**USER DOCUMENTATION**

**# About Swordfish API**

* The Swordfish emulator allows user to create, read, delete and update Rest operations .
* This Swordfish is an extension of Redfish emulator (“**DMTF REDFISH EMULATOR**”).
* Users have to follow some installation steps which are declared in installation documentation.

**# Use of Swordfish**

**Restfull Operations:**

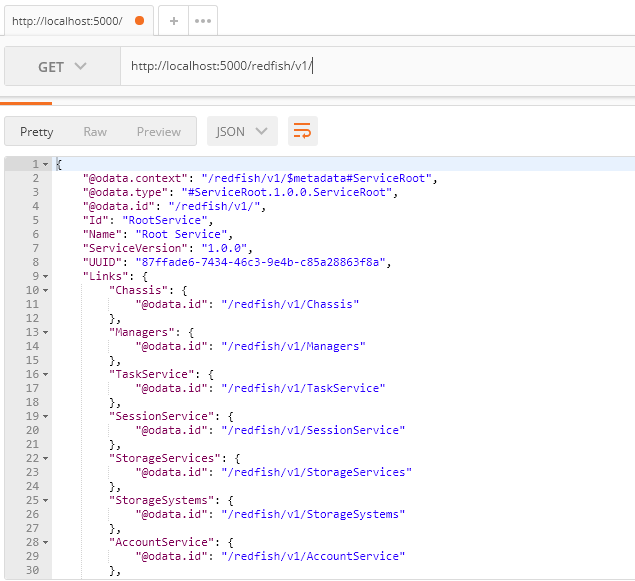
Swordfish supports restful commands( GET, POST,PUT and DELETE) . To perform these operations we can use Postman or any other add-on.(which is chrome extension)

**Postman:**

Basically postman is supported In chrome browser. User has to create an account in postman and use it for testing rest-api.

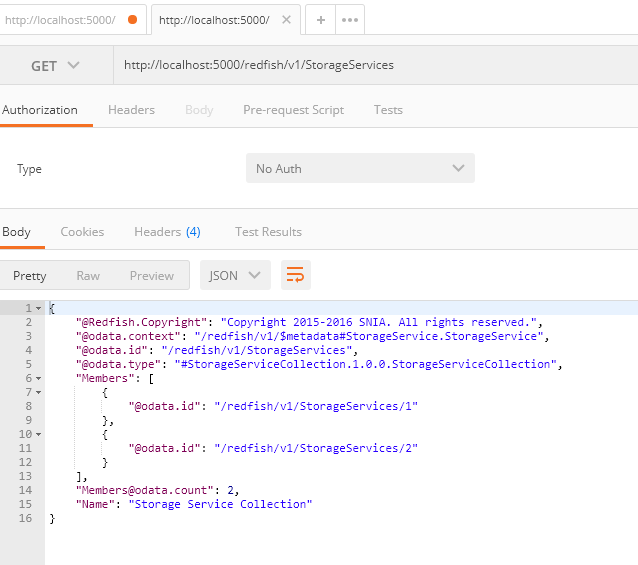
Postman is used for all rest operations. Below are the figures with description.

1. Basic GET operation for Service root



Below screen is used to GET “Storage Services Collection data. Here we can see Members and members @odata.count.

**Command**: **GET** http://localhost:5000/redfish/v1/StorageServices



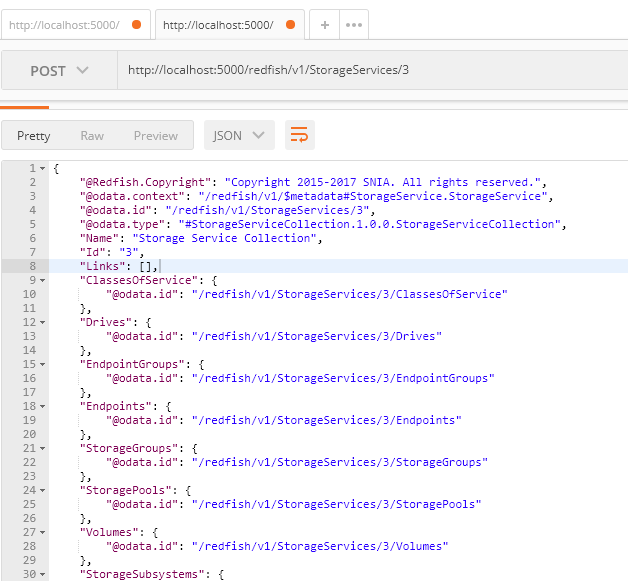
User can add new member of a collection data by using” post” command. It will create a new instance of resource and subresources.

1. Adding New element /member to collection using “POST”

**Command**: **POST** <http://localhost:5000/redfish/v1/StorageServices/3>

The below result shows after adding “3” to members of Storageservices.

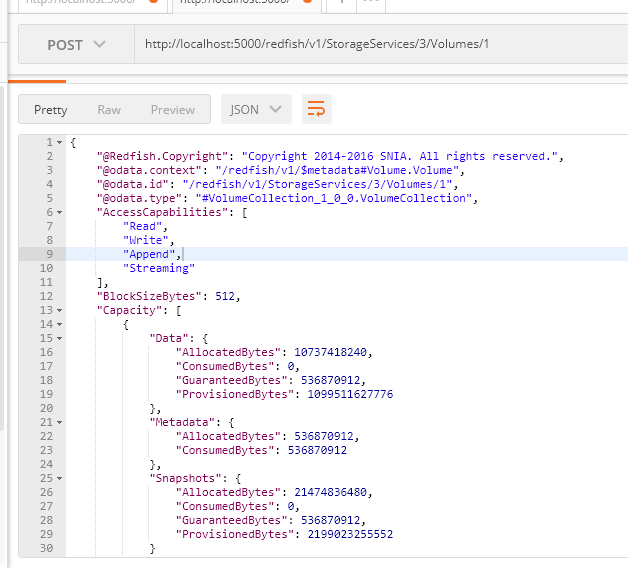
Check It By **GET** http://localhost:5000/redfish/v1/StorageServices/3



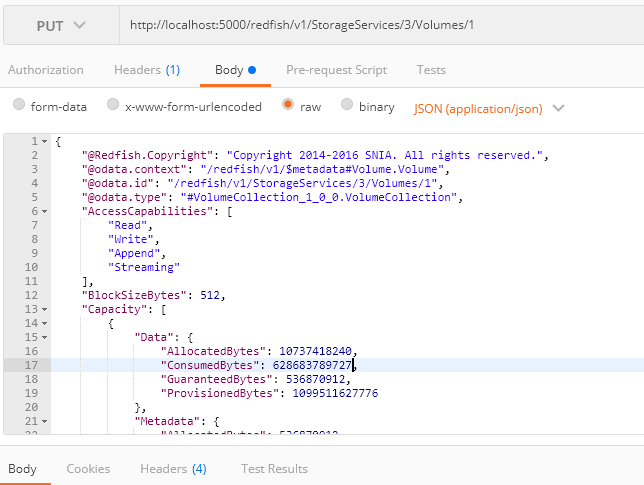
1. Adding a member of Volume using “Post”

user creating new child collection i.e. StorageServices/3/Volumes/1

**Command**: **POST** http://localhost:5000/redfish/v1/StorageServices/3/Volumes/1



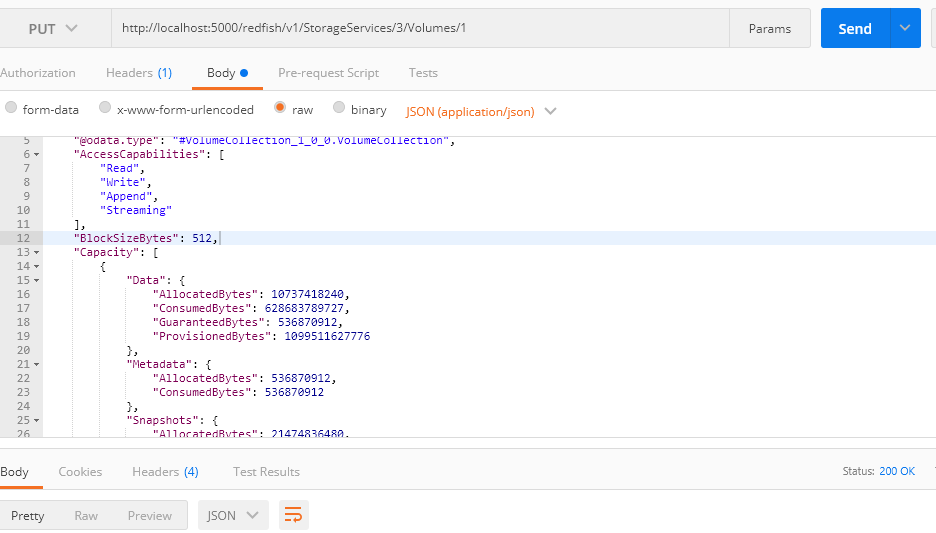
1. Edit all the properties of an element in a collection using “PUT”



User have to pass complete json file to the body and modify the content in volumes /1

User need to declare Json(application/json) before sending the request.

Below is the updated volume as shown in figure.



1. Deleting an element of member Collection in “Volumes”.

User can’t delete more than one element in a single request .

