

**VSM REST API Developer Guide**

Version 0.2

Nov., 2015

This document describes the REST API exposed by VSM, includes the input/output form, possible scenarios to apply, and current status, etc.

Contents

[1 Introduction 4](#_Toc436810084)

[1.1 System Requirements 5](#_Toc436810085)

[1.2 What’s in the Rest of This Guide 6](#_Toc436810086)

[2 REST API Definition 7](#_Toc436810087)

[2.1 General guidelines 7](#_Toc436810088)

[2.2 API Definitions 7](#_Toc436810089)

[2.2.1 Appnode 7](#_Toc436810090)

[2.2.2 cluster 11](#_Toc436810091)

[2.2.3 device 16](#_Toc436810092)

[2.2.4 mds 20](#_Toc436810093)

[2.2.5 monitor 22](#_Toc436810094)

[2.2.6 osd 25](#_Toc436810095)

[2.2.7 performance\_metric 32](#_Toc436810096)

[2.2.8 placement\_group 33](#_Toc436810097)

[2.2.9 pool\_usage 35](#_Toc436810098)

[2.2.10 rbd\_pool 37](#_Toc436810099)

[2.2.11 server 38](#_Toc436810100)

[2.2.12 storage\_group 45](#_Toc436810101)

[2.2.13 storage\_pool 49](#_Toc436810102)

[2.2.14 vsm\_setting 58](#_Toc436810103)

[2.2.15 zone 60](#_Toc436810104)

[3 Example 63](#_Toc436810105)

**Revision History**

| **Revision** | **Date** | **Description** |
| --- | --- | --- |
| 0.1 | Oct. 15, 2015 | Initial version |
| 0.2 | Nov. 03, 2015 | Change according to initial review, format changed. |

# Introduction

VSM (Virtual Storage Manager) is an open source Ceph management tool, it targets to lower the barrier to adopt Ceph in production environment.

VSM includes a few modules inside, there are

* **Python-vsmclient**
  + This is a client for the vsm API, it consists of
  + a Python API (the vsmclient module),
  + a command-line script (vsm). Each implements 100% of the vsm API.
* **Vsm**
  + A major module for ceph management
* **Vsm-dashboard**
  + web based management interface for VSM.
* **Vsm-deploy**
  + The ceph deployment tool kit provided by VSM .

VSM opens the possibility for third-party integration by exposing a set of REST API in python-vsmclient modules. Below diagram depicts the basic architecture of VSM and the relationships between modules.

Vsmclient

API

conductor

scheduler



Dashboard

**storage**

agent

**storage**

agent

**storage**

agent

VSM consists of two key Roles:

* Controller (also referred to as VSM Controller):
  + Runs on dedicated server (or server instance)
  + Connects to Ceph cluster through VSM agent
  + Connects to OpenStack Nova controller (optional) via SSH
* Agent (also referred to as VSM node):
* Runs on every server in the Ceph cluster
* Accepts VSM controller’s commands and execute locally
* Relays server configuration & status information to VSM controller

The controller and agents could be running on physical machines or virtual machine (VM) instances.

VSM is being released under the Apache 2.0 license, and hosted at <http://github.com/01org/virtual-storage-manager/>.

A mailing list has been established for VSM at the following location: <http://vsm-discuss.33411.n7.nabble.com/>.

## System Requirements

VSM development team takes diversity into account, so far the supporting matrix is as following:

* OS:
  + CentOS 6.5 (1.x only)/7 (Basic Web Server)
  + Ubuntu 14.04.2 LTS Server
* Ceph
  + Firefly,
  + Giant,
  + Hammer
* Openstack
  + Havana
  + Icehouse
  + Juno
  + Kilo
  + Liberty

***NOTE:***

* Other combos might also be working, but the development team didn't try yet. If you are sure some unlisted combos could work, it’s appreciated to make the development team and community aware of.
* Throughout this document, command line is **bolded** and *italicized;* yellow text is used for emphasis, to draw attention to specific information*.*

## What’s in the Rest of This Guide

This document describes how to install, configure, and use COSBench, a cloud storage benchmarking tool.

* Section 2 covers the REST API definitions
* Section 3 instructs how to test API through tools

# REST API Definition

## General guidelines

The REST API is still rapid evolving, the development team is cleaning up it and formulating it, it’s expected to see changes without prior notice.

## API Definitions

### Appnode

1. Create

Description: Create a new appnode which connects to openstack.

CURRENTLY USED: YES

USER: admin

NOTE:

ACTION: **POST**

URI: **/v1/{tenant\_id}/appnodes**

Request

* + Header:
  + Body: None

{

"appnodes": {

"os\_tenant\_name": "admin",

"os\_username": "admin",

"os\_password": "intel@123",

"os\_auth\_url": "http://192.168.100.100:5000/v2.0",

"os\_region\_name": "RegionOne",

"ssh\_user": "intel"

}

}

Response:

* + Header: STATUS 201 Created
  + Body: None

OR

* + Header: STATUS 500 Internal Server Error
  + Body

{

"computeFault": {

"message": "The server has either erred or is incapable of performing the requested operation.",

"code": 500

}

}

1. List

Description: List all appnodes which connects to openstack.

CURRENTLY USED: YES

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/appnodes**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"appnodes": [

{

"os\_username": "admin",

"xtrust\_user": "jack",

"os\_tenant\_name": "admin",

"os\_auth\_url": "http://192.168.81.232:5000/v2.0",

"os\_region\_name": "RegionOne",

"os\_password": "admin",

"vsmapp\_id": 1,

"id": 1,

"uuid": "1515c24e-5ba6-42fd-ae2b-64c0ad3be215",

"ssh\_status": "reachable",

"log\_info": null

}

]

}

NOTE:

1. Delete

Description: Delete an appnode which connects to openstack by appnode id.

CURRENTLY USED: **NO [API is ok]**

USER: admin

ACTION: **DELETE**

URI: **/v1/{tenant\_id}/appnodes/{appnode\_id}**

Request Body(JSON):

None

Response:

STATUS 201 Created

None

OR

STATUS 404 Found

{

"itemNotFound": {

"message": "AppNodeNotFound: App Node \u0001ould not be found.",

"code": 404

}

}

NOTE:

1. Update

Description: Update an appnode which connects to openstack by appnode id.

CURRENTLY USED: **YES**

USER: admin

ACTION: **PUT**

URI: **/v1/{tenant\_id}/appnodes/{appnode\_id}**

Request Body(JSON):

{

"appnode": {

"os\_tenant\_name": "admin",

"os\_username": "admin",

"os\_password": "intel@123",

"os\_auth\_url": "http://192.168.100.100:5000/v2.0",

"os\_region\_name": "RegionOne",

"xtrust\_user": "intel",

"ssh\_status": "",

"log\_info": ""

}

}

Response:

STATUS 201 Created

None

OR

{

"computeFault": {

"message": "The server has either erred or is incapable of performing the requested operation.",

"code": 500

}

}

NOTE:

### cluster

1. Create

Description: Create a ceph cluster.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/****clusters**

Request Body(JSON):

{

"cluster": {

"name": "default",

"file\_system": "xfs",

"journal\_size": "",

"size": "",

"management\_network": "",

"ceph\_public\_network": "",

"cluster\_network": "",

"primary\_public\_netmask": "",

"secondary\_public\_netmask": "",

"cluster\_netmask": "",

"servers": [

{"is\_storage": "True", "is\_monitor": "True", "id": "1"},

{"is\_storage": "True", "is\_monitor": "True", "id": "2"},

{"is\_storage": "True", "is\_monitor": "True", "id": "3"}

]

}

}

Response:

STATUS 202 Accepted

None

NOTE:

1. List

Description: List all ceph clusters.

CURRENTLY USED: **NO [API is ok]**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/clusters**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"clusters": [

{

"cluster\_ip\_netmask": "",

"id": 1,

"size": "",

"management\_network": "",

"name": "cluster\_a",

"scecondary\_public\_ip\_netmask": "",

"primary\_public\_ip\_netmask": "",

"file\_system": "",

"ceph\_public\_network": "1",

"cluster\_network": "14",

"journal\_size": ""

}

]

}

NOTE:

1. Summary

Description: The summary of ceph cluster.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/clusters/summary**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"cluster-summary": {

"status": [],

"cluster": "c658028e-7d64-11e5-9495-000c296d7a45",

"health\_list": [

"HEALTH\_WARN",

"clock skew detected on mon.2, mon.0"

],

"detail": [

"mon.2 addr 192.168.81.236:6789/0 clock skew 0.782834s > max 0.2s (latency 0.00442355s)",

"mon.0 addr 192.168.81.238:6789/0 clock skew 0.466396s > max 0.2s (latency 0.00393478s)"

],

"updated\_at": "2015-10-28 13:30:49"

}

}

NOTE:

1. Refresh

Description: Refresh the ceph cluster status

CURRENTLY USED: **YES [used in refresh-cluster-status script]**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/clusters/refresh**

Request Body(JSON):

None

Response:

STATUS 200 OK

None

NOTE:

1. Import\_ceph\_conf

Description: Import ceph config.

CURRENTLY USED: **YES [used in import\_ceph\_conf script]**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/clusters/import\_ceph\_conf**

Request Body(JSON):

{

"cluster": {

"cluster\_name": "cluster\_a",

"ceph\_conf\_path": ""

}

}

Response:

{

"message": "Success"

}

NOTE:

1. Stop\_cluster

Description: Stop the ceph cluster.

CURRENTLY USED: **NO[API is ok]**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/clusters/stop\_cluster**

Request Body(JSON):

{

"cluster": {

"id": 1

}

}

Response:

STATUS 200 OK

{

"message": "Success"

}

NOTE:

1. Start\_cluster

Description: Start the ceph cluster.

CURRENTLY USED: **NO [API is ok]**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/clusters/start\_cluster**

Request Body(JSON):

{

"cluster": {

"id": 1

}

}

Response:

STATUS 200 OK

{

"message": "Success"

}

NOTE:

### device

1. List

Description: List all devices.

CURRENTLY USED: **NO [API is ok]**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/devices**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"devices": [

{

"journal\_state": "OK",

"state": "OK",

"name": "/dev/sdb1",

"device\_type": "10krpm\_sas",

"used\_capacity\_kb": 37508,

"path": "/dev/sdb1",

"journal": "/dev/sdc1",

"total\_capacity\_kb": 10473472,

"id": 1,

"avail\_capacity\_kb": 10435964

},

{

"journal\_state": "OK",

"state": "OK",

"name": "/dev/sdb1",

"device\_type": "10krpm\_sas",

"used\_capacity\_kb": 37704,

"path": "/dev/sdb1",

"journal": "/dev/sdc1",

"total\_capacity\_kb": 10473472,

"id": 2,

"avail\_capacity\_kb": 10435768

},

{

"journal\_state": "OK",

"state": "OK",

"name": "/dev/sdb1",

"device\_type": "10krpm\_sas",

"used\_capacity\_kb": 36956,

"path": "/dev/sdb1",

"journal": "/dev/sdc1",

"total\_capacity\_kb": 10473472,

"id": 3,

"avail\_capacity\_kb": 10436516

}

]

}

NOTE:

1. Get\_available\_disks

Description: Get available disks.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/devices/get\_available\_disks{?query\_string}**

1. URI: **/v1/{tenant\_id}/devices/get\_available\_disks?result\_mode=get\_disks&server\_id={serves\_id}**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"available\_disks": [

"/dev/disk/by-path/pci-0000:00:10.0-scsi-0:0:0:0",

"/dev/disk/by-path/pci-0000:00:10.0-scsi-0:0:0:0-part5",

"/dev/disk/by-path/pci-0000:00:10.0-scsi-0:0:2:0",

"/dev/disk/by-path/pci-0000:00:10.0-scsi-0:0:0:0-part2",

"/dev/disk/by-path/pci-0000:00:10.0-scsi-0:0:1:0",

"/dev/sda5",

"/dev/disk/by-path/pci-0000:00:10.0-scsi-0:0:2:0-part1",

"/dev/sdb",

"/dev/sdc",

"/dev/sda2",

"/dev/sda"

]

}

NOTE:

1. Get\_smart\_info

Description: Get smart info of device.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/devices/get\_smart\_info{?query\_string}**

1. URI: **/v1/{tenant\_id}/devices/get\_smart\_info?device\_path={path}&device\_id={device\_id}**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"smart\_info": {

"smart": {},

"basic": {

"Serial Number": "",

"Drive Family": " VMware,",

"Drive Status": " OK",

"Firmware Version": ""

}

}

}

1. URI: **/v1/{tenant\_id}/devices/get\_smart\_info?device\_path=&device\_id={device\_id}**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"smart\_info": {

"smart": {},

"basic": {

"Serial Number": "",

"Drive Family": "",

"Drive Status": "",

"Firmware Version": ""

}

}

}

NOTE:

### mds

1. List

Description: Get a list of mds.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/mdses**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"mdses": [

{

"name": "mds\_name",

"updated\_at": "2015-10-28 14:33:48",

"state": "state",

"gid": 1234,

"address": "address",

"id": 1234

}

]

}

NOTE:

1. Summary

Description: Summary info of mds.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/mdses/summary**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"mds-summary": {

"num\_stopped\_mdses": 0,

"num\_max\_mdses": 1,

"epoch": 11,

"metadata\_pool": 1,

"updated\_at": "2015-10-28 14:45:38",

"data\_pools": [

0

],

"num\_failed\_mdses": 0,

"num\_in\_mdses": 1,

"num\_up\_mdses": 1

}

}

NOTE:

### monitor

1. List

Description: Get a list of monitors.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/monitors**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"monitors": [

{

"details": "-",

"address": "192.168.81.235:6789/0",

"health": "HEALTH\_OK",

"id": 1,

"name": "1"

},

{

"details": "clock skew 0.784328s > max 0.2s",

"address": "192.168.81.236:6789/0",

"health": "HEALTH\_WARN",

"id": 2,

"name": "2"

},

{

"details": "clock skew 0.464823s > max 0.2s",

"address": "192.168.81.238:6789/0",

"health": "HEALTH\_WARN",

"id": 3,

"name": "0"

}

]

}

NOTE:

1. Summary

Description: Summary info of monitor.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/monitors/summary**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"monitor-summary": {

"election\_epoch": 10,

"quorum": "1 2 0",

"monmap\_epoch": 1,

"updated\_at": "2015-10-28 14:56:22",

"overall\_status": "HEALTH\_WARN",

"quorum\_leader\_name": "1",

"monitors": 3

}

}

NOTE:

1. Restart

Description: Restart monitor by monitor id.

CURRENTLY USED: **NO[API is ok]**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/monitors/{monitor\_id}/action**

Request Body(JSON):

{

"restart": ""

}

Response:

STATUS 202 Accepted

None

NOTE:

### osd

1. Get

Description: Get a osd by osd id.

CURRENTLY USED: **NO[API is ok]**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/osds/{osd\_id}**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"osd": {

"cluster\_ip": "192.168.81.238",

"storage\_group": {

"status": "IN",

"name": "performance",

"deleted": false,

"created\_at": "2015-10-28T11:00:49.000000",

"friendly\_name": "Performance\_Disk",

"updated\_at": "2015-10-28T11:12:56.000000",

"rule\_id": 0,

"drive\_extended\_threshold": 3,

"storage\_class": "10krpm\_sas",

"deleted\_at": null,

"id": 1

},

"zone\_id": 1,

"weight": 1,

"deleted": false,

"storage\_group\_id": 1,

"created\_at": "2015-10-28T11:01:35.000000",

"osd\_name": "osd.0",

"updated\_at": "2015-10-28T15:09:10.000000",

"public\_ip": "192.168.81.238",

"state": "In-Up",

"operation\_status": "Present",

"service\_id": 3,

"device": {

"mount\_point": "/var/lib/ceph/osd/osd0",

"name": "/dev/sdb1",

"used\_capacity\_kb": 37520,

"deleted": false,

"created\_at": "2015-10-28T11:01:35.000000",

"updated\_at": "2015-10-28T15:11:17.000000",

"interface\_type": null,

"id": 1,

"journal\_state": "OK",

"state": "OK",

"fs\_type": "xfs",

"device\_type": "10krpm\_sas",

"service\_id": 3,

"journal": "/dev/sdc1",

"path": "/dev/sdb1",

"deleted\_at": null,

"total\_capacity\_kb": 10473472,

"avail\_capacity\_kb": 10435952

},

"cluster\_id": 1,

"deleted\_at": null,

"id": 1,

"device\_id": 1

}

}

NOTE:

1. List

Description: Get a list of osds.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/osds**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"osds": [

{

"state": "In-Up",

"operation\_status": "Present",

"weight": 1,

"updated\_at": "2015-10-27T18:39:59.000000",

"service\_id": 3,

"osd\_name": "osd.0",

"id": 1,

"device\_id": 1

},

{

"state": "In-Up",

"operation\_status": "Present",

"weight": 1,

"updated\_at": "2015-10-27T18:39:59.000000",

"service\_id": 5,

"osd\_name": "osd.1",

"id": 2,

"device\_id": 2

},

{

"state": "In-Up",

"operation\_status": "Present",

"weight": 1,

"updated\_at": "2015-10-27T18:39:59.000000",

"service\_id": 7,

"osd\_name": "osd.2",

"id": 3,

"device\_id": 3

}

]

}

NOTE:

1. Restart

Description: Restart osd by osd id.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/osds/{osd\_id}/action**

Request Body(JSON):

{

"restart": ""

}

Response:

STATUS 202 Accepted

None

NOTE:

1. Remove

Description: Remove osd by osd id.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/osds/{osd\_id}/action**

Request Body(JSON):

{

"remove": ""

}

Response:

STATUS 202 Accepted

None

NOTE:

1. Add

Description: Add new disks into ceph cluster.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/osds/****add**

Request Body(JSON):

{

"server\_id": 1,

"osdinfo": [

{"storage\_group\_id": 1, "weight": 1.0, "journal": "/dev/sde1", "data": "/dev/sdd1"}

]

}

Response:

STATUS 202 Accepted

NOTE:

1. Restore

Description: Restore a osd by osd id.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/osds/{osd\_id}/action**

Request Body(JSON):

{

"restore": ""

}

Response:

STATUS 202 Accepted

NOTE:

1. Refresh

Description: Refresh a osd by osd id.

CURRENTLY USED: **NO[API is ok]**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/osds/{osd\_id}/action**

Request Body(JSON):

None

Response:

STATUS 200 OK

None

NOTE:

1. Summary

Description: Summary info of osd.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/osds/summary**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"osd-summary": {

"full": false,

"nearfull": false,

"num\_osds": 4,

"updated\_at": "2015-10-28 15:32:14",

"num\_up\_osds": 4,

"epoch": 72,

"num\_in\_osds": 4

}

}

NOTE:

### performance\_metric

1. Get\_metrics

Description: Get a list of metric by metric name and timestamp.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/performance\_metrics/get\_metrics{?query\_string}**

Request Body(JSON):

None

Response:

{

"metrics": [

{

"deleted": null,

"timestamp": 1446475096,

"created\_at": null,

"hostname": "vsm-node1",

"updated\_at": null,

"value": "4638",

"instance": "osd\_0",

"deleted\_at": null,

"id": 197,

"metric": "osd\_op\_in\_bytes"

},

{

"deleted": null,

"timestamp": 1446475096,

"created\_at": null,

"hostname": "vsm-node1",

"updated\_at": null,

"value": "0",

"instance": "osd\_0",

"deleted\_at": null,

"id": 198,

"metric": "osd\_op\_out\_bytes"

},

……

]

}

NOTE:

### placement\_group

1. List

Description: Get a list of placement groups.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/placement\_groups**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"placement\_groups": [

{

"acting": "abc",

"state": "st",

"pg\_id": 12345,

"id": 1234,

"up": "true"

},

{

"acting": "abc",

"state": "st",

"pg\_id": 12345,

"id": 1234,

"up": "true"

}

]

}

NOTE:

1. Summary

Description: Summary info of placement group.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/placement\_groups/summary**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"placement\_group-summary": {

"bytes\_total": 42899341312,

"degraded\_objects": 0,

"num\_pgs": 399,

"data\_bytes": 1878,

"degraded\_total": 0,

"bytes\_used": 162758656,

"unfound\_ratio": 0,

"op\_per\_sec": 0,

"write\_bytes\_sec": 0,

"updated\_at": "2015-10-28 15:50:34",

"unfound\_objects": 0,

"version": 338,

"pgs\_by\_state": [

{

"count": 399,

"state\_name": "active+clean"

}

],

"read\_bytes\_sec": 0,

"degraded\_ratio": 0,

"bytes\_avail": 42736582656,

"unfound\_total": 0

}

}

NOTE:

### pool\_usage

1. Create

Description: Present pools to openstack.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/poolusages**

Request Body(JSON):

{

"poolusages": [

{"pool\_id": 1, "cinder\_volume\_host": "volume", "appnode\_id": 2}

]

}

Response:

STATUS 202 Accepted

{

"status": "ok",

"host": []

}

NOTE:

1. List

Description: Get a list of pool usage.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/poolusages**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"poolusages": [

{

"attach\_at": "2015-10-28T16:00:16.000000",

"attach\_status": "success",

"vsmapp\_id": 1,

"id": 1,

"pool\_id": 4

}

]

}

NOTE:

### rbd\_pool

1. List

Description: Get a list of rbd pools.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/rbd\_pools**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"rbd\_pools": []

}

NOTE:

1. Summary

Description: Summary info of rbd pool.

CURRENTLY USED: **NO[API is ok]**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/rbd\_pools/summary**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"rbd-summary": {

"full": false,

"num\_up\_rbd\_pools": 8,

"num\_rbd\_pools": 12,

"nearfull": false,

"epoch": 123,

"num\_in\_rbd\_pools": 8

}

}

NOTE:

### server

1. Get

Description: Get a vsm server by server id.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/servers/{server\_id}**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"server": {

"cluster\_ip": "192.168.81.235",

"raw\_ip": "192.168.1.3,192.168.2.3,192.168.3.3",

"secondary\_public\_ip": "192.168.81.235",

"primary\_public\_ip": "192.168.81.235",

"host": "vsm-node2",

"ceph\_ver": "0.80.10",

"id": 2,

"zone\_id": 1,

"osds": "1",

"status": "Active",

"type": ""

}

}

NOTE:

1. List

Description: Get a list of vsm servers.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/servers**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"servers": [

{

"cluster\_ip": "192.168.81.238",

"raw\_ip": "192.168.1.3,192.168.2.3,192.168.3.3",

"secondary\_public\_ip": "192.168.81.238",

"primary\_public\_ip": "192.168.81.238",

"host": "vsm-node1",

"ceph\_ver": "0.80.10",

"id": 1,

"zone\_id": 1,

"osds": "2",

"status": "Active",

"service\_id": 3,

"type": "monitor,storage,"

},

{

"cluster\_ip": "192.168.81.235",

"raw\_ip": "192.168.1.3,192.168.2.3,192.168.3.3",

"secondary\_public\_ip": "192.168.81.235",

"primary\_public\_ip": "192.168.81.235",

"host": "vsm-node2",

"ceph\_ver": "0.80.10",

"id": 2,

"zone\_id": 1,

"osds": "1",

"status": "Active",

"service\_id": 5,

"type": ""

},

{

"cluster\_ip": "192.168.81.236",

"raw\_ip": "192.168.1.3,192.168.2.3,192.168.3.3",

"secondary\_public\_ip": "192.168.81.236",

"primary\_public\_ip": "192.168.81.236",

"host": "vsm-node3",

"ceph\_ver": "0.80.10",

"id": 3,

"zone\_id": 1,

"osds": "1",

"status": "Active",

"service\_id": 7,

"type": ""

},

{

"cluster\_ip": "192.168.81.138",

"raw\_ip": "192.168.1.3,192.168.2.3,192.168.3.3",

"secondary\_public\_ip": "192.168.81.138",

"primary\_public\_ip": "192.168.81.138",

"host": "vsm-node4",

"ceph\_ver": "0.80.10",

"id": 4,

"zone\_id": 1,

"osds": "1",

"status": "available",

"service\_id": 9,

"type": "monitor,storage,"

}

]

}

NOTE:

1. Add

Description: Add a vsm server to ceph cluster.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/servers/add**

Request Body(JSON):

{

"servers":[

{"cluster\_id":1,"id":4,"is\_monitor":"true","is\_storage":"true","zone\_id":1}

]

}

Response:

STATUS 202 Accepted

None

NOTE:

1. Remove

Description: Remove a vsm server from ceph cluster.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/servers/remove**

Request Body(JSON):

{

"servers":[

{"cluster\_id":1,"id":4,"remove\_monitor":"true","remove\_storage":"true"}

]

}

Response:

STATUS 202 Accepted

None

NOTE:

1. Reset\_status

Description: Reset status of vsm server.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/servers/reset\_status**

Request Body(JSON):

{

"servers": 1

}

Response:

STATUS 200 OK

{

"status": "ok"

}

NOTE:

1. Start

Description: Start a vsm server.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/servers/start**

Request Body(JSON):

{

"servers":[

{"cluster\_id":1,"id":4}

]

}

Response:

STATUS 202 Accepted

None

NOTE:

1. Stop

Description: Stop a vsm server.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/servers/stop**

Request Body(JSON):

{

"servers":[

{"cluster\_id":1,"id":4,"remove\_monitor":"true","remove\_storage":"true"}

]

}

Response:

STATUS 202 Accepted

None

NOTE:

1. Ceph\_upgrade

Description: Ceph upgrade.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/servers/ceph\_upgrade**

Request Body(JSON):

{

"pkg\_url": "",

"key\_url": "",

"proxy": "",

"ssh\_user": ""

}

Response:

{

"message": "ceph upgrade from 0.80.10 to 0.94.2 success"

}

NOTE:

### storage\_group

1. Create

Description: Create a storage group

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/storage\_groups**

Request Body(JSON):

{

"storage\_group": {

"name": "group",

"friendly\_name": "name",

"storage\_class": "ssd",

"cluster\_id": 1

}

}

Response:

STATUS 202 Accepted

None

NOTE:

1. List

Description: Get a list of storage groups.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/storage\_groups{/detail}{?query\_string}**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"storage\_groups": [

{

"status": "IN",

"capacity\_used": 162596,

"attached\_pools": 4,

"updated\_at": "2015-10-28 15:59:41",

"capacity\_avail": 41731292,

"capacity\_total": 41893888,

"id": 1,

"name": "performance",

"friendly\_name": "Performance Disk",

"largest\_node\_capacity\_used": 77740,

"storage\_class": "10krpm\_sas",

"attached\_osds": 4

},

{

"status": "IN",

"capacity\_used": 0,

"attached\_pools": 0,

"updated\_at": "2015-10-28 15:59:41",

"capacity\_avail": 0,

"capacity\_total": 0,

"id": 2,

"name": "capacity",

"friendly\_name": "Economy Disk",

"largest\_node\_capacity\_used": 0,

"storage\_class": "7200\_rpm\_sata",

"attached\_osds": 0

},

{

"status": "IN",

"capacity\_used": 0,

"attached\_pools": 0,

"updated\_at": "2015-10-28 15:59:41",

"capacity\_avail": 0,

"capacity\_total": 0,

"id": 3,

"name": "high\_performance",

"friendly\_name": "High Performance",

"largest\_node\_capacity\_used": 0,

"storage\_class": "ssd",

"attached\_osds": 0

}

]

}

NOTE:

1. Summary

Description: Summary info of storage group.

CURRENTLY USED: **NO[API is ok]**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/storage\_groups/summary**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"storage\_group-summary": {

"full": false,

"num\_up\_storage\_groups": 8,

"num\_storage\_groups": 12,

"nearfull": false,

"epoch": 123,

"num\_in\_storage\_groups": 8

}

}

NOTE:

### storage\_pool

1. List

Description: Get a list of storage pools.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/storage\_pools{/detail}{?query\_string}**

Request Body(JSON):

None

Response:

{

"pool": [

{

"status": "running",

"num\_write\_kb": 0,

"write\_bytes\_sec": null,

"op\_per\_sec": null,

"updated\_at": "2015-10-28 16:39:24",

"num\_objects\_degraded": 0,

"createdDate": "2015-10-28T11:12:57.000000",

"clusterId": 1,

"quota": null,

"replica\_storage\_group": null,

"tag": "SYSTEM",

"num\_read": 0,

"createdBy": "ceph",

"crashRelayInterval": 0,

"pgpNum": 133,

"id": 1,

"size": 3,

"num\_objects\_unfound": 0,

"crushRuleset": 0,

"name": "rbd",

"num\_object\_clones": 0,

"cache\_tier\_status": null,

"recipeId": null,

"num\_objects": 0,

"pool\_id": 2,

"num\_read\_kb": 0,

"minSize": 2,

"erasure\_code\_status": null,

"storageGroup": "performance",

"ruleset": null,

"poolId": 2,

"read\_bytes\_sec": null,

"pgNum": 166,

"num\_write": 0,

"num\_bytes": 0

},

{

"status": "running",

"num\_write\_kb": 0,

"write\_bytes\_sec": null,

"op\_per\_sec": null,

"updated\_at": "2015-10-28 16:39:24",

"num\_objects\_degraded": 0,

"createdDate": "2015-10-28T11:12:57.000000",

"clusterId": 1,

"quota": null,

"replica\_storage\_group": null,

"tag": "SYSTEM",

"num\_read": 0,

"createdBy": "ceph",

"crashRelayInterval": 45,

"pgpNum": 133,

"id": 2,

"size": 3,

"num\_objects\_unfound": 0,

"crushRuleset": 0,

"name": "data",

"num\_object\_clones": 0,

"cache\_tier\_status": null,

"recipeId": null,

"num\_objects": 0,

"pool\_id": 0,

"num\_read\_kb": 0,

"minSize": 2,

"erasure\_code\_status": null,

"storageGroup": "performance",

"ruleset": null,

"poolId": 0,

"read\_bytes\_sec": null,

"pgNum": 133,

"num\_write": 0,

"num\_bytes": 0

},

{

"status": "running",

"num\_write\_kb": 0,

"write\_bytes\_sec": 257,

"op\_per\_sec": 0,

"updated\_at": "2015-10-28 16:39:24",

"num\_objects\_degraded": 0,

"createdDate": "2015-10-28T11:12:57.000000",

"clusterId": 1,

"quota": null,

"replica\_storage\_group": null,

"tag": "SYSTEM",

"num\_read": 0,

"createdBy": "ceph",

"crashRelayInterval": 0,

"pgpNum": 133,

"id": 3,

"size": 3,

"num\_objects\_unfound": 0,

"crushRuleset": 0,

"name": "metadata",

"num\_object\_clones": 0,

"cache\_tier\_status": null,

"recipeId": null,

"num\_objects": 19,

"pool\_id": 1,

"num\_read\_kb": 0,

"minSize": 2,

"erasure\_code\_status": null,

"storageGroup": "performance",

"ruleset": null,

"poolId": 1,

"read\_bytes\_sec": null,

"pgNum": 133,

"num\_write": 0,

"num\_bytes": 1867

},

{

"status": "running",

"num\_write\_kb": 0,

"write\_bytes\_sec": null,

"op\_per\_sec": null,

"updated\_at": "2015-10-28 16:39:24",

"num\_objects\_degraded": 0,

"createdDate": "2015-10-28T15:53:49.000000",

"clusterId": 1,

"quota": 1,

"replica\_storage\_group": null,

"tag": "vsm",

"num\_read": 0,

"createdBy": "VSM",

"crashRelayInterval": 0,

"pgpNum": 133,

"id": 4,

"size": 3,

"num\_objects\_unfound": 0,

"crushRuleset": 0,

"name": "testp1",

"num\_object\_clones": 0,

"cache\_tier\_status": null,

"recipeId": null,

"num\_objects": 0,

"pool\_id": 3,

"num\_read\_kb": 0,

"minSize": 2,

"erasure\_code\_status": "",

"storageGroup": "performance",

"ruleset": null,

"poolId": 3,

"read\_bytes\_sec": null,

"pgNum": 133,

"num\_write": 0,

"num\_bytes": 0

}

]

}

NOTE:

1. Add\_cache\_tier

Description: Add cache\_tier pool to ceph cluster.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/storage\_pools/add\_cache\_tier**

Request Body(JSON):

{

"cache\_tier":{

"storage\_pool\_id":"4",

"cache\_pool\_id":"3",

"cache\_mode":"readonly",

"force\_nonempty":true,

"options":{

"hit\_set\_type":"bloom",

"hit\_set\_count":"1",

"hit\_set\_period\_s":"3600",

"target\_max\_mem\_mb":"1000000",

"target\_dirty\_ratio":"0.4",

"target\_full\_ratio":"0.8",

"target\_max\_objects":"1000000",

"target\_min\_flush\_age\_m":"10",

"target\_min\_evict\_age\_m":"20"

}

}

}

Response:

STATUS 200 OK

None

NOTE:

1. Remove\_cache\_tier

Description: Remove cahce\_tier from ceph cluster.

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/storage\_pools/remove\_cache\_tier**

Request Body(JSON):

{

"cache\_tier":{

"cache\_pool\_id":"3"

}

}

Response:

STATUS 200 OK

None

NOTE:

1. Ec\_profiles

Description: Get a list of ec\_profiles.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/storage\_pools/get\_ec\_profile\_list**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"ec\_profiles": [

{

"id": 1,

"name": "default\_profile"

}

]

}

NOTE:

1. Create

Description: Create replicated pool or ec\_pool.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/storage\_pools/create**

Request Body(JSON):

{

"pool":{

"name":"tp01",

"storageGroupId":"1",

"replicatedStorageGroupId":"",

"replicationFactor":"3",

"max\_pg\_num\_per\_osd":"100",

"tag":"vsm",

"clusterId":"0",

"createdBy":"VSM",

"enablePoolQuota":true,

"poolQuota":"1"

}

}

Response:

STATUS 200 OK

None

Request Body(JSON):

{

"pool":{

"name":"ec\_pp",

"storageGroupId":"1",

"tag":"vsm",

"clusterId":"0",

"createdBy":"VSM",

"ecProfileId":"1",

"ecFailureDomain":"osd",

"enablePoolQuota":true,

"poolQuota":"1"

}

}

Response:

STATUS 200 OK

{

"message": "pool ec\_ppcreated"

}

NOTE:

### vsm\_setting

1. Get

Description: Get vsm setting by name

CURRENTLY USED: **NO[API is ok]**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/vsm\_settings/get\_by\_name{?query\_string}**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"setting": {

"id": 1,

"value": "65",

"name": "storage\_group\_near\_full\_threshold"

}

}

NOTE:

1. List

Description: Get a list of vsm settings.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/vsm\_settings**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"settings": [

{

"id": 1,

"value": "65",

"name": "storage\_group\_near\_full\_threshold"

},

{

"id": 2,

"value": "85",

"name": "storage\_group\_full\_threshold"

},

{

"id": 3,

"value": "75",

"name": "ceph\_near\_full\_threshold"

},

{

"id": 4,

"value": "90",

"name": "ceph\_full\_threshold"

},

……

]

}

NOTE:

1. Create

Description: Create a vsm settings. [now for update]

CURRENTLY USED: **YES**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/vsm\_settings**

Request Body(JSON):

{

"setting": {

"name": "storage\_group\_near\_full\_threshold",

"value": "80"

}

}

Response:

STATUS 202 Accepted

{

"setting": {

"id": 1,

"value": "80",

"name": "storage\_group\_near\_full\_threshold"

}

}

NOTE:

### zone

1. Create

Description: Create a zone.

CURRENTLY USED: **NO[API is ok]**

USER: admin

ACTION: **POST**

URI: **/v1/{tenant\_id}/zones**

Request Body(JSON):

{

"zone":{

"name":"testzone",

"cluster\_id":1

}

}

Response:

STATUS 202 Accepted

None

NOTE:

1. List

Description: Get a list of zones.

CURRENTLY USED: **YES**

USER: admin

ACTION: **GET**

URI: **/v1/{tenant\_id}/zones**

Request Body(JSON):

None

Response:

STATUS 200 OK

{

"zones": [

{

"id": 1,

"name": "zone\_one"

}

]

}

NOTE:

# Example

1. Running “admin-token” command to get token-tenant\_id
2. GET:

curl -XGET --header 'Content-type: application/json' --header 'X-Auth-Token:706683f02dd147e391f2df8d7e7084a4' http://192.168.123.193:8778/v1/733c7b55f69f41a3ac33b046cff111a3/clusters

1. POST:

curl -XPOST --header 'Content-type: application/json' --header 'X-Auth-Token:706683f02dd147e391f2df8d7e7084a4' http://192.168.123.193:8778/v1/733c7b55f69f41a3ac33b046cff111a3/storage\_groups --data '{"storage\_group": {"name": "group","friendly\_name": "name","storage\_class": "ssd","cluster\_id": 1}}'