIBUTION OF ANY DELL SOFTWARE DESCRIBED IN THIS PUBLICATION REQUIRES AN APPLICABLE SOFTWARE LICENSE.

Dell, EMC, and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be the property of their respective owners.
Published in the USA.

EMC Corporation
Hopkinton, Massachusetts 01748-9103
1-508-435-1000 In North America 1-866-464-7381
www.EMC.com

CONTENTS

Tables		5
	PREFACE	7
Chapter 1	Introduction	9
	Product overview.....	10
	REST model.....	10
	Unisphere for VMAX REST Client.....	10
	New in this release.....	11
	Additional resources.....	11
Chapter 2	Documentation	13
	Downloading documentation.....	14
	Transport protocol.....	14
	Product version and compatibility.....	14
	Backward compatibility.....	14
	Client/Server compatibility.....	15
	Making the JSON root element backwards compatible.....	15
Chapter 3	Iterators	17
	How iterators are used in REST API.....	18
	Sample iterator methods and typical use case.....	18
Chapter 4	Sample	19
	Setting up user authentication.....	20
	Client authentication.....	20
	Sample client request and response examples.....	21

CONTENTS

TABLES

1 REST resource methods..... 10

PREFACE

As part of an effort to improve its product lines, EMC periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your EMC technical support professional if a product does not function properly or does not function as described in this document.

Note

This document was accurate at publication time. Go to EMC Online Support (<https://support.emc.com>) to ensure that you are using the latest version of this document.

Purpose

This document describes how to configure and use EMC Unisphere for VMAX REST API.

Audience

This guide is intended for programmers who want to create interfaces, using these API's, in any of the programming environments that support standard REST clients such as web browsers and programming platforms that can issue HTTP requests.

Related documentation

The following EMC publications provide additional information:

- EMC Unisphere for VMAX Release Notes
- EMC Unisphere for VMAX Installation Guide

Special notice conventions used in this document

EMC uses the following conventions for special notices:

DANGER

Indicates a hazardous situation which, if not avoided, will result in death or serious injury.

WARNING

Indicates a hazardous situation which, if not avoided, could result in death or serious injury.

CAUTION

Indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.

NOTICE

Addresses practices not related to personal injury.

Note

Presents information that is important, but not hazard-related.

Typographical conventions

EMC uses the following type style conventions in this document:

Bold	Used for names of interface elements, such as names of windows, dialog boxes, buttons, fields, tab names, key names, and menu paths (what the user specifically selects or clicks)
<i>Italic</i>	Used for full titles of publications referenced in text
Monospace	Used for: <ul style="list-style-type: none"> • System code • System output, such as an error message or script • Pathnames, filenames, prompts, and syntax • Commands and options
<i>Monospace italic</i>	Used for variables
Monospace bold	Used for user input
[]	Square brackets enclose optional values
	Vertical bar indicates alternate selections - the bar means “or”
{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate nonessential information omitted from the example

Where to get help

EMC support, product, and licensing information can be obtained as follows:

Product information

For documentation, release notes, software updates, or information about EMC products, go to EMC Online Support at <https://support.emc.com>.

Technical support

Go to EMC Online Support and click Service Center. You will see several options for contacting EMC Technical Support. Note that to open a service request, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

Your comments

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to VMAXContentFeedback@emc.com.

CHAPTER 1

Introduction

This chapter contains the following:

- [Product overview](#) 10
- [REST model](#) 10
- [Unisphere for VMAX REST Client](#) 10
- [New in this release](#) 11
- [Additional resources](#) 11

Product overview

The EMC Unisphere for VMAX REST (Representational State Transfer) API allows for accessing diagnostic performance data, accessing configuration data, and performing storage provisioning operations for VMAX hardware through robust APIs. These API's can be used in any of the programming environments that support standard REST clients such as web browsers and programming platforms that can issue HTTP requests. The Unisphere for VMAX REST API supports both XML and JavaScript Object Notation (JSON) MIME types.

REST model

REST is a manner of building networked software systems that allows merging of documents, data and information services into an ecosystem of named resources.

- Resource - Anything important enough to be referenced as a noun.
- Resource name - Unique identification of the resource.
- Resource representation - Useful information about the current state of resource. Resource representation is what is used to transfer its information to the client using the HTTP protocol.
- Resource link - Link to another representation of the same resource.
- Resource interface - Uniform interface for accessing the resource and manipulating its state.

The following table describes the standard resource methods.

Table 1 REST resource methods

HTTP operation	Description	Access to Method
PUT	Modify resource	Administrator Storage Administrator
GET	Retrieve state of resource	All roles
POST	Create a new resource	Administrator Performance Monitor (for Performance URLs only) Storage Administrator
DELETE	Delete resource	Administrator Storage Administrator

For assigning user roles for each array, refer to Setting up user authentication.

Unisphere for VMAX REST Client

The Unisphere for VMAX REST Client provides both a front-end GUI for the REST API documents and a Unisphere for VMAX-specific REST client based on the WADL file received from the Unisphere for VMAX Host.

For more information, reference the Unisphere for VMAX REST Client available on the VMAX Community on ECN: <https://community.emc.com/community/products/vmax>

New in this release

Unisphere for VMAX REST API 8.4 has the following changes:

- All non-Performance and non-Common REST calls are now available under /84/ endpoints. All Performance calls are now available under the root endpoint (no version in the URL).
- All POST and PUT REST calls can now be made asynchronously in Replication, Migration, SLO Provisioning and VVol. Users can specify whether they want a synchronous or asynchronous execution option. The default is synchronous.
- New GET pattern and responses:
 - Response code should be used for checking status
 - Non Unique Values for an object will be optional
 - Non Unique Values will no longer be returned in response NA, N/A, “”, “”, -1
- Return codes are standardized across all REST calls:
 - 200 — Success, 201 Successful and Resource object Created, 202 Accepted (asynchronous)
 - 401 — Incorrect Username or Password
 - 403 — User not Authorized to make the call
 - 404 — Object not found
 - 409 — Object already exists
 - 500 — Server Side error Check SMAS logs, services etc

Additional resources

This section lists the resources available for the Unisphere for VMAX REST API.

- Information on how to get started with Unisphere for VMAX REST API: <https://community.emc.com/docs/DOC-56447>
- Change log — This document lists the changes to the Unisphere for VMAX REST API in each release. It can be found on the EMC VMAX Community page under "Developer Resources": <https://community.emc.com/community/products/vmax#developer>
- Enunciate — This is the engine used to generate the Unisphere for VMAX REST API. More information can be found here: <http://enunciate.webcohesion.com/>
- EMC Open Source Community: <https://emccode.github.io>
- Unisphere for VMAX-specific examples: <https://github.com/ciarams87/PyU4V>

CHAPTER 2

Documentation

This chapter contains the following:

- [Downloading documentation](#).....14
- [Transport protocol](#).....14
- [Product version and compatibility](#).....14
- [Making the JSON root element backwards compatible](#).....15

Downloading documentation

Clients can download the UNIVMAX REST API documentation (URLs, required parameters, and returned results) as follows:

Before you begin

Prior to downloading REST API documentation, Unisphere for VMAX must be installed and user access roles must be setup (see [Setting up user authentication](#) on page 20).

Procedure

1. Point the browser to: `https://{UNIVMAX_IP}:{UNIVMAX_PORT}/univmax/restapi/docs`

Where, *UNIVMAX_IP* is the IP address and *UNIVMAX_PORT* is the port of the host running Unisphere for VMAX.

2. Copy the zip file (restapi-docs.zip) locally, unzip the file, and navigate to `target/docs/index.html`.
3. Open the html file to access the documented resources.

Transport protocol

REST API uses the Hypertext Transfer Protocol Secure (HTTPS) version 1.1 as the transport for API requests. For PUT and POST requests, method arguments are passed in the HTTP Request message body. API methods return standard HTTP status codes and result content in the HTTP Response message body.

Unisphere for VMAX REST API supports two media-types:

- `application/xml` — Allows marshalling/un-marshalling using XML
- `application/json` — Allows marshalling/ un-marshalling using JSON

Product version and compatibility

The Unisphere REST API is the same version as the Unisphere for VMAX application.

Backward compatibility

Since the REST API is a method-based API, the methods will remain backwards compatible with older versions as long as the argument and result types have only optional properties added to newer versions of the API.

If additional properties are required or existing properties require modification, an additional method should be created. Deprecated methods will be maintained to service older clients, and eventually removed. New REST API methods are backwards compatible.

REST API guarantees backwards compatibility for two major releases. Versioning is handled in two different ways:

- For example, server REST API version 8.0.x will be available in versions 8.1.x and 8.2.x, however in 8.3.x and all future versions any 8.0.x requests will be re-directed where possible to oldest available version, so in case of 8.3.x this will be 8.1.x, and so forth. This applies to:

- DSA
- Migration
- WLP
- VVol
- System
- Provisioning
- Replication
- Migration
- For example, if the server REST API version is 8.0.x, then it is backward compatible with REST API version 7.6.x or 7.5.x on the client. This applies to:
 - Management
 - Common
 - Performance

Client/Server compatibility

If the Unisphere for VMAX REST Client API version is older than the server version, interaction proceeds as designed. If the client REST API version is newer than the server version, the server delivers an HTTP status 400 or HTTP status 404, indicating an illegal request.

Making the JSON root element backwards compatible

For prior versions of the Unisphere for VMAX REST API, the JSON root is included in the API request and response. The current version of Unisphere for VMAX REST API does not include the JSON root in the API request and response. To make the server backwards compatible, the property that controls this must be modified as follows:

Procedure

1. Stop the SMAS service.
2. Remove `<location of Unisphere installation>\jboss\domain\deploy0` directory (and `deploy1`, `deploy2` etc if they exist).

Note

Do not remove `deploy` directory.

3. Locate `<location of Unisphere installation>\jboss\domain\deploy\smc.ear\smc.war\WEB-INF\web.xml`.
4. Open the file and set the property `com.emc.em.restapi.WrapJsonRoot` to `"true"`.
5. Start SMAS.

Example 1 REST API request and response example

The following example shows a REST API request and response with and without the JSON root:

Example 1 REST API request and response example (continued)**Request:**

```
https://<UnimaxIP:UnimaxPort>/univmax/restapi/provisioning/
symmetrix/000195900063/storagegroup
```

Where *UNIVMAX_IP* is the IP address and *UNIVMAX_PORT* is the port of the host running Unisphere for VMAX.

Response without JSON root (default):

```
{
  "num_of_storage_groups": 64,
  "storageGroupId":
  [
    "appEng",
    "appEng2",
  ],
  "success": true
}
```

Response with JSON root:

```
{
  "listStorageGroupResult":
  {
    "num_of_storage_groups": 64,
    "storageGroupId":
    [
      "appEng",
      "appEng2",
    ],
    "success": true
  }
}
```


CHAPTER 3

Iterators

This chapter contains the following:

- [How iterators are used in REST API](#).....18
- [Sample iterator methods and typical use case](#)..... 18

How iterators are used in REST API

An iterator is an object that allows you to cycle through the elements in a collection and display each element. Unisphere for VMAX REST API uses iterators to retrieve a collection of performance data only when the API interface defines that there are unbounded instances in the result. For example, the method to retrieve array metrics takes an `ArrayParamType` argument and returns an unbounded number of `ArrayResultType` results.

See Downloading documentation on page 14 for instructions on how to access the REST resources. From the EMC Unisphere for VMAX RESTAPI Home page, select Performance, then any `/metrics` resource, and under Response Body select Iterator to view how the iterator cycles through the elements in a collection results.

The Unisphere for VMAX REST API also uses iterators to control resource consumption on the server when multiple clients are querying for large sets of data. Each iterator has a unique ID and provides two ways to control excess server consumption: maximum page size, and an expiration time for the iterator instance. The page size and expiration time are dependent on server load and resource availability. In addition, Unisphere for VMAX imposes limits on the number of concurrent requests it will accept. See Sample iterator methods and typical use case.

Sample iterator methods and typical use case

The infrastructure provides three method endpoints for clients to interact with the iterators.

- Get iterator info
- Get iterator page
- Delete iterator

A typical use case is as follows:

Note

If a client becomes unresponsive, or disconnects from the Unisphere for VMAX server, the iterator expires, based on a pre-configured timeout. When the iterator expires, server resources are released. If the iterator expires before the client has finished paging over all the results, the client must reissue the request.

1. The client invokes a REST API method that returns an iterator.
2. The client pages over the results of an existing iterator until all results are retrieved by issuing `Get common /iterator/{iteratorId}/page`.
3. If the iterator has not expired, the client deletes the iterator by issuing `Delete common/iterator/{iteratorId}`.

CHAPTER 4

Sample

This chapter contains the following:

- [Setting up user authentication](#)..... 20
- [Client authentication](#)..... 20
- [Sample client request and response examples](#)..... 21

Setting up user authentication

Prior to using the Unisphere for VMAX REST API, user authorization must be assigned for each VMAX array the user is permitted to access.

Procedure

1. Log into Unisphere for VMAX.
2. From the system selector, select **All Symmetrix**.
3. Select **Home > Administration > Security > Authorized Users and Groups** to open the **Authorized Users and Groups** list view.
4. Click **Add** to open the **Add Authorization Rule** dialog box.
5. Select the **Roles** tab and create a user login profile for each VMAX array to be accessed by the user, and assign them to any of the following roles:
 - Users that are allowed VMAX access:
 - Administrator — Can initiate GET, POST, PUT, and DELETE methods.
 - Storage Administrator — Can initiate GET, POST, PUT, and DELETE methods.
 - Performance Monitor — Can initiate GET and POST (for Performance URLs only) methods.
 - Monitor — Can initiate GET methods.
 - Auditor — Can initiate GET methods.
 - Security Administrator — Can initiate GET methods.
 - Users that are not allowed VMAX access:
 - None
6. Click **OK**.

Client authentication

Note

The minimum supported TLS version is 1.2.

REST API uses HTTP Basic Access Authentication to authenticate API clients, described in RFC 2617. HTTP Basic Access Authentication is simple, since it does not require cookies, session handling, or login pages. Instead, HTTP Basic Access Authentication uses static headers, requiring no handshakes.

Users of the REST API are assigned user credentials for associated VMAX arrays within Unisphere for VMAX. A username and password is supplied with every request to REST API in the “Authorization” header, as specified in RFC 2617. Every request to REST API is authenticated and authorized.

Note

If X509 certificate-based client authentication is enabled, the username/password REST basic authentication credential is ignored by Unisphere. However, to meet REST standard's basic authentication requirement, REST clients must provide a non-empty username/password credential of their own creation.

Sample client request and response examples

For sample client request and response examples, refer to the REST API documentation. [Downloading documentation](#) on page 14 describes how to obtain the documentation.

Sample