



Red Hat Virtualization 4.1

REST API Guide

Using the Red Hat Virtualization REST Application Programming Interface

Red Hat Virtualization 4.1 REST API Guide

Using the Red Hat Virtualization REST Application Programming Interface

Red Hat Virtualization Documentation Team
rhev-docs@redhat.com

Legal Notice

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

This guide describes the Red Hat Virtualization Manager Representational State Transfer Application Programming Interface. This guide is generated from documentation comments in the ovirt-engine-api-model code, and is currently partially complete. Updated versions of this documentation will be published as new content becomes available.

Table of Contents

CHAPTER 1. INTRODUCTION	35
1.1. REPRESENTATIONAL STATE TRANSFER	35
1.2. API PREREQUISITES	35
CHAPTER 2. AUTHENTICATION AND SECURITY	37
2.1. TLS/SSL CERTIFICATION	37
2.1.1. Obtaining the CA Certificate	37
2.1.2. Importing a Certificate to a Client	39
2.2. AUTHENTICATION	39
2.2.1. OAuth Authentication	39
2.2.2. Basic Authentication	41
2.2.3. Authentication Sessions	42
2.2.3.1. Requesting an Authenticated Session	42
CHAPTER 3. COMMON CONCEPTS	44
3.1. TYPES	44
3.2. IDENTIFIED TYPES	44
3.3. OBJECTS	44
3.4. COLLECTIONS	44
3.5. REPRESENTATIONS	45
3.5.1. XML representation	45
3.5.2. JSON representation	45
3.6. SERVICES	46
3.7. SEARCHING	49
3.7.1. Maximum results parameter	49
3.7.2. Case sensitivity	49
3.7.3. Search syntax	50
3.7.4. Wildcards	50
3.7.5. Pagination	50
3.8. PERMISSIONS	51
3.9. HANDLING ERRORS	52
CHAPTER 4. QUICK START EXAMPLE	53
4.1. EXAMPLE: ACCESS API ENTRY POINT	53
4.2. EXAMPLE: LIST DATA CENTERS	55
4.3. EXAMPLE: LIST HOST CLUSTERS	56
4.4. EXAMPLE: LIST LOGICAL NETWORKS	57
4.5. EXAMPLE: LIST HOSTS	58
4.6. EXAMPLE: CREATE NFS DATA STORAGE	59
4.7. EXAMPLE: CREATE NFS ISO STORAGE	60
4.8. EXAMPLE: ATTACH STORAGE DOMAINS TO DATA CENTER	62
4.9. EXAMPLE: CREATE VIRTUAL MACHINE	63
4.10. EXAMPLE: CREATE A VIRTUAL MACHINE NIC	65
4.11. EXAMPLE: CREATE VIRTUAL MACHINE DISK	65
4.12. EXAMPLE: ATTACH ISO IMAGE TO VIRTUAL MACHINE	67
4.13. EXAMPLE: START THE VIRTUAL MACHINE	68
CHAPTER 5. REQUESTS	70
CHAPTER 6. SERVICES	98
6.1. AFFINITYGROUP	98
6.1.1. get GET	98
6.1.2. remove DELETE	98

6.1.3. update PUT	99
6.2. AFFINITYGROUPVM	99
6.2.1. remove DELETE	99
6.3. AFFINITYGROUPVMS	99
6.3.1. add POST	99
6.3.2. list GET	100
6.3.2.1. max	100
6.4. AFFINITYGROUPS	100
6.4.1. add POST	100
6.4.2. list GET	101
6.4.2.1. max	101
6.5. AFFINITYLABEL	101
6.5.1. get GET	101
6.5.2. remove DELETE	102
6.5.3. update PUT	102
6.6. AFFINITYLABELHOST	102
6.6.1. get GET	102
6.6.2. remove DELETE	102
6.7. AFFINITYLABELHOSTS	103
6.7.1. add POST	103
6.7.2. list GET	103
6.8. AFFINITYLABELVM	103
6.8.1. get GET	103
6.8.2. remove DELETE	104
6.9. AFFINITYLABELVMS	104
6.9.1. add POST	104
6.9.2. list GET	104
6.10. AFFINITYLABELS	104
6.10.1. add POST	105
6.10.2. list GET	105
6.10.2.1. max	105
6.11. ASSIGNEDAFFINITYLABEL	105
6.11.1. get GET	105
6.11.2. remove DELETE	106
6.12. ASSIGNEDAFFINITYLABELS	106
6.12.1. add POST	106
6.12.2. list GET	106
6.13. ASSIGNEDCPUPROFILE	106
6.13.1. get GET	107
6.13.2. remove DELETE	107
6.14. ASSIGNEDCPUPROFILES	107
6.14.1. add POST	107
6.14.2. list GET	107
6.14.2.1. max	108
6.15. ASSIGNEDDISKPROFILE	108
6.15.1. get GET	108
6.15.2. remove DELETE	108
6.16. ASSIGNEDDISKPROFILES	108
6.16.1. add POST	109
6.16.2. list GET	109
6.16.2.1. max	109
6.17. ASSIGNEDPERMISSIONS	109
6.17.1. add POST	109

6.17.2. list GET	110
6.18. ASSIGNEDROLES	111
6.18.1. list GET	111
6.18.1.1. max	111
6.19. ASSIGNEDTAG	111
6.19.1. get GET	112
6.19.2. remove DELETE	112
6.20. ASSIGNEDTAGS	112
6.20.1. add POST	113
6.20.2. list GET	113
6.20.2.1. max	114
6.21. ASSIGNEDVNICPROFILE	114
6.21.1. get GET	114
6.21.2. remove DELETE	114
6.22. ASSIGNEDVNICPROFILES	114
6.22.1. add POST	114
6.22.2. list GET	115
6.22.2.1. max	115
6.23. ATTACHEDSTORAGEDOMAIN	115
6.23.1. activate POST	115
6.23.2. deactivate POST	116
6.23.3. get GET	116
6.23.4. remove DELETE	116
6.24. ATTACHEDSTORAGEDOMAINS	116
6.24.1. add POST	117
6.24.2. list GET	117
6.24.2.1. max	117
6.25. BALANCE	117
6.25.1. get GET	117
6.25.2. remove DELETE	118
6.26. BALANCES	118
6.26.1. add POST	118
6.26.2. list GET	118
6.26.2.1. max	119
6.27. BOOKMARK	119
6.27.1. get GET	119
6.27.2. remove DELETE	119
6.27.3. update PUT	120
6.28. BOOKMARKS	120
6.28.1. add POST	120
6.28.2. list GET	121
6.28.2.1. max	121
6.29. CLUSTER	121
6.29.1. get GET	122
6.29.2. remove DELETE	123
6.29.3. resetemulatedmachine POST	124
6.29.4. update PUT	124
6.30. CLUSTERLEVEL	124
6.30.1. get GET	125
6.31. CLUSTERLEVELS	125
6.31.1. list GET	126
6.32. CLUSTERNETWORK	126
6.32.1. get GET	126

6.32.2. remove DELETE	127
6.32.3. update PUT	127
6.33. CLUSTERNETWORKS	127
6.33.1. add POST	127
6.33.2. list GET	128
6.33.2.1. max	128
6.34. CLUSTERS	128
6.34.1. add POST	128
6.34.2. list GET	129
6.34.2.1. case_sensitive	129
6.34.2.2. max	129
6.35. COPYABLE	129
6.35.1. copy POST	129
6.36. CPUPROFILE	130
6.36.1. get GET	130
6.36.2. remove DELETE	130
6.36.3. update PUT	130
6.37. CPUFILES	131
6.37.1. add POST	131
6.37.2. list GET	131
6.37.2.1. max	131
6.38. DATACENTER	131
6.38.1. get GET	131
6.38.2. remove DELETE	132
6.38.2.1. force	133
6.38.3. update PUT	133
6.39. DATACENTERNETWORK	134
6.39.1. get GET	134
6.39.2. remove DELETE	134
6.39.3. update PUT	134
6.40. DATACENTERNETWORKS	134
6.40.1. add POST	135
6.40.2. list GET	135
6.40.2.1. max	135
6.41. DATACENTERS	135
6.41.1. add POST	136
6.41.2. list GET	136
6.41.2.1. case_sensitive	137
6.41.2.2. max	138
6.42. DISK	138
6.42.1. copy POST	138
6.42.2. export POST	139
6.42.3. get GET	139
6.42.4. move POST	139
6.42.5. remove DELETE	140
6.42.6. sparsify POST	140
6.43. DISKATTACHMENT	140
6.43.1. get GET	141
6.43.2. remove DELETE	141
6.43.2.1. detach_only	141
6.43.3. update PUT	141
6.44. DISKATTACHMENTS	142
6.44.1. add POST	142

6.44.2. list GET	143
6.45. DISKPROFILE	143
6.45.1. get GET	144
6.45.2. remove DELETE	144
6.45.3. update PUT	144
6.46. DISKPROFILES	144
6.46.1. add POST	144
6.46.2. list GET	145
6.46.2.1. max	145
6.47. DISKSNAPSHOT	145
6.47.1. get GET	145
6.47.2. remove DELETE	145
6.48. DISKSNAPSHOTS	146
6.48.1. list GET	146
6.48.1.1. max	146
6.49. DISKS	146
6.49.1. add POST	146
6.49.2. list GET	148
6.49.2.1. case_sensitive	149
6.49.2.2. max	149
6.49.2.3. unregistered	149
6.50. DOMAIN	149
6.50.1. get GET	150
6.51. DOMAINGROUP	150
6.51.1. get GET	150
6.52. DOMAINGROUPS	150
6.52.1. list GET	151
6.52.1.1. case_sensitive	151
6.52.1.2. max	151
6.53. DOMAINUSER	151
6.53.1. get GET	151
6.54. DOMAINUSERS	152
6.54.1. list GET	152
6.54.1.1. case_sensitive	153
6.54.1.2. max	153
6.55. DOMAINS	153
6.55.1. list GET	153
6.55.1.1. max	154
6.56. ENGINEKATELLOERRATA	154
6.56.1. list GET	154
6.56.1.1. max	155
6.57. EVENT	155
6.57.1. get GET	155
6.57.2. remove DELETE	156
6.58. EVENTS	156
6.58.1. add POST	156
6.58.2. list GET	157
6.58.2.1. case_sensitive	158
6.58.2.2. from	158
6.58.2.3. max	159
6.58.2.4. search	159
6.58.3. undelete POST	160
6.59. EXTERNALCOMPUTERESOURCE	160

6.59.1. get GET	160
6.60. EXTERNALCOMPUTERESOURCES	160
6.60.1. list GET	160
6.60.1.1. max	161
6.61. EXTERNALDISCOVEREDHOST	161
6.61.1. get GET	161
6.62. EXTERNALDISCOVEREDHOSTS	161
6.62.1. list GET	161
6.62.1.1. max	162
6.63. EXTERNALHOST	162
6.63.1. get GET	162
6.64. EXTERNALHOSTGROUP	162
6.64.1. get GET	162
6.65. EXTERNALHOSTGROUPS	162
6.65.1. list GET	162
6.65.1.1. max	163
6.66. EXTERNALHOSTPROVIDER	163
6.66.1. get GET	163
6.66.2. importcertificates POST	163
6.66.3. remove DELETE	164
6.66.4. testconnectivity POST	164
6.66.5. update PUT	164
6.67. EXTERNALHOSTPROVIDERS	164
6.67.1. add POST	164
6.67.2. list GET	165
6.67.2.1. max	165
6.68. EXTERNALHOSTS	165
6.68.1. list GET	165
6.68.1.1. max	165
6.69. EXTERNALPROVIDER	165
6.69.1. importcertificates POST	166
6.69.2. testconnectivity POST	166
6.70. EXTERNALPROVIDERCERTIFICATE	166
6.70.1. get GET	166
6.71. EXTERNALPROVIDERCERTIFICATES	166
6.71.1. list GET	167
6.71.1.1. max	167
6.72. EXTERNALVMIMPORTS	167
6.72.1. add POST	167
6.73. FENCEAGENT	168
6.73.1. get GET	168
6.73.2. remove DELETE	168
6.73.3. update PUT	168
6.74. FENCEAGENTS	169
6.74.1. add POST	169
6.74.2. list GET	169
6.74.2.1. max	169
6.75. FILE	169
6.75.1. get GET	170
6.76. FILES	170
6.76.1. list GET	170
6.76.1.1. case_sensitive	170
6.76.1.2. max	170

6.77. FILTER	171
6.77.1. get GET	171
6.77.2. remove DELETE	171
6.78. FILTERS	171
6.78.1. add POST	171
6.78.2. list GET	172
6.78.2.1. max	172
6.79. GLUSTERBRICK	172
6.79.1. get GET	172
6.79.2. remove DELETE	173
6.79.3. replace POST	174
6.80. GLUSTERBRICKS	174
6.80.1. activate POST	174
6.80.2. add POST	175
6.80.3. list GET	176
6.80.3.1. max	176
6.80.4. migrate POST	176
6.80.5. remove DELETE	177
6.80.6. stopmigrate POST	178
6.80.6.1. bricks	178
6.81. GLUSTERHOOK	178
6.81.1. disable POST	179
6.81.2. enable POST	179
6.81.3. get GET	179
6.81.4. remove DELETE	180
6.81.5. resolve POST	180
6.82. GLUSTERHOOKS	180
6.82.1. list GET	180
6.82.1.1. max	181
6.83. GLUSTERVOLUME	181
6.83.1. get GET	182
6.83.2. getprofilestatistics POST	182
6.83.3. rebalance POST	183
6.83.3.1. fix_layout	183
6.83.3.2. force	183
6.83.4. remove DELETE	183
6.83.5. resetalloptions POST	184
6.83.6. resetoption POST	184
6.83.7. setoption POST	185
6.83.8. start POST	185
6.83.8.1. force	186
6.83.9. startprofile POST	186
6.83.10. stop POST	186
6.83.11. stopprofile POST	186
6.83.12. stoprebalance POST	187
6.84. GLUSTERVOLUMES	187
6.84.1. add POST	187
6.84.2. list GET	188
6.84.2.1. case_sensitive	188
6.84.2.2. max	189
6.85. GROUP	189
6.85.1. get GET	189
6.85.2. remove DELETE	189

6.86. GROUPS	189
6.86.1. add POST	189
6.86.2. list GET	190
6.86.2.1. case_sensitive	190
6.86.2.2. max	190
6.87. HOST	190
6.87.1. activate POST	192
6.87.2. approve POST	192
6.87.3. commitnetconfig POST	192
6.87.4. deactivate POST	193
6.87.4.1. stop_gluster_service	193
6.87.5. enrollcertificate POST	193
6.87.6. fence POST	193
6.87.7. forceselectspm POST	194
6.87.8. get GET	194
6.87.9. install POST	195
6.87.9.1. deploy_hosted_engine	196
6.87.9.2. undeploy_hosted_engine	196
6.87.10. iscsidiscover POST	196
6.87.11. iscsilogin POST	197
6.87.12. refresh POST	197
6.87.13. remove DELETE	197
6.87.14. setupnetworks POST	198
6.87.15. unregisteredstoragedomainsdiscover POST	201
6.87.16. update PUT	202
6.87.17. upgrade POST	202
6.87.18. upgradecheck POST	202
6.88. HOSTDEVICE	203
6.88.1. get GET	203
6.89. HOSTDEVICES	203
6.89.1. list GET	203
6.89.1.1. max	204
6.90. HOSTHOOK	204
6.90.1. get GET	204
6.91. HOSTHOOKS	204
6.91.1. list GET	204
6.91.1.1. max	205
6.92. HOSTNIC	205
6.92.1. get GET	205
6.92.2. updatevirtualfunctionsconfiguration POST	205
6.93. HOSTNICS	205
6.93.1. list GET	206
6.93.1.1. max	206
6.94. HOSTNUMANODE	206
6.94.1. get GET	206
6.95. HOSTNUMANODES	206
6.95.1. list GET	207
6.95.1.1. max	207
6.96. HOSTSTORAGE	207
6.96.1. list GET	207
6.96.1.1. report_status	208
6.97. HOSTS	208
6.97.1. add POST	209

6.97.1.1. deploy_hosted_engine	210
6.97.1.2. undeploy_hosted_engine	210
6.97.2. list GET	210
6.97.2.1. case_sensitive	211
6.97.2.2. max	211
6.98. ICON	211
6.98.1. get GET	211
6.99. ICONS	211
6.99.1. list GET	212
6.99.1.1. max	212
6.100. IMAGE	212
6.100.1. get GET	212
6.100.2. import POST	213
6.101. IMAGETRANSFER	213
6.101.1. extend POST	216
6.101.2. finalize POST	216
6.101.3. get GET	217
6.101.4. pause POST	217
6.101.5. resume POST	217
6.102. IMAGETRANSFERS	217
6.102.1. add POST	217
6.102.2. list GET	218
6.103. IMAGES	218
6.103.1. list GET	218
6.103.1.1. max	218
6.104. INSTANCETYPE	218
6.104.1. get GET	219
6.104.2. remove DELETE	219
6.104.3. update PUT	219
6.105. INSTANCETYPEGRAPHICSCONSOLE	220
6.105.1. get GET	220
6.105.2. remove DELETE	220
6.106. INSTANCETYPEGRAPHICSCONSOLES	221
6.106.1. add POST	221
6.106.2. list GET	221
6.106.2.1. max	221
6.107. INSTANCETYPENIC	221
6.107.1. get GET	222
6.107.2. remove DELETE	222
6.107.3. update PUT	222
6.108. INSTANCETYPENICS	222
6.108.1. add POST	223
6.108.2. list GET	223
6.108.2.1. max	223
6.109. INSTANCETYPEWATCHDOG	223
6.109.1. get GET	224
6.109.2. remove DELETE	224
6.109.3. update PUT	224
6.110. INSTANCETYPEWATCHDOGS	224
6.110.1. add POST	224
6.110.2. list GET	225
6.110.2.1. max	225
6.111. INSTANCETYPES	225

6.111.1. add POST	225
6.111.2. list GET	227
6.111.2.1. case_sensitive	227
6.111.2.2. max	227
6.112. ISCSIBOND	227
6.112.1. get GET	228
6.112.2. remove DELETE	228
6.112.3. update PUT	228
6.113. ISCSIBONDS	229
6.113.1. add POST	229
6.113.2. list GET	229
6.113.2.1. max	230
6.114. JOB	230
6.114.1. clear POST	230
6.114.2. end POST	230
6.114.2.1. succeeded	231
6.114.3. get GET	231
6.115. JOBS	232
6.115.1. add POST	232
6.115.2. list GET	233
6.115.2.1. max	233
6.116. KATELLOERRATA	233
6.116.1. list GET	234
6.116.1.1. max	234
6.117. KATELLOERRATUM	234
6.117.1. get GET	235
6.118. MACPOOL	235
6.118.1. get GET	235
6.118.2. remove DELETE	236
6.118.3. update PUT	236
6.119. MACPOOLS	237
6.119.1. add POST	237
6.119.2. list GET	237
6.119.2.1. max	238
6.120. MEASURABLE	238
6.121. MOVEABLE	238
6.121.1. move POST	238
6.122. NETWORK	238
6.122.1. get GET	238
6.122.2. remove DELETE	239
6.122.3. update PUT	240
6.123. NETWORKATTACHMENT	240
6.123.1. get GET	241
6.123.2. remove DELETE	241
6.123.3. update PUT	241
6.124. NETWORKATTACHMENTS	241
6.124.1. add POST	242
6.124.2. list GET	242
6.124.2.1. max	242
6.125. NETWORKFILTER	242
6.125.1. get GET	243
6.126. NETWORKFILTERS	243
6.126.1. list GET	244

6.127. NETWORKLABEL	244
6.127.1. get GET	244
6.127.2. remove DELETE	245
6.128. NETWORKLABELS	245
6.128.1. add POST	245
6.128.2. list GET	246
6.128.2.1. max	246
6.129. NETWORKS	246
6.129.1. add POST	246
6.129.2. list GET	247
6.129.2.1. case_sensitive	248
6.129.2.2. max	248
6.130. OPENSTACKIMAGE	248
6.130.1. get GET	248
6.130.2. import POST	248
6.131. OPENSTACKIMAGEPROVIDER	249
6.131.1. get GET	250
6.131.2. importcertificates POST	250
6.131.3. remove DELETE	250
6.131.4. testconnectivity POST	250
6.131.5. update PUT	250
6.132. OPENSTACKIMAGEPROVIDERS	251
6.132.1. add POST	251
6.132.2. list GET	251
6.132.2.1. max	251
6.133. OPENSTACKIMAGES	251
6.133.1. list GET	252
6.133.1.1. max	252
6.134. OPENSTACKNETWORK	252
6.134.1. get GET	252
6.134.2. import POST	252
6.134.2.1. data_center	253
6.135. OPENSTACKNETWORKPROVIDER	253
6.135.1. get GET	253
6.135.2. importcertificates POST	254
6.135.3. remove DELETE	254
6.135.4. testconnectivity POST	254
6.135.5. update PUT	254
6.136. OPENSTACKNETWORKPROVIDERS	255
6.136.1. add POST	255
6.136.2. list GET	255
6.136.2.1. max	256
6.137. OPENSTACKNETWORKS	256
6.137.1. list GET	256
6.137.1.1. max	256
6.138. OPENSTACKSUBNET	256
6.138.1. get GET	256
6.138.2. remove DELETE	257
6.139. OPENSTACKSUBNETS	257
6.139.1. add POST	257
6.139.2. list GET	257
6.139.2.1. max	257
6.140. OPENSTACKVOLUMEAUTHENTICATIONKEY	258

6.140.1. get GET	258
6.140.2. remove DELETE	258
6.140.3. update PUT	258
6.141. OPENSTACKVOLUMEAUTHENTICATIONKEYS	258
6.141.1. add POST	259
6.141.2. list GET	259
6.141.2.1. max	259
6.142. OPENSTACKVOLUMEPROVIDER	259
6.142.1. get GET	260
6.142.2. importcertificates POST	260
6.142.3. remove DELETE	260
6.142.4. testconnectivity POST	260
6.142.5. update PUT	260
6.143. OPENSTACKVOLUMEPROVIDERS	261
6.143.1. add POST	261
6.143.2. list GET	262
6.143.2.1. max	262
6.144. OPENSTACKVOLUMETYPE	262
6.144.1. get GET	262
6.145. OPENSTACKVOLUMETYPES	262
6.145.1. list GET	262
6.145.1.1. max	263
6.146. OPERATINGSYSTEM	263
6.146.1. get GET	263
6.147. OPERATINGSYSTEMS	263
6.147.1. list GET	263
6.147.1.1. max	264
6.148. PERMISSION	264
6.148.1. get GET	264
6.148.2. remove DELETE	264
6.149. PERMIT	264
6.149.1. get GET	264
6.149.2. remove DELETE	265
6.150. PERMITS	265
6.150.1. add POST	265
6.150.2. list GET	266
6.150.2.1. max	266
6.151. QOS	266
6.151.1. get GET	267
6.151.2. remove DELETE	267
6.151.3. update PUT	267
6.152. QOSS	267
6.152.1. add POST	268
6.152.2. list GET	268
6.152.2.1. max	268
6.153. QUOTA	268
6.153.1. get GET	268
6.153.2. remove DELETE	269
6.153.3. update PUT	269
6.154. QUOTA_CLUSTERLIMIT	270
6.154.1. get GET	270
6.154.2. remove DELETE	270
6.155. QUOTA_CLUSTERLIMITS	270

6.155.1. add POST	271
6.155.2. list GET	271
6.155.2.1. max	271
6.156. QUOTASTORAGELIMIT	271
6.156.1. get GET	271
6.156.2. remove DELETE	271
6.157. QUOTASTORAGELIMITS	272
6.157.1. add POST	272
6.157.2. list GET	272
6.157.2.1. max	272
6.158. QUOTAS	272
6.158.1. add POST	273
6.158.2. list GET	273
6.158.2.1. max	273
6.159. ROLE	273
6.159.1. get GET	274
6.159.2. remove DELETE	274
6.159.3. update PUT	274
6.160. ROLES	275
6.160.1. add POST	275
6.160.2. list GET	276
6.160.2.1. max	276
6.161. SCHEDULINGPOLICIES	276
6.161.1. add POST	277
6.161.2. list GET	277
6.161.2.1. max	277
6.162. SCHEDULINGPOLICY	277
6.162.1. get GET	277
6.162.2. remove DELETE	278
6.162.3. update PUT	278
6.163. SCHEDULINGPOLICYUNIT	278
6.163.1. get GET	278
6.163.2. remove DELETE	279
6.164. SCHEDULINGPOLICYUNITS	279
6.164.1. list GET	279
6.164.1.1. max	279
6.165. SNAPSHOT	279
6.165.1. get GET	280
6.165.2. remove DELETE	280
6.165.3. restore POST	280
6.166. SNAPSHOTCDROM	281
6.166.1. get GET	281
6.167. SNAPSHOTCDROMS	281
6.167.1. list GET	281
6.167.1.1. max	281
6.168. SNAPSHOTDISK	281
6.168.1. get GET	282
6.169. SNAPSHOTDISKS	282
6.169.1. list GET	282
6.169.1.1. max	282
6.170. SNAPSHOTNIC	282
6.170.1. get GET	282
6.171. SNAPSHOTNICS	283

6.171.1. list GET	283
6.171.1.1. max	283
6.172. SNAPSHOTS	283
6.172.1. add POST	283
6.172.2. list GET	284
6.172.2.1. max	284
6.173. SSHPUBLICKEY	284
6.173.1. get GET	285
6.173.2. remove DELETE	285
6.173.3. update PUT	285
6.174. SSHPUBLICKEYS	285
6.174.1. add POST	285
6.174.2. list GET	286
6.174.2.1. max	286
6.175. STATISTIC	286
6.175.1. get GET	286
6.176. STATISTICS	286
6.176.1. list GET	286
6.176.1.1. max	288
6.177. STEP	288
6.177.1. end POST	288
6.177.1.1. succeeded	288
6.177.2. get GET	289
6.178. STEPS	289
6.178.1. add POST	289
6.178.2. list GET	290
6.178.2.1. max	291
6.179. STORAGE	291
6.179.1. get GET	291
6.179.1.1. report_status	291
6.180. STORAGEDOMAIN	292
6.180.1. get GET	293
6.180.2. isattached POST	293
6.180.3. reduceluns POST	293
6.180.4. refreshluns POST	294
6.180.5. remove DELETE	294
6.180.5.1. destroy	295
6.180.5.2. host	295
6.180.6. update PUT	295
6.180.7. updateovfstore POST	296
6.181. STORAGEDOMAINCONTENTDISK	296
6.181.1. get GET	297
6.182. STORAGEDOMAINCONTENTDISKS	297
6.182.1. list GET	297
6.182.1.1. case_sensitive	297
6.182.1.2. max	297
6.183. STORAGEDOMAINSERVERCONNECTION	298
6.183.1. get GET	298
6.183.2. remove DELETE	298
6.184. STORAGEDOMAINSERVERCONNECTIONS	298
6.184.1. add POST	298
6.184.2. list GET	299
6.184.2.1. max	299

6.185. STORAGEDOMAINTEMPLATE	299
6.185.1. get GET	299
6.185.2. import POST	299
6.185.2.1. clone	300
6.185.3. register POST	300
6.185.3.1. allow_partial_import	301
6.185.4. remove DELETE	301
6.186. STORAGEDOMAINTEMPLATES	301
6.186.1. list GET	301
6.186.1.1. max	302
6.186.1.2. unregistered	302
6.187. STORAGEDOMAINVM	302
6.187.1. get GET	302
6.187.2. import POST	303
6.187.2.1. clone	304
6.187.2.2. collapse_snapshots	304
6.187.3. register POST	304
6.187.3.1. allow_partial_import	305
6.187.3.2. reassign_bad_macs	305
6.187.4. remove DELETE	305
6.188. STORAGEDOMAINVMDISKATTACHMENT	306
6.188.1. get GET	306
6.189. STORAGEDOMAINVMDISKATTACHMENTS	306
6.189.1. list GET	306
6.190. STORAGEDOMAINVMS	306
6.190.1. list GET	307
6.190.1.1. max	307
6.190.1.2. unregistered	307
6.191. STORAGEDOMAINS	308
6.191.1. add POST	308
6.191.2. list GET	309
6.191.2.1. case_sensitive	310
6.191.2.2. max	310
6.192. STORAGESERVERCONNECTION	310
6.192.1. get GET	310
6.192.2. remove DELETE	310
6.192.2.1. host	311
6.192.3. update PUT	311
6.192.3.1. force	311
6.193. STORAGESERVERCONNECTIONEXTENSION	312
6.193.1. get GET	312
6.193.2. remove DELETE	312
6.193.3. update PUT	312
6.194. STORAGESERVERCONNECTIONEXTENSIONS	313
6.194.1. add POST	313
6.194.2. list GET	314
6.194.2.1. max	314
6.195. STORAGESERVERCONNECTIONS	314
6.195.1. add POST	314
6.195.2. list GET	315
6.195.2.1. max	315
6.196. SYSTEM	315
6.196.1. get GET	315

6.196.2. reloadconfigurations POST	317
6.197. SYSTEMPERMISSIONS	317
6.197.1. add POST	317
6.197.2. list GET	318
6.198. TAG	319
6.198.1. get GET	319
6.198.2. remove DELETE	319
6.198.3. update PUT	320
6.199. TAGS	320
6.199.1. add POST	321
6.199.2. list GET	321
6.199.2.1. max	322
6.200. TEMPLATE	322
6.200.1. export POST	323
6.200.1.1. exclusive	323
6.200.2. get GET	323
6.200.3. remove DELETE	323
6.200.4. update PUT	324
6.201. TEMPLATECDROM	324
6.201.1. get GET	325
6.201.1.1. cdrom	325
6.202. TEMPLATECDROMS	325
6.202.1. list GET	326
6.202.1.1. max	326
6.203. TEMPLATEDISK	326
6.203.1. copy POST	326
6.203.2. export POST	326
6.203.3. get GET	327
6.203.4. remove DELETE	327
6.204. TEMPLATEDISKATTACHMENT	327
6.204.1. get GET	327
6.204.2. remove DELETE	327
6.205. TEMPLATEDISKATTACHMENTS	328
6.205.1. list GET	328
6.206. TEMPLATEDISKS	328
6.206.1. list GET	329
6.206.1.1. max	329
6.207. TEMPLATEGRAPHICSCONSOLE	329
6.207.1. get GET	329
6.207.2. remove DELETE	329
6.208. TEMPLATEGRAPHICSCONSOLES	329
6.208.1. add POST	330
6.208.2. list GET	330
6.208.2.1. max	330
6.209. TEMPLATENIC	330
6.209.1. get GET	331
6.209.2. remove DELETE	331
6.209.3. update PUT	331
6.210. TEMPLATENICS	331
6.210.1. add POST	331
6.210.2. list GET	331
6.210.2.1. max	332
6.211. TEMPLATEWATCHDOG	332

6.211.1. get GET	332
6.211.2. remove DELETE	332
6.211.3. update PUT	332
6.212. TEMPLATEWATCHDOGS	333
6.212.1. add POST	333
6.212.2. list GET	333
6.212.2.1. max	333
6.213. TEMPLATES	333
6.213.1. add POST	334
6.213.1.1. clone_permissions	334
6.213.1.2. seal	335
6.213.2. list GET	335
6.213.2.1. case_sensitive	336
6.213.2.2. max	336
6.214. UNMANAGEDNETWORK	336
6.214.1. get GET	336
6.214.2. remove DELETE	336
6.215. UNMANAGEDNETWORKS	336
6.215.1. list GET	337
6.215.1.1. max	337
6.216. USER	337
6.216.1. get GET	337
6.216.2. remove DELETE	338
6.217. USERS	338
6.217.1. add POST	338
6.217.2. list GET	339
6.217.2.1. case_sensitive	340
6.217.2.2. max	340
6.218. VIRTUALFUNCTIONALLOWEDNETWORK	340
6.218.1. get GET	340
6.218.2. remove DELETE	341
6.219. VIRTUALFUNCTIONALLOWEDNETWORKS	341
6.219.1. add POST	341
6.219.2. list GET	341
6.219.2.1. max	341
6.220. VM	341
6.220.1. cancelmigration POST	343
6.220.2. clone POST	343
6.220.3. commitsnapshot POST	343
6.220.4. detach POST	344
6.220.5. export POST	344
6.220.6. freezefilesystems POST	345
6.220.7. get GET	345
6.220.7.1. all_content	346
6.220.7.2. next_run	346
6.220.8. logon POST	347
6.220.9. maintenance POST	347
6.220.10. migrate POST	348
6.220.10.1. cluster	348
6.220.10.2. force	348
6.220.10.3. host	348
6.220.11. previewsnapshot POST	348
6.220.12. reboot POST	349

6.220.13. remove DELETE	349
6.220.13.1. force	350
6.220.14. reordermacaddresses POST	350
6.220.15. shutdown POST	350
6.220.16. start POST	350
6.220.16.1. pause	351
6.220.16.2. use_cloud_init	351
6.220.16.3. use_sysprep	351
6.220.16.4. vm	351
6.220.17. stop POST	352
6.220.18. suspend POST	352
6.220.19. thawfilesystems POST	353
6.220.20. ticket POST	353
6.220.21. undosnapshot POST	354
6.220.22. update PUT	354
6.220.22.1. next_run	355
6.221. VMAPPLICATION	355
6.221.1. get GET	355
6.221.1.1. application	355
6.222. VMAPPLICATIONS	355
6.222.1. list GET	356
6.222.1.1. applications	356
6.222.1.2. max	356
6.223. VMCDROM	356
6.223.1. get GET	357
6.223.1.1. current	357
6.223.2. update PUT	357
6.223.2.1. current	358
6.224. VMCDROMS	359
6.224.1. list GET	359
6.224.1.1. max	359
6.225. VMDISK	359
6.225.1. activate POST	360
6.225.2. deactivate POST	360
6.225.3. export POST	360
6.225.4. get GET	360
6.225.5. move POST	360
6.225.6. remove DELETE	361
6.225.7. update PUT	361
6.226. VMDISKS	361
6.226.1. add POST	362
6.226.2. list GET	362
6.226.2.1. max	362
6.227. VMGRAPHICSCONSOLE	362
6.227.1. get GET	362
6.227.1.1. current	363
6.227.2. proxyticket POST	363
6.227.3. remoteviewerconnectionfile POST	363
6.227.3.1. remote_viewer_connection_file	365
6.227.4. remove DELETE	365
6.227.5. ticket POST	365
6.228. VMGRAPHICSCONSOLES	366
6.228.1. add POST	366

6.228.2. list GET	366
6.228.2.1. current	366
6.228.2.2. max	366
6.229. VMHOSTDEVICE	367
6.229.1. get GET	367
6.229.2. remove DELETE	367
6.230. VMHOSTDEVICES	368
6.230.1. add POST	368
6.230.2. list GET	369
6.230.2.1. max	369
6.231. VMNIC	369
6.231.1. activate POST	370
6.231.2. deactivate POST	370
6.231.3. get GET	370
6.231.4. remove DELETE	370
6.231.5. update PUT	371
6.232. VMNICS	371
6.232.1. add POST	372
6.232.2. list GET	373
6.232.2.1. max	373
6.233. VMNUMANODE	373
6.233.1. get GET	373
6.233.2. remove DELETE	374
6.233.3. update PUT	374
6.234. VMNUMANODES	374
6.234.1. add POST	375
6.234.2. list GET	375
6.234.2.1. max	376
6.235. VMPOOL	376
6.235.1. allocatevm POST	376
6.235.2. get GET	376
6.235.3. remove DELETE	377
6.235.4. update PUT	377
6.236. VMPOOLS	378
6.236.1. add POST	378
6.236.2. list GET	379
6.236.2.1. case_sensitive	379
6.236.2.2. max	379
6.237. VMREPORTEDDEVICE	379
6.237.1. get GET	380
6.238. VMREPORTEDDEVICES	380
6.238.1. list GET	380
6.238.1.1. max	380
6.239. VMSESSION	380
6.239.1. get GET	380
6.240. VMSESSIONS	381
6.240.1. list GET	381
6.240.1.1. max	381
6.241. VMWATCHDOG	382
6.241.1. get GET	382
6.241.1.1. watchdog	382
6.241.2. remove DELETE	382
6.241.3. update PUT	383

6.241.3.1. watchdog	383
6.242. VMWATCHDOGS	383
6.242.1. add POST	384
6.242.1.1. watchdog	384
6.242.2. list GET	384
6.242.2.1. max	385
6.242.2.2. watchdogs	385
6.243. VMS	385
6.243.1. add POST	385
6.243.1.1. clone	388
6.243.1.2. clone_permissions	389
6.243.2. list GET	389
6.243.2.1. all_content	389
6.243.2.2. case_sensitive	390
6.244. VNICPROFILE	390
6.244.1. get GET	390
6.244.2. remove DELETE	391
6.244.3. update PUT	391
6.245. VNICPROFILES	391
6.245.1. add POST	391
6.245.2. list GET	392
6.245.2.1. max	393
6.246. WEIGHT	393
6.246.1. get GET	393
6.246.2. remove DELETE	393
6.247. WEIGHTS	393
6.247.1. add POST	394
6.247.2. list GET	394
6.247.2.1. max	394
CHAPTER 7. TYPES	395
7.1. ACCESSPROTOCOL ENUM	395
7.2. ACTION STRUCT	395
7.3. AFFINITYGROUP STRUCT	399
7.3.1. enforcing	399
7.3.2. positive	400
7.4. AFFINITYLABEL STRUCT	400
7.4.1. read_only	401
7.5. AFFINITYRULE STRUCT	401
7.5.1. enabled	401
7.5.2. enforcing	401
7.5.3. positive	402
7.6. AGENT STRUCT	402
7.6.1. host	403
7.7. AGENTCONFIGURATION STRUCT	403
7.8. API STRUCT	403
7.9. APISUMMARY STRUCT	404
7.10. APISUMMARYITEM STRUCT	405
7.11. APPLICATION STRUCT	405
7.12. ARCHITECTURE ENUM	405
7.13. AUTHORIZEDKEY STRUCT	406
7.14. AUTONUMASTATUS ENUM	406
7.15. BALANCE STRUCT	406

7.16. BIOS STRUCT	407
7.17. BLOCKSTATISTIC STRUCT	407
7.18. BONDING STRUCT	407
7.18.1. ad_partner_mac	408
7.18.2. options	408
7.18.3. slaves	408
7.18.4. active_slave	408
7.19. BOOKMARK STRUCT	409
7.20. BOOT STRUCT	409
7.21. BOOTDEVICE ENUM	409
7.22. BOOTMENU STRUCT	409
7.23. BOOTPROTOCOL ENUM	410
7.23.1. autoconf	410
7.23.2. dhcp	410
7.24. BRICKPROFILEDETAIL STRUCT	410
7.25. CDROM STRUCT	411
7.25.1. vms	411
7.26. CERTIFICATE STRUCT	411
7.27. CLOUDINIT STRUCT	412
7.28. CLUSTER STRUCT	412
7.28.1. custom_scheduling_policy_properties	416
7.28.2. fencing_policy	417
7.28.3. gluster_tuned_profile	417
7.28.4. required_rng_sources	417
7.28.5. version	418
7.28.6. scheduling_policy	419
7.29. CLUSTERLEVEL STRUCT	419
7.30. CONFIGURATION STRUCT	420
7.30.1. data	420
7.31. CONFIGURATIONTYPE ENUM	421
7.32. CONSOLE STRUCT	422
7.33. CORE STRUCT	422
7.34. CPU STRUCT	422
7.35. CPUMODE ENUM	423
7.36. CUPROFILE STRUCT	423
7.37. CPUTOPOLOGY STRUCT	423
7.38. CPUTUNE STRUCT	424
7.39. CPUTYPE STRUCT	424
7.40. CREATIONSTATUS ENUM	424
7.41. CUSTOMPROPERTY STRUCT	424
7.42. DATACENTER STRUCT	425
7.42.1. version	425
7.43. DATACENTERSTATUS ENUM	426
7.44. DEVICE STRUCT	427
7.44.1. vms	427
7.45. DISK STRUCT	428
7.45.1. active	429
7.45.2. actual_size	429
7.45.3. bootable	429
7.45.4. initial_size	430
7.45.5. interface	430
7.45.6. provisioned_size	430
7.45.7. qcow_version	430

7.45.8. shareable	430
7.45.9. wipe_after_delete	430
7.45.10. statistics	431
7.45.11. storage_domains	432
7.45.12. vms	432
7.46. DISKATTACHMENT STRUCT	432
7.46.1. active	433
7.46.2. logical_name	433
7.46.3. read_only	433
7.46.4. uses_scsi_reservation	433
7.47. DISKFORMAT ENUM	434
7.48. DISKINTERFACE ENUM	434
7.49. DISKPROFILE STRUCT	434
7.50. DISKSNAPSHOT STRUCT	435
7.50.1. active	436
7.50.2. actual_size	437
7.50.3. bootable	437
7.50.4. initial_size	437
7.50.5. interface	437
7.50.6. provisioned_size	437
7.50.7. qcow_version	437
7.50.8. shareable	438
7.50.9. wipe_after_delete	438
7.50.10. statistics	439
7.50.11. storage_domains	439
7.50.12. vms	439
7.51. DISKSTATUS ENUM	439
7.52. DISKSTORAGETYPE ENUM	440
7.53. DISKTYPE ENUM	440
7.54. DISPLAY STRUCT	440
7.55. DISPLAYTYPE ENUM	441
7.56. DNS STRUCT	441
7.57. DNSRESOLVERCONFIGURATION STRUCT	442
7.57.1. name_servers	442
7.58. DOMAIN STRUCT	442
7.58.1. users	442
7.59. ENTITYEXTERNALSTATUS ENUM	443
7.59.1. error	443
7.59.2. failure	443
7.60. ENTITYPROFILEDETAIL STRUCT	443
7.61. ERRORHANDLING STRUCT	443
7.62. EVENT STRUCT	443
7.62.1. correlation_id	444
7.62.2. flood_rate	444
7.62.3. cluster	445
7.62.4. data_center	445
7.62.5. host	445
7.62.6. storage_domain	445
7.62.7. template	445
7.62.8. user	445
7.62.9. vm	445
7.63. EXTERNALCOMPUTERESOURCE STRUCT	445
7.64. EXTERNALDISCOVEREDHOST STRUCT	446

7.65. EXTERNALHOST STRUCT	447
7.66. EXTERNALHOSTGROUP STRUCT	447
7.67. EXTERNALHOSTPROVIDER STRUCT	448
7.67.1. requires_authentication	448
7.68. EXTERNALPROVIDER STRUCT	449
7.68.1. requires_authentication	449
7.69. EXTERNALSTATUS ENUM	450
7.70. EXTERNALSYSTEMTYPE ENUM	450
7.71. EXTERNALVMIMPORT STRUCT	450
7.71.1. url	451
7.71.2. cpu_profile	451
7.71.3. drivers_iso	451
7.71.4. host	452
7.71.5. quota	452
7.71.6. vm	452
7.72. EXTERNALVMPROVIDERTYPE ENUM	452
7.73. FAULT STRUCT	452
7.74. FENCETYPE ENUM	452
7.75. FENCINGPOLICY STRUCT	453
7.75.1. skip_if_connectivity_broken	453
7.75.2. skip_if_gluster_bricks_up	453
7.75.3. skip_if_gluster_quorum_not_met	453
7.75.4. skip_if_sd_active	454
7.76. FILE STRUCT	454
7.77. FILTER STRUCT	454
7.78. FLOPPY STRUCT	455
7.78.1. vms	455
7.79. FOPSTATISTIC STRUCT	455
7.80. GLUSTERBRICK STRUCT	456
7.80.1. vms	457
7.81. GLUSTERBRICKADVANCEDDETAILS STRUCT	457
7.81.1. vms	458
7.82. GLUSTERBRICKMEMORYINFO STRUCT	458
7.83. GLUSTERBRICKSTATUS ENUM	458
7.84. GLUSTERCLIENT STRUCT	458
7.85. GLUSTERHOOK STRUCT	459
7.86. GLUSTERHOOKSTATUS ENUM	460
7.87. GLUSTERMEMORYPOOL STRUCT	460
7.88. GLUSTERSERVERHOOK STRUCT	461
7.89. GLUSTERSTATE ENUM	461
7.90. GLUSTERVOLUME STRUCT	461
7.91. GLUSTERVOLUMEPROFILEDETAILS STRUCT	462
7.92. GLUSTERVOLUMESTATUS ENUM	463
7.93. GLUSTERVOLUMETYPE ENUM	463
7.93.1. disperse	464
7.93.2. distribute	464
7.93.3. distributed_disperse	464
7.93.4. distributed_replicate	464
7.93.5. distributed_stripe	464
7.93.6. distributed_stripped_replicate	465
7.93.7. replicate	465
7.93.8. stripe	465
7.93.9. striped_replicate	465

7.94. GRACEPERIOD STRUCT	465
7.95. GRAPHICSCONSOLE STRUCT	465
7.96. GRAPHICSTYPE ENUM	466
7.97. GROUP STRUCT	466
7.97.1. roles	467
7.98. GUESTOPERATINGSYSTEM STRUCT	467
7.99. HARDWAREINFORMATION STRUCT	468
7.100. HIGHAVAILABILITY STRUCT	468
7.100.1. priority	468
7.101. HOOK STRUCT	469
7.102. HOOKCONTENTTYPE ENUM	470
7.103. HOOKSTAGE ENUM	470
7.104. HOOKSTATUS ENUM	470
7.105. HOST STRUCT	470
7.105.1. external_status	472
7.105.2. kdump_status	473
7.105.3. ksm	473
7.105.4. libvirt_version	473
7.105.5. override_iptables	473
7.105.6. protocol	473
7.105.7. se_linux	473
7.105.8. spm	474
7.105.9. status_detail	474
7.105.10. transparent_huge_pages	474
7.105.11. version	474
7.105.12. katello_errata	475
7.105.13. statistics	476
7.106. HOSTDEVICE STRUCT	477
7.106.1. driver	478
7.107. HOSTDEVICEPASSTHROUGH STRUCT	478
7.108. HOSTNIC STRUCT	479
7.108.1. ad_aggregator_id	481
7.108.2. bridged	481
7.108.3. statistics	481
7.108.4. network	482
7.109. HOSTNICVIRTUALFUNCTIONSCONFIGURATION STRUCT	482
7.109.1. max_number_of_virtual_functions	483
7.109.2. number_of_virtual_functions	483
7.110. HOSTPROTOCOL ENUM	483
7.111. HOSTSTATUS ENUM	483
7.111.1. error	484
7.111.2. initializing	484
7.111.3. install_failed	484
7.111.4. installing_os	484
7.111.5. maintenance	485
7.111.6. non_operational	485
7.111.7. non_responsive	485
7.111.8. pending_approval	485
7.111.9. preparing_for_maintenance	485
7.112. HOSTSTORAGE STRUCT	485
7.112.1. nfs_retrans	486
7.112.2. nfs_timeo	486
7.113. HOSTTYPE ENUM	486

7.113.1. ovirt_node	487
7.114. HOSTEDENGINE STRUCT	487
7.115. ICON STRUCT	487
7.115.1. media_type	488
7.116. IDENTIFIED STRUCT	488
7.117. IMAGE STRUCT	488
7.118. IMAGETRANSFER STRUCT	489
7.118.1. direction	489
7.118.2. phase	490
7.118.3. proxy_url	490
7.119. IMAGETRANSFERDIRECTION ENUM	490
7.120. IMAGETRANSFERPHASE ENUM	490
7.120.1. cancelled	491
7.120.2. finalizing_success	491
7.120.3. finished_failure	491
7.120.4. finished_success	491
7.120.5. initializing	492
7.120.6. paused_system	492
7.120.7. resuming	492
7.120.8. unknown	492
7.121. INHERITABLEBOOLEAN ENUM	492
7.122. INITIALIZATION STRUCT	492
7.123. INSTANCETYPE STRUCT	494
7.123.1. cpu	497
7.123.2. custom_compatibility_version	497
7.123.3. high_availability	497
7.123.4. large_icon	498
7.123.5. lease	498
7.123.6. memory	498
7.123.7. migration_downtime	498
7.123.8. origin	498
7.123.9. small_icon	499
7.123.10. sso	499
7.124. IO STRUCT	500
7.125. IP STRUCT	500
7.125.1. address	500
7.125.2. netmask	501
7.125.3. version	501
7.126. IPADDRESSASSIGNMENT STRUCT	501
7.127. IPVERSION ENUM	501
7.128. ISCSIBOND STRUCT	501
7.129. ISCSIDetails STRUCT	502
7.130. JOB STRUCT	503
7.130.1. external	504
7.131. JOBSTATUS ENUM	504
7.131.1. aborted	504
7.131.2. finished	504
7.131.3. started	504
7.131.4. unknown	504
7.132. KATELLOERRATUM STRUCT	504
7.132.1. severity	505
7.132.2. type	505
7.133. KDUMPSTATUS ENUM	506

7.134. KERNEL STRUCT	506
7.135. KSM STRUCT	506
7.136. LOGSEVERITY ENUM	506
7.136.1. alert	507
7.136.2. error	507
7.136.3. normal	507
7.136.4. warning	507
7.137. LOGICALUNIT STRUCT	507
7.137.1. discard_max_size	508
7.137.2. discard_zeroes_data	508
7.138. LUNSTATUS ENUM	508
7.139. MAC STRUCT	508
7.140. MACPOOL STRUCT	509
7.140.1. allow_duplicates	509
7.140.2. default_pool	509
7.140.3. ranges	510
7.141. MEMORYOVERCOMMIT STRUCT	510
7.142. MEMORYPOLICY STRUCT	510
7.142.1. max	510
7.143. MESSAGEBROKERTYPE ENUM	510
7.144. METHOD STRUCT	511
7.145. MIGRATEONERROR ENUM	511
7.146. MIGRATIONBANDWIDTH STRUCT	511
7.146.1. custom_value	511
7.147. MIGRATIONBANDWIDTHASSIGNMENTMETHOD ENUM	511
7.147.1. auto	512
7.148. MIGRATIONOPTIONS STRUCT	512
7.149. MIGRATIONPOLICY STRUCT	512
7.150. NETWORK STRUCT	513
7.150.1. usages	514
7.151. NETWORKATTACHMENT STRUCT	515
7.151.1. dns_resolver_configuration	517
7.151.2. properties	518
7.152. NETWORKCONFIGURATION STRUCT	519
7.153. NETWORKFILTER STRUCT	519
7.153.1. version	520
7.154. NETWORKLABEL STRUCT	520
7.155. NETWORKPLUGINTYPE ENUM	521
7.156. NETWORKSTATUS ENUM	521
7.157. NETWORKUSAGE ENUM	521
7.157.1. management	521
7.158. NFSPROFILEDETAIL STRUCT	522
7.159. NFSVERSION ENUM	522
7.159.1. v4_2	522
7.160. NIC STRUCT	522
7.160.1. network	524
7.160.2. vms	524
7.161. NICCONFIGURATION STRUCT	524
7.162. NICINTERFACE ENUM	524
7.163. NICSTATUS ENUM	525
7.164. NUMANODE STRUCT	525
7.164.1. statistics	526
7.165. NUMANODEPIN STRUCT	527

7.165.1. host_numa_node	527
7.165.2. pinned	527
7.166. NUMATUNEMODE ENUM	528
7.167. OPENSTACKIMAGE STRUCT	528
7.168. OPENSTACKIMAGEPROVIDER STRUCT	528
7.168.1. requires_authentication	529
7.169. OPENSTACKNETWORK STRUCT	529
7.170. OPENSTACKNETWORKPROVIDER STRUCT	530
7.170.1. read_only	531
7.170.2. requires_authentication	531
7.171. OPENSTACKNETWORKPROVIDERTYPE ENUM	531
7.171.1. external	531
7.171.2. neutron	531
7.172. OPENSTACKPROVIDER STRUCT	532
7.172.1. requires_authentication	532
7.173. OPENSTACKSUBNET STRUCT	532
7.173.1. ip_version	533
7.174. OPENSTACKVOLUMEPROVIDER STRUCT	533
7.174.1. requires_authentication	534
7.175. OPENSTACKVOLUMETYPE STRUCT	534
7.176. OPENSTACKVOLUMEAUTHENTICATIONKEY STRUCT	535
7.177. OPENSTACKVOLUMEAUTHENTICATIONKEYUSAGETYPE ENUM	535
7.178. OPERATINGSYSTEM STRUCT	536
7.178.1. custom_kernel_cmdline	536
7.178.2. reported_kernel_cmdline	536
7.179. OPERATINGSYSTEMINFO STRUCT	537
7.180. OPTION STRUCT	537
7.181. OSTYPE ENUM	537
7.182. PACKAGE STRUCT	539
7.183. PAYLOAD STRUCT	539
7.184. PAYLOADENCODING ENUM	539
7.185. PERMISSION STRUCT	539
7.186. PERMIT STRUCT	540
7.187. PMPROXY STRUCT	541
7.188. PMPROXYTYPE ENUM	541
7.189. POLICYUNITTYPE ENUM	541
7.190. PORTMIRRORING STRUCT	542
7.191. POWERMANAGEMENT STRUCT	542
7.191.1. agents	542
7.191.2. automatic_pm_enabled	543
7.191.3. kdump_detection	543
7.191.4. type	543
7.192. POWERMANAGEMENTSTATUS ENUM	543
7.193. PRODUCT STRUCT	543
7.194. PRODUCTINFO STRUCT	543
7.194.1. vendor	544
7.195. PROFILEDETAIL STRUCT	544
7.196. PROPERTY STRUCT	545
7.197. PROXYTICKET STRUCT	545
7.198. QCOWVERSION ENUM	545
7.198.1. qcow2_v3	545
7.199. QOS STRUCT	545
7.199.1. cpu_limit	547

7.199.2. inbound_average	547
7.199.3. inbound_burst	547
7.199.4. inbound_peak	548
7.199.5. max_iops	548
7.199.6. max_read_iops	548
7.199.7. max_read_throughput	548
7.199.8. max_throughput	548
7.199.9. max_write_iops	548
7.199.10. max_write_throughput	548
7.199.11. outbound_average	548
7.199.12. outbound_average_linkshare	549
7.199.13. outbound_average_realttime	549
7.199.14. outbound_average_upperlimit	549
7.199.15. outbound_burst	549
7.199.16. outbound_peak	549
7.200. QOSTYPE ENUM	549
7.201. QUOTA STRUCT	550
7.202. QUOTA_CLUSTERLIMIT STRUCT	551
7.203. QUOTA_MODETYPE ENUM	552
7.204. QUOTA_STORAGELIMIT STRUCT	552
7.205. RANGE STRUCT	553
7.206. RATE STRUCT	553
7.207. REPORTEDCONFIGURATION STRUCT	553
7.208. REPORTEDDEVICE STRUCT	553
7.209. REPORTEDDEVICETYPE ENUM	554
7.210. RESOLUTIONTYPE ENUM	554
7.211. RNGDEVICE STRUCT	554
7.212. RNGSOURCE ENUM	555
7.212.1. urandom	555
7.213. ROLE STRUCT	555
7.213.1. mutable	556
7.214. ROLETYPE ENUM	556
7.215. SCHEDULINGPOLICY STRUCT	556
7.216. SCHEDULINGPOLICYUNIT STRUCT	557
7.217. SCSIGENERICIO ENUM	557
7.218. SELINUX STRUCT	558
7.219. SELINUXMODE ENUM	558
7.220. SERIALNUMBER STRUCT	558
7.221. SERIALNUMBERPOLICY ENUM	558
7.222. SESSION STRUCT	559
7.222.1. console_user	559
7.222.2. ip	559
7.222.3. protocol	559
7.222.4. user	560
7.223. SKIPIFCONNECTIVITYBROKEN STRUCT	560
7.223.1. enabled	560
7.223.2. threshold	560
7.224. SKIPIFSDACTIVE STRUCT	560
7.224.1. enabled	560
7.225. SNAPSHOT STRUCT	561
7.225.1. cpu	564
7.225.2. custom_compatibility_version	565
7.225.3. high_availability	565

7.225.4. large_icon	565
7.225.5. lease	565
7.225.6. memory	565
7.225.7. migration_downtime	566
7.225.8. next_run_configuration_exists	566
7.225.9. origin	566
7.225.10. persist_memorystate	566
7.225.11. placement_policy	567
7.225.12. small_icon	567
7.225.13. sso	567
7.225.14. stop_reason	567
7.225.15. affinity_labels	569
7.225.16. katello_errata	569
7.225.17. original_template	570
7.226. SNAPSHOTSTATUS ENUM	570
7.227. SNAPSHOTTYPE ENUM	570
7.228. SPECIALOBJECTS STRUCT	570
7.229. SPM STRUCT	571
7.230. SPMSTATUS ENUM	571
7.231. SSH STRUCT	571
7.232. SSHAUTHENTICATIONMETHOD ENUM	572
7.233. SSHPUBLICKEY STRUCT	572
7.234. SSO STRUCT	572
7.235. SSOMETHOD ENUM	573
7.236. STATISTIC STRUCT	573
7.237. STATISTICKIND ENUM	575
7.238. STATISTICUNIT ENUM	575
7.239. STEP STRUCT	575
7.239.1. external	576
7.240. STEPENUM ENUM	576
7.240.1. executing	577
7.240.2. finalizing	577
7.240.3. rebalancing_volume	577
7.240.4. removing_bricks	577
7.240.5. unknown	577
7.240.6. validating	577
7.241. STEPSTATUS ENUM	577
7.241.1. aborted	578
7.241.2. finished	578
7.241.3. started	578
7.241.4. unknown	578
7.242. STORAGECONNECTION STRUCT	578
7.243. STORAGECONNECTIONEXTENSION STRUCT	579
7.244. STORAGEDOMAIN STRUCT	580
7.244.1. discard_after_delete	582
7.244.2. supports_discard	582
7.244.3. supports_discard_zeroes_data	582
7.244.4. wipe_after_delete	583
7.244.5. data_center	583
7.245. STORAGEDOMAINLEASE STRUCT	583
7.246. STORAGEDOMAINSTATUS ENUM	584
7.247. STORAGEDOMAINTYPE ENUM	584
7.248. STORAGEFORMAT ENUM	585

7.249. STORAGETYPE ENUM	585
7.249.1. cinder	586
7.249.2. glance	586
7.249.3. glusterfs	586
7.250. SWITCHTYPE ENUM	586
7.251. TAG STRUCT	586
7.252. TEMPLATE STRUCT	587
7.252.1. cpu	589
7.252.2. custom_compatibility_version	590
7.252.3. high_availability	590
7.252.4. large_icon	590
7.252.5. lease	590
7.252.6. memory	590
7.252.7. migration_downtime	591
7.252.8. origin	591
7.252.9. small_icon	591
7.252.10. sso	591
7.253. TEMPLATESTATUS ENUM	592
7.254. TEMPLATEVERSION STRUCT	592
7.254.1. version_number	593
7.255. TICKET STRUCT	593
7.256. TIMEZONE STRUCT	593
7.256.1. utc_offset	593
7.257. TRANSPARENTHUGEPAGES STRUCT	594
7.258. TRANSPORTTYPE ENUM	594
7.259. UNMANAGEDNETWORK STRUCT	594
7.260. USB STRUCT	594
7.261. USBTYPE ENUM	595
7.261.1. legacy	595
7.262. USER STRUCT	595
7.262.1. namespace	596
7.262.2. principal	596
7.262.3. user_name	596
7.263. VALUE STRUCT	597
7.264. VALUETYPE ENUM	597
7.265. VCPUPIN STRUCT	597
7.266. VENDOR STRUCT	597
7.267. VERSION STRUCT	598
7.268. VIRTIOSCSI STRUCT	598
7.269. VIRTUALNUMANODE STRUCT	598
7.269.1. statistics	600
7.270. VLAN STRUCT	601
7.271. VM STRUCT	601
7.271.1. cpu	604
7.271.2. custom_compatibility_version	605
7.271.3. high_availability	605
7.271.4. large_icon	605
7.271.5. lease	605
7.271.6. memory	605
7.271.7. migration_downtime	606
7.271.8. next_run_configuration_exists	606
7.271.9. origin	606
7.271.10. placement_policy	606

7.271.11. small_icon	607
7.271.12. sso	607
7.271.13. stop_reason	607
7.271.14. affinity_labels	609
7.271.15. katello_errata	609
7.271.16. original_template	609
7.272. VMAFFINITY ENUM	610
7.273. VMBASE STRUCT	610
7.273.1. cpu	612
7.273.2. custom_compatibility_version	613
7.273.3. high_availability	613
7.273.4. large_icon	613
7.273.5. lease	613
7.273.6. memory	613
7.273.7. migration_downtime	614
7.273.8. origin	614
7.273.9. small_icon	614
7.273.10. sso	614
7.274. VMDEVICETYPE ENUM	615
7.275. VMPLACEMENTPOLICY STRUCT	615
7.276. VMPOOL STRUCT	615
7.276.1. auto_storage_select	616
7.276.2. stateful	617
7.276.3. instance_type	617
7.277. VMPOOLTYPE ENUM	617
7.278. VMSTATUS ENUM	617
7.278.1. paused	618
7.278.2. powering_up	619
7.278.3. restoring_state	619
7.278.4. saving_state	619
7.278.5. suspended	619
7.278.6. unknown	619
7.278.7. up	619
7.278.8. wait_for_launch	619
7.279. VMSUMMARY STRUCT	619
7.280. VMTYPE ENUM	620
7.280.1. desktop	620
7.280.2. server	620
7.281. VNICPASSTHROUGH STRUCT	620
7.282. VNICPASSTHROUGHMODE ENUM	620
7.283. VNICPROFILE STRUCT	620
7.283.1. migratable	621
7.283.2. pass_through	621
7.283.3. port_mirroring	621
7.283.4. network_filter	622
7.283.5. qos	622
7.284. VNICPROFILEMAPPING STRUCT	622
7.285. VOLUMEGROUP STRUCT	623
7.286. WATCHDOG STRUCT	624
7.286.1. model	624
7.286.2. vms	624
7.287. WATCHDOGACTION ENUM	624
7.287.1. none	625

7.288. WATCHDOGMODEL ENUM	625
7.289. WEIGHT STRUCT	625
APPENDIX A. PRIMITIVE TYPES	627
A.1. STRING PRIMITIVE	627
A.2. BOOLEAN PRIMITIVE	627
A.3. INTEGER PRIMITIVE	627
A.4. DECIMAL PRIMITIVE	628
A.5. DATE PRIMITIVE	628
APPENDIX B. CHANGES IN VERSION 4 OF THE API	630
B.1. REMOVED YAML SUPPORT	630
B.2. RENAMED COMPLEX TYPES	630
B.3. REPLACED THE STATUS TYPE WITH ENUM TYPES	631
B.4. REMOVE THE NIC NETWORK AND PORT_MIRRORING PROPERTIES	632
B.5. REMOVE THE NIC ACTIVE PROPERTY	633
B.6. REMOVE THE DISK TYPE PROPERTY	633
B.7. REMOVE THE DISK SIZE PROPERTY	633
B.8. REMOVED SUPPORT FOR PINNING A VM TO A SINGLE HOST	633
B.9. REMOVED THE CAPABILITIES.PERMITS ELEMENT	633
B.10. REMOVED THE STORAGE_MANAGER ELEMENT	634
B.11. REMOVED THE DATA CENTER STORAGE_TYPE ELEMENT	634
B.12. REMOVE THE TIMEZONE ELEMENT	634
B.13. REMOVED THE GUEST_INFO ELEMENT	635
B.14. REPLACED CPU ID ATTRIBUTE WITH TYPE ELEMENT	636
B.15. USE ELEMENTS INSTEAD OF ATTRIBUTES IN CPU TOPOLOGY	636
B.16. USE ELEMENTS INSTEAD OF ATTRIBUTES IN VCPU PIN	636
B.17. USE ELEMENTS INSTEAD OF ATTRIBUTES IN VCPU PIN	637
B.18. USE ELEMENTS INSTEAD OF ATTRIBUTES IN MEMORY OVERCOMMIT	637
B.19. USE ELEMENTS INSTEAD OF ATTRIBUTES IN CONSOLE	637
B.20. USE ELEMENTS INSTEAD OF ATTRIBUTES IN VIRTIO SCSI	638
B.21. USE ELEMENT INSTEAD OF ATTRIBUTE FOR POWER MANAGEMENT AGENT TYPE	638
B.22. USE ELEMENTS INSTEAD OF ATTRIBUTES IN POWER MANAGEMENT AGENT OPTIONS	638
B.23. USE ELEMENTS INSTEAD OF ATTRIBUTES IN IP ADDRESS:	639
B.24. USE ELEMENTS INSTEAD OF ATTRIBUTES IN MAC ADDRESS:	639
B.25. USE ELEMENTS INSTEAD OF ATTRIBUTES IN BOOT DEVICE:	639
B.26. USE ELEMENT INSTEAD OF ATTRIBUTE FOR OPERATING SYSTEM TYPE	639
B.27. REMOVED THE FORCE PARAMETER FROM THE REQUEST TO RETRIEVE A HOST	640
B.28. REMOVED DEPRECATED HOST POWER MANAGEMENT CONFIGURATION	640
B.29. USE MULTIPLE BOOT.DEVICES.DEVICE INSTEAD OF MULTIPLE BOOT	641
B.30. REMOVED THE DISKS.CLONE AND DISKS.DETACH_ONLY ELEMENTS	641
B.31. RENAME ELEMENT VMPOOL TO VM_POOL	642
B.32. USE LOGICAL_UNITS INSTEAD OF MULTIPLE LOGICAL_UNIT	642
B.33. REMOVED THE SNAPSHOTS.COLLAPSE_SNAPSHOTS ELEMENT	643
B.34. RENAMED STORAGE AND HOST_STORAGE ELEMENTS	644
B.35. REMOVED THE PERMISSIONS.CLONE ELEMENT	644
B.36. RENAMED THE RANDOM NUMBER GENERATOR SOURCE ELEMENTS	645
B.37. REMOVED THE INTERMEDIATE TAG.PARENT ELEMENT	646
B.38. REMOVE SCHEDULING BUILT-IN NAMES AND THRESHOLDS	647
B.39. REMOVED THE BRICKS.REPLICA_COUNT AND BRICKS.STRIPE_COUNT ELEMENTS	648
B.40. RENAMED THE STATISTICS TYPE PROPERTY TO KIND	648
B.41. USE MULTIPLE VCPU_PINS.VCPU_PIN INSTEAD OF MULTIPLE VCPU_PIN	648
B.42. USE FORCE PARAMETER TO FORCE REMOVE A DATA CENTER	649

B.43. USE FORCE PARAMETER TO FORCE REMOVE A HOST	649
B.44. USE PARAMETERS FOR FORCE REMOVE STORAGE DOMAIN	650
B.45. USE HOST PARAMETER TO REMOVE STORAGE SERVER CONNECTION	650
B.46. USE FORCE AND STORAGE_DOMAIN PARAMETERS TO REMOVE TEMPLATE DISKS	651
B.47. DON'T REMOVE DISKS VIA THE VM DISK API	651
B.48. USE FORCE QUERY PARAMETER TO FORCE REMOVE A VIRTUAL MACHINE	651
B.49. USE POST INSTEAD OF DELETE TO REMOVE MULTIPLE BRICKS	651
B.50. REMOVED THE SCHEDULING_POLICY.POLICY ELEMENT	652
B.51. ADDED SNAPSHOT.SNAPSHOT_TYPE	652
B.52. REMOVED MOVE ACTION FROM VM	653
B.53. MOVED REPORTED_CONFIGURATIONS.IN_SYNC TO NETWORK_ATTACHMENT	653
B.54. REPLACED CAPABILITIES WITH CLUSTERLEVELS	653
B.55. REPLACED DISKS WITH DISKATTACHMENTS	654
B.56. USE ISCSI_TARGETS ELEMENT TO DISCOVER UNREGISTERED STORAGE	656

CHAPTER 1. INTRODUCTION

The Red Hat Virtualization Manager provides a **Representational State Transfer (REST) API**. The API provides software developers and system administrators with control over their Red Hat Virtualization environment outside of the standard web interface. The API is useful for developers and administrators to integrate the functionality of a Red Hat Virtualization environment with custom scripts or external applications that access the API via the standard Hypertext Transfer Protocol (HTTP).

The benefits of the API are:

- Broad client support - Any programming language, framework, or system with support for HTTP protocol can use the API.
- Self descriptive - Client applications require minimal knowledge of the virtualization infrastructure, as many details are discovered at runtime.
- Resource-based model - The resource-based REST model provides a natural way to manage a virtualization platform.

This provides developers and administrators with the ability to:

- Integrate with enterprise IT systems.
- Integrate with third-party virtualization software.
- Perform automated maintenance or error-checking tasks.
- Automate repetitive tasks in a Red Hat Virtualization environment with scripts.

This documentation acts as a reference for the Red Hat Virtualization API. It aims to provide developers and administrators with instructions and examples to help harness the functionality of their Red Hat Virtualization environment through the API, either directly or using the provided SDKs.

1.1. REPRESENTATIONAL STATE TRANSFER

Representational State Transfer (REST) is a design architecture that focuses on resources for a specific service and their representations. A resource representation is a key abstraction of information that corresponds to one specific managed element on a server. A client sends a request to a server element located at a Uniform Resource Identifier (URI) and performs operations with standard HTTP methods, such as **GET**, **POST**, **PUT**, and **DELETE**. This provides a stateless communication between the client and server where each request acts independently of any other request, and contains all the information necessary to complete the request.

1.2. API PREREQUISITES

Prerequisites for using the Red Hat Virtualization API:

- A networked installation of Red Hat Virtualization Manager, which includes the API.
- A client or programming library that initiates and receives HTTP requests from the API server. For example:
 - The [oVirt Python SDK](#).
 - The [oVirt Ruby SDK](#).

- The [oVirt Java SDK](#).
- The [cURL](#) command line tool.
- [RESTClient](#), a debugger for RESTful web services.
- Knowledge of Hypertext Transfer Protocol (HTTP), the protocol used for REST API interactions. The Internet Engineering Task Force provides a Request for Comments (RFC) explaining the Hypertext Transfer Protocol at <http://www.ietf.org/rfc/rfc2616.txt>.
- Knowledge of Extensible Markup Language (XML) or JavaScript Object Notation (JSON), which the API uses to construct resource representations. The W3C provides a full specification on XML at <http://www.w3.org/TR/xml>. ECMA International provide a free publication on JSON at <http://www.ecma-international.org>.

CHAPTER 2. AUTHENTICATION AND SECURITY

2.1. TLS/SSL CERTIFICATION

The Red Hat Virtualization API requires Hypertext Transfer Protocol Secure (HTTPS) ^[1] for secure interaction with client software, such as the SDK and CLI components. This involves obtaining the CA certificate used by the server, and importing it into the certificate store of your client.

2.1.1. Obtaining the CA Certificate

You can obtain the CA certificate from the Red Hat Virtualization Manager and transfer it to the client machine using one of these methods:

Method 1

The preferred method for obtaining the CA certificate is to use the `openssl s_client` command line tool to perform a real TLS handshake with the server, and then extract the certificates that it presents. Run a command like this:

```
$ openssl s_client \
  -connect myengine.example.com:443 \
  -showcerts \
  < /dev/null
```

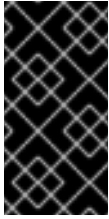
This command will connect to the server and display output similar to the following:

```
CONNECTED(00000003)
depth=1 C = US, O = Example Inc., CN = myengine.example.com.23416
verify error:num=19:self signed certificate in certificate chain
---
Certificate chain
 0 s:/C=US/O=Example Inc./CN=myengine.example.com
  i:/C=US/O=Example Inc./CN=myengine.example.com.23416
-----BEGIN CERTIFICATE-----
MIIEEaTCCA1GgAwIBAgICEAQwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMVCVVMx
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW51NDEuZXhhbXBs
SV1Je7e5FTEtHJGTAewWM6dGbsFhip5VXM0gfqq=
-----END CERTIFICATE-----
 1 s:/C=US/O=Example Inc./CN=myengine.example.com.23416
  i:/C=US/O=Example Inc./CN=myengine.example.com.23416
-----BEGIN CERTIFICATE-----
MIIDxjCCAq6gAwIBAgICEAAwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMVCVVMx
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW51NDEuZXhhbXBs
Pkyg1rQHR6ebGQ==
-----END CERTIFICATE-----
```

The text between the `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----` marks shows the certificates presented by the server. The first one is the certificate of the server itself, and the last one is the certificate of the CA. Copy the CA certificate, including the marks, to the `ca.crt` file. The result should look like this:

```
-----BEGIN CERTIFICATE-----
MIIDxjCCAq6gAwIBAgICEAAwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMVCVVMx
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW51NDEuZXhhbXBs
```

```
Pkyg1rQHR6ebGQ==
-----END CERTIFICATE-----
```



IMPORTANT

This is the most reliable method to obtain the CA certificate used by the server. The rest of the methods described here will work in most cases, but they will not obtain the correct CA certificate if it has been manually replaced by the administrator of the server.

Method 2

If you cannot use the `openssl s_client` method described above, you can instead use a command line tool to download the CA certificate from the Red Hat Virtualization Manager. Examples of command line tools include `curl` and `wget`, both of which are available on multiple platforms.

If using `curl`:

```
$ curl \
--output ca.crt \
'http://myengine.example.com/ovirt-engine/services/pki-resource?
resource=ca-certificate&format=X509-PEM-CA'
```

If using `wget`:

```
$ wget \
--output-document ca.crt \
'http://myengine.example.com/ovirt-engine/services/pki-resource?
resource=ca-certificate&format=X509-PEM-CA'
```

Method 3

Use a web browser to navigate to the certificate located at:

```
https://myengine.example.com/ovirt-engine/services/pki-resource?
resource=ca-certificate&format=X509-PEM-CA
```

Depending on the chosen browser, the certificate either downloads or imports into the browser's keystore.

1. **If the browser downloads the certificate:** save the file as `ca.crt`.
2. **If the browser imports the certificate:** export it from the browser's certification options and save it as `ca.crt`.

Method 4

Log in to the Red Hat Virtualization Manager, export the certificate from the truststore, and copy it to your client machine.

1. Log in to the Red Hat Virtualization Manager machine as `root`.
2. Export the certificate from the truststore using the Java `keytool` management utility:

```
# keytool \
-keystore /etc/pki/ovirt-engine/.truststore \
-storepass mypass \
-exportcert \
-alias cacert \
-rfc \
-file ca.crt
```

This creates a certificate file called **ca.crt**.

3. Copy the certificate to the client machine using the **scp** command:

```
$ scp ca.crt myuser@myclient.example.com:/home/myuser/.
```

Each of these methods results in a certificate file named **ca.crt** on your client machine. You must then import this file into the certificate store of the client.

2.1.2. Importing a Certificate to a Client

Importing a certificate to a client relies on how the client stores and interprets certificates. See your client documentation for more information on importing a certificate.

2.2. AUTHENTICATION

Any user with a Red Hat Virtualization Manager account has access to the API. All requests must be authenticated using either **OAuth** or basic authentication, as described below.

2.2.1. OAuth Authentication

Since version 4.0 of Red Hat Virtualization the preferred authentication mechanism is [OAuth 2.0](#), as described in [RFC 6749](#).

OAuth is a sophisticated protocol, with several mechanisms for obtaining authorization and access tokens. For use with the Red Hat Virtualization API, the only supported one is the *Resource Owner Password Credentials Grant*, as described in [section 4.3](#) of RFC 6749.

You must first obtain a *token*, sending the user name and password to the Red Hat Virtualization Manager single sign-on service:

```
POST /ovirt-engine/sso/oauth/token HTTP/1.1
Host: myengine.example.com
Content-Type: application/x-www-form-urlencoded
Accept: application/json
```

The request body must contain the **grant_type**, **scope**, **username**, and **password** parameters:

Table 2.1. OAuth token request parameters

Name	Value
grant_type	password

Name	Value
scope	ovirt-app-api
username	admin@internal
password	mypassword

These parameters must be [URL-encoded](#). For example, the @ character in the user name needs to be encoded as %40. The resulting request body will be something like this:

```
grant_type=password&scope=ovirt-app-api&username=admin%40internal&password=mypassword
```



IMPORTANT

The **scope** parameter is described as optional in the **OAuth** RFC, but when using it with the Red Hat Virtualization API it is mandatory, and its value must be **ovirt-app-api**.

If the user name and password are valid, the Red Hat Virtualization Manager single sign-on service will respond with a JSON document similar to this one:

```
{
  "access_token": "fqBR1ftzh8wBCviLxJcYuV5oSDI=",
  "token_type": "bearer",
  "scope": "...",
  ...
}
```

For API authentication purposes, the only relevant name/value pair is the **access_token**. Do not manipulate this in any way; use it exactly as provided by the SSO service.

Once the token has been obtained, it can be used to perform requests to the API by including it in the HTTP **Authorization** header, and using the **Bearer** scheme. For example, to get the list of virtual machines, send a request like this:

```
GET /ovirt-engine/api/vms HTTP/1.1
Host: myengine.example.com
Accept: application/xml
Authorization: Bearer fqBR1ftzh8wBCviLxJcYuV5oSDI=
```

The token can be used multiple times, for multiple requests, but it will eventually expire. When it expires, the server will reject the request with the 401 HTTP response code:

```
HTTP/1.1 401 Unauthorized
```

When this happens, a new token is needed, as the Red Hat Virtualization Manager single sign-on service does not currently support refreshing tokens. A new token can be requested using the same method described above.

2.2.2. Basic Authentication



IMPORTANT

Basic authentication is supported only for backwards compatibility; it is deprecated since version 4.0 of Red Hat Virtualization, and will be removed in the future.

Each request uses HTTP Basic Authentication [2] to encode the credentials. If a request does not include an appropriate **Authorization** header, the server sends a **401 Authorization Required** response:

```
HEAD /ovirt-engine/api HTTP/1.1
Host: myengine.example.com

HTTP/1.1 401 Authorization Required
```

Request are issued with an **Authorization** header for the specified realm. Encode an appropriate Red Hat Virtualization Manager domain and user in the supplied credentials with the **username@domain:password** convention.

The following table shows the process for encoding credentials in [Base64](#).

Table 2.2. Encoding credentials for API access

Item	Value
User name	admin
Domain	internal
Password	mypassword
Unencoded credentials	admin@internal:mypassword
Base64 encoded credentials	YWRtaW5AaW50ZXJuYWw6bXlwYXNzd29yZA==

Provide the Base64-encoded credentials as shown:

```
HEAD /ovirt-engine/api HTTP/1.1
Host: myengine.example.com
Authorization: Basic YWRtaW5AaW50ZXJuYWw6bXlwYXNzd29yZA==

HTTP/1.1 200 OK
```



IMPORTANT

Basic authentication involves potentially sensitive information, such as passwords, sent as plain text. The API requires Hypertext Transfer Protocol Secure (HTTPS) for transport-level encryption of plain-text requests.



IMPORTANT

Some Base64 libraries break the result into multiple lines and terminate each line with a newline character. This breaks the header and causes a faulty request. The **Authorization** header requires the encoded credentials on a single line within the header.

2.2.3. Authentication Sessions

The API also provides authentication session support. Send an initial request with authentication details, then send all subsequent requests using a session cookie to authenticate.

2.2.3.1. Requesting an Authenticated Session

1. Send a request with the **Authorization** and **Prefer: persistent-auth** headers:

```
HEAD /ovirt-engine/api HTTP/1.1
Host: myengine.example.com
Authorization: Basic YWRtaW5AaW50ZXJuYWw6bXlwYXNzd29yZA==
Prefer: persistent-auth

HTTP/1.1 200 OK
...
```

This returns a response with the following header:

```
Set-Cookie: JSESSIONID=5dQja5ubr4yvI2MM2z+LZxrK; Path=/ovirt-
engine/api; Secure
```

Take note of the **JSESSIONID=** value. In this example the value is **5dQja5ubr4yvI2MM2z+LZxrK**.

2. Send all subsequent requests with the **Prefer: persistent-auth** and **Cookie** headers with the **JSESSIONID=** value. The **Authorization** header is no longer needed when using an authenticated session.

```
HEAD /ovirt-engine/api HTTP/1.1
Host: myengine.example.com
Prefer: persistent-auth
Cookie: JSESSIONID=5dQja5ubr4yvI2MM2z+LZxrK

HTTP/1.1 200 OK
...
```

3. When the session is no longer required, perform a request to the sever without the **Prefer: persistent-auth** header.

```
HEAD /ovirt-engine/api HTTP/1.1
Host: myengine.example.com
Authorization: Basic YWRtaW5AaW50ZXJuYWw6bXlwYXNzd29yZA==

HTTP/1.1 200 OK
...
```

[1] HTTPS is described in [RFC 2818 HTTP Over TLS](#).

[2] Basic Authentication is described in [RFC 2617 HTTP Authentication: Basic and Digest Access Authentication](#).

CHAPTER 3. COMMON CONCEPTS

3.1. TYPES

The API uses the *type* concept to describe the different kinds of objects accepted and returned.

There are three relevant kinds of types:

Primitive types

Describe simple kinds of objects, like [strings](#) or [integers](#).

Enumerated types

Describe lists of valid values like [VmStatus](#) or [DiskFormat](#).

Structured types

Describe structured objects, with multiple attributes and links, like [Vm](#) or [Disk](#).

3.2. IDENTIFIED TYPES

Many of the types used by the API represent *identified* objects, objects that have a unique identifier and exist independently of other objects. The types used to describe those objects extend the [Identified](#) type, which contains the following set of common attributes:

Attribute	Type	Description
id	String	Each object in the virtualization infrastructure contains an id , which acts as a unique identifier.
href	String	The canonical location of the object as an absolute path.
name	String	A user-supplied human readable name for the object. The name name is unique across all objects of the same type.
description	String	A free-form user-supplied human readable description of the object.



IMPORTANT

Currently for most types of objects the **id** attribute is actually a randomly generated [UUID](#), but this is an implementation detail, and users should not rely on that, as it may change in the future. Instead users should assume that these identifiers are just strings.

3.3. OBJECTS

Objects are the individual instances of the types supported by the API. For example, the virtual machine with identifier **123** is an object of the [Vm](#) type.

3.4. COLLECTIONS

A collection is a set of objects of the same type.

3.5. REPRESENTATIONS

The state of objects needs to be represented when it is transferred between the client and the server. The API supports XML and JSON as the representation of the state of objects, both for input and output.

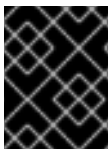
3.5.1. XML representation

The XML representation of an object consists of an XML element corresponding to the type of the object, XML attributes for the **id** and **href** attributes, and nested XML elements for the rest of the attributes. For example, the XML representation for a virtual machine appears as follows:

```
<vm id="123" href="/ovirt-engine/api/vms/123">
  <name>myvm</name>
  <description>My VM</description>
  <memory>1073741824</memory>
  ...
</vm>
```

The XML representation of a collection of objects consists of an XML element, named after the type of the objects, in plural. This contains the representations of the objects of the collection. For example, the XML representation for a collection of virtual machines appears as follows:

```
<vms>
  <vm id="123" href="/ovirt-engine/api/vms/123">
    <name>yourvm</name>
    <description>Your VM</description>
    <memory>1073741824</memory>
    ...
  </vm>
  <vm id="456" href="/ovirt-engine/api/vms/456">
    <name>myname</name>
    <description>My description</description>
    <memory>2147483648</memory>
    ...
  </vm>
  ...
</vms>
```



IMPORTANT

In the XML representation of objects the **id** and **href** attributes are the only ones that are represented as XML attributes, the rest are represented as nested XML elements.

3.5.2. JSON representation

The JSON representation of an object consists of a JSON document containing a name/value pair for each attribute (including **id** and **href**). For example, the JSON representation of a virtual machine appears as follows:

```
{
  "id": "123",
  "href": "/ovirt-engine/api/vms/123",
  "name": "myvm",
  "description": "My VM",
```

```

    "memory": 1073741824,
    ...
  }

```

The JSON representation of a collection of objects consists of a JSON document containing a name/value pair (named after the type of the objects, in singular) which in turn contains an array with the representations of the objects of the collection. For example, the JSON representation for a collection of virtual machines appears as follows:

```

{
  "vm": [
    {
      "id": "123",
      "href": "/ovirt-engine/api/vms/123",
      "name": "myvm",
      "description": "My VM",
      "memory": 1073741824,
      ...
    },
    {
      "id": "456",
      "href": "/ovirt-engine/api/vms/456",
      "name": "yourvm",
      "description": "Your VM",
      "memory": 2147483648,
      ...
    },
  ],
}

```

3.6. SERVICES

Services are the parts of the server responsible for retrieving, adding, updating, removing and executing actions on the objects supported by the API.

There are two relevant kinds of services:

Services that manage a collection of objects

These services are responsible for listing existing objects and adding new objects. For example, the [Vms](#) service is responsible for managing the collection of virtual machines available in the system.

Services that manage a specific object

These services are responsible for retrieving, updating, deleting and executing actions in specific objects. For example, the [Vm](#) service is responsible for managing a specific virtual machine.

Each service is accessible via a particular *path* within the server. For example, the service that manages the collection of virtual machines available in the system is available via the path `/vms`, and the service that manages the virtual machine `123` is available via the path `/vms/123`.

All kinds of services have a set of *methods* that represent the operations that they can perform. The services that manage collections of objects usually have the **list** and **add** methods. The services that manage specific objects usually have the **get**, **update** and **remove** methods. In addition, services may also have *action* methods, that represent less common operations. For example, the [Vm](#) service has a [start](#) method that is used to start a virtual machine.

For the more usual methods there is a direct mapping between the name of the method and the name of the HTTP method:

Method name	HTTP method
add	POST
get	GET
list	GET
update	PUT
remove	DELETE

The path used in the HTTP request is the path of the service, with the **/ovirt-engine/api** prefix.

For example, the request to **list** the virtual machines should be like this, using the HTTP**GET** method and the path **/vms**:

```
GET /ovirt-engine/api/vms
```

For action methods the HTTP method is always **POST**, and the name of the method is added as a suffix to the path. For example, the request to start virtual machine **123** should look like this, using the HTTP **POST** method and the path **/vms/123/start**:

```
POST /ovirt-engine/api/vms/123/start
```

Each method has a set of parameters.

Parameters are classified into two categories:

Main parameter

The main parameter corresponds the object or collection that is retrieved, added or updated. This only applies to the **add**, **get**, **list** and **update** methods, and there will be exactly one such main parameter per method.

Secondary parameters

The rest of the parameters.

For example, the operation that adds a virtual machine (see [here](#)) has three parameters: **vm**, **clone** and **clone_permissions**. The main parameter is **vm**, as it describes the object that is added. The **clone** and **clone_permissions** parameters are secondary parameters.

The main parameter, when used for input, must be included in the body of the HTTP request. For example, when adding a virtual machine, the **vm** parameter, of type **Vm**, must be included in the request body. So the complete request to add a virtual machine, including all the HTTP details, must look like this:

```
POST /ovirt-engine/api/vms HTTP/1.1
Host: myengine.example.com
```

```

Authorization: Bearer fqbR1ftzh8wBCviLxJcYuV5oSDI=
Content-Type: application/xml
Accept: application/xml

<vm>
  <name>myvm</name>
  <description>My VM</description>
  <cluster>
    <name>Default</name>
  </cluster>
  <template>
    <name>Blank</name>
  </template>
</vm>

```

When used for output, the main parameters are included in the response body. For example, when adding a virtual machine, the `vm` parameter will be included in the response body. So the complete response body will look like this:

```

HTTP/1.1 201 Created
Content-Type: application/xml

<vm href="/ovirt-engine/api/vms/123" id="123">
  <name>myvm</name>
  <description>My VM</description>
  ...
</vm>

```

Secondary parameters are only allowed for input (except for action methods, which are described later), and they must be included as query parameters. For example, when adding a virtual machine with the `clone` parameter set to `true`, the complete request must look like this:

```

POST /ovirt-engine/api/vms?clone=true HTTP/1.1
Host: myengine.example.com
Authorization: Bearer fqbR1ftzh8wBCviLxJcYuV5oSDI=
Content-Type: application/xml
Accept: application/xml

<vm>
  <name>myvm</name>
  <description>My VM</description>
  <cluster>
    <name>Default</name>
  </cluster>
  <template>
    <name>Blank</name>
  </template>
</vm>

```

Action methods only have secondary parameters. They can be used for input and output, and they should be included in the request body, wrapped with an `action` element. For action method used to start a virtual machine (see [here](#)) has a `vm` parameter to describe how the virtual machine should be started, and a `use_cloud_init` parameter to specify if `cloud-init` should be used to configure the guest operating system. So the complete request to start virtual machine `123` using `cloud-init` will look like this when using XML:

```
POST /ovirt-engine/api/vms/123/start HTTP/1.1
Host: myengine.example.com
Authorization: Bearer fqbR1ftzh8wBCviLxJcYuV5oSDI=
Content-Type: application/xml
Accept: application/xml
```

```
<action>
  <use_cloud_init>true</use_cloud_init>
  <vm>
    <initialization>
      <nic_configurations>
        <nic_configuration>
          <name>eth0</name>
          <on_boot>true</on_boot>
          <boot_protocol>static</boot_protocol>
          <ip>
            <address>192.168.0.100</address>
            <netmask>255.255.255.0</netmask>
            <gateway>192.168.0.1</netmask>
          </ip>
        </nic_configuration>
      </nic_configurations>
      <dns_servers>192.168.0.1</dns_servers>
    </initialization>
  </vm>
</action>
```

3.7. SEARCHING

The **list** method of some services has a **search** parameter that can be used to specify a search criteria. When used, the server will only returns objects within the collection that satisfy those criteria. For example, the following request will return only the virtual machine named **myvm**:

```
GET /ovirt-engine/api/vms?search=name%3Dmyvm
```

3.7.1. Maximum results parameter

Use the **max** parameter to limit the number of objects returned. For example, the following request will only return one virtual machine, regardless of how many are available in the system:

```
GET /ovirt-engine/api/vms?max=1
```

A search request without the **max** parameter will return all the objects. Specifying the **max** parameter is recommended to reduce the impact of requests in the overall performance of the system.

3.7.2. Case sensitivity

By default queries are not case sensitive. For example, the following request will return the virtual machines named **myvm**, **MyVM** and **MYVM**:

```
GET /ovirt-engine/api/vms?search=name%3Dmyvm
```

The optional **case_sensitive** boolean parameter can be used to change this behaviour. For example, to get exactly the virtual machine named **myhost**, and not **MyHost** or **MYHOST**, send a request like this:

```
GET /ovirt-engine/api/vms?search=name%3D=myvm&case_sensitive=true
```

3.7.3. Search syntax

The **search** parameters uses the same format as the Red Hat Virtualization query language:

```
(criteria) [sortby (element) asc|desc]
```

The **sortby** clause is optional and only needed when ordering results.

Example search queries:

Collection	Criteria	Result
hosts	vms.status=up	Returns a list of all hosts running virtual machines that are up .
vms	domain=example.com	Returns a list of all virtual machines running on the specified domain.
vms	users.name=mary	Returns a list of all virtual machines belonging to users with the user name mary .
events	severity > normal sortby time	Returns a list of all events with severity higher than normal and sorted by the the value of their time attribute.
events	severity > normal sortby time desc	Returns a list of all events with severity higher than normal and sorted by the the value of their time attribute in descending order.

The value of the **search** parameter must be [URL-encoded](#) to translate reserved characters, such as operators and spaces. For example, the equals sign should be encoded as **%3D**:

```
GET /ovirt-engine/api/vms?search=name%3Dmyvm
```

3.7.4. Wildcards

The asterisk can be used as part of a value, to indicate that any string matches, including the empty string. For example, the following request will return all the virtual machines with names beginning with **myvm** such as **myvm**, **myvm2**, **myvma** or **myvm-webserver**:

```
GET /ovirt-engine/api/vms?search=name%3Dmyvm*
```

3.7.5. Pagination

Some Red Hat Virtualization environments contain large collections of objects. Retrieving all of them with one request isn't practical, and hurts performance. To allow retrieving them page by page the **search** parameter supports an optional **page** clause. This, combined with the **max** parameter, is the basis for paging. For example, to get the first page of virtual machines, with a page size of 10 virtual machines, send request like this:

```
GET /ovirt-engine/api/vms?search=page%201&max=10
```



NOTE

The search parameter is URL-encoded, the actual value of the **search** parameter, before encoding, is **page 1**, so this is actually requesting the first page.

Increase the **page** value to retrieve the next page:

```
GET /ovirt-engine/api/vms?search=page%202&max=10
```

The **page** clause can be used in conjunction with other clauses inside the **search** parameter. For example, the following request will return the second page of virtual machines, but sorting by name:

```
GET /ovirt-engine/api/vms?search=sortBy%20name%20page%202&max=10
```



IMPORTANT

The API is stateless; it is not possible to retain a state between different requests since all requests are independent from each other. As a result, if a status change occurs between your requests, then the page results may be inconsistent.

For example, if you request a specific page from a list of virtual machines, and virtual machines are created or removed before you request the next page, then your results may be missing some of them, or contain duplicates.

3.8. PERMISSIONS

Many of the services that manage a single object provide a reference to a **permissions** service that manages the permissions assigned to that object. Each permission contains links to the user or group, the role and the object. For example, the permissions assigned to a specific virtual machine can be retrieved sending a request like this:

```
GET /ovirt-engine/api/vms/123/permissions
```

The response body will look like this:

```
<permissions>
  <permission id="456" href="/ovirt-engine/api/vms/123/permissions/456">
    <user id="789" href="/ovirt-engine/api/users/789"/>
    <role id="abc" href="/ovirt-engine/api/roles/abc"/>
    <vm id="123" href="/ovirt-engine/api/vms/123"/>
  </permission>
  ...
</permissions>
```

A permission is added to an object sending a **POST** request with a permission representation to this service. Each new permission requires a role and a user.

3.9. HANDLING ERRORS

Some errors require further explanation beyond a standard HTTP status code. For example, the API reports an unsuccessful object state update or action with a **fault** in the response body. The fault contains the **reason** and **detail** attributes. For example, when the server receives a request to create a virtual machine without the mandatory **name** attribute it will respond with the following HTTP response line:

```
HTTP/1.1 400 Bad Request
```

And the following response body:

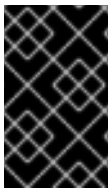
```
<fault>  
  <reason>Incomplete parameters</reason>  
  <detail>Vm [name] required for add</detail>  
</fault>
```


CHAPTER 4. QUICK START EXAMPLE

This chapter provides an example to demonstrate the REST API's ability to setup a basic Red Hat Virtualization environment and create a virtual machine. In addition to the standard prerequisites, this example requires the following:

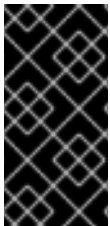
- A networked and configured Red Hat Virtualization installation;
- An ISO file containing a desired virtual machine operating system to install. This chapter uses [CentOS 7](#) for our installation ISO example; and
- Red Hat Virtualization's **engine-iso-uploader** tool to upload your chosen operating system ISO file.

This example uses [curl](#) to demonstrate API requests with a client application. Note that any application capable of HTTP requests can substitute for **curl**.



IMPORTANT

For simplicity, the HTTP request headers in this example omit the **Host** and **Authorization** headers. However, these fields are mandatory and require data specific to your installation of Red Hat Virtualization.



IMPORTANT

All the **curl** examples use **admin@internal** as the user name, **mypassword** as the password, **/etc/pki/ovirt-engine/ca.pem** as the certificate location and **myengine.example.com** as the host name. These are just examples, Make sure to replace them with valid values for your environment.



NOTE

Red Hat Virtualization generates a unique identifier for the **id** attribute for each resource. Identifier codes in this example might appear different to the identifier codes in your Red Hat Virtualization environment.



NOTE

In many examples of this section some of the attributes of results returned by the API have been omitted, to make them shorter. You can always use the reference to find the complete list of attributes. For example, if you want to see the complete list of attributes of the **Cluster** type, just go [here](#).

4.1. EXAMPLE: ACCESS API ENTRY POINT

The following request retrieves a representation of the main entry point for version 4 of of the API:

```
GET /ovirt-engine/api HTTP/1.1
Version: 4
Accept: application/xml
```

Same request, but using the **/v4** URL prefix instead of the **Version** header:

```
GET /ovirt-engine/api/v4 HTTP/1.1
Accept: application/xml
```

Same request, using the **curl** command:

```
curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--request GET \
--header 'Version: 4' \
--header 'Accept: application/xml' \
--user 'admin@internal:mypassword' \
https://myengine.example.com/ovirt-engine/api
```

The result will be an object of type [Api](#):

```
<api>
  <link href="/ovirt-engine/api/clusters" rel="clusters"/>
  <link href="/ovirt-engine/api/datacenters" rel="datacenters"/>
  ...
  <product_info>
    <name>oVirt Engine</name>
    <vendor>ovirt.org</vendor>
    <version>
      <build>0</build>
      <full_version>4.0.0-0.0.e17</full_version>
      <major>4</major>
      <minor>0</minor>
      <revision>0</revision>
    </version>
  </product_info>
  <special_objects>
    <blank_template href="..." id="..."/>
    <root_tag href="..." id="..."/>
  </special_objects>
  <summary>
    <hosts>
      <active>23</active>
      <total>30</total>
    </hosts>
    <storage_domains>
      <active>5</active>
      <total>6</total>
    </storage_domains>
    <users>
      <active>12</active>
      <total>102</total>
    </users>
    <vms>
      <active>253</active>
      <total>545</total>
    </vms>
  </summary>
  <time>2016-10-06T15:38:18.548+02:00</time>
</api>
```



IMPORTANT

When neither the header nor the URL prefix are used, the server will automatically select a version. The default is version **4**. You can change the default version using the **ENGINE_API_DEFAULT_VERSION** configuration parameter:

```
# echo "ENGINE_API_DEFAULT_VERSION=3" > \
/etc/ovirt-engine/engine.conf.d/99-set-default-version.conf
# systemctl restart ovirt-engine
```

Changing this parameter affects all users of the API that don't specify the version explicitly.

The entry point provides a user with links to the collections in a virtualization environment. The **rel** attribute of each collection link provides a reference point for each link. The next step in this example examines the data center collection, which is available through the **datacenters** link.

The entry point also contains other data such as [product_info](#), [special_objects](#) and [summary](#). This data is covered in chapters outside this example.

4.2. EXAMPLE: LIST DATA CENTERS

Red Hat Virtualization creates a **Default** data center on installation. This example uses the **Default** data center as the basis for our virtual environment.

The following request retrieves a representation of the data centers:

```
GET /ovirt-engine/api/datacenters HTTP/1.1
Accept: application/xml
```

Same request, using the **curl** command:

```
# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--request GET \
--header 'Version: 4' \
--header 'Accept: application/xml' \
--user 'admin@internal:mypassword' \
https://myengine.example.com/ovirt-engine/api/datacenters
```

The result will be a list of objects of type [DataCenter](#):

```
<data_centers>
  <data_center href="/ovirt-engine/api/datacenters/001" id="001">
    <name>Default</name>
    <description>The default Data Center</description>
    <link href="/ovirt-engine/api/datacenters/001/clusters"
rel="clusters"/>
    <link href="/ovirt-engine/api/datacenters/001/storagedomains"
rel="storagedomains"/>
    ...
    <local>>false</local>
    <quota_mode>disabled</quota_mode>
    <status>up</status>
```

```

    <supported_versions>
      <version>
        <major>4</major>
        <minor>0</minor>
      </version>
    </supported_versions>
    <version>
      <major>4</major>
      <minor>0</minor>
    </version>
  </data_center>
  ...
</data_centers>

```

Note the **id** of your **Default** data center. It identifies this data center in relation to other resources of your virtual environment.

The data center also contains a link to the [service](#) that manages the storage domains attached to the data center:

```

<link href="/ovirt-engine/api/datacenters/001/storagedomains"
rel="storagedomains"/>

```

That service is used to attach storage domains from the main **storagedomains** collection, which this example covers later.

4.3. EXAMPLE: LIST HOST CLUSTERS

Red Hat Virtualization creates a **Default** hosts cluster on installation. This example uses the **Default** cluster to group resources in your Red Hat Virtualization environment.

The following request retrieves a representation of the cluster collection:

```

GET /ovirt-engine/api/clusters HTTP/1.1
Accept: application/xml

```

Same request, using the **curl** command:

```

curl \
  --cacert '/etc/pki/ovirt-engine/ca.pem' \
  --request GET \
  --header 'Version: 4' \
  --header 'Accept: application/xml' \
  --user 'admin@internal:mypassword' \
  https://myengine.example.com/ovirt-engine/api/clusters

```

The result will be a list of objects of type [Cluster](#):

```

<clusters>
  <cluster href="/ovirt-engine/api/clusters/002" id="002">
    <name>Default</name>
    <description>The default server cluster</description>
    <link href="/ovirt-engine/api/clusters/002/networks" rel="networks"/>
    <link href="/ovirt-engine/api/clusters/002" rel="permissions"/>
  </cluster>
</clusters>

```

```

...
<cpu>
  <architecture>x86_64</architecture>
  <type>Intel Conroe Family</type>
</cpu>
<version>
  <major>4</major>
  <minor>0</minor>
</version>
<data_center href="/ovirt-engine/api/datacenters/001" id="001"/>
</cluster>
...
</clusters>

```

Note the **id** of your **Default** host cluster. It identifies this host cluster in relation to other resources of your virtual environment.

The **Default** cluster is associated with the **Default** data center through a relationship using the **id** and **href** attributes of the **data_center** link:

```
<data_center href="/ovirt-engine/api/datacenters/001" id="001"/>
```

The **networks** link is a reference to the [service](#) that manages the networks associated to this cluster. The next section examines the networks collection in more detail.

4.4. EXAMPLE: LIST LOGICAL NETWORKS

Red Hat Virtualization creates a default **ovirtmgmt** network on installation. This network acts as the management network for Red Hat Virtualization Manager to access hosts.

This network is associated with our **Default** cluster and is a member of the **Default** data center. This example uses the **ovirtmgmt** network to connect our virtual machines.

The following request retrieves the list of logical networks:

```
GET /ovirt-engine/api/networks HTTP/1.1
Accept: application/xml
```

Same request, using the **curl** command:

```
# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--request GET \
--header 'Version: 4' \
--header 'Accept: application/xml' \
--user 'admin@internal:mypassword' \
https://myengine.example.com/ovirt-engine/api/networks
```

The result will be a list of objects of type [Network](#):

```
<networks>
  <network href="/ovirt-engine/api/networks/003" id="003">
    <name>ovirtmgmt</name>
    <description>Management Network</description>
  </network>
</networks>
```

```

    <link href="/ovirt-engine/api/networks/003/permissions"
rel="permissions"/>
    <link href="/ovirt-engine/api/networks/003/vnicprofiles"
rel="vnicprofiles"/>
    <link href="/ovirt-engine/api/networks/003/networklabels"
rel="networklabels"/>
    <mtu>0</mtu>
    <stp>>false</stp>
    <usages>
      <usage>vm</usage>
    </usages>
    <data_center href="/ovirt-engine/api/datacenters/001" id="001"/>
  </network>
  ...
</networks>

```

The **ovirtmgmt** network is attached to the **Default** data center through a relationship using the data center's **id**.

The **ovirtmgmt** network is also attached to the **Default** cluster through a relationship in the cluster's network sub-collection.

4.5. EXAMPLE: LIST HOSTS

This example retrieves the list of hosts and shows a host named **myhost** registered with the virtualization environment:

```

GET /ovirt-engine/api/hosts HTTP/1.1
Accept: application/xml

```

Same request, using the **curl** command:

```

# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--request GET \
--header 'Version: 4' \
--header 'Accept: application/xml' \
--user 'admin@internal:mypassword' \
https://myengine.example.com/ovirt-engine/api/hosts

```

The result will be a list of objects of type [Host](#):

```

<hosts>
  <host href="/ovirt-engine/api/hosts/004" id="004">
    <name>myhost</name>
    <link href="/ovirt-engine/api/hosts/004/nics" rel="nics"/>
    ...
    <address>node40.example.com</address>
    <cpu>
      <name>Intel Core Processor (Haswell, no TSX)</name>
      <speed>3600</speed>
      <topology>
        <cores>1</cores>
        <sockets>2</sockets>
      </topology>
    </cpu>
  </host>
</hosts>

```

```

        <threads>1</threads>
      </topology>
    </cpu>
    <memory>8371830784</memory>
    <os>
      <type>RHEL</type>
      <version>
        <full_version>7 - 2.1511.el7.centos.2.10</full_version>
        <major>7</major>
      </version>
    </os>
    <port>54321</port>
    <status>up</status>
    <cluster href="/ovirt-engine/api/clusters/002" id="002"/>
  </host>
  ...
</hosts>

```

Note the **id** of your host. It identifies this host in relation to other resources of your virtual environment.

This host is a member of the **Default** cluster and accessing the **nics** sub-collection shows this host has a connection to the **ovirtmgmt** network.

4.6. EXAMPLE: CREATE NFS DATA STORAGE

An NFS data storage domain is an exported NFS share attached to a data center and provides storage for virtualized guest images. Creation of a new storage domain requires a **POST** request, with the storage domain representation included, sent to the URL of the storage domain collection.

You can enable the wipe after delete option by default on the storage domain. To configure this specify **wipe_after_delete** in the POST request. This option can be edited after the domain is created, but doing so will not change the wipe after delete property of disks that already exist.

The request should be like this:

```

POST /ovirt-engine/api/storagedomains HTTP/1.1
Accept: application/xml
Content-type: application/xml

```

And the request body should be like this:

```

<storage_domain>
  <name>mydata</name>
  <type>data</type>
  <description>My data</description>
  <storage>
    <type>nfs</type>
    <address>mynfs.example.com</address>
    <path>/exports/mydata</path>
  </storage>
  <host>
    <name>myhost</name>
  </host>
</storage_domain>

```

The same request, using the **curl** command:

```
# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request POST \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<storage_domain>
  <name>mydata</name>
  <description>My data</description>
  <type>data</type>
  <storage>
    <type>nfs</type>
    <address>my nfs.example.com</address>
    <path>/exports/mydata</path>
  </storage>
  <host>
    <name>myhost</name>
  </host>
</storage_domain>
' \
https://myengine.example.com/ovirt-engine/api/storagedomains
```

The server uses host **myhost** to create a NFS data storage domain called **mydata** with an export path of **my nfs.example.com:/exports/mydata**. The API also returns the following representation of the newly created storage domain resource (of type [StorageDomain](#)):

```
<storage_domain href="/ovirt-engine/api/storagedomains/005" id="005">
  <name>mydata</name>
  <description>My data</description>
  <available>42949672960</available>
  <committed>0</committed>
  <master>>false</master>
  <status>unattached</status>
  <storage>
    <address>my nfs.example.com</address>
    <path>/exports/mydata</path>
    <type>nfs</type>
  </storage>
  <storage_format>v3</storage_format>
  <type>data</type>
  <used>9663676416</used>
</storage_domain>
```

4.7. EXAMPLE: CREATE NFS ISO STORAGE

An NFS ISO storage domain is a mounted NFS share attached to a data center and provides storage for DVD/CD-ROM ISO and virtual floppy disk (VFD) image files. Creation of a new storage domain requires a **POST** request, with the storage domain representation included, sent to the URL of the storage domain collection:

The request should be like this:

```
POST /ovirt-engine/api/storagedomains HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

And the request body should be like this:

```
<storage_domain>
  <name>myisos</name>
  <description>My ISOs</description>
  <type>iso</type>
  <storage>
    <type>nfs</type>
    <address>my nfs.example.com</address>
    <path>/exports/myisos</path>
  </storage>
  <host>
    <name>myhost</name>
  </host>
</storage_domain>
```

The same request, using the **curl** command:

```
# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request POST \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<storage_domain>
  <name>myisos</name>
  <description>My ISOs</description>
  <type>iso</type>
  <storage>
    <type>nfs</type>
    <address>my nfs.example.com</address>
    <path>/exports/myisos</path>
  </storage>
  <host>
    <name>myhost</name>
  </host>
</storage_domain>
' \
https://myengine.example.com/ovirt-engine/api/storagedomains
```

The server uses host **myhost** to create a NFS ISO storage domain called **myisos** with an export path of **my nfs.example.com:/exports/myisos**. The API also returns the following representation of the newly created storage domain resource (of type [StorageDomain](#)):

```
<storage_domain href="/ovirt-engine/api/storagedomains/006" id="006">
  <name>myiso</name>
  <description>My ISOs</description>
```

```

<available>42949672960</available>
<committed>0</committed>
<master>>false</master>
<status>unattached</status>
<storage>
  <address>my nfs.example.com</address>
  <path>/exports/myisos</path>
  <type>nfs</type>
</storage>
<storage_format>v1</storage_format>
<type>iso</type>
<used>9663676416</used>
</storage_domain>

```

4.8. EXAMPLE: ATTACH STORAGE DOMAINS TO DATA CENTER

The following example attaches the **mydata** and **myisos** storage domains to the **Default** data center.

To attach the **mydata** storage domain, send a request like this:

```

POST /ovirt-engine/api/datacenters/001/storagedomains HTTP/1.1
Accept: application/xml
Content-type: application/xml

```

With a request body like this:

```

<storage_domain>
  <name>mydata</name>
</storage_domain>

```

Same request, using the **curl** command:

```

# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request POST \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<storage_domain>
  <name>mydata</name>
</storage_domain>
' \
https://myengine.example.com/ovirt-
engine/api/datacenters/001/storagedomains

```

To attach the **myisos** storage domain, send a request like this:

```

POST /ovirt-engine/api/datacenters/001/storagedomains HTTP/1.1
Accept: application/xml
Content-type: application/xml

```

With a request body like this:

```
<storage_domain>
  <name>myisos</name>
</storage_domain>
```

Same request, using the `curl` command:

```
# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request POST \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<storage_domain>
  <name>myisos</name>
</storage_domain>
' \
https://myengine.example.com/ovirt-
engine/api/datacenters/001/storagedomains
```

4.9. EXAMPLE: CREATE VIRTUAL MACHINE

The following example creates a virtual machine called `myvm` on the `Default` cluster using the virtualization environment's `Blank` template as a basis. The request also defines the virtual machine's memory as 512 MiB and sets the boot device to a virtual hard disk.

The request should be contain an object of type `Vm` describing the virtual machine to create:

```
POST /ovirt-engine/api/vms HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

And the request body should be like this:

```
<vm>
  <name>myvm</name>
  <description>My VM</description>
  <cluster>
    <name>Default</name>
  </cluster>
  <template>
    <name>Blank</name>
  </template>
  <memory>536870912</memory>
  <os>
    <boot>
      <devices>
        <device>hd</device>
      </devices>
```

```

    </boot>
  </os>
</vm>

```

Same request, using the **curl** command:

```

# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request POST \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<vm>
  <name>myvm</name>
  <description>My VM</description>
  <cluster>
    <name>Default</name>
  </cluster>
  <template>
    <name>Blank</name>
  </template>
  <memory>536870912</memory>
  <os>
    <boot>
      <devices>
        <device>hd</device>
      </devices>
    </boot>
  </os>
</vm>
' \
https://myengine.example.com/ovirt-engine/api/vms

```

The response body will be an object of the **Vm** type:

```

<vm href="/ovirt-engine/api/vms/007" id="007">
  <name>myvm</name>
  <link href="/ovirt-engine/api/vms/007/diskattachments"
rel="diskattachments"/>
  <link href="/ovirt-engine/api/vms/007/nics" rel="nics"/>
  ...
  <cpu>
    <architecture>x86_64</architecture>
    <topology>
      <cores>1</cores>
      <sockets>1</sockets>
      <threads>1</threads>
    </topology>
  </cpu>
  <memory>1073741824</memory>
  <os>
    <boot>
      <devices>

```

```

        <device>hd</device>
      </devices>
    </boot>
    <type>other</type>
  </os>
  <type>desktop</type>
  <cluster href="/ovirt-engine/api/clusters/002" id="002"/>
  <status>down</status>
  <original_template href="/ovirt-engine/api/templates/000" id="00"/>
  <template href="/ovirt-engine/api/templates/000" id="000"/>
</vm>

```

4.10. EXAMPLE: CREATE A VIRTUAL MACHINE NIC

The following example creates a virtual network interface to connect the example virtual machine to the **ovirtmgmt** network.

The request should be like this:

```

POST /ovirt-engine/api/vms/007/nics HTTP/1.1
Content-Type: application/xml
Accept: application/xml

```

The request body should contain an object of type [Nic](#) describing the NIC to be created:

```

<nic>
  <name>mynic</name>
  <description>My network interface card</description>
</nic>

```

Same request, using the **curl** command:

```

# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request POST \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<nic>
  <name>mynic</name>
  <description>My network interface card</description>
</nic>
' \
https://myengine.example.com/ovirt-engine/api/vms/007/nics

```

4.11. EXAMPLE: CREATE VIRTUAL MACHINE DISK

The following example creates an 8 GiB *copy-on-write* disk for the example virtual machine.

The request should be like this:

```
POST /ovirt-engine/api/vms/007/diskattachments HTTP/1.1
Content-Type: application/xml
Accept: application/xml
```

The request body should be an object of type [DiskAttachment](#) describing the disk and how it will be attached to the virtual machine:

```
<disk_attachment>
  <bootable>>false</bootable>
  <interface>virtio</interface>
  <active>>true</active>
  <disk>
    <description>My disk</description>
    <format>cow</format>
    <name>mydisk</name>
    <provisioned_size>8589934592</provisioned_size>
    <storage_domains>
      <storage_domain>
        <name>mydata</name>
      </storage_domain>
    </storage_domains>
  </disk>
</disk_attachment>
```

Same request, using the **curl** command:

```
# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request POST \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<disk_attachment>
  <bootable>>false</bootable>
  <interface>virtio</interface>
  <active>>true</active>
  <disk>
    <description>My disk</description>
    <format>cow</format>
    <name>mydisk</name>
    <provisioned_size>8589934592</provisioned_size>
    <storage_domains>
      <storage_domain>
        <name>mydata</name>
      </storage_domain>
    </storage_domains>
  </disk>
</disk_attachment>
' \
https://myengine.example.com/ovirt-engine/api/vms/007/diskattachments
```

The **storage_domains** attribute tells the API to store the disk on the **mydata** storage domain.

4.12. EXAMPLE: ATTACH ISO IMAGE TO VIRTUAL MACHINE

The boot media for our example virtual machine requires an CD-ROM or DVD ISO image for an operating system installation. This example uses a CentOS 7 image for installation.

ISO images must be available in the **myisos** ISO domain for the virtual machines to use. Red Hat Virtualization provides an uploader tool that ensures that the ISO images are uploaded into the correct directory path with the correct user permissions.

Once the ISO is uploaded, an API can be used to request the list of files from the ISO storage domain:

```
GET /ovirt-engine/api/storagedomains/006/files HTTP/1.1
Accept: application/xml
```

Same request, using the **curl** command:

```
# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request GET \
--header 'Version: 4' \
--header 'Accept: application/xml' \
https://myengine.example.com/ovirt-engine/api/storagedomains/006/files
```

The server returns the following list of objects of type **File**, one for each available ISO (or floppy) image:

```
<files>
  <file href="..." id="CentOS-7-x86_64-Minimal.iso">
    <name>CentOS-7-x86_64-Minimal.iso</name>
  </file>
  ...
</files>
```

An API user attaches the **CentOS-7-x86_64-Minimal.iso** to our example virtual machine. Attaching an ISO image is equivalent to using the *Change CD* button in the administration or user portal applications.

The request should be like this:

```
PUT /ovirt-engine/api/vms/007/cdroms/00000000-0000-0000-0000-000000000000
HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

The request body should be an object of type **Cdrom** containing an inner **file** attribute to indicate the identifier of the ISO (or floppy) image:

```
<cdrom>
  <file id="CentOS-7-x86_64-Minimal.iso"/>
</cdrom>
```

Same request, using the **curl** command:

```
# curl \
```

```

--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request PUT \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<cdrom>
  <file id="CentOS-7-x86_64-Minimal.iso"/>
</cdrom>
' \
https://myengine.example.com/ovirt-engine/api/vms/007/cdroms/00000000-
0000-0000-0000-000000000000

```

For more details see the documentation of the [service](#) that manages virtual machine CD-ROMS.

4.13. EXAMPLE: START THE VIRTUAL MACHINE

The virtual environment is complete and the virtual machine contains all necessary components to function. This example starts the virtual machine using the [start](#) method.

The request should be like this:

```

POST /ovirt-engine/api/vms/007/start HTTP/1.1
Accept: application/xml
Content-type: application/xml

```

The request body should be like this:

```

<action>
  <vm>
    <os>
      <boot>
        <devices>
          <device>cdrom</device>
        </devices>
      </boot>
    </os>
  </vm>
</action>

```

Same request, using the **curl** command:

```

# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request POST \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<action>
  <vm>
    <os>

```



```
<boot>
  <devices>
    <device>cdrom</device>
  </devices>
</boot>
</os>
</vm>
</action>
' \
https://myengine.example.com/ovirt-engine/api/vms/007/start
```

The additional request body sets the virtual machine's boot device to CD-ROM for this boot only. This enables the virtual machine to install the operating system from the attached ISO image. The boot device reverts back to disk for all future boots.

CHAPTER 5. REQUESTS

This section enumerates all the requests that are available in the API.

- [POST /affinitylabels](#)
- [GET /affinitylabels](#)
- [GET /affinitylabels/{label:id}](#)
- [PUT /affinitylabels/{label:id}](#)
- [DELETE /affinitylabels/{label:id}](#)
- [POST /affinitylabels/{label:id}/hosts](#)
- [GET /affinitylabels/{label:id}/hosts](#)
- [DELETE /affinitylabels/{label:id}/hosts/{host:id}](#)
- [GET /affinitylabels/{label:id}/hosts/{host:id}](#)
- [POST /affinitylabels/{label:id}/vms](#)
- [GET /affinitylabels/{label:id}/vms](#)
- [DELETE /affinitylabels/{label:id}/vms/{vm:id}](#)
- [GET /affinitylabels/{label:id}/vms/{vm:id}](#)
- [POST /bookmarks](#)
- [GET /bookmarks](#)
- [GET /bookmarks/{bookmark:id}](#)
- [PUT /bookmarks/{bookmark:id}](#)
- [DELETE /bookmarks/{bookmark:id}](#)
- [GET /clusterlevels](#)
- [GET /clusterlevels/{level:id}](#)
- [POST /clusters](#)
- [GET /clusters](#)
- [GET /clusters/{cluster:id}](#)
- [PUT /clusters/{cluster:id}](#)
- [DELETE /clusters/{cluster:id}](#)
- [POST /clusters/{cluster:id}/affinitygroups](#)
- [GET /clusters/{cluster:id}/affinitygroups](#)

- GET /clusters/{cluster:id}/affinitygroups/{group:id}
- PUT /clusters/{cluster:id}/affinitygroups/{group:id}
- DELETE /clusters/{cluster:id}/affinitygroups/{group:id}
- POST /clusters/{cluster:id}/affinitygroups/{group:id}/vms
- GET /clusters/{cluster:id}/affinitygroups/{group:id}/vms
- DELETE /clusters/{cluster:id}/affinitygroups/{group:id}/vms/{vm:id}
- POST /clusters/{cluster:id}/cpuprofiles
- GET /clusters/{cluster:id}/cpuprofiles
- GET /clusters/{cluster:id}/cpuprofiles/{profile:id}
- DELETE /clusters/{cluster:id}/cpuprofiles/{profile:id}
- GET /clusters/{cluster:id}/glusterhooks
- GET /clusters/{cluster:id}/glusterhooks/{hook:id}
- DELETE /clusters/{cluster:id}/glusterhooks/{hook:id}
- POST /clusters/{cluster:id}/glusterhooks/{hook:id}/disable
- POST /clusters/{cluster:id}/glusterhooks/{hook:id}/enable
- POST /clusters/{cluster:id}/glusterhooks/{hook:id}/resolve
- POST /clusters/{cluster:id}/glustervolumes
- GET /clusters/{cluster:id}/glustervolumes
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}
- DELETE /clusters/{cluster:id}/glustervolumes/{volume:id}
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/getprofilestatistics
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- DELETE /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/activate
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/migrate
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/stopmigrate
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}
- DELETE /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}

- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/replace](#)
- [GET /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/statistics](#)
- [GET /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/statistics/{statistic:id}](#)
- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/rebalance](#)
- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/resetalloptions](#)
- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/resetoption](#)
- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/setoption](#)
- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/start](#)
- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/startprofile](#)
- [GET /clusters/{cluster:id}/glustervolumes/{volume:id}/statistics](#)
- [GET /clusters/{cluster:id}/glustervolumes/{volume:id}/statistics/{statistic:id}](#)
- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/stop](#)
- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/stopprofile](#)
- [POST /clusters/{cluster:id}/glustervolumes/{volume:id}/stoprebalance](#)
- [GET /clusters/{cluster:id}/networkfilters](#)
- [GET /clusters/{cluster:id}/networkfilters/{networkfilter:id}](#)
- [POST /clusters/{cluster:id}/networks](#)
- [GET /clusters/{cluster:id}/networks](#)
- [GET /clusters/{cluster:id}/networks/{network:id}](#)
- [DELETE /clusters/{cluster:id}/networks/{network:id}](#)
- [PUT /clusters/{cluster:id}/networks/{network:id}](#)
- [POST /clusters/{cluster:id}/permissions](#)
- [GET /clusters/{cluster:id}/permissions](#)
- [GET /clusters/{cluster:id}/permissions/{permission:id}](#)
- [DELETE /clusters/{cluster:id}/permissions/{permission:id}](#)
- [POST /clusters/{cluster:id}/resetemulatedmachine](#)
- [POST /cpuprofiles](#)
- [GET /cpuprofiles](#)
- [GET /cpuprofiles/{profile:id}](#)

- PUT /cpuprofiles/{profile:id}
- DELETE /cpuprofiles/{profile:id}
- POST /cpuprofiles/{profile:id}/permissions
- GET /cpuprofiles/{profile:id}/permissions
- GET /cpuprofiles/{profile:id}/permissions/{permission:id}
- DELETE /cpuprofiles/{profile:id}/permissions/{permission:id}
- POST /datacenters
- GET /datacenters
- GET /datacenters/{datacenter:id}
- PUT /datacenters/{datacenter:id}
- DELETE /datacenters/{datacenter:id}
- POST /datacenters/{datacenter:id}/clusters
- GET /datacenters/{datacenter:id}/clusters
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}
- PUT /datacenters/{datacenter:id}/clusters/{cluster:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}
- PUT /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/vms
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/vms
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/vms/{vm:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/cpuprofiles
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/cpuprofiles
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/cpuprofiles/{profile:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/cpuprofiles/{profile:id}
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glusterhooks

- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glusterhooks/{hook:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/glusterhooks/{hook:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glusterhooks/{hook:id}/disable
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glusterhooks/{hook:id}/enable
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glusterhooks/{hook:id}/resolve
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}
- POST
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/getprofilestatistics
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- DELETE
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- POST
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/activate
- POST
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/migrate
- POST
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/stopmigrat
- GET
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}
- DELETE
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}
- POST
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/r
- GET
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/s
- GET
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/s
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/rebalance
- POST
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/resetalloptions
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/resetoption

- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/setoption
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/start
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/startprofile
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/statistics
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/statistics/{statistic:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/stop
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/stopprofile
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/stoprebalance
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/networkfilters
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/networkfilters/{networkfilter:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/networks
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/networks
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/networks/{network:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/networks/{network:id}
- PUT /datacenters/{datacenter:id}/clusters/{cluster:id}/networks/{network:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/permissions
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/permissions
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/permissions/{permission:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/permissions/{permission:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/resetemulatedmachine
- POST /datacenters/{datacenter:id}/iscsibonds
- GET /datacenters/{datacenter:id}/iscsibonds
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}
- PUT /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}
- PUT /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}

- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/networklabels
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/networklabels
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/networklabels/{label:id}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/networklabels/{label:id}
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/permissions
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/permissions
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/permissions/{permissionid}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/permissions/{permissionid}
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}/ports
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}/ports
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}/ports/{portid}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}/ports/{portid}
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections/{storageconnectionid}
- PUT /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections/{storageconnectionid}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections/{storageconnectionid}
- POST /datacenters/{datacenter:id}/networks

- GET /datacenters/{datacenter:id}/networks
- GET /datacenters/{datacenter:id}/networks/{network:id}
- DELETE /datacenters/{datacenter:id}/networks/{network:id}
- PUT /datacenters/{datacenter:id}/networks/{network:id}
- POST /datacenters/{datacenter:id}/permissions
- GET /datacenters/{datacenter:id}/permissions
- GET /datacenters/{datacenter:id}/permissions/{permission:id}
- DELETE /datacenters/{datacenter:id}/permissions/{permission:id}
- POST /datacenters/{datacenter:id}/qoss
- GET /datacenters/{datacenter:id}/qoss
- GET /datacenters/{datacenter:id}/qoss/{qos:id}
- PUT /datacenters/{datacenter:id}/qoss/{qos:id}
- DELETE /datacenters/{datacenter:id}/qoss/{qos:id}
- POST /datacenters/{datacenter:id}/quotas
- GET /datacenters/{datacenter:id}/quotas
- GET /datacenters/{datacenter:id}/quotas/{quota:id}
- PUT /datacenters/{datacenter:id}/quotas/{quota:id}
- DELETE /datacenters/{datacenter:id}/quotas/{quota:id}
- POST /datacenters/{datacenter:id}/quotas/{quota:id}/permissions
- GET /datacenters/{datacenter:id}/quotas/{quota:id}/permissions
- GET /datacenters/{datacenter:id}/quotas/{quota:id}/permissions/{permission:id}
- DELETE /datacenters/{datacenter:id}/quotas/{quota:id}/permissions/{permission:id}
- POST /datacenters/{datacenter:id}/quotas/{quota:id}/quotaclusterlimits
- GET /datacenters/{datacenter:id}/quotas/{quota:id}/quotaclusterlimits
- GET /datacenters/{datacenter:id}/quotas/{quota:id}/quotaclusterlimits/{limit:id}
- DELETE /datacenters/{datacenter:id}/quotas/{quota:id}/quotaclusterlimits/{limit:id}
- POST /datacenters/{datacenter:id}/quotas/{quota:id}/quotastoragelimits
- GET /datacenters/{datacenter:id}/quotas/{quota:id}/quotastoragelimits
- GET /datacenters/{datacenter:id}/quotas/{quota:id}/quotastoragelimits/{limit:id}

- DELETE /datacenters/{datacenter:id}/quotas/{quota:id}/quotastoragelimits/{limit:id}
- POST /datacenters/{datacenter:id}/storagedomains
- GET /datacenters/{datacenter:id}/storagedomains
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}
- DELETE /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/activate
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/deactivate
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}
- DELETE /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/copy
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/export
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/move
- POST
/datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions
- GET
/datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permis
- DELETE
/datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permis
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/sparsify
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/statistics
- GET
/datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/statistics/{statistic:id}
- POST /diskprofiles
- GET /diskprofiles
- GET /diskprofiles/{diskprofile:id}
- PUT /diskprofiles/{diskprofile:id}
- DELETE /diskprofiles/{diskprofile:id}
- POST /diskprofiles/{diskprofile:id}/permissions

- GET /diskprofiles/{diskprofile:id}/permissions
- GET /diskprofiles/{diskprofile:id}/permissions/{permission:id}
- DELETE /diskprofiles/{diskprofile:id}/permissions/{permission:id}
- POST /disks
- GET /disks
- GET /disks/{disk:id}
- DELETE /disks/{disk:id}
- POST /disks/{disk:id}/copy
- POST /disks/{disk:id}/export
- POST /disks/{disk:id}/move
- POST /disks/{disk:id}/permissions
- GET /disks/{disk:id}/permissions
- GET /disks/{disk:id}/permissions/{permission:id}
- DELETE /disks/{disk:id}/permissions/{permission:id}
- POST /disks/{disk:id}/sparsify
- GET /disks/{disk:id}/statistics
- GET /disks/{disk:id}/statistics/{statistic:id}
- GET /domains
- GET /domains/{domain:id}
- GET /domains/{domain:id}/groups
- GET /domains/{domain:id}/groups/{group:id}
- GET /domains/{domain:id}/users
- GET /domains/{domain:id}/users/{user:id}
- POST /events
- GET /events
- POST /events/undelete
- GET /events/{event:id}
- DELETE /events/{event:id}
- POST /externalhostproviders

- [GET /externalhostproviders](#)
- [GET /externalhostproviders/{provider:id}](#)
- [PUT /externalhostproviders/{provider:id}](#)
- [DELETE /externalhostproviders/{provider:id}](#)
- [GET /externalhostproviders/{provider:id}/certificates](#)
- [GET /externalhostproviders/{provider:id}/certificates/{certificate:id}](#)
- [GET /externalhostproviders/{provider:id}/computeresources](#)
- [GET /externalhostproviders/{provider:id}/computeresources/{resource:id}](#)
- [GET /externalhostproviders/{provider:id}/discoveredhosts](#)
- [GET /externalhostproviders/{provider:id}/discoveredhosts/{host:id}](#)
- [GET /externalhostproviders/{provider:id}/hostgroups](#)
- [GET /externalhostproviders/{provider:id}/hostgroups/{group:id}](#)
- [GET /externalhostproviders/{provider:id}/hosts](#)
- [GET /externalhostproviders/{provider:id}/hosts/{host:id}](#)
- [POST /externalhostproviders/{provider:id}/importcertificates](#)
- [POST /externalhostproviders/{provider:id}/testconnectivity](#)
- [POST /externalvmimports](#)
- [POST /groups](#)
- [GET /groups](#)
- [GET /groups/{group:id}](#)
- [DELETE /groups/{group:id}](#)
- [POST /groups/{group:id}/permissions](#)
- [GET /groups/{group:id}/permissions](#)
- [GET /groups/{group:id}/permissions/{permission:id}](#)
- [DELETE /groups/{group:id}/permissions/{permission:id}](#)
- [GET /groups/{group:id}/roles](#)
- [GET /groups/{group:id}/roles/{role:id}](#)
- [DELETE /groups/{group:id}/roles/{role:id}](#)
- [PUT /groups/{group:id}/roles/{role:id}](#)

- [POST /groups/{group:id}/roles/{role:id}/permits](#)
- [GET /groups/{group:id}/roles/{role:id}/permits](#)
- [GET /groups/{group:id}/roles/{role:id}/permits/{permit:id}](#)
- [DELETE /groups/{group:id}/roles/{role:id}/permits/{permit:id}](#)
- [POST /groups/{group:id}/tags](#)
- [GET /groups/{group:id}/tags](#)
- [GET /groups/{group:id}/tags/{tag:id}](#)
- [DELETE /groups/{group:id}/tags/{tag:id}](#)
- [POST /hosts](#)
- [GET /hosts](#)
- [GET /hosts/{host:id}](#)
- [PUT /hosts/{host:id}](#)
- [DELETE /hosts/{host:id}](#)
- [POST /hosts/{host:id}/activate](#)
- [POST /hosts/{host:id}/affinitylabels](#)
- [GET /hosts/{host:id}/affinitylabels](#)
- [GET /hosts/{host:id}/affinitylabels/{label:id}](#)
- [DELETE /hosts/{host:id}/affinitylabels/{label:id}](#)
- [POST /hosts/{host:id}/approve](#)
- [POST /hosts/{host:id}/commitnetconfig](#)
- [POST /hosts/{host:id}/deactivate](#)
- [GET /hosts/{host:id}/devices](#)
- [GET /hosts/{host:id}/devices/{device:id}](#)
- [POST /hosts/{host:id}/enrollcertificate](#)
- [POST /hosts/{host:id}/fence](#)
- [POST /hosts/{host:id}/fenceagents](#)
- [GET /hosts/{host:id}/fenceagents](#)
- [GET /hosts/{host:id}/fenceagents/{agent:id}](#)
- [PUT /hosts/{host:id}/fenceagents/{agent:id}](#)

- DELETE /hosts/{host:id}/fenceagents/{agent:id}
- POST /hosts/{host:id}/forceselectspm
- GET /hosts/{host:id}/hooks
- GET /hosts/{host:id}/hooks/{hook:id}
- POST /hosts/{host:id}/install
- POST /hosts/{host:id}/iscsidiscover
- POST /hosts/{host:id}/iscsilogin
- GET /hosts/{host:id}/katelloerrata
- GET /hosts/{host:id}/katelloerrata/{katelloerratum:id}
- POST /hosts/{host:id}/networkattachments
- GET /hosts/{host:id}/networkattachments
- GET /hosts/{host:id}/networkattachments/{attachment:id}
- PUT /hosts/{host:id}/networkattachments/{attachment:id}
- DELETE /hosts/{host:id}/networkattachments/{attachment:id}
- GET /hosts/{host:id}/nics
- GET /hosts/{host:id}/nics/{nic:id}
- POST /hosts/{host:id}/nics/{nic:id}/networkattachments
- GET /hosts/{host:id}/nics/{nic:id}/networkattachments
- GET /hosts/{host:id}/nics/{nic:id}/networkattachments/{attachment:id}
- PUT /hosts/{host:id}/nics/{nic:id}/networkattachments/{attachment:id}
- DELETE /hosts/{host:id}/nics/{nic:id}/networkattachments/{attachment:id}
- POST /hosts/{host:id}/nics/{nic:id}/networklabels
- GET /hosts/{host:id}/nics/{nic:id}/networklabels
- GET /hosts/{host:id}/nics/{nic:id}/networklabels/{label:id}
- DELETE /hosts/{host:id}/nics/{nic:id}/networklabels/{label:id}
- GET /hosts/{host:id}/nics/{nic:id}/statistics
- GET /hosts/{host:id}/nics/{nic:id}/statistics/{statistic:id}
- POST /hosts/{host:id}/nics/{nic:id}/updatevirtualfunctionsconfiguration
- POST /hosts/{host:id}/nics/{nic:id}/virtualfunctionallowedlabels

- GET /hosts/{host:id}/nics/{nic:id}/virtualfunctionallowedlabels
- GET /hosts/{host:id}/nics/{nic:id}/virtualfunctionallowedlabels/{label:id}
- DELETE /hosts/{host:id}/nics/{nic:id}/virtualfunctionallowedlabels/{label:id}
- POST /hosts/{host:id}/nics/{nic:id}/virtualfunctionallowednetworks
- GET /hosts/{host:id}/nics/{nic:id}/virtualfunctionallowednetworks
- GET /hosts/{host:id}/nics/{nic:id}/virtualfunctionallowednetworks/{network:id}
- DELETE /hosts/{host:id}/nics/{nic:id}/virtualfunctionallowednetworks/{network:id}
- GET /hosts/{host:id}/numanodes
- GET /hosts/{host:id}/numanodes/{node:id}
- GET /hosts/{host:id}/numanodes/{node:id}/statistics
- GET /hosts/{host:id}/numanodes/{node:id}/statistics/{statistic:id}
- POST /hosts/{host:id}/permissions
- GET /hosts/{host:id}/permissions
- GET /hosts/{host:id}/permissions/{permission:id}
- DELETE /hosts/{host:id}/permissions/{permission:id}
- POST /hosts/{host:id}/refresh
- POST /hosts/{host:id}/setupnetworks
- GET /hosts/{host:id}/statistics
- GET /hosts/{host:id}/statistics/{statistic:id}
- GET /hosts/{host:id}/storage
- GET /hosts/{host:id}/storage/{storage:id}
- POST /hosts/{host:id}/storageconnectionextensions
- GET /hosts/{host:id}/storageconnectionextensions
- GET /hosts/{host:id}/storageconnectionextensions/{storageconnectionextension:id}
- PUT /hosts/{host:id}/storageconnectionextensions/{storageconnectionextension:id}
- DELETE /hosts/{host:id}/storageconnectionextensions/{storageconnectionextension:id}
- POST /hosts/{host:id}/tags
- GET /hosts/{host:id}/tags
- GET /hosts/{host:id}/tags/{tag:id}

- DELETE /hosts/{host:id}/tags/{tag:id}
- GET /hosts/{host:id}/unmanagednetworks
- GET /hosts/{host:id}/unmanagednetworks/{unmanagednetwork:id}
- DELETE /hosts/{host:id}/unmanagednetworks/{unmanagednetwork:id}
- POST /hosts/{host:id}/unregisteredstoragedomainsdiscover
- POST /hosts/{host:id}/upgrade
- POST /hosts/{host:id}/upgradecheck
- GET /icons
- GET /icons/{icon:id}
- POST /imagetransfers
- GET /imagetransfers
- GET /imagetransfers/{imagetransfer:id}
- POST /imagetransfers/{imagetransfer:id}/extend
- POST /imagetransfers/{imagetransfer:id}/finalize
- POST /imagetransfers/{imagetransfer:id}/pause
- POST /imagetransfers/{imagetransfer:id}/resume
- POST /instancetypes
- GET /instancetypes
- GET /instancetypes/{instancetype:id}
- PUT /instancetypes/{instancetype:id}
- DELETE /instancetypes/{instancetype:id}
- POST /instancetypes/{instancetype:id}/graphicsconsoles
- GET /instancetypes/{instancetype:id}/graphicsconsoles
- GET /instancetypes/{instancetype:id}/graphicsconsoles/{console:id}
- DELETE /instancetypes/{instancetype:id}/graphicsconsoles/{console:id}
- POST /instancetypes/{instancetype:id}/nics
- GET /instancetypes/{instancetype:id}/nics
- GET /instancetypes/{instancetype:id}/nics/{nic:id}
- PUT /instancetypes/{instancetype:id}/nics/{nic:id}

- DELETE /instancetypes/{instancetype:id}/nics/{nic:id}
- POST /instancetypes/{instancetype:id}/watchdogs
- GET /instancetypes/{instancetype:id}/watchdogs
- GET /instancetypes/{instancetype:id}/watchdogs/{watchdog:id}
- PUT /instancetypes/{instancetype:id}/watchdogs/{watchdog:id}
- DELETE /instancetypes/{instancetype:id}/watchdogs/{watchdog:id}
- POST /jobs
- GET /jobs
- GET /jobs/{job:id}
- POST /jobs/{job:id}/clear
- POST /jobs/{job:id}/end
- POST /jobs/{job:id}/steps
- GET /jobs/{job:id}/steps
- GET /jobs/{job:id}/steps/{step:id}
- POST /jobs/{job:id}/steps/{step:id}/end
- GET /jobs/{job:id}/steps/{step:id}/statistics
- GET /jobs/{job:id}/steps/{step:id}/statistics/{statistic:id}
- GET /katelloerrata
- GET /katelloerrata/{katelloerratum:id}
- POST /macpools
- GET /macpools
- GET /macpools/{macpool:id}
- PUT /macpools/{macpool:id}
- DELETE /macpools/{macpool:id}
- GET /networkfilters
- GET /networkfilters/{networkfilter:id}
- POST /networks
- GET /networks
- GET /networks/{network:id}

- PUT /networks/{network:id}
- DELETE /networks/{network:id}
- POST /networks/{network:id}/networklabels
- GET /networks/{network:id}/networklabels
- GET /networks/{network:id}/networklabels/{label:id}
- DELETE /networks/{network:id}/networklabels/{label:id}
- POST /networks/{network:id}/permissions
- GET /networks/{network:id}/permissions
- GET /networks/{network:id}/permissions/{permission:id}
- DELETE /networks/{network:id}/permissions/{permission:id}
- POST /networks/{network:id}/vnicprofiles
- GET /networks/{network:id}/vnicprofiles
- GET /networks/{network:id}/vnicprofiles/{profile:id}
- DELETE /networks/{network:id}/vnicprofiles/{profile:id}
- POST /networks/{network:id}/vnicprofiles/{profile:id}/permissions
- GET /networks/{network:id}/vnicprofiles/{profile:id}/permissions
- GET /networks/{network:id}/vnicprofiles/{profile:id}/permissions/{permission:id}
- DELETE /networks/{network:id}/vnicprofiles/{profile:id}/permissions/{permission:id}
- POST /openstackimageproviders
- GET /openstackimageproviders
- GET /openstackimageproviders/{provider:id}
- PUT /openstackimageproviders/{provider:id}
- DELETE /openstackimageproviders/{provider:id}
- GET /openstackimageproviders/{provider:id}/certificates
- GET /openstackimageproviders/{provider:id}/certificates/{certificate:id}
- GET /openstackimageproviders/{provider:id}/images
- GET /openstackimageproviders/{provider:id}/images/{image:id}
- POST /openstackimageproviders/{provider:id}/images/{image:id}/import
- POST /openstackimageproviders/{provider:id}/importcertificates

- [POST /openstackimageproviders/{provider:id}/testconnectivity](#)
- [POST /openstacknetworkproviders](#)
- [GET /openstacknetworkproviders](#)
- [GET /openstacknetworkproviders/{provider:id}](#)
- [PUT /openstacknetworkproviders/{provider:id}](#)
- [DELETE /openstacknetworkproviders/{provider:id}](#)
- [GET /openstacknetworkproviders/{provider:id}/certificates](#)
- [GET /openstacknetworkproviders/{provider:id}/certificates/{certificate:id}](#)
- [POST /openstacknetworkproviders/{provider:id}/importcertificates](#)
- [GET /openstacknetworkproviders/{provider:id}/networks](#)
- [GET /openstacknetworkproviders/{provider:id}/networks/{network:id}](#)
- [POST /openstacknetworkproviders/{provider:id}/networks/{network:id}/import](#)
- [POST /openstacknetworkproviders/{provider:id}/networks/{network:id}/subnets](#)
- [GET /openstacknetworkproviders/{provider:id}/networks/{network:id}/subnets](#)
- [GET /openstacknetworkproviders/{provider:id}/networks/{network:id}/subnets/{subnet:id}](#)
- [DELETE /openstacknetworkproviders/{provider:id}/networks/{network:id}/subnets/{subnet:id}](#)
- [POST /openstacknetworkproviders/{provider:id}/testconnectivity](#)
- [POST /openstackvolumeproviders](#)
- [GET /openstackvolumeproviders](#)
- [GET /openstackvolumeproviders/{provider:id}](#)
- [PUT /openstackvolumeproviders/{provider:id}](#)
- [DELETE /openstackvolumeproviders/{provider:id}](#)
- [POST /openstackvolumeproviders/{provider:id}/authenticationkeys](#)
- [GET /openstackvolumeproviders/{provider:id}/authenticationkeys](#)
- [GET /openstackvolumeproviders/{provider:id}/authenticationkeys/{key:id}](#)
- [PUT /openstackvolumeproviders/{provider:id}/authenticationkeys/{key:id}](#)
- [DELETE /openstackvolumeproviders/{provider:id}/authenticationkeys/{key:id}](#)
- [GET /openstackvolumeproviders/{provider:id}/certificates](#)
- [GET /openstackvolumeproviders/{provider:id}/certificates/{certificate:id}](#)

- [POST /openstackvolumeproviders/{provider:id}/importcertificates](#)
- [POST /openstackvolumeproviders/{provider:id}/testconnectivity](#)
- [GET /openstackvolumeproviders/{provider:id}/volumetypes](#)
- [GET /openstackvolumeproviders/{provider:id}/volumetypes/{type:id}](#)
- [GET /operatingsystems](#)
- [GET /operatingsystems/{operatingsystem:id}](#)
- [POST /permissions](#)
- [GET /permissions](#)
- [GET /permissions/{permission:id}](#)
- [DELETE /permissions/{permission:id}](#)
- [POST /roles](#)
- [GET /roles](#)
- [GET /roles/{role:id}](#)
- [DELETE /roles/{role:id}](#)
- [PUT /roles/{role:id}](#)
- [POST /roles/{role:id}/permits](#)
- [GET /roles/{role:id}/permits](#)
- [GET /roles/{role:id}/permits/{permit:id}](#)
- [DELETE /roles/{role:id}/permits/{permit:id}](#)
- [POST /schedulingpolicies](#)
- [GET /schedulingpolicies](#)
- [GET /schedulingpolicies/{policy:id}](#)
- [PUT /schedulingpolicies/{policy:id}](#)
- [DELETE /schedulingpolicies/{policy:id}](#)
- [POST /schedulingpolicies/{policy:id}/balances](#)
- [GET /schedulingpolicies/{policy:id}/balances](#)
- [GET /schedulingpolicies/{policy:id}/balances/{balance:id}](#)
- [DELETE /schedulingpolicies/{policy:id}/balances/{balance:id}](#)
- [POST /schedulingpolicies/{policy:id}/filters](#)

- GET /schedulingpolicies/{policy:id}/filters
- GET /schedulingpolicies/{policy:id}/filters/{filter:id}
- DELETE /schedulingpolicies/{policy:id}/filters/{filter:id}
- POST /schedulingpolicies/{policy:id}/weights
- GET /schedulingpolicies/{policy:id}/weights
- GET /schedulingpolicies/{policy:id}/weights/{weight:id}
- DELETE /schedulingpolicies/{policy:id}/weights/{weight:id}
- GET /schedulingpolicyunits
- GET /schedulingpolicyunits/{unit:id}
- DELETE /schedulingpolicyunits/{unit:id}
- POST /storageconnections
- GET /storageconnections
- GET /storageconnections/{storageconnection:id}
- PUT /storageconnections/{storageconnection:id}
- DELETE /storageconnections/{storageconnection:id}
- POST /storagedomains
- GET /storagedomains
- GET /storagedomains/{storagedomain:id}
- PUT /storagedomains/{storagedomain:id}
- DELETE /storagedomains/{storagedomain:id}
- POST /storagedomains/{storagedomain:id}/diskprofiles
- GET /storagedomains/{storagedomain:id}/diskprofiles
- GET /storagedomains/{storagedomain:id}/diskprofiles/{profile:id}
- DELETE /storagedomains/{storagedomain:id}/diskprofiles/{profile:id}
- POST /storagedomains/{storagedomain:id}/disks
- GET /storagedomains/{storagedomain:id}/disks
- GET /storagedomains/{storagedomain:id}/disks/{disk:id}
- DELETE /storagedomains/{storagedomain:id}/disks/{disk:id}
- POST /storagedomains/{storagedomain:id}/disks/{disk:id}/copy

- [POST /storagedomains/{storagedomain:id}/disks/{disk:id}/export](#)
- [POST /storagedomains/{storagedomain:id}/disks/{disk:id}/move](#)
- [POST /storagedomains/{storagedomain:id}/disks/{disk:id}/permissions](#)
- [GET /storagedomains/{storagedomain:id}/disks/{disk:id}/permissions](#)
- [GET /storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permission:id}](#)
- [DELETE /storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permission:id}](#)
- [POST /storagedomains/{storagedomain:id}/disks/{disk:id}/sparsify](#)
- [GET /storagedomains/{storagedomain:id}/disks/{disk:id}/statistics](#)
- [GET /storagedomains/{storagedomain:id}/disks/{disk:id}/statistics/{statistic:id}](#)
- [GET /storagedomains/{storagedomain:id}/disksnapshots](#)
- [GET /storagedomains/{storagedomain:id}/disksnapshots/{snapshot:id}](#)
- [DELETE /storagedomains/{storagedomain:id}/disksnapshots/{snapshot:id}](#)
- [GET /storagedomains/{storagedomain:id}/files](#)
- [GET /storagedomains/{storagedomain:id}/files/{file:id}](#)
- [GET /storagedomains/{storagedomain:id}/images](#)
- [GET /storagedomains/{storagedomain:id}/images/{image:id}](#)
- [POST /storagedomains/{storagedomain:id}/images/{image:id}/import](#)
- [POST /storagedomains/{storagedomain:id}/isattached](#)
- [POST /storagedomains/{storagedomain:id}/permissions](#)
- [GET /storagedomains/{storagedomain:id}/permissions](#)
- [GET /storagedomains/{storagedomain:id}/permissions/{permission:id}](#)
- [DELETE /storagedomains/{storagedomain:id}/permissions/{permission:id}](#)
- [POST /storagedomains/{storagedomain:id}/reduceluns](#)
- [POST /storagedomains/{storagedomain:id}/refreshluns](#)
- [POST /storagedomains/{storagedomain:id}/storageconnections](#)
- [GET /storagedomains/{storagedomain:id}/storageconnections](#)
- [GET /storagedomains/{storagedomain:id}/storageconnections/{connection:id}](#)
- [DELETE /storagedomains/{storagedomain:id}/storageconnections/{connection:id}](#)
- [GET /storagedomains/{storagedomain:id}/templates](#)

- GET /storagedomains/{storagedomain:id}/templates/{template:id}
- DELETE /storagedomains/{storagedomain:id}/templates/{template:id}
- GET /storagedomains/{storagedomain:id}/templates/{template:id}/disks
- GET /storagedomains/{storagedomain:id}/templates/{template:id}/disks/{disk:id}
- POST /storagedomains/{storagedomain:id}/templates/{template:id}/import
- POST /storagedomains/{storagedomain:id}/templates/{template:id}/register
- POST /storagedomains/{storagedomain:id}/updateovfstore
- GET /storagedomains/{storagedomain:id}/vms
- GET /storagedomains/{storagedomain:id}/vms/{vm:id}
- DELETE /storagedomains/{storagedomain:id}/vms/{vm:id}
- GET /storagedomains/{storagedomain:id}/vms/{vm:id}/diskattachments
- GET /storagedomains/{storagedomain:id}/vms/{vm:id}/diskattachments/{attachment:id}
- GET /storagedomains/{storagedomain:id}/vms/{vm:id}/disks
- GET /storagedomains/{storagedomain:id}/vms/{vm:id}/disks/{disk:id}
- POST /storagedomains/{storagedomain:id}/vms/{vm:id}/import
- POST /storagedomains/{storagedomain:id}/vms/{vm:id}/register
- POST /tags
- GET /tags
- GET /tags/{tag:id}
- PUT /tags/{tag:id}
- DELETE /tags/{tag:id}
- POST /templates
- GET /templates
- GET /templates/{template:id}
- PUT /templates/{template:id}
- DELETE /templates/{template:id}
- GET /templates/{template:id}/cdroms
- GET /templates/{template:id}/cdroms/{cdrom:id}
- GET /templates/{template:id}/diskattachments

- GET /templates/{template:id}/diskattachments/{attachment:id}
- DELETE /templates/{template:id}/diskattachments/{attachment:id}
- POST /templates/{template:id}/export
- POST /templates/{template:id}/graphicsconsoles
- GET /templates/{template:id}/graphicsconsoles
- GET /templates/{template:id}/graphicsconsoles/{console:id}
- DELETE /templates/{template:id}/graphicsconsoles/{console:id}
- POST /templates/{template:id}/nics
- GET /templates/{template:id}/nics
- GET /templates/{template:id}/nics/{nic:id}
- PUT /templates/{template:id}/nics/{nic:id}
- DELETE /templates/{template:id}/nics/{nic:id}
- POST /templates/{template:id}/permissions
- GET /templates/{template:id}/permissions
- GET /templates/{template:id}/permissions/{permission:id}
- DELETE /templates/{template:id}/permissions/{permission:id}
- POST /templates/{template:id}/tags
- GET /templates/{template:id}/tags
- GET /templates/{template:id}/tags/{tag:id}
- DELETE /templates/{template:id}/tags/{tag:id}
- POST /templates/{template:id}/watchdogs
- GET /templates/{template:id}/watchdogs
- GET /templates/{template:id}/watchdogs/{watchdog:id}
- PUT /templates/{template:id}/watchdogs/{watchdog:id}
- DELETE /templates/{template:id}/watchdogs/{watchdog:id}
- POST /users
- GET /users
- GET /users/{user:id}
- DELETE /users/{user:id}

- POST /users/{user:id}/permissions
- GET /users/{user:id}/permissions
- GET /users/{user:id}/permissions/{permission:id}
- DELETE /users/{user:id}/permissions/{permission:id}
- GET /users/{user:id}/roles
- GET /users/{user:id}/roles/{role:id}
- DELETE /users/{user:id}/roles/{role:id}
- PUT /users/{user:id}/roles/{role:id}
- POST /users/{user:id}/roles/{role:id}/permits
- GET /users/{user:id}/roles/{role:id}/permits
- GET /users/{user:id}/roles/{role:id}/permits/{permit:id}
- DELETE /users/{user:id}/roles/{role:id}/permits/{permit:id}
- POST /users/{user:id}/sshpublickeys
- GET /users/{user:id}/sshpublickeys
- GET /users/{user:id}/sshpublickeys/{key:id}
- PUT /users/{user:id}/sshpublickeys/{key:id}
- DELETE /users/{user:id}/sshpublickeys/{key:id}
- POST /users/{user:id}/tags
- GET /users/{user:id}/tags
- GET /users/{user:id}/tags/{tag:id}
- DELETE /users/{user:id}/tags/{tag:id}
- POST /vmpools
- GET /vmpools
- GET /vmpools/{pool:id}
- PUT /vmpools/{pool:id}
- DELETE /vmpools/{pool:id}
- POST /vmpools/{pool:id}/allocatevm
- POST /vmpools/{pool:id}/permissions
- GET /vmpools/{pool:id}/permissions

- GET /vmpools/{pool:id}/permissions/{permission:id}
- DELETE /vmpools/{pool:id}/permissions/{permission:id}
- POST /vms
- GET /vms
- GET /vms/{vm:id}
- PUT /vms/{vm:id}
- DELETE /vms/{vm:id}
- POST /vms/{vm:id}/affinitylabels
- GET /vms/{vm:id}/affinitylabels
- GET /vms/{vm:id}/affinitylabels/{label:id}
- DELETE /vms/{vm:id}/affinitylabels/{label:id}
- GET /vms/{vm:id}/applications
- GET /vms/{vm:id}/applications/{application:id}
- POST /vms/{vm:id}/cancelmigration
- GET /vms/{vm:id}/cdroms
- GET /vms/{vm:id}/cdroms/{cdrom:id}
- PUT /vms/{vm:id}/cdroms/{cdrom:id}
- POST /vms/{vm:id}/clone
- POST /vms/{vm:id}/commitsnapshot
- POST /vms/{vm:id}/detach
- POST /vms/{vm:id}/diskattachments
- GET /vms/{vm:id}/diskattachments
- GET /vms/{vm:id}/diskattachments/{attachment:id}
- DELETE /vms/{vm:id}/diskattachments/{attachment:id}
- PUT /vms/{vm:id}/diskattachments/{attachment:id}
- POST /vms/{vm:id}/export
- POST /vms/{vm:id}/freezefilesystems
- POST /vms/{vm:id}/graphicsconsoles
- GET /vms/{vm:id}/graphicsconsoles

- GET /vms/{vm:id}/graphicsconsoles/{console:id}
- DELETE /vms/{vm:id}/graphicsconsoles/{console:id}
- POST /vms/{vm:id}/graphicsconsoles/{console:id}/proxyticket
- POST /vms/{vm:id}/graphicsconsoles/{console:id}/remoteviewerconnectionfile
- POST /vms/{vm:id}/graphicsconsoles/{console:id}/ticket
- POST /vms/{vm:id}/hostdevices
- GET /vms/{vm:id}/hostdevices
- GET /vms/{vm:id}/hostdevices/{device:id}
- DELETE /vms/{vm:id}/hostdevices/{device:id}
- GET /vms/{vm:id}/katelloerrata
- GET /vms/{vm:id}/katelloerrata/{katelloerratum:id}
- POST /vms/{vm:id}/logon
- POST /vms/{vm:id}/maintenance
- POST /vms/{vm:id}/migrate
- POST /vms/{vm:id}/nics
- GET /vms/{vm:id}/nics
- GET /vms/{vm:id}/nics/{nic:id}
- PUT /vms/{vm:id}/nics/{nic:id}
- DELETE /vms/{vm:id}/nics/{nic:id}
- POST /vms/{vm:id}/nics/{nic:id}/activate
- POST /vms/{vm:id}/nics/{nic:id}/deactivate
- GET /vms/{vm:id}/nics/{nic:id}/reporteddevices
- GET /vms/{vm:id}/nics/{nic:id}/reporteddevices/{reporteddevice:id}
- GET /vms/{vm:id}/nics/{nic:id}/statistics
- GET /vms/{vm:id}/nics/{nic:id}/statistics/{statistic:id}
- POST /vms/{vm:id}/numanodes
- GET /vms/{vm:id}/numanodes
- GET /vms/{vm:id}/numanodes/{node:id}
- PUT /vms/{vm:id}/numanodes/{node:id}

- DELETE /vms/{vm:id}/numanodes/{node:id}
- POST /vms/{vm:id}/permissions
- GET /vms/{vm:id}/permissions
- GET /vms/{vm:id}/permissions/{permission:id}
- DELETE /vms/{vm:id}/permissions/{permission:id}
- POST /vms/{vm:id}/previewsnapshot
- POST /vms/{vm:id}/reboot
- POST /vms/{vm:id}/reordermacaddresses
- GET /vms/{vm:id}/reporteddevices
- GET /vms/{vm:id}/reporteddevices/{reporteddevice:id}
- GET /vms/{vm:id}/sessions
- GET /vms/{vm:id}/sessions/{session:id}
- POST /vms/{vm:id}/shutdown
- POST /vms/{vm:id}/snapshots
- GET /vms/{vm:id}/snapshots
- GET /vms/{vm:id}/snapshots/{snapshot:id}
- DELETE /vms/{vm:id}/snapshots/{snapshot:id}
- GET /vms/{vm:id}/snapshots/{snapshot:id}/cdroms
- GET /vms/{vm:id}/snapshots/{snapshot:id}/cdroms/{cdrom:id}
- GET /vms/{vm:id}/snapshots/{snapshot:id}/disks
- GET /vms/{vm:id}/snapshots/{snapshot:id}/disks/{disk:id}
- GET /vms/{vm:id}/snapshots/{snapshot:id}/nics
- GET /vms/{vm:id}/snapshots/{snapshot:id}/nics/{nic:id}
- POST /vms/{vm:id}/snapshots/{snapshot:id}/restore
- POST /vms/{vm:id}/start
- GET /vms/{vm:id}/statistics
- GET /vms/{vm:id}/statistics/{statistic:id}
- POST /vms/{vm:id}/stop
- POST /vms/{vm:id}/suspend

- [POST /vms/{vm:id}/tags](#)
- [GET /vms/{vm:id}/tags](#)
- [GET /vms/{vm:id}/tags/{tag:id}](#)
- [DELETE /vms/{vm:id}/tags/{tag:id}](#)
- [POST /vms/{vm:id}/thawfilesystems](#)
- [POST /vms/{vm:id}/ticket](#)
- [POST /vms/{vm:id}/undosnapshot](#)
- [POST /vms/{vm:id}/watchdogs](#)
- [GET /vms/{vm:id}/watchdogs](#)
- [GET /vms/{vm:id}/watchdogs/{watchdog:id}](#)
- [PUT /vms/{vm:id}/watchdogs/{watchdog:id}](#)
- [DELETE /vms/{vm:id}/watchdogs/{watchdog:id}](#)
- [POST /vnicprofiles](#)
- [GET /vnicprofiles](#)
- [GET /vnicprofiles/{profile:id}](#)
- [PUT /vnicprofiles/{profile:id}](#)
- [DELETE /vnicprofiles/{profile:id}](#)
- [POST /vnicprofiles/{profile:id}/permissions](#)
- [GET /vnicprofiles/{profile:id}/permissions](#)
- [GET /vnicprofiles/{profile:id}/permissions/{permission:id}](#)
- [DELETE /vnicprofiles/{profile:id}/permissions/{permission:id}](#)

CHAPTER 6. SERVICES

This section enumerates all the services that are available in the API.

6.1. AFFINITYGROUP

This service manages a single affinity group.

Table 6.1. Methods summary

Name	Summary
get	Retrieve the affinity group details.
remove	Remove the affinity group.
update	Update the affinity group.

6.1.1. get GET

Retrieve the affinity group details.

```
<affinity_group id="00000000-0000-0000-0000-000000000000">
  <name>AF_GROUP_001</name>
  <cluster id="00000000-0000-0000-0000-000000000000" />
  <positive>true</positive>
  <enforcing>true</enforcing>
</affinity_group>
```

Table 6.2. Parameters summary

Name	Type	Direction	Summary
group	AffinityGroup	Out	The affinity group.

6.1.2. remove DELETE

Remove the affinity group.

```
DELETE /ovirt-engine/api/clusters/000-000/affinitygroups/123-456
```

Table 6.3. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the removal should be performed asynchronously.

6.1.3. update PUT

Update the affinity group.

Table 6.4. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
group	AffinityGroup	In/Out	The affinity group.

6.2. AFFINITYGROUPVM

This service manages a single virtual machine to affinity group assignment.

Table 6.5. Methods summary

Name	Summary
remove	Remove this virtual machine from the affinity group.

6.2.1. remove DELETE

Remove this virtual machine from the affinity group.

Table 6.6. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the removal should be performed asynchronously.

6.3. AFFINITYGROUPVMS

This service manages a collection of all the virtual machines assigned to an affinity group.

Table 6.7. Methods summary

Name	Summary
add	Add a virtual machine to the affinity group.
list	List all virtual machines assigned to this affinity group.

6.3.1. add POST

Add a virtual machine to the affinity group.

For example, to add the virtual machine 000-000 to affinity group 123-456 send a request to:

```
POST /ovirt-engine/api/clusters/000-000/affinitygroups/123-456/vms
```

With the following body:

```
<vm id="000-000"/>
```

Table 6.8. Parameters summary

Name	Type	Direction	Summary
vm	Vm	In/Out	

6.3.2. list GET

List all virtual machines assigned to this affinity group.

Table 6.9. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of virtual machines to return.
vms	Vm[]	Out	

6.3.2.1. max

Sets the maximum number of virtual machines to return. If not specified, all the virtual machines are returned.

6.4. AFFINITYGROUPS

The affinity groups service manages virtual machine relationships and dependencies.

Table 6.10. Methods summary

Name	Summary
add	Create a new affinity group.
list	List existing affinity groups.

6.4.1. add POST

Create a new affinity group.

Post a request like in the example below to create a new affinity group:


```
POST /ovirt-engine/api/clusters/000-000/affinitygroups
```

And use the following example in its body:

```
<affinity_group>
  <name>AF_GROUP_001</name>
  <positive>true</positive>
  <enforcing>true</enforcing>
</affinity_group>
```

Table 6.11. Parameters summary

Name	Type	Direction	Summary
group	AffinityGroup	In/Out	The affinity group object to create.

6.4.2. list GET

List existing affinity groups.

Table 6.12. Parameters summary

Name	Type	Direction	Summary
groups	AffinityGroup []	Out	The list of existing affinity groups.
max	Integer	In	Sets the maximum number of affinity groups to return.

6.4.2.1. max

Sets the maximum number of affinity groups to return. If not specified all the affinity groups are returned.

6.5. AFFINITYLABEL

The details of a single affinity label.

Table 6.13. Methods summary

Name	Summary
get	Retrieves the details of a label.
remove	Removes a label from the system and clears all assignments of the removed label.
update	Updates a label.

6.5.1. get GET

Retrieves the details of a label.

Table 6.14. Parameters summary

Name	Type	Direction	Summary
<code>label1</code>	AffinityLabel	Out	

6.5.2. remove DELETE

Removes a label from the system and clears all assignments of the removed label.

6.5.3. update PUT

Updates a label. This call will update all metadata, such as the name or description.

Table 6.15. Parameters summary

Name	Type	Direction	Summary
<code>label1</code>	AffinityLabel	In/Out	

6.6. AFFINITYLABELHOST

This service represents a host that has a specific label when accessed through the `affinitylabels/hosts` subcollection.

Table 6.16. Methods summary

Name	Summary
<code>get</code>	Retrieves details about a host that has this label assigned.
<code>remove</code>	Remove a label from a host.

6.6.1. get GET

Retrieves details about a host that has this label assigned.

Table 6.17. Parameters summary

Name	Type	Direction	Summary
<code>host</code>	Host	Out	

6.6.2. remove DELETE

Remove a label from a host.

6.7. AFFINITYLABELHOSTS

This service represents list of hosts that have a specific label when accessed through the affinitylabels/hosts subcollection.

Table 6.18. Methods summary

Name	Summary
add	Add a label to a host.
list	List all hosts with the label.

6.7.1. add POST

Add a label to a host.

Table 6.19. Parameters summary

Name	Type	Direction	Summary
host	Host	In/Out	

6.7.2. list GET

List all hosts with the label.

Table 6.20. Parameters summary

Name	Type	Direction	Summary
hosts	Host[]	Out	

6.8. AFFINITYLABELVM

This service represents a vm that has a specific label when accessed through the affinitylabels/vms subcollection.

Table 6.21. Methods summary

Name	Summary
get	Retrieves details about a vm that has this label assigned.
remove	Remove a label from a vm.

6.8.1. get GET

Retrieves details about a vm that has this label assigned.

Table 6.22. Parameters summary

Name	Type	Direction	Summary
vm	Vm	Out	

6.8.2. remove DELETE

Remove a label from a vm.

6.9. AFFINITYLABELVMS

This service represents list of vms that have a specific label when accessed through the affinitylabels/vms subcollection.

Table 6.23. Methods summary

Name	Summary
add	Add a label to a vm.
list	List all vms with the label.

6.9.1. add POST

Add a label to a vm.

Table 6.24. Parameters summary

Name	Type	Direction	Summary
vm	Vm	In/Out	

6.9.2. list GET

List all vms with the label.

Table 6.25. Parameters summary

Name	Type	Direction	Summary
vms	Vm[]	Out	

6.10. AFFINITYLABELS

Manages the affinity labels available in the system.

Table 6.26. Methods summary

Name	Summary
add	Creates a new label.
list	Lists all labels present in the system.

6.10.1. add POST

Creates a new label. The label is automatically attached to all entities mentioned in the vms or hosts lists.

Table 6.27. Parameters summary

Name	Type	Direction	Summary
label	AffinityLabel	In/Out	

6.10.2. list GET

Lists all labels present in the system.

Table 6.28. Parameters summary

Name	Type	Direction	Summary
labels	AffinityLabel[]	Out	
max	Integer	In	Sets the maximum number of labels to return.

6.10.2.1. max

Sets the maximum number of labels to return. If not specified all the labels are returned.

6.11. ASSIGNEDAFFINITYLABEL

This service represents one label to entity assignment when accessed using the entities/affinitylabels subcollection.

Table 6.29. Methods summary

Name	Summary
get	Retrieves details about the attached label.
remove	Removes the label from an entity.

6.11.1. get GET

Retrieves details about the attached label.

Table 6.30. Parameters summary

Name	Type	Direction	Summary
<code>label</code>	AffinityLabel	Out	

6.11.2. remove DELETE

Removes the label from an entity. Does not touch the label itself.

6.12. ASSIGNEDAFFINITYLABELS

This service is used to list and manipulate affinity labels that are assigned to supported entities when accessed using `entities/affinitylabels`.

Table 6.31. Methods summary

Name	Summary
<code>add</code>	Attaches a label to an entity.
<code>list</code>	Lists all labels that are attached to an entity.

6.12.1. add POST

Attaches a label to an entity.

Table 6.32. Parameters summary

Name	Type	Direction	Summary
<code>label</code>	AffinityLabel	In/Out	

6.12.2. list GET

Lists all labels that are attached to an entity.

Table 6.33. Parameters summary

Name	Type	Direction	Summary
<code>label</code>	AffinityLabel[]	Out	

6.13. ASSIGNEDCPUPROFILE

Table 6.34. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	

6.13.1. get GET

Table 6.35. Parameters summary

Name	Type	Direction	Summary
<code>profile</code>	CpuProfile	Out	

6.13.2. remove DELETE

Table 6.36. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the remove should be performed asynchronously.

6.14. ASSIGNEDCPUPROFILES

Table 6.37. Methods summary

Name	Summary
<code>add</code>	
<code>list</code>	

6.14.1. add POST

Table 6.38. Parameters summary

Name	Type	Direction	Summary
<code>profile</code>	CpuProfile	In/Out	

6.14.2. list GET

Table 6.39. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of profiles to return.
profiles	CpuProfile[]	Out	

6.14.2.1. max

Sets the maximum number of profiles to return. If not specified all the profiles are returned.

6.15. ASSIGNEDDISKPROFILE

Table 6.40. Methods summary

Name	Summary
get	
remove	

6.15.1. get GET

Table 6.41. Parameters summary

Name	Type	Direction	Summary
disk_profile	DiskProfile	Out	

6.15.2. remove DELETE

Table 6.42. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.16. ASSIGNEDDISKPROFILES

Table 6.43. Methods summary

Name	Summary
add	

Name	Summary
<code>list</code>	

6.16.1. add POST

Table 6.44. Parameters summary

Name	Type	Direction	Summary
<code>profile</code>	DiskProfile	In/Out	

6.16.2. list GET

Table 6.45. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of profiles to return.
<code>profiles</code>	DiskProfile[]	Out	

6.16.2.1. max

Sets the maximum number of profiles to return. If not specified all the profiles are returned.

6.17. ASSIGNEDPERMISSIONS

Represents a permission sub-collection, scoped by user, group or some entity type.

Table 6.46. Methods summary

Name	Summary
<code>add</code>	Assign a new permission to a user or group for specific entity.
<code>list</code>	List all the permissions of the specific entity.

6.17.1. add POST

Assign a new permission to a user or group for specific entity.

For example, to assign the **UserVmManager** role to the virtual machine with id **123** to the user with id **456** send a request like this:

```
POST /ovirt-engine/api/vms/123/permissions
```

With a request body like this:

```
<permission>
  <role>
    <name>UserVmManager</name>
  </role>
  <user id="456"/>
</permission>
```

To assign the **SuperUser** role to the system to the user with id **456** send a request like this:

```
POST /ovirt-engine/api/permissions
```

With a request body like this:

```
<permission>
  <role>
    <name>SuperUser</name>
  </role>
  <user id="456"/>
</permission>
```

If you want to assign permission to the group instead of the user please replace the **user** element with the **group** element with proper **id** of the group. For example to assign the **UserRole** role to the cluster with id **123** to the group with id **789** send a request like this:

```
POST /ovirt-engine/api/clusters/123/permissions
```

With a request body like this:

```
<permission>
  <role>
    <name>UserRole</name>
  </role>
  <group id="789"/>
</permission>
```

Table 6.47. Parameters summary

Name	Type	Direction	Summary
permission	Permission	In/Out	The permission.

6.17.2. list GET

List all the permissions of the specific entity.

For example to list all the permissions of the cluster with id **123** send a request like this:

```
GET /ovirt-engine/api/clusters/123/permissions
```

```

<permissions>
  <permission id="456">
    <cluster id="123"/>
    <role id="789"/>
    <user id="451"/>
  </permission>
  <permission id="654">
    <cluster id="123"/>
    <role id="789"/>
    <group id="127"/>
  </permission>
</permissions>

```

Table 6.48. Parameters summary

Name	Type	Direction	Summary
permissions	Permission[]	Out	The list of permissions.

6.18. ASSIGNEDROLES

Represents a roles sub-collection, for example scoped by user.

Table 6.49. Methods summary

Name	Summary
list	

6.18.1. list GET

Table 6.50. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of roles to return.
roles	Role[]	Out	

6.18.1.1. max

Sets the maximum number of roles to return. If not specified all the roles are returned.

6.19. ASSIGNEDTAG

A service to manage assignment of specific tag to specific entities in system.

Table 6.51. Methods summary

Name	Summary
get	Gets the information about the assigned tag.
remove	Unassign tag from specific entity in the system.

6.19.1. get GET

Gets the information about the assigned tag.

For example to retrieve the information about the tag with the id **456** which is assigned to virtual machine with id **123** send a request like this:

```
GET /ovirt-engine/api/vms/123/tags/456
```

```
<tag href="/ovirt-engine/api/tags/456" id="456">
  <name>root</name>
  <description>root</description>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</tag>
```

Table 6.52. Parameters summary

Name	Type	Direction	Summary
tag	Tag	Out	The assigned tag.

6.19.2. remove DELETE

Unassign tag from specific entity in the system.

For example to unassign the tag with id **456** from virtual machine with id **123** send a request like this:

```
DELETE /ovirt-engine/api/vms/123/tags/456
```

Table 6.53. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.20. ASSIGNEDTAGS

A service to manage collection of assignment of tags to specific entities in system.

Table 6.54. Methods summary

Name	Summary
add	Assign tag to specific entity in the system.
list	List all tags assigned to the specific entity.

6.20.1. add POST

Assign tag to specific entity in the system.

For example to assign tag **mytag** to virtual machine with the id **123** send a request like this:

```
POST /ovirt-engine/api/vms/123/tags
```

With a request body like this:

```
<tag>
  <name>mytag</name>
</tag>
```

Table 6.55. Parameters summary

Name	Type	Direction	Summary
tag	Tag	In/Out	The assigned tag.

6.20.2. list GET

List all tags assigned to the specific entity.

For example to list all the tags of the virtual machine with id **123** send a request like this:

```
GET /ovirt-engine/api/vms/123/tags
```

```
<tags>
  <tag href="/ovirt-engine/api/tags/222" id="222">
    <name>mytag</name>
    <description>mytag</description>
    <vm href="/ovirt-engine/api/vms/123" id="123"/>
  </tag>
</tags>
```

Table 6.56. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of tags to return.
tags	Tag[]	Out	The list of assigned tags.

6.20.2.1. max

Sets the maximum number of tags to return. If not specified all the tags are returned.

6.21. ASSIGNEDVNICPROFILE

Table 6.57. Methods summary

Name	Summary
get	
remove	

6.21.1. get GET

Table 6.58. Parameters summary

Name	Type	Direction	Summary
profile	VnicProfile	Out	

6.21.2. remove DELETE

Table 6.59. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.22. ASSIGNEDVNICPROFILES

Table 6.60. Methods summary

Name	Summary
add	
list	

6.22.1. add POST

Table 6.61. Parameters summary

Name	Type	Direction	Summary
profile	VnicProfile	In/Out	

6.22.2. list GET

Table 6.62. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of profiles to return.
profiles	VnicProfile[]	Out	

6.22.2.1. max

Sets the maximum number of profiles to return. If not specified all the profiles are returned.

6.23. ATTACHEDSTORAGEDOMAIN

Table 6.63. Methods summary

Name	Summary
activate	This operation activates an attached storage domain.
deactivate	This operation deactivates an attached storage domain.
get	
remove	

6.23.1. activate POST

This operation activates an attached storage domain. Once the storage domain is activated it is ready for use with the data center.

```
POST /ovirt-engine/api/datacenters/123/storagedomains/456/activate
```

The activate action does not take any action specific parameters, so the request body should contain an empty **action**:

```
<action/>
```

Table 6.64. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the activation should be performed asynchronously.

6.23.2. deactivate POST

This operation deactivates an attached storage domain. Once the storage domain is deactivated it will not be used with the data center.

```
POST /ovirt-engine/api/datacenters/123/storagedomains/456/deactivate
```

The deactivate action does not take any action specific parameters, so the request body should contain an empty **action**:

```
<action/>
```

Table 6.65. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the deactivation should be performed asynchronously.

6.23.3. get GET

Table 6.66. Parameters summary

Name	Type	Direction	Summary
storage_domain	StorageDomain	Out	

6.23.4. remove DELETE

Table 6.67. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.24. ATTACHEDSTORAGEDOMAINS

Table 6.68. Methods summary

Name	Summary
<code>add</code>	
<code>list</code>	

6.24.1. add POST

Table 6.69. Parameters summary

Name	Type	Direction	Summary
<code>storage_domain</code>	StorageDomain	In/Out	

6.24.2. list GET

Table 6.70. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of storage domains to return.
<code>storage_domains</code>	StorageDomain[]	Out	

6.24.2.1. max

Sets the maximum number of storage domains to return. If not specified all the storage domains are returned.

6.25. BALANCE

Table 6.71. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	

6.25.1. get GET

Table 6.72. Parameters summary

Name	Type	Direction	Summary
balance	Balance	Out	
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

6.25.2. remove DELETE

Table 6.73. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.26. BALANCES

Table 6.74. Methods summary

Name	Summary
add	
list	

6.26.1. add POST

Table 6.75. Parameters summary

Name	Type	Direction	Summary
balance	Balance	In/Out	

6.26.2. list GET

Table 6.76. Parameters summary

Name	Type	Direction	Summary
balances	Balance[]	Out	
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
max	Integer	In	Sets the maximum number of balances to return.

6.26.2.1. max

Sets the maximum number of balances to return. If not specified all the balances are returned.

6.27. BOOKMARK

A service to manage a bookmark.

Table 6.77. Methods summary

Name	Summary
get	Get a bookmark.
remove	Remove a bookmark.
update	Update a bookmark.

6.27.1. get GET

Get a bookmark.

An example for getting a bookmark:

```
GET /ovirt-engine/api/bookmarks/123
```

```
<bookmark href="/ovirt-engine/api/bookmarks/123" id="123">
  <name>example_vm</name>
  <value>vm: name=example*</value>
</bookmark>
```

Table 6.78. Parameters summary

Name	Type	Direction	Summary
bookmark	Bookmark	Out	The requested bookmark.

6.27.2. remove DELETE

Remove a bookmark.

An example for removing a bookmark:

```
DELETE /ovirt-engine/api/bookmarks/123
```

Table 6.79. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.27.3. update PUT

Update a bookmark.

An example for updating a bookmark:

```
PUT /ovirt-engine/api/bookmarks/123
```

With the request body:

```
<bookmark>
  <name>new_example_vm</name>
  <value>vm: name=new_example*</value>
</bookmark>
```

Table 6.80. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
bookmark	Bookmark	In/Out	The updated bookmark.

6.28. BOOKMARKS

A service to manage bookmarks.

Table 6.81. Methods summary

Name	Summary
add	Adding a new bookmark.
list	Listing all the available bookmarks.

6.28.1. add POST

Adding a new bookmark.

Example of adding a bookmark:

```
POST /ovirt-engine/api/bookmarks
```

```
<bookmark>
  <name>new_example_vm</name>
  <value>vm: name=new_example*</value>
</bookmark>
```

Table 6.82. Parameters summary

Name	Type	Direction	Summary
bookmark	Bookmark	In/Out	The added bookmark.

6.28.2. list GET

Listing all the available bookmarks.

Example of listing bookmarks:

```
GET /ovirt-engine/api/bookmarks
```

```
<bookmarks>
  <bookmark href="/ovirt-engine/api/bookmarks/123" id="123">
    <name>database</name>
    <value>vm: name=database*</value>
  </bookmark>
  <bookmark href="/ovirt-engine/api/bookmarks/456" id="456">
    <name>example</name>
    <value>vm: name=example*</value>
  </bookmark>
</bookmarks>
```

Table 6.83. Parameters summary

Name	Type	Direction	Summary
bookmarks	Bookmark[]	Out	The list of available bookmarks.
max	Integer	In	Sets the maximum number of bookmarks to return.

6.28.2.1. max

Sets the maximum number of bookmarks to return. If not specified all the bookmarks are returned.

6.29. CLUSTER

A service to manage specific cluster.

Table 6.84. Methods summary

Name	Summary
get	Get information about the cluster.
remove	Removes cluster from the system.
resetemulate dmachine	
update	Updates information about the cluster.

6.29.1. get GET

Get information about the cluster.

An example of getting a cluster:

```
GET /ovirt-engine/api/clusters/123
```

```
<cluster href="/ovirt-engine/api/clusters/123" id="123">
  <actions>
    <link href="/ovirt-engine/api/clusters/123/resetemulatedmachine"
rel="resetemulatedmachine"/>
  </actions>
  <name>Default</name>
  <description>The default server cluster</description>
  <link href="/ovirt-engine/api/clusters/123/networks" rel="networks"/>
  <link href="/ovirt-engine/api/clusters/123/permissions"
rel="permissions"/>
  <link href="/ovirt-engine/api/clusters/123/glustervolumes"
rel="glustervolumes"/>
  <link href="/ovirt-engine/api/clusters/123/glusterhooks"
rel="glusterhooks"/>
  <link href="/ovirt-engine/api/clusters/123/affinitygroups"
rel="affinitygroups"/>
  <link href="/ovirt-engine/api/clusters/123/cpuprofiles"
rel="cpuprofiles"/>
  <ballooning_enabled>>false</ballooning_enabled>
  <cpu>
    <architecture>x86_64</architecture>
    <type>Intel Penryn Family</type>
  </cpu>
  <error_handling>
    <on_error>migrate</on_error>
  </error_handling>
  <fencing_policy>
    <enabled>>true</enabled>
    <skip_if_connectivity_broken>
      <enabled>>false</enabled>
      <threshold>50</threshold>
    </skip_if_connectivity_broken>
    <skip_if_sd_active>
```

```

    <enabled>>false</enabled>
  </skip_if_sd_active>
</fencing_policy>
<gluster_service>>false</gluster_service>
<ha_reservation>>false</ha_reservation>
<ksm>
  <enabled>>true</enabled>
  <merge_across_nodes>>true</merge_across_nodes>
</ksm>
<maintenance_reason_required>>false</maintenance_reason_required>
<memory_policy>
  <over_commit>
    <percent>100</percent>
  </over_commit>
  <transparent_hugepages>
    <enabled>>true</enabled>
  </transparent_hugepages>
</memory_policy>
<migration>
  <auto_converge>inherit</auto_converge>
  <bandwidth>
    <assignment_method>auto</assignment_method>
  </bandwidth>
  <compressed>inherit</compressed>
</migration>
<optional_reason>>false</optional_reason>
<required_rng_sources>
  <required_rng_source>random</required_rng_source>
</required_rng_sources>
<scheduling_policy href="/ovirt-engine/api/schedulingpolicies/456"
id="456"/>
  <threads_as_cores>>false</threads_as_cores>
  <trusted_service>>false</trusted_service>
  <tunnel_migration>>false</tunnel_migration>
  <version>
    <major>4</major>
    <minor>0</minor>
  </version>
  <virt_service>>true</virt_service>
  <data_center href="/ovirt-engine/api/datacenters/111" id="111"/>
</cluster>

```

Table 6.85. Parameters summary

Name	Type	Direction	Summary
cluster	Cluster	Out	
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

6.29.2. remove DELETE

Removes cluster from the system.

```
DELETE /ovirt-engine/api/clusters/00000000-0000-0000-0000-000000000000
```

Table 6.86. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.29.3. resetemulatedmachine POST

Table 6.87. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the reset should be performed asynchronously.

6.29.4. update PUT

Updates information about the cluster.

Only specified fields are updated, others remain unchanged.

E.g. update cluster's CPU:

```
PUT /ovirt-engine/api/clusters/123
```

With request body like:

```
<cluster>
  <cpu>
    <type>Intel Haswell-noTSX Family</type>
  </cpu>
</cluster>
```

Table 6.88. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
cluster	Cluster	In/Out	

6.30. CLUSTERLEVEL

Provides information about a specific cluster level. See the [ClusterLevels](#) service for more information.

Table 6.89. Methods summary

Name	Summary
get	Provides the information about the capabilities of the specific cluster level managed by this service.

6.30.1. get GET

Provides the information about the capabilities of the specific cluster level managed by this service.

For example, to find what CPU types are supported by level 3.6 you can send a request like this:

```
GET /ovirt-engine/api/clusterlevels/3.6
```

That will return a [ClusterLevel](#) object containing the supported CPU types, and other information which describes the cluster level:

```
<cluster_level id="3.6">
  <cpu_types>
    <cpu_type>
      <name>Intel Conroe Family</name>
      <level>3</level>
      <architecture>x86_64</architecture>
    </cpu_type>
    ...
  </cpu_types>
  <permits>
    <permit id="1">
      <name>create_vm</name>
      <administrative>>false</administrative>
    </permit>
    ...
  </permits>
</cluster_level>
```

Table 6.90. Parameters summary

Name	Type	Direction	Summary
level	ClusterLevel	Out	Retrieved cluster level.

6.31. CLUSTERLEVELS

Provides information about the capabilities of different cluster levels supported by the engine. Version 4.0 of the engine supports levels 4.0 and 3.6. Each of these levels support different sets of CPU types, for example. This service provides that information.

Table 6.91. Methods summary

Name	Summary
list	Lists the cluster levels supported by the system.

6.31.1. list GET

Lists the cluster levels supported by the system.

```
GET /ovirt-engine/api/clusterlevels
```

This will return a list of available cluster levels.

```
<cluster_levels>
  <cluster_level id="4.0">
    ...
  </cluster_level>
  ...
</cluster_levels>
```

Table 6.92. Parameters summary

Name	Type	Direction	Summary
levels	ClusterLevel []	Out	Retrieved cluster levels.

6.32. CLUSTERNETWORK

A service to manage a specific cluster network.

Table 6.93. Methods summary

Name	Summary
get	Retrieves the cluster network details.
remove	Unassigns the network from a cluster.
update	Updates the network in the cluster.

6.32.1. get GET

Retrieves the cluster network details.

Table 6.94. Parameters summary

Name	Type	Direction	Summary
network	Network	Out	The cluster network.

6.32.2. remove DELETE

Unassigns the network from a cluster.

6.32.3. update PUT

Updates the network in the cluster.

Table 6.95. Parameters summary

Name	Type	Direction	Summary
network	Network	In/Out	The cluster network.

6.33. CLUSTER NETWORKS

A service to manage cluster networks.

Table 6.96. Methods summary

Name	Summary
add	Assigns the network to a cluster.
list	Lists the networks that are assigned to the cluster.

6.33.1. add POST

Assigns the network to a cluster.

Post a request like in the example below to assign the network to a cluster:

```
POST /ovirt-engine/api/clusters/123/networks
```

Use the following example in its body:

```
<network id="123" />
```

Table 6.97. Parameters summary

Name	Type	Direction	Summary
network	Network	In/Out	The network object to be assigned to the cluster.

6.33.2. list GET

Lists the networks that are assigned to the cluster.

Table 6.98. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of networks to return.
networks	Network[]	Out	The list of networks that are assigned to the cluster.

6.33.2.1. max

Sets the maximum number of networks to return. If not specified, all the networks are returned.

6.34. CLUSTERS

A service to manage clusters.

Table 6.99. Methods summary

Name	Summary
add	Creates a new cluster.
list	

6.34.1. add POST

Creates a new cluster.

This requires the **name**, **cpu.type** and **data_center** attributes. Identify the data center with either the **id** or **name** attributes.

```
POST /ovirt-engine/api/clusters
```

With a request body like this:

```
<cluster>
  <name>mycluster</name>
  <cpu>
    <type>Intel Penryn Family</type>
  </cpu>
  <data_center id="123"/>
</cluster>
```

Table 6.100. Parameters summary

Name	Type	Direction	Summary
c luster	Cluster	In/Out	

6.34.2. list GET

Table 6.101. Parameters summary

Name	Type	Direction	Summary
case_sens itive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
c lusters	Cluster[]	Out	
f ilter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
m ax	Integer	In	Sets the maximum number of clusters to return.
s earch	String	In	A query string used to restrict the returned clusters.

6.34.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.34.2.2. max

Sets the maximum number of clusters to return. If not specified all the clusters are returned.

6.35. COPYABLE

Table 6.102. Methods summary

Name	Summary
copy	

6.35.1. copy POST

Table 6.103. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the copy should be performed asynchronously.

6.36. CPUPROFILE

Table 6.104. Methods summary

Name	Summary
get	
remove	
update	

6.36.1. get GET

Table 6.105. Parameters summary

Name	Type	Direction	Summary
profile	CpuProfile	Out	

6.36.2. remove DELETE

Table 6.106. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.36.3. update PUT

Table 6.107. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
profile	CpuProfile	In/Out	

6.37. CPUPROFILES

Table 6.108. Methods summary

Name	Summary
<code>add</code>	
<code>list</code>	

6.37.1. add POST

Table 6.109. Parameters summary

Name	Type	Direction	Summary
<code>profile</code>	CpuProfile	In/Out	

6.37.2. list GET

Table 6.110. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of profiles to return.
<code>profile</code>	CpuProfile[]	Out	

6.37.2.1. max

Sets the maximum number of profiles to return. If not specified all the profiles are returned.

6.38. DATACENTER

A service to manage a data center.

Table 6.111. Methods summary

Name	Summary
<code>get</code>	Get a data center.
<code>remove</code>	Removes the data center.
<code>update</code>	Updates the data center.

6.38.1. get GET

Get a data center.

An example of getting a data center:

```
GET /ovirt-engine/api/datacenters/123
```

```
<data_center href="/ovirt-engine/api/datacenters/123" id="123">
  <name>Default</name>
  <description>The default Data Center</description>
  <link href="/ovirt-engine/api/datacenters/123/clusters" rel="clusters"/>
  <link href="/ovirt-engine/api/datacenters/123/storagedomains"
rel="storagedomains"/>
  <link href="/ovirt-engine/api/datacenters/123/permissions"
rel="permissions"/>
  <link href="/ovirt-engine/api/datacenters/123/networks" rel="networks"/>
  <link href="/ovirt-engine/api/datacenters/123/quotas" rel="quotas"/>
  <link href="/ovirt-engine/api/datacenters/123/qoss" rel="qoss"/>
  <link href="/ovirt-engine/api/datacenters/123/iscsibonds"
rel="iscsibonds"/>
  <local>>false</local>
  <quota_mode>disabled</quota_mode>
  <status>up</status>
  <storage_format>v3</storage_format>
  <supported_versions>
    <version>
      <major>4</major>
      <minor>0</minor>
    </version>
  </supported_versions>
  <version>
    <major>4</major>
    <minor>0</minor>
  </version>
  <mac_pool href="/ovirt-engine/api/macpools/456" id="456"/>
</data_center>
```

Table 6.112. Parameters summary

Name	Type	Direction	Summary
data_center	DataCenter	Out	
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

6.38.2. remove DELETE

Removes the data center.

```
DELETE /ovirt-engine/api/datacenters/123
```


Without any special parameters, the storage domains attached to the data center are detached and then removed from the storage. If something fails when performing this operation, for example if there is no host available to remove the storage domains from the storage, the complete operation will fail.

If the **force** parameter is **true** then the operation will always succeed, even if something fails while removing one storage domain, for example. The failure is just ignored and the data center is removed from the database anyway.

Table 6.113. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.
force	Boolean	In	Indicates if the operation should succeed, and the storage domain removed from the database, even if something fails during the operation.

6.38.2.1. force

Indicates if the operation should succeed, and the storage domain removed from the database, even if something fails during the operation.

This parameter is optional, and the default value is **false**.

6.38.3. update PUT

Updates the data center.

The **name**, **description**, **storage_type**, **version**, **storage_format** and **mac_pool** elements are updatable post-creation. For example, to change the name and description of data center **123** send a request like this:

```
PUT /ovirt-engine/api/datacenters/123
```

With a request body like this:

```
<data_center>
  <name>myupdatedname</name>
  <description>An updated description for the data center</description>
</data_center>
```

Table 6.114. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.

Name	Type	Direction	Summary
data_center	DataCenter	In/Out	The data center that is being updated.

6.39. DATACENTERNETWORK

A service to manage a specific data center network.

Table 6.115. Methods summary

Name	Summary
get	Retrieves the data center network details.
remove	Removes the network.
update	Updates the network in the data center.

6.39.1. get GET

Retrieves the data center network details.

Table 6.116. Parameters summary

Name	Type	Direction	Summary
network	Network	Out	The data center network.

6.39.2. remove DELETE

Removes the network.

6.39.3. update PUT

Updates the network in the data center.

Table 6.117. Parameters summary

Name	Type	Direction	Summary
network	Network	In/Out	The data center network.

6.40. DATACENTERNETWORKS

A service to manage data center networks.

Table 6.118. Methods summary

Name	Summary
add	Create a new network in a data center.
list	Lists networks in the data center.

6.40.1. add POST

Create a new network in a data center.

Post a request like in the example below to create a new network in a data center with an ID of **123**.

```
POST /ovirt-engine/api/datacenters/123/networks
```

Use the following example in its body:

```
<network>
  <name>mynetwork</name>
</network>
```

Table 6.119. Parameters summary

Name	Type	Direction	Summary
network	Network	In/Out	The network object to be created in the data center.

6.40.2. list GET

Lists networks in the data center.

Table 6.120. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of networks to return.
networks	Network[]	Out	The list of networks which are in the data center.

6.40.2.1. max

Sets the maximum number of networks to return. If not specified, all the networks are returned.

6.41. DATACENTERS

A service to manage data centers.

Table 6.121. Methods summary

Name	Summary
add	Creates a new data center.
list	Lists the data centers.

6.41.1. add POST

Creates a new data center.

Creation of a new data center requires the **name** and **local** elements. For example, to create a data center named **mydc** that uses shared storage (NFS, iSCSI or Fibre Channel) send a request like this:

```
POST /ovirt-engine/api/datacenters
```

With a request body like this:

```
<data_center>
  <name>mydc</name>
  <local>false</local>
</data_center>
```

Table 6.122. Parameters summary

Name	Type	Direction	Summary
data_center	DataCenter	In/Out	The data center that is being added.

6.41.2. list GET

Lists the data centers.

The following request retrieves a representation of the data centers:

```
GET /ovirt-engine/api/datacenters
```

The above request performed with **curl**:

```
curl \
  --request GET \
  --cacert /etc/pki/ovirt-engine/ca.pem \
  --header "Version: 4" \
  --header "Accept: application/xml" \
  --user "admin@internal:mypassword" \
  https://myengine.example.com/ovirt-engine/api/datacenters
```

This is what an example response could look like:

```
<data_center href="/ovirt-engine/api/datacenters/123" id="123">
```

```

<name>Default</name>
<description>The default Data Center</description>
<link href="/ovirt-engine/api/datacenters/123/networks" rel="networks"/>
<link href="/ovirt-engine/api/datacenters/123/storagedomains"
rel="storagedomains"/>
<link href="/ovirt-engine/api/datacenters/123/permissions"
rel="permissions"/>
<link href="/ovirt-engine/api/datacenters/123/clusters" rel="clusters"/>
<link href="/ovirt-engine/api/datacenters/123/qoss" rel="qoss"/>
<link href="/ovirt-engine/api/datacenters/123/iscsibonds"
rel="iscsibonds"/>
<link href="/ovirt-engine/api/datacenters/123/quotas" rel="quotas"/>
<local>>false</local>
<quota_mode>disabled</quota_mode>
<status>up</status>
<supported_versions>
  <version>
    <major>4</major>
    <minor>0</minor>
  </version>
</supported_versions>
<version>
  <major>4</major>
  <minor>0</minor>
</version>
</data_center>

```

Note the **id** code of your **Default** data center. This code identifies this data center in relation to other resources of your virtual environment.

The data center also contains a link to the storage domains collection. The data center uses this collection to attach storage domains from the storage domains main collection.

Table 6.123. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
data_centers	DataCenter[]	Out	
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
max	Integer	In	Sets the maximum number of data centers to return.
search	String	In	A query string used to restrict the returned data centers.

6.41.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.41.2.2. max

Sets the maximum number of data centers to return. If not specified all the data centers are returned.

6.42. DISK

Manages a single disk.

Table 6.124. Methods summary

Name	Summary
copy	This operation copies a disk to the specified storage domain.
export	
get	
move	Moves a disk to another storage domain.
remove	
sparsify	Sparsify the disk.

6.42.1. copy POST

This operation copies a disk to the specified storage domain.

For example, copy of a disk can be facilitated using the following request:

```
POST /ovirt-engine/api/disks/123/copy
```

With a request body like this:

```
<action>
  <storage_domain id="456"/>
  <disk>
    <name>mydisk</name>
  </disk>
</action>
```

Table 6.125. Parameters summary

Name	Type	Direction	Summary
------	------	-----------	---------

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the copy should be performed asynchronously.
disk	Disk	In	
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
storage_domain	StorageDomain	In	

6.42.2. export POST

Table 6.126. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the export should be performed asynchronously.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
storage_domain	StorageDomain	In	

6.42.3. get GET

Table 6.127. Parameters summary

Name	Type	Direction	Summary
disk	Disk	Out	

6.42.4. move POST

Moves a disk to another storage domain.

For example, to move the disk with identifier **123** to a storage domain with identifier **456** send the following request:

```
POST /ovirt-engine/api/disks/123/move
```

With the following request body:

```
<action>
  <storage_domain id="456"/>
</action>
```

Table 6.128. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the move should be performed asynchronously.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
storage_domain	StorageDomain	In	

6.42.5. remove DELETE

Table 6.129. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.42.6. sparsify POST

Sparsify the disk.

Sparsification frees space in the disk image that is not used by its filesystem. As a result, the image will occupy less space on the storage.

Currently sparsification works only on disks without snapshots. Disks having derived disks are also not allowed.

6.43. DISKATTACHMENT

This service manages the attachment of a disk to a virtual machine.

Table 6.130. Methods summary

Name	Summary
get	Returns the details of the attachment, including the bootable flag and link to the disk.
remove	Removes the disk attachment.
update	Update the disk attachment and the disk properties within it.

6.43.1. get GET

Returns the details of the attachment, including the bootable flag and link to the disk.

An example of getting a disk attachment:

```
GET /ovirt-engine/api/vms/123/diskattachments/456
```

```
<disk_attachment href="/ovirt-engine/api/vms/123/diskattachments/456"
id="456">
  <active>true</active>
  <bootable>true</bootable>
  <interface>virtio</interface>
  <disk href="/ovirt-engine/api/disks/456" id="456"/>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</disk_attachment>
```

Table 6.131. Parameters summary

Name	Type	Direction	Summary
attachmen t	DiskAttachm ent	Out	

6.43.2. remove DELETE

Removes the disk attachment.

This will only detach the disk from the virtual machine, but won't remove it from the system, unless the **detach_only** parameter is **false**.

An example of removing a disk attachment:

```
DELETE /ovirt-engine/api/vms/123/diskattachments/456?detach_only=true
```

Table 6.132. Parameters summary

Name	Type	Direction	Summary
detach_on ly	Boolean	In	Indicates if the disk should only be detached from the virtual machine, but not removed from the system.

6.43.2.1. detach_only

Indicates if the disk should only be detached from the virtual machine, but not removed from the system. The default value is **true**, which won't remove the disk from the system.

6.43.3. update PUT

Update the disk attachment and the disk properties within it.

```

PUT /vms/{vm:id}/disksattachments/{attachment:id}
<disk_attachment>
  <bootable>true</bootable>
  <interface>ide</interface>
  <active>true</active>
  <disk>
    <name>mydisk</name>
    <provisioned_size>1024</provisioned_size>
    ...
  </disk>
</disk_attachment>

```

Table 6.133. Parameters summary

Name	Type	Direction	Summary
disk_attachment	DiskAttachment	In/Out	

6.44. DISKATTACHMENTS

This service manages the set of disks attached to a virtual machine. Each attached disk is represented by a [DiskAttachment](#), containing the bootable flag, the disk interface and the reference to the disk.

Table 6.134. Methods summary

Name	Summary
add	Adds a new disk attachment to the virtual machine.
list	List the disk that are attached to the virtual machine.

6.44.1. add POST

Adds a new disk attachment to the virtual machine. The **attachment** parameter can contain just a reference, if the disk already exists:

```

<disk_attachment>
  <bootable>true</bootable>
  <pass_discard>true</pass_discard>
  <interface>ide</interface>
  <active>true</active>
  <disk id="123"/>
</disk_attachment>

```

Or it can contain the complete representation of the disk, if the disk doesn't exist yet:

```

<disk_attachment>
  <bootable>true</bootable>
  <pass_discard>true</pass_discard>
  <interface>ide</interface>

```

```

<active>>true</active>
<disk>
  <name>mydisk</name>
  <provisioned_size>1024</provisioned_size>
  ...
</disk>
</disk_attachment>

```

In this case the disk will be created and then attached to the virtual machine.

In both cases, use the following URL for a virtual machine with an id **345**:

```
POST /ovirt-engine/api/vms/345/diskattachments
```



IMPORTANT

The server accepts requests that don't contain the **active** attribute, but the effect is undefined. In some cases the disk will be automatically activated and in other cases it won't. To avoid issues it is strongly recommended to always include the **active** attribute with the desired value.

Table 6.135. Parameters summary

Name	Type	Direction	Summary
attachmen t	DiskAttachm ent	In/Out	

6.44.2. list GET

List the disk that are attached to the virtual machine.

Table 6.136. Parameters summary

Name	Type	Direction	Summary
attachmen ts	DiskAttachm ent[]	Out	

6.45. DISKPROFILE

Table 6.137. Methods summary

Name	Summary
get	
remove	

Name	Summary
update	

6.45.1. get GET

Table 6.138. Parameters summary

Name	Type	Direction	Summary
profile	DiskProfile	Out	

6.45.2. remove DELETE

Table 6.139. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.45.3. update PUT

Table 6.140. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
profile	DiskProfile	In/Out	

6.46. DISKPROFILES

Table 6.141. Methods summary

Name	Summary
add	
list	

6.46.1. add POST

Table 6.142. Parameters summary

Name	Type	Direction	Summary
<code>profile</code>	DiskProfile	In/Out	

6.46.2. list GET

Table 6.143. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of profiles to return.
<code>profile</code>	DiskProfile[]	Out	

6.46.2.1. max

Sets the maximum number of profiles to return. If not specified all the profiles are returned.

6.47. DISKSNAPSHOT

Table 6.144. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	

6.47.1. get GET

Table 6.145. Parameters summary

Name	Type	Direction	Summary
<code>snapshot</code>	DiskSnapshot	Out	

6.47.2. remove DELETE

Table 6.146. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the remove should be performed asynchronously.

6.48. DISKSNAPSHOTS

Table 6.147. Methods summary

Name	Summary
<code>list</code>	

6.48.1. list GET

Table 6.148. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of snapshots to return.
<code>snapshots</code>	DiskSnapshot[]	Out	

6.48.1.1. max

Sets the maximum number of snapshots to return. If not specified all the snapshots are returned.

6.49. DISKS

Manages the collection of disks available in the system.

Table 6.149. Methods summary

Name	Summary
<code>add</code>	Adds a new floating disk.
<code>list</code>	Get list of disks.

6.49.1. add POST

Adds a new floating disk.

There are three types of disks that can be added - disk image, direct LUN and [Cinder](#) disk.

Adding a new image disk:

When creating a new floating image [Disk](#), the API requires the `storage_domain`, `provisioned_size` and `format` attributes.

To create a new floating image disk with specified `provisioned_size`, `format` and `name` on a storage domain with an id `123`, send a request as follows:

```
POST /ovirt-engine/api/disks
```

With a request body as follows:

```
<disk>
  <storage_domains>
    <storage_domain id="123"/>
  </storage_domains>
  <name>mydisk</name>
  <provisioned_size>1048576</provisioned_size>
  <format>cow</format>
</disk>
```

Adding a new direct LUN disk:

When adding a new floating direct LUN via the API, there are two flavors that can be used:

1. With a **host** element - in this case, the host is used for sanity checks (e.g., that the LUN is visible) and to retrieve basic information about the LUN (e.g., size and serial).
2. Without a **host** element - in this case, the operation is a database-only operation, and the storage is never accessed.

To create a new floating direct LUN disk with a **host** element with an id **123**, specified **alias**, **type** and **logical_unit** with an id **456** (that has the attributes **address**, **port** and **target**), send a request as follows:

```
POST /ovirt-engine/api/disks
```

With a request body as follows:

```
<disk>
  <alias>mylun</alias>
  <lun_storage>
    <host id="123"/>
    <type>iscsi</type>
    <logical_units>
      <logical_unit id="456">
        <address>10.35.10.20</address>
        <port>3260</port>
        <target>iqn.2017-01.com.myhost:444</target>
      </logical_unit>
    </logical_units>
  </lun_storage>
</disk>
```

To create a new floating direct LUN disk without using a host, remove the **host** element.

Adding a new Cinder disk:

To create a new floating Cinder disk, send a request as follows:

```
POST /ovirt-engine/api/disks
```

With a request body as follows:

-

```

<disk>
  <openstack_volume_type>
    <name>myceph</name>
  </openstack_volume_type>
  <storage_domains>
    <storage_domain>
      <name>cinderDomain</name>
    </storage_domain>
  </storage_domains>
  <provisioned_size>1073741824</provisioned_size>
  <interface>virtio</interface>
  <format>raw</format>
</disk>

```

Table 6.150. Parameters summary

Name	Type	Direction	Summary
disk	Disk	In/Out	The disk.

6.49.2. list GET

Get list of disks.

```
GET /ovirt-engine/api/disks
```

You will get a XML response which will look like this one:

```

<disks>
  <disk id="123">
    <actions>...</actions>
    <name>MyDisk</name>
    <description>MyDisk description</description>
    <link href="/ovirt-engine/api/disks/123/permissions"
rel="permissions"/>
    <link href="/ovirt-engine/api/disks/123/statistics" rel="statistics"/>
    <actual_size>5345845248</actual_size>
    <alias>MyDisk alias</alias>
    ...
    <status>ok</status>
    <storage_type>image</storage_type>
    <wipe_after_delete>false</wipe_after_delete>
    <disk_profile id="123"/>
    <quota id="123"/>
    <storage_domains>...</storage_domains>
  </disk>
  ...
</disks>

```

Table 6.151. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
disks	Disk[]	Out	List of retrieved disks.
max	Integer	In	Sets the maximum number of disks to return.
search	String	In	A query string used to restrict the returned disks.
unregistered	Boolean	In	Indicates whether to retrieve a list of registered or unregistered disks in the storage domain.

6.49.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.49.2.2. max

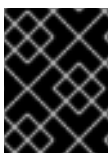
Sets the maximum number of disks to return. If not specified all the disks are returned.

6.49.2.3. unregistered

Indicates whether to retrieve a list of registered or unregistered disks in the storage domain. To get a list of unregistered disks in the storage domain the call should indicate the unregistered flag. For example, to get a list of unregistered disks the REST API call should look like this:

```
GET /ovirt-engine/api/storagedomains/123/disks?unregistered=true
```

The default value of the unregistered flag is **false**.



IMPORTANT

This parameter only applies to the **disks** sub-collection of attached storage domains, it does not apply to the top level **disks** collection.

6.50. DOMAIN

A service to view details of an authentication domain in the system.

Table 6.152. Methods summary

Name	Summary
get	Gets the authentication domain information.

6.50.1. get GET

Gets the authentication domain information.

Usage:

```
GET /ovirt-engine/api/domains/5678
```

Will return the domain information:

```
<domain href="/ovirt-engine/api/domains/5678" id="5678">
  <name>internal-authz</name>
  <link href="/ovirt-engine/api/domains/5678/users" rel="users"/>
  <link href="/ovirt-engine/api/domains/5678/groups" rel="groups"/>
  <link href="/ovirt-engine/api/domains/5678/users?search={query}"
rel="users/search"/>
  <link href="/ovirt-engine/api/domains/5678/groups?search={query}"
rel="groups/search"/>
</domain>
```

Table 6.153. Parameters summary

Name	Type	Direction	Summary
<code>domain</code>	Domain	Out	The authentication domain.

6.51. DOMAINGROUP

Table 6.154. Methods summary

Name	Summary
<code>get</code>	

6.51.1. get GET

Table 6.155. Parameters summary

Name	Type	Direction	Summary
<code>get</code>	Group	Out	

6.52. DOMAINGROUPS

Table 6.156. Methods summary

Name	Summary
<code>list</code>	

6.52.1. list GET

Table 6.157. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
groups	Group[]	Out	
max	Integer	In	Sets the maximum number of groups to return.
search	String	In	A query string used to restrict the returned groups.

6.52.1.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.52.1.2. max

Sets the maximum number of groups to return. If not specified all the groups are returned.

6.53. DOMAINUSER

A service to view a domain user in the system.

Table 6.158. Methods summary

Name	Summary
get	Gets the domain user information.

6.53.1. get GET

Gets the domain user information.

Usage:

```
GET /ovirt-engine/api/domains/5678/users/1234
```

Will return the domain user information:

```
<user href="/ovirt-engine/api/users/1234" id="1234">
  <name>admin</name>
  <namespace>*</namespace>
  <principal>admin</principal>
```

```

<user_name>admin@internal-authz</user_name>
<domain href="/ovirt-engine/api/domains/5678" id="5678">
  <name>internal-authz</name>
</domain>
<groups/>
</user>

```

Table 6.159. Parameters summary

Name	Type	Direction	Summary
user	User	Out	The domain user.

6.54. DOMAINUSERS

A service to list all domain users in the system.

Table 6.160. Methods summary

Name	Summary
list	List all the users in the domain.

6.54.1. list GET

List all the users in the domain.

Usage:

```
GET /ovirt-engine/api/domains/5678/users
```

Will return the list of users in the domain:

```

<users>
  <user href="/ovirt-engine/api/domains/5678/users/1234" id="1234">
    <name>admin</name>
    <namespace>*</namespace>
    <principal>admin</principal>
    <user_name>admin@internal-authz</user_name>
    <domain href="/ovirt-engine/api/domains/5678" id="5678">
      <name>internal-authz</name>
    </domain>
    <groups/>
  </user>
</users>

```

Table 6.161. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
max	Integer	In	Sets the maximum number of users to return.
search	String	In	A query string used to restrict the returned users.
users	User[]	Out	The list of users in the domain.

6.54.1.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.54.1.2. max

Sets the maximum number of users to return. If not specified all the users are returned.

6.55. DOMAINS

A service to list all authentication domains in the system.

Table 6.162. Methods summary

Name	Summary
list	List all the authentication domains in the system.

6.55.1. list GET

List all the authentication domains in the system.

Usage:

```
GET /ovirt-engine/api/domains
```

Will return the list of domains:

```
<domains>
  <domain href="/ovirt-engine/api/domains/5678" id="5678">
    <name>internal-authz</name>
    <link href="/ovirt-engine/api/domains/5678/users" rel="users"/>
    <link href="/ovirt-engine/api/domains/5678/groups" rel="groups"/>
    <link href="/ovirt-engine/api/domains/5678/users?search={query}"
```

```

rel="users/search"/>
  <link href="/ovirt-engine/api/domains/5678/groups?search={query}"
rel="groups/search"/>
</domain>
</domains>

```

Table 6.163. Parameters summary

Name	Type	Direction	Summary
domains	Domain[]	Out	The list of domains.
max	Integer	In	Sets the maximum number of domains to return.

6.55.1.1. max

Sets the maximum number of domains to return. If not specified all the domains are returned.

6.56. ENGINEKATELLOERRATA

A service to manage Katello errata assigned to the engine. The information is retrieved from Katello.

Table 6.164. Methods summary

Name	Summary
list	Retrieves the representation of the Katello errata.

6.56.1. list GET

Retrieves the representation of the Katello errata.

```
GET /ovirt-engine/api/katelloerrata
```

You will receive response in XML like this one:

```

<katello_errata>
  <katello_erratum href="/ovirt-engine/api/katelloerrata/123" id="123">
    <name>RHBA-2013:XYZ</name>
    <description>The description of the erratum</description>
    <title>some bug fix update</title>
    <type>bugfix</type>
    <issued>2013-11-20T02:00:00.000+02:00</issued>
    <solution>Few guidelines regarding the solution</solution>
    <summary>Updated packages that fix one bug are now available for
XYZ</summary>
    <packages>
      <package>
        <name>libipa_hbac-1.9.2-82.11.el6_4.i686</name>
      </package>
      ...
    </packages>
  </katello_erratum>
</katello_errata>

```

```

    </packages>
  </katello_erratum>
  ...
</katello_errata>

```

Table 6.165. Parameters summary

Name	Type	Direction	Summary
errata	KatelloErratum[]	Out	A representation of Katello errata.
max	Integer	In	Sets the maximum number of errata to return.

6.56.1.1. max

Sets the maximum number of errata to return. If not specified all the errata are returned.

6.57. EVENT

A service to manage an event in the system.

Table 6.166. Methods summary

Name	Summary
get	Get an event.
remove	Removes an event from internal audit log.

6.57.1. get GET

Get an event.

An example of getting an event:

```
GET /ovirt-engine/api/events/123
```

```

<event href="/ovirt-engine/api/events/123" id="123">
  <description>Host example.com was added by admin@internal-
  authz.</description>
  <code>42</code>
  <correlation_id>135</correlation_id>
  <custom_id>-1</custom_id>
  <flood_rate>30</flood_rate>
  <origin>oVirt</origin>
  <severity>normal</severity>
  <time>2016-12-11T11:13:44.654+02:00</time>
  <cluster href="/ovirt-engine/api/clusters/456" id="456"/>

```

```
<host href="/ovirt-engine/api/hosts/789" id="789"/>
  <user href="/ovirt-engine/api/users/987" id="987"/>
</event>
```

Note that the number of fields changes according to the information that resides on the event. For example, for storage domain related events you will get the storage domain reference, as well as the reference for the data center this storage domain resides in.

Table 6.167. Parameters summary

Name	Type	Direction	Summary
event	Event	Out	

6.57.2. remove DELETE

Removes an event from internal audit log.

An event can be removed by sending following request

```
DELETE /ovirt-engine/api/events/123
```

Table 6.168. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.58. EVENTS

A service to manage events in the system.

Table 6.169. Methods summary

Name	Summary
add	Adds an external event to the internal audit log.
list	Get list of events.
undelete	

6.58.1. add POST

Adds an external event to the internal audit log.

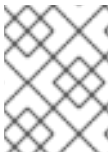
This is intended for integration with external systems that detect or produce events relevant for the administrator of the system. For example, an external monitoring tool may be able to detect that a file system is full inside the guest operating system of a virtual machine. This event can be added to the

internal audit log sending a request like this:

```
POST /ovirt-engine/api/events
<event>
  <description>File system /home is full</description>
  <severity>alert</severity>
  <origin>mymonitor</origin>
  <custom_id>1467879754</custom_id>
</event>
```

Events can also be linked to specific objects. For example, the above event could be linked to the specific virtual machine where it happened, using the `vm` link:

```
POST /ovirt-engine/api/events
<event>
  <description>File system /home is full</description>
  <severity>alert</severity>
  <origin>mymonitor</origin>
  <custom_id>1467879754</custom_id>
  <vm id="aae98225-5b73-490d-a252-899209af17e9"/>
</event>
```



NOTE

When using links, like the `vm` in the previous example, only the `id` attribute is accepted. The `name` attribute, if provided, is simply ignored.

Table 6.170. Parameters summary

Name	Type	Direction	Summary
event	Event	In/Out	

6.58.2. list GET

Get list of events.

```
GET /ovirt-engine/api/events
```

To the above request we get following response:

```
<events>
  <event href="/ovirt-engine/api/events/2" id="2">
    <description>User admin@internal-authz logged out.</description>
    <code>31</code>
    <correlation_id>1e892ea9</correlation_id>
    <custom_id>-1</custom_id>
    <flood_rate>30</flood_rate>
    <origin>oVirt</origin>
    <severity>normal</severity>
    <time>2016-09-14T12:14:34.541+02:00</time>
    <user href="/ovirt-engine/api/users/57d91d48-00da-0137-0138-
```

```

000000000244" id="57d91d48-00da-0137-0138-000000000244"/>
  </event>
  <event href="/ovirt-engine/api/events/1" id="1">
    <description>User admin logged in.</description>
    <code>30</code>
    <correlation_id>1fbd81f4</correlation_id>
    <custom_id>-1</custom_id>
    <flood_rate>30</flood_rate>
    <origin>oVirt</origin>
    <severity>normal</severity>
    <time>2016-09-14T11:54:35.229+02:00</time>
    <user href="/ovirt-engine/api/users/57d91d48-00da-0137-0138-
000000000244" id="57d91d48-00da-0137-0138-000000000244"/>
  </event>
</events>

```

The following events occur:

- id="1" - The API logs in the admin user account.
- id="2" - The API logs out of the admin user account.

Table 6.171. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
events	Event[]	Out	
from	Integer	In	Indicates the identifier of the the first event that should be returned.
max	Integer	In	Sets the maximum number of events to return.
search	String	In	The events service provides search queries similar to other resource services.

6.58.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.58.2.2. from

Indicates the identifier of the the first event that should be returned. The identifiers of events are strictly increasing, so when this parameter is used only the events with that identifiers equal or greater than the given value will be returned. For example, the following request will return only the events with identifiers greater or equal than **123**:

■

```
GET /ovirt-engine/api/events?from=123
```

This parameter is optional, and if not specified then the first event returned will be most recently generated.

6.58.2.3. max

Sets the maximum number of events to return. If not specified all the events are returned.

6.58.2.4. search

The events service provides search queries similar to other resource services.

We can search by providing specific severity.

```
GET /ovirt-engine/api/events?search=severity%3Dnormal
```

To the above request we get a list of events which severity is equal to **normal**:

```
<events>
  <event href="/ovirt-engine/api/events/2" id="2">
    <description>User admin@internal-authz logged out.</description>
    <code>31</code>
    <correlation_id>1fbd81f4</correlation_id>
    <custom_id>-1</custom_id>
    <flood_rate>30</flood_rate>
    <origin>oVirt</origin>
    <severity>normal</severity>
    <time>2016-09-14T11:54:35.229+02:00</time>
    <user href="/ovirt-engine/api/users/57d91d48-00da-0137-0138-000000000244" id="57d91d48-00da-0137-0138-000000000244"/>
  </event>
  <event href="/ovirt-engine/api/events/1" id="1">
    <description>Affinity Rules Enforcement Manager started.</description>
    <code>10780</code>
    <custom_id>-1</custom_id>
    <flood_rate>30</flood_rate>
    <origin>oVirt</origin>
    <severity>normal</severity>
    <time>2016-09-14T11:52:18.861+02:00</time>
  </event>
</events>
```

A virtualization environment generates a large amount of events after a period of time. However, the API only displays a default number of events for one search query. To display more than the default, the API separates results into pages with the page command in a search query. The following search query tells the API to paginate results using a page value in combination with the sortby clause:

```
sortby time asc page 1
```

Below example paginates event resources. The URL-encoded request is:

```
GET /ovirt-engine/api/events?search=sortby%20time%20asc%20page%201
```

Increase the page value to view the next page of results.

```
GET /ovirt-engine/api/events?search=sortBy%20time%20asc%20page%202
```

6.58.3. undelete POST

Table 6.172. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the un-delete should be performed asynchronously.

6.59. EXTERNALCOMPUTERESOURCE

Table 6.173. Methods summary

Name	Summary
get	

6.59.1. get GET

Table 6.174. Parameters summary

Name	Type	Direction	Summary
resource	ExternalComputeResource	Out	

6.60. EXTERNALCOMPUTERESOURCES

Table 6.175. Methods summary

Name	Summary
list	

6.60.1. list GET

Table 6.176. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of resources to return.

Name	Type	Direction	Summary
resources	ExternalComputeResource[]	Out	

6.60.1.1. max

Sets the maximum number of resources to return. If not specified all the resources are returned.

6.61. EXTERNALDISCOVEREDHOST

Table 6.177. Methods summary

Name	Summary
get	

6.61.1. get GET

Table 6.178. Parameters summary

Name	Type	Direction	Summary
host	ExternalDiscoveredHost	Out	

6.62. EXTERNALDISCOVEREDHOSTS

Table 6.179. Methods summary

Name	Summary
list	

6.62.1. list GET

Table 6.180. Parameters summary

Name	Type	Direction	Summary
hosts	ExternalDiscoveredHost[]	Out	
max	Integer	In	Sets the maximum number of hosts to return.

6.62.1.1. max

Sets the maximum number of hosts to return. If not specified all the hosts are returned.

6.63. EXTERNALHOST

Table 6.181. Methods summary

Name	Summary
get	

6.63.1. get GET

Table 6.182. Parameters summary

Name	Type	Direction	Summary
host	ExternalHost	Out	

6.64. EXTERNALHOSTGROUP

Table 6.183. Methods summary

Name	Summary
get	

6.64.1. get GET

Table 6.184. Parameters summary

Name	Type	Direction	Summary
group	ExternalHost Group	Out	

6.65. EXTERNALHOSTGROUPS

Table 6.185. Methods summary

Name	Summary
list	

6.65.1. list GET

Table 6.186. Parameters summary

Name	Type	Direction	Summary
groups	ExternalHost Group[]	Out	
max	Integer	In	Sets the maximum number of groups to return.

6.65.1.1. max

Sets the maximum number of groups to return. If not specified all the groups are returned.

6.66. EXTERNALHOSTPROVIDER

Table 6.187. Methods summary

Name	Summary
get	
importcertificates	
remove	
testconnectivity	
update	

6.66.1. get GET

Table 6.188. Parameters summary

Name	Type	Direction	Summary
provider	ExternalHost Provider	Out	

6.66.2. importcertificates POST

Table 6.189. Parameters summary

Name	Type	Direction	Summary
certificates	Certificate[]	In	

6.66.3. remove DELETE

Table 6.190. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.66.4. testconnectivity POST

Table 6.191. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the test should be performed asynchronously.

6.66.5. update PUT

Table 6.192. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
provider	ExternalHost Provider	In/Out	

6.67. EXTERNALHOSTPROVIDERS

Table 6.193. Methods summary

Name	Summary
add	
list	

6.67.1. add POST

Table 6.194. Parameters summary

Name	Type	Direction	Summary
------	------	-----------	---------

Name	Type	Direction	Summary
provider	ExternalHost Provider	In/Out	

6.67.2. list GET

Table 6.195. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of providers to return.
providers	ExternalHost Provider[]	Out	

6.67.2.1. max

Sets the maximum number of providers to return. If not specified all the providers are returned.

6.68. EXTERNALHOSTS

Table 6.196. Methods summary

Name	Summary
list	

6.68.1. list GET

Table 6.197. Parameters summary

Name	Type	Direction	Summary
hosts	ExternalHost []	Out	
max	Integer	In	Sets the maximum number of hosts to return.

6.68.1.1. max

Sets the maximum number of hosts to return. If not specified all the hosts are returned.

6.69. EXTERNALPROVIDER

Table 6.198. Methods summary

Name	Summary
<code>importcertificates</code>	
<code>testconnectivity</code>	

6.69.1. importcertificates POST

Table 6.199. Parameters summary

Name	Type	Direction	Summary
<code>certificates</code>	Certificate[]	In	

6.69.2. testconnectivity POST

Table 6.200. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the test should be performed asynchronously.

6.70. EXTERNALPROVIDERCERTIFICATE

Table 6.201. Methods summary

Name	Summary
<code>get</code>	

6.70.1. get GET

Table 6.202. Parameters summary

Name	Type	Direction	Summary
<code>certificate</code>	Certificate	Out	

6.71. EXTERNALPROVIDERCERTIFICATES

Table 6.203. Methods summary

Name	Summary
<code>list</code>	

6.71.1. list GET

Table 6.204. Parameters summary

Name	Type	Direction	Summary
<code>certificates</code>	Certificate[]	Out	
<code>max</code>	Integer	In	Sets the maximum number of certificates to return.

6.71.1.1. max

Sets the maximum number of certificates to return. If not specified all the certificates are returned.

6.72. EXTERNALVMIMPORTS

Provides capability to import external virtual machines.

Table 6.205. Methods summary

Name	Summary
<code>add</code>	This operation is used to import a virtual machine from external hypervisor, such as KVM, XEN or VMware.

6.72.1. add POST

This operation is used to import a virtual machine from external hypervisor, such as KVM, XEN or VMware.

For example import of a virtual machine from VMware can be facilitated using the following request:

```
POST /externalvmimports
```

With request body of type [ExternalVmImport](#), for example:

```
<external_vm_import>
  <vm>
    <name>my_vm</name>
  </vm>
  <cluster id="360014051136c20574f743bdbd28177fd" />
  <storage_domain id="8bb5ade5-e988-4000-8b93-dbf6717fe50" />
  <name>vm_name_as_is_in_vmware</name>
  <sparse>true</sparse>
```

```

<username>vmware_user</username>
<password>123456</password>
<provider>VMWARE</provider>
<url>vpx://vmware_user@vcenter-host/DataCenter/Cluster/esxi-host?
no_verify=1</url>
<drivers_iso id="virtio-win-1.6.7.iso" />
</external_vm_import>

```

Table 6.206. Parameters summary

Name	Type	Direction	Summary
import	ExternalVml mport	In/Out	

6.73. FENCEAGENT

Table 6.207. Methods summary

Name	Summary
get	
remove	
update	

6.73.1. get GET

Table 6.208. Parameters summary

Name	Type	Direction	Summary
agent	Agent	Out	

6.73.2. remove DELETE

Table 6.209. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.73.3. update PUT

Table 6.210. Parameters summary

Name	Type	Direction	Summary
agent	Agent	In/Out	
async	Boolean	In	Indicates if the update should be performed asynchronously.

6.74. FENCEAGENTS

Table 6.211. Methods summary

Name	Summary
add	
list	

6.74.1. add POST

Table 6.212. Parameters summary

Name	Type	Direction	Summary
agent	Agent	In/Out	

6.74.2. list GET

Table 6.213. Parameters summary

Name	Type	Direction	Summary
agents	Agent[]	Out	
max	Integer	In	Sets the maximum number of agents to return.

6.74.2.1. max

Sets the maximum number of agents to return. If not specified all the agents are returned.

6.75. FILE

Table 6.214. Methods summary

Name	Summary
get	

6.75.1. get GET

Table 6.215. Parameters summary

Name	Type	Direction	Summary
file	File	Out	

6.76. FILES

Provides a way for clients to list available files.

This services is specifically targeted to ISO storage domains, which contain ISO images and virtual floppy disks (VFDs) that an administrator uploads.

The addition of a CDROM device to a virtual machine requires an ISO image from the files of an ISO storage domain.

Table 6.216. Methods summary

Name	Summary
list	

6.76.1. list GET

Table 6.217. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
file	File[]	Out	
max	Integer	In	Sets the maximum number of files to return.
search	String	In	A query string used to restrict the returned files.

6.76.1.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.76.1.2. max

Sets the maximum number of files to return. If not specified all the files are returned.

6.77. FILTER

Table 6.218. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	

6.77.1. get GET

Table 6.219. Parameters summary

Name	Type	Direction	Summary
<code>filter</code>	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
<code>result</code>	Filter	Out	

6.77.2. remove DELETE

Table 6.220. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the remove should be performed asynchronously.

6.78. FILTERS

Table 6.221. Methods summary

Name	Summary
<code>add</code>	
<code>list</code>	

6.78.1. add POST

Table 6.222. Parameters summary

Name	Type	Direction	Summary
<code>filter</code>	Filter	In/Out	

6.78.2. list GET

Table 6.223. Parameters summary

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
filters	Filter[]	Out	
max	Integer	In	Sets the maximum number of filters to return.

6.78.2.1. max

Sets the maximum number of filters to return. If not specified all the filters are returned.

6.79. GLUSTERBRICK

This service manages a single gluster brick.

Table 6.224. Methods summary

Name	Summary
get	Get details of a brick.
remove	Removes a brick.
replace	Replaces this brick with a new one.

6.79.1. get GET

Get details of a brick.

Retrieves status details of brick from underlying gluster volume with header **All-Content** set to **true**. This is the equivalent of running **gluster volume status <volumename> <brickname> detail**.

For example, to get the details of brick **234** of gluster volume **123**, send a request like this:

```
GET /ovirt-engine/api/clusters/567/glustervolumes/123/glusterbricks/234
```

Which will return a response body like this:

```
<brick id="234">
  <name>host1:/rhgs/data/brick1</name>
  <brick_dir>/rhgs/data/brick1</brick_dir>
  <server_id>111</server_id>
  <status>up</status>
```



```

<device>/dev/mapper/RHGS_vg1-lv_vmaddldisks</device>
<fs_name>xfss</fs_name>
<gluster_clients>
  <gluster_client>
    <bytes_read>2818417648</bytes_read>
    <bytes_written>1384694844</bytes_written>
    <client_port>1011</client_port>
    <host_name>client2</host_name>
  </gluster_client>
</gluster_clients>
<memory_pools>
  <memory_pool>
    <name>data-server:fd_t</name>
    <alloc_count>1626348</alloc_count>
    <cold_count>1020</cold_count>
    <hot_count>4</hot_count>
    <max_alloc>23</max_alloc>
    <max_stdalloc>0</max_stdalloc>
    <padded_size>140</padded_size>
    <pool_misses>0</pool_misses>
  </memory_pool>
</memory_pools>

<mnt_options>rw,seclabel,noatime,nodiratime,attr2,inode64,sunit=512,swidth
=2048,noquota</mnt_options>
  <pid>25589</pid>
  <port>49155</port>
</brick>

```

Table 6.225. Parameters summary

Name	Type	Direction	Summary
brick	GlusterBrick	Out	

6.79.2. remove DELETE

Removes a brick.

Removes a brick from the underlying gluster volume and deletes entries from database. This can be used only when removing a single brick without data migration. To remove multiple bricks and with data migration, use [migrate](#) instead.

For example, to delete brick **234** from gluster volume **123**, send a request like this:

```
DELETE /ovirt-engine/api/clusters/567/glustervolumes/123/glusterbricks/234
```

Table 6.226. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.79.3. replace POST

Replaces this brick with a new one.



IMPORTANT

This operation has been deprecated since version 3.5 of the engine and will be removed in the future. Use [add brick\(s\)](#) and [migrate brick\(s\)](#) instead.

Table 6.227. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the replacement should be performed asynchronously.
force	Boolean	In	

6.80. GLUSTERBRICKS

This service manages the gluster bricks in a gluster volume

Table 6.228. Methods summary

Name	Summary
activate	Activate the bricks post data migration of remove brick operation.
add	Adds a list of bricks to gluster volume.
list	Lists the bricks of a gluster volume.
migrate	Start migration of data prior to removing bricks.
remove	Removes bricks from gluster volume.
stopmigrate	Stops migration of data from bricks for a remove brick operation.

6.80.1. activate POST

Activate the bricks post data migration of remove brick operation.

Used to activate brick(s) once the data migration from bricks is complete but user no longer wishes to remove bricks. The bricks that were previously marked for removal will now be used as normal bricks.

For example, to retain the bricks that on glustervolume **123** from which data was migrated, send a request like this:

```
POST /ovirt-
engine/api/clusters/567/glustervolumes/123/glusterbricks/activate
```

With a request body like this:

```
<action>
  <bricks>
    <brick>
      <name>host1:/rhgs/brick1</name>
    </brick>
  </bricks>
</action>
```

Table 6.229. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the activation should be performed asynchronously.
bricks	GlusterBrick[]	In	The list of bricks that need to be re-activated.

6.80.2. add POST

Adds a list of bricks to gluster volume.

Used to expand a gluster volume by adding bricks. For replicated volume types, the parameter **replica_count** needs to be passed. In case the replica count is being increased, then the number of bricks needs to be equivalent to the number of replica sets.

For example, to add bricks to gluster volume **123**, send a request like this:

```
POST /ovirt-engine/api/clusters/567/glustervolumes/123/glusterbricks
```

With a request body like this:

```
<bricks>
  <brick>
    <server_id>111</server_id>
    <brick_dir>/export/data/brick3</brick_dir>
  </brick>
</bricks>
```

Table 6.230. Parameters summary

Name	Type	Direction	Summary
bricks	GlusterBrick[]	In/Out	The list of bricks to be added to the volume
replica_count	Integer	In	Replica count of volume post add operation.

Name	Type	Direction	Summary
stripe_count	Integer	In	Stripe count of volume post add operation.

6.80.3. list GET

Lists the bricks of a gluster volume.

For example, to list bricks of gluster volume **123**, send a request like this:

```
GET /ovirt-engine/api/clusters/567/glustervolumes/123/glusterbricks
```

Provides an output as below:

```
<bricks>
  <brick id="234">
    <name>host1:/rhgs/data/brick1</name>
    <brick_dir>/rhgs/data/brick1</brick_dir>
    <server_id>111</server_id>
    <status>up</status>
  </brick>
  <brick id="233">
    <name>host2:/rhgs/data/brick1</name>
    <brick_dir>/rhgs/data/brick1</brick_dir>
    <server_id>222</server_id>
    <status>up</status>
  </brick>
</bricks>
```

Table 6.231. Parameters summary

Name	Type	Direction	Summary
bricks	GlusterBrick[]	Out	
max	Integer	In	Sets the maximum number of bricks to return.

6.80.3.1. max

Sets the maximum number of bricks to return. If not specified all the bricks are returned.

6.80.4. migrate POST

Start migration of data prior to removing bricks.

Removing bricks is a two-step process, where the data on bricks to be removed, is first migrated to remaining bricks. Once migration is completed the removal of bricks is confirmed via the API [remove](#). If at any point, the action needs to be cancelled [stopmigrate](#) has to be called.

For instance, to delete a brick from a gluster volume with id **123**, send a request:

```
POST /ovirt-engine/api/clusters/567/glustervolumes/123/glusterbricks/migrate
```

With a request body like this:

```
<action>
  <bricks>
    <brick>
      <name>host1:/rhgs/brick1</name>
    </brick>
  </bricks>
</action>
```

The migration process can be tracked from the job id returned from the API using [job](#) and steps in job using [step](#)

Table 6.232. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the migration should be performed asynchronously.
bricks	GlusterBrick[]	In	List of bricks for which data migration needs to be started.

6.80.5. remove DELETE

Removes bricks from gluster volume.

The recommended way to remove bricks without data loss is to first migrate the data using [stopmigrate](#) and then removing them. If migrate was not called on bricks prior to remove, the bricks are removed without data migration which may lead to data loss.

For example, to delete the bricks from gluster volume **123**, send a request like this:

```
DELETE /ovirt-engine/api/clusters/567/glustervolumes/123/glusterbricks
```

With a request body like this:

```
<bricks>
  <brick>
    <name>host:brick_directory</name>
  </brick>
</bricks>
```

Table 6.233. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.
bricks	GlusterBrick[]	In	The list of bricks to be removed
replica_count	Integer	In	Replica count of volume post add operation.

6.80.6. stopmigrate POST

Stops migration of data from bricks for a remove brick operation.

To cancel data migration that was started as part of the 2-step remove brick process in case the user wishes to continue using the bricks. The bricks that were marked for removal will function as normal bricks post this operation.

For example, to stop migration of data from the bricks of gluster volume **123**, send a request like this:

```
POST /ovirt-engine/api/clusters/567/glustervolumes/123/glusterbricks/stopmigrate
```

With a request body like this:

```
<bricks>
  <brick>
    <name>host:brick_directory</name>
  </brick>
</bricks>
```

Table 6.234. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
bricks	GlusterBrick[]	In	List of bricks for which data migration needs to be stopped.

6.80.6.1. bricks

List of bricks for which data migration needs to be stopped. This list should match the arguments passed to [migrate](#).

6.81. GLUSTERHOOK

Table 6.235. Methods summary

Name	Summary
disable	Resolves status conflict of hook among servers in cluster by disabling Gluster hook in all servers of the cluster.
enable	Resolves status conflict of hook among servers in cluster by disabling Gluster hook in all servers of the cluster.
get	
remove	Removes the this Gluster hook from all servers in cluster and deletes it from the database.
resolve	Resolves missing hook conflict depending on the resolution type.

6.81.1. disable POST

Resolves status conflict of hook among servers in cluster by disabling Gluster hook in all servers of the cluster. This updates the hook status to **DISABLED** in database.

Table 6.236. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.

6.81.2. enable POST

Resolves status conflict of hook among servers in cluster by disabling Gluster hook in all servers of the cluster. This updates the hook status to **DISABLED** in database.

Table 6.237. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.

6.81.3. get GET

Table 6.238. Parameters summary

Name	Type	Direction	Summary
hook	GlusterHook	Out	

6.81.4. remove DELETE

Removes the this Gluster hook from all servers in cluster and deletes it from the database.

Table 6.239. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.81.5. resolve POST

Resolves missing hook conflict depending on the resolution type.

For **ADD** resolves by copying hook stored in engine database to all servers where the hook is missing. The engine maintains a list of all servers where hook is missing.

For **COPY** resolves conflict in hook content by copying hook stored in engine database to all servers where the hook is missing. The engine maintains a list of all servers where the content is conflicting. If a host id is passed as parameter, the hook content from the server is used as the master to copy to other servers in cluster.

Table 6.240. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
host	Host	In	
resolution_type	String	In	

6.82. GLUSTERHOOKS

Table 6.241. Methods summary

Name	Summary
list	

6.82.1. list GET

Table 6.242. Parameters summary

Name	Type	Direction	Summary
hooks	GlusterHook[]	Out	
max	Integer	In	Sets the maximum number of hooks to return.

6.82.1.1. max

Sets the maximum number of hooks to return. If not specified all the hooks are returned.

6.83. GLUSTERVOLUME

This service manages a single gluster volume.

Table 6.243. Methods summary

Name	Summary
get	Get the gluster volume details.
getprofilestatistics	Get gluster volume profile statistics.
rebalance	Rebalance the gluster volume.
remove	Removes the gluster volume.
resetalloptions	Resets all the options set in the gluster volume.
resetoption	Resets a particular option in the gluster volume.
setoption	Sets a particular option in the gluster volume.
start	Starts the gluster volume.
startprofile	Start profiling the gluster volume.
stop	Stops the gluster volume.
stopprofile	Stop profiling the gluster volume.
stoprebalance	Stop rebalancing the gluster volume.

6.83.1. get GET

Get the gluster volume details.

For example, to get details of a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
GET /ovirt-engine/api/clusters/456/glustervolumes/123
```

This GET request will return the following output:

```
<gluster_volume id="123">
  <name>data</name>
  <link href="/ovirt-engine/api/clusters/456/glustervolumes/123/glusterbricks"
rel="glusterbricks"/>
  <disperse_count>0</disperse_count>
  <options>
    <option>
      <name>storage.owner-gid</name>
      <value>36</value>
    </option>
    <option>
      <name>performance.io-cache</name>
      <value>off</value>
    </option>
    <option>
      <name>cluster.data-self-heal-algorithm</name>
      <value>full</value>
    </option>
  </options>
  <redundancy_count>0</redundancy_count>
  <replica_count>3</replica_count>
  <status>up</status>
  <stripe_count>0</stripe_count>
  <transport_types>
    <transport_type>tcp</transport_type>
  </transport_types>
  <volume_type>replicate</volume_type>
</gluster_volume>
```

Table 6.244. Parameters summary

Name	Type	Direction	Summary
volume	GlusterVolume	Out	Representation of the gluster volume.

6.83.2. getprofilestatistics POST

Get gluster volume profile statistics.

For example, to get profile statistics for a gluster volume with identifier **123** in cluster **456**, send a request like this:

■

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/getprofilestatistics
```

Table 6.245. Parameters summary

Name	Type	Direction	Summary
details	GlusterVolumeProfileDetails	Out	Gluster volume profiling information returned from the action.

6.83.3. rebalance POST

Rebalance the gluster volume.

Rebalancing a gluster volume helps to distribute the data evenly across all the bricks. After expanding or shrinking a gluster volume (without migrating data), we need to rebalance the data among the bricks. In a non-replicated volume, all bricks should be online to perform the rebalance operation. In a replicated volume, at least one of the bricks in the replica should be online.

For example, to rebalance a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/rebalance
```

Table 6.246. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the rebalance should be performed asynchronously.
fix_layout	Boolean	In	If set to true, rebalance will only fix the layout so that new data added to the volume is distributed across all the hosts.
force	Boolean	In	Indicates if the rebalance should be force started.

6.83.3.1. fix_layout

If set to true, rebalance will only fix the layout so that new data added to the volume is distributed across all the hosts. But it will not migrate/rebalance the existing data. Default is **false**.

6.83.3.2. force

Indicates if the rebalance should be force started. The rebalance command can be executed with the force option even when the older clients are connected to the cluster. However, this could lead to a data loss situation. Default is **false**.

6.83.4. remove DELETE

Removes the gluster volume.

For example, to remove a volume with identifier **123** in cluster **456**, send a request like this:

```
DELETE /ovirt-engine/api/clusters/456/glustervolumes/123
```

Table 6.247. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.83.5. resetalloptions POST

Resets all the options set in the gluster volume.

For example, to reset all options in a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/resetalloptions
```

Table 6.248. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the reset should be performed asynchronously.

6.83.6. resetoption POST

Resets a particular option in the gluster volume.

For example, to reset a particular option **option1** in a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/resetoption
```

With the following request body:

```
<action>
  <option name="option1"/>
</action>
```

Table 6.249. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the reset should be performed asynchronously.

Name	Type	Direction	Summary
force	Boolean	In	
option	Option	In	Option to reset.

6.83.7. setoption POST

Sets a particular option in the gluster volume.

For example, to set **option1** with value **value1** in a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/setoption
```

With the following request body:

```
<action>
  <option name="option1" value="value1"/>
</action>
```

Table 6.250. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
option	Option	In	Option to set.

6.83.8. start POST

Starts the gluster volume.

A Gluster Volume should be started to read/write data. For example, to start a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/start
```

Table 6.251. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
force	Boolean	In	Indicates if the volume should be force started.

6.83.8.1. force

Indicates if the volume should be force started. If a gluster volume is started already but few/all bricks are down then force start can be used to bring all the bricks up. Default is **false**.

6.83.9. startprofile POST

Start profiling the gluster volume.

For example, to start profiling a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/startprofile
```

Table 6.252. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.

6.83.10. stop POST

Stops the gluster volume.

Stopping a volume will make its data inaccessible.

For example, to stop a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/stop
```

Table 6.253. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
force	Boolean	In	

6.83.11. stopprofile POST

Stop profiling the gluster volume.

For example, to stop profiling a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/stopprofile
```

Table 6.254. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.

6.83.12. stoprebalance POST

Stop rebalancing the gluster volume.

For example, to stop rebalancing a gluster volume with identifier **123** in cluster **456**, send a request like this:

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/stoprebalance
```

Table 6.255. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.

6.84. GLUSTERVOLUMES

This service manages a collection of gluster volumes available in a cluster.

Table 6.256. Methods summary

Name	Summary
add	Creates a new gluster volume.
list	Lists all gluster volumes in the cluster.

6.84.1. add POST

Creates a new gluster volume.

The volume is created based on properties of the **volume** parameter. The properties **name**, **volume_type** and **bricks** are required.

For example, to add a volume with name **myvolume** to the cluster **123**, send the following request:

```
POST /ovirt-engine/api/clusters/123/glustervolumes
```

With the following request body:

```
<gluster_volume>
  <name>myvolume</name>
  <volume_type>replicate</volume_type>
```

```

<replica_count>3</replica_count>
<bricks>
  <brick>
    <server_id>server1</server_id>
    <brick_dir>/exp1</brick_dir>
  </brick>
  <brick>
    <server_id>server2</server_id>
    <brick_dir>/exp1</brick_dir>
  </brick>
  <brick>
    <server_id>server3</server_id>
    <brick_dir>/exp1</brick_dir>
  </brick>
</bricks>
</gluster_volume>

```

Table 6.257. Parameters summary

Name	Type	Direction	Summary
volume	GlusterVolume	In/Out	The gluster volume definition from which to create the volume is passed as input and the newly created volume is returned.

6.84.2. list GET

Lists all gluster volumes in the cluster.

For example, to list all Gluster Volumes in cluster **456**, send a request like this:

```
GET /ovirt-engine/api/clusters/456/glustervolumes
```

Table 6.258. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
max	Integer	In	Sets the maximum number of volumes to return.
search	String	In	A query string used to restrict the returned volumes.
volumes	GlusterVolume[]	Out	

6.84.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.84.2.2. max

Sets the maximum number of volumes to return. If not specified all the volumes are returned.

6.85. GROUP

Table 6.259. Methods summary

Name	Summary
get	
remove	

6.85.1. get GET

Table 6.260. Parameters summary

Name	Type	Direction	Summary
get	Group	Out	

6.85.2. remove DELETE

Table 6.261. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.86. GROUPS

Table 6.262. Methods summary

Name	Summary
add	Add group from a directory service.
list	

6.86.1. add POST

Add group from a directory service. Please note that domain name is name of the authorization provider.

For example, to add the **Developers** group from the **internal-authz** authorization provider send a request like this:

```
POST /ovirt-engine/api/groups
```

With a request body like this:

```
<group>
  <name>Developers</name>
  <domain>
    <name>internal-authz</name>
  </domain>
</group>
```

Table 6.263. Parameters summary

Name	Type	Direction	Summary
group	Group	In/Out	

6.86.2. list GET

Table 6.264. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
groups	Group[]	Out	
max	Integer	In	Sets the maximum number of groups to return.
search	String	In	A query string used to restrict the returned groups.

6.86.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.86.2.2. max

Sets the maximum number of groups to return. If not specified all the groups are returned.

6.87. HOST

A service to manage a host.

Table 6.265. Methods summary

Name	Summary
activate	Activate the host for use, such as running virtual machines.
approve	Approve a pre-installed Hypervisor host for usage in the virtualization environment.
commitnetconfig	Marks the network configuration as good and persists it inside the host.
deactivate	Deactivate the host to perform maintenance tasks.
enrollcertificate	Enroll certificate of the host.
fence	Controls host's power management device.
forceselectspm	Manually set a host as the storage pool manager (SPM).
get	Get the host details.
install	Install VDSM and related software on the host.
iscsidiscover	Discover iSCSI targets on the host, using the initiator details.
iscsilogin	Login to iSCSI targets on the host, using the target details.
refresh	Refresh the host devices and capabilities.
remove	Remove the host from the system.
setupnetworks	This method is used to change the configuration of the network interfaces of a host.
unregisteredstoragedomainsdiscover	
update	Update the host properties.
upgrade	Upgrade VDSM and selected software on the host.
upgradecheck	Check if there are upgrades available for the host.

6.87.1. activate POST

Activate the host for use, such as running virtual machines.

Table 6.266. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the activation should be performed asynchronously.

6.87.2. approve POST

Approve a pre-installed Hypervisor host for usage in the virtualization environment.

This action also accepts an optional cluster element to define the target cluster for this host.

Table 6.267. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the approval should be performed asynchronously.
cluster	Cluster	In	

6.87.3. commitnetconfig POST

Marks the network configuration as good and persists it inside the host.

An API user commits the network configuration to persist a host network interface attachment or detachment, or persist the creation and deletion of a bonded interface.



IMPORTANT

Networking configuration is only committed after the engine has established that host connectivity is not lost as a result of the configuration changes. If host connectivity is lost, the host requires a reboot and automatically reverts to the previous networking configuration.

For example, to commit the network configuration of host with id **123** send a request like this:

```
POST /ovirt-engine/api/hosts/123/commitnetconfig
```

With a request body like this:

```
<action/>
```

Table 6.268. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.

6.87.4. deactivate POST

Deactivate the host to perform maintenance tasks.

Table 6.269. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the deactivation should be performed asynchronously.
reason	String	In	
stop_gluster_service	Boolean	In	Indicates if the gluster service should be stopped as part of deactivating the host.

6.87.4.1. stop_gluster_service

Indicates if the gluster service should be stopped as part of deactivating the host. It can be used while performing maintenance operations on the gluster host. Default value for this variable is **false**.

6.87.5. enrollcertificate POST

Enroll certificate of the host. Useful in case you get a warning that it is about to, or already expired.

Table 6.270. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the enrollment should be performed asynchronously.

6.87.6. fence POST

Controls host's power management device.

For example, let's assume you want to start the host. This can be done via:

```
#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."
```

```

curl \
  --verbose \
  --cacert /etc/pki/ovirt-engine/ca.pem \
  --user "${user}:${password}" \
  --request POST \
  --header "Version: 4" \
  --header "Content-Type: application/xml" \
  --header "Accept: application/xml" \
  --data '
<action>
  <fence_type>start</fence_type>
</action>
' \
"${url}/hosts/123/fence"

```

Table 6.271. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the fencing should be performed asynchronously.
fence_type	String	In	
power_management	PowerManagement	Out	

6.87.7. forceselectspm POST

Manually set a host as the storage pool manager (SPM).

```
POST /ovirt-engine/api/hosts/123/forceselectspm
```

With a request body like this:

```
<action/>
```

Table 6.272. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.

6.87.8. get GET

Get the host details.

Table 6.273. Parameters summary

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
host	Host	Out	

6.87.9. install POST

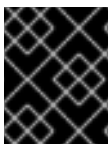
Install VDSM and related software on the host. The host type defines additional parameters for the action.

Example of installing a host, using **curl** and JSON, plain:

```
curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--request PUT \
--header "Content-Type: application/json" \
--header "Accept: application/json" \
--header "Version: 4" \
--user "admin@internal:..." \
--data '
{
  "root_password": "myrootpassword"
}
' \
"https://engine.example.com/ovirt-engine/api/hosts/123"
```

Example of installing a host, using **curl** and JSON, with hosted engine components:

```
curl \
curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--request PUT \
--header "Content-Type: application/json" \
--header "Accept: application/json" \
--header "Version: 4" \
--user "admin@internal:..." \
--data '
{
  "root_password": "myrootpassword"
}
' \
"https://engine.example.com/ovirt-engine/api/hosts/123?
deploy_hosted_engine=true"
```



IMPORTANT

Since version 4.1.2 of the engine when a host is reinstalled we override the host firewall definitions by default.

Table 6.274. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the installation should be performed asynchronously.
deploy_hosted_engine	Boolean	In	When set to true it means this host should deploy also hosted engine components.
host	Host	In	This override_iptables property is used to indicate if the firewall configuration should be replaced by the default one.
image	String	In	When installing an oVirt node a image ISO file is needed.
root_password	String	In	The password of of the root user, used to connect to the host via SSH.
ssh	Ssh	In	The SSH details used to connect to the host.
undeploy_hosted_engine	Boolean	In	When set to true it means this host should un-deploy hosted engine components and this host will not function as part of the High Availability cluster.

6.87.9.1. deploy_hosted_engine

When set to **true** it means this host should deploy also hosted engine components. Missing value is treated as **true** i.e deploy. Omitting this parameter means **false** and will perform no operation in hosted engine area.

6.87.9.2. undeploy_hosted_engine

When set to **true** it means this host should un-deploy hosted engine components and this host will not function as part of the High Availability cluster. Missing value is treated as **true** i.e un-deploy Omitting this parameter means **false** and will perform no operation in hosted engine area.

6.87.10. iscsidiscover POST

Discover iSCSI targets on the host, using the initiator details.

Table 6.275. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the discovery should be performed asynchronously.

Name	Type	Direction	Summary
iscsi	iscsiDetails	In	The target iSCSI device.
iscsi_targets	String[]	Out	The iSCSI targets.

6.87.11. iscsilogin POST

Login to iSCSI targets on the host, using the target details.

Table 6.276. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the login should be performed asynchronously.
iscsi	iscsiDetails	In	The target iSCSI device.

6.87.12. refresh POST

Refresh the host devices and capabilities.

Table 6.277. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the refresh should be performed asynchronously.

6.87.13. remove DELETE

Remove the host from the system.

```
#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."

curl \
  --verbose \
  --cacert /etc/pki/ovirt-engine/ca.pem \
  --user "${user}:${password}" \
  --request DELETE \
  --header "Version: 4" \
  "${url}/hosts/1ff7a191-2f3b-4eff-812b-9f91a30c3acc"
```

Table 6.278. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.87.14. setupnetworks POST

This method is used to change the configuration of the network interfaces of a host.

For example, lets assume that you have a host with three network interfaces **eth0**, **eth1** and **eth2** and that you want to configure a new bond using **eth0** and **eth1**, and put a VLAN on top of it. Using a simple shell script and the **curl** command line HTTP client that can be done as follows:

```
#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."

curl \
  --verbose \
  --cacert /etc/pki/ovirt-engine/ca.pem \
  --user "${user}:${password}" \
  --request POST \
  --header "Version: 4" \
  --header "Content-Type: application/xml" \
  --header "Accept: application/xml" \
  --data '
<action>
  <modified_bonds>
    <host_nic>
      <name>bond0</name>
      <bonding>
        <options>
          <option>
            <name>mode</name>
            <value>4</value>
          </option>
          <option>
            <name>miimon</name>
            <value>100</value>
          </option>
        </options>
        <slaves>
          <host_nic>
            <name>eth1</name>
          </host_nic>
          <host_nic>
            <name>eth2</name>
          </host_nic>
        </slaves>
      </bonding>
    </host_nic>
  </modified_bonds>
```

```

<modified_network_attachments>
  <network_attachment>
    <network>
      <name>myvlan</name>
    </network>
    <host_nic>
      <name>bond0</name>
    </host_nic>
    <ip_address_assignments>
      <assignment_method>static</assignment_method>
      <ip_address_assignment>
        <ip>
          <address>192.168.122.10</address>
          <netmask>255.255.255.0</netmask>
        </ip>
      </ip_address_assignment>
    </ip_address_assignments>
    <dns_resolver_configuration>
      <name_servers>
        <name_server>1.1.1.1</name_server>
        <name_server>2.2.2.2</name_server>
      </name_servers>
    </dns_resolver_configuration>
  </network_attachment>
</modified_network_attachments>
</action>
' \
"${url}/hosts/1ff7a191-2f3b-4eff-812b-9f91a30c3acc/setupnetworks"

```

Note that this is valid for version 4 of the API. In previous versions some elements were represented as XML attributes instead of XML elements. In particular the **options** and **ip** elements were represented as follows:

```

<options name="mode" value="4"/>
<options name="miimon" value="100"/>
<ip address="192.168.122.10" netmask="255.255.255.0"/>

```

Using the Python SDK the same can be done with the following code:

```

# Find the service that manages the collection of hosts:
hosts_service = connection.system_service().hosts_service()

# Find the host:
host = hosts_service.list(search='name=myhost')[0]

# Find the service that manages the host:
host_service = hosts_service.host_service(host.id)

# Configure the network adding a bond with two slaves and attaching it to
a
# network with an static IP address:
host_service.setup_networks(
    modified_bonds=[
        types.HostNic(
            name='bond0',

```

```

        bonding=types.Bonding(
            options=[
                types.Option(
                    name='mode',
                    value='4',
                ),
                types.Option(
                    name='miimon',
                    value='100',
                ),
            ],
            slaves=[
                types.HostNic(
                    name='eth1',
                ),
                types.HostNic(
                    name='eth2',
                ),
            ],
        ),
    ],
),
],
modified_network_attachments=[
    types.NetworkAttachment(
        network=types.Network(
            name='myvlan',
        ),
        host_nic=types.HostNic(
            name='bond0',
        ),
        ip_address_assignments=[
            types.IpAddressAssignment(
                assignment_method=types.BootProtocol.STATIC,
                ip=types.Ip(
                    address='192.168.122.10',
                    netmask='255.255.255.0',
                ),
            ),
        ],
        dns_resolver_configuration=types.DnsResolverConfiguration(
            name_servers=[
                '1.1.1.1',
                '2.2.2.2',
            ],
        ),
    ),
],
)

```

After modifying the network configuration it is very important to make it

persistent:

```
host_service.commit_net_config()
```

**IMPORTANT**

To make sure that the network configuration has been saved in the host, and that it will be applied when the host is rebooted, remember to call [commitnetconfig](#).

Table 6.279. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
check_connectivity	Boolean	In	
connectivity_timeout	Integer	In	
modified_bonds	HostNic[]	In	
modified_labels	NetworkLabel[]	In	
modified_network_attachments	NetworkAttachment[]	In	
removed_bonds	HostNic[]	In	
removed_labels	NetworkLabel[]	In	
removed_network_attachments	NetworkAttachment[]	In	
synchronized_network_attachments	NetworkAttachment[]	In	

6.87.15. unregisteredstoragedomainsdiscover POST**Table 6.280. Parameters summary**

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the discovery should be performed asynchronously.
iscsi	IscsiDetails	In	
storage_domains	StorageDomain[]	Out	

6.87.16. update PUT

Update the host properties.

For example, to update a the kernel command line of a host send a request like this:

```
PUT /ovirt-engine/api/hosts/123
```

With request body like this:

```
<host>
  <os>

  <custom_kernel_cmdline>vfio_iommu_type1.allow_unsafe_interrupts=1</custom_
kernel_cmdline>
  </os>
</host>
```

Table 6.281. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
host	Host	In/Out	

6.87.17. upgrade POST

Upgrade VDSM and selected software on the host.

Table 6.282. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the upgrade should be performed asynchronously.

6.87.18. upgradecheck POST

Check if there are upgrades available for the host. If there are upgrades available an icon will be displayed next to host status icon in the webadmin. Audit log messages are also added to indicate the availability of upgrades. The upgrade can be started from the webadmin or by using the [upgrade](#) host action.

6.88. HOSTDEVICE

A service to access a particular device of a host.

Table 6.283. Methods summary

Name	Summary
<code>get</code>	Retrieve information about a particular host's device.

6.88.1. get GET

Retrieve information about a particular host's device.

An example of getting a host device:

```
GET /ovirt-engine/api/hosts/123/devices/456
```

```
<host_device href="/ovirt-engine/api/hosts/123/devices/456" id="456">
  <name>usb_1_9_1_1_0</name>
  <capability>usb</capability>
  <host href="/ovirt-engine/api/hosts/123" id="123"/>
  <parent_device href="/ovirt-engine/api/hosts/123/devices/789" id="789">
    <name>usb_1_9_1</name>
  </parent_device>
</host_device>
```

Table 6.284. Parameters summary

Name	Type	Direction	Summary
<code>device</code>	HostDevice	Out	

6.89. HOSTDEVICES

A service to access host devices.

Table 6.285. Methods summary

Name	Summary
<code>list</code>	List the devices of a host.

6.89.1. list GET

List the devices of a host.

Table 6.286. Parameters summary

Name	Type	Direction	Summary
devices	HostDevice[]	Out	
max	Integer	In	Sets the maximum number of devices to return.

6.89.1.1. max

Sets the maximum number of devices to return. If not specified all the devices are returned.

6.90. HOSTHOOK

Table 6.287. Methods summary

Name	Summary
get	

6.90.1. get GET

Table 6.288. Parameters summary

Name	Type	Direction	Summary
hook	Hook	Out	

6.91. HOSTHOOKS

Table 6.289. Methods summary

Name	Summary
list	

6.91.1. list GET

Table 6.290. Parameters summary

Name	Type	Direction	Summary
hooks	Hook[]	Out	
max	Integer	In	Sets the maximum number of hooks to return.

6.91.1.1. max

Sets the maximum number of hooks to return. If not specified all the hooks are returned.

6.92. HOSTNIC

A service to manage a network interface of a host.

Table 6.291. Methods summary

Name	Summary
<code>get</code>	
<code>updatevirtualfunctionsconfiguration</code>	The action updates virtual function configuration in case the current resource represents an SR-IOV enabled NIC.

6.92.1. get GET

Table 6.292. Parameters summary

Name	Type	Direction	Summary
<code>nic</code>	HostNic	Out	

6.92.2. updatevirtualfunctionsconfiguration POST

The action updates virtual function configuration in case the current resource represents an SR-IOV enabled NIC. The input should be consisted of at least one of the following properties:

- `allNetworksAllowed`
- `numberOfVirtualFunctions`

Please see the `HostNicVirtualFunctionsConfiguration` type for the meaning of the properties.

Table 6.293. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the update should be performed asynchronously.
<code>virtual_functions_configuration</code>	HostNicVirtualFunctionsConfiguration	In	

6.93. HOSTNICS

A service to manage the network interfaces of a host.

Table 6.294. Methods summary

Name	Summary
<code>list</code>	

6.93.1. list GET

Table 6.295. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of NICs to return.
<code>nics</code>	HostNic[]	Out	

6.93.1.1. max

Sets the maximum number of NICs to return. If not specified all the NICs are returned.

6.94. HOSTNUMANODE

Table 6.296. Methods summary

Name	Summary
<code>get</code>	

6.94.1. get GET

Table 6.297. Parameters summary

Name	Type	Direction	Summary
<code>node</code>	NumaNode	Out	

6.95. HOSTNUMANODES

Table 6.298. Methods summary

Name	Summary
<code>list</code>	

6.95.1. list GET

Table 6.299. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of nodes to return.
nodes	NumaNode[]	Out	

6.95.1.1. max

Sets the maximum number of nodes to return. If not specified all the nodes are returned.

6.96. HOSTSTORAGE

A service to manage host storages.

Table 6.300. Methods summary

Name	Summary
list	Get list of storages.

6.96.1. list GET

Get list of storages.

```
GET /ovirt-engine/api/hosts/123/storage
```

The XML response you get will be like this one:

```
<host_storages>
  <host_storage id="123">
    ...
  </host_storage>
  ...
</host_storages>
```

Table 6.301. Parameters summary

Name	Type	Direction	Summary
report_status	Boolean	In	Indicates if the status of the LUNs in the storage should be checked.
storages	HostStorage[]	Out	Retrieved list of storages.

6.96.1.1. report_status

Indicates if the status of the LUNs in the storage should be checked. Checking the status of the LUN is an heavy weight operation and this data is not always needed by the user. This parameter will give the option to not perform the status check of the LUNs.

The default is **true** for backward compatibility.

Here an example with the LUN status :

```
<host_storage id="123">
  <logical_units>
    <logical_unit id="123">
      <lun_mapping>0</lun_mapping>
      <paths>1</paths>
      <product_id>lun0</product_id>
      <serial>123</serial>
      <size>10737418240</size>
      <status>used</status>
      <vendor_id>LIO-ORG</vendor_id>
      <volume_group_id>123</volume_group_id>
    </logical_unit>
  </logical_units>
  <type>iscsi</type>
  <host id="123"/>
</host_storage>
```

Here an example without the LUN status :

```
<host_storage id="123">
  <logical_units>
    <logical_unit id="123">
      <lun_mapping>0</lun_mapping>
      <paths>1</paths>
      <product_id>lun0</product_id>
      <serial>123</serial>
      <size>10737418240</size>
      <vendor_id>LIO-ORG</vendor_id>
      <volume_group_id>123</volume_group_id>
    </logical_unit>
  </logical_units>
  <type>iscsi</type>
  <host id="123"/>
</host_storage>
```

6.97. HOSTS

A service that manages hosts.

Table 6.302. Methods summary

Name	Summary
add	Creates a new host.

Name	Summary
list	Get a list of all available hosts.

6.97.1. add POST

Creates a new host.

The host is created based on the attributes of the **host** parameter. The **name**, **address** and **root_password** properties are required.

For example, to add a host send the following request:

```
POST /ovirt-engine/api/hosts
```

With the following request body:

```
<host>
  <name>myhost</name>
  <address>myhost.example.com</address>
  <root_password>myrootpassword</root_password>
</host>
```



NOTE

The **root_password** element is only included in the client-provided initial representation and is not exposed in the representations returned from subsequent requests.



IMPORTANT

Since version 4.1.2 of the engine when a host is newly added we override the host firewall definitions by default.

To add a hosted engine host, use the optional **deploy_hosted_engine** parameter:

```
POST /ovirt-engine/api/hosts?deploy_hosted_engine=true
```

Table 6.303. Parameters summary

Name	Type	Direction	Summary
deploy_hosted_engine	Boolean	In	When set to true it means this host should deploy also hosted engine components.
host	Host	In/Out	The host definition from which to create the new host is passed as parameter, and the newly created host is returned.

Name	Type	Direction	Summary
undeploy_hosted_engine	Boolean	In	When set to true it means this host should un-deploy hosted engine components and this host will not function as part of the High Availability cluster.

6.97.1.1. deploy_hosted_engine

When set to **true** it means this host should deploy also hosted engine components. Missing value is treated as **true** i.e deploy. Omitting this parameter means **false** and will perform no operation in hosted engine area.

6.97.1.2. undeploy_hosted_engine

When set to **true** it means this host should un-deploy hosted engine components and this host will not function as part of the High Availability cluster. Missing value is treated as **true** i.e un-deploy. Omitting this parameter means **false** and will perform no operation in hosted engine area.

6.97.2. list GET

Get a list of all available hosts.

For example, to list the hosts send the following request:

```
GET /ovirt-engine/api/hosts
```

The response body will be something like this:

```
<hosts>
  <host href="/ovirt-engine/api/hosts/123" id="123">
    ...
  </host>
  <host href="/ovirt-engine/api/hosts/456" id="456">
    ...
  </host>
  ...
</hosts>
```

Table 6.304. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

Name	Type	Direction	Summary
hosts	Host[]	Out	
max	Integer	In	Sets the maximum number of hosts to return.
search	String	In	A query string used to restrict the returned hosts.

6.97.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.97.2.2. max

Sets the maximum number of hosts to return. If not specified all the hosts are returned.

6.98. ICON

A service to manage an icon (read-only).

Table 6.305. Methods summary

Name	Summary
get	Get an icon.

6.98.1. get GET

Get an icon.

```
GET /ovirt-engine/api/icons/123
```

You will get a XML response like this one:

```
<icon id="123">
  <data>Some binary data here</data>
  <media_type>image/png</media_type>
</icon>
```

Table 6.306. Parameters summary

Name	Type	Direction	Summary
icon	Icon	Out	Retrieved icon.

6.99. ICONS

A service to manage icons.

Table 6.307. Methods summary

Name	Summary
<code>list</code>	Get a list of icons.

6.99.1. list GET

Get a list of icons.

```
GET /ovirt-engine/api/icons
```

You will get a XML response which is similar to this one:

```
<icons>
  <icon id="123">
    <data>...</data>
    <media_type>image/png</media_type>
  </icon>
  ...
</icons>
```

Table 6.308. Parameters summary

Name	Type	Direction	Summary
<code>icons</code>	<code>Icon[]</code>	Out	Retrieved list of icons.
<code>max</code>	<code>Integer</code>	In	Sets the maximum number of icons to return.

6.99.1.1. max

Sets the maximum number of icons to return. If not specified all the icons are returned.

6.100. IMAGE

Table 6.309. Methods summary

Name	Summary
<code>get</code>	
<code>import</code>	Imports an image.

6.100.1. get GET

Table 6.310. Parameters summary

Name	Type	Direction	Summary
image	Image	Out	

6.100.2. import POST

Imports an image.

If the **import_as_template** parameter is **true** then the image will be imported as a template, otherwise it will be imported as a disk.

When imported as a template, the name of the template can be specified by the optional **template.name** parameter. If that parameter is not specified, then the name of the template will be automatically assigned by the engine as **GlanceTemplate-x** (where **x** will be seven random hexadecimal characters).

When imported as a disk, the name of the disk can be specified by the optional **disk.name** parameter. If that parameter is not specified, then the name of the disk will be automatically assigned by the engine as **GlanceDisk-x** (where **x** will be the seven hexadecimal characters of the image identifier).

It is recommended to always explicitly specify the template or disk name, to avoid these automatic names generated by the engine.

Table 6.311. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the import should be performed asynchronously.
cluster	Cluster	In	The cluster to which the image should be imported if the import_as_template parameter is set to true .
disk	Disk	In	The disk to import.
import_as_template	Boolean	In	Specifies if a template should be created from the imported disk.
storage_domain	StorageDomain	In	The storage domain to which the disk should be imported.
template	Template	In	The name of the template being created if the import_as_template parameter is set to true .

6.101. IMAGETRANSFER

This service provides a mechanism to control an image transfer. The client will have to create a transfer by using [add](#) of the [Section 6.102, “ImageTransfers”](#) service, stating the image to transfer data to/from.

After doing that, the transfer is managed by this service.

E.g., for uploading to the disk image with id **52cb593f-837c-4633-a444-35a0a0383706**, the client can use oVirt's Python's SDK as follows:

```
transfers_service = system_service.image_transfers_service()
transfer = transfers_service.add(
    types.ImageTransfer(
        image=types.Image(
            id='52cb593f-837c-4633-a444-35a0a0383706'
        )
    )
)
```

If the user wishes to download a disk rather than upload, he/she should specify **download** as the [direction](#) attribute of the transfer. This will grant a read permission from the image, instead of a write permission.

E.g:

```
transfers_service = system_service.image_transfers_service()
transfer = transfers_service.add(
    types.ImageTransfer(
        image=types.Image(
            id='52cb593f-837c-4633-a444-35a0a0383706'
        ),
        direction=types.ImageTransferDirection.DOWNLOAD
    )
)
```

Transfers have phases, which govern the flow of the upload/download. A client implementing such a flow should poll/check the transfer's phase and act accordingly. All the possible phases can be found in [ImageTransferPhase](#).

After adding a new transfer, its phase will be [initializing](#). The client will have to poll on the transfer's phase until it changes. When the phase becomes [transferring](#), the session is ready to start the transfer.

For example:

```
transfer_service = transfers_service.image_transfer_service(transfer.id)
while transfer.phase == types.ImageTransferPhase.INITIALIZING:
    time.sleep(3)
transfer = transfer_service.get()
```

At that stage, if the transfer's phase is [paused_system](#), then the session was not successfully established. One possible reason for that is that the `ovirt-imageio-daemon` is not running in the host that was selected for transfer. The transfer can be resumed by calling [resume](#) of the service that manages it.

If the session was successfully established - the returned transfer entity will contain the [proxy_url](#) and [signed_ticket](#) attributes, which the client needs to use in order to transfer the required data. The client can choose whatever technique and tool for sending the HTTPS request with the image's data.

- **proxy_url** is the address of a proxy server to the image, to do I/O to.

- **signed_ticket** is the content that needs to be added to the **Authentication** header in the HTTPS request, in order to perform a trusted communication.

For example, Python's `HTTPSConnection` can be used in order to perform a transfer, so an **transfer_headers** dict is set for the upcoming transfer:

```
transfer_headers = {
    'Authorization' : transfer.signed_ticket,
}
```

Using Python's **HTTPSConnection**, a new connection is established:

```
# Extract the URI, port, and path from the transfer's proxy_url.
url = urlparse.urlparse(transfer.proxy_url)

# Create a new instance of the connection.
proxy_connection = HTTPSConnection(
    url.hostname,
    url.port,
    context=ssl.SSLContext(ssl.PROTOCOL_SSLv23)
)
```

For upload, the specific content range being sent must be noted in the **Content-Range** HTTPS header. This can be used in order to split the transfer into several requests for a more flexible process.

For doing that, the client will have to repeatedly extend the transfer session to keep the channel open. Otherwise, the session will terminate and the transfer will get into **paused_system** phase, and HTTPS requests to the server will be rejected.

E.g., the client can iterate on chunks of the file, and send them to the proxy server while asking the service to extend the session:

```
path = "/path/to/image"
MB_per_request = 32
with open(path, "rb") as disk:
    size = os.path.getsize(path)
    chunk_size = 1024*1024*MB_per_request
    pos = 0
    while (pos < size):
        transfer_service.extend()
        transfer_headers['Content-Range'] = "bytes %d-%d/%d" % (pos, min(pos
+ chunk_size, size)-1, size)
        proxy_connection.request(
            'PUT',
            url.path,
            disk.read(chunk_size),
            headers=transfer_headers
        )
        r = proxy_connection.getresponse()
        print r.status, r.reason, "Completed", "{:.0%}".format(pos/
float(size))
        pos += chunk_size
```

Similarly, for a download transfer, a **Range** header must be sent, making the download process more easily managed by downloading the disk in chunks.

E.g., the client will again iterate on chunks of the disk image, but this time he/she will download it to a local file, rather than uploading its own file to the image:

```
output_file = "/home/user/downloaded_image"
MiB_per_request = 32
chunk_size = 1024*1024*MiB_per_request
total = disk_size

with open(output_file, "wb") as disk:
    pos = 0
    while pos < total:
        transfer_service.extend()
        transfer_headers['Range'] = "bytes=%d-%d" % (pos, min(total, pos +
chunk_size) - 1)
        proxy_connection.request('GET', proxy_url.path,
headers=transfer_headers)
        r = proxy_connection.getresponse()
        disk.write(r.read())
        print "Completed", "{:.0%}".format(pos/ float(total))
        pos += chunk_size
```

When finishing the transfer, the user should call `finalize`. This will make the final adjustments and verifications for finishing the transfer process.

For example:

```
transfer_service.finalize()
```

In case of an error, the transfer's phase will be changed to `finished_failure`, and the disk's status will be changed to **illegal**. Otherwise it will be changed to `finished_success`, and the disk will be ready to be used. In both cases, the transfer entity will be removed shortly after.

Table 6.312. Methods summary

Name	Summary
<code>extend</code>	Extend the image transfer session.
<code>finalize</code>	After finishing to transfer the data, finalize the transfer.
<code>get</code>	Get the image transfer entity.
<code>pause</code>	Pause the image transfer session.
<code>resume</code>	Resume the image transfer session.

6.101.1. extend POST

Extend the image transfer session.

6.101.2. finalize POST

After finishing to transfer the data, finalize the transfer.

This will make sure that the data being transferred is valid and fits the image entity that was targeted in the transfer. Specifically, will verify that if the image entity is a QCOW disk, the data uploaded is indeed a QCOW file, and that the image doesn't have a backing file.

6.101.3. get GET

Get the image transfer entity.

Table 6.313. Parameters summary

Name	Type	Direction	Summary
<code>image_transfer</code>	<code>ImageTransfer</code>	Out	

6.101.4. pause POST

Pause the image transfer session.

6.101.5. resume POST

Resume the image transfer session. The client will need to poll the transfer's phase until it is different than `resuming`. For example:

```
transfer_service = transfers_service.image_transfer_service(transfer.id)
transfer_service.resume()
transfer = transfer_service.get()

while transfer.phase == types.ImageTransferPhase.RESUMING:
    time.sleep(1)
    transfer = transfer_service.get()
```

6.102. IMAGETRANSFERS

This service manages image transfers, for performing Image I/O API in oVirt. Please refer to [image transfer](#) for further documentation.

Table 6.314. Methods summary

Name	Summary
<code>add</code>	Add a new image transfer.
<code>list</code>	Retrieves the list of image transfers that are currently being performed.

6.102.1. add POST

Add a new image transfer. An image needs to be specified in order to make a new transfer.

Table 6.315. Parameters summary

Name	Type	Direction	Summary
<code>image_transfer</code>	ImageTransfer	In/Out	

6.102.2. list GET

Retrieves the list of image transfers that are currently being performed.

Table 6.316. Parameters summary

Name	Type	Direction	Summary
<code>image_transfer</code>	ImageTransfer[]	Out	

6.103. IMAGES

Table 6.317. Methods summary

Name	Summary
<code>list</code>	

6.103.1. list GET

Table 6.318. Parameters summary

Name	Type	Direction	Summary
<code>images</code>	Image[]	Out	
<code>max</code>	Integer	In	Sets the maximum number of images to return.

6.103.1.1. max

Sets the maximum number of images to return. If not specified all the images are returned.

6.104. INSTANCETYPE

Table 6.319. Methods summary

Name	Summary
<code>get</code>	Get a specific instance type and it's attributes.

Name	Summary
remove	Removes a specific instance type from the system.
update	Update a specific instance type and it's attributes.

6.104.1. get GET

Get a specific instance type and it's attributes.

```
GET /ovirt-engine/api/instancetypes/123
```

Table 6.320. Parameters summary

Name	Type	Direction	Summary
instance_type	InstanceType	Out	

6.104.2. remove DELETE

Removes a specific instance type from the system.

If a virtual machine was created using an instance type X after removal of the instance type the virtual machine's instance type will be set to **custom**.

```
DELETE /ovirt-engine/api/instancetypes/123
```

Table 6.321. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.104.3. update PUT

Update a specific instance type and it's attributes.

All the attributes are editable after creation. If a virtual machine was created using an instance type X and some configuration in instance type X was updated, the virtual machine's configuration will be updated automatically by the engine.

```
PUT /ovirt-engine/api/instancetypes/123
```

For example, to update the memory of instance type **123** to 1 GiB and set the cpu topology to 2 sockets and 1 core, send a request like this:

```

<instance_type>
  <memory>1073741824</memory>
  <cpu>
    <topology>
      <cores>1</cores>
      <sockets>2</sockets>
      <threads>1</threads>
    </topology>
  </cpu>
</instance_type>

```

Table 6.322. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
instance_type	InstanceType	In/Out	

6.105. INSTANCETYPEGRAPHICSCONSOLE

Table 6.323. Methods summary

Name	Summary
get	Gets graphics console configuration of the instance type.
remove	Remove the graphics console from the instance type.

6.105.1. get GET

Gets graphics console configuration of the instance type.

Table 6.324. Parameters summary

Name	Type	Direction	Summary
console	GraphicsConsole	Out	The information about the graphics console of the instance type.

6.105.2. remove DELETE

Remove the graphics console from the instance type.

Table 6.325. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.106. INSTANCETYPEGRAPHICSCONSOLES

Table 6.326. Methods summary

Name	Summary
add	Add new graphics console to the instance type.
list	Lists all the configured graphics consoles of the instance type.

6.106.1. add POST

Add new graphics console to the instance type.

Table 6.327. Parameters summary

Name	Type	Direction	Summary
console	GraphicsConsole	In/Out	

6.106.2. list GET

Lists all the configured graphics consoles of the instance type.

Table 6.328. Parameters summary

Name	Type	Direction	Summary
consoles	GraphicsConsole[]	Out	The list of graphics consoles of the instance type.
max	Integer	In	Sets the maximum number of consoles to return.

6.106.2.1. max

Sets the maximum number of consoles to return. If not specified all the consoles are returned.

6.107. INSTANCETYPENIC

Table 6.329. Methods summary

Name	Summary
get	Gets network interface configuration of the instance type.
remove	Remove the network interface from the instance type.
update	Updates the network interface configuration of the instance type.

6.107.1. get GET

Gets network interface configuration of the instance type.

Table 6.330. Parameters summary

Name	Type	Direction	Summary
nic	Nic	Out	

6.107.2. remove DELETE

Remove the network interface from the instance type.

Table 6.331. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.107.3. update PUT

Updates the network interface configuration of the instance type.

Table 6.332. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
nic	Nic	In/Out	

6.108. INSTANCETYPEPENICS

Table 6.333. Methods summary

Name	Summary
add	Add new network interface to the instance type.
list	Lists all the configured network interface of the instance type.

6.108.1. add POST

Add new network interface to the instance type.

Table 6.334. Parameters summary

Name	Type	Direction	Summary
nic	Nic	In/Out	

6.108.2. list GET

Lists all the configured network interface of the instance type.

Table 6.335. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of NICs to return.
nics	Nic[]	Out	
search	String	In	A query string used to restrict the returned templates.

6.108.2.1. max

Sets the maximum number of NICs to return. If not specified all the NICs are returned.

6.109. INSTANCETYPEWATCHDOG

Table 6.336. Methods summary

Name	Summary
get	Gets watchdog configuration of the instance type.
remove	Remove a watchdog from the instance type.
update	Updates the watchdog configuration of the instance type.

6.109.1. get GET

Gets watchdog configuration of the instance type.

Table 6.337. Parameters summary

Name	Type	Direction	Summary
<code>watchdog</code>	Watchdog	Out	

6.109.2. remove DELETE

Remove a watchdog from the instance type.

Table 6.338. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the remove should be performed asynchronously.

6.109.3. update PUT

Updates the watchdog configuration of the instance type.

Table 6.339. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the update should be performed asynchronously.
<code>watchdog</code>	Watchdog	In/Out	

6.110. INSTANCETYPEWATCHDOGS

Table 6.340. Methods summary

Name	Summary
<code>add</code>	Add new watchdog to the instance type.
<code>list</code>	Lists all the configured watchdogs of the instance type.

6.110.1. add POST

Add new watchdog to the instance type.

Table 6.341. Parameters summary

Name	Type	Direction	Summary
watchdog	Watchdog	In/Out	

6.110.2. list GET

Lists all the configured watchdogs of the instance type.

Table 6.342. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of watchdogs to return.
search	String	In	A query string used to restrict the returned templates.
watchdogs	Watchdog[]	Out	

6.110.2.1. max

Sets the maximum number of watchdogs to return. If not specified all the watchdogs are returned.

6.111. INSTANCETYPES

Table 6.343. Methods summary

Name	Summary
add	Creates a new instance type.
list	Lists all existing instance types in the system.

6.111.1. add POST

Creates a new instance type.

This requires only a name attribute and can include all hardware configurations of the virtual machine.

```
POST /ovirt-engine/api/instancetypes
```

With a request body like this:

```
<instance_type>
  <name>myinstancetype</name>
</template>
```

Creating an instance type with all hardware configurations with a request body like this:

```
<instance_type>
  <name>myinstancetype</name>
  <console>
    <enabled>>true</enabled>
  </console>
  <cpu>
    <topology>
      <cores>2</cores>
      <sockets>2</sockets>
      <threads>1</threads>
    </topology>
  </cpu>
  <custom_cpu_model>AMD Opteron_G2</custom_cpu_model>
  <custom_emulated_machine>q35</custom_emulated_machine>
  <display>
    <monitors>1</monitors>
    <single_qxl_pci>true</single_qxl_pci>
    <smartcard_enabled>true</smartcard_enabled>
    <type>spice</type>
  </display>
  <high_availability>
    <enabled>true</enabled>
    <priority>1</priority>
  </high_availability>
  <io>
    <threads>2</threads>
  </io>
  <memory>4294967296</memory>
  <memory_policy>
    <ballooning>true</ballooning>
    <guaranteed>268435456</guaranteed>
  </memory_policy>
  <migration>
    <auto_converge>inherit</auto_converge>
    <compressed>inherit</compressed>
    <policy id="00000000-0000-0000-0000-000000000000"/>
  </migration>
  <migration_downtime>2</migration_downtime>
  <os>
    <boot>
      <devices>
        <device>hd</device>
      </devices>
    </boot>
  </os>
  <rng_device>
    <rate>
      <bytes>200</bytes>
      <period>2</period>
    </rate>
    <source>urandom</source>
  </rng_device>
  <soundcard_enabled>true</soundcard_enabled>
  <usb>
    <enabled>true</enabled>
    <type>native</type>
```

```

</usb>
<virtio_scsi>
  <enabled>true</enabled>
</virtio_scsi>
</instance_type>

```

Table 6.344. Parameters summary

Name	Type	Direction	Summary
instance_type	InstanceType	In/Out	

6.111.2. list GET

Lists all existing instance types in the system.

Table 6.345. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
instance_type	InstanceType[]	Out	
max	Integer	In	Sets the maximum number of instance types to return.
search	String	In	A query string used to restrict the returned templates.

6.111.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.111.2.2. max

Sets the maximum number of instance types to return. If not specified all the instance types are returned.

6.112. ISCSIBOND

Table 6.346. Methods summary

Name	Summary
get	

Name	Summary
remove	Removes of an existing iSCSI bond.
update	Updates an iSCSI bond.

6.112.1. get GET

Table 6.347. Parameters summary

Name	Type	Direction	Summary
bond	IscsiBond	Out	

6.112.2. remove DELETE

Removes of an existing iSCSI bond.

For example, to remove the iSCSI bond **456** send a request like this:

```
DELETE /ovirt-engine/api/datacenters/123/iscsibonds/456
```

Table 6.348. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.112.3. update PUT

Updates an iSCSI bond.

Updating of an iSCSI bond can be done on the **name** and the **description** attributes only. For example, to update the iSCSI bond **456** of data center **123**, send a request like this:

```
PUT /ovirt-engine/api/datacenters/123/iscsibonds/1234
```

The request body should look like this:

```
<iscsi_bond>
  <name>mybond</name>
  <description>My iSCSI bond</description>
</iscsi_bond>
```

Table 6.349. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
bond	IscsiBond	In/Out	

6.113. ISCSIBONDS

Table 6.350. Methods summary

Name	Summary
add	Create a new iSCSI bond on a data center.
list	

6.113.1. add POST

Create a new iSCSI bond on a data center.

For example, to create a new iSCSI bond on data center **123** using storage connections **456** and **789**, send a request like this:

```
POST /ovirt-engine/api/datacenters/123/iscsibonds
```

The request body should look like this:

```
<iscsi_bond>
  <name>mybond</name>
  <storage_connections>
    <storage_connection id="456"/>
    <storage_connection id="789"/>
  </storage_connections>
  <networks>
    <network id="abc"/>
  </networks>
</iscsi_bond>
```

Table 6.351. Parameters summary

Name	Type	Direction	Summary
bond	IscsiBond	In/Out	

6.113.2. list GET

Table 6.352. Parameters summary

Name	Type	Direction	Summary
bonds	iscsiBond[]	Out	
max	Integer	In	Sets the maximum number of bonds to return.

6.113.2.1. max

Sets the maximum number of bonds to return. If not specified all the bonds are returned.

6.114. JOB

A service to manage a job.

Table 6.353. Methods summary

Name	Summary
clear	Set an external job execution to be cleared by the system.
end	Marks an external job execution as ended.
get	Retrieves a job.

6.114.1. clear POST

Set an external job execution to be cleared by the system.

For example, to set a job with identifier **123** send the following request:

```
POST /ovirt-engine/api/jobs/clear
```

With the following request body:

```
<action/>
```

Table 6.354. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.

6.114.2. end POST

Marks an external job execution as ended.

For example, to terminate a job with identifier **123** send the following request:

```
POST /ovirt-engine/api/jobs/end
```

With the following request body:

```
<action>
  <force>true</force>
  <status>finished</status>
</action>
```

Table 6.355. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
force	Boolean	In	Indicates if the job should be forcibly terminated.
succeeded	Boolean	In	Indicates if the job should be marked as successfully finished or as failed.

6.114.2.1. succeeded

Indicates if the job should be marked as successfully finished or as failed.

This parameter is optional, and the default value is **true**.

6.114.3. get GET

Retrieves a job.

```
GET /ovirt-engine/api/jobs/123
```

You will receive response in XML like this one:

```
<job href="/ovirt-engine/api/jobs/123" id="123">
  <actions>
    <link href="/ovirt-engine/api/jobs/123/clear" rel="clear"/>
    <link href="/ovirt-engine/api/jobs/123/end" rel="end"/>
  </actions>
  <description>Adding Disk</description>
  <link href="/ovirt-engine/api/jobs/123/steps" rel="steps"/>
  <auto_cleared>true</auto_cleared>
  <end_time>2016-12-12T23:07:29.758+02:00</end_time>
  <external>>false</external>
  <last_updated>2016-12-12T23:07:29.758+02:00</last_updated>
  <start_time>2016-12-12T23:07:26.593+02:00</start_time>
  <status>failed</status>
  <owner href="/ovirt-engine/api/users/456" id="456"/>
</job>
```

Table 6.356. Parameters summary

Name	Type	Direction	Summary
job	Job	Out	Retrieves the representation of the job.

6.115. JOBS

A service to manage jobs.

Table 6.357. Methods summary

Name	Summary
add	Add an external job.
list	Retrieves the representation of the jobs.

6.115.1. add POST

Add an external job.

For example, to add a job with the following request:

```
POST /ovirt-engine/api/jobs
```

With the following request body:

```
<job>
  <description>Doing some work</description>
  <auto_cleared>>true</auto_cleared>
</job>
```

The response should look like:

```
<job href="/ovirt-engine/api/jobs/123" id="123">
  <actions>
    <link href="/ovirt-engine/api/jobs/123/clear" rel="clear"/>
    <link href="/ovirt-engine/api/jobs/123/end" rel="end"/>
  </actions>
  <description>Doing some work</description>
  <link href="/ovirt-engine/api/jobs/123/steps" rel="steps"/>
  <auto_cleared>>true</auto_cleared>
  <external>>true</external>
  <last_updated>2016-12-13T02:15:42.130+02:00</last_updated>
  <start_time>2016-12-13T02:15:42.130+02:00</start_time>
  <status>started</status>
  <owner href="/ovirt-engine/api/users/456" id="456"/>
</job>
```

Table 6.358. Parameters summary

Name	Type	Direction	Summary
job	Job	In/Out	Job that will be added.

6.115.2. list GET

Retrieves the representation of the jobs.

```
GET /ovirt-engine/api/jobs
```

You will receive response in XML like this one:

```
<jobs>
  <job href="/ovirt-engine/api/jobs/123" id="123">
    <actions>
      <link href="/ovirt-engine/api/jobs/123/clear" rel="clear"/>
      <link href="/ovirt-engine/api/jobs/123/end" rel="end"/>
    </actions>
    <description>Adding Disk</description>
    <link href="/ovirt-engine/api/jobs/123/steps" rel="steps"/>
    <auto_cleared>true</auto_cleared>
    <end_time>2016-12-12T23:07:29.758+02:00</end_time>
    <external>>false</external>
    <last_updated>2016-12-12T23:07:29.758+02:00</last_updated>
    <start_time>2016-12-12T23:07:26.593+02:00</start_time>
    <status>failed</status>
    <owner href="/ovirt-engine/api/users/456" id="456"/>
  </job>
  ...
</jobs>
```

Table 6.359. Parameters summary

Name	Type	Direction	Summary
jobs	Job[]	Out	A representation of jobs.
max	Integer	In	Sets the maximum number of jobs to return.

6.115.2.1. max

Sets the maximum number of jobs to return. If not specified all the jobs are returned.

6.116. KATELLOERRATA

A service to manage Katello errata. The information is retrieved from Katello.

Table 6.360. Methods summary

Name	Summary
list	Retrieves the representation of the Katello errata.

6.116.1. list GET

Retrieves the representation of the Katello errata.

```
GET /ovirt-engine/api/katelloerrata
```

You will receive response in XML like this one:

```
<katello_errata>
  <katello_erratum href="/ovirt-engine/api/katelloerrata/123" id="123">
    <name>RHBA-2013:XYZ</name>
    <description>The description of the erratum</description>
    <title>some bug fix update</title>
    <type>bugfix</type>
    <issued>2013-11-20T02:00:00.000+02:00</issued>
    <solution>Few guidelines regarding the solution</solution>
    <summary>Updated packages that fix one bug are now available for
XYZ</summary>
    <packages>
      <package>
        <name>libipa_hbac-1.9.2-82.11.el6_4.i686</name>
      </package>
      ...
    </packages>
  </katello_erratum>
  ...
</katello_errata>
```

Table 6.361. Parameters summary

Name	Type	Direction	Summary
errata	KatelloErratum[]	Out	A representation of Katello errata.
max	Integer	In	Sets the maximum number of errata to return.

6.116.1.1. max

Sets the maximum number of errata to return. If not specified all the errata are returned.

6.117. KATELLOERRATUM

A service to manage a Katello erratum.

Table 6.362. Methods summary

Name	Summary
get	Retrieves a Katello erratum.

6.117.1. get GET

Retrieves a Katello erratum.

```
GET /ovirt-engine/api/katelloerrata/123
```

You will receive response in XML like this one:

```
<katello_erratum href="/ovirt-engine/api/katelloerrata/123" id="123">
  <name>RHBA-2013:XYZ</name>
  <description>The description of the erratum</description>
  <title>some bug fix update</title>
  <type>bugfix</type>
  <issued>2013-11-20T02:00:00.000+02:00</issued>
  <solution>Few guidelines regarding the solution</solution>
  <summary>Updated packages that fix one bug are now available for
XYZ</summary>
  <packages>
    <package>
      <name>libipa_hbac-1.9.2-82.11.el6_4.i686</name>
    </package>
    ...
  </packages>
</katello_erratum>
```

Table 6.363. Parameters summary

Name	Type	Direction	Summary
erratum	KatelloErratum	Out	Retrieves the representation of the Katello erratum.

6.118. MACPOOL

Table 6.364. Methods summary

Name	Summary
get	
remove	Removes a MAC address pool.
update	Updates a MAC address pool.

6.118.1. get GET

Table 6.365. Parameters summary

Name	Type	Direction	Summary
<code>pool</code>	MacPool	Out	

6.118.2. remove DELETE

Removes a MAC address pool.

For example, to remove the MAC address pool having id **123** send a request like this:

```
DELETE /ovirt-engine/api/macpools/123
```

Table 6.366. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the remove should be performed asynchronously.

6.118.3. update PUT

Updates a MAC address pool.

The **name**, **description**, **allow_duplicates**, and **ranges** attributes can be updated.

For example, to update the MAC address pool of id **123** send a request like this:

```
PUT /ovirt-engine/api/macpools/123
```

With a request body like this:

```
<mac_pool>
  <name>UpdatedMACPool</name>
  <description>An updated MAC address pool</description>
  <allow_duplicates>>false</allow_duplicates>
  <ranges>
    <range>
      <from>00:1A:4A:16:01:51</from>
      <to>00:1A:4A:16:01:e6</to>
    </range>
    <range>
      <from>02:1A:4A:01:00:00</from>
      <to>02:1A:4A:FF:FF:FF</to>
    </range>
  </ranges>
</mac_pool>
```

Table 6.367. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
pool	MacPool	In/Out	

6.119. MACPOOLS

Table 6.368. Methods summary

Name	Summary
add	Creates a new MAC address pool.
list	

6.119.1. add POST

Creates a new MAC address pool.

Creation of a MAC address pool requires values for the **name** and **ranges** attributes.

For example, to create MAC address pool send a request like this:

```
POST /ovirt-engine/api/macpools
```

With a request body like this:

```
<mac_pool>
  <name>MACPool</name>
  <description>A MAC address pool</description>
  <allow_duplicates>>true</allow_duplicates>
  <default_pool>>false</default_pool>
  <ranges>
    <range>
      <from>00:1A:4A:16:01:51</from>
      <to>00:1A:4A:16:01:e6</to>
    </range>
  </ranges>
</mac_pool>
```

Table 6.369. Parameters summary

Name	Type	Direction	Summary
pool	MacPool	In/Out	

6.119.2. list GET

Table 6.370. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of pools to return.
pools	MacPool[]	Out	

6.119.2.1. max

Sets the maximum number of pools to return. If not specified all the pools are returned.

6.120. MEASURABLE

6.121. MOVEABLE

Table 6.371. Methods summary

Name	Summary
move	

6.121.1. move POST

Table 6.372. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the move should be performed asynchronously.

6.122. NETWORK

A service managing a network

Table 6.373. Methods summary

Name	Summary
get	Gets a logical network.
remove	Removes a logical network, or the association of a logical network to a data center.
update	Updates a logical network.

6.122.1. get GET

Gets a logical network.

For example:

```
GET /ovirt-engine/api/networks/123
```

Will respond:

```
<network href="/ovirt-engine/api/networks/123" id="123">
  <name>ovirtmgmt</name>
  <description>Default Management Network</description>
  <link href="/ovirt-engine/api/networks/123/permissions"
rel="permissions"/>
  <link href="/ovirt-engine/api/networks/123/vnicprofiles"
rel="vnicprofiles"/>
  <link href="/ovirt-engine/api/networks/123/networklabels"
rel="networklabels"/>
  <mtu>0</mtu>
  <stp>>false</stp>
  <usages>
    <usage>vm</usage>
  </usages>
  <data_center href="/ovirt-engine/api/datacenters/456" id="456"/>
</network>
```

Table 6.374. Parameters summary

Name	Type	Direction	Summary
network	Network	Out	

6.122.2. remove DELETE

Removes a logical network, or the association of a logical network to a data center.

For example, to remove the logical network **123** send a request like this:

```
DELETE /ovirt-engine/api/networks/123
```

Each network is bound exactly to one data center. So if we disassociate network with data center it has the same result as if we would just remove that network. However it might be more specific to say we're removing network **456** of data center **123**.

For example, to remove the association of network **456** to data center **123** send a request like this:

```
DELETE /ovirt-engine/api/datacenters/123/networks/456
```

Table 6.375. Parameters summary

Name	Type	Direction	Summary
------	------	-----------	---------

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.122.3. update PUT

Updates a logical network.

The **name**, **description**, **ip**, **vlan**, **stp** and **display** attributes can be updated.

For example, to update the description of the logical network **123** send a request like this:

```
PUT /ovirt-engine/api/networks/123
```

With a request body like this:

```
<network>
  <description>My updated description</description>
</network>
```

The maximum transmission unit of a network is set using a PUT request to specify the integer value of the **mtu** attribute.

For example, to set the maximum transmission unit send a request like this:

```
PUT /ovirt-engine/api/datacenters/123/networks/456
```

With a request body like this:

```
<network>
  <mtu>1500</mtu>
</network>
```

Table 6.376. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
network	Network	In/Out	

6.123. NETWORKATTACHMENT

Table 6.377. Methods summary

Name	Summary
get	
remove	
update	

6.123.1. get GET

Table 6.378. Parameters summary

Name	Type	Direction	Summary
attachmen t	NetworkAttac hment	Out	

6.123.2. remove DELETE

Table 6.379. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.123.3. update PUT

Table 6.380. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
attachmen t	NetworkAttac hment	In/Out	

6.124. NETWORKATTACHMENTS

Table 6.381. Methods summary

Name	Summary
add	

Name	Summary
list	

6.124.1. add POST

Table 6.382. Parameters summary

Name	Type	Direction	Summary
attachmen t	NetworkAttac hment	In/Out	

6.124.2. list GET

Table 6.383. Parameters summary

Name	Type	Direction	Summary
attachmen ts	NetworkAttac hment[]	Out	
max	Integer	In	Sets the maximum number of attachments to return.

6.124.2.1. max

Sets the maximum number of attachments to return. If not specified all the attachments are returned.

6.125. NETWORKFILTER

Manages a network filter.

```
<network_filter id="00000019-0019-0019-0019-00000000026b">
  <name>example-network-filter-b</name>
  <version>
    <major>4</major>
    <minor>0</minor>
    <build>-1</build>
    <revision>-1</revision>
  </version>
</network_filter>
```

Please note that version is referring to the minimal support version for the specific filter.

Table 6.384. Methods summary

Name	Summary
get	Retrieves a representation of the network filter.

6.125.1. get GET

Retrieves a representation of the network filter.

Table 6.385. Parameters summary

Name	Type	Direction	Summary
network_filter	NetworkFilter	Out	

6.126. NETWORKFILTERS

Represents a readonly network filters sub-collection.

The network filter enables to filter packets send to/from the VM's nic according to defined rules. For more information please refer to [NetworkFilter](#) service documentation

Network filters are supported in different versions, starting from version 3.0.

A network filter is defined for each vnic profile.

A vnic profile is defined for a specific network.

A network can be assigned to several different clusters. In the future, each network will be defined in cluster level.

Currently, each network is being defined at data center level. Potential network filters for each network are determined by the network's data center compatibility version V . V must be \geq the network filter version in order to configure this network filter for a specific network. Please note, that if a network is assigned to cluster with a version supporting a network filter, the filter may not be available due to the data center version being smaller then the network filter's version.

Example of listing all of the supported network filters for a specific cluster:

```
GET http://localhost:8080/ovirt-
engine/api/clusters/{cluster:id}/networkfilters
```

Output:

```
<network_filters>
  <network_filter id="00000019-0019-0019-0019-00000000026c">
    <name>example-network-filter-a</name>
    <version>
      <major>4</major>
      <minor>0</minor>
      <build>-1</build>
      <revision>-1</revision>
```

```

    </version>
  </network_filter>
  <network_filter id="00000019-0019-0019-0019-00000000026b">
    <name>example-network-filter-b</name>
    <version>
      <major>4</major>
      <minor>0</minor>
      <build>-1</build>
      <revision>-1</revision>
    </version>
  </network_filter>
  <network_filter id="00000019-0019-0019-0019-00000000026a">
    <name>example-network-filter-a</name>
    <version>
      <major>3</major>
      <minor>0</minor>
      <build>-1</build>
      <revision>-1</revision>
    </version>
  </network_filter>
</network_filters>

```

Table 6.386. Methods summary

Name	Summary
list	Retrieves the representations of the network filters.

6.126.1. list GET

Retrieves the representations of the network filters.

Table 6.387. Parameters summary

Name	Type	Direction	Summary
filters	NetworkFilter []	Out	

6.127. NETWORKLABEL

Table 6.388. Methods summary

Name	Summary
get	
remove	Removes a label from a logical network.

6.127.1. get GET

Table 6.389. Parameters summary

Name	Type	Direction	Summary
label	NetworkLabel	Out	

6.127.2. remove DELETE

Removes a label from a logical network.

For example, to remove the label **exemplary** from a logical network having id **123** send the following request:

```
DELETE /ovirt-engine/api/networks/123/labels/exemplary
```

Table 6.390. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.128. NETWORKLABELS

Table 6.391. Methods summary

Name	Summary
add	Attaches label to logical network.
list	

6.128.1. add POST

Attaches label to logical network.

You can attach labels to a logical network to automate the association of that logical network with physical host network interfaces to which the same label has been attached.

For example, to attach the label **mylabel** to a logical network having id **123** send a request like this:

```
POST /ovirt-engine/api/networks/123/labels
```

With a request body like this:

```
<label id="mylabel"/>
```

Table 6.392. Parameters summary

Name	Type	Direction	Summary
label	NetworkLabel	In/Out	

6.128.2. list GET

Table 6.393. Parameters summary

Name	Type	Direction	Summary
labels	NetworkLabel[]	Out	
max	Integer	In	Sets the maximum number of labels to return.

6.128.2.1. max

Sets the maximum number of labels to return. If not specified all the labels are returned.

6.129. NETWORKS

Manages logical networks.

The engine creates a default **ovirtmgmt** network on installation. This network acts as the management network for access to hypervisor hosts. This network is associated with the **Default** cluster and is a member of the **Default** data center.

Table 6.394. Methods summary

Name	Summary
add	Creates a new logical network, or associates an existing network with a data center.
list	List logical networks.

6.129.1. add POST

Creates a new logical network, or associates an existing network with a data center.

Creation of a new network requires the **name** and **data_center** elements.

For example, to create a network named **mynetwork** for data center **123** send a request like this:

```
POST /ovirt-engine/api/networks
```

With a request body like this:

```
<network>
```

```
<name>mynetwork</name>
<data_center id="123"/>
</network>
```

To associate the existing network **456** with the data center **123** send a request like this:

```
POST /ovirt-engine/api/datacenters/123/networks
```

With a request body like this:

```
<network>
  <name>ovirtmgmt</name>
</network>
```

Table 6.395. Parameters summary

Name	Type	Direction	Summary
network	Network	In/Out	

6.129.2. list GET

List logical networks.

For example:

```
GET /ovirt-engine/api/networks
```

Will respond:

```
<networks>
  <network href="/ovirt-engine/api/networks/123" id="123">
    <name>ovirtmgmt</name>
    <description>Default Management Network</description>
    <link href="/ovirt-engine/api/networks/123/permissions"
rel="permissions"/>
    <link href="/ovirt-engine/api/networks/123/vnicprofiles"
rel="vnicprofiles"/>
    <link href="/ovirt-engine/api/networks/123/networklabels"
rel="networklabels"/>
    <mtu>0</mtu>
    <stp>>false</stp>
    <usages>
      <usage>vm</usage>
    </usages>
    <data_center href="/ovirt-engine/api/datacenters/456" id="456"/>
  </network>
  ...
</networks>
```

Table 6.396. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
max	Integer	In	Sets the maximum number of networks to return.
networks	Network[]	Out	
search	String	In	A query string used to restrict the returned networks.

6.129.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.129.2.2. max

Sets the maximum number of networks to return. If not specified all the networks are returned.

6.130. OPENSTACKIMAGE

Table 6.397. Methods summary

Name	Summary
get	
import	Imports a virtual machine from a Glance image storage domain.

6.130.1. get GET

Table 6.398. Parameters summary

Name	Type	Direction	Summary
image	OpenStackImage	Out	

6.130.2. import POST

Imports a virtual machine from a Glance image storage domain.

For example, to import the image with identifier **456** from the storage domain with identifier **123** send a request like this:

```
POST /ovirt-engine/api/openstackimageproviders/123/images/456/import
```

With a request body like this:

```
<action>
  <storage_domain>
    <name>images0</name>
  </storage_domain>
  <cluster>
    <name>images0</name>
  </cluster>
</action>
```

Table 6.399. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the import should be performed asynchronously.
cluster	Cluster	In	This parameter is mandatory in case of using import_as_template and indicates which cluster should be used for import glance image as template.
disk	Disk	In	
import_as_template	Boolean	In	Indicates whether the image should be imported as a template.
storage_domain	StorageDomain	In	
template	Template	In	

6.131. OPENSTACKIMAGEPROVIDER

Table 6.400. Methods summary

Name	Summary
get	
importcertificates	
remove	

Name	Summary
testconnectivity	
update	

6.131.1. get GET

Table 6.401. Parameters summary

Name	Type	Direction	Summary
provider	OpenStackImageProvider	Out	

6.131.2. importcertificates POST

Table 6.402. Parameters summary

Name	Type	Direction	Summary
certificates	Certificate[]	In	

6.131.3. remove DELETE

Table 6.403. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.131.4. testconnectivity POST

Table 6.404. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the test should be performed asynchronously.

6.131.5. update PUT

Table 6.405. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
provider	OpenStackImageProvider	In/Out	

6.132. OPENSTACKIMAGEPROVIDERS

Table 6.406. Methods summary

Name	Summary
add	
list	

6.132.1. add POST

Table 6.407. Parameters summary

Name	Type	Direction	Summary
provider	OpenStackImageProvider	In/Out	

6.132.2. list GET

Table 6.408. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of providers to return.
providers	OpenStackImageProvider[]	Out	

6.132.2.1. max

Sets the maximum number of providers to return. If not specified all the providers are returned.

6.133. OPENSTACKIMAGES

Table 6.409. Methods summary

Name	Summary
<code>list</code>	Lists the images of a Glance image storage domain.

6.133.1. list GET

Lists the images of a Glance image storage domain.

Table 6.410. Parameters summary

Name	Type	Direction	Summary
<code>images</code>	OpenStack image[]	Out	
<code>max</code>	Integer	In	Sets the maximum number of images to return.

6.133.1.1. max

Sets the maximum number of images to return. If not specified all the images are returned.

6.134. OPENSTACKNETWORK

Table 6.411. Methods summary

Name	Summary
<code>get</code>	
<code>import</code>	This operation imports an external network into oVirt.

6.134.1. get GET

Table 6.412. Parameters summary

Name	Type	Direction	Summary
<code>network</code>	OpenStackNetwork	Out	

6.134.2. import POST

This operation imports an external network into oVirt. The network will be added to the data center specified.

Table 6.413. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the import should be performed asynchronously.
data_center	DataCenter	In	The data center into which the network is to be imported.

6.134.2.1. data_center

The data center into which the network is to be imported. Data center is mandatory, and can be specified using the **id** or **name** attributes, the rest of the attributes will be ignored.

6.135. OPENSTACKNETWORKPROVIDER

This service manages OpenStack network provider.

Table 6.414. Methods summary

Name	Summary
get	Returns the representation of the object managed by this service.
importcertificates	
remove	Removes the provider.
testconnectivity	
update	Updates the provider.

6.135.1. get GET

Returns the representation of the object managed by this service.

For example, to get the OpenStack network provider with identifier **1234**, send a request like this:

```
GET /ovirt-engine/api/openstacknetworkproviders/1234
```

Table 6.415. Parameters summary

Name	Type	Direction	Summary
provider	OpenStackNetworkProvider	Out	

6.135.2. importcertificates POST

Table 6.416. Parameters summary

Name	Type	Direction	Summary
certificates	Certificate[]	In	

6.135.3. remove DELETE

Removes the provider.

For example, to remove the OpenStack network provider with identifier **1234**, send a request like this:

```
DELETE /ovirt-engine/api/openstacknetworkproviders/1234
```

Table 6.417. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.135.4. testconnectivity POST

Table 6.418. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the test should be performed asynchronously.

6.135.5. update PUT

Updates the provider.

For example, to update **provider_name**, **requires_authentication**, **url**, **tenant_name** and **type** properties, for the OpenStack network provider with identifier **1234**, send a request like this:

```
PUT /ovirt-engine/api/openstacknetworkproviders/1234
```

With a request body like this:

```
<openstack_network_provider>
  <name>ovn-network-provider</name>
  <requires_authentication>>false</requires_authentication>
  <url>http://some_server_url.domain.com:9696</url>
```

```

<tenant_name>oVirt</tenant_name>
<type>external</type>
</openstack_network_provider>

```

Table 6.419. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
provider	OpenStackNetworkProvider	In/Out	The provider to update.

6.136. OPENSTACKNETWORKPROVIDERS

This service manages OpenStack network providers.

Table 6.420. Methods summary

Name	Summary
add	The operation adds a new network provider to the system.
list	

6.136.1. add POST

The operation adds a new network provider to the system. If the **type** property is not present, a default value of **NEUTRON** will be used.

Table 6.421. Parameters summary

Name	Type	Direction	Summary
provider	OpenStackNetworkProvider	In/Out	

6.136.2. list GET

Table 6.422. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of providers to return.

Name	Type	Direction	Summary
providers	OpenStackNetworkProvider[]	Out	

6.136.2.1. max

Sets the maximum number of providers to return. If not specified all the providers are returned.

6.137. OPENSTACKNETWORKS

Table 6.423. Methods summary

Name	Summary
list	

6.137.1. list GET

Table 6.424. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of networks to return.
networks	OpenStackNetwork[]	Out	

6.137.1.1. max

Sets the maximum number of networks to return. If not specified all the networks are returned.

6.138. OPENSTACKSUBNET

Table 6.425. Methods summary

Name	Summary
get	
remove	

6.138.1. get GET

Table 6.426. Parameters summary

Name	Type	Direction	Summary
subnet	OpenStackS ubnet	Out	

6.138.2. remove DELETE

Table 6.427. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.139. OPENSTACKSUBNETS

Table 6.428. Methods summary

Name	Summary
add	
list	

6.139.1. add POST

Table 6.429. Parameters summary

Name	Type	Direction	Summary
subnet	OpenStackS ubnet	In/Out	

6.139.2. list GET

Table 6.430. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of sub-networks to return.
subnets	OpenStackS ubnet[]	Out	

6.139.2.1. max

Sets the maximum number of sub-networks to return. If not specified all the sub-networks are returned.

6.140. OPENSTACKVOLUMEAUTHENTICATIONKEY

Table 6.431. Methods summary

Name	Summary
get	
remove	
update	

6.140.1. get GET

Table 6.432. Parameters summary

Name	Type	Direction	Summary
key	OpenstackVolumeAuthenticationKey	Out	

6.140.2. remove DELETE

Table 6.433. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.140.3. update PUT

Table 6.434. Parameters summary

Name	Type	Direction	Summary
key	OpenstackVolumeAuthenticationKey	In/Out	

6.141. OPENSTACKVOLUMEAUTHENTICATIONKEYS

Table 6.435. Methods summary

Name	Summary
add	
list	

6.141.1. add POST

Table 6.436. Parameters summary

Name	Type	Direction	Summary
key	OpenstackVolumeAuthenticationKey	In/Out	

6.141.2. list GET

Table 6.437. Parameters summary

Name	Type	Direction	Summary
keys	OpenstackVolumeAuthenticationKey[]	Out	
max	Integer	In	Sets the maximum number of keys to return.

6.141.2.1. max

Sets the maximum number of keys to return. If not specified all the keys are returned.

6.142. OPENSTACKVOLUMEPROVIDER

Table 6.438. Methods summary

Name	Summary
get	
importcertificates	
remove	
testconnectivity	

Name	Summary
update	

6.142.1. get GET

Table 6.439. Parameters summary

Name	Type	Direction	Summary
provider	OpenStackVolumeProvider	Out	

6.142.2. importcertificates POST

Table 6.440. Parameters summary

Name	Type	Direction	Summary
certificates	Certificate[]	In	

6.142.3. remove DELETE

Table 6.441. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.142.4. testconnectivity POST

Table 6.442. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the test should be performed asynchronously.

6.142.5. update PUT

Table 6.443. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
provider	OpenStackVolumeProvider	In/Out	

6.143. OPENSTACKVOLUMEPROVIDERS

Table 6.444. Methods summary

Name	Summary
add	Adds a new volume provider.
list	Retrieves the list of volume providers.

6.143.1. add POST

Adds a new volume provider.

For example:

```
POST /ovirt-engine/api/openstackvolumeproviders
```

With a request body like this:

```
<openstack_volume_provider>
  <name>mycinder</name>
  <url>https://mycinder.example.com:8776</url>
  <data_center>
    <name>mydc</name>
  </data_center>
  <requires_authentication>true</requires_authentication>
  <username>admin</username>
  <password>mypassword</password>
  <tenant_name>mytenant</tenant_name>
</openstack_volume_provider>
```

Table 6.445. Parameters summary

Name	Type	Direction	Summary
provider	OpenStackVolumeProvider	In/Out	

6.143.2. list GET

Retrieves the list of volume providers.

Table 6.446. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of providers to return.
<code>providers</code>	OpenStackVolumeProvider[]	Out	

6.143.2.1. max

Sets the maximum number of providers to return. If not specified all the providers are returned.

6.144. OPENSTACKVOLUMETYPE

Table 6.447. Methods summary

Name	Summary
<code>get</code>	

6.144.1. get GET

Table 6.448. Parameters summary

Name	Type	Direction	Summary
<code>type</code>	OpenStackVolumeType	Out	

6.145. OPENSTACKVOLUMETYPES

Table 6.449. Methods summary

Name	Summary
<code>list</code>	

6.145.1. list GET

Table 6.450. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of volume types to return.
types	OpenStackVolumeType[]	Out	

6.145.1.1. max

Sets the maximum number of volume types to return. If not specified all the volume types are returned.

6.146. OPERATINGSYSTEM

Table 6.451. Methods summary

Name	Summary
get	

6.146.1. get GET

Table 6.452. Parameters summary

Name	Type	Direction	Summary
operating_system	OperatingSystemInfo	Out	

6.147. OPERATINGSYSTEMS

Table 6.453. Methods summary

Name	Summary
list	

6.147.1. list GET

Table 6.454. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of networks to return.
operating_system	OperatingSystemInfo[]	Out	

6.147.1.1. max

Sets the maximum number of networks to return. If not specified all the networks are returned.

6.148. PERMISSION

Table 6.455. Methods summary

Name	Summary
get	
remove	

6.148.1. get GET

Table 6.456. Parameters summary

Name	Type	Direction	Summary
permission	Permission	Out	

6.148.2. remove DELETE

Table 6.457. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.149. PERMIT

A service to manage a specific permit of the role.

Table 6.458. Methods summary

Name	Summary
get	Gets the information about the permit of the role.
remove	Removes the permit from the role.

6.149.1. get GET

Gets the information about the permit of the role.

For example to retrieve the information about the permit with the id **456** of the role with the id **123** send a request like this:

```
GET /ovirt-engine/api/roles/123/permits/456
```

```
<permit href="/ovirt-engine/api/roles/123/permits/456" id="456">
  <name>change_vm_cd</name>
  <administrative>false</administrative>
  <role href="/ovirt-engine/api/roles/123" id="123"/>
</permit>
```

Table 6.459. Parameters summary

Name	Type	Direction	Summary
permit	Permit	Out	The permit of the role.

6.149.2. remove DELETE

Removes the permit from the role.

For example to remove the permit with id **456** from the role with id **123** send a request like this:

```
DELETE /ovirt-engine/api/roles/123/permits/456
```

Table 6.460. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.150. PERMITS

Represents a permits sub-collection of the specific role.

Table 6.461. Methods summary

Name	Summary
add	Adds a permit to the role.
list	List the permits of the role.

6.150.1. add POST

Adds a permit to the role. The permit name can be retrieved from the [Section 6.31, “ClusterLevels”](#) service.

For example to assign a permit `create_vm` to the role with id `123` send a request like this:

```
POST /ovirt-engine/api/roles/123/permits
```

With a request body like this:

```
<permit>
  <name>create_vm</name>
</permit>
```

Table 6.462. Parameters summary

Name	Type	Direction	Summary
<code>permit</code>	Permit	In/Out	The permit to add.

6.150.2. list GET

List the permits of the role.

For example to list the permits of the role with the id `123` send a request like this:

```
GET /ovirt-engine/api/roles/123/permits
```

```
<permits>
  <permit href="/ovirt-engine/api/roles/123/permits/5" id="5">
    <name>change_vm_cd</name>
    <administrative>false</administrative>
    <role href="/ovirt-engine/api/roles/123" id="123"/>
  </permit>
  <permit href="/ovirt-engine/api/roles/123/permits/7" id="7">
    <name>connect_to_vm</name>
    <administrative>false</administrative>
    <role href="/ovirt-engine/api/roles/123" id="123"/>
  </permit>
</permits>
```

Table 6.463. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of permits to return.
<code>permits</code>	Permit[]	Out	List of permits.

6.150.2.1. max

Sets the maximum number of permits to return. If not specified all the permits are returned.

6.151. QOS

Table 6.464. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	
<code>update</code>	

6.151.1. `get` GET

Table 6.465. Parameters summary

Name	Type	Direction	Summary
<code>qos</code>	Qos	Out	

6.151.2. `remove` DELETE

Table 6.466. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the remove should be performed asynchronously.

6.151.3. `update` PUT

Table 6.467. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the update should be performed asynchronously.
<code>qos</code>	Qos	In/Out	

6.152. QOSS

Table 6.468. Methods summary

Name	Summary
<code>add</code>	
<code>list</code>	

Name	Summary
------	---------

6.152.1. add POST

Table 6.469. Parameters summary

Name	Type	Direction	Summary
qos	Qos	In/Out	

6.152.2. list GET

Table 6.470. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of QoS descriptors to return.
qoss	Qos[]	Out	

6.152.2.1. max

Sets the maximum number of QoS descriptors to return. If not specified all the descriptors are returned.

6.153. QUOTA

Table 6.471. Methods summary

Name	Summary
get	Retrieves a quota.
remove	Delete a quota.
update	Updates a quota.

6.153.1. get GET

Retrieves a quota.

An example of retrieving a quota:

```
GET /ovirt-engine/api/datacenters/123/quotas/456
```



```
<quota id="456">
  <name>myquota</name>
  <description>My new quota for virtual machines</description>
  <cluster_hard_limit_pct>20</cluster_hard_limit_pct>
  <cluster_soft_limit_pct>80</cluster_soft_limit_pct>
  <storage_hard_limit_pct>20</storage_hard_limit_pct>
  <storage_soft_limit_pct>80</storage_soft_limit_pct>
</quota>
```

Table 6.472. Parameters summary

Name	Type	Direction	Summary
quota	Quota	Out	

6.153.2. remove DELETE

Delete a quota.

An example of deleting a quota:

```
DELETE /ovirt-engine/api/datacenters/123-456/quotas/654-321
-0472718ab224 HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

Table 6.473. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.153.3. update PUT

Updates a quota.

An example of updating a quota:

```
PUT /ovirt-engine/api/datacenters/123/quotas/456
```

```
<quota>
  <cluster_hard_limit_pct>30</cluster_hard_limit_pct>
  <cluster_soft_limit_pct>70</cluster_soft_limit_pct>
  <storage_hard_limit_pct>20</storage_hard_limit_pct>
  <storage_soft_limit_pct>80</storage_soft_limit_pct>
</quota>
```

Table 6.474. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
quota	Quota	In/Out	

6.154. QUOTA_CLUSTER_LIMIT

Table 6.475. Methods summary

Name	Summary
get	
remove	

6.154.1. get GET

Table 6.476. Parameters summary

Name	Type	Direction	Summary
limit	QuotaClusterLimit	Out	

6.154.2. remove DELETE

Table 6.477. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.155. QUOTA_CLUSTER_LIMITS

Table 6.478. Methods summary

Name	Summary
add	
list	

6.155.1. add POST

Table 6.479. Parameters summary

Name	Type	Direction	Summary
<code>limit</code>	QuotaClusterLimit	In/Out	

6.155.2. list GET

Table 6.480. Parameters summary

Name	Type	Direction	Summary
<code>limits</code>	QuotaClusterLimit[]	Out	
<code>max</code>	Integer	In	Sets the maximum number of limits to return.

6.155.2.1. max

Sets the maximum number of limits to return. If not specified all the limits are returned.

6.156. QUOTASTORAGELIMIT

Table 6.481. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	

6.156.1. get GET

Table 6.482. Parameters summary

Name	Type	Direction	Summary
<code>limit</code>	QuotaStorageLimit	Out	

6.156.2. remove DELETE

Table 6.483. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.

6.157. QUOTASTORAGELIMITS

Table 6.484. Methods summary

Name	Summary
add	
list	

6.157.1. add POST

Table 6.485. Parameters summary

Name	Type	Direction	Summary
limit	QuotaStorageLimit	In/Out	

6.157.2. list GET

Table 6.486. Parameters summary

Name	Type	Direction	Summary
limits	QuotaStorageLimit[]	Out	
max	Integer	In	Sets the maximum number of limits to return.

6.157.2.1. max

Sets the maximum number of limits to return. If not specified all the limits are returned.

6.158. QUOTAS

Table 6.487. Methods summary

Name	Summary
add	Creates a new quota.

Name	Summary
list	Lists quotas of a data center

6.158.1. add POST

Creates a new quota.

An example of creating a new quota:

```
POST /ovirt-engine/api/datacenters/123/quotas
```

```
<quota>
  <name>myquota</name>
  <description>My new quota for virtual machines</description>
</quota>
```

Table 6.488. Parameters summary

Name	Type	Direction	Summary
quota	Quota	In/Out	

6.158.2. list GET

Lists quotas of a data center

Table 6.489. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of quota descriptors to return.
quotas	Quota[]	Out	

6.158.2.1. max

Sets the maximum number of quota descriptors to return. If not specified all the descriptors are returned.

6.159. ROLE

Table 6.490. Methods summary

Name	Summary
get	Get the role.

Name	Summary
remove	Removes the role.
update	Updates a role.

6.159.1. get GET

Get the role.

```
GET /ovirt-engine/api/roles/123
```

You will receive XML response like this one:

```
<role id="123">
  <name>MyRole</name>
  <description>MyRole description</description>
  <link href="/ovirt-engine/api/roles/123/permits" rel="permits"/>
  <administrative>true</administrative>
  <mutable>false</mutable>
</role>
```

Table 6.491. Parameters summary

Name	Type	Direction	Summary
role	Role	Out	Retrieved role.

6.159.2. remove DELETE

Removes the role.

To remove the role you need to know its id, then send request like this:

```
DELETE /ovirt-engine/api/roles/{role_id}
```

Table 6.492. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.159.3. update PUT

Updates a role. You are allowed to update **name**, **description** and **administrative** attributes after role is created. Within this endpoint you can't add or remove roles permits you need to use [service](#) that manages permits of role.

For example to update role's **name**, **description** and **administrative** attributes send a request like this:

```
PUT /ovirt-engine/api/roles/123
```

With a request body like this:

```
<role>
  <name>MyNewRoleName</name>
  <description>My new description of the role</description>
  <administrative>true</administrative>
</group>
```

Table 6.493. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
role	Role	In/Out	Updated role.

6.160. ROLES

Provides read-only access to the global set of roles

Table 6.494. Methods summary

Name	Summary
add	Create a new role.
list	List roles.

6.160.1. add POST

Create a new role. The role can be administrative or non-administrative and can have different permits.

For example, to add the **MyRole** non-administrative role with permits to login and create virtual machines send a request like this (note that you have to pass permit id):

```
POST /ovirt-engine/api/roles
```

With a request body like this:

```
<role>
  <name>MyRole</name>
  <description>My custom role to create virtual machines</description>
  <administrative>false</administrative>
  <permits>
```

```

    <permit id="1"/>
    <permit id="1300"/>
  </permits>
</group>

```

Table 6.495. Parameters summary

Name	Type	Direction	Summary
role	Role	In/Out	Role that will be added.

6.160.2. list GET

List roles.

```
GET /ovirt-engine/api/roles
```

You will receive response in XML like this one:

```

<roles>
  <role id="123">
    <name>SuperUser</name>
    <description>Roles management administrator</description>
    <link href="/ovirt-engine/api/roles/123/permits" rel="permits"/>
    <administrative>true</administrative>
    <mutable>false</mutable>
  </role>
  ...
</roles>

```

Table 6.496. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of roles to return.
roles	Role[]	Out	Retrieved list of roles.

6.160.2.1. max

Sets the maximum number of roles to return. If not specified all the roles are returned.

6.161. SCHEDULINGPOLICIES

Table 6.497. Methods summary

Name	Summary
add	

Name	Summary
<code>list</code>	

6.161.1. add POST

Table 6.498. Parameters summary

Name	Type	Direction	Summary
<code>policy</code>	SchedulingPolicy	In/Out	

6.161.2. list GET

Table 6.499. Parameters summary

Name	Type	Direction	Summary
<code>filter</code>	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
<code>max</code>	Integer	In	Sets the maximum number of policies to return.
<code>policies</code>	SchedulingPolicy[]	Out	

6.161.2.1. max

Sets the maximum number of policies to return. If not specified all the policies are returned.

6.162. SCHEDULINGPOLICY

Table 6.500. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	
<code>update</code>	

6.162.1. get GET

Table 6.501. Parameters summary

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
policy	SchedulingPolicy	Out	

6.162.2. remove DELETE

Table 6.502. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.162.3. update PUT

Table 6.503. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
policy	SchedulingPolicy	In/Out	

6.163. SCHEDULINGPOLICYUNIT

Table 6.504. Methods summary

Name	Summary
get	
remove	

6.163.1. get GET

Table 6.505. Parameters summary

Name	Type	Direction	Summary
------	------	-----------	---------

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
unit	SchedulingPolicyUnit	Out	

6.163.2. remove DELETE

Table 6.506. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.164. SCHEDULINGPOLICYUNITS

Table 6.507. Methods summary

Name	Summary
list	

6.164.1. list GET

Table 6.508. Parameters summary

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
max	Integer	In	Sets the maximum number of policy units to return.
units	SchedulingPolicyUnit[]	Out	

6.164.1.1. max

Sets the maximum number of policy units to return. If not specified all the policy units are returned.

6.165. SNAPSHOT

Table 6.509. Methods summary

Name	Summary
get	
remove	
restore	Restores a virtual machine snapshot.

6.165.1. get GET

Table 6.510. Parameters summary

Name	Type	Direction	Summary
snapshot	Snapshot	Out	

6.165.2. remove DELETE

Table 6.511. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.165.3. restore POST

Restores a virtual machine snapshot.

For example, to restore the snapshot with identifier **456** of virtual machine with identifier **123** send a request like this:

```
POST /ovirt-engine/api/vms/123/snapshots/456/restore
```

With an empty **action** in the body:

```
<action/>
```

Table 6.512. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the restore should be performed asynchronously.
disks	Disk[]	In	

Name	Type	Direction	Summary
<code>restore_memory</code>	Boolean	In	

6.166. SNAPSHOTCDROM

Table 6.513. Methods summary

Name	Summary
<code>get</code>	

6.166.1. get GET

Table 6.514. Parameters summary

Name	Type	Direction	Summary
<code>cdrom</code>	Cdrom	Out	

6.167. SNAPSHOTCDROMS

Table 6.515. Methods summary

Name	Summary
<code>list</code>	

6.167.1. list GET

Table 6.516. Parameters summary

Name	Type	Direction	Summary
<code>cdroms</code>	Cdrom[]	Out	
<code>max</code>	Integer	In	Sets the maximum number of CDROMS to return.

6.167.1.1. max

Sets the maximum number of CDROMS to return. If not specified all the CDROMS are returned.

6.168. SNAPSHOTDISK

Table 6.517. Methods summary

Name	Summary
<code>get</code>	

6.168.1. `get` GET

Table 6.518. Parameters summary

Name	Type	Direction	Summary
<code>disk</code>	Disk	Out	

6.169. SNAPSHOTDISKS

Table 6.519. Methods summary

Name	Summary
<code>list</code>	

6.169.1. `list` GET

Table 6.520. Parameters summary

Name	Type	Direction	Summary
<code>disks</code>	Disk[]	Out	
<code>max</code>	Integer	In	Sets the maximum number of disks to return.

6.169.1.1. `max`

Sets the maximum number of disks to return. If not specified all the disks are returned.

6.170. SNAPSHOTNIC

Table 6.521. Methods summary

Name	Summary
<code>get</code>	

6.170.1. `get` GET

Table 6.522. Parameters summary

Name	Type	Direction	Summary
<code>nic</code>	<code>Nic</code>	Out	

6.171. SNAPSHOTNICS

Table 6.523. Methods summary

Name	Summary
<code>list</code>	

6.171.1. list GET

Table 6.524. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	<code>Integer</code>	In	Sets the maximum number of NICs to return.
<code>nics</code>	<code>Nic[]</code>	Out	

6.171.1.1. max

Sets the maximum number of NICs to return. If not specified all the NICs are returned.

6.172. SNAPSNOTS

Table 6.525. Methods summary

Name	Summary
<code>add</code>	Creates a virtual machine snapshot.
<code>list</code>	

6.172.1. add POST

Creates a virtual machine snapshot.

For example, to create a new snapshot for virtual machine **123** send a request like this:

```
POST /ovirt-engine/api/vms/123/snapshots
```

With a request body like this:

```
<snapshot>
  <description>My snapshot</description>
</snapshot>
```

IMPORTANT

When a snapshot is created the default value for the `persist_memorystate` attribute is `true`. That means that the content of the memory of the virtual machine will be included in the snapshot, and it also means that the virtual machine will be paused for a longer time. That can negatively affect applications that are very sensitive to timing (NTP servers, for example). In those cases make sure that you set the attribute to `false`:

```
<snapshot>
  <description>My snapshot</description>
  <persist_memorystate>>false</persist_memorystate>
</snapshot>
```

Table 6.526. Parameters summary

Name	Type	Direction	Summary
<code>snapshot</code>	<code>Snapshot</code>	In/Out	

6.172.2. list GET

Table 6.527. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	<code>Integer</code>	In	Sets the maximum number of snapshots to return.
<code>snapshots</code>	<code>Snapshot[]</code>	Out	

6.172.2.1. max

Sets the maximum number of snapshots to return. If not specified all the snapshots are returned.

6.173. SSHPUBLICKEY

Table 6.528. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	
<code>update</code>	

6.173.1. get GET

Table 6.529. Parameters summary

Name	Type	Direction	Summary
key	SshPublicKey	Out	

6.173.2. remove DELETE

Table 6.530. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.173.3. update PUT

Table 6.531. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
key	SshPublicKey	In/Out	

6.174. SSHPUBLICKEYS

Table 6.532. Methods summary

Name	Summary
add	
list	

6.174.1. add POST

Table 6.533. Parameters summary

Name	Type	Direction	Summary
------	------	-----------	---------

Name	Type	Direction	Summary
key	SshPublicKey	In/Out	

6.174.2. list GET

Table 6.534. Parameters summary

Name	Type	Direction	Summary
keys	SshPublicKey[]	Out	
max	Integer	In	Sets the maximum number of keys to return.

6.174.2.1. max

Sets the maximum number of keys to return. If not specified all the keys are returned.

6.175. STATISTIC

Table 6.535. Methods summary

Name	Summary
get	

6.175.1. get GET

Table 6.536. Parameters summary

Name	Type	Direction	Summary
statistic	Statistic	Out	

6.176. STATISTICS

Table 6.537. Methods summary

Name	Summary
list	Retrieves a list of statistics.

6.176.1. list GET

Retrieves a list of statistics.

For example, to retrieve the statistics for virtual machine **123** send a request like this:

```
GET /ovirt-engine/api/vms/123/statistics
```

The result will be like this:

```
<statistics>
  <statistic href="/ovirt-engine/api/vms/123/statistics/456" id="456">
    <name>memory.installed</name>
    <description>Total memory configured</description>
    <kind>gauge</kind>
    <type>integer</type>
    <unit>bytes</unit>
    <values>
      <value>
        <datum>1073741824</datum>
      </value>
    </values>
    <vm href="/ovirt-engine/api/vms/123" id="123"/>
  </statistic>
  ...
</statistics>
```

Just a single part of the statistics can be retrieved by specifying its id at the end of the URI. That means:

```
GET /ovirt-engine/api/vms/123/statistics/456
```

Outputs:

```
<statistic href="/ovirt-engine/api/vms/123/statistics/456" id="456">
  <name>memory.installed</name>
  <description>Total memory configured</description>
  <kind>gauge</kind>
  <type>integer</type>
  <unit>bytes</unit>
  <values>
    <value>
      <datum>1073741824</datum>
    </value>
  </values>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</statistic>
```

Table 6.538. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of statistics to return.
statistics	Statistic[]	Out	

6.176.1.1. max

Sets the maximum number of statistics to return. If not specified all the statistics are returned.

6.177. STEP

A service to manage a step.

Table 6.539. Methods summary

Name	Summary
end	Marks an external step execution as ended.
get	Retrieves a step.

6.177.1. end POST

Marks an external step execution as ended.

For example, to terminate a step with identifier **456** which belongs to a **job** with identifier **123** send the following request:

```
POST /ovirt-engine/api/jobs/123/steps/456/end
```

With the following request body:

```
<action>
  <force>true</force>
  <succeeded>true</succeeded>
</action>
```

Table 6.540. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
force	Boolean	In	Indicates if the step should be forcibly terminated.
succeeded	Boolean	In	Indicates if the step should be marked as successfully finished or as failed.

6.177.1.1. succeeded

Indicates if the step should be marked as successfully finished or as failed.

This parameter is optional, and the default value is **true**.

6.177.2. get GET

Retrieves a step.

```
GET /ovirt-engine/api/jobs/123/steps/456
```

You will receive response in XML like this one:

```
<step href="/ovirt-engine/api/jobs/123/steps/456" id="456">
  <actions>
    <link href="/ovirt-engine/api/jobs/123/steps/456/end" rel="end"/>
  </actions>
  <description>Validating</description>
  <end_time>2016-12-12T23:07:26.627+02:00</end_time>
  <external>>false</external>
  <number>0</number>
  <start_time>2016-12-12T23:07:26.605+02:00</start_time>
  <status>finished</status>
  <type>validating</type>
  <job href="/ovirt-engine/api/jobs/123" id="123"/>
</step>
```

Table 6.541. Parameters summary

Name	Type	Direction	Summary
step	Step	Out	Retrieves the representation of the step.

6.178. STEPS

A service to manage steps.

Table 6.542. Methods summary

Name	Summary
add	Add an external step to an existing job or to an existing step.
list	Retrieves the representation of the steps.

6.178.1. add POST

Add an external step to an existing job or to an existing step.

For example, to add a step to **job** with identifier **123** send the following request:

```
POST /ovirt-engine/api/jobs/123/steps
```

With the following request body:

```
<step>
```

```

<description>Validating</description>
<start_time>2016-12-12T23:07:26.605+02:00</start_time>
<status>started</status>
<type>validating</type>
</step>

```

The response should look like:

```

<step href="/ovirt-engine/api/jobs/123/steps/456" id="456">
  <actions>
    <link href="/ovirt-engine/api/jobs/123/steps/456/end" rel="end"/>
  </actions>
  <description>Validating</description>
  <link href="/ovirt-engine/api/jobs/123/steps/456/statistics"
rel="statistics"/>
  <external>true</external>
  <number>2</number>
  <start_time>2016-12-13T01:06:15.380+02:00</start_time>
  <status>started</status>
  <type>validating</type>
  <job href="/ovirt-engine/api/jobs/123" id="123"/>
</step>

```

Table 6.543. Parameters summary

Name	Type	Direction	Summary
step	Step	In/Out	Step that will be added.

6.178.2. list GET

Retrieves the representation of the steps.

```
GET /ovirt-engine/api/job/123/steps
```

You will receive response in XML like this one:

```

<steps>
  <step href="/ovirt-engine/api/jobs/123/steps/456" id="456">
    <actions>
      <link href="/ovirt-engine/api/jobs/123/steps/456/end" rel="end"/>
    </actions>
    <description>Validating</description>
    <link href="/ovirt-engine/api/jobs/123/steps/456/statistics"
rel="statistics"/>
    <external>true</external>
    <number>2</number>
    <start_time>2016-12-13T01:06:15.380+02:00</start_time>
    <status>started</status>
    <type>validating</type>
    <job href="/ovirt-engine/api/jobs/123" id="123"/>
  </step>

```

```

</step>
...
</steps>

```

Table 6.544. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of steps to return.
steps	Step[]	Out	A representation of steps.

6.178.2.1. max

Sets the maximum number of steps to return. If not specified all the steps are returned.

6.179. STORAGE

Table 6.545. Methods summary

Name	Summary
get	

6.179.1. get GET

Table 6.546. Parameters summary

Name	Type	Direction	Summary
report_status	Boolean	In	Indicates if the status of the LUNs in the storage should be checked.
storage	HostStorage	Out	

6.179.1.1. report_status

Indicates if the status of the LUNs in the storage should be checked. Checking the status of the LUN is an heavy weight operation and this data is not always needed by the user. This parameter will give the option to not perform the status check of the LUNs.

The default is **true** for backward compatibility.

Here an example with the LUN status :

```

<host_storage id="360014051136c20574f743bdbd28177fd">
  <logical_units>
    <logical_unit id="360014051136c20574f743bdbd28177fd">
      <lun_mapping>0</lun_mapping>
      <paths>1</paths>
    </logical_unit>
  </logical_units>
</host_storage>

```

```

    <product_id>lun0</product_id>
    <serial>SLIO-ORG_lun0_1136c205-74f7-43bd-bd28-177fd5ce6993</serial>
    <size>10737418240</size>
    <status>used</status>
    <vendor_id>LIO-ORG</vendor_id>
    <volume_group_id>09Du7I-RahN-ECe1-dZ1w-nh0b-64io-
MNzIBZ</volume_group_id>
  </logical_unit>
</logical_units>
<type>iscsi</type>
<host id="8bb5ade5-e988-4000-8b93-dbfc6717fe50" />
</host_storage>

```

Here an example without the LUN status :

```

<host_storage id="360014051136c20574f743bdbd28177fd">
  <logical_units>
    <logical_unit id="360014051136c20574f743bdbd28177fd">
      <lun_mapping>0</lun_mapping>
      <paths>1</paths>
      <product_id>lun0</product_id>
      <serial>SLIO-ORG_lun0_1136c205-74f7-43bd-bd28-177fd5ce6993</serial>
      <size>10737418240</size>
      <vendor_id>LIO-ORG</vendor_id>
      <volume_group_id>09Du7I-RahN-ECe1-dZ1w-nh0b-64io-
MNzIBZ</volume_group_id>
    </logical_unit>
  </logical_units>
  <type>iscsi</type>
  <host id="8bb5ade5-e988-4000-8b93-dbfc6717fe50" />
</host_storage>

```

6.180. STORAGE DOMAIN

Table 6.547. Methods summary

Name	Summary
get	
isattached	
reduceluns	This operation reduces logical units from the storage domain.
refreshluns	This operation refreshes the LUN size.
remove	Removes the storage domain.
update	Updates a storage domain.
updateovfstore	This operation forces the update of the OVF_STORE of this storage domain.

Name	Summary
------	---------

6.180.1. get GET

Table 6.548. Parameters summary

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
storage_domain	StorageDomain	Out	

6.180.2. isattached POST

Table 6.549. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
host	Host	In	
is_attached	Boolean	Out	

6.180.3. reduceluns POST

This operation reduces logical units from the storage domain.

In order to do so the data stored on the provided logical units will be moved to other logical units of the storage domain and only then they will be reduced from the storage domain.

For example, in order to reduce two logical units from a storage domain send a request like this:

```
POST /ovirt-engine/api/storagedomains/123/reduceluns
```

With a request body like this:

```
<action>
  <logical_units>
    <logical_unit id="1IET_00010001"/>
```

```

    <logical_unit id="1IET_00010002"/>
  </logical_units>
</action>

```

Table 6.550. Parameters summary

Name	Type	Direction	Summary
logical_units	LogicalUnit[]	In	The logical units that needs to be reduced from the storage domain.

6.180.4. refreshluns POST

This operation refreshes the LUN size.

After increasing the size of the underlying LUN on the storage server, the user can refresh the LUN size. This action forces a rescan of the provided LUNs and updates the database with the new size if required.

For example, in order to refresh the size of two LUNs send a request like this:

```
POST /ovirt-engine/api/storagedomains/262b056b-aede-40f1-9666-
b883eff59d40/refreshluns
```

With a request body like this:

```

<action>
  <logical_units>
    <logical_unit id="1IET_00010001"/>
    <logical_unit id="1IET_00010002"/>
  </logical_units>
</action>

```

Table 6.551. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the refresh should be performed asynchronously.
logical_units	LogicalUnit[]	In	The LUNs that need to be refreshed.

6.180.5. remove DELETE

Removes the storage domain.

Without any special parameters, the storage domain is detached from the system and removed from the database. The storage domain can then be imported to the same or different setup, with all the data on it. If the storage isn't accessible the operation will fail.

If the **destroy** parameter is **true** then the operation will always succeed, even if the storage isn't

accessible, the failure is just ignored and the storage domain is removed from the database anyway.

If the **format** parameter is **true** then the actual storage is formatted, and the metadata is removed from the LUN or directory, so it can no longer be imported to the same or a different setup.

Table 6.552. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.
destroy	Boolean	In	Indicates if the operation should succeed, and the storage domain removed from the database, even if the storage isn't accessible.
format	Boolean	In	Indicates if the actual storage should be formatted, removing all the metadata from the underlying LUN or directory: [source] ---- DELETE /ovirt-engine/api/storagedomains/123?format=true ---- This parameter is optional, and the default value is false .
host	String	In	Indicates what host should be used to remove the storage domain.

6.180.5.1. destroy

Indicates if the operation should succeed, and the storage domain removed from the database, even if the storage isn't accessible.

```
DELETE /ovirt-engine/api/storagedomains/123?destroy=true
```

This parameter is optional, and the default value is **false**.

6.180.5.2. host

Indicates what host should be used to remove the storage domain.

This parameter is mandatory, and it can contain the name or the identifier of the host. For example, to use the host named **myhost** to remove the storage domain with identifier **123** send a request like this:

```
DELETE /ovirt-engine/api/storagedomains/123?host=myhost
```

6.180.6. update PUT

Updates a storage domain.

Not all of the [StorageDomain](#)'s attributes are updatable post-creation. Those that can be updated are:

name, **description**, **comment**, **warning_low_space_indicator**, **critical_space_action_blocker** and **wipe_after_delete** (note that changing the **wipe_after_delete** attribute will not change the wipe after delete property of disks that already exist).

To update the **name** and **wipe_after_delete** attributes of a storage domain with an identifier **123**, send a request as follows:

```
PUT /ovirt-engine/api/storagedomains/123
```

With a request body as follows:

```
<storage_domain>
  <name>data2</name>
  <wipe_after_delete>>true</wipe_after_delete>
</storage_domain>
```

Table 6.553. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
storage_domain	StorageDomain	In/Out	

6.180.7. updateovfstore POST

This operation forces the update of the **OVF_STORE** of this storage domain.

The **OVF_STORE** is a disk image that contains the meta-data of virtual machines and disks that reside in the storage domain. This meta-data is used in case the domain is imported or exported to or from a different data center or a different installation.

By default the **OVF_STORE** is updated periodically (set by default to 60 minutes) but users might want to force an update after an important change, or when they believe the **OVF_STORE** is corrupt.

When initiated by the user, **OVF_STORE** update will be performed whether an update is needed or not.

Table 6.554. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the OVF_STORE update should be performed asynchronously.

6.181. STORAGEDOMAINCONTENTDISK

Table 6.555. Methods summary

Name	Summary
<code>get</code>	

6.181.1. get GET

Table 6.556. Parameters summary

Name	Type	Direction	Summary
<code>disk</code>	Disk	Out	
<code>filter</code>	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

6.182. STORAGEDOMAINCONTENTDISKS

Table 6.557. Methods summary

Name	Summary
<code>list</code>	

6.182.1. list GET

Table 6.558. Parameters summary

Name	Type	Direction	Summary
<code>case_sensitive</code>	Boolean	In	Indicates if the search performed using the <code>search</code> parameter should be performed taking case into account.
<code>disks</code>	Disk[]	Out	
<code>max</code>	Integer	In	Sets the maximum number of disks to return.
<code>search</code>	String	In	A query string used to restrict the returned disks.

6.182.1.1. case_sensitive

Indicates if the search performed using the `search` parameter should be performed taking case into account. The default value is `true`, which means that case is taken into account. If you want to search ignoring case set it to `false`.

6.182.1.2. max

Sets the maximum number of disks to return. If not specified all the disks are returned.

6.183. STORAGEDOMAINSERVERCONNECTION

Table 6.559. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	Detaches a storage connection from storage.

6.183.1. get GET

Table 6.560. Parameters summary

Name	Type	Direction	Summary
<code>connection</code>	StorageConnection	Out	

6.183.2. remove DELETE

Detaches a storage connection from storage.

Table 6.561. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the action should be performed asynchronously.

6.184. STORAGEDOMAINSERVERCONNECTIONS

Table 6.562. Methods summary

Name	Summary
<code>add</code>	
<code>list</code>	

6.184.1. add POST

Table 6.563. Parameters summary

Name	Type	Direction	Summary
connection	StorageConnection	In/Out	

6.184.2. list GET

Table 6.564. Parameters summary

Name	Type	Direction	Summary
connections	StorageConnection[]	Out	
max	Integer	In	Sets the maximum number of connections to return.

6.184.2.1. max

Sets the maximum number of connections to return. If not specified all the connections are returned.

6.185. STORAGEDOMAINTEMPLATE

Table 6.565. Methods summary

Name	Summary
get	
import	Action to import a template from an export storage domain.
register	
remove	

6.185.1. get GET

Table 6.566. Parameters summary

Name	Type	Direction	Summary
template	Template	Out	

6.185.2. import POST

Action to import a template from an export storage domain.

For example, to import the template **456** from the storage domain **123** send the following request:

```
POST /ovirt-engine/api/storagedomains/123/templates/456/import
```

With the following request body:

```
<action>
  <storage_domain>
    <name>myexport</name>
  </storage_domain>
  <cluster>
    <name>mycluster</name>
  </cluster>
</action>
```

Table 6.567. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the import should be performed asynchronously.
clone	Boolean	In	Use the optional clone parameter to generate new UUIDs for the imported template and its entities.
cluster	Cluster	In	
exclusive	Boolean	In	
storage_domain	StorageDomain	In	
template	Template	In	
vm	Vm	In	

6.185.2.1. clone

Use the optional **clone** parameter to generate new UUIDs for the imported template and its entities.

The user might want to import a template with the **clone** parameter set to **false** when importing a template from an export domain, with templates that was exported by a different Red Hat Virtualization environment.

6.185.3. register POST

Table 6.568. Parameters summary

Name	Type	Direction	Summary
allow_partial_import	Boolean	In	Indicates whether a template is allowed to be registered with only some of its disks.
async	Boolean	In	Indicates if the registration should be performed asynchronously.
clone	Boolean	In	
cluster	Cluster	In	
exclusive	Boolean	In	
template	Template	In	

6.185.3.1. allow_partial_import

Indicates whether a template is allowed to be registered with only some of its disks.

If this flag is **true**, the engine will not fail in the validation process if an image is not found, but instead it will allow the template to be registered without the missing disks. This is mainly used during registration of a template when some of the storage domains are not available. The default value is **false**.

6.185.4. remove DELETE

Table 6.569. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.186. STORAGEDOMAINTEMPLATES

Table 6.570. Methods summary

Name	Summary
list	

6.186.1. list GET

Table 6.571. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of templates to return.
templates	Template[]	Out	
unregistered	Boolean	In	Indicates whether to retrieve a list of registered or unregistered templates which contain disks on the storage domain.

6.186.1.1. max

Sets the maximum number of templates to return. If not specified all the templates are returned.

6.186.1.2. unregistered

Indicates whether to retrieve a list of registered or unregistered templates which contain disks on the storage domain. To get a list of unregistered templates the call should indicate the unregistered flag. For example, to get a list of unregistered templates the REST API call should look like this:

```
GET /ovirt-engine/api/storagedomains/123/templates?unregistered=true
```

The default value of the unregistered flag is **false**. The request only apply to storage domains that are attached.

6.187. STORAGEDOMAINVM

Table 6.572. Methods summary

Name	Summary
get	
import	Imports a virtual machine from an export storage domain.
register	
remove	Deletes a virtual machine from an export storage domain.

6.187.1. get GET

Table 6.573. Parameters summary

Name	Type	Direction	Summary
vm	Vm	Out	

6.187.2. import POST

Imports a virtual machine from an export storage domain.

For example, send a request like this:

```
POST /ovirt-engine/api/storagedomains/123/vms/456/import
```

With a request body like this:

```
<action>
  <storage_domain>
    <name>mydata</name>
  </storage_domain>
  <cluster>
    <name>mycluster</name>
  </cluster>
</action>
```

To import a virtual machine as a new entity add the **clone** parameter:

```
<action>
  <storage_domain>
    <name>mydata</name>
  </storage_domain>
  <cluster>
    <name>mycluster</name>
  </cluster>
  <clone>true</clone>
  <vm>
    <name>myvm</name>
  </vm>
</action>
```

Include an optional **disks** parameter to choose which disks to import. For example, to import the disks of the template that have the identifiers **123** and **456** send the following request body:

```
<action>
  <cluster>
    <name>mycluster</name>
  </cluster>
  <vm>
    <name>myvm</name>
  </vm>
  <disks>
    <disk id="123"/>
    <disk id="456"/>
  </disks>
</action>
```

Table 6.574. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the import should be performed asynchronously.
clone	Boolean	In	Indicates if the identifiers of the imported virtual machine should be regenerated.
cluster	Cluster	In	
collapse_snapshots	Boolean	In	Indicates of the snapshots of the virtual machine that is imported should be collapsed, so that the result will be a virtual machine without snapshots.
storage_domain	StorageDomain	In	
vm	Vm	In	

6.187.2.1. clone

Indicates if the identifiers of the imported virtual machine should be regenerated.

By default when a virtual machine is imported the identifiers are preserved. This means that the same virtual machine can't be imported multiple times, as that identifiers needs to be unique. To allow importing the same machine multiple times set this parameter to **true**, as the default is **false**.

6.187.2.2. collapse_snapshots

Indicates of the snapshots of the virtual machine that is imported should be collapsed, so that the result will be a virtual machine without snapshots.

This parameter is optional, and if it isn't explicitly specified the default value is **false**.

6.187.3. register POST

Table 6.575. Parameters summary

Name	Type	Direction	Summary
allow_partial_import	Boolean	In	Indicates whether a virtual machine is allowed to be registered with only some of its disks.
async	Boolean	In	Indicates if the registration should be performed asynchronously.
clone	Boolean	In	

Name	Type	Direction	Summary
cluster	Cluster	In	
reassign_bad_macs	Boolean	In	Indicates if the problematic MAC addresses should be re-assigned during the import process by the engine.
vm	Vm	In	
vnic_profile_mappings	VnicProfileMapping[]	In	Mapping rules for virtual NIC profiles that will be applied during the import process.

6.187.3.1. allow_partial_import

Indicates whether a virtual machine is allowed to be registered with only some of its disks.

If this flag is **true**, the engine will not fail in the validation process if an image is not found, but instead it will allow the virtual machine to be registered without the missing disks. This is mainly used during registration of a virtual machine when some of the storage domains are not available. The default value is **false**.

6.187.3.2. reassign_bad_macs

Indicates if the problematic MAC addresses should be re-assigned during the import process by the engine.

A MAC address would be considered as a problematic one if one of the following is true:

- It conflicts with a MAC address that is already allocated to a virtual machine in the target environment.
- It's out of the range of the target MAC address pool.

6.187.4. remove DELETE

Deletes a virtual machine from an export storage domain.

For example, to delete the virtual machine **456** from the storage domain **123**, send a request like this:

```
DELETE /ovirt-engine/api/storagedomains/123/vms/456
```

Table 6.576. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.188. STORAGEDOMAINVMDISKATTACHMENT

Returns the details of the disks attached to a virtual machine in the export domain.

Table 6.577. Methods summary

Name	Summary
<code>get</code>	Returns the details of the attachment with all its properties and a link to the disk.

6.188.1. get GET

Returns the details of the attachment with all its properties and a link to the disk.

Table 6.578. Parameters summary

Name	Type	Direction	Summary
<code>attachment</code>	DiskAttachment	Out	The disk attachment.

6.189. STORAGEDOMAINVMDISKATTACHMENTS

Returns the details of a disk attached to a virtual machine in the export domain.

Table 6.579. Methods summary

Name	Summary
<code>list</code>	List the disks that are attached to the virtual machine.

6.189.1. list GET

List the disks that are attached to the virtual machine.

Table 6.580. Parameters summary

Name	Type	Direction	Summary
<code>attachments</code>	DiskAttachment[]	Out	

6.190. STORAGEDOMAINVMS

Lists the virtual machines of an export storage domain.

For example, to retrieve the virtual machines that are available in the storage domain with identifier **123** send the following request:

```
GET /ovirt-engine/api/storagedomains/123/vms
```

This will return the following response body:

```
<vms>
  <vm id="456" href="/api/storagedomains/123/vms/456">
    <name>vm1</name>
    ...
    <storage_domain id="123" href="/api/storagedomains/123"/>
    <actions>
      <link rel="import" href="/api/storagedomains/123/vms/456/import"/>
    </actions>
  </vm>
</vms>
```

Virtual machines and templates in these collections have a similar representation to their counterparts in the top-level [Vm](#) and [Template](#) collections, except they also contain a [StorageDomain](#) reference and an [import](#) action.

Table 6.581. Methods summary

Name	Summary
<code>list</code>	

6.190.1. list GET

Table 6.582. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of virtual machines to return.
<code>unregistered</code>	Boolean	In	Indicates whether to retrieve a list of registered or unregistered virtual machines which contain disks on the storage domain.
<code>vm</code>	Vm[]	Out	

6.190.1.1. max

Sets the maximum number of virtual machines to return. If not specified all the virtual machines are returned.

6.190.1.2. unregistered

Indicates whether to retrieve a list of registered or unregistered virtual machines which contain disks on the storage domain. To get a list of unregistered virtual machines the call should indicate the unregistered flag. For example, to get a list of unregistered virtual machines the REST API call should look like this:

```
GET /ovirt-engine/api/storagedomains/123/vms?unregistered=true
```

The default value of the `unregistered` flag is **false**. The request only apply to storage domains that are attached.

6.191. STORAGEDOMAINS

Table 6.583. Methods summary

Name	Summary
add	Adds a new storage domain.
list	

6.191.1. add POST

Adds a new storage domain.

Creation of a new [StorageDomain](#) requires the **name**, **type**, **host** and **storage** attributes. Identify the **host** attribute with the **id** or **name** attributes. In oVirt 3.6 and later you can enable the wipe after delete option by default on the storage domain. To configure this, specify **wipe_after_delete** in the POST request. This option can be edited after the domain is created, but doing so will not change the wipe after delete property of disks that already exist.

To add a new storage domain with specified **name**, **type**, **storage.type**, **storage.address** and **storage.path** and by using a host with an id **123**, send a request as follows:

```
POST /ovirt-engine/api/storagedomains
```

With a request body as follows:

```
<storage_domain>
  <name>mydata</name>
  <type>data</type>
  <storage>
    <type>nfs</type>
    <address>my nfs.example.com</address>
    <path>/exports/mydata</path>
  </storage>
</host>
  <name>myhost</name>
</host>
</storage_domain>
```

To create a new NFS ISO storage domain send a request like this:

```
<storage_domain>
  <name>myisos</name>
  <type>iso</type>
  <storage>
    <type>nfs</type>
```



```

    <address>my nfs.example.com</address>
    <path>/export/myisos</path>
  </storage>
</host>
  <name>myhost</name>
</host>
</storage_domain>

```

To create a new iSCSI storage domain send a request like this:

```

<storage_domain>
  <name>myiscsi</name>
  <type>data</type>
  <storage>
    <type>iscsi</type>
    <logical_units>
      <logical_unit id="3600144f09dbd050000004eedbd340001"/>
      <logical_unit id="3600144f09dbd050000004eedbd340002"/>
    </logical_units>
  </storage>
  <host>
    <name>myhost</name>
  </host>
</storage_domain>

```

Table 6.584. Parameters summary

Name	Type	Direction	Summary
storage_domain	StorageDomain	In/Out	

6.191.2. list GET

Table 6.585. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
max	Integer	In	Sets the maximum number of storage domains to return.
search	String	In	A query string used to restrict the returned storage domains.

Name	Type	Direction	Summary
storage_domains	StorageDomain[]	Out	

6.191.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.191.2.2. max

Sets the maximum number of storage domains to return. If not specified all the storage domains are returned.

6.192. STORAGE SERVER CONNECTION

Table 6.586. Methods summary

Name	Summary
get	
remove	Removes a storage connection.
update	Updates the storage connection.

6.192.1. get GET

Table 6.587. Parameters summary

Name	Type	Direction	Summary
connection	StorageConnection	Out	

6.192.2. remove DELETE

Removes a storage connection.

A storage connection can only be deleted if neither storage domain nor LUN disks reference it. The host name or id is optional; providing it disconnects (unmounts) the connection from that host.

Table 6.588. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.
host	String	In	The name or identifier of the host from which the connection would be unmounted (disconnected).

6.192.2.1. host

The name or identifier of the host from which the connection would be unmounted (disconnected). If not provided, no host will be disconnected.

For example, to use the host with identifier **456** to delete the storage connection with identifier **123** send a request like this:

```
DELETE /ovirt-engine/api/storageconnections/123?host=456
```

6.192.3. update PUT

Updates the storage connection.

For example, to change the address of the storage server send a request like this:

```
PUT /ovirt-engine/api/storageconnections/123
```

With a request body like this:

```
<storage_connection>
  <address>mynewnfs.example.com</address>
  <host>
    <name>myhost</name>
  </host>
</storage_connection>
```

Table 6.589. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
connection	StorageConnection	In/Out	
force	Boolean	In	Indicates if the operation should succeed regardless to the relevant storage domain's status (i.

6.192.3.1. force

Indicates if the operation should succeed regardless to the relevant storage domain's status (i.e. updating is also applicable when storage domain's status is not maintenance).

This parameter is optional, and the default value is **false**.

6.193. STORAGE SERVER CONNECTION EXTENSION

Table 6.590. Methods summary

Name	Summary
get	
remove	
update	Update a storage server connection extension for the given host.

6.193.1. get GET

Table 6.591. Parameters summary

Name	Type	Direction	Summary
extension	StorageConnectionExtension	Out	

6.193.2. remove DELETE

Table 6.592. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.193.3. update PUT

Update a storage server connection extension for the given host.

To update the storage connection **456** of host **123** send a request like this:

```
PUT /ovirt-engine/api/hosts/123/storageconnectionextensions/456
```

With a request body like this:

```
<storage_connection_extension>
  <target>iqn.2016-01.com.example:mytarget</target>
  <username>myuser</username>
```

```
<password>mypassword</password>
</storage_connection_extension>
```

Table 6.593. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
extension	StorageConnectionExtension	In/Out	

6.194. STORAGE SERVER CONNECTION EXTENSIONS

Table 6.594. Methods summary

Name	Summary
add	Creates a new storage server connection extension for the given host.
list	

6.194.1. add POST

Creates a new storage server connection extension for the given host.

The extension lets the user define credentials for an iSCSI target for a specific host. For example to use **myuser** and **mypassword** as the credentials when connecting to the iSCSI target from host **123** send a request like this:

```
POST /ovirt-engine/api/hosts/123/storageconnectionextensions
```

With a request body like this:

```
<storage_connection_extension>
  <target>iqn.2016-01.com.example:mytarget</target>
  <username>myuser</username>
  <password>mypassword</password>
</storage_connection_extension>
```

Table 6.595. Parameters summary

Name	Type	Direction	Summary
extension	StorageConnectionExtension	In/Out	

6.194.2. list GET

Table 6.596. Parameters summary

Name	Type	Direction	Summary
extension s	StorageConn ectionExtensi on[]	Out	
max	Integer	In	Sets the maximum number of extensions to return.

6.194.2.1. max

Sets the maximum number of extensions to return. If not specified all the extensions are returned.

6.195. STORAGESERVERCONNECTIONS

Table 6.597. Methods summary

Name	Summary
add	Creates a new storage connection.
list	

6.195.1. add POST

Creates a new storage connection.

For example, to create a new storage connection for the NFS server `my nfs . example . com` and NFS share `/export/mydata` send a request like this:

```
POST /ovirt-engine/api/storageconnections
```

With a request body like this:

```
<storage_connection>
  <type>nfs</type>
  <address>my nfs . example . com</address>
  <path>/export/mydata</path>
  <host>
    <name>myhost</name>
  </host>
</storage_connection>
```

Table 6.598. Parameters summary

Name	Type	Direction	Summary
connection	StorageConnection	In/Out	

6.195.2. list GET

Table 6.599. Parameters summary

Name	Type	Direction	Summary
connections	StorageConnection[]	Out	
max	Integer	In	Sets the maximum number of connections to return.

6.195.2.1. max

Sets the maximum number of connections to return. If not specified all the connections are returned.

6.196. SYSTEM

Table 6.600. Methods summary

Name	Summary
get	Returns basic information describing the API, like the product name, the version number and a summary of the number of relevant objects.
reloadconfigurations	

6.196.1. get GET

Returns basic information describing the API, like the product name, the version number and a summary of the number of relevant objects.

```
GET /ovirt-engine/api
```

We get following response:

```
<api>
  <link rel="capabilities" href="/api/capabilities"/>
  <link rel="clusters" href="/api/clusters"/>
  <link rel="clusters/search" href="/api/clusters?search={query}"/>
  <link rel="datacenters" href="/api/datacenters"/>
  <link rel="datacenters/search" href="/api/datacenters?search={query}"/>
```

```

<link rel="events" href="/api/events"/>
<link rel="events/search" href="/api/events?search={query}"/>
<link rel="hosts" href="/api/hosts"/>
<link rel="hosts/search" href="/api/hosts?search={query}"/>
<link rel="networks" href="/api/networks"/>
<link rel="roles" href="/api/roles"/>
<link rel="storagedomains" href="/api/storagedomains"/>
<link rel="storagedomains/search" href="/api/storagedomains?search={query}"/>
<link rel="tags" href="/api/tags"/>
<link rel="templates" href="/api/templates"/>
<link rel="templates/search" href="/api/templates?search={query}"/>
<link rel="users" href="/api/users"/>
<link rel="groups" href="/api/groups"/>
<link rel="domains" href="/api/domains"/>
<link rel="vmpools" href="/api/vmpools"/>
<link rel="vmpools/search" href="/api/vmpools?search={query}"/>
<link rel="vms" href="/api/vms"/>
<link rel="vms/search" href="/api/vms?search={query}"/>
<product_info>
  <name>oVirt Engine</name>
  <vendor>ovirt.org</vendor>
  <version>
    <build>4</build>
    <full_version>4.0.4</full_version>
    <major>4</major>
    <minor>0</minor>
    <revision>0</revision>
  </version>
</product_info>
<special_objects>
  <blank_template href="/ovirt-engine/api/templates/00000000-0000-0000-0000-000000000000" id="00000000-0000-0000-0000-000000000000"/>
  <root_tag href="/ovirt-engine/api/tags/00000000-0000-0000-0000-000000000000" id="00000000-0000-0000-0000-000000000000"/>
</special_objects>
<summary>
  <hosts>
    <active>0</active>
    <total>0</total>
  </hosts>
  <storage_domains>
    <active>0</active>
    <total>1</total>
  </storage_domains>
  <users>
    <active>1</active>
    <total>1</total>
  </users>
  <vms>
    <active>0</active>
    <total>0</total>
  </vms>
</summary>
<time>2016-09-14T12:00:48.132+02:00</time>
</api>

```


The entry point provides a user with links to the collections in a virtualization environment. The **rel** attribute of each collection link provides a reference point for each link.

The entry point also contains other data such as **product_info**, **special_objects** and **summary**.

Table 6.601. Parameters summary

Name	Type	Direction	Summary
api	Api	Out	

6.196.2. reloadconfigurations POST

Table 6.602. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the reload should be performed asynchronously.

6.197. SYSTEMPERMISSIONS

This service doesn't add any new methods, it is just a placeholder for the annotation that specifies the path of the resource that manages the permissions assigned to the system object.

Table 6.603. Methods summary

Name	Summary
add	Assign a new permission to a user or group for specific entity.
list	List all the permissions of the specific entity.

6.197.1. add POST

Assign a new permission to a user or group for specific entity.

For example, to assign the **UserVmManager** role to the virtual machine with id **123** to the user with id **456** send a request like this:

```
POST /ovirt-engine/api/vms/123/permissions
```

With a request body like this:

```
<permission>
  <role>
    <name>UserVmManager</name>
  </role>
  <user id="456"/>
</permission>
```

To assign the **SuperUser** role to the system to the user with id **456** send a request like this:

```
POST /ovirt-engine/api/permissions
```

With a request body like this:

```
<permission>
  <role>
    <name>SuperUser</name>
  </role>
  <user id="456"/>
</permission>
```

If you want to assign permission to the group instead of the user please replace the **user** element with the **group** element with proper **id** of the group. For example to assign the **UserRole** role to the cluster with id **123** to the group with id **789** send a request like this:

```
POST /ovirt-engine/api/clusters/123/permissions
```

With a request body like this:

```
<permission>
  <role>
    <name>UserRole</name>
  </role>
  <group id="789"/>
</permission>
```

Table 6.604. Parameters summary

Name	Type	Direction	Summary
permission	Permission	In/Out	The permission.

6.197.2. list GET

List all the permissions of the specific entity.

For example to list all the permissions of the cluster with id **123** send a request like this:

```
GET /ovirt-engine/api/clusters/123/permissions
```

```
<permissions>
  <permission id="456">
    <cluster id="123"/>
    <role id="789"/>
    <user id="451"/>
  </permission>
  <permission id="654">
    <cluster id="123"/>
    <role id="789"/>
  </permission>
</permissions>
```

```

    <group id="127"/>
  </permission>
</permissions>

```

Table 6.605. Parameters summary

Name	Type	Direction	Summary
permissions	Permission[]	Out	The list of permissions.

6.198. TAG

A service to manage a specific tag in the system.

Table 6.606. Methods summary

Name	Summary
get	Gets the information about the tag.
remove	Removes the tag from the system.
update	Updates the tag entity.

6.198.1. get GET

Gets the information about the tag.

For example to retrieve the information about the tag with the id **123** send a request like this:

```
GET /ovirt-engine/api/tags/123
```

```

<tag href="/ovirt-engine/api/tags/123" id="123">
  <name>root</name>
  <description>root</description>
</tag>

```

Table 6.607. Parameters summary

Name	Type	Direction	Summary
tag	Tag	Out	The tag.

6.198.2. remove DELETE

Removes the tag from the system.

For example to remove the tag with id **123** send a request like this:

```
DELETE /ovirt-engine/api/tags/123
```

Table 6.608. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.198.3. update PUT

Updates the tag entity.

For example to update parent tag to tag with id **456** of the tag with id **123** send a request like this:

```
PUT /ovirt-engine/api/tags/123
```

With request body like:

```
<tag>
  <parent id="456"/>
</tag>
```

You may also specify a tag name instead of id. For example to update parent tag to tag with name **mytag** of the tag with id **123** send a request like this:

```
<tag>
  <parent>
    <name>mytag</name>
  </parent>
</tag>
```

Table 6.609. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
tag	Tag	In/Out	The updated tag.

6.199. TAGS

Represents a service to manage collection of the tags in the system.

Table 6.610. Methods summary

Name	Summary
add	Add a new tag to the system.
list	List the tags in the system.

6.199.1. add POST

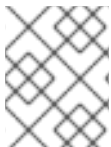
Add a new tag to the system.

For example, to add new tag with name **mytag** to the system send a request like this:

```
POST /ovirt-engine/api/tags
```

With a request body like this:

```
<tag>
  <name>mytag</name>
</tag>
```



NOTE

The root tag is a special pseudo-tag assumed as the default parent tag if no parent tag is specified. The root tag cannot be deleted nor assigned a parent tag.

To create new tag with specific parent tag send a request body like this:

```
<tag>
  <name>mytag</name>
  <parent>
    <name>myparenttag</name>
  </parent>
</tag>
```

Table 6.611. Parameters summary

Name	Type	Direction	Summary
tag	Tag	In/Out	The added tag.

6.199.2. list GET

List the tags in the system.

For example to list the full hierarchy of the tags in the system send a request like this:

```
GET /ovirt-engine/api/tags
```

```
<tags>
```

```

<tag href="/ovirt-engine/api/tags/222" id="222">
  <name>root2</name>
  <description>root2</description>
  <parent href="/ovirt-engine/api/tags/111" id="111"/>
</tag>
<tag href="/ovirt-engine/api/tags/333" id="333">
  <name>root3</name>
  <description>root3</description>
  <parent href="/ovirt-engine/api/tags/222" id="222"/>
</tag>
<tag href="/ovirt-engine/api/tags/111" id="111">
  <name>root</name>
  <description>root</description>
</tag>
</tags>

```

In the previous XML output you can see the following hierarchy of the tags:

```

root:      (id: 111)
- root2   (id: 222)
- root3   (id: 333)

```

Table 6.612. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of tags to return.
tags	Tag[]	Out	List of all tags in the system.

6.199.2.1. max

Sets the maximum number of tags to return. If not specified all the tags are returned.

6.200. TEMPLATE

Manages the virtual machine template and template versions.

Table 6.613. Methods summary

Name	Summary
export	Exports a template to the data center export domain.
get	Returns the information about this template or template version.
remove	Removes a virtual machine template.
update	Updates the template.

6.200.1. export POST

Exports a template to the data center export domain.

For example, the operation can be facilitated using the following request:

```
POST /ovirt-engine/api/templates/123/export
```

With a request body like this:

```
<action>
  <storage_domain id="456"/>
  <exclusive>true</exclusive>
</action>
```

Table 6.614. Parameters summary

Name	Type	Direction	Summary
exclusive	Boolean	In	Indicates if the existing templates with the same name should be overwritten.
storage_domain	StorageDomain	In	Specifies the destination export storage domain.

6.200.1.1. exclusive

Indicates if the existing templates with the same name should be overwritten.

The export action reports a failed action if a template of the same name exists in the destination domain. Set this parameter to **true** to change this behavior and overwrite any existing template.

6.200.2. get GET

Returns the information about this template or template version.

Table 6.615. Parameters summary

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
template	Template	Out	The information about the template or template version.

6.200.3. remove DELETE

Removes a virtual machine template.

```
DELETE /ovirt-engine/api/templates/123
```

Table 6.616. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.200.4. update PUT

Updates the template.

The **name**, **description**, **type**, **memory**, **cpu**, **topology**, **os**, **high_availability**, **display**, **stateless**, **usb** and **timezone** elements can be updated after a template has been created.

For example, to update a template to so that it has 1 GiB of memory send a request like this:

```
PUT /ovirt-engine/api/templates/123
```

With the following request body:

```
<template>
  <memory>1073741824</memory>
</template>
```

The **version_name** name attribute is the only one that can be updated within the **version** attribute used for template versions:

```
<template>
  <version>
    <version_name>mytemplate_2</version_name>
  </version>
</template>
```

Table 6.617. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
template	Template	In/Out	

6.201. TEMPLATECDROM

A service managing a CD-ROM device on templates.

Table 6.618. Methods summary

Name	Summary
get	Returns the information about this CD-ROM device.

6.201.1. get GET

Returns the information about this CD-ROM device.

For example, to get information about the CD-ROM device of template **123** send a request like:

```
GET /ovirt-engine/api/templates/123/cdroms/
```

Table 6.619. Parameters summary

Name	Type	Direction	Summary
cdrom	Cdrom	Out	The information about the CD-ROM device.

6.201.1.1. cdrom

The information about the CD-ROM device.

The information consists of **cdrom** attribute containing reference to the CD-ROM device, the template, and optionally the inserted disk.

If there is a disk inserted then the **file** attribute will contain a reference to the ISO image:

```
<cdrom href="..." id="00000000-0000-0000-0000-000000000000">
  <template href="/ovirt-engine/api/templates/123" id="123"/>
  <file id="mycd.iso"/>
</cdrom>
```

If there is no disk inserted then the **file** attribute won't be reported:

```
<cdrom href="..." id="00000000-0000-0000-0000-000000000000">
  <template href="/ovirt-engine/api/templates/123" id="123"/>
</cdrom>
```

6.202. TEMPLATECDROMS

Lists the CD-ROM devices of a template.

Table 6.620. Methods summary

Name	Summary
list	

6.202.1. list GET

Table 6.621. Parameters summary

Name	Type	Direction	Summary
cdroms	Cdrom[]	Out	The list of CD-ROM devices of the template.
max	Integer	In	Sets the maximum number of CD-ROMs to return.

6.202.1.1. max

Sets the maximum number of CD-ROMs to return. If not specified all the CD-ROMs are returned.

6.203. TEMPLATEDISK

Table 6.622. Methods summary

Name	Summary
copy	
export	
get	
remove	

6.203.1. copy POST

Table 6.623. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the copy should be performed asynchronously.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

6.203.2. export POST

Table 6.624. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the export should be performed asynchronously.

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

6.203.3. get GET

Table 6.625. Parameters summary

Name	Type	Direction	Summary
disk	Disk	Out	

6.203.4. remove DELETE

Table 6.626. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.204. TEMPLATEDISKATTACHMENT

This service manages the attachment of a disk to a template.

Table 6.627. Methods summary

Name	Summary
get	Returns the details of the attachment.
remove	Removes the disk from the template.

6.204.1. get GET

Returns the details of the attachment.

Table 6.628. Parameters summary

Name	Type	Direction	Summary
attachmen t	DiskAttachm ent	Out	

6.204.2. remove DELETE

Removes the disk from the template. The disk will only be removed if there are other existing copies of the disk on other storage domains.

A storage domain has to be specified to determine which of the copies should be removed (template disks can have copies on multiple storage domains).

```
DELETE /ovirt-
engine/api/templates/{template:id}/diskattachments/{attachment:id}?
storage_domain=072fbaa1-08f3-4a40-9f34-a5ca22dd1d74
```

Table 6.629. Parameters summary

Name	Type	Direction	Summary
force	Boolean	In	
storage_domain	String	In	Specifies the identifier of the storage domain the image to be removed resides on.

6.205. TEMPLATEDISKATTACHMENTS

This service manages the set of disks attached to a template. Each attached disk is represented by a [DiskAttachment](#).

Table 6.630. Methods summary

Name	Summary
list	List the disks that are attached to the template.

6.205.1. list GET

List the disks that are attached to the template.

Table 6.631. Parameters summary

Name	Type	Direction	Summary
attachments	DiskAttachment[]	Out	

6.206. TEMPLATEDISKS

Table 6.632. Methods summary

Name	Summary
list	

6.206.1. list GET

Table 6.633. Parameters summary

Name	Type	Direction	Summary
disks	Disk[]	Out	
max	Integer	In	Sets the maximum number of disks to return.

6.206.1.1. max

Sets the maximum number of disks to return. If not specified all the disks are returned.

6.207. TEMPLATEGRAPHICSCONSOLE

Table 6.634. Methods summary

Name	Summary
get	Gets graphics console configuration of the template.
remove	Remove the graphics console from the template.

6.207.1. get GET

Gets graphics console configuration of the template.

Table 6.635. Parameters summary

Name	Type	Direction	Summary
console	GraphicsConsole	Out	The information about the graphics console of the template.

6.207.2. remove DELETE

Remove the graphics console from the template.

Table 6.636. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.208. TEMPLATEGRAPHICSCONSOLES

Table 6.637. Methods summary

Name	Summary
add	Add new graphics console to the template.
list	Lists all the configured graphics consoles of the template.

6.208.1. add POST

Add new graphics console to the template.

Table 6.638. Parameters summary

Name	Type	Direction	Summary
console	GraphicsConsole	In/Out	

6.208.2. list GET

Lists all the configured graphics consoles of the template.

Table 6.639. Parameters summary

Name	Type	Direction	Summary
consoles	GraphicsConsole[]	Out	The list of graphics consoles of the template.
max	Integer	In	Sets the maximum number of consoles to return.

6.208.2.1. max

Sets the maximum number of consoles to return. If not specified all the consoles are returned.

6.209. TEMPLATENIC**Table 6.640. Methods summary**

Name	Summary
get	
remove	
update	

6.209.1. get GET

Table 6.641. Parameters summary

Name	Type	Direction	Summary
nic	Nic	Out	

6.209.2. remove DELETE

Table 6.642. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.209.3. update PUT

Table 6.643. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
nic	Nic	In/Out	

6.210. TEMPLATENICS

Table 6.644. Methods summary

Name	Summary
add	
list	

6.210.1. add POST

Table 6.645. Parameters summary

Name	Type	Direction	Summary
nic	Nic	In/Out	

6.210.2. list GET

Table 6.646. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of NICs to return.
nics	Nic[]	Out	

6.210.2.1. max

Sets the maximum number of NICs to return. If not specified all the NICs are returned.

6.211. TEMPLATEWATCHDOG

Table 6.647. Methods summary

Name	Summary
get	
remove	
update	

6.211.1. get GET

Table 6.648. Parameters summary

Name	Type	Direction	Summary
watchdog	Watchdog	Out	

6.211.2. remove DELETE

Table 6.649. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.211.3. update PUT

Table 6.650. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
watchdog	Watchdog	In/Out	

6.212. TEMPLATEWATCHDOGS

Table 6.651. Methods summary

Name	Summary
add	
list	

6.212.1. add POST

Table 6.652. Parameters summary

Name	Type	Direction	Summary
watchdog	Watchdog	In/Out	

6.212.2. list GET

Table 6.653. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of watchdogs to return.
watchdogs	Watchdog[]	Out	

6.212.2.1. max

Sets the maximum number of watchdogs to return. If not specified all the watchdogs are returned.

6.213. TEMPLATES

This service manages the virtual machine templates available in the system.

Table 6.654. Methods summary

Name	Summary
add	Creates a new template.
list	Returns the list of virtual machine templates.

6.213.1. add POST

Creates a new template.

This requires the **name** and **vm** elements. Identify the virtual machine with the **id** **name** attributes.

```
POST /ovirt-engine/api/templates
```

With a request body like this:

```
<template>
  <name>mytemplate</name>
  <vm id="123"/>
</template>
```

The template can be created as a sub version of an existing template. This requires the **name** and **vm** attributes for the new template, and the **base_template** and **version_name** attributes for the new template version. The **base_template** and **version_name** attributes must be specified within a **version** section enclosed in the **template** section. Identify the virtual machine with the **id** or **name** attributes.

```
<template>
  <name>mytemplate</name>
  <vm id="123"/>
  <version>
    <base_template id="456"/>
    <version_name>mytemplate_001</version_name>
  </version>
</template>
```

Table 6.655. Parameters summary

Name	Type	Direction	Summary
clone_permissions	Boolean	In	Specifies if the permissions of the virtual machine should be copied to the template.
seal	Boolean	In	Seal the template.
template	Template	In/Out	The information about the template or template version.

6.213.1.1. clone_permissions

Specifies if the permissions of the virtual machine should be copied to the template.

If this optional parameter is provided, and its value is **true** then the permissions of the virtual machine (only the direct ones, not the inherited ones) will be copied to the created template. For example, to create a template from the **myvm** virtual machine copying its permissions, send a request like this:

```
POST /ovirt-engine/api/templates?clone_permissions=true
```

With a request body like this:

```
<template>
  <name>mytemplate</name>
  <vm>
    <name>myvm</name>
  </vm>
</template>
```

6.213.1.2. seal

Seal the template.

If this optional parameter is provided and its value is **true**, then the template is sealed after creation.

Sealing erases all host-specific configuration from the filesystem: SSH keys, UDEV rules, MAC addresses, system ID, hostname etc., thus making easy to use the template to create multiple virtual machines without manual intervention.

Currently sealing is supported only for Linux OS.

6.213.2. list GET

Returns the list of virtual machine templates.

For example:

```
GET /ovirt-engine/api/templates
```

Will return the list of virtual machines and virtual machine templates.

Table 6.656. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
max	Integer	In	Sets the maximum number of templates to return.

Name	Type	Direction	Summary
search	String	In	A query string used to restrict the returned templates.
templates	Template[]	Out	The list of virtual machine templates.

6.213.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.213.2.2. max

Sets the maximum number of templates to return. If not specified all the templates are returned.

6.214. UNMANAGEDNETWORK

Table 6.657. Methods summary

Name	Summary
get	
remove	

6.214.1. get GET

Table 6.658. Parameters summary

Name	Type	Direction	Summary
network	Unmanaged Network	Out	

6.214.2. remove DELETE

Table 6.659. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.215. UNMANAGEDNETWORKS

Table 6.660. Methods summary

Name	Summary
list	

6.215.1. list GET

Table 6.661. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of networks to return.
networks	Unmanaged Network[]	Out	

6.215.1.1. max

Sets the maximum number of networks to return. If not specified all the networks are returned.

6.216. USER

A service to manage a user in the system. Use this service to either get users details or remove users. In order to add new users please use [Section 6.217, “Users”](#).

Table 6.662. Methods summary

Name	Summary
get	Gets the system user information.
remove	Removes the system user.

6.216.1. get GET

Gets the system user information.

Usage:

```
GET /ovirt-engine/api/users/1234
```

Will return the user information:

```
<user href="/ovirt-engine/api/users/1234" id="1234">
  <name>admin</name>
  <link href="/ovirt-engine/api/users/1234/sshpublickeys"
rel="sshpublickeys"/>
  <link href="/ovirt-engine/api/users/1234/roles" rel="roles"/>
  <link href="/ovirt-engine/api/users/1234/permissions"
rel="permissions"/>
```

```

<link href="/ovirt-engine/api/users/1234/tags" rel="tags"/>
<department></department>
<domain_entry_id>23456</domain_entry_id>
<email>user1@domain.com</email>
<last_name>Lastname</last_name>
<namespace>*</namespace>
<principal>user1</principal>
<user_name>user1@domain-authz</user_name>
<domain href="/ovirt-engine/api/domains/45678" id="45678">
  <name>domain-authz</name>
</domain>
</user>

```

Table 6.663. Parameters summary

Name	Type	Direction	Summary
user	User	Out	The system user.

6.216.2. remove DELETE

Removes the system user.

Usage:

```
DELETE /ovirt-engine/api/users/1234
```

Table 6.664. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.217. USERS

A service to manage the users in the system.

Table 6.665. Methods summary

Name	Summary
add	Add user from a directory service.
list	List all the users in the system.

6.217.1. add POST

Add user from a directory service.

For example, to add the **myuser** user from the **myextension-authz** authorization provider send a request like this:

```
POST /ovirt-engine/api/users
```

With a request body like this:

```
<user>
  <user_name>myuser@myextension-authz</user_name>
  <domain>
    <name>myextension-authz</name>
  </domain>
</user>
```

In case you are working with Active Directory you have to pass user principal name (UPN) as **username**, followed by authorization provider name. Due to [bug 1147900](#) you need to provide also **principal** parameter set to UPN of the user.

For example, to add the user with UPN **myuser@mysubdomain.mydomain.com** from the **myextension-authz** authorization provider send a request body like this:

```
<user>
  <principal>myuser@mysubdomain.mydomain.com</principal>
  <user_name>myuser@mysubdomain.mydomain.com@myextension-authz</user_name>
  <domain>
    <name>myextension-authz</name>
  </domain>
</user>
```

Table 6.666. Parameters summary

Name	Type	Direction	Summary
user	User	In/Out	

6.217.2. list GET

List all the users in the system.

Usage:

```
GET /ovirt-engine/api/users
```

Will return the list of users:

```
<users>
  <user href="/ovirt-engine/api/users/1234" id="1234">
    <name>admin</name>
    <link href="/ovirt-engine/api/users/1234/sshpublickeys"
rel="sshpublickeys"/>
    <link href="/ovirt-engine/api/users/1234/roles" rel="roles"/>
    <link href="/ovirt-engine/api/users/1234/permissions"
```

```

rel="permissions"/>
  <link href="/ovirt-engine/api/users/1234/tags" rel="tags" />
  <domain_entry_id>23456</domain_entry_id>
  <namespace>*</namespace>
  <principal>user1</principal>
  <user_name>user1@domain-authz</user_name>
  <domain href="/ovirt-engine/api/domains/45678" id="45678">
    <name>domain-authz</name>
  </domain>
</user>
</users>

```

Table 6.667. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
max	Integer	In	Sets the maximum number of users to return.
search	String	In	A query string used to restrict the returned users.
users	User[]	Out	The list of users.

6.217.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.217.2.2. max

Sets the maximum number of users to return. If not specified all the users are returned.

6.218. VIRTUALFUNCTIONALLOWEDNETWORK

Table 6.668. Methods summary

Name	Summary
get	
remove	

6.218.1. get GET

Table 6.669. Parameters summary

Name	Type	Direction	Summary
<code>network</code>	Network	Out	

6.218.2. remove DELETE

Table 6.670. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the remove should be performed asynchronously.

6.219. VIRTUALFUNCTIONALLOWEDNETWORKS

Table 6.671. Methods summary

Name	Summary
<code>add</code>	
<code>list</code>	

6.219.1. add POST

Table 6.672. Parameters summary

Name	Type	Direction	Summary
<code>network</code>	Network	In/Out	

6.219.2. list GET

Table 6.673. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of networks to return.
<code>networks</code>	Network[]	Out	

6.219.2.1. max

Sets the maximum number of networks to return. If not specified all the networks are returned.

6.220. VM

Table 6.674. Methods summary

Name	Summary
cancelmigration	This operation stops any migration of a virtual machine to another physical host.
clone	
commitsnapshot	Permanently restores the virtual machine to the state of the previewed snapshot.
detach	Detaches a virtual machine from a pool.
export	Exports a virtual machine to an export domain.
freezefilesystems	Freezes virtual machine file systems.
get	Retrieves the description of the virtual machine.
logon	Initiates the automatic user logon to access a virtual machine from an external console.
maintenance	Sets the global maintenance mode on the hosted engine virtual machine.
migrate	Migrates a virtual machine to another physical host.
previewsnapshot	Temporarily restores the virtual machine to the state of a snapshot.
reboot	Sends a reboot request to a virtual machine.
remove	Removes the virtual machine, including the virtual disks attached to it.
reordermacaddresses	
shutdown	This operation sends a shutdown request to a virtual machine.
start	Starts the virtual machine.
stop	This operation forces a virtual machine to power-off.
suspend	This operation saves the virtual machine state to disk and stops it.
thawfilesystems	Thaws virtual machine file systems.

Name	Summary
ticket	Generates a time-sensitive authentication token for accessing a virtual machine's display.
undosnapshot	Restores the virtual machine to the state it had before previewing the snapshot.
update	

6.220.1. cancelmigration POST

This operation stops any migration of a virtual machine to another physical host.

```
POST /ovirt-engine/api/vms/123/cancelmigration
```

The cancel migration action does not take any action specific parameters; therefore, the request body should contain an empty **action**:

```
<action/>
```

Table 6.675. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the migration should cancelled asynchronously.

6.220.2. clone POST

Table 6.676. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the clone should be performed asynchronously.
vm	Vm	In	

6.220.3. commitsnapshot POST

Permanently restores the virtual machine to the state of the previewed snapshot.

See the [preview_snapshot](#) operation for details.

Table 6.677. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the snapshots should be committed asynchronously.

6.220.4. detach POST

Detaches a virtual machine from a pool.

```
POST /ovirt-engine/api/vms/123/detach
```

The detach action does not take any action specific parameters; therefore, the request body should contain an empty **action**:

```
<action/>
```

Table 6.678. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the detach action should be performed asynchronously.

6.220.5. export POST

Exports a virtual machine to an export domain.

For example, to export virtual machine **123** to the export domain **myexport**:

```
POST /ovirt-engine/api/vms/123/export
```

With a request body like this:

```
<action>
  <storage_domain>
    <name>myexport</name>
  </storage_domain>
  <exclusive>true</exclusive>
  <discard_snapshots>true</discard_snapshots>
</action>
```

Table 6.679. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the export should be performed asynchronously.

Name	Type	Direction	Summary
discard_snapshots	Boolean	In	Use the discard_snapshots parameter when the virtual machine should be exported with all of its snapshots collapsed.
exclusive	Boolean	In	Use the exclusive parameter when the virtual machine should be exported even if another copy of it already exists in the export domain (override).
storage_domain	StorageDomain	In	

6.220.6. freezefilesystems POST

Freezes virtual machine file systems.

This operation freezes a virtual machine's file systems using the QEMU guest agent when taking a live snapshot of a running virtual machine. Normally, this is done automatically by the manager, but this must be executed manually with the API for virtual machines using OpenStack Volume (Cinder) disks.

Example:

```
POST /ovirt-engine/api/vms/123/freezefilesystems
```

```
<action/>
```

Table 6.680. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the freeze should be performed asynchronously.

6.220.7. get GET

Retrieves the description of the virtual machine.

Table 6.681. Parameters summary

Name	Type	Direction	Summary
all_content	Boolean	In	Indicates if all of the attributes of the virtual machine should be included in the response.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

Name	Type	Direction	Summary
next_run	Boolean	In	Indicates if the returned result describes the virtual machine as it is currently running or if describes the virtual machine with the modifications that have already been performed but that will only come into effect when the virtual machine is restarted.
vm	Vm	Out	Description of the virtual machine.

6.220.7.1. all_content

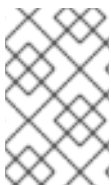
Indicates if all of the attributes of the virtual machine should be included in the response.

By default the following attributes are excluded:

- **console**
- **initialization.configuration.data** - The OVF document describing the virtual machine.
- **rng_source**
- **soundcard**
- **virtio_scsi**

For example, to retrieve the complete representation of the virtual machine '123':

```
GET /ovirt-engine/api/vms/123?all_content=true
```



NOTE

These attributes are not included by default as they reduce performance. These attributes are seldom used and require additional queries to the database. Only use this parameter when required as it will reduce performance.

6.220.7.2. next_run

Indicates if the returned result describes the virtual machine as it is currently running or if describes the virtual machine with the modifications that have already been performed but that will only come into effect when the virtual machine is restarted. By default the value is **false**.

If the parameter is included in the request, but without a value, it is assumed that the value is **true**. The the following request:

```
GET /vms/{vm:id};next_run
```

Is equivalent to using the value **true**:

```
GET /vms/{vm:id};next_run=true
```

6.220.8. logon POST

Initiates the automatic user logon to access a virtual machine from an external console.

This action requires the **ovirt-guest-agent-gdm-plugin** and the **ovirt-guest-agent-pam-module** packages to be installed and the **ovirt-guest-agent** service to be running on the virtual machine.

Users require the appropriate user permissions for the virtual machine in order to access the virtual machine from an external console.

For example:

```
POST /ovirt-engine/api/vms/123/logon
```

Request body:

```
<action/>
```

Table 6.682. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the logon should be performed asynchronously.

6.220.9. maintenance POST

Sets the global maintenance mode on the hosted engine virtual machine.

This action has no effect on other virtual machines.

Example:

```
POST /ovirt-engine/api/vms/123/maintenance
```

```
<action>
  <maintenance_enabled>true<maintenance_enabled/>
</action>
```

Table 6.683. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the global maintenance action should be performed asynchronously.
maintenance_enabled	Boolean	In	Indicates if global maintenance should be enabled or disabled.

6.220.10. migrate POST

Migrates a virtual machine to another physical host.

Example:

```
POST /ovirt-engine/api/vms/123/migrate
```

To specify a specific host to migrate the virtual machine to:

```
<action>
  <host id="2ab5e1da-b726-4274-bbf7-0a42b16a0fc3"/>
</action>
```

Table 6.684. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the migration should be performed asynchronously.
cluster	Cluster	In	Specifies the cluster the virtual machine should migrate to.
force	Boolean	In	Specifies that the virtual machine should migrate even if the virtual machine is defined as non-migratable.
host	Host	In	Specifies a specific host that the virtual machine should migrate to.

6.220.10.1. cluster

Specifies the cluster the virtual machine should migrate to. This is an optional parameter. By default, the virtual machine is migrated to another host within the same cluster.

6.220.10.2. force

Specifies that the virtual machine should migrate even if the virtual machine is defined as non-migratable. This is an optional parameter. By default, it is set to **false**.

6.220.10.3. host

Specifies a specific host that the virtual machine should migrate to. This is an optional parameter. By default, the Red Hat Virtualization Manager automatically selects a default host for migration within the same cluster. If an API user requires a specific host, the user can specify the host with either an **id** or **name** parameter.

6.220.11. previewsnapshot POST

Temporarily restores the virtual machine to the state of a snapshot.

The snapshot is indicated with the `snapshot.id` parameter. It is restored temporarily, so that the content can be inspected. Once that inspection is finished, the state of the virtual machine can be made permanent, using the `commit_snapshot` method, or discarded using the `undo_snapshot` method.

Table 6.685. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	<code>Boolean</code>	In	Indicates if the preview should be performed asynchronously.
<code>disks</code>	<code>Disk[]</code>	In	
<code>restore_memory</code>	<code>Boolean</code>	In	
<code>snapshot</code>	<code>Snapshot</code>	In	
<code>vm</code>	<code>Vm</code>	In	

6.220.12. reboot POST

Sends a reboot request to a virtual machine.

For example:

```
POST /ovirt-engine/api/vms/123/reboot
```

The reboot action does not take any action specific parameters; therefore, the request body should contain an empty `action`:

```
<action/>
```

Table 6.686. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	<code>Boolean</code>	In	Indicates if the reboot should be performed asynchronously.

6.220.13. remove DELETE

Removes the virtual machine, including the virtual disks attached to it.

For example, to remove the virtual machine with identifier `123`:

```
DELETE /ovirt-engine/api/vms/123
```

Table 6.687. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.
detach_only	Boolean	In	Indicates if the attached virtual disks should be detached first and preserved instead of being removed.
force	Boolean	In	Indicates if the virtual machine should be forcibly removed.

6.220.13.1. force

Indicates if the virtual machine should be forcibly removed.

Locked virtual machines and virtual machines with locked disk images cannot be removed without this flag set to true.

6.220.14. reordermacaddresses POST

Table 6.688. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.

6.220.15. shutdown POST

This operation sends a shutdown request to a virtual machine.

For example:

```
POST /ovirt-engine/api/vms/123/shutdown
```

The shutdown action does not take any action specific parameters; therefore, the request body should contain an empty **action**:

```
<action/>
```

Table 6.689. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the shutdown should be performed asynchronously.

6.220.16. start POST

Starts the virtual machine.

If the virtual environment is complete and the virtual machine contains all necessary components to function, it can be started.

This example starts the virtual machine:

```
POST /ovirt-engine/api/vms/123/start
```

With a request body:

```
<action/>
```

Table 6.690. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the start action should be performed asynchronously.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
pause	Boolean	In	If set to true , start the virtual machine in paused mode.
use_cloud_init	Boolean	In	If set to true , the initialization type is set to <i>cloud-init</i> .
use_sysprep	Boolean	In	If set to true , the initialization type is set to <i>Sysprep</i> .
vm	Vm	In	The definition of the virtual machine for this specific run.

6.220.16.1. pause

If set to **true**, start the virtual machine in paused mode. The default is **false**.

6.220.16.2. use_cloud_init

If set to **true**, the initialization type is set to *cloud-init*. The default value is **false**. See [this](#) for details.

6.220.16.3. use_sysprep

If set to **true**, the initialization type is set to *Sysprep*. The default value is **false**. See [this](#) for details.

6.220.16.4. vm

The definition of the virtual machine for this specific run.

For example:

```
<action>
  <vm>
    <os>
      <boot>
        <devices>
          <device>cdrom</device>
        </devices>
      </boot>
    </os>
  </vm>
</action>
```

This will set the boot device to the CDROM only for this specific start. After the virtual machine is powered off, this definition will be reverted.

6.220.17. stop POST

This operation forces a virtual machine to power-off.

For example:

```
POST /ovirt-engine/api/vms/123/stop
```

The stop action does not take any action specific parameters; therefore, the request body should contain an empty **action**:

```
<action/>
```

Table 6.691. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the stop action should be performed asynchronously.

6.220.18. suspend POST

This operation saves the virtual machine state to disk and stops it. Start a suspended virtual machine and restore the virtual machine state with the start action.

For example:

```
POST /ovirt-engine/api/vms/123/suspend
```

The suspend action does not take any action specific parameters; therefore, the request body should contain an empty **action**:

```
<action/>
```

Table 6.692. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the suspend action should be performed asynchronously.

6.220.19. thawfilesystems POST

Thaws virtual machine file systems.

This operation thaws a virtual machine's file systems using the QEMU guest agent when taking a live snapshot of a running virtual machine. Normally, this is done automatically by the manager, but this must be executed manually with the API for virtual machines using OpenStack Volume (Cinder) disks.

Example:

```
POST /api/vms/123/thawfilesystems
```

```
<action/>
```

Table 6.693. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the thaw file systems action should be performed asynchronously.

6.220.20. ticket POST

Generates a time-sensitive authentication token for accessing a virtual machine's display.

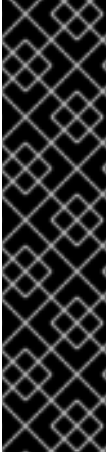
For example:

```
POST /ovirt-engine/api/vms/123/ticket
```

The client-provided action optionally includes a desired ticket value and/or an expiry time in seconds.

The response specifies the actual ticket value and expiry used.

```
<action>
  <ticket>
    <value>abcd12345</value>
    <expiry>120</expiry>
  </ticket>
</action>
```



IMPORTANT

If the virtual machine is configured to support only one graphics protocol then the generated authentication token will be valid for that protocol. But if the virtual machine is configured to support multiple protocols, VNC and SPICE, then the authentication token will only be valid for the SPICE protocol.

In order to obtain an authentication token for a specific protocol, for example for VNC, use the `ticket` method of the [service](#), which manages the graphics consoles of the virtual machine, by sending a request:

```
POST /ovirt-engine/api/vms/123/graphicsconsoles/456/ticket
```

Table 6.694. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the generation of the ticket should be performed asynchronously.
<code>ticket</code>	Ticket	In/Out	

6.220.21. undosnapshot POST

Restores the virtual machine to the state it had before previewing the snapshot.

See the [preview_snapshot](#) operation for details.

Table 6.695. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the undo snapshot action should be performed asynchronously.

6.220.22. update PUT

Table 6.696. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the update should be performed asynchronously.
<code>next_run</code>	Boolean	In	Indicates if the update should be applied to the virtual machine immediately or if it should be applied only when the virtual machine is restarted.
<code>vm</code>	Vm	In/Out	

6.220.22.1. next_run

Indicates if the update should be applied to the virtual machine immediately or if it should be applied only when the virtual machine is restarted. The default value is **false**, so by default changes are applied immediately.

6.221. VMAPPLICATION

A service that provides information about an application installed in a virtual machine.

Table 6.697. Methods summary

Name	Summary
get	Returns the information about the application.

6.221.1. get GET

Returns the information about the application.

Table 6.698. Parameters summary

Name	Type	Direction	Summary
applicati on	Application	Out	The information about the application.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

6.221.1.1. application

The information about the application.

The information consists of **name** attribute containing the name of the application (which is an arbitrary string that may also contain additional information such as version) and **vm** attribute identifying the virtual machine.

For example, a request like this:

```
GET /ovirt-engine/api/vms/123/applications/789
```

May return information like this:

```
<application href="/ovirt-engine/api/vms/123/applications/789" id="789">
  <name>ovirt-guest-agent-common-1.0.12-3.el7</name>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</application>
```

6.222. VMAPPLICATIONS

A service that provides information about applications installed in a virtual machine.

Table 6.699. Methods summary

Name	Summary
list	Returns a list of applications installed in the virtual machine.

6.222.1. list GET

Returns a list of applications installed in the virtual machine.

Table 6.700. Parameters summary

Name	Type	Direction	Summary
applicati ons	Application[]	Out	A list of applications installed in the virtual machine.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
max	Integer	In	Sets the maximum number of applications to return.

6.222.1.1. applications

A list of applications installed in the virtual machine.

For example, a request like this:

```
GET /ovirt-engine/api/vms/123/applications/
```

May return a list like this:

```
<applications>
  <application href="/ovirt-engine/api/vms/123/applications/456" id="456">
    <name>kernel-3.10.0-327.36.1.el7</name>
    <vm href="/ovirt-engine/api/vms/123" id="123"/>
  </application>
  <application href="/ovirt-engine/api/vms/123/applications/789" id="789">
    <name>ovirt-guest-agent-common-1.0.12-3.el7</name>
    <vm href="/ovirt-engine/api/vms/123" id="123"/>
  </application>
</applications>
```

6.222.1.2. max

Sets the maximum number of applications to return. If not specified all the applications are returned.

6.223. VMCDROM

Manages a CDROM device of a virtual machine.

Changing and ejecting the disk is done using always the **update** method, to change the value of the **file** attribute.

Table 6.701. Methods summary

Name	Summary
get	Returns the information about this CDROM device.
update	Updates the information about this CDROM device.

6.223.1. get GET

Returns the information about this CDROM device.

The information consists of **cdrom** attribute containing reference to the CDROM device, the virtual machine, and optionally the inserted disk.

If there is a disk inserted then the **file** attribute will contain a reference to the ISO image:

```
<cdrom href="..." id="00000000-0000-0000-0000-000000000000">
  <file id="mycd.iso"/>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</cdrom>
```

If there is no disk inserted then the **file** attribute won't be reported:

```
<cdrom href="..." id="00000000-0000-0000-0000-000000000000">
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</cdrom>
```

Table 6.702. Parameters summary

Name	Type	Direction	Summary
cdrom	Cdrom	Out	The information about the CDROM device.
current	Boolean	In	Indicates if the operation should return the information for the currently running virtual machine.

6.223.1.1. current

Indicates if the operation should return the information for the currently running virtual machine. This parameter is optional, and the default value is **false**.

6.223.2. update PUT

Updates the information about this CDROM device.

It allows to change or eject the disk by changing the value of the **file** attribute. For example, to insert or change the disk send a request like this:

```
PUT /ovirt-engine/api/vms/123/cdroms/00000000-0000-0000-0000-000000000000
```

The body should contain the new value for the **file** attribute:

```
<cdrom>
  <file id="mycd.iso"/>
</cdrom>
```

The value of the **id** attribute, **mycd.iso** in this example, should correspond to a file available in an attached ISO storage domain.

To eject the disk use a **file** with an empty **id**:

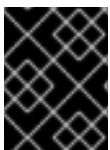
```
<cdrom>
  <file id=""/>
</cdrom>
```

By default the above operations change permanently the disk that will be visible to the virtual machine after the next boot, but they don't have any effect on the currently running virtual machine. If you want to change the disk that is visible to the current running virtual machine, add the **current=true** parameter. For example, to eject the current disk send a request like this:

```
PUT /ovirt-engine/api/vms/123/cdroms/00000000-0000-0000-0000-000000000000?
current=true
```

With a request body like this:

```
<cdrom>
  <file id=""/>
</cdrom>
```



IMPORTANT

The changes made with the **current=true** parameter are never persisted, so they won't have any effect after the virtual machine is rebooted.

Table 6.703. Parameters summary

Name	Type	Direction	Summary
cdrom	Cdrom	In/Out	The information about the CDROM device.
current	Boolean	In	Indicates if the update should apply to the currently running virtual machine, or to the virtual machine after the next boot.

6.223.2.1. current

Indicates if the update should apply to the currently running virtual machine, or to the virtual machine after the next boot. This parameter is optional, and the default value is **false**, which means that by default the update will have effect only after the next boot.

6.224. VMCDROMS

Manages the CDROM devices of a virtual machine.

Currently virtual machines have exactly one CDROM device. No new devices can be added, and the existing one can't be removed, thus there are no **add** or **remove** methods. Changing and ejecting CDROM disks is done with the [update](#) method of the [service](#) that manages the CDROM device.

Table 6.704. Methods summary

Name	Summary
list	Returns the list of CDROM devices of the virtual machine.

6.224.1. list GET

Returns the list of CDROM devices of the virtual machine.

Table 6.705. Parameters summary

Name	Type	Direction	Summary
cdroms	Cdrom[]	Out	The list of CDROM devices of the virtual machine.
max	Integer	In	Sets the maximum number of CDROMs to return.

6.224.1.1. max

Sets the maximum number of CDROMs to return. If not specified all the CDROMs are returned.

6.225. VMDISK

Table 6.706. Methods summary

Name	Summary
activate	
deactivate	
export	
get	
move	

Name	Summary
remove	Detach the disk from the virtual machine.
update	

6.225.1. activate POST

Table 6.707. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the activation should be performed asynchronously.

6.225.2. deactivate POST

Table 6.708. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the deactivation should be performed asynchronously.

6.225.3. export POST

Table 6.709. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the export should be performed asynchronously.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

6.225.4. get GET

Table 6.710. Parameters summary

Name	Type	Direction	Summary
disk	Disk	Out	

6.225.5. move POST

Table 6.711. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the move should be performed asynchronously.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.

6.225.6. remove DELETE

Detach the disk from the virtual machine.



NOTE

In version 3 of the API this used to also remove the disk completely from the system, but starting with version 4 it doesn't. If you need to remove it completely use the [remove method of the top level disk service](#).

Table 6.712. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.225.7. update PUT

Table 6.713. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
disk	Disk	In/Out	

6.226. VMDISKS

Table 6.714. Methods summary

Name	Summary
add	
list	

6.226.1. add POST

Table 6.715. Parameters summary

Name	Type	Direction	Summary
disk	Disk	In/Out	

6.226.2. list GET

Table 6.716. Parameters summary

Name	Type	Direction	Summary
disks	Disk[]	Out	
max	Integer	In	Sets the maximum number of disks to return.

6.226.2.1. max

Sets the maximum number of disks to return. If not specified all the disks are returned.

6.227. VMGRAPHICSCONSOLE

Table 6.717. Methods summary

Name	Summary
get	Gets graphics console configuration of the virtual machine.
proxyticket	
remoteviewer connectionfile	Generates the file which is compatible with remote-viewer client.
remove	Remove the graphics console from the virtual machine.
ticket	Generates a time-sensitive authentication token for accessing this virtual machine's console.

6.227.1. get GET

Gets graphics console configuration of the virtual machine.

Table 6.718. Parameters summary

Name	Type	Direction	Summary
console	GraphicsConsole	Out	The information about the graphics console of the virtual machine.
current	Boolean	In	Use the following query to obtain the current run-time configuration of the graphics console.

6.227.1.1. current

Use the following query to obtain the current run-time configuration of the graphics console.

```
GET /ovirt-engine/api/vms/123/graphicsconsoles/456?current=true
```

The default value is **false**.

6.227.2. proxyticket POST

Table 6.719. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the generation of the ticket should be performed asynchronously.
proxy_ticket	ProxyTicket	Out	

6.227.3. remoteviewerconnectionfile POST

Generates the file which is compatible with **remote-viewer** client.

Use the following request to generate remote viewer connection file of the graphics console. Note that this action generates the file only if virtual machine is running.

```
POST /ovirt-engine/api/vms/123/graphicsconsoles/456/remoteviewerconnectionfile
```

The **remoteviewerconnectionfile** action does not take any action specific parameters, so the request body should contain an empty **action**:

```
<action/>
```

The response contains the file, which can be used with **remote-viewer** client.

```
<action>
  <remote_viewer_connection_file>
    [virt-viewer]
    type=spice
```

```

host=192.168.1.101
port=-1
password=123456789
delete-this-file=1
fullscreen=0
toggle-fullscreen=shift+f11
release-cursor=shift+f12
secure-attention=ctrl+alt+end
tls-port=5900
enable-smartcard=0
enable-usb-autoshare=0
usb-filter=null
tls-ciphers=DEFAULT
host-subject=0=local,CN=example.com
ca=...
</remote_viewer_connection_file>
</action>

```

E.g., to fetch the content of remote viewer connection file and save it into temporary file, user can use oVirt Python SDK as follows:

```

# Find the virtual machine:
vm = vms_service.list(search='name=myvm')[0]

# Locate the service that manages the virtual machine, as that is where
# the locators are defined:
vm_service = vms_service.vm_service(vm.id)

# Find the graphic console of the virtual machine:
graphics_consoles_service = vm_service.graphics_consoles_service()
graphics_console = graphics_consoles_service.list()[0]

# Generate the remote viewer connection file:
console_service =
graphics_consoles_service.console_service(graphics_console.id)
remote_viewer_connection_file =
console_service.remote_viewer_connection_file()

# Write the content to file "/tmp/remote_viewer_connection_file.vv"
path = "/tmp/remote_viewer_connection_file.vv"
with open(path, "w") as f:
    f.write(remote_viewer_connection_file)

```

When you create the remote viewer connection file, then you can connect to virtual machine graphic console, as follows:

```

#!/bin/sh -ex

remote-viewer --ovirt-ca-file=/etc/pki/ovirt-engine/ca.pem
/tmp/remote_viewer_connection_file.vv

```

Table 6.720. Parameters summary

Name	Type	Direction	Summary
remote_viewer_connection_file	String	Out	Contains the file which is compatible with remote-viewer client.

6.227.3.1. remote_viewer_connection_file

Contains the file which is compatible with **remote-viewer** client.

User can use the content of this attribute to create a file, which can be passed to **remote-viewer** client to connect to virtual machine graphic console.

6.227.4. remove DELETE

Remove the graphics console from the virtual machine.

Table 6.721. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.227.5. ticket POST

Generates a time-sensitive authentication token for accessing this virtual machine's console.

```
POST /ovirt-engine/api/vms/123/graphicsconsoles/456/ticket
```

The client-provided action optionally includes a desired ticket value and/or an expiry time in seconds.

In any case, the response specifies the actual ticket value and expiry used.

```
<action>
  <ticket>
    <value>abcd12345</value>
    <expiry>120</expiry>
  </ticket>
</action>
```

Table 6.722. Parameters summary

Name	Type	Direction	Summary
ticket	Ticket	In/Out	The generated ticket that can be used to access this console.

6.228. VMGRAPHICSCONSOLES

Table 6.723. Methods summary

Name	Summary
add	Add new graphics console to the virtual machine.
list	Lists all the configured graphics consoles of the virtual machine.

6.228.1. add POST

Add new graphics console to the virtual machine.

Table 6.724. Parameters summary

Name	Type	Direction	Summary
console	GraphicsConsole	In/Out	

6.228.2. list GET

Lists all the configured graphics consoles of the virtual machine.

Table 6.725. Parameters summary

Name	Type	Direction	Summary
consoles	GraphicsConsole[]	Out	The list of graphics consoles of the virtual machine.
current	Boolean	In	Use the following query to obtain the current run-time configuration of the graphics consoles.
max	Integer	In	Sets the maximum number of consoles to return.

6.228.2.1. current

Use the following query to obtain the current run-time configuration of the graphics consoles.

```
GET /ovirt-engine/api/vms/123/graphicsconsoles?current=true
```

The default value is **false**.

6.228.2.2. max

Sets the maximum number of consoles to return. If not specified all the consoles are returned.

6.229. VMHOSTDEVICE

A service to manage individual host device attached to a virtual machine.

Table 6.726. Methods summary

Name	Summary
get	Retrieve information about particular host device attached to given virtual machine.
remove	Remove the attachment of this host device from given virtual machine.

6.229.1. get GET

Retrieve information about particular host device attached to given virtual machine.

Example:

```
GET /ovirt-engine/api/vms/123/hostdevices/456
```

```
<host_device href="/ovirt-engine/api/hosts/543/devices/456" id="456">
  <name>pci_0000_04_00_0</name>
  <capability>pci</capability>
  <iommu_group>30</iommu_group>
  <placeholder>true</placeholder>
  <product id="0x13ba">
    <name>GM107GL [Quadro K2200]</name>
  </product>
  <vendor id="0x10de">
    <name>NVIDIA Corporation</name>
  </vendor>
  <host href="/ovirt-engine/api/hosts/543" id="543"/>
  <parent_device href="/ovirt-engine/api/hosts/543/devices/456" id="456">
    <name>pci_0000_00_03_0</name>
  </parent_device>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</host_device>
```

Table 6.727. Parameters summary

Name	Type	Direction	Summary
device	HostDevice	Out	Retrieved information about the host device attached to given virtual machine.

6.229.2. remove DELETE

Remove the attachment of this host device from given virtual machine.

**NOTE**

In case this device serves as an IOMMU placeholder, it cannot be removed (remove will result only in setting its **placeholder** flag to **true**). Note that all IOMMU placeholder devices will be removed automatically as soon as there will be no more non-placeholder devices (all devices from given IOMMU group are detached).

```
DELETE /ovirt-engine/api/vms/123/hostdevices/456
```

Table 6.728. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.230. VMHOSTDEVICES

A service to manage host devices attached to a virtual machine.

Table 6.729. Methods summary

Name	Summary
add	Attach target device to given virtual machine.
list	List the host devices assigned to given virtual machine.

6.230.1. add POST

Attach target device to given virtual machine.

Example:

```
POST /ovirt-engine/api/vms/123/hostdevices
```

With request body of type [HostDevice](#), for example

```
<host_device id="123" />
```

**NOTE**

A necessary precondition for a successful host device attachment is that the virtual machine must be pinned to **exactly** one host. The device ID is then taken relative to this host.

**NOTE**

Attachment of a PCI device that is part of a bigger IOMMU group will result in attachment of the remaining devices from that IOMMU group as "placeholders". These devices are then identified using the **placeholder** attribute of the [HostDevice](#) type set to **true**.

In case you want attach a device that already serves as an IOMMU placeholder, simply issue an explicit Add operation for it, and its **placeholder** flag will be cleared, and the device will be accessible to the virtual machine.

Table 6.730. Parameters summary

Name	Type	Direction	Summary
device	HostDevice	In/Out	The host device to be attached to given virtual machine.

6.230.2. list GET

List the host devices assigned to given virtual machine.

Table 6.731. Parameters summary

Name	Type	Direction	Summary
device	HostDevice[]	Out	Retrieved list of host devices attached to given virtual machine.
max	Integer	In	Sets the maximum number of devices to return.

6.230.2.1. max

Sets the maximum number of devices to return. If not specified all the devices are returned.

6.231. VMNIC**Table 6.732. Methods summary**

Name	Summary
activate	
deactivate	
get	
remove	Removes the NIC.
update	Updates the NIC.

6.231.1. activate POST

Table 6.733. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the activation should be performed asynchronously.

6.231.2. deactivate POST

Table 6.734. Parameters summary

Name	Type	Direction	Summary
<code>async</code>	Boolean	In	Indicates if the deactivation should be performed asynchronously.

6.231.3. get GET

Table 6.735. Parameters summary

Name	Type	Direction	Summary
<code>nic</code>	Nic	Out	

6.231.4. remove DELETE

Removes the NIC.

For example, to remove the NIC with id **456** from the virtual machine with id **123** send a request like this:

```
DELETE /ovirt-engine/api/vms/123/nics/456
```



IMPORTANT

The hotplugging feature only supports virtual machine operating systems with hotplugging operations. Example operating systems include:

- Red Hat Enterprise Linux 6
- Red Hat Enterprise Linux 5
- Windows Server 2008 and
- Windows Server 2003

Table 6.736. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.231.5. update PUT

Updates the NIC.

For example, to update the NIC having with **456** belonging to virtual the machine with id **123** send a request like this:

```
PUT /ovirt-engine/api/vms/123/nics/456
```

With a request body like this:

```
<nic>
  <name>myNIC</name>
  <interface>e1000</interface>
  <vnic_profile id='789' />
</nic>
```



IMPORTANT

The hotplugging feature only supports virtual machine operating systems with hotplugging operations. Example operating systems include:

- Red Hat Enterprise Linux 6
- Red Hat Enterprise Linux 5
- Windows Server 2008 and
- Windows Server 2003

Table 6.737. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
nic	Nic	In/Out	

6.232. VMNICS

Table 6.738. Methods summary

Name	Summary
add	Adds a NIC to the virtual machine.
list	

6.232.1. add POST

Adds a NIC to the virtual machine.

The following example adds a network interface named **mynic** using **virtio** and the **ovirtmgmt** network to the virtual machine.

```
POST /ovirt-engine/api/vms/123/nics
```

```
<nic>
  <interface>virtio</interface>
  <name>mynic</name>
  <network>
    <name>ovirtmgmt</name>
  </network>
</nic>
```

The following example sends that request using **curl**:

```
curl \
  --request POST \
  --header "Version: 4" \
  --header "Content-Type: application/xml" \
  --header "Accept: application/xml" \
  --user "admin@internal:mypassword" \
  --cacert /etc/pki/ovirt-engine/ca.pem \
  --data '
<nic>
  <name>mynic</name>
  <network>
    <name>ovirtmgmt</name>
  </network>
</nic>
' \
https://myengine.example.com/ovirt-engine/api/vms/123/nics
```




IMPORTANT

The hotplugging feature only supports virtual machine operating systems with hotplugging operations. Example operating systems include:

- Red Hat Enterprise Linux 6
- Red Hat Enterprise Linux 5
- Windows Server 2008 and
- Windows Server 2003

Table 6.739. Parameters summary

Name	Type	Direction	Summary
<code>nic</code>	Nic	In/Out	

6.232.2. list GET

Table 6.740. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of NICs to return.
<code>nics</code>	Nic[]	Out	

6.232.2.1. max

Sets the maximum number of NICs to return. If not specified all the NICs are returned.

6.233. VMNUMANODE

Table 6.741. Methods summary

Name	Summary
<code>get</code>	
<code>remove</code>	Removes a virtual NUMA node.
<code>update</code>	Updates a virtual NUMA node.

6.233.1. get GET

Table 6.742. Parameters summary

Name	Type	Direction	Summary
node	VirtualNuma Node	Out	

6.233.2. remove DELETE

Removes a virtual NUMA node.

An example of removing a virtual NUMA node:

```
DELETE /ovirt-engine/api/vms/123/numanodes/456
```

Table 6.743. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.233.3. update PUT

Updates a virtual NUMA node.

An example of pinning a virtual NUMA node to a physical NUMA node on the host:

```
PUT /ovirt-engine/api/vms/123/numanodes/456
```

The request body should contain the following:

```
<vm_numa_node>
  <numa_node_pins>
    <numa_node_pin>
      <index>0</index>
    </numa_node_pin>
  </numa_node_pins>
</vm_numa_node>
```

Table 6.744. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
node	VirtualNuma Node	In/Out	

6.234. VMNUMANODES

Table 6.745. Methods summary

Name	Summary
add	Creates a new virtual NUMA node for the virtual machine.
list	Lists virtual NUMA nodes of a virtual machine.

6.234.1. add POST

Creates a new virtual NUMA node for the virtual machine.

An example of creating a NUMA node:

```
POST /ovirt-engine/api/vms/c7ecd2dc/numanodes
Accept: application/xml
Content-type: application/xml
```

The request body can contain the following:

```
<vm_numa_node>
  <cpu>
    <cores>
      <core>
        <index>0</index>
      </core>
    </cores>
  </cpu>
  <index>0</index>
  <memory>1024</memory>
</vm_numa_node>
```

Table 6.746. Parameters summary

Name	Type	Direction	Summary
node	VirtualNuma Node	In/Out	

6.234.2. list GET

Lists virtual NUMA nodes of a virtual machine.

Table 6.747. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of nodes to return.

Name	Type	Direction	Summary
nodes	VirtualNuma Node[]	Out	

6.234.2.1. max

Sets the maximum number of nodes to return. If not specified all the nodes are returned.

6.235. VMPOOL

A service to manage a virtual machines pool.

Table 6.748. Methods summary

Name	Summary
allocatevm	This operation allocates a virtual machine in the virtual machine pool.
get	Get the virtual machine pool.
remove	Removes a virtual machine pool.
update	Update the virtual machine pool.

6.235.1. allocatevm POST

This operation allocates a virtual machine in the virtual machine pool.

```
POST /ovirt-engine/api/vmpools/123/allocatevm
```

The allocate virtual machine action does not take any action specific parameters, so the request body should contain an empty **action**:

```
<action/>
```

Table 6.749. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the allocation should be performed asynchronously.

6.235.2. get GET

Get the virtual machine pool.

```
GET /ovirt-engine/api/vmpools/123
```

You will get a XML response like that one:

```
<vm_pool id="123">
  <actions>...</actions>
  <name>MyVmPool</name>
  <description>MyVmPool description</description>
  <link href="/ovirt-engine/api/vmpools/123/permissions"
rel="permissions"/>
  <max_user_vms>1</max_user_vms>
  <prestarted_vms>0</prestarted_vms>
  <size>100</size>
  <stateful>>false</stateful>
  <type>automatic</type>
  <use_latest_template_version>>false</use_latest_template_version>
  <cluster id="123"/>
  <template id="123"/>
  <vm id="123">...</vm>
  ...
</vm_pool>
```

Table 6.750. Parameters summary

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
pool	VmPool	Out	Retrieved virtual machines pool.

6.235.3. remove DELETE

Removes a virtual machine pool.

```
DELETE /ovirt-engine/api/vmpools/123
```

Table 6.751. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.235.4. update PUT

Update the virtual machine pool.

```
PUT /ovirt-engine/api/vmpools/123
```

The **name**, **description**, **size**, **prestarted_vms** and **max_user_vms** attributes can be updated after the virtual machine pool has been created.

```
<vmPool>
  <name>VM_Pool_B</name>
  <description>Virtual Machine Pool B</description>
  <size>3</size>
  <prestarted_vms>1</size>
  <max_user_vms>2</size>
</vmPool>
```

Table 6.752. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
pool	VmPool	In/Out	The virtual machine pool that is being updated.

6.236. VMPOOLS

Provides read-write access to virtual machines pools.

Table 6.753. Methods summary

Name	Summary
add	Creates a new virtual machine pool.
list	Get a list of available virtual machines pools.

6.236.1. add POST

Creates a new virtual machine pool.

A new pool requires the **name**, **cluster** and **template** attributes. Identify the cluster and template with the **id** or **name** nested attributes:

```
POST /ovirt-engine/api/vmpools
```

With the following body:

```
<vmPool>
  <name>mypool</name>
  <cluster id="123"/>
  <template id="456"/>
</vmPool>
```

Table 6.754. Parameters summary

Name	Type	Direction	Summary
pool	VmPool	In/Out	Pool to add.

6.236.2. list GET

Get a list of available virtual machines pools.

```
GET /ovirt-engine/api/vmpools
```

You will receive the following response:

```
<vm_pools>
  <vm_pool id="123">
    ...
  </vm_pool>
  ...
</vm_pools>
```

Table 6.755. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
max	Integer	In	Sets the maximum number of pools to return.
pools	VmPool[]	Out	Retrieved pools.
search	String	In	A query string used to restrict the returned pools.

6.236.2.1. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.236.2.2. max

Sets the maximum number of pools to return. If this value is not specified, all of the pools are returned.

6.237. VMREPORTEDDEVICE

Table 6.756. Methods summary

Name	Summary
get	

6.237.1. get GET

Table 6.757. Parameters summary

Name	Type	Direction	Summary
reported_device	ReportedDevice	Out	

6.238. VMREPORTEDDEVICES

Table 6.758. Methods summary

Name	Summary
list	

6.238.1. list GET

Table 6.759. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of devices to return.
reported_device	ReportedDevice[]	Out	

6.238.1.1. max

Sets the maximum number of devices to return. If not specified all the devices are returned.

6.239. VMSESSION

Table 6.760. Methods summary

Name	Summary
get	

6.239.1. get GET

Table 6.761. Parameters summary

Name	Type	Direction	Summary
<code>session</code>	Session	Out	

6.240. VMSESSIONS

Provides information about virtual machine user sessions.

Table 6.762. Methods summary

Name	Summary
<code>list</code>	Lists all user sessions for this virtual machine.

6.240.1. list GET

Lists all user sessions for this virtual machine.

For example, to retrieve the session information for virtual machine **123** send a request like this:

```
GET /ovirt-engine/api/vms/123/sessions
```

The response body will contain something like this:

```
<sessions>
  <session href="/ovirt-engine/api/vms/123/sessions/456" id="456">
    <console_user>true</console_user>
    <ip>
      <address>192.168.122.1</address>
    </ip>
    <user href="/ovirt-engine/api/users/789" id="789"/>
    <vm href="/ovirt-engine/api/vms/123" id="123"/>
  </session>
  ...
</sessions>
```

Table 6.763. Parameters summary

Name	Type	Direction	Summary
<code>max</code>	Integer	In	Sets the maximum number of sessions to return.
<code>sessions</code>	Session[]	Out	

6.240.1.1. max

Sets the maximum number of sessions to return. If not specified all the sessions are returned.

6.241. VMWATCHDOG

A service managing a watchdog on virtual machines.

Table 6.764. Methods summary

Name	Summary
get	Returns the information about the watchdog.
remove	Removes the watchdog from the virtual machine.
update	Updates the information about the watchdog.

6.241.1. get GET

Returns the information about the watchdog.

Table 6.765. Parameters summary

Name	Type	Direction	Summary
watchdog	Watchdog	Out	The information about the watchdog.

6.241.1.1. watchdog

The information about the watchdog.

The information consists of **model** element, **action** element and the reference to the virtual machine. It may look like this:

```
<watchdogs>
  <watchdog href="/ovirt-engine/api/vms/123/watchdogs/00000000-0000-0000-0000-000000000000" id="00000000-0000-0000-0000-000000000000">
    <vm href="/ovirt-engine/api/vms/123" id="123"/>
    <action>poweroff</action>
    <model>i6300esb</model>
  </watchdog>
</watchdogs>
```

6.241.2. remove DELETE

Removes the watchdog from the virtual machine.

For example, to remove a watchdog from a virtual machine, send a request like this:

```
DELETE /ovirt-engine/api/vms/123/watchdogs/00000000-0000-0000-0000-000000000000
```

Table 6.766. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.241.3. update PUT

Updates the information about the watchdog.

You can update the information using **action** and **model** elements.

For example, to update a watchdog, send a request like this:

```
PUT /ovirt-engine/api/vms/123/watchdogs
<watchdog>
  <action>reset</action>
</watchdog>
```

with response body:

```
<watchdog href="/ovirt-engine/api/vms/123/watchdogs/00000000-0000-0000-0000-000000000000" id="00000000-0000-0000-0000-000000000000">
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
  <action>reset</action>
  <model>i6300esb</model>
</watchdog>
```

Table 6.767. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
watchdog	Watchdog	In/Out	The information about the watchdog.

6.241.3.1. watchdog

The information about the watchdog.

The request data must contain at least one of **model** and **action** elements. The response data contains complete information about the updated watchdog.

6.242. VMWATCHDOGS

Lists the watchdogs of a virtual machine.

Table 6.768. Methods summary

Name	Summary
add	Adds new watchdog to the virtual machine.
list	The list of watchdogs of the virtual machine.

6.242.1. add POST

Adds new watchdog to the virtual machine.

For example, to add a watchdog to a virtual machine, send a request like this:

```
POST /ovirt-engine/api/vms/123/watchdogs
<watchdog>
  <action>poweroff</action>
  <model>i6300esb</model>
</watchdog>
```

with response body:

```
<watchdog href="/ovirt-engine/api/vms/123/watchdogs/00000000-0000-0000-0000-000000000000" id="00000000-0000-0000-0000-000000000000">
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
  <action>poweroff</action>
  <model>i6300esb</model>
</watchdog>
```

Table 6.769. Parameters summary

Name	Type	Direction	Summary
watchdog	Watchdog	In/Out	The information about the watchdog.

6.242.1.1. watchdog

The information about the watchdog.

The request data must contain **model** element (such as **i6300esb**) and **action** element (one of **none**, **reset**, **poweroff**, **dump**, **pause**). The response data additionally contains references to the added watchdog and to the virtual machine.

6.242.2. list GET

The list of watchdogs of the virtual machine.

Table 6.770. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of watchdogs to return.

Name	Type	Direction	Summary
watchdogs	Watchdog[]	Out	The information about the watchdog.

6.242.2.1. max

Sets the maximum number of watchdogs to return. If not specified all the watchdogs are returned.

6.242.2.2. watchdogs

The information about the watchdog.

The information consists of **model** element, **action** element and the reference to the virtual machine. It may look like this:

```
<watchdogs>
  <watchdog href="/ovirt-engine/api/vms/123/watchdogs/00000000-0000-0000-
0000-000000000000" id="00000000-0000-0000-0000-000000000000">
    <vm href="/ovirt-engine/api/vms/123" id="123"/>
    <action>poweroff</action>
    <model>i6300esb</model>
  </watchdog>
</watchdogs>
```

6.243. VMS

Table 6.771. Methods summary

Name	Summary
add	Creates a new virtual machine.
list	

6.243.1. add POST

Creates a new virtual machine.

The virtual machine can be created in different ways:

- From a template. In this case the identifier or name of the template must be provided. For example, using a plain shell script and XML:

```
#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."
```

```

curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--user "${user}:${password}" \
--request POST \
--header "Version: 4" \
--header "Content-Type: application/xml" \
--header "Accept: application/xml" \
--data '
<vm>
  <name>myvm</name>
  <template>
    <name>Blank</name>
  </template>
  <cluster>
    <name>mycluster</name>
  </cluster>
</vm>
' \
"${url}/vms"

```

- From a snapshot. In this case the identifier of the snapshot has to be provided. For example, using a plain shel script and XML:

```

#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."
curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--user "${user}:${password}" \
--request POST \
--header "Content-Type: application/xml" \
--header "Accept: application/xml" \
--data '
<vm>
  <name>myvm</name>
  <snapshots>
    <snapshot id="266742a5-6a65-483c-816d-d2ce49746680"/>
  </snapshots>
  <cluster>
    <name>mycluster</name>
  </cluster>
</vm>
' \
"${url}/vms"

```

When creating a virtual machine from a template or from a snapshot it is usually useful to explicitly indicate in what storage domain to create the disks for the virtual machine. If the virtual machine is created from a template then this is achieved passing a set of **disk_attachment** elements that indicate the mapping:

```
<vm>
```

```

...
<disk_attachments>
  <disk_attachment>
    <disk id="8d4bd566-6c86-4592-a4a7-912dbf93c298">
      <storage_domains>
        <storage_domain id="9cb6cb0a-cf1d-41c2-92ca-5a6d665649c9"/>
      </storage_domains>
    </disk>
  <disk_attachment>
</disk_attachments>
</vm>

```

When the virtual machine is created from a snapshot this set of disks is slightly different, it uses the **image_id** attribute instead of **id**.

```

<vm>
...
<disk_attachments>
  <disk_attachment>
    <disk>
      <image_id>8d4bd566-6c86-4592-a4a7-912dbf93c298</image_id>
      <storage_domains>
        <storage_domain id="9cb6cb0a-cf1d-41c2-92ca-5a6d665649c9"/>
      </storage_domains>
    </disk>
  <disk_attachment>
</disk_attachments>
</vm>

```

It is possible to specify additional virtual machine parameters in the XML description, e.g. a virtual machine of **desktop** type, with 2 GiB of RAM and additional description can be added sending a request body like the following:

```

<vm>
  <name>myvm</name>
  <description>My Desktop Virtual Machine</description>
  <type>desktop</type>
  <memory>2147483648</memory>
  ...
</vm>

```

A bootable CDROM device can be set like this:

```

<vm>
...
<os>
  <boot dev="cdrom"/>
</os>
</vm>

```

In order to boot from CDROM, you first need to insert a disk, as described in the [CDROM service](#). Then booting from that CDROM can be specified using the **os.boot.devices** attribute:

```

<vm>

```

```

...
<os>
  <boot>
    <devices>
      <device>cdrom</device>
    </devices>
  </boot>
</os>
</vm>

```

In all cases the name or identifier of the cluster where the virtual machine will be created is mandatory.

Table 6.772. Parameters summary

Name	Type	Direction	Summary
clone	Boolean	In	Specifies if the virtual machine should be independent of the template.
clone_permissions	Boolean	In	Specifies if the permissions of the template should be copied to the virtual machine.
vm	Vm	In/Out	

6.243.1.1. clone

Specifies if the virtual machine should be independent of the template.

When a virtual machine is created from a template by default the disks of the virtual machine depend on the disks of the template, they are using the *copy on write* mechanism so that only the differences from the template take up real storage space. If this parameter is specified and the value is **true** then the disks of the created virtual machine will be *cloned*, and independent of the template. For example, to create an independent virtual machine, send a request like this:

```
POST /ovirt-engine/vms?clone=true
```

With a request body like this:

```

<vm>
  <name>myvm</name>
  <template>
    <name>mytemplate</name>
  </template>
  <cluster>
    <name>mycluster</name>
  </cluster>
</vm>

```



NOTE

When this parameter is **true** the permissions of the template will also be copied, as when using **clone_permissions=true**.

6.243.1.2. clone_permissions

Specifies if the permissions of the template should be copied to the virtual machine.

If this optional parameter is provided, and its value is **true** then the permissions of the template (only the direct ones, not the inherited ones) will be copied to the created virtual machine. For example, to create a virtual machine from the **mytemplate** template copying its permissions, send a request like this:

```
POST /ovirt-engine/api/vms?clone_permissions=true
```

With a request body like this:

```
<vm>
  <name>myvm</name>
  <template>
    <name>mytemplate</name>
  </template>
  <cluster>
    <name>mycluster</name>
  </cluster>
</vm>
```

6.243.2. list GET

Table 6.773. Parameters summary

Name	Type	Direction	Summary
all_content	Boolean	In	Indicates if all the attributes of the virtual machines should be included in the response.
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
max	Integer	In	The maximum number of results to return.
search	String	In	A query string used to restrict the returned virtual machines.
vms	Vm[]	Out	

6.243.2.1. all_content

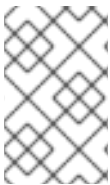
Indicates if all the attributes of the virtual machines should be included in the response.

By default the following attributes are excluded:

- **console**
- **initialization.configuration.data** - The OVF document describing the virtual machine.
- **rng_source**
- **soundcard**
- **virtio_scsi**

For example, to retrieve the complete representation of the virtual machines send a request like this:

```
GET /ovirt-engine/api/vms?all_content=true
```



NOTE

The reason for not including these attributes is performance: they are seldom used and they require additional queries to the database. So try to use the this parameter only when it is really needed.

6.243.2.2. case_sensitive

Indicates if the search performed using the **search** parameter should be performed taking case into account. The default value is **true**, which means that case is taken into account. If you want to search ignoring case set it to **false**.

6.244. VNICPROFILE

This service manages a vNIC profile.

Table 6.774. Methods summary

Name	Summary
get	Retrieves details about a vNIC profile.
remove	Removes the vNIC profile.
update	Updates details of a vNIC profile.

6.244.1. get GET

Retrieves details about a vNIC profile.

Table 6.775. Parameters summary

Name	Type	Direction	Summary
profile	VnicProfile	Out	

6.244.2. remove DELETE

Removes the vNIC profile.

Table 6.776. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.244.3. update PUT

Updates details of a vNIC profile.

Table 6.777. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the update should be performed asynchronously.
profile	VnicProfile	In/Out	The vNIC profile that is being updated.

6.245. VNICPROFILES

This service manages the collection of all vNIC profiles.

Table 6.778. Methods summary

Name	Summary
add	Add a vNIC profile.
list	List all vNIC profiles.

6.245.1. add POST

Add a vNIC profile.

For example to add vNIC profile **123** to network **456** send a request to:

```
POST /ovirt-engine/api/networks/456/vnicprofiles
```

With the following body:

```
<vnic_profile id="123">
  <name>new_vNIC_name</name>
  <pass_through>
    <mode>disabled</mode>
```

```

</pass_through>
<port_mirroring>>false</port_mirroring>
</vnic_profile>

```

Please note that there is a default network filter to each vNIC profile. For more details of how the default network filter is calculated please refer to the documentation in [NetworkFilters](#).

The output of creating a new vNIC profile depends in the body arguments that were given. In case no network filter was given, the default network filter will be configured. For example:

```

<vnic_profile href="/ovirt-engine/api/vnicprofiles/123" id="123">
  <name>new_vNIC_name</name>
  <link href="/ovirt-engine/api/vnicprofiles/123/permissions"
rel="permissions"/>
  <pass_through>
    <mode>disabled</mode>
  </pass_through>
  <port_mirroring>>false</port_mirroring>
  <network href="/ovirt-engine/api/networks/456" id="456"/>
  <network_filter href="/ovirt-engine/api/networkfilters/789" id="789"/>
</vnic_profile>

```

In case an empty network filter was given, no network filter will be configured for the specific vNIC profile regardless of the vNIC profile's default network filter. For example:

```

<vnic_profile>
  <name>no_network_filter</name>
  <network_filter/>
</vnic_profile>

```

In case that a specific valid network filter id was given, the vNIC profile will be configured with the given network filter regardless of the vNIC profiles's default network filter. For example:

```

<vnic_profile>
  <name>user_choice_network_filter</name>
  <network_filter id="0000001b-001b-001b-001b-0000000001d5"/>
</vnic_profile>

```

Table 6.779. Parameters summary

Name	Type	Direction	Summary
profile	VnicProfile	In/Out	The vNIC profile that is being added.

6.245.2. list GET

List all vNIC profiles.

Table 6.780. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of profiles to return.
profiles	VnicProfile[]	Out	The list of all vNIC profiles.

6.245.2.1. max

Sets the maximum number of profiles to return. If not specified all the profiles are returned.

6.246. WEIGHT

Table 6.781. Methods summary

Name	Summary
get	
remove	

6.246.1. get GET

Table 6.782. Parameters summary

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
weight	Weight	Out	

6.246.2. remove DELETE

Table 6.783. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.

6.247. WEIGHTS

Table 6.784. Methods summary

Name	Summary
add	
list	

6.247.1. add POST

Table 6.785. Parameters summary

Name	Type	Direction	Summary
weight	Weight	In/Out	

6.247.2. list GET

Table 6.786. Parameters summary

Name	Type	Direction	Summary
filter	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
max	Integer	In	Sets the maximum number of weights to return.
weights	Weight[]	Out	

6.247.2.1. max

Sets the maximum number of weights to return. If not specified all the weights are returned.

CHAPTER 7. TYPES

This section enumerates all the data types that are available in the API.

7.1. ACCESSPROTOCOL ENUM

Table 7.1. Values summary

Name	Summary
<code>cifs</code>	
<code>gluster</code>	
<code>nfs</code>	

7.2. ACTION STRUCT

Table 7.2. Attributes summary

Name	Type	Summary
<code>allow_partial_import</code>	Boolean	
<code>async</code>	Boolean	
<code>bricks</code>	GlusterBrick[]	
<code>certificates</code>	Certificate[]	
<code>check_connectivity</code>	Boolean	
<code>clone</code>	Boolean	
<code>cluster</code>	Cluster	
<code>collapse_snapshots</code>	Boolean	
<code>comment</code>	String	Free text containing comments about this object.
<code>connectivity_timeout</code>	Integer	
<code>data_center</code>	DataCenter	

Name	Type	Summary
deploy_hosted_engine	Boolean	
description	String	A human-readable description in plain text.
details	GlusterVolumeProfileDetails	
discard_snapshots	Boolean	
disk	Disk	
disks	Disk[]	
exclusive	Boolean	
fault	Fault	
fence_type	String	
filter	Boolean	
fix_layout	Boolean	
force	Boolean	
grace_period	GracePeriod	
host	Host	
id	String	A unique identifier.
image	String	
import_as_template	Boolean	
is_attached	Boolean	
iscsi	IscsiDetails	
iscsi_targets	String[]	

Name	Type	Summary
job	Job	
logical_units	LogicalUnit[]	
maintenance_enabled	Boolean	
modified_bonds	HostNic[]	
modified_labels	NetworkLabel[]	
modified_network_attachments	NetworkAttachment[]	
name	String	A human-readable name in plain text.
option	Option	
pause	Boolean	
power_management	PowerManagement	
proxy_ticket	ProxyTicket	
reason	String	
reassign_bad_macs	Boolean	
remote_viewer_connection_file	String	
removed_bonds	HostNic[]	
removed_labels	NetworkLabel[]	
removed_network_attachments	NetworkAttachment[]	

Name	Type	Summary
resolution_type	String	
restore_memory	Boolean	
root_password	String	
snapshot	Snapshot	
ssh	Ssh	
status	String	
stop_gluster_service	Boolean	
storage_domain	StorageDomain	
storage_domains	StorageDomain[]	
succeeded	Boolean	
synchronized_network_attachments	NetworkAttachment[]	
template	Template	
ticket	Ticket	
undeploy_hosted_engine	Boolean	
use_cloud_init	Boolean	
use_sysprep	Boolean	
virtual_functions_configuration	HostNicVirtualFunctionsConfiguration	

Name	Type	Summary
vm	Vm	
vnic_profile_mappings	VnicProfileMapping[]	

7.3. AFFINITYGROUP STRUCT

An affinity group represents a group of virtual machines with a defined relationship.

Table 7.3. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
enforcing	Boolean	Specifies whether the affinity group uses hard or soft enforcement of the affinity applied to virtual machines that are members of that affinity group.
hosts_rule	AffinityRule	Specifies the affinity rule applied between virtual machines and hosts that are members of this affinity group.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
positive	Boolean	Specifies whether the affinity group applies positive affinity or negative affinity to virtual machines that are members of that affinity group.
vms_rule	AffinityRule	Specifies the affinity rule applied to virtual machines that are members of this affinity group.

7.3.1. enforcing

Specifies whether the affinity group uses hard or soft enforcement of the affinity applied to virtual machines that are members of that affinity group.

**WARNING**

Please note that this attribute has been deprecated since version 4.1 of the engine, and will be removed in the future. Use the **vms_rule** attribute from now on.

7.3.2. positive

Specifies whether the affinity group applies positive affinity or negative affinity to virtual machines that are members of that affinity group.

**WARNING**

Please note that this attribute has been deprecated since version 4.1 of the engine, and will be removed in the future. Use the **vms_rule** attribute from now on.

Table 7.4. Links summary

Name	Type	Summary
cluster	Cluster	A reference to the cluster to which the affinity group applies.
hosts	Host[]	A list of all hosts assigned to this affinity group.
vms	Vm[]	A list of all virtual machines assigned to this affinity group.

7.4. AFFINITYLABEL STRUCT

The affinity label can influence virtual machine scheduling. It is most frequently used to create a sub-cluster from the available hosts.

Table 7.5. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Name	Type	Summary
read_only	Boolean	The read_only property marks a label that can not be modified.

7.4.1. read_only

The **read_only** property marks a label that can not be modified. This is usually the case when listing internally-generated labels.

Table 7.6. Links summary

Name	Type	Summary
hosts	Host[]	A list of hosts that were labeled using this scheduling label.
vms	Vm[]	A list of virtual machines that were labeled using this scheduling label.

7.5. AFFINITYRULE STRUCT

Generic rule definition for affinity group. Each supported resource type (virtual machine, host) is controlled by a separate rule. This allows expressing of rules like: no affinity between defined virtual machines, but hard affinity between defined virtual machines and virtual hosts.

Table 7.7. Attributes summary

Name	Type	Summary
enabled	Boolean	Specifies whether the affinity group uses this rule or not.
enforcing	Boolean	Specifies whether the affinity group uses hard or soft enforcement of the affinity applied to the resources that are controlled by this rule.
positive	Boolean	Specifies whether the affinity group applies positive affinity or negative affinity to the resources that are controlled by this rule.

7.5.1. enabled

Specifies whether the affinity group uses this rule or not. This attribute is optional during creation and is considered to be **true** when it is not provided. In case this attribute is not provided to the update operation, it is considered to be **true** if AffinityGroup **positive** attribute is set as well. The backend **enabled** value will be preserved when both **enabled** and **positive** attributes are missing.

7.5.2. enforcing

Specifies whether the affinity group uses hard or soft enforcement of the affinity applied to the resources that are controlled by this rule. This argument is mandatory if the rule is enabled and is ignored when the rule is disabled.

7.5.3. positive

Specifies whether the affinity group applies positive affinity or negative affinity to the resources that are controlled by this rule. This argument is mandatory if the rule is enabled and is ignored when the rule is disabled.

7.6. AGENT STRUCT

Type representing a fence agent.

Table 7.8. Attributes summary

Name	Type	Summary
address	String	Fence agent address.
comment	String	Free text containing comments about this object.
concurrent	Boolean	Specifies whether the agent should be used concurrently or sequentially.
description	String	A human-readable description in plain text.
encrypt_options	Boolean	Specifies whether the options should be encrypted.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
options	Option[]	Fence agent options (comma-delimited list of key-value pairs).
order	Integer	The order of this agent if used with other agents.
password	String	Fence agent password.
port	Integer	Fence agent port.
type	String	Fence agent type.
username	String	Fence agent user name.

Table 7.9. Links summary

Name	Type	Summary
host	Host	Reference to the host service.

7.6.1. host

Reference to the host service. Each fence agent belongs to a single host.

7.7. AGENTCONFIGURATION STRUCT

Table 7.10. Attributes summary

Name	Type	Summary
address	String	
broker_type	MessageBrokerType	
network_mappings	String	
password	String	
port	Integer	
username	String	

7.8. API STRUCT

This type contains the information returned by the root service of the API.

To get that information send a request like this:

```
GET /ovirt-engine/api
```

The result will be like this:

```
<api>
  <link rel="hosts" href="/ovirt-engine/api/hosts"/>
  <link rel="vms" href="/ovirt-engine/api/vms"/>
  ...
  <product_info>
    <name>oVirt Engine</name>
    <vendor>ovirt.org</vendor>
    <version>
      <build>0</build>
      <full_version>4.1.0_master</full_version>
      <major>4</major>
```

```

    <minor>1</minor>
    <revision>0</revision>
  </version>
</product_info>
<special_objects>
  <link rel="templates/blank" href="...">
  <link rel="tags/root" href="...">
</special_objects>
<summary>
  <vms>
    <total>10</total>
    <active>3</active>
  </vms>
  <hosts>
    <total>2</total>
    <active>2</active>
  </hosts>
  <users>
    <total>8</total>
    <active>2</active>
  </users>
  <storage_domains>
    <total>2</total>
    <active>2</active>
  </storage_domains>
</summary>
<time>2016-12-12T12:22:25.866+01:00</time>
</api>

```

Table 7.11. Attributes summary

Name	Type	Summary
product_info	ProductInfo	Information about the product, such as its name, the name of the vendor, and the version.
special_objects	SpecialObjects	References to special objects, such as the blank template and the root of the hierarchy of tags.
summary	ApiSummary	A summary containing the total number of relevant objects, such as virtual machines, hosts, and storage domains.
time	Date	The date and time when this information was generated.

7.9. APISUMMARY STRUCT

A summary containing the total number of relevant objects, such as virtual machines, hosts, and storage domains.

Table 7.12. Attributes summary

Name	Type	Summary
hosts	ApiSummaryItem	The summary of hosts.
storage_domains	ApiSummaryItem	The summary of storage domains.
users	ApiSummaryItem	The summary of users.
vms	ApiSummaryItem	The summary of virtual machines.

7.10. APISUMMARYITEM STRUCT

This type contains an item of the API summary. Each item contains the total and active number of some kind of object.

Table 7.13. Attributes summary

Name	Type	Summary
active	Integer	The total number of active objects.
total	Integer	The total number of objects.

7.11. APPLICATION STRUCT

Table 7.14. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.15. Links summary

Name	Type	Summary
vm	Vm	

7.12. ARCHITECTURE ENUM

Table 7.16. Values summary

Name	Summary
ppc64	
undefined	
x86_64	

7.13. AUTHORIZEDKEY STRUCT

Table 7.17. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
key	String	
name	String	A human-readable name in plain text.

Table 7.18. Links summary

Name	Type	Summary
user	User	

7.14. AUTONUMASTATUS ENUM

Table 7.19. Values summary

Name	Summary
disable	
enable	
unknown	

7.15. BALANCE STRUCT

Table 7.20. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.21. Links summary

Name	Type	Summary
scheduling_p olicy	SchedulingPolicy	
scheduling_p olicy_unit	SchedulingPolicyU nit	

7.16. BIOS STRUCT

Table 7.22. Attributes summary

Name	Type	Summary
boot_menu	BootMenu	

7.17. BLOCKSTATISTIC STRUCT

Table 7.23. Attributes summary

Name	Type	Summary
statistics	Statistic[]	

7.18. BONDING STRUCT

Represents a network interfaces bond.

Table 7.24. Attributes summary

Name	Type	Summary
ad_partner_mac	Mac	The ad_partner_mac property of the partner bond in mode 4.
options	Option[]	A list of option elements for a bonded interface.
slaves	HostNic[]	A list of slave NICs for a bonded interface.

7.18.1. ad_partner_mac

The **ad_partner_mac** property of the partner bond in mode 4. Bond mode 4 is the 802.3ad standard, which is also called dynamic link aggregation. See [Wikipedia](#) and [Presentation](#) for more information. **ad_partner_mac** is the MAC address of the system (switch) at the other end of a bond. This parameter is read-only. Setting it will have no effect on the bond. It is retrieved from `/sys/class/net/bondX/bonding/ad_partner_mac` file on the system where the bond is located.

7.18.2. options

A list of option elements for a bonded interface. Each option contains property name and value attributes. Only required when adding bonded interfaces.

7.18.3. slaves

A list of slave NICs for a bonded interface. Only required when adding bonded interfaces.

Table 7.25. Links summary

Name	Type	Summary
active_slave	HostNic	The active_slave property of the bond in modes that support it (active-backup, balance-alb and balance-tlb).

7.18.4. active_slave

The **active_slave** property of the bond in modes that support it (active-backup, balance-alb and balance-tlb). See [Linux documentation](#) for further details. This parameter is read-only. Setting it will have no effect on the bond. It is retrieved from `/sys/class/net/bondX/bonding/active_slave` file on the system where the bond is located.

For example:

```
GET /ovirt-engine/api/hosts/123/nics/321
```

Will respond:

```
<host_nic href="/ovirt-engine/api/hosts/123/nics/321" id="321">
  ...
```

```

<bonding>
  <slaves>
    <host_nic href="/ovirt-engine/api/hosts/123/nics/456" id="456" />
    ...
  </slaves>
  <active_slave href="/ovirt-engine/api/hosts/123/nics/456" id="456" />
</bonding>
...
</host_nic>

```

7.19. BOOKMARK STRUCT

Table 7.26. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
value	String	

7.20. BOOT STRUCT

Table 7.27. Attributes summary

Name	Type	Summary
devices	BootDevice[]	

7.21. BOOTDEVICE ENUM

Table 7.28. Values summary

Name	Summary
cdrom	
hd	
network	

7.22. BOOTMENU STRUCT

Table 7.29. Attributes summary

Name	Type	Summary
<code>enabled</code>	Boolean	

7.23. BOOTPROTOCOL ENUM

Defines the options of the IP address assignment method to a NIC.

Table 7.30. Values summary

Name	Summary
<code>autoconf</code>	Stateless address auto-configuration.
<code>dhcp</code>	Dynamic host configuration protocol.
<code>none</code>	No address configuration.
<code>static</code>	Statically-defined address, mask and gateway.

7.23.1. autoconf

Stateless address auto-configuration.

The mechanism is defined by [RFC 4862](#). Please refer to [this wikipedia article](#) for more information.



NOTE

The value is valid for IPv6 addresses only.

7.23.2. dhcp

Dynamic host configuration protocol.

Please refer to [this wikipedia article](#) for more information.

7.24. BRICKPROFILEDETAIL STRUCT

Table 7.31. Attributes summary

Name	Type	Summary
<code>profile_details</code>	ProfileDetail[]	

Table 7.32. Links summary

Name	Type	Summary
brick	GlusterBrick	

7.25. CDROM STRUCT

Table 7.33. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
file	File	
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.34. Links summary

Name	Type	Summary
instance_type	InstanceType	Optionally references to an instance type the device is used by.
template	Template	Optionally references to a template the device is used by.
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

7.25.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk may be used simultaneously by two or more virtual machines.

7.26. CERTIFICATE STRUCT

Table 7.35. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.

Name	Type	Summary
content	String	
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
organization	String	
subject	String	

7.27. CLOUDINIT STRUCT

Table 7.36. Attributes summary

Name	Type	Summary
authorized_keys	AuthorizedKey[]	
files	File[]	
host	Host	
network_configuration	NetworkConfiguration	
regenerate_ssh_keys	Boolean	
timezone	String	
users	User[]	

7.28. CLUSTER STRUCT

Type representation of a cluster.

A JSON representation of a cluster

```
{
  "cluster" : [ {
    "ballooning_enabled" : "false",
    "cpu" : {
      "architecture" : "x86_64",
```



```

    "type" : "Intel SandyBridge Family"
  },
  "custom_scheduling_policy_properties" : {
    "property" : [ {
      "name" : "HighUtilization",
      "value" : "80"
    }, {
      "name" : "CpuOverCommitDurationMinutes",
      "value" : "2"
    } ]
  },
  "error_handling" : {
    "on_error" : "migrate"
  },
  "fencing_policy" : {
    "enabled" : "true",
    "skip_if_connectivity_broken" : {
      "enabled" : "false",
      "threshold" : "50"
    },
    "skip_if_gluster_bricks_up" : "false",
    "skip_if_gluster_quorum_not_met" : "false",
    "skip_if_sd_active" : {
      "enabled" : "false"
    }
  },
  "gluster_service" : "false",
  "ha_reservation" : "false",
  "ksm" : {
    "enabled" : "true",
    "merge_across_nodes" : "true"
  },
  "maintenance_reason_required" : "false",
  "memory_policy" : {
    "over_commit" : {
      "percent" : "100"
    },
    "transparent_hugepages" : {
      "enabled" : "true"
    }
  },
  "migration" : {
    "auto_converge" : "inherit",
    "bandwidth" : {
      "assignment_method" : "auto"
    },
    "compressed" : "inherit",
    "policy" : {
      "id" : "00000000-0000-0000-0000-000000000000"
    }
  },
  "optional_reason" : "false",
  "required_rng_sources" : {
    "required_rng_source" : [ "random" ]
  },
  "switch_type" : "legacy",

```

```

"threads_as_cores" : "false",
"trusted_service" : "false",
"tunnel_migration" : "false",
"version" : {
  "major" : "4",
  "minor" : "1"
},
"virt_service" : "true",
"data_center" : {
  "href" : "/ovirt-engine/api/datacenters/123",
  "id" : "123"
},
"mac_pool" : {
  "href" : "/ovirt-engine/api/macpools/456",
  "id" : "456"
},
"scheduling_policy" : {
  "href" : "/ovirt-engine/api/schedulingpolicies/789",
  "id" : "789"
},
"actions" : {
  "link" : [ {
    "href" : "/ovirt-engine/api/clusters/234/resetemulatedmachine",
    "rel" : "resetemulatedmachine"
  } ]
},
"name" : "Default",
"description" : "The default server cluster",
"href" : "/ovirt-engine/api/clusters/234",
"id" : "234",
"link" : [ {
  "href" : "/ovirt-engine/api/clusters/234/permissions",
  "rel" : "permissions"
}, {
  "href" : "/ovirt-engine/api/clusters/234/cpuprofiles",
  "rel" : "cpuprofiles"
}, {
  "href" : "/ovirt-engine/api/clusters/234/networkfilters",
  "rel" : "networkfilters"
}, {
  "href" : "/ovirt-engine/api/clusters/234/networks",
  "rel" : "networks"
}, {
  "href" : "/ovirt-engine/api/clusters/234/affinitygroups",
  "rel" : "affinitygroups"
}, {
  "href" : "/ovirt-engine/api/clusters/234/glusterhooks",
  "rel" : "glusterhooks"
}, {
  "href" : "/ovirt-engine/api/clusters/234/glustervolumes",
  "rel" : "glustervolumes"
} ]
} ]
}

```

Table 7.37. Attributes summary

Name	Type	Summary
ballooning_enabled	Boolean	
comment	String	Free text containing comments about this object.
cpu	Cpu	
custom_scheduling_policy_properties	Property[]	Custom scheduling policy properties of the cluster.
description	String	A human-readable description in plain text.
display	Display	
error_handling	ErrorHandling	
fencing_policy	FencingPolicy	Custom fencing policy can be defined for a cluster.
gluster_service	Boolean	
gluster_tuned_profile	String	The name of the https://fedorahosted .
ha_reservation	Boolean	
id	String	A unique identifier.
ksm	Ksm	
maintenance_reason_required	Boolean	
memory_policy	MemoryPolicy	
migration	MigrationOptions	
name	String	A human-readable name in plain text.

Name	Type	Summary
optional_reason	Boolean	
required_rng_sources	RngSource[]	Set of random number generator (RNG) sources required from each host in the cluster.
serial_number	SerialNumber	
supported_versions	Version[]	
switch_type	SwitchType	Type of switch to be used by all networks in given cluster.
threads_as_cores	Boolean	
trusted_service	Boolean	
tunnel_migration	Boolean	
version	Version	The compatibility version of the cluster.
virt_service	Boolean	

7.28.1. custom_scheduling_policy_properties

Custom scheduling policy properties of the cluster. These optional properties override the properties of the scheduling policy specified by the **scheduling_policy** link, and apply only for this specific cluster.

For example, to update the custom properties of the cluster, send a request:

```
PUT /ovirt-engine/api/clusters/123
```

With a request body:

```
<cluster>
  <custom_scheduling_policy_properties>
    <property>
      <name>HighUtilization</name>
      <value>70</value>
    </property>
  </custom_scheduling_policy_properties>
</cluster>
```

Update operations using the `custom_scheduling_policy_properties` attribute will not update the properties of the scheduling policy specified by the `scheduling_policy` link, they will only be reflected on this specific cluster.

7.28.2. fencing_policy

Custom fencing policy can be defined for a cluster.

Here is an example:

```
PUT /ovirt-engine/api/cluster/123
```

With request body:

```
<cluster>
  <fencing_policy>
    <enabled>true</enabled>
    <skip_if_sd_active>
      <enabled>>false</enabled>
    </skip_if_sd_active>
    <skip_if_connectivity_broken>
      <enabled>>false</enabled>
      <threshold>50</threshold>
    </skip_if_connectivity_broken>
  </fencing_policy>
</cluster>
```

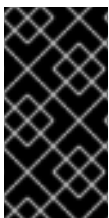
7.28.3. gluster_tuned_profile

The name of the `tuned` profile to set on all the hosts in the cluster. This is not mandatory and relevant only for clusters with gluster service.

7.28.4. required_rng_sources

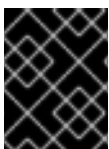
Set of random number generator (RNG) sources required from each host in the cluster.

When read, it returns the implicit `urandom` (for cluster version 4.1 and higher) or `random` (for cluster version 4.0 and lower) plus additional selected RNG sources. When written, the implicit `urandom` and `random` RNG sources cannot be removed.



IMPORTANT

Before version 4.1 of the engine, the set of required random number generators was completely controllable by the administrator; any source could be added or removed, including the `random` source. But starting with version 4.1, the `urandom` and `random` sources will always be part of the set, and can't be removed.



IMPORTANT

Engine version 4.1 introduces a new RNG source `urandom` that replaces `random` RNG source in clusters with compatibility version 4.1 or higher.

7.28.5. version

The compatibility version of the cluster.

All hosts in this cluster must support at least this compatibility version.

For example:

```
GET /ovirt-engine/api/clusters/123
```

Will respond:

```
<cluster>
  ...
  <version>
    <major>4</major>
    <minor>0</minor>
  </version>
  ...
</cluster>
```

To update the compatibility version, use:

```
PUT /ovirt-engine/api/clusters/123
```

With a request body:

```
<cluster>
  <version>
    <major>4</major>
    <minor>1</minor>
  </version>
</cluster>
```

In order to update the cluster compatibility version, all hosts in the cluster must support the new compatibility version.

Table 7.38. Links summary

Name	Type	Summary
affinity_groups	AffinityGroup[]	
cpu_profiles	CpuProfile[]	
data_center	DataCenter	
gluster_hooks	GlusterHook[]	

Name	Type	Summary
gluster_volumes	GlusterVolume[]	
mac_pool	MacPool	A reference to the MAC pool used by this cluster.
management_network	Network	
network_filters	NetworkFilter[]	
networks	Network[]	
permissions	Permission[]	
scheduling_policy	SchedulingPolicy	Reference to the default scheduling policy used by this cluster.

7.28.6. scheduling_policy

Reference to the default scheduling policy used by this cluster.



NOTE

The scheduling policy properties are taken by default from the referenced scheduling policy, but they are overridden by the properties specified in the **custom_scheduling_policy_properties** attribute for this cluster.

7.29. CLUSTERLEVEL STRUCT

Describes the capabilities supported by a specific cluster level.

Table 7.39. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
cpu_types	CpuType[]	The CPU types supported by this cluster level.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Name	Type	Summary
permits	Permit[]	The permits supported by this cluster level.

7.30. CONFIGURATION STRUCT

Table 7.40. Attributes summary

Name	Type	Summary
data	String	The document describing the virtual machine.
type	ConfigurationType	

7.30.1. data

The document describing the virtual machine.

Example of the OVF document:

```
<?xml version='1.0' encoding='UTF-8'?>
<ovf:Envelope xmlns:ovf="http://schemas.dmtf.org/ovf/envelope/1/"
  xmlns:rasd="http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/CIM_ResourceAllocationSettingData"
  xmlns:vssd="http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/CIM_VirtualSystemSettingData"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  ovf:version="3.5.0.0">
  <References/>
  <Section xsi:type="ovf:NetworkSection_Type">
    <Info>List of networks</Info>
    <Network ovf:name="Network 1"/>
  </Section>
  <Section xsi:type="ovf:DiskSection_Type">
    <Info>List of Virtual Disks</Info>
  </Section>
  <Content ovf:id="out" xsi:type="ovf:VirtualSystem_Type">
    <CreationDate>2014/12/03 04:25:45</CreationDate>
    <ExportDate>2015/02/09 14:12:24</ExportDate>
    <DeleteProtected>>false</DeleteProtected>
    <SsoMethod>guest_agent</SsoMethod>
    <IsSmartcardEnabled>>false</IsSmartcardEnabled>
    <TimeZone>Etc/GMT</TimeZone>
    <default_boot_sequence>0</default_boot_sequence>
    <Generation>1</Generation>
    <VmType>1</VmType>
    <MinAllocatedMem>1024</MinAllocatedMem>
    <IsStateless>>false</IsStateless>
    <IsRunAndPause>>false</IsRunAndPause>
    <AutoStartup>>false</AutoStartup>
    <Priority>1</Priority>
    <CreatedByUserId>fdfc627c-d875-11e0-90f0-
```



```

83df133b58cc</CreatedByUserId>
  <IsBootMenuEnabled>>false</IsBootMenuEnabled>
  <IsSpiceFileTransferEnabled>>true</IsSpiceFileTransferEnabled>
  <IsSpiceCopyPasteEnabled>>true</IsSpiceCopyPasteEnabled>
  <Name>VM_export</Name>
  <TemplateId>00000000-0000-0000-0000-000000000000</TemplateId>
  <TemplateName>Blank</TemplateName>
  <IsInitilized>>false</IsInitilized>
  <Origin>3</Origin>
  <DefaultDisplayType>1</DefaultDisplayType>
  <TrustedService>>false</TrustedService>
  <OriginalTemplateId>00000000-0000-0000-0000-
000000000000</OriginalTemplateId>
  <OriginalTemplateName>Blank</OriginalTemplateName>
  <UseLatestVersion>>false</UseLatestVersion>
  <Section ovf:id="70b4d9a7-4f73-4def-89ca-24fc5f60e01a"
    ovf:required="false"
    xsi:type="ovf:OperatingSystemSection_Type">
    <Info>Guest Operating System</Info>
    <Description>other</Description>
  </Section>
  <Section xsi:type="ovf:VirtualHardwareSection_Type">
    <Info>1 CPU, 1024 Memory</Info>
    <System>
      <vssd:VirtualSystemType>ENGINE 3.5.0.0</vssd:VirtualSystemType>
    </System>
    <Item>
      <rasd:Caption>1 virtual cpu</rasd:Caption>
      <rasd:Description>Number of virtual CPU</rasd:Description>
      <rasd:InstanceId>1</rasd:InstanceId>
      <rasd:ResourceType>3</rasd:ResourceType>
      <rasd:num_of_sockets>1</rasd:num_of_sockets>
      <rasd:cpu_per_socket>1</rasd:cpu_per_socket>
    </Item>
    <Item>
      <rasd:Caption>1024 MB of memory</rasd:Caption>
      <rasd:Description>Memory Size</rasd:Description>
      <rasd:InstanceId>2</rasd:InstanceId>
      <rasd:ResourceType>4</rasd:ResourceType>
      <rasd:AllocationUnits>MegaBytes</rasd:AllocationUnits>
      <rasd:VirtualQuantity>1024</rasd:VirtualQuantity>
    </Item>
    <Item>
      <rasd:Caption>USB Controller</rasd:Caption>
      <rasd:InstanceId>3</rasd:InstanceId>
      <rasd:ResourceType>23</rasd:ResourceType>
      <rasd:UsbPolicy>DISABLED</rasd:UsbPolicy>
    </Item>
  </Section>
</Content>
</ovf:Envelope>

```

7.31. CONFIGURATIONTYPE ENUM

Table 7.41. Values summary

Name	Summary
ovf	

7.32. CONSOLE STRUCT

Table 7.42. Attributes summary

Name	Type	Summary
enabled	Boolean	

7.33. CORE STRUCT

Table 7.43. Attributes summary

Name	Type	Summary
index	Integer	
socket	Integer	

7.34. CPU STRUCT

Table 7.44. Attributes summary

Name	Type	Summary
architecture	Architecture	
cores	Core[]	
cpu_tune	CpuTune	
level	Integer	
mode	CpuMode	
name	String	
speed	Decimal	
topology	CpuTopology	
type	String	

7.35. CPUMODE ENUM

Table 7.45. Values summary

Name	Summary
<code>custom</code>	
<code>host_model</code>	
<code>host_passthrough</code>	

7.36. CPUPROFILE STRUCT

Table 7.46. Attributes summary

Name	Type	Summary
<code>comment</code>	String	Free text containing comments about this object.
<code>description</code>	String	A human-readable description in plain text.
<code>id</code>	String	A unique identifier.
<code>name</code>	String	A human-readable name in plain text.

Table 7.47. Links summary

Name	Type	Summary
<code>cluster</code>	Cluster	
<code>permissions</code>	Permission[]	
<code>qos</code>	Qos	

7.37. CPUTOPOLOGY STRUCT

Table 7.48. Attributes summary

Name	Type	Summary
<code>cores</code>	Integer	
<code>sockets</code>	Integer	

Name	Type	Summary
threads	Integer	

7.38. CPUTUNE STRUCT

Table 7.49. Attributes summary

Name	Type	Summary
vcpu_pins	VcpuPin[]	

7.39. CPUTYPE STRUCT

Describes a supported CPU type.

Table 7.50. Attributes summary

Name	Type	Summary
architecture	Architecture	The architecture of the CPU.
level	Integer	The level of the CPU type.
name	String	The name of the CPU type, for example Intel Conroe Family .

7.40. CREATIONSTATUS ENUM

Table 7.51. Values summary

Name	Summary
complete	
failed	
in_progress	
pending	

7.41. CUSTOMPROPERTY STRUCT

Table 7.52. Attributes summary

Name	Type	Summary
name	String	
regexp	String	
value	String	

7.42. DATACENTER STRUCT

Table 7.53. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
local	Boolean	
name	String	A human-readable name in plain text.
quota_mode	QuotaModeType	
status	DataCenterStatus	
storage_format	StorageFormat	
supported_versions	Version[]	
version	Version	The compatibility version of the data center.

7.42.1. version

The compatibility version of the data center.

All clusters in this data center must already be set to at least this compatibility version.

For example:

```
GET /ovirt-engine/api/datacenters/123
```

Will respond:

```

<data_center>
  ...
  <version>
    <major>4</major>
    <minor>0</minor>
  </version>
  ...
</data_center>

```

To update the compatibility version, use:

```
PUT /ovirt-engine/api/datacenters/123
```

With a request body:

```

<data_center>
  <version>
    <major>4</major>
    <minor>1</minor>
  </version>
</data_center>

```

Table 7.54. Links summary

Name	Type	Summary
clusters	Cluster[]	Reference to clusters inside this data center.
iscsi_bonds	IscsiBond[]	Reference to iSCSI bonds used by this data center.
mac_pool	MacPool	Reference to the MAC pool used by this data center.
networks	Network[]	Reference to networks attached to this data center.
permissions	Permission[]	Reference to permissions assigned to this data center.
qoss	Qos[]	Reference to quality of service used by this data center.
quotas	Quota[]	Reference to quotas assigned to this data center.
storage_domains	StorageDomain[]	Reference to storage domains attached to this data center.

7.43. DATACENTERSTATUS ENUM

Table 7.55. Values summary

Name	Summary
contend	
maintenance	
not_operational	
problematic	
uninitialized	
up	

7.44. DEVICE STRUCT

A device wraps links to potential parents of a device.

Table 7.56. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.57. Links summary

Name	Type	Summary
instance_type	InstanceType	Optionally references to an instance type the device is used by.
template	Template	Optionally references to a template the device is used by.
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

7.44.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk may be used simultaneously by two or more virtual machines.

7.45. DISK STRUCT

Represents a virtual disk device.

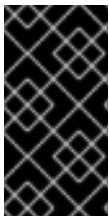
Table 7.58. Attributes summary

Name	Type	Summary
active	Boolean	Indicates if the disk is visible to the virtual machine.
actual_size	Integer	The actual size of the disk, in bytes.
alias	String	
bootable	Boolean	Indicates if the disk is marked as bootable.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
format	DiskFormat	The underlying storage format.
id	String	A unique identifier.
image_id	String	
initial_size	Integer	The initial size of a sparse image disk created on block storage, in bytes.
interface	DiskInterface	The type of interface driver used to connect the disk device to the virtual machine.
logical_name	String	
lun_storage	HostStorage	
name	String	A human-readable name in plain text.
propagate_errors	Boolean	Indicates if disk errors should not cause virtual machine to be paused and, instead, disk errors should be propagated to the the guest operating system.
provisioned_size	Integer	The virtual size of the disk, in bytes.
qcow_version	QcowVersion	The underlying QCOW version of a QCOW volume.

Name	Type	Summary
read_only	Boolean	Indicates if the disk is in read-only mode.
sgio	ScsiGenericIO	
shareable	Boolean	Indicates if the disk can be attached to multiple virtual machines.
sparse	Boolean	Indicates if the physical storage for the disk should not be preallocated.
status	DiskStatus	The status of the disk device.
storage_type	DiskStorageType	
uses_scsi_reservation	Boolean	
wipe_after_delete	Boolean	Indicates if the disk's blocks will be read back as zeros after it is deleted: - On block storage, the disk will be zeroed and only then deleted.

7.45.1. active

Indicates if the disk is visible to the virtual machine.



IMPORTANT

When adding a disk attachment to a virtual machine, the server accepts requests that don't contain this attribute, but the effect is then undefined. In some cases the disk will be automatically activated and in other cases it won't. To avoid issues it is strongly recommended to always include the this attribute with the desired value.

7.45.2. actual_size

The actual size of the disk, in bytes.

The actual size is the number of bytes actually used by the disk, and it will be smaller than the provisioned size for disks that use the **cow** format.

7.45.3. bootable

Indicates if the disk is marked as bootable.



IMPORTANT

This attribute only makes sense for disks that are actually connected to virtual machines, and in version 4 of the API it has been moved to the [DiskAttachment](#) type. It is preserved here only for backwards compatibility, and it will be removed in the future.

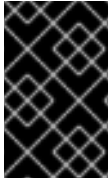
7.45.4. initial_size

The initial size of a sparse image disk created on block storage, in bytes.

The initial size is the number of bytes a sparse disk is initially allocated with when created on block storage. The initial size will be smaller than the provisioned size. If not specified the default initial size used by the system will be allocated.

7.45.5. interface

The type of interface driver used to connect the disk device to the virtual machine.



IMPORTANT

This attribute only makes sense for disks that are actually connected to virtual machines, and in version 4 of the API it has been moved to the [DiskAttachment](#) type. It is preserved here only for backwards compatibility, and it will be removed in the future.

7.45.6. provisioned_size

The virtual size of the disk, in bytes.

This attribute is mandatory when creating a new disk.

7.45.7. qcow_version

The underlying QCOW version of a QCOW volume. The QCOW version specifies to the qemu which qemu version the volume supports. This field can be updated using the update API and will be reported only for QCOW volumes, it is determined by the storage domain's version which the disk is created on. Storage domains with version lower than V4 support QCOW2 volumes, while V4 storage domains also support QCOW2v3. For more information about features of the different QCOW versions, see [here](#).

7.45.8. shareable

Indicates if the disk can be attached to multiple virtual machines.



IMPORTANT

When a disk is attached to multiple virtual machines it is the responsibility of the guest operating systems of those virtual machines to coordinate access to it, to avoid corruption of the data, for example using a shared file system like [GlusterFS](#) or [GFS](#).

7.45.9. wipe_after_delete

Indicates if the disk's blocks will be read back as zeros after it is deleted:

- On block storage, the disk will be zeroed and only then deleted.
- On file storage, since the file system already guarantees that previously removed blocks are read back as zeros, the disk will be deleted immediately.

Table 7.59. Links summary

Name	Type	Summary
disk_profile	DiskProfile	
instance_type	InstanceType	Optionally references to an instance type the device is used by.
openstack_volume_type	OpenStackVolumeType	
permissions	Permission[]	
quota	Quota	
snapshot	Snapshot	
statistics	Statistic[]	Statistics exposed by the disk.
storage_domain	StorageDomain	
storage_domains	StorageDomain[]	The storage domains associated with this disk.
template	Template	Optionally references to a template the device is used by.
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

7.45.10. statistics

Statistics exposed by the disk. For example:

```
<statistics>
  <statistic href="/ovirt-engine/api/disks/123/statistics/456" id="456">
    <name>data.current.read</name>
    <description>Read data rate</description>
    <kind>gauge</kind>
    <type>decimal</type>
    <unit>bytes_per_second</unit>
    <values>
      <value>
        <datum>1052</datum>
      </value>
    </values>
    <disk href="/ovirt-engine/api/disks/123" id="123"/>
  </statistic>
  ...
</statistics>
```

These statistics aren't directly included when the disk is retrieved, only a link. To obtain the statistics follow that link:

```
GET /ovirt-engine/api/disks/123/statistics
```

7.45.11. storage_domains

The storage domains associated with this disk.



NOTE

Only required when the first disk is being added to a virtual machine that was not itself created from a template.

7.45.12. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk may be used simultaneously by two or more virtual machines.

7.46. DISKATTACHMENT STRUCT

Describes how a disk is attached to a virtual machine.

Table 7.60. Attributes summary

Name	Type	Summary
active	Boolean	Defines whether the disk is active in the virtual machine it's attached to.
bootable	Boolean	Defines whether the disk is bootable.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
interface	DiskInterface	The type of interface driver used to connect the disk device to the virtual machine.
logical_name	String	The logical name of the virtual machine's disk, as seen from inside the virtual machine.
name	String	A human-readable name in plain text.
pass_discard	Boolean	Defines whether the virtual machine passes discard commands to the storage.

Name	Type	Summary
read_only	Boolean	Indicates whether the disk is connected to the virtual machine as read only.
uses_scsi_reservation	Boolean	Defines whether SCSI reservation is enabled for this disk.

7.46.1. active

Defines whether the disk is active in the virtual machine it's attached to.

A disk attached to a virtual machine in an active status is connected to the virtual machine at run time and can be used.

7.46.2. logical_name

The logical name of the virtual machine's disk, as seen from inside the virtual machine.

The logical name of a disk is reported only when the guest agent is installed and running inside the virtual machine.

For example, if the guest operating system is Linux and the disk is connected via a VirtIO interface, the logical name will be reported as **/dev/vda**:

```
<disk_attachment>
...
<logical_name>/dev/vda</logical_name>
</disk_attachment>
```

If the guest operating system is Windows, the logical name will be reported as **\\.\PHYSICALDRIVE0**.

7.46.3. read_only

Indicates whether the disk is connected to the virtual machine as read only.

When adding a new disk attachment the default value is **false**.

```
<disk_attachment>
...
<read_only>>true</read_only>
</disk_attachment>
```

7.46.4. uses_scsi_reservation

Defines whether SCSI reservation is enabled for this disk.

Virtual machines with VIRTIO-SCSI passthrough enabled can set persistent SCSI reservations on disks. If they set persistent SCSI reservations, those virtual machines cannot be migrated to a different host because they would lose access to the disk, because SCSI reservations are specific to SCSI initiators,

and therefore hosts. This scenario cannot be automatically detected. To avoid migrating these virtual machines, the user can set this attribute to **true**, to indicate the virtual machine is using SCSI reservations.

Table 7.61. Links summary

Name	Type	Summary
disk	Disk	The reference to the disk.
template	Template	The reference to the template.
vm	Vm	The reference to the virtual machine.

7.47. DISKFORMAT ENUM

The underlying storage format of disks.

Table 7.62. Values summary

Name	Summary
cow	The <i>Copy On Write</i> format allows snapshots, with a small performance overhead.
raw	The raw format does not allow snapshots, but offers improved performance.

7.48. DISKINTERFACE ENUM

Table 7.63. Values summary

Name	Summary
ide	
spapr_vscsi	
virtio	
virtio_scsi	

7.49. DISKPROFILE STRUCT

Table 7.64. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.

Name	Type	Summary
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.65. Links summary

Name	Type	Summary
permissions	Permission[]	
qos	Qos	
storage_domain	StorageDomain	

7.50. DISKSNAPSHOT STRUCT

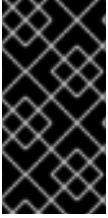
Table 7.66. Attributes summary

Name	Type	Summary
active	Boolean	Indicates if the disk is visible to the virtual machine.
actual_size	Integer	The actual size of the disk, in bytes.
alias	String	
bootable	Boolean	Indicates if the disk is marked as bootable.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
format	DiskFormat	The underlying storage format.
id	String	A unique identifier.
image_id	String	
initial_size	Integer	The initial size of a sparse image disk created on block storage, in bytes.

Name	Type	Summary
interface	DiskInterface	The type of interface driver used to connect the disk device to the virtual machine.
logical_name	String	
lun_storage	HostStorage	
name	String	A human-readable name in plain text.
propagate_errors	Boolean	Indicates if disk errors should not cause virtual machine to be paused and, instead, disk errors should be propagated to the the guest operating system.
provisioned_size	Integer	The virtual size of the disk, in bytes.
qcow_version	QcowVersion	The underlying QCOW version of a QCOW volume.
read_only	Boolean	Indicates if the disk is in read-only mode.
sgio	ScsiGenericIO	
shareable	Boolean	Indicates if the disk can be attached to multiple virtual machines.
sparse	Boolean	Indicates if the physical storage for the disk should not be preallocated.
status	DiskStatus	The status of the disk device.
storage_type	DiskStorageType	
uses_scsi_reservation	Boolean	
wipe_after_delete	Boolean	Indicates if the disk's blocks will be read back as zeros after it is deleted: - On block storage, the disk will be zeroed and only then deleted.

7.50.1. active

Indicates if the disk is visible to the virtual machine.



IMPORTANT

When adding a disk attachment to a virtual machine, the server accepts requests that don't contain this attribute, but the effect is then undefined. In some cases the disk will be automatically activated and in other cases it won't. To avoid issues it is strongly recommended to always include the this attribute with the desired value.

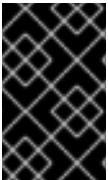
7.50.2. `actual_size`

The actual size of the disk, in bytes.

The actual size is the number of bytes actually used by the disk, and it will be smaller than the provisioned size for disks that use the `cow` format.

7.50.3. `bootable`

Indicates if the disk is marked as bootable.



IMPORTANT

This attribute only makes sense for disks that are actually connected to virtual machines, and in version 4 of the API it has been moved to the [DiskAttachment](#) type. It is preserved here only for backwards compatibility, and it will be removed in the future.

7.50.4. `initial_size`

The initial size of a sparse image disk created on block storage, in bytes.

The initial size is the number of bytes a sparse disk is initially allocated with when created on block storage. The initial size will be smaller than the provisioned size. If not specified the default initial size used by the system will be allocated.

7.50.5. `interface`

The type of interface driver used to connect the disk device to the virtual machine.



IMPORTANT

This attribute only makes sense for disks that are actually connected to virtual machines, and in version 4 of the API it has been moved to the [DiskAttachment](#) type. It is preserved here only for backwards compatibility, and it will be removed in the future.

7.50.6. `provisioned_size`

The virtual size of the disk, in bytes.

This attribute is mandatory when creating a new disk.

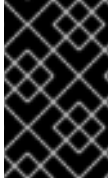
7.50.7. `qcow_version`

The underlying QCOW version of a QCOW volume. The QCOW version specifies to the qemu which qemu version the volume supports. This field can be updated using the update API and will be reported only for QCOW volumes, it is determined by the storage domain's version which the disk is created on.

Storage domains with version lower than V4 support QCOW2 volumes, while V4 storage domains also support QCOW2v3. For more information about features of the different QCOW versions, see [here](#).

7.50.8. shareable

Indicates if the disk can be attached to multiple virtual machines.



IMPORTANT

When a disk is attached to multiple virtual machines it is the responsibility of the guest operating systems of those virtual machines to coordinate access to it, to avoid corruption of the data, for example using a shared file system like [GlusterFS](#) or [GFS](#).

7.50.9. wipe_after_delete

Indicates if the disk's blocks will be read back as zeros after it is deleted:

- On block storage, the disk will be zeroed and only then deleted.
- On file storage, since the file system already guarantees that previously removed blocks are read back as zeros, the disk will be deleted immediately.

Table 7.67. Links summary

Name	Type	Summary
disk	Disk	
disk_profile	DiskProfile	
instance_type	InstanceType	Optionally references to an instance type the device is used by.
openstack_volume_type	OpenStackVolumeType	
permissions	Permission[]	
quota	Quota	
snapshot	Snapshot	
statistics	Statistic[]	Statistics exposed by the disk.
storage_domain	StorageDomain	
storage_domains	StorageDomain[]	The storage domains associated with this disk.
template	Template	Optionally references to a template the device is used by.

Name	Type	Summary
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

7.50.10. statistics

Statistics exposed by the disk. For example:

```
<statistics>
  <statistic href="/ovirt-engine/api/disks/123/statistics/456" id="456">
    <name>data.current.read</name>
    <description>Read data rate</description>
    <kind>gauge</kind>
    <type>decimal</type>
    <unit>bytes_per_second</unit>
    <values>
      <value>
        <datum>1052</datum>
      </value>
    </values>
    <disk href="/ovirt-engine/api/disks/123" id="123"/>
  </statistic>
  ...
</statistics>
```

These statistics aren't directly included when the disk is retrieved, only a link. To obtain the statistics follow that link:

```
GET /ovirt-engine/api/disks/123/statistics
```

7.50.11. storage_domains

The storage domains associated with this disk.



NOTE

Only required when the first disk is being added to a virtual machine that was not itself created from a template.

7.50.12. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk may be used simultaneously by two or more virtual machines.

7.51. DISKSTATUS ENUM

Table 7.68. Values summary

Name	Summary
illegal	
locked	
ok	

7.52. DISKSTORAGETYPE ENUM

Table 7.69. Values summary

Name	Summary
cinder	
image	
lun	

7.53. DISKTYPE ENUM

Table 7.70. Values summary

Name	Summary
data	
system	

7.54. DISPLAY STRUCT

Table 7.71. Attributes summary

Name	Type	Summary
address	String	
allow_override	Boolean	
certificate	Certificate	
copy_paste_enabled	Boolean	

Name	Type	Summary
disconnect_action	String	
file_transferrer_enabled	Boolean	
keyboard_layout	String	
monitors	Integer	
port	Integer	
proxy	String	
secure_port	Integer	
single_qxl_pci	Boolean	
smartcard_enabled	Boolean	
type	DisplayType	

7.55. DISPLAYTYPE ENUM

Table 7.72. Values summary

Name	Summary
spice	
vnc	

7.56. DNS STRUCT

Represents the DNS resolver configuration.

Table 7.73. Attributes summary

Name	Type	Summary
search_domains	Host[]	Array of hosts serving as search domains.

Name	Type	Summary
servers	Host[]	Array of hosts serving as DNS servers.

7.57. DNSRESOLVERCONFIGURATION STRUCT

Represents the DNS resolver configuration.

Table 7.74. Attributes summary

Name	Type	Summary
name_servers	String[]	Array of addresses of name servers.

7.57.1. name_servers

Array of addresses of name servers. Either IPv4 or IPv6 addresses may be specified.

7.58. DOMAIN STRUCT

This type represents a directory service domain.

Table 7.75. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
user	User	

Table 7.76. Links summary

Name	Type	Summary
groups	Group[]	A reference to all groups in the directory service.
users	User[]	A reference to a list of all users in the directory service.

7.58.1. users

A reference to a list of all users in the directory service. This information is used to add new users to the Red Hat Virtualization environment.

7.59. ENTITYEXTERNALSTATUS ENUM

Type representing an external entity status.

Table 7.77. Values summary

Name	Summary
error	The external entity status is erroneous.
failure	The external entity has an issue that causes failures.
info	There external entity status is okay but with some information that might be relevant.
ok	The external entity status is okay.
warning	The external entity status is okay but with an issue that might require attention.

7.59.1. error

The external entity status is erroneous. This might require a moderate attention.

7.59.2. failure

The external entity has an issue that causes failures. This might require immediate attention.

7.60. ENTITYPROFILEDETAIL STRUCT

Table 7.78. Attributes summary

Name	Type	Summary
profile_details	ProfileDetail[]	

7.61. ERRORHANDLING STRUCT

Table 7.79. Attributes summary

Name	Type	Summary
on_error	MigrateOnError	

7.62. EVENT STRUCT

Type representing an event.

Table 7.80. Attributes summary

Name	Type	Summary
code	Integer	The event code.
comment	String	Free text containing comments about this object.
correlation_id	String	The event correlation identifier.
custom_data	String	Free text representing custom event data.
custom_id	Integer	A custom event identifier.
description	String	A human-readable description in plain text.
flood_rate	Integer	Defines the flood rate.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
origin	String	Free text identifying the origin of the event.
severity	LogSeverity	The event severity.
time	Date	The event time.

7.62.1. correlation_id

The event correlation identifier. Used in order to correlate several events together.

7.62.2. flood_rate

Defines the flood rate. This prevents flooding in case an event appeared more than once in the defined rate. Defaults is 30 seconds.

Table 7.81. Links summary

Name	Type	Summary
cluster	Cluster	Reference to the cluster service.
data_center	DataCenter	Reference to the data center service.
host	Host	Reference to the host service.

Name	Type	Summary
storage_domain	StorageDomain	Reference to the storage domain service.
template	Template	Reference to the template service.
user	User	Reference to the user service.
vm	Vm	Reference to the virtual machine service.

7.62.3. cluster

Reference to the cluster service. Event can be associated with a cluster.

7.62.4. data_center

Reference to the data center service. Event can be associated with a data center.

7.62.5. host

Reference to the host service. Event can be associated with a host.

7.62.6. storage_domain

Reference to the storage domain service. Event can be associated with a storage domain.

7.62.7. template

Reference to the template service. Event can be associated with a template.

7.62.8. user

Reference to the user service. Event can be associated with a user.

7.62.9. vm

Reference to the virtual machine service. Event can be associated with a virtual machine.

7.63. EXTERNALCOMPUTERESOURCE STRUCT

Table 7.82. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.

Name	Type	Summary
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
provider	String	
url	String	
user	String	

Table 7.83. Links summary

Name	Type	Summary
external_host_provider	ExternalHostProvider	

7.64. EXTERNALDISCOVEREDHOST STRUCT

Table 7.84. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
ip	String	
last_report	String	
mac	String	
name	String	A human-readable name in plain text.
subnet_name	String	

Table 7.85. Links summary

Name	Type	Summary
external_host_provider	ExternalHostProvider	

7.65. EXTERNALHOST STRUCT

Table 7.86. Attributes summary

Name	Type	Summary
address	String	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.87. Links summary

Name	Type	Summary
external_host_provider	ExternalHostProvider	

7.66. EXTERNALHOSTGROUP STRUCT

Table 7.88. Attributes summary

Name	Type	Summary
architecture_name	String	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
domain_name	String	
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Name	Type	Summary
operating_system_name	String	
subnet_name	String	

Table 7.89. Links summary

Name	Type	Summary
external_host_provider	ExternalHostProvider	

7.67. EXTERNALHOSTPROVIDER STRUCT

Table 7.90. Attributes summary

Name	Type	Summary
authentication_url	String	Defines the external provider authentication URL address.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
password	String	Defines password for the user during the authentication process.
properties	Property[]	Array of provider name/value properties.
requires_authentication	Boolean	Defines whether provider authentication is required or not.
url	String	Defines URL address of the external provider.
username	String	Defines user name to be used during authentication process.

7.67.1. requires_authentication

Defines whether provider authentication is required or not.

If authentication is required, both **username** and **password** attributes will be used during authentication.

Table 7.91. Links summary

Name	Type	Summary
certificates	Certificate[]	
compute_resources	ExternalComputeResource[]	
discovered_hosts	ExternalDiscoveredHost[]	
host_groups	ExternalHostGroup[]	
hosts	Host[]	

7.68. EXTERNALPROVIDER STRUCT

Represents an external provider.

Table 7.92. Attributes summary

Name	Type	Summary
authentication_url	String	Defines the external provider authentication URL address.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
password	String	Defines password for the user during the authentication process.
properties	Property[]	Array of provider name/value properties.
requires_authentication	Boolean	Defines whether provider authentication is required or not.
url	String	Defines URL address of the external provider.
username	String	Defines user name to be used during authentication process.

7.68.1. requires_authentication

Defines whether provider authentication is required or not.

If authentication is required, both **username** and **password** attributes will be used during authentication.

7.69. EXTERNALSTATUS ENUM

Table 7.93. Values summary

Name	Summary
error	
failure	
info	
ok	
warning	

7.70. EXTERNALSYSTEMTYPE ENUM

Represents the type of the external system that is associated with the **step**.

Table 7.94. Values summary

Name	Summary
gluster	Represents Gluster as the external system which is associated with the step .
vds	Represents VDSM as the external system which is associated with the step .

7.71. EXTERNALVMIMPORT STRUCT

Describes the parameters for the virtual machine import operation from an external system.

Table 7.95. Attributes summary

Name	Type	Summary
name	String	The name of the virtual machine to be imported, as is defined within the external system.
password	String	The password to authenticate against the external hypervisor system.
provider	ExternalVmProviderType	The type of external virtual machine provider.

Name	Type	Summary
sparse	Boolean	Specifies the disk allocation policy of the resulting virtual machine: true for sparse, false for preallocated.
url	String	The URL to be passed to the virt-v2v tool for conversion.
username	String	The username to authenticate against the external hypervisor system.

7.71.1. url

The URL to be passed to the **virt-v2v** tool for conversion.

Example:

```
vpx://wmware_user@vcenter-host/DataCenter/Cluster/esxi-host?no_verify=1
```

More examples can be found at <http://libguestfs.org/virt-v2v.1.html>.

Table 7.96. Links summary

Name	Type	Summary
cluster	Cluster	Specifies the target cluster for the resulting virtual machine.
cpu_profile	CpuProfile	Optional.
drivers_iso	File	Optional.
host	Host	Optional.
quota	Quota	Optional.
storage_domain	StorageDomain	Specifies the target storage domain for converted disks.
vm	Vm	The virtual machine entity used to specify a name for the newly created virtual machine.

7.71.2. cpu_profile

Optional. Specifies the CPU profile of the resulting virtual machine.

7.71.3. drivers_iso

Optional. The name of the ISO containing drivers that can be used during the **virt-v2v** conversion process.

7.71.4. host

Optional. Specifies the host (using host's ID) to be used for the conversion process. If not specified, one is selected automatically.

7.71.5. quota

Optional. Specifies the quota that will be applied to the resulting virtual machine.

7.71.6. vm

The virtual machine entity used to specify a name for the newly created virtual machine.

If a name is not specified, the source virtual machine name will be used.

7.72. EXTERNALVMPROVIDERTYPE ENUM

Describes the type of external hypervisor system.

Table 7.97. Values summary

Name	Summary
kvm	
vmware	
xen	

7.73. FAULT STRUCT

Table 7.98. Attributes summary

Name	Type	Summary
detail	String	
reason	String	

7.74. FENCETYPE ENUM

Type representing the type of the fence operation.

Table 7.99. Values summary

Name	Summary
manual	Manual host fencing via power management.

Name	Summary
restart	Restart the host via power management.
start	Start the host via power management.
status	Check the host power status via power management.
stop	Stop the host via power management.

7.75. FENCINGPOLICY STRUCT

Type representing a cluster fencing policy.

Table 7.100. Attributes summary

Name	Type	Summary
enabled	Boolean	Enable or disable fencing on this cluster.
skip_if_connectivity_broken	SkipIfConnectivity Broken	If enabled, we will not fence a host in case more than a configurable percentage of hosts in the cluster lost connectivity as well.
skip_if_gluster_bricks_up	Boolean	A flag indicating if fencing should be skipped if Gluster bricks are up and running in the host being fenced.
skip_if_gluster_quorum_not_met	Boolean	A flag indicating if fencing should be skipped if Gluster bricks are up and running and Gluster quorum will not be met without those bricks.
skip_if_sd_active	SkipIfSdActive	If enabled, we will skip fencing in case the host maintains its lease in the storage.

7.75.1. skip_if_connectivity_broken

If enabled, we will not fence a host in case more than a configurable percentage of hosts in the cluster lost connectivity as well. This comes to prevent fencing *storm* in cases where there is a global networking issue in the cluster.

7.75.2. skip_if_gluster_bricks_up

A flag indicating if fencing should be skipped if Gluster bricks are up and running in the host being fenced. This flag is optional, and the default value is **false**.

7.75.3. skip_if_gluster_quorum_not_met

A flag indicating if fencing should be skipped if Gluster bricks are up and running and Gluster quorum will not be met without those bricks. This flag is optional, and the default value is **false**.

7.75.4. skip_if_sd_active

If enabled, we will skip fencing in case the host maintains its lease in the storage. It means that if the host still has storage access then it won't get fenced.

7.76. FILE STRUCT

Table 7.101. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
content	String	
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
type	String	

Table 7.102. Links summary

Name	Type	Summary
storage_domain	StorageDomain	

7.77. FILTER STRUCT

Table 7.103. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Name	Type	Summary
position	Integer	

Table 7.104. Links summary

Name	Type	Summary
scheduling_policy_unit	SchedulingPolicyUnit	

7.78. FLOPPY STRUCT

Table 7.105. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
file	File	
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.106. Links summary

Name	Type	Summary
instance_type	InstanceType	Optionally references to an instance type the device is used by.
template	Template	Optionally references to a template the device is used by.
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

7.78.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk may be used simultaneously by two or more virtual machines.

7.79. FOPSTATISTIC STRUCT

Table 7.107. Attributes summary

Name	Type	Summary
name	String	
statistics	Statistic[]	

7.80. GLUSTERBRICK STRUCT

Table 7.108. Attributes summary

Name	Type	Summary
brick_dir	String	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
device	String	
fs_name	String	
gluster_clients	GlusterClient[]	
id	String	A unique identifier.
memory_pools	GlusterMemoryPool[]	
mnt_options	String	
name	String	A human-readable name in plain text.
pid	Integer	
port	Integer	
server_id	String	
status	GlusterBrickStatus	

Table 7.109. Links summary

Name	Type	Summary
gluster_volum e	GlusterVolume	
instance_type e	InstanceType	Optionally references to an instance type the device is used by.
statistics	Statistic[]	
template	Template	Optionally references to a template the device is used by.
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

7.80.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk may be used simultaneously by two or more virtual machines.

7.81. GLUSTERBRICKADVANCEDDETAILS STRUCT

Table 7.110. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
device	String	
fs_name	String	
gluster_clients	GlusterClient[]	
id	String	A unique identifier.
memory_pools	GlusterMemoryPool[]	
mnt_options	String	
name	String	A human-readable name in plain text.

Name	Type	Summary
<code>pid</code>	Integer	
<code>port</code>	Integer	

Table 7.111. Links summary

Name	Type	Summary
<code>instance_type</code>	InstanceType	Optionally references to an instance type the device is used by.
<code>template</code>	Template	Optionally references to a template the device is used by.
<code>vm</code>	Vm	Don't use this element, use vms instead.
<code>vms</code>	Vm[]	References to the virtual machines that are using this device.

7.81.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk may be used simultaneously by two or more virtual machines.

7.82. GLUSTERBRICKMEMORYINFO STRUCT

Table 7.112. Attributes summary

Name	Type	Summary
<code>memory_pools</code>	GlusterMemoryPool[]	

7.83. GLUSTERBRICKSTATUS ENUM

Table 7.113. Values summary

Name	Summary
<code>down</code>	Brick is in down state, the data cannot be stored or retrieved from it.
<code>unknown</code>	When the status cannot be determined due to host being non-responsive.
<code>up</code>	Brick is in up state, the data can be stored or retrieved from it.

7.84. GLUSTERCLIENT STRUCT

Table 7.114. Attributes summary

Name	Type	Summary
<code>bytes_read</code>	Integer	
<code>bytes_written</code>	Integer	
<code>client_port</code>	Integer	
<code>host_name</code>	String	

7.85. GLUSTERHOOK STRUCT

Table 7.115. Attributes summary

Name	Type	Summary
<code>checksum</code>	String	
<code>comment</code>	String	Free text containing comments about this object.
<code>conflict_status</code>	Integer	
<code>conflicts</code>	String	
<code>content</code>	String	
<code>content_type</code>	HookContentType	
<code>description</code>	String	A human-readable description in plain text.
<code>gluster_command</code>	String	
<code>id</code>	String	A unique identifier.
<code>name</code>	String	A human-readable name in plain text.
<code>stage</code>	HookStage	
<code>status</code>	GlusterHookStatus	

Table 7.116. Links summary

Name	Type	Summary
cluster	Cluster	
server_hooks	GlusterServerHook []	

7.86. GLUSTERHOOKSTATUS ENUM

Table 7.117. Values summary

Name	Summary
disabled	Hook is disabled in the cluster.
enabled	Hook is enabled in the cluster.
missing	Unknown/missing hook status.

7.87. GLUSTERMEMORYPOOL STRUCT

Table 7.118. Attributes summary

Name	Type	Summary
alloc_count	Integer	
cold_count	Integer	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
hot_count	Integer	
id	String	A unique identifier.
max_alloc	Integer	
max_stdalloc	Integer	
name	String	A human-readable name in plain text.
padded_size	Integer	
pool_misses	Integer	

Name	Type	Summary
type	String	

7.88. GLUSTERSERVERHOOK STRUCT

Table 7.119. Attributes summary

Name	Type	Summary
checksum	String	
comment	String	Free text containing comments about this object.
content_type	HookContentType	
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
status	GlusterHookStatus	

Table 7.120. Links summary

Name	Type	Summary
host	Host	

7.89. GLUSTERSTATE ENUM

Table 7.121. Values summary

Name	Summary
down	
unknown	
up	

7.90. GLUSTERVOLUME STRUCT

Table 7.122. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
disperse_count	Integer	
id	String	A unique identifier.
name	String	A human-readable name in plain text.
options	Option[]	
redundancy_count	Integer	
replica_count	Integer	
status	GlusterVolumeStatus	
stripe_count	Integer	
transport_types	TransportType[]	
volume_type	GlusterVolumeType	

Table 7.123. Links summary

Name	Type	Summary
bricks	GlusterBrick[]	
cluster	Cluster	
statistics	Statistic[]	

7.91. GLUSTERVOLUMEPROFILEDETAILS STRUCT

Table 7.124. Attributes summary

Name	Type	Summary
brick_profile_details	BrickProfileDetail[]	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
nfs_profile_details	NfsProfileDetail[]	

7.92. GLUSTERVOLUMESTATUS ENUM

Table 7.125. Values summary

Name	Summary
down	Volume needs to be started, for clients to be able to mount and use it.
unknown	When the status cannot be determined due to host being non-responsive.
up	Volume is started, and can be mounted and used by clients.

7.93. GLUSTERVOLUMETYPE ENUM

Type representing the type of Gluster Volume.

Table 7.126. Values summary

Name	Summary
disperse	Dispersed volumes are based on erasure codes, providing space-efficient protection against disk or server failures.
distribute	Distributed volumes distributes files throughout the bricks in the volume.
distributed_disperse	Distributed dispersed volumes distribute files across dispersed subvolumes.
distributed_replicate	Distributed replicated volumes distributes files across replicated bricks in the volume.

Name	Summary
distributed_stripe	Distributed striped volumes stripe data across two or more nodes in the cluster.
distributed_stripped_replicate	Distributed striped replicated volumes distributes striped data across replicated bricks in the cluster.
replicate	Replicated volumes replicates files across bricks in the volume.
stripe	Striped volumes stripes data across bricks in the volume.
striped_replicate	Striped replicated volumes stripes data across replicated bricks in the cluster.

7.93.1. disperse

Dispersed volumes are based on erasure codes, providing space-efficient protection against disk or server failures.

Dispersed volumes an encoded fragment of the original file to each brick in a way that only a subset of the fragments is needed to recover the original file. The number of bricks that can be missing without losing access to data is configured by the administrator on volume creation time.

7.93.2. distribute

Distributed volumes distributes files throughout the bricks in the volume.

Distributed volumes can be used where the requirement is to scale storage and the redundancy is either not important or is provided by other hardware/software layers.

7.93.3. distributed_disperse

Distributed dispersed volumes distribute files across dispersed subvolumes.

This has the same advantages of distribute replicate volumes, but using disperse to store the data into the bricks.

7.93.4. distributed_replicate

Distributed replicated volumes distributes files across replicated bricks in the volume.

Distributed replicated volumes can be used in environments where the requirement is to scale storage and high-reliability is critical. Distributed replicated volumes also offer improved read performance in most environments.

7.93.5. distributed_stripe

Distributed striped volumes stripe data across two or more nodes in the cluster.

Distributed striped volumes should be used where the requirement is to scale storage and in high concurrency environments accessing very large files is critical.

Note: With the introduction of Sharding in Glusterfs 3.7 releases, striped volumes are not recommended and it will be removed in future release.

7.93.6. distributed_striped_replicate

Distributed striped replicated volumes distributes striped data across replicated bricks in the cluster.

For best results, distributed striped replicated volumes should be used in highly concurrent environments where parallel access of very large files and performance is critical.

Note: With the introduction of Sharding in Glusterfs 3.7 releases, striped volumes are not recommended and it will be removed in future release.

7.93.7. replicate

Replicated volumes replicates files across bricks in the volume.

Replicated volumes can be used in environments where high-availability and high-reliability are critical.

7.93.8. stripe

Striped volumes stripes data across bricks in the volume.

For best results, striped volumes should only in high concurrency environments accessing very large files.

Note: With the introduction of Sharding in Glusterfs 3.7 releases, striped volumes are not recommended and it will be removed in future release.

7.93.9. striped_replicate

Striped replicated volumes stripes data across replicated bricks in the cluster.

For best results, striped replicated volumes should be used in highly concurrent environments where there is parallel access of very large files and performance is critical.

Note: With the introduction of Sharding in Glusterfs 3.7 releases, striped volumes are not recommended and it will be removed in future release.

7.94. GRACEPERIOD STRUCT

Table 7.127. Attributes summary

Name	Type	Summary
expiry	Integer	

7.95. GRAPHICSCONSOLE STRUCT

Table 7.128. Attributes summary

Name	Type	Summary
address	String	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
port	Integer	
protocol	GraphicsType	
tls_port	Integer	

Table 7.129. Links summary

Name	Type	Summary
instance_type	InstanceType	
template	Template	
vm	Vm	

7.96. GRAPHICSTYPE ENUM

Table 7.130. Values summary

Name	Summary
spice	
vnc	

7.97. GROUP STRUCT

This type represents all groups in the directory service.

Table 7.131. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
domain_entry_id	String	The containing directory service domain id.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
namespace	String	Namespace where group resides.

Table 7.132. Links summary

Name	Type	Summary
domain	Domain	A link to the domain containing this group.
permissions	Permission[]	A link to the permissions sub-collection for permissions attached to this group.
roles	Role[]	A link to the roles sub-collection for roles attached to this group.
tags	Tag[]	A link to the tags sub-collection for tags attached to this group.

7.97.1. roles

A link to the roles sub-collection for roles attached to this group.

Used only to represent the initial role assignments for a new group; thereafter, modification of role assignments is only supported via the **roles** sub-collection.

7.98. GUESTOPERATINGSYSTEM STRUCT

Table 7.133. Attributes summary

Name	Type	Summary
architecture	String	
codename	String	
distribution	String	

Name	Type	Summary
family	String	
kernel	Kernel	
version	Version	

7.99. HARDWAREINFORMATION STRUCT

Table 7.134. Attributes summary

Name	Type	Summary
family	String	
manufacturer	String	
product_name	String	
serial_number	String	
supported_rng_sources	RngSource[]	
uuid	String	
version	String	

7.100. HIGHAVAILABILITY STRUCT

Table 7.135. Attributes summary

Name	Type	Summary
enabled	Boolean	
priority	Integer	Indicates the priority of the virtual machine inside the run and migration queues.

7.100.1. priority

Indicates the priority of the virtual machine inside the run and migration queues.

Virtual machines with higher priorities will be started and migrated before virtual machines with lower priorities.

The value is an integer between 0 and 100. The higher the value, the higher the priority.

The graphical user interface (GUI) does not allow specifying all the possible values, instead it only allows you to select *Low*, *Medium* or *High*. When the value is set using the API, the GUI will set the label as follows:

API Value	GUI Label
0 - 25	Low
26 - 74	Medium
75 - 100	High

When the label is selected using the GUI, the value in the API will be set as follows:

GUI Label	API Value
Low	1
Medium	50
High	100

7.101. HOOK STRUCT

Represents a hook.

Table 7.136. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
event_name	String	Name of the event to execute the hook on.
id	String	A unique identifier.
md5	String	Checksum of the hook.
name	String	A human-readable name in plain text.

Table 7.137. Links summary

Name	Type	Summary
host	Host	Reference to the host the hook belongs to.

7.102. HOOKCONTENTTYPE ENUM

Represents content type of hook script.

Table 7.138. Values summary

Name	Summary
binary	Binary content type of the hook.
text	Text content type of the hook.

7.103. HOOKSTAGE ENUM

Table 7.139. Values summary

Name	Summary
post	
pre	

7.104. HOOKSTATUS ENUM

Type represents the status of a hook.

Table 7.140. Values summary

Name	Summary
disabled	Hook is disabled.
enabled	Hook is enabled.
missing	Hook is missing.

7.105. HOST STRUCT

Type representing a host.

Table 7.141. Attributes summary

Name	Type	Summary
address	String	The host address (FQDN/IP).
auto_numa_status	AutoNumaStatus	The host auto <i>non uniform memory access</i> (NUMA) status.
certificate	Certificate	The host certificate.
comment	String	Free text containing comments about this object.
cpu	Cpu	The CPU type of this host.
description	String	A human-readable description in plain text.
device_passthrough	HostDevicePassthrough	Specifies whether host device passthrough is enabled on this host.
display	Display	Optionally specify the display address of this host explicitly.
external_status	ExternalStatus	The host external status.
hardware_information	HardwareInformation	The host hardware information.
hosted_engine	HostedEngine	The hosted engine status on this host.
id	String	A unique identifier.
iscsi	IscsiDetails	The host iSCSI details.
kdump_status	KdumpStatus	The host KDUMP status.
ksm	Ksm	Kernel SamePage Merging (KSM) reduces references to memory pages from multiple identical pages to a single page reference.
libvirt_version	Version	The host libvirt version.
max_scheduling_memory	Integer	The max scheduling memory on this host in bytes.
memory	Integer	The amount of physical memory on this host in bytes.
name	String	A human-readable name in plain text.

Name	Type	Summary
numa_supported	Boolean	Specifies whether <i>non uniform memory access</i> (NUMA) is supported on this host.
os	OperatingSystem	The operating system on this host.
override_iptables	Boolean	Specifies whether we should override firewall definitions.
port	Integer	The host port.
power_management	PowerManagement	The host power management definitions.
protocol	HostProtocol	The protocol that the engine uses to communicate with the host.
root_password	String	When creating a new host, a root password is required if the password authentication method is chosen, but this is not subsequently included in the representation.
se_linux	SeLinux	The host SELinux status.
spm	Spm	The host <i>storage pool manager</i> (SPM) status and definition.
ssh	Ssh	The SSH definitions.
status	HostStatus	The host status.
status_detail	String	The host status details.
summary	VmSummary	The virtual machine summary - how many are active, migrating and total.
transparent_huge_pages	TransparentHugePages	Transparent huge page support expands the size of memory pages beyond the standard 4 KiB limit.
type	HostType	Indicates if the host contains a full installation of the operating system or a scaled-down version intended only to host virtual machines.
update_available	Boolean	Specifies whether there is an oVirt-related update on this host.
version	Version	The version of VDSM.

7.105.1. external_status

The host external status. This can be used by third-party software to change the host external status in case of an issue. This has no effect on the host lifecycle, unless a third-party software checks for this status and acts accordingly.

7.105.2. kdump_status

The host KDUMP status. KDUMP happens when the host kernel has crashed and it is now going through memory dumping.

7.105.3. ksm

Kernel SamePage Merging (KSM) reduces references to memory pages from multiple identical pages to a single page reference. This helps with optimization for memory density.

For example, to enable KSM for host **123**, send a request like this:

```
PUT /ovirt-engine/api/hosts/123
```

With a request body like this:

```
<host>
  <ksm>
    <enabled>true</enabled>
  </ksm>
</host>
```

7.105.4. libvirt_version

The host libvirt version. For more information on libvirt please go to [libvirt](#).

7.105.5. override_iptables

Specifies whether we should override firewall definitions. This applies only when the host is installed or re-installed.

7.105.6. protocol

The protocol that the engine uses to communicate with the host.



WARNING

Since version 4.1 of the engine the protocol is always set to **stomp** since **xml** was removed.

7.105.7. se_linux

The host SELinux status. *Security-Enhanced Linux (SELinux)* is a component in the Linux kernel that provides a mechanism for supporting access control security policies.

7.105.8. spm

The host *storage pool manager* (SPM) status and definition. Use it to set the SPM priority of this host, and to see whether this is the current SPM or not.

7.105.9. status_detail

The host status details. Relevant for Gluster hosts.

7.105.10. transparent_huge_pages

Transparent huge page support expands the size of memory pages beyond the standard 4 KiB limit. This reduces memory consumption and increases host performance.

For example, to enable transparent huge page support for host **123**, send a request like this:

```
PUT /ovirt-engine/api/hosts/123
```

With a request body like this:

```
<host>
  <transparent_hugepages>
    <enabled>true</enabled>
  </transparent_hugepages>
</host>
```

7.105.11. version

The version of VDSM.

For example:

```
GET /ovirt-engine/api/hosts/123
```

This **GET** request will return the following output:

```
<host>
  ...
  <version>
    <build>999</build>
    <full_version>vds-4.18.999-419.gitcf06367.el7</full_version>
    <major>4</major>
    <minor>18</minor>
    <revision>0</revision>
  </version>
  ...
</host>
```

Table 7.142. Links summary

Name	Type	Summary
affinity_labels	AffinityLabel[]	
agents	Agent[]	
cluster	Cluster	
devices	Device[]	
external_host_provider	ExternalHostProvider	
hooks	Hook[]	
katello_errata	KatelloErratum[]	Lists all the Katello errata assigned to the host.
network_attachments	NetworkAttachment[]	
nic	HostNic[]	
numa_nodes	NumaNode[]	
permissions	Permission[]	
statistics	Statistic[]	Each host resource exposes a statistics sub-collection for host-specific statistics.
storage_connection_extensions	StorageConnectionExtension[]	
storages	HostStorage[]	
tags	Tag[]	
unmanaged_networks	UnmanagedNetwork[]	

7.105.12. katello_errata

Lists all the Katello errata assigned to the host.

```
GET /ovirt-engine/api/hosts/123/katelloerrata
```

You will receive response in XML like this one:

```
<katello_errata>
  <katello_erratum href="/ovirt-engine/api/katelloerrata/456" id="456">
    <name>RHBA-2013:XYZ</name>
    <description>The description of the erratum</description>
    <title>some bug fix update</title>
    <type>bugfix</type>
    <issued>2013-11-20T02:00:00.000+02:00</issued>
    <solution>Few guidelines regarding the solution</solution>
    <summary>Updated packages that fix one bug are now available for
XYZ</summary>
    <packages>
      <package>
        <name>libipa_hbac-1.9.2-82.11.el6_4.i686</name>
      </package>
      ...
    </packages>
  </katello_erratum>
  ...
</katello_errata>
```

7.105.13. statistics

Each host resource exposes a statistics sub-collection for host-specific statistics.

An example of an XML representation:

```
<statistics>
  <statistic href="/ovirt-engine/api/hosts/123/statistics/456" id="456">
    <name>memory.total</name>
    <description>Total memory</description>
    <kind>gauge</kind>
    <type>integer</type>
    <unit>bytes</unit>
    <values>
      <value>
        <datum>25165824000</datum>
      </value>
    </values>
    <host href="/ovirt-engine/api/hosts/123" id="123"/>
  </statistic>
  ...
</statistics>
```



NOTE

This statistics sub-collection is read-only.

The following list shows the statistic types for hosts:

Name	Description
<code>memory.total</code>	Total memory in bytes on the host.
<code>memory.used</code>	Memory in bytes used on the host.
<code>memory.free</code>	Memory in bytes free on the host.
<code>memory.shared</code>	Memory in bytes shared on the host.
<code>memory.buffers</code>	I/O buffers in bytes.
<code>memory.cached</code>	OS caches in bytes.
<code>swap.total</code>	Total swap memory in bytes on the host.
<code>swap.free</code>	Swap memory in bytes free on the host.
<code>swap.used</code>	Swap memory in bytes used on the host.
<code>swap.cached</code>	Swap memory in bytes also cached in host's memory.
<code>ksm.cpu.current</code>	Percentage of CPU usage for Kernel SamePage Merging.
<code>cpu.current.user</code>	Percentage of CPU usage for user slice.
<code>cpu.current.system</code>	Percentage of CPU usage for system.
<code>cpu.current.idle</code>	Percentage of idle CPU usage.
<code>cpu.load.avg.5m</code>	CPU load average per five minutes.
<code>boot.time</code>	Boot time of the machine.

7.106. HOSTDEVICE STRUCT

Table 7.143. Attributes summary

Name	Type	Summary
<code>capability</code>	String	
<code>comment</code>	String	Free text containing comments about this object.
<code>description</code>	String	A human-readable description in plain text.

Name	Type	Summary
driver	String	The name of the driver this device is bound to.
id	String	A unique identifier.
iommu_group	Integer	
name	String	A human-readable name in plain text.
physical_function	HostDevice	
placeholder	Boolean	
product	Product	
vendor	Vendor	
virtual_functions	Integer	

7.106.1. driver

The name of the driver this device is bound to.

For example: **pcieport** or **uhci_hcd**.

Table 7.144. Links summary

Name	Type	Summary
host	Host	
parent_device	HostDevice	
vm	Vm	

7.107. HOSTDEVICEPASSTHROUGH STRUCT

Table 7.145. Attributes summary

Name	Type	Summary
enabled	Boolean	

7.108. HOSTNIC STRUCT

Represents a host NIC.

For example, the XML representation of a host NIC looks like this:

```
<host_nic href="/ovirt-engine/api/hosts/123/nics/456" id="456">
  <name>eth0</name>
  <boot_protocol>static</boot_protocol>
  <bridged>true</bridged>
  <custom_configuration>true</custom_configuration>
  <ip>
    <address>192.168.122.39</address>
    <gateway>192.168.122.1</gateway>
    <netmask>255.255.255.0</netmask>
    <version>v4</version>
  </ip>
  <ipv6>
    <gateway>:</gateway>
    <version>v6</version>
  </ipv6>
  <ipv6_boot_protocol>none</ipv6_boot_protocol>
  <mac>
    <address>52:54:00:0c:79:1d</address>
  </mac>
  <mtu>1500</mtu>
  <status>up</status>
</host_nic>
```

A bonded interface is represented as a [HostNic](#) object containing the **bonding** and **slaves** attributes.

For example, the XML representation of a bonded host NIC looks like this:

```
<host_nic href="/ovirt-engine/api/hosts/123/nics/456" id="456">
  <name>bond0</name>
  <mac address="00:00:00:00:00:00"/>
  <ip>
    <address>192.168.122.39</address>
    <gateway>192.168.122.1</gateway>
    <netmask>255.255.255.0</netmask>
    <version>v4</version>
  </ip>
  <boot_protocol>dhcp</boot_protocol>
  <bonding>
    <options>
      <option>
        <name>mode</name>
        <value>4</value>
        <type>Dynamic link aggregation (802.3ad)</type>
      </option>
    </options>
  </bonding>
</host_nic>
```

```

    </option>
    <option>
      <name>miimon</name>
      <value>100</value>
    </option>
  </options>
  <slaves>
    <host_nic id="123"/>
    <host_nic id="456"/>
  </slaves>
</bonding>
<mtu>1500</mtu>
<bridged>true</bridged>
<custom_configuration>>false</custom_configuration>
</host_nic>

```

Table 7.146. Attributes summary

Name	Type	Summary
ad_aggregator_id	Integer	The ad_aggregator_id property of a bond or bond slave, for bonds in mode 4.
base_interface	String	The base interface of the NIC.
bonding	Bonding	The bonding parameters of the NIC.
boot_protocol	BootProtocol	The IPv4 boot protocol configuration of the NIC.
bridged	Boolean	Defines the bridged network status.
check_connectivity	Boolean	
comment	String	Free text containing comments about this object.
custom_configuration	Boolean	
description	String	A human-readable description in plain text.
id	String	A unique identifier.
ip	Ip	The IPv4 address of the NIC.
ipv6	Ip	The IPv6 address of the NIC.
ipv6_boot_protocol	BootProtocol	The IPv6 boot protocol configuration of the NIC.

Name	Type	Summary
mac	Mac	The MAC address of the NIC.
mtu	Integer	The maximum transmission unit for the interface.
name	String	A human-readable name in plain text.
network_labels	NetworkLabel[]	The labels that are applied to this NIC.
override_configuration	Boolean	
properties	Property[]	
speed	Integer	
statistics	Statistic[]	A link to the statistics of the NIC.
status	NicStatus	
virtual_functions_configuration	HostNicVirtualFunctionsConfiguration	Describes the virtual functions configuration of a physical function NIC.
vlan	Vlan	

7.108.1. ad_aggregator_id

The **ad_aggregator_id** property of a bond or bond slave, for bonds in mode 4. Bond mode 4 is the 802.3ad standard, also called dynamic link aggregation. (See [Wikipedia](#) and [Presentation](#) for more information). This is only valid for bonds in mode 4, or NICs which are part of a bond. It is not present for bonds in other modes, or NICs which are not part of a bond in mode 4. The **ad_aggregator_id** property indicates which of the bond slaves are active. The value of the **ad_aggregator_id** of an active slave is the same as the value of the **ad_aggregator_id** property of the bond. This parameter is read only. Setting it will have no effect on the bond/NIC. It is retrieved from the `/sys/class/net/bondX/bonding/ad_aggregator` file for a bond, and the `/sys/class/net/ensX/bonding_slave/ad_aggregator_id` file for a NIC.

7.108.2. bridged

Defines the bridged network status. Set to **true** for a bridged network and **false** for a bridgeless network.

7.108.3. statistics

A link to the statistics of the NIC.

The data types for HostNic statistical values:

- `data.current.rx` - The rate in bytes per second of data received.
- `data.current.tx` - The rate in bytes per second of data transmitted.
- `data.total.rx` - Total received data.
- `data.total.tx` - Total transmitted data.
- `errors.total.rx` - Total errors from receiving data.
- `errors.total.tx` - Total errors from transmitting data.

Table 7.147. Links summary

Name	Type	Summary
<code>host</code>	Host	
<code>network</code>	Network	A reference to the network to which the interface should be connected.
<code>physical_function</code>	HostNic	A reference to the physical function NIC of a SR-IOV virtual function NIC.
<code>qos</code>	Qos	A link to the quality-of-service configuration of the interface.

7.108.4. network

A reference to the network to which the interface should be connected. A blank network ID is allowed.

7.109. HOSTNICVIRTUALFUNCTIONSCONFIGURATION STRUCT

Describes the virtual functions configuration of an SR-IOV-enabled physical function NIC.

Table 7.148. Attributes summary

Name	Type	Summary
<code>all_networks_allowed</code>	Boolean	Defines whether all networks are allowed to be defined on the related virtual functions, or specified ones only.
<code>max_number_of_virtual_functions</code>	Integer	The maximum number of virtual functions the NIC supports.
<code>number_of_virtual_functions</code>	Integer	The number of virtual functions currently defined.

7.109.1. max_number_of_virtual_functions

The maximum number of virtual functions the NIC supports. This property is read-only.

7.109.2. number_of_virtual_functions

The number of virtual functions currently defined. A user-defined value between 0 and `max_number_of_virtual_functions`.

7.110. HOSTPROTOCOL ENUM

The protocol used by the engine to communicate with a host.



WARNING

Since version 4.1 of the engine the protocol is always set to `stomp` since `xml` was removed.

Table 7.149. Values summary

Name	Summary
<code>stomp</code>	JSON-RPC protocol on top of STOMP.
<code>xml</code>	XML-RPC protocol.

7.111. HOSTSTATUS ENUM

Type representing a host status.

Table 7.150. Values summary

Name	Summary
<code>connecting</code>	The engine cannot communicate with the host for a specific threshold so it is now trying to connect before going through fencing.
<code>down</code>	The host is down.
<code>error</code>	The host is in error status.
<code>initializing</code>	The host is initializing.
<code>install_failed</code>	The host installation failed.

Name	Summary
installing	The host is being installed.
installing_os	The host operating system is now installing.
kdumping	The host kernel has crashed and it is now going through memory dumping.
maintenance	The host is in maintenance status.
non_operational	The host is non operational.
non_responsive	The host is not responsive.
pending_approval	The host is pending administrator approval.
preparing_for_maintenance	The host is preparing for maintenance.
reboot	The host is being rebooted.
unassigned	The host is in activation process.
up	The host is up.

7.111.1. error

The host is in error status. This will happen if we will try to run a virtual machine several times and it will fail.

7.111.2. initializing

The host is initializing. This is an intermediate step before moving the host to 'up' status.

7.111.3. install_failed

The host installation failed. In such cases look at the event log to understand what failed the installation, and issue a re-install.

7.111.4. installing_os

The host operating system is now installing. This status is relevant when using a Satellite/Foreman provider, and issuing a bare-metal provisioning (discovered host provisioning).

7.111.5. maintenance

The host is in maintenance status. When a host is in maintenance it cannot run virtual machines.

7.111.6. non_operational

The host is non operational. This can happen due to various reasons, such as not having a connection with the storage, not supporting a mandatory network, not supporting the cluster level, and more.

7.111.7. non_responsive

The host is not responsive. This means that the engine is not able to communicate with the host.

7.111.8. pending_approval

The host is pending administrator approval. This is relevant only for vintage ovirt-node / RHV-H.

7.111.9. preparing_for_maintenance

The host is preparing for maintenance. During this time the engine makes sure to live migrate all the virtual machines from this host to other hosts. Once all migrations have been completed the host will move to 'maintenance' status.

7.112. HOSTSTORAGE STRUCT

Table 7.151. Attributes summary

Name	Type	Summary
address	String	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
logical_units	LogicalUnit[]	
mount_options	String	
name	String	A human-readable name in plain text.
nfs_retrans	Integer	The number of times to retry a request before attempting further recovery actions.
nfs_timeo	Integer	The time in tenths of a second to wait for a response before retrying NFS requests.

Name	Type	Summary
nfs_version	NfsVersion	
override_luns	Boolean	
password	String	
path	String	
port	Integer	
portal	String	
target	String	
type	StorageType	
username	String	
vfs_type	String	
volume_group	VolumeGroup	

7.112.1. nfs_retrans

The number of times to retry a request before attempting further recovery actions. The value must be in the range of 0 to 65535. For more details see the description of the **retrans** mount option in the **nfs** man page.

7.112.2. nfs_timeo

The time in tenths of a second to wait for a response before retrying NFS requests. The value must be in the range of 0 to 65535. For more details see the description of the **timeo** mount option in the **nfs** man page.

Table 7.152. Links summary

Name	Type	Summary
host	Host	

7.113. HOSTTYPE ENUM

This enumerated type is used to determine which type of operating system is used by the host.

Table 7.153. Values summary

Name	Summary
ovirt_node	The host contains Red Hat Virtualization Host (RHVH): a new implementation of Red Hat Enterprise Virtualization Hypervisor (RHEV-H) which uses the same installer as Red Hat Enterprise Linux, CentOS, or Fedora.
rhel	The host contains a full Red Hat Enterprise Linux, CentOS, or Fedora installation.
rhev_h	The host contains Red Hat Enterprise Virtualization Hypervisor (RHEV-H), a small-scaled version of Red Hat Enterprise Linux, CentOS, or Fedora, used solely to host virtual machines.

7.113.1. ovirt_node

The host contains Red Hat Virtualization Host (RHVH): a new implementation of Red Hat Enterprise Virtualization Hypervisor (RHEV-H) which uses the same installer as Red Hat Enterprise Linux, CentOS, or Fedora. The main difference between RHVH and legacy RHEV-H is that RHVH has a writeable file system and will handle its own installation instead of having RPMs pushed to it by the Manager like in legacy RHEV-H.

7.114. HOSTEDENGINE STRUCT

Table 7.154. Attributes summary

Name	Type	Summary
active	Boolean	
configured	Boolean	
global_maintenance	Boolean	
local_maintenance	Boolean	
score	Integer	

7.115. ICON STRUCT

Icon of virtual machine or template.

Table 7.155. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.

Name	Type	Summary
data	String	Base64 encode content of the icon file.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
media_type	String	Format of icon file.
name	String	A human-readable name in plain text.

7.115.1. media_type

Format of icon file.

One of:

- **image/jpeg**
- **image/png**
- **image/gif**

7.116. IDENTIFIED STRUCT

This interface is the base model for all types that represent objects with an identifier.

Table 7.156. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

7.117. IMAGE STRUCT

Table 7.157. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.

Name	Type	Summary
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.158. Links summary

Name	Type	Summary
storage_domain	StorageDomain	

7.118. IMAGETRANSFER STRUCT

This type contains information regarding an image transfer being performed.

Table 7.159. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
direction	ImageTransferDirection	The direction indicates whether the transfer is sending image data (upload) or receiving image data (download).
id	String	A unique identifier.
name	String	A human-readable name in plain text.
phase	ImageTransferPhase	The current phase of the image transfer in progress.
proxy_url	String	The URL of the proxy server that the user inputs or outputs to.
signed_ticket	String	The signed ticket that should be attached as an Authentication header in the HTTPS request for the proxy server to input or output to (See the proxy_url attribute).

7.118.1. direction

The direction indicates whether the transfer is sending image data (**upload**) or receiving image data (**download**).

If a direction is not set during an addition of a new transfer, The default direction for the transfer will be **upload**.

7.118.2. phase

The current phase of the image transfer in progress. Each transfer needs a managed session, which must be opened for the user to input or output an image. Please refer to [image transfer](#) for further documentation.

7.118.3. proxy_url

The URL of the proxy server that the user inputs or outputs to. This attribute is available only if the image transfer entity is in the [transferring](#) phase. See **phase** for details.

Table 7.160. Links summary

Name	Type	Summary
host	Host	The host which will be used to write to the image which is targeted for input or output.
image	Image	The image which is targeted for input or output.

7.119. IMAGETRANSFERDIRECTION ENUM

The [image transfer](#) direction for a transfer.

When adding a new transfer, the user can choose whether the transfer will be to an image, choosing **upload**, or to transfer from an image- choosing **download** as an ImageTransferDirection.

Please refer to [image transfer](#) for further documentation.

Table 7.161. Values summary

Name	Summary
download	The user must choose download when he/she wants to stream data from an image.
upload	The user can choose upload when he/she wants to stream data to an image.

7.120. IMAGETRANSFERPHASE ENUM

A list of possible phases for an [image transfer](#) entity. Each of these values defines a specific point in a transfer flow.

Please refer to [image transfer](#) for more information.

Table 7.162. Values summary

Name	Summary
cancelled	This phase will be set as a result of the user cancelling the transfer.
finalizing_failure	This phase can only be set in the Administration Portal, and indicates that there was an error during the transfer, and it is being finalized with a failure.
finalizing_success	This phase will be set when the user calls finalize .
finished_failure	Indicates that the targeted image failed the verification, and cannot be used.
finished_success	Indicates that the transfer session was successfully closed, and the targeted image was verified and ready to be used.
initializing	The initial phase of an image transfer.
paused_system	This phase means the session timed out, or some other error occurred with this transfer; for example <code>ovirt-imageio-daemon</code> is not running in the selected host.
paused_user	This phase is a result of a pause call by the user, using pause .
resuming	The phase where the transfer has been resumed by the client calling resume .
transferring	The phase where the transfer session is open, and the client can input or output the desired image using the preferred tools.
unknown	An unknown phase.

7.120.1. cancelled

This phase will be set as a result of the user cancelling the transfer. The cancellation can only be performed in the Administration Portal.

7.120.2. finalizing_success

This phase will be set when the user calls [finalize](#). Calling `finalize` is essential to finish the transfer session, and finish using the targeted image. After finalizing, the phase will be changed to **finished_success** or **finished_failure**.

Refer to [image transfer](#) for more information.

7.120.3. finished_failure

Indicates that the targeted image failed the verification, and cannot be used. After reaching this phase, the image transfer entity will be deleted, and the targeted image will be set to illegal.

7.120.4. finished_success

Indicates that the transfer session was successfully closed, and the targeted image was verified and ready to be used. After reaching this phase, the image transfer entity will be deleted.

7.120.5. initializing

The initial phase of an image transfer. It is set while the transfer session is establishing. Once the session is established, the phase will be changed to **transferring**

7.120.6. paused_system

This phase means the session timed out, or some other error occurred with this transfer; for example `ovirt-imageio-daemon` is not running in the selected host. To resume the session, the client should call [resume](#). After resuming, the phase will change to **resuming**.

7.120.7. resuming

The phase where the transfer has been resumed by the client calling [resume](#). Resuming starts a new session, and after calling it, the phase will be changed to **transferring**, or **paused_system** in case of a failure.

7.120.8. unknown

An unknown phase. This will only be set in cases of unpredictable errors.

7.121. INHERITABLEBOOLEAN ENUM

Enum representing the boolean value that can be either set, or inherited from a higher level. The inheritance order is virtual machine → cluster → engine-config.

Table 7.163. Values summary

Name	Summary
false	Set the value to false on this level.
inherit	Inherit the value from higher level.
true	Set the value to true on this level.

7.122. INITIALIZATION STRUCT

Table 7.164. Attributes summary

Name	Type	Summary
active_directory_ou	String	
authorized_ssh_keys	String	

Name	Type	Summary
cloud_init	CloudInit	
configuration	Configuration	
custom_script	String	
dns_search	String	
dns_servers	String	
domain	String	
host_name	String	
input_locale	String	
nic_configurations	NicConfiguration[]	
org_name	String	
regenerate_ids	Boolean	
regenerate_ssh_keys	Boolean	
root_password	String	
system_locale	String	
timezone	String	
ui_language	String	
user_locale	String	
user_name	String	
windows_license_key	String	

7.123. INSTANCETYPE STRUCT

Describes the hardware configuration of virtual machines.

For example **medium** instance type includes 1 virtual CPU and 4 GiB of memory. It is a top-level entity (e.g. not bound to any data center or cluster). The attributes that are used for instance types and are common to virtual machine and template types are:

- **console**
- **cpu**
- **custom_cpu_model**
- **custom_emulated_machine**
- **display**
- **high_availability**
- **io**
- **memory**
- **memory_policy**
- **migration**
- **migration_downtime**
- **os**
- **rng_device**
- **soundcard_enabled**
- **usb**
- **virtio_scsi**

When creating a virtual machine from both an instance type and a template, the virtual machine will inherit the hardware configurations from the instance type



NOTE

An instance type inherits its attributes from the template entity although most template attributes are not used in instance types.

Table 7.165. Attributes summary

Name	Type	Summary
bios	Bios	Reference to virtual machine's BIOS configuration.

Name	Type	Summary
comment	String	Free text containing comments about this object.
console	Console	Console configured for this virtual machine.
cpu	Cpu	The configuration of the virtual machine CPU.
cpu_shares	Integer	
creation_time	Date	The virtual machine creation date.
custom_compatibility_version	Version	Virtual machine custom compatibility version.
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	Properties sent to VDSM to configure various hooks.
delete_protected	Boolean	If true , the virtual machine cannot be deleted.
description	String	A human-readable description in plain text.
display	Display	The virtual machine display configuration.
domain	Domain	Domain configured for this virtual machine.
high_availability	HighAvailability	The virtual machine high availability configuration.
id	String	A unique identifier.
initialization	Initialization	Reference to virtual machine's initialization configuration.
io	Io	For performance tuning of IO threading.
large_icon	Icon	Virtual machine's large icon.

Name	Type	Summary
lease	StorageDomainLease	Reference to the storage domain this virtual machine/template lease reside on.
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	Reference to virtual machine's memory management configuration.
migration	MigrationOptions	Reference to configuration of migration of running virtual machine to another host.
migration_downtime	Integer	Maximum time the virtual machine can be non responsive during its live migration to another host in ms.
name	String	A human-readable name in plain text.
origin	String	The origin of this virtual machine.
os	OperatingSystem	Operating system type installed on the virtual machine.
rng_device	RngDevice	Random Number Generator device configuration for this virtual machine.
serial_number	SerialNumber	Virtual machine's serial number in a cluster.
small_icon	Icon	Virtual machine's small icon.
soundcard_enabled	Boolean	If true , the sound card is added to the virtual machine.
sso	Sso	Reference to the Single Sign On configuration this virtual machine is configured for.
start_paused	Boolean	If true , the virtual machine will be initially in 'paused' state after start.
stateless	Boolean	If true , the virtual machine is stateless - it's state (disks) are rolled-back after shutdown.
status	TemplateStatus	The status of the template.
time_zone	TimeZone	The virtual machine's time zone set by oVirt.
tunnel_migration	Boolean	If true , the network data transfer will be encrypted during virtual machine live migration.

Name	Type	Summary
type	VmType	Determines whether the virtual machine is optimized for desktop or server.
usb	Usb	Configuration of USB devices for this virtual machine (count, type).
version	TemplateVersion	Indicates whether this is a base version or a sub version of another template.
virtio_scsi	VirtioScsi	Reference to VirtIO SCSI configuration.
vm	Vm	The virtual machine configuration associated with this template.

7.123.1. cpu

The configuration of the virtual machine CPU.

The socket configuration can be updated without rebooting the virtual machine. The cores and the threads require a reboot.

For example, to change the number of sockets to 4 immediately, and the number of cores and threads to 2 after reboot, send the following request:

```
PUT /ovirt-engine/api/vms/123
```

With a request body:

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

7.123.2. custom_compatibility_version

Virtual machine custom compatibility version.

Enables a virtual machine to be customized to its own compatibility version. If **custom_compatibility_version** is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in, and is checked against capabilities of the host the virtual machine is planned to run on.

7.123.3. high_availability

The virtual machine high availability configuration. If set, the virtual machine will be automatically restarted when it unexpectedly goes down.

7.123.4. large_icon

Virtual machine's large icon. Either set by user or refers to image set according to operating system.

7.123.5. lease

Reference to the storage domain this virtual machine/template lease reside on.

A virtual machine running with a lease requires checking while running that the lease is not taken by another host, preventing another instance of this virtual machine from running on another host. This provides protection against split-brain in highly available virtual machines. A template can also have a storage domain defined for a lease in order to have the virtual machines created from this template to be preconfigured with this storage domain as the location of the leases.

7.123.6. memory

The virtual machine's memory, in bytes.

For example, to update a virtual machine to contain 1 Gibibyte (GiB) of memory, send the following request:

```
PUT /ovirt-engine/api/vms/123
```

With the following request body:

```
<vm>
  <memory>1073741824</memory>
</vm>
```



NOTE

Memory in the example is converted to bytes using the following formula:
 $1 \text{ GiB} = 2^{30} \text{ bytes} = 1073741824 \text{ bytes}$.



NOTE

Memory hot plug is supported from Red Hat Virtualization 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

7.123.7. migration_downtime

Maximum time the virtual machine can be non responsive during its live migration to another host in ms.

Set either explicitly for the virtual machine or by **engine-config -s DefaultMaximumMigrationDowntime=[value]**

7.123.8. origin

The origin of this virtual machine.

Possible values:

- `ovirt`
- `rhev`
- `vmware`
- `xen`
- `external`
- `hosted_engine`
- `managed_hosted_engine`
- `kvm`
- `physical_machine`
- `hyperv`

7.123.9. `small_icon`

Virtual machine's small icon. Either set by user or refers to image set according to operating system.

7.123.10. `sso`

Reference to the Single Sign On configuration this virtual machine is configured for. The user can be automatically signed in the virtual machine's operating system when console is opened.

Table 7.166. Links summary

Name	Type	Summary
<code>cdroms</code>	Cdrom[]	References to the CD-ROM devices attached to the template.
<code>cluster</code>	Cluster	Reference to cluster the virtual machine belongs to.
<code>cpu_profile</code>	CpuProfile	Reference to CPU profile used by this virtual machine.
<code>disk_attachments</code>	DiskAttachment[]	References to the disks attached to the template.
<code>graphics_consoles</code>	GraphicsConsole[]	References to the graphic consoles attached to the template.
<code>nics</code>	Nic[]	References to the network interfaces attached to the template.
<code>permissions</code>	Permission[]	References to the user permissions attached to the template.
<code>quota</code>	Quota	Reference to quota configuration set for this virtual machine.

Name	Type	Summary
storage_domain	StorageDomain	Reference to storage domain the virtual machine belongs to.
tags	Tag[]	References to the tags attached to the template.
watchdogs	Watchdog[]	References to the watchdog devices attached to the template.

7.124. IO STRUCT

Table 7.167. Attributes summary

Name	Type	Summary
threads	Integer	

7.125. IP STRUCT

Represents the IP configuration of a network interface.

Table 7.168. Attributes summary

Name	Type	Summary
address	String	The text representation of the IP address.
gateway	String	The address of the default gateway.
netmask	String	The network mask.
version	IpVersion	The version of the IP protocol.

7.125.1. address

The text representation of the IP address.

For example, an IPv4 address will be represented as follows:

```
<ip>
  <address>192.168.0.1</address>
  ...
</ip>
```

An IPv6 address will be represented as follows:

```
<ip>
  <address>2620:52:0:20f0:4216:7eff:feaa:1b50</address>
```



```
...
</ip>
```

7.125.2. netmask

The network mask.

For IPv6 addresses the value is an integer in the range of 0-128, which represents the subnet prefix.

7.125.3. version

The version of the IP protocol.



NOTE

From version 4.1 of the Manager this attribute will be optional, and when a value is not provided, it will be inferred from the value of the **address** attribute.

7.126. IPADDRESSASSIGNMENT STRUCT

Represents an IP address assignment for a network device.

For a static boot protocol assignment, subnet mask and IP address (and optionally default gateway) must be provided in the IP configuration.

Table 7.169. Attributes summary

Name	Type	Summary
assignment_method	BootProtocol	Sets the boot protocol used to assign the IP configuration for a network device.
ip	Ip	Sets the IP configuration for a network device.

7.127. IPVERSION ENUM

Defines the values for the IP protocol version.

Table 7.170. Values summary

Name	Summary
v4	IPv4.
v6	IPv6.

7.128. ISCSIBOND STRUCT

Table 7.171. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.172. Links summary

Name	Type	Summary
data_center	DataCenter	
networks	Network[]	
storage_connections	StorageConnection []	

7.129. ISCSIDetails Struct

Table 7.173. Attributes summary

Name	Type	Summary
address	String	
disk_id	String	
initiator	String	
lun_mapping	Integer	
password	String	
paths	Integer	
port	Integer	
portal	String	
product_id	String	
serial	String	

Name	Type	Summary
size	Integer	
status	String	
storage_domain_id	String	
target	String	
username	String	
vendor_id	String	
volume_group_id	String	

7.130. JOB STRUCT

Represents a job, which monitors execution of a flow in the system. A job can contain multiple steps in a hierarchic structure. The steps can be processed in parallel, depends on the implementation of the flow.

Table 7.174. Attributes summary

Name	Type	Summary
auto_cleared	Boolean	Indicates if the job should be cleared automatically after it was completed by the system.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
end_time	Date	The end time of the job.
external	Boolean	Indicates if the job is originated by an external system.
id	String	A unique identifier.
last_updated	Date	The last update date of the job.
name	String	A human-readable name in plain text.
start_time	Date	The start time of the job.
status	JobStatus	The status of the job.

7.130.1. external

Indicates if the job is originated by an external system. External jobs are managed externally, by the creator of the job.

Table 7.175. Links summary

Name	Type	Summary
owner	User	The user who is the owner of the job.
steps	Step[]	The steps of the job.

7.131. JOBSTATUS ENUM

Represents the status of the job.

Table 7.176. Values summary

Name	Summary
aborted	The aborted job status.
failed	The failed job status.
finished	The finished job status.
started	The started job status.
unknown	The unknown job status.

7.131.1. aborted

The aborted job status. This status is applicable for an external job that was forcibly aborted.

7.131.2. finished

The finished job status. This status describes a completed job execution.

7.131.3. started

The started job status. This status represents a job which is currently being executed.

7.131.4. unknown

The unknown job status. This status represents jobs which their resolution is not known, i.e. jobs that were executed before the system was unexpectedly restarted.

7.132. KATELLOERRATUM STRUCT

Type representing a Katello erratum.

Table 7.177. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
issued	Date	The date when the Katello erratum was issued.
name	String	A human-readable name in plain text.
packages	Package[]	The list of packages which solve the issue reported by the Katello erratum.
severity	String	The severity of the Katello erratum.
solution	String	The solution for the issue described by the Katello erratum.
summary	String	The summary of the Katello erratum.
title	String	The title of the Katello erratum.
type	String	The type of the Katello erratum.

7.132.1. severity

The severity of the Katello erratum.

The supported severities are **moderate**, **important** or **critical**.

7.132.2. type

The type of the Katello erratum.

The supported types are **bugfix**, **enhancement** or **security**.

Table 7.178. Links summary

Name	Type	Summary
host	Host	Reference to the host that the Katello erratum is assigned to.
vm	Vm	Reference to the virtual machine that the Katello erratum is assigned to.

7.133. KDUMPSTATUS ENUM

Table 7.179. Values summary

Name	Summary
disabled	
enabled	
unknown	

7.134. KERNEL STRUCT

Table 7.180. Attributes summary

Name	Type	Summary
version	Version	

7.135. KSM STRUCT

Table 7.181. Attributes summary

Name	Type	Summary
enabled	Boolean	
merge_across_nodes	Boolean	

7.136. LOGSEVERITY ENUM

Enum representing a severity of an event.

Table 7.182. Values summary

Name	Summary
alert	Alert severity.
error	Error severity.
normal	Normal severity.
warning	Warning severity.

7.136.1. alert

Alert severity. Used to specify a condition that requires an immediate attention.

7.136.2. error

Error severity. Used to specify that there is an error that needs to be examined.

7.136.3. normal

Normal severity. Used for information events.

7.136.4. warning

Warning severity. Used to warn something might be wrong.

7.137. LOGICALUNIT STRUCT

Table 7.183. Attributes summary

Name	Type	Summary
address	String	
discard_max_size	Integer	The maximum number of bytes that can be discarded by the logical unit's underlying storage in a single operation.
discard_zeroes_data	Boolean	True, if previously discarded blocks in the logical unit's underlying storage are read back as zeros.
disk_id	String	
id	String	
lun_mapping	Integer	
password	String	
paths	Integer	
port	Integer	
portal	String	
product_id	String	
serial	String	
size	Integer	

Name	Type	Summary
status	LunStatus	
storage_domain_id	String	
target	String	
username	String	
vendor_id	String	
volume_group_id	String	

7.137.1. discard_max_size

The maximum number of bytes that can be discarded by the logical unit's underlying storage in a single operation. A value of 0 means that the device does not support discard functionality.



NOTE

This is the software limit, and not the hardware limit, as noted in the [documentation](#) of `queue-sysfs` for `discard_max_bytes`.

7.137.2. discard_zeroes_data

True, if previously discarded blocks in the logical unit's underlying storage are read back as zeros. For more information please see the [documentation](#) of `queue-sysfs` for `discard_zeroes_data`.

7.138. LUNSTATUS ENUM

Table 7.184. Values summary

Name	Summary
free	
unusable	
used	

7.139. MAC STRUCT

Table 7.185. Attributes summary

Name	Type	Summary
address	String	

7.140. MACPOOL STRUCT

Represents a MAC address pool.

Example of an XML representation of a MAC address pool:

```
<mac_pool href="/ovirt-engine/api/macpools/123" id="123">
  <name>Default</name>
  <description>Default MAC pool</description>
  <allow_duplicates>false</allow_duplicates>
  <default_pool>true</default_pool>
  <ranges>
    <range>
      <from>00:1A:4A:16:01:51</from>
      <to>00:1A:4A:16:01:E6</to>
    </range>
  </ranges>
</mac_pool>
```

Table 7.186. Attributes summary

Name	Type	Summary
allow_duplicates	Boolean	Defines whether duplicate MAC addresses are permitted in the pool.
comment	String	Free text containing comments about this object.
default_pool	Boolean	Defines whether this is the default pool.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
ranges	Range[]	Defines the range of MAC addresses for the pool.

7.140.1. allow_duplicates

Defines whether duplicate MAC addresses are permitted in the pool. If not specified, defaults to **false**.

7.140.2. default_pool

Defines whether this is the default pool. If not specified, defaults to **false**.

7.140.3. ranges

Defines the range of MAC addresses for the pool. Multiple ranges can be defined.

7.141. MEMORYOVERCOMMIT STRUCT

Table 7.187. Attributes summary

Name	Type	Summary
percent	Integer	

7.142. MEMORYPOLICY STRUCT

Logical grouping of memory related properties of virtual machine-like entities.

Table 7.188. Attributes summary

Name	Type	Summary
ballooning	Boolean	
guaranteed	Integer	
max	Integer	Maximum virtual machine's memory, in bytes.
over_commit	MemoryOverCom mit	
transparent_ huge_pages	TransparentHugeP ages	

7.142.1. max

Maximum virtual machine's memory, in bytes.

The user provides the value in bytes, and the engine rounds the value down to the nearest lower MiB value.

For example, if the user enters a value of 1073741825 (1 GiB + 1 byte), then the Red Hat Virtualization Manager will truncate that value to the nearest lower MiB boundary: in this case 1073741824 (1 GiB).

7.143. MESSAGEBROKERTYPE ENUM

Table 7.189. Values summary

Name	Summary
qpid	

Name	Summary
<code>rabbit_mq</code>	

7.144. METHOD STRUCT

Table 7.190. Attributes summary

Name	Type	Summary
<code>id</code>	SsoMethod	

7.145. MIGRATEONERROR ENUM

Table 7.191. Values summary

Name	Summary
<code>do_not_migrate</code>	
<code>migrate</code>	
<code>migrate_highly_available</code>	

7.146. MIGRATIONBANDWIDTH STRUCT

Defines the bandwidth used by migration.

Table 7.192. Attributes summary

Name	Type	Summary
<code>assignment_method</code>	MigrationBandwidthAssignmentMethod	The method used to assign the bandwidth.
<code>custom_value</code>	Integer	Custom bandwidth in Mbps.

7.146.1. custom_value

Custom bandwidth in Mbps. Will be applied only if the `assignmentMethod` attribute is `custom`.

7.147. MIGRATIONBANDWIDTHASSIGNMENTMETHOD ENUM

Defines how the migration bandwidth is assigned.

Table 7.193. Values summary

Name	Summary
auto	Takes the bandwidth from the Quality of Service if the Quality of Service is defined.
custom	Custom defined bandwidth in Mbit/s.
hypervisor_default	Takes the value as configured on the hypervisor.

7.147.1. auto

Takes the bandwidth from the Quality of Service if the Quality of Service is defined. If the Quality of Service is not defined the bandwidth is taken from the detected link speed being used. If nothing is detected, bandwidth falls back to the `hypervisor_default` value.

7.148. MIGRATIONOPTIONS STRUCT

The type for migration options.

Table 7.194. Attributes summary

Name	Type	Summary
auto_convergence	InheritableBoolean	
bandwidth	MigrationBandwidth	The bandwidth that is allowed to be used by the migration.
compressed	InheritableBoolean	

Table 7.195. Links summary

Name	Type	Summary
policy	MigrationPolicy	A reference to the migration policy, as defined using engine-config .

7.149. MIGRATIONPOLICY STRUCT

A policy describing how the migration is treated, such as convergence or how many parallel migrations are allowed.

Table 7.196. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

7.150. NETWORK STRUCT

The type for a logical network.

An example of the JSON representation of a logical network:

```
{
  "network" : [ {
    "data_center" : {
      "href" : "/ovirt-engine/api/datacenters/123",
      "id" : "123"
    },
    "stp" : "false",
    "mtu" : "0",
    "usages" : {
      "usage" : [ "vm" ]
    },
    "name" : "ovirtmgmt",
    "description" : "Management Network",
    "href" : "/ovirt-engine/api/networks/456",
    "id" : "456",
    "link" : [ {
      "href" : "/ovirt-engine/api/networks/456/permissions",
      "rel" : "permissions"
    }, {
      "href" : "/ovirt-engine/api/networks/456/vnicprofiles",
      "rel" : "vnicprofiles"
    }, {
      "href" : "/ovirt-engine/api/networks/456/labels",
      "rel" : "labels"
    } ]
  } ]
}
```

An example of the XML representation of the same logical network:

```
<network href="/ovirt-engine/api/networks/456" id="456">
  <name>ovirtmgmt</name>
  <description>Management Network</description>
  <link href="/ovirt-engine/api/networks/456/permissions"
rel="permissions"/>
  <link href="/ovirt-engine/api/networks/456/vnicprofiles">
```

```

rel="vnicprofiles"/>
  <link href="/ovirt-engine/api/networks/456/labels" rel="labels"/>
  <data_center href="/ovirt-engine/api/datacenters/123" id="123"/>
  <stp>>false</stp>
  <mtu>0</mtu>
  <usages>
    <usage>vm</usage>
  </usages>
</network>

```

Table 7.197. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
display	Boolean	
dns_resolver_configuration	DnsResolverConfiguration	
id	String	A unique identifier.
ip	Ip	
mtu	Integer	Specifies the maximum transmission unit for the network.
name	String	A human-readable name in plain text.
profile_required	Boolean	
required	Boolean	
status	NetworkStatus	
stp	Boolean	Specifies whether the spanning tree protocol is enabled for the network.
usages	NetworkUsage[]	Defines a set of usage elements for the network.
vlan	Vlan	

7.150.1. usages

Defines a set of usage elements for the network.

For example, users can specify that the network is to be used for virtual machine traffic and also for display traffic with the `vm` and `display` values.

Table 7.198. Links summary

Name	Type	Summary
<code>cluster</code>	Cluster	
<code>data_center</code>	DataCenter	A reference to the data center that the network is a member of.
<code>network_labels</code>	NetworkLabel[]	A reference to the labels assigned to the network.
<code>permissions</code>	Permission[]	A reference to the permissions of the network.
<code>qos</code>	Qos	
<code>vnics_profiles</code>	VnicProfile[]	A reference to the profiles of the network.

7.151. NETWORKATTACHMENT STRUCT

Describes how a host connects to a network.

An XML representation of a network attachment on a host:

```
<network_attachment href="/ovirt-engine/api/hosts/123/nics/456/networkattachments/789" id="789">
  <network href="/ovirt-engine/api/networks/234" id="234"/>
  <host_nic href="/ovirt-engine/api/hosts/123/nics/123" id="123"/>
  <in_sync>true</in_sync>
  <ip_address_assignments>
    <ip_address_assignment>
      <assignment_method>static</assignment_method>
      <ip>
        <address>192.168.122.39</address>
        <gateway>192.168.122.1</gateway>
        <netmask>255.255.255.0</netmask>
        <version>v4</version>
      </ip>
    </ip_address_assignment>
  </ip_address_assignments>
  <reported_configurations>
    <reported_configuration>
      <name>mtu</name>
      <expected_value>1500</expected_value>
      <actual_value>1500</actual_value>
      <in_sync>true</in_sync>
    </reported_configuration>
    <reported_configuration>
      <name>bridged</name>
```

```

    <expected_value>true</expected_value>
    <actual_value>true</actual_value>
    <in_sync>true</in_sync>
  </reported_configuration>
  ...
</reported_configurations>
</network_attachment>

```

The network element, with either a **name** or an **id**, is required in order to attach a network to a network interface card (NIC).

For example, to attach a network to a host network interface card, send a request like this:

```
POST /ovirt-engine/api/hosts/123/nics/456/networkattachments
```

With a request body like this:

```

<networkattachment>
  <network id="234"/>
</networkattachment>

```

To attach a network to a host, send a request like this:

```
POST /ovirt-engine/api/hosts/123/networkattachments
```

With a request body like this:

```

<network_attachment>
  <network id="234"/>
  <host_nic id="456"/>
</network_attachment>

```

The **ip_address_assignments** and **properties** elements are updatable post-creation.

For example, to update a network attachment, send a request like this:

```
PUT /ovirt-engine/api/hosts/123/nics/456/networkattachments/789
```

With a request body like this:

```

<network_attachment>
  <ip_address_assignments>
    <ip_address_assignment>
      <assignment_method>static</assignment_method>
      <ip>
        <address>7.1.1.1</address>
        <gateway>7.1.1.2</gateway>
        <netmask>255.255.255.0</netmask>
        <version>v4</version>
      </ip>
    </ip_address_assignment>
  </ip_address_assignments>
</network_attachment>

```


To detach a network from the network interface card send a request like this:

```
DELETE /ovirt-engine/api/hosts/123/nics/456/networkattachments/789
```



IMPORTANT

Changes to network attachment configuration must be explicitly committed.

An XML representation of a network attachment's **properties** sub-collection:

```
<network_attachment>
  <properties>
    <property>
      <name>bridge_opts</name>
      <value>
        forward_delay=1500 group_fwd_mask=0x0 multicast_snooping=1
      </value>
    </property>
  </properties>
  ...
</network_attachment>
```

Table 7.199. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
dns_resolver_configuration	DnsResolverConfiguration	DNS resolver configuration will be reported when retrieving the network attachment using GET.
id	String	A unique identifier.
in_sync	Boolean	
ip_address_assignments	IpAddressAssignment[]	The IP configuration of the network.
name	String	A human-readable name in plain text.
properties	Property[]	Defines custom properties for the network configuration.
reported_configurations	ReportedConfiguration[]	A read-only list of configuration properties.

7.151.1. dns_resolver_configuration

DNS resolver configuration will be reported when retrieving the network attachment using GET. It is optional when creating a new network attachment or updating an existing one.

7.151.2. properties

Defines custom properties for the network configuration.

Bridge options have the set name of `bridge_opts`. Separate multiple entries with a whitespace character. The following keys are valid for **bridge_opts**:

Name	Default value
<code>forward_delay</code>	<code>1500</code>
<code>gc_timer</code>	<code>3765</code>
<code>group_addr</code>	<code>1:80:c2:0:0:0</code>
<code>group_fwd_mask</code>	<code>0x0</code>
<code>hash_elasticity</code>	<code>4</code>
<code>hash_max</code>	<code>512</code>
<code>hello_time</code>	<code>200</code>
<code>hello_timer</code>	<code>70</code>
<code>max_age</code>	<code>2000</code>
<code>multicast_last_member_count</code>	<code>2</code>
<code>multicast_last_member_interval</code>	<code>100</code>
<code>multicast_membership_interval</code>	<code>26000</code>
<code>multicast_querier</code>	<code>0</code>
<code>multicast_querier_interval</code>	<code>25500</code>
<code>multicast_query_interval</code>	<code>13000</code>
<code>multicast_query_response_interval</code>	<code>1000</code>
<code>multicast_query_use_ifaddr</code>	<code>0</code>
<code>multicast_router</code>	<code>1</code>

Name	Default value
<code>multicast_snooping</code>	1
<code>multicast_startup_query_count</code>	2
<code>multicast_startup_query_interval</code>	3125

Table 7.200. Links summary

Name	Type	Summary
<code>host</code>	Host	
<code>host_nic</code>	HostNic	A reference to the host network interface.
<code>network</code>	Network	A reference to the network that the interface is attached to.
<code>qos</code>	Qos	

7.152. NETWORKCONFIGURATION STRUCT

Table 7.201. Attributes summary

Name	Type	Summary
<code>dns</code>	Dns	
<code>nics</code>	Nic[]	

7.153. NETWORKFILTER STRUCT

Network filters filter packets sent to and from the virtual machine's NIC according to defined rules.

There are several types of network filters supported based on libvirt. For more details about the different network filters see [here](#).

In addition to libvirt's network filters, there are two additional network filters: The first is called **vds-m-no-mac-spoofing** and is composed of **no-mac-spoofing** and **no-arp-mac-spoofing**. The second is called **ovirt-no-filter** and is used when no network filter is to be defined for the virtual machine's NIC. The **ovirt-no-filter** network filter is only used for internal implementation, and does not exist on the NICs.

This is an example of the XML representation:

```
<network_filter id="00000019-0019-0019-0019-00000000026c">
  <name>example-filter</name>
  <version>
```

```

    <major>4</major>
    <minor>0</minor>
    <build>-1</build>
    <revision>-1</revision>
  </version>
</network_filter>

```

If any part of the version is not present, it is represented by -1.

Table 7.202. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
version	Version	The minimum supported version of a specific NetworkFilter.

7.153.1. version

The minimum supported version of a specific NetworkFilter. This is the version that the NetworkFilter was first introduced in.

7.154. NETWORKLABEL STRUCT

Represents a label which can be added to a host network interface.

Table 7.203. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.204. Links summary

Name	Type	Summary
host_nic	HostNic	

Name	Type	Summary
network	Network	

7.155. NETWORKPLUGINTYPE ENUM

Table 7.205. Values summary

Name	Summary
open_vswitch	

7.156. NETWORKSTATUS ENUM

Table 7.206. Values summary

Name	Summary
non_operational	
operational	

7.157. NETWORKUSAGE ENUM

This type indicates the purpose that the network is used for in the cluster.

Table 7.207. Values summary

Name	Summary
display	The network will be used for SPICE and VNC traffic.
gluster	The network will be used for Gluster (bricks) data traffic.
management	The network will be used for communication between the Red Hat Virtualization Manager and the nodes.
migration	The network will be used for virtual machine migration.
vm	

7.157.1. management

The network will be used for communication between the Red Hat Virtualization Manager and the nodes. This is the network where the ovirtmgmt bridge will be created.

7.158. NFSPROFILEDETAIL STRUCT

Table 7.208. Attributes summary

Name	Type	Summary
nfs_server_ip	String	
profile_details	ProfileDetail[]	

7.159. NFSVERSION ENUM

Table 7.209. Values summary

Name	Summary
auto	
v3	
v4	
v4_1	
v4_2	NFS 4.

7.159.1. v4_2

NFS 4.2.

7.160. NIC STRUCT

Represents a virtual machine NIC.

For example, the XML representation of a NIC will look like this:

```
<nic href="/ovirt-engine/api/vms/123/nics/456" id="456">
  <name>nic1</name>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
  <interface>virtio</interface>
  <linked>true</linked>
  <mac>
    <address>02:00:00:00:00:00</address>
  </mac>
  <plugged>true</plugged>
  <vnic_profile href="/ovirt-engine/api/vnicprofiles/789" id="789"/>
</nic>
```

Table 7.210. Attributes summary

Name	Type	Summary
boot_protocol	BootProtocol	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
interface	NicInterface	The type of driver used for the NIC.
linked	Boolean	Defines if the NIC is linked to the virtual machine.
mac	Mac	The MAC address of the interface.
name	String	A human-readable name in plain text.
on_boot	Boolean	
plugged	Boolean	Defines if the NIC is plugged in to the virtual machine.

Table 7.211. Links summary

Name	Type	Summary
instance_type	InstanceType	Optionally references to an instance type the device is used by.
network	Network	A reference to the network that the interface should be connected to.
network_attachments	NetworkAttachment[]	
network_labels	NetworkLabel[]	
reported_devices	ReportedDevice[]	
statistics	Statistic[]	A link to the statistics for the NIC.
template	Template	Optionally references to a template the device is used by.

Name	Type	Summary
virtual_function_allowed_labels	NetworkLabel[]	
virtual_function_allowed_networks	Network[]	
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.
vnic_profile	VnicProfile	

7.160.1. network

A reference to the network that the interface should be connected to. A blank network ID is allowed.

Usage of this element for creating or updating a NIC is deprecated; use **vnic_profile** instead. It is preserved because it is still in use by the **initialization** element, as a holder for IP addresses and other network details.

7.160.2. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk may be used simultaneously by two or more virtual machines.

7.161. NICCONFIGURATION STRUCT

Table 7.212. Attributes summary

Name	Type	Summary
boot_protocol	BootProtocol	
ip	Ip	
name	String	
on_boot	Boolean	

7.162. NICINTERFACE ENUM

Table 7.213. Values summary

Name	Summary
<code>e1000</code>	
<code>pci_passthrough</code>	
<code>rtl8139</code>	
<code>rtl8139_virtio</code>	
<code>spapr_vlan</code>	
<code>virtio</code>	

7.163. NICSTATUS ENUM

Table 7.214. Values summary

Name	Summary
<code>down</code>	
<code>up</code>	

7.164. NUMANODE STRUCT

Represents a physical NUMA node.

Example XML representation:

```
<host_numa_node href="/ovirt-engine/api/hosts/0923f1ea/numanodes/007cf1ab"
id="007cf1ab">
  <cpu>
    <cores>
      <core>
        <index>0</index>
      </core>
    </cores>
  </cpu>
</index>0</index>
<memory>65536</memory>
<node_distance>40 20 40 10</node_distance>
<host href="/ovirt-engine/api/hosts/0923f1ea" id="0923f1ea"/>
</host_numa_node>
```

Table 7.215. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
cpu	Cpu	
description	String	A human-readable description in plain text.
id	String	A unique identifier.
index	Integer	
memory	Integer	Memory of the NUMA node in MB.
name	String	A human-readable name in plain text.
node_distance	String	

Table 7.216. Links summary

Name	Type	Summary
host	Host	
statistics	Statistic[]	Each host NUMA node resource exposes a statistics sub-collection for host NUMA node specific statistics.

7.164.1. statistics

Each host NUMA node resource exposes a statistics sub-collection for host NUMA node specific statistics.

An example of an XML representation:

```
<statistics>
  <statistic href="/ovirt-
engine/api/hosts/123/numanodes/456/statistics/789" id="789">
    <name>memory.total</name>
    <description>Total memory</description>
    <kind>gauge</kind>
    <type>integer</type>
    <unit>bytes</unit>
    <values>
      <value>
        <datum>25165824000</datum>
      </value>
    </values>
    <host_numa_node href="/ovirt-engine/api/hosts/123/numanodes/456"
id="456" />
  </statistic>
</statistics>
```

```

</statistic>
...
</statistics>

```

**NOTE**

This statistics sub-collection is read-only.

The following list shows the statistic types for a host NUMA node:

Name	Description
<code>memory.total</code>	Total memory in bytes on the NUMA node.
<code>memory.used</code>	Memory in bytes used on the NUMA node.
<code>memory.free</code>	Memory in bytes free on the NUMA node.
<code>cpu.current.user</code>	Percentage of CPU usage for user slice.
<code>cpu.current.system</code>	Percentage of CPU usage for system.
<code>cpu.current.idle</code>	Percentage of idle CPU usage.

7.165. NUMANODEPIN STRUCT

Represents the pinning of a virtual NUMA node to a physical NUMA node.

Table 7.217. Attributes summary

Name	Type	Summary
<code>host_numa_node</code>	NumaNode	Deprecated.
<code>index</code>	Integer	The index of a physical NUMA node to which the virtual NUMA node is pinned.
<code>pinned</code>	Boolean	Deprecated.

7.165.1. host_numa_node

Deprecated. Has no function.

7.165.2. pinned

Deprecated. Should always be `true`.

7.166. NUMATUNEMODE ENUM

Table 7.218. Values summary

Name	Summary
interleave	
preferred	
strict	

7.167. OPENSTACKIMAGE STRUCT

Table 7.219. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.220. Links summary

Name	Type	Summary
openstack_image_provider	OpenStackImageProvider	

7.168. OPENSTACKIMAGEPROVIDER STRUCT

Table 7.221. Attributes summary

Name	Type	Summary
authentication_url	String	Defines the external provider authentication URL address.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.

Name	Type	Summary
name	String	A human-readable name in plain text.
password	String	Defines password for the user during the authentication process.
properties	Property[]	Array of provider name/value properties.
requires_authentication	Boolean	Defines whether provider authentication is required or not.
tenant_name	String	
url	String	Defines URL address of the external provider.
username	String	Defines user name to be used during authentication process.

7.168.1. requires_authentication

Defines whether provider authentication is required or not.

If authentication is required, both **username** and **password** attributes will be used during authentication.

Table 7.222. Links summary

Name	Type	Summary
certificates	Certificate[]	
images	OpenStackImage[]	

7.169. OPENSTACKNETWORK STRUCT

Table 7.223. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.224. Links summary

Name	Type	Summary
openstack_network_provider	OpenStackNetworkProvider	

7.170. OPENSTACKNETWORKPROVIDER STRUCT

Table 7.225. Attributes summary

Name	Type	Summary
agent_configuration	AgentConfiguration	Agent configuration settings.
authentication_url	String	Defines the external provider authentication URL address.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
password	String	Defines password for the user during the authentication process.
plugin_type	NetworkPluginType	Network plugin type.
properties	Property[]	Array of provider name/value properties.
read_only	Boolean	Indicates whether the provider is read-only.
requires_authentication	Boolean	Defines whether provider authentication is required or not.
tenant_name	String	
type	OpenStackNetworkProviderType	The type of provider.
url	String	Defines URL address of the external provider.
username	String	Defines user name to be used during authentication process.

7.170.1. read_only

Indicates whether the provider is read-only.

A read-only provider does not allow adding, modifying, or deleting of networks or subnets. Port-related operations are allowed, as they are required for the provisioning of virtual NICs.

7.170.2. requires_authentication

Defines whether provider authentication is required or not.

If authentication is required, both **username** and **password** attributes will be used during authentication.

Table 7.226. Links summary

Name	Type	Summary
certificates	Certificate[]	Reference to the certificates list.
networks	OpenStackNetwork[]	Reference to OpenStack networks list.
subnets	OpenStackSubnet[]	Reference to OpenStack networks subnets list.

7.171. OPENSTACKNETWORKPROVIDERTYPE ENUM

The OpenStack network provider can either be implemented by OpenStack Neutron, in which case the Neutron agent is automatically installed on the hosts, or it can be an external provider implementing the OpenStack API, in which case the virtual interface driver is a custom solution installed manually.

Table 7.227. Values summary

Name	Summary
external	Indicates that the provider is an external one, implementing the OpenStack Neutron API.
neutron	Indicates that the provider is OpenStack Neutron.

7.171.1. external

Indicates that the provider is an external one, implementing the OpenStack Neutron API. The virtual interface driver in this case is implemented by the external provider.

7.171.2. neutron

Indicates that the provider is OpenStack Neutron. The standard OpenStack Neutron agent is used as the virtual interface driver.

7.172. OPENSTACKPROVIDER STRUCT

Table 7.228. Attributes summary

Name	Type	Summary
authentication_url	String	Defines the external provider authentication URL address.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
password	String	Defines password for the user during the authentication process.
properties	Property[]	Array of provider name/value properties.
requires_authentication	Boolean	Defines whether provider authentication is required or not.
tenant_name	String	
url	String	Defines URL address of the external provider.
username	String	Defines user name to be used during authentication process.

7.172.1. requires_authentication

Defines whether provider authentication is required or not.

If authentication is required, both **username** and **password** attributes will be used during authentication.

7.173. OPENSTACKSUBNET STRUCT

Table 7.229. Attributes summary

Name	Type	Summary
cidr	String	Defines network CIDR.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.

Name	Type	Summary
dns_servers	String[]	Defines a list of DNS servers.
gateway	String	Defines IP gateway.
id	String	A unique identifier.
ip_version	String	Defines IP version.
name	String	A human-readable name in plain text.

7.173.1. ip_version

Defines IP version.

Values can be `v4` for IPv4 or `v6` for IPv6.

Table 7.230. Links summary

Name	Type	Summary
openstack_network	OpenStackNetwork	Reference to the service managing the OpenStack network.

7.174. OPENSTACKVOLUMEPROVIDER STRUCT

Table 7.231. Attributes summary

Name	Type	Summary
authentication_url	String	Defines the external provider authentication URL address.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
password	String	Defines password for the user during the authentication process.
properties	Property[]	Array of provider name/value properties.

Name	Type	Summary
requires_authentication	Boolean	Defines whether provider authentication is required or not.
tenant_name	String	
url	String	Defines URL address of the external provider.
username	String	Defines user name to be used during authentication process.

7.174.1. requires_authentication

Defines whether provider authentication is required or not.

If authentication is required, both **username** and **password** attributes will be used during authentication.

Table 7.232. Links summary

Name	Type	Summary
authentication_keys	OpenstackVolumeAuthenticationKey[]	
certificates	Certificate[]	
data_center	DataCenter	
volume_types	OpenStackVolumeType[]	

7.175. OPENSTACKVOLUMETYPE STRUCT

Table 7.233. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
properties	Property[]	

Table 7.234. Links summary

Name	Type	Summary
<code>openstack_volume_provider</code>	OpenStackVolumeProvider	

7.176. OPENSTACKVOLUMEAUTHENTICATIONKEY STRUCT

Table 7.235. Attributes summary

Name	Type	Summary
<code>comment</code>	String	Free text containing comments about this object.
<code>creation_date</code>	Date	
<code>description</code>	String	A human-readable description in plain text.
<code>id</code>	String	A unique identifier.
<code>name</code>	String	A human-readable name in plain text.
<code>usage_type</code>	OpenstackVolumeAuthenticationKeyUsageType	
<code>uuid</code>	String	
<code>value</code>	String	

Table 7.236. Links summary

Name	Type	Summary
<code>openstack_volume_provider</code>	OpenStackVolumeProvider	

7.177. OPENSTACKVOLUMEAUTHENTICATIONKEYUSAGETYPE ENUM

Table 7.237. Values summary

Name	Summary
ceph	

7.178. OPERATINGSYSTEM STRUCT

Information describing the operating system. This is used for both virtual machines and hosts.

Table 7.238. Attributes summary

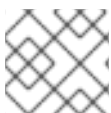
Name	Type	Summary
boot	Boot	
cmdline	String	
custom_kernel_cmdline	String	A custom part of the host kernel command line.
initrd	String	
kernel	String	
reported_kernel_cmdline	String	The host kernel command line as reported by a running host.
type	String	
version	Version	

7.178.1. custom_kernel_cmdline

A custom part of the host kernel command line. This will be merged with the existing kernel command line.

You must reinstall and then reboot the host to apply the changes implemented by this attribute.

During each host deploy procedure, kernel parameters that were added in the previous host deploy procedure are removed using **grubby --update-kernel DEFAULT --remove-args <previous_custom_params>**, and the current kernel command line customization is applied using **grubby --update-kernel DEFAULT --args <custom_params>**. The Manager internally keeps track of the last-applied kernel parameters customization.



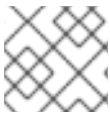
NOTE

This attribute is currently only used for hosts.

7.178.2. reported_kernel_cmdline

The host kernel command line as reported by a running host.

This is a read-only attribute. Attempts to change this attribute are silently ignored.



NOTE

This attribute is currently only used for hosts.

7.179. OPERATINGSYSTEMINFO STRUCT

Table 7.239. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
large_icon	Icon	
name	String	A human-readable name in plain text.
small_icon	Icon	

7.180. OPTION STRUCT

Table 7.240. Attributes summary

Name	Type	Summary
name	String	
type	String	
value	String	

7.181. OSTYPE ENUM

Table 7.241. Values summary

Name	Summary
other	
other_linux	

Name	Summary
rhel_3	
rhel_3x64	
rhel_4	
rhel_4x64	
rhel_5	
rhel_5x64	
rhel_6	
rhel_6x64	
unassigned	
windows_2003	
windows_2003 x64	
windows_2008	
windows_2008 r2x64	
windows_2008 x64	
windows_2012 x64	
windows_7	
windows_7x64	
windows_8	
windows_8x64	
windows_xp	

7.182. PACKAGE STRUCT

Type representing a package.

This is an example of the package element:

```
<package>
  <name>libipa_hbac-1.9.2-82.11.el6_4.i686</name>
</package>
```

Table 7.242. Attributes summary

Name	Type	Summary
name	String	The name of the package.

7.183. PAYLOAD STRUCT

Table 7.243. Attributes summary

Name	Type	Summary
files	File[]	
type	VmDeviceType	
volume_id	String	

7.184. PAYLOADENCODING ENUM

Table 7.244. Values summary

Name	Summary
base64	
plaintext	

7.185. PERMISSION STRUCT

Table 7.245. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.

Name	Type	Summary
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.246. Links summary

Name	Type	Summary
cluster	Cluster	
data_center	DataCenter	
disk	Disk	
group	Group	
host	Host	
role	Role	
storage_domain	StorageDomain	
template	Template	
user	User	
vm	Vm	
vm_pool	VmPool	

7.186. PERMIT STRUCT

Type represents a permit.

Table 7.247. Attributes summary

Name	Type	Summary
administrative	Boolean	Specifies whether permit is administrative or not.
comment	String	Free text containing comments about this object.

Name	Type	Summary
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.248. Links summary

Name	Type	Summary
role	Role	Reference to the role the permit belongs to.

7.187. PMPROXY STRUCT

Table 7.249. Attributes summary

Name	Type	Summary
type	PmProxyType	

7.188. PMPROXYTYPE ENUM

Table 7.250. Values summary

Name	Summary
cluster	The fence proxy is selected from the same cluster as the fenced host.
dc	The fence proxy is selected from the same data center as the fenced host.
other_dc	The fence proxy is selected from a different data center than the fenced host.

7.189. POLICYUNITTYPE ENUM

Holds the types of all internal policy unit types.

Table 7.251. Values summary

Name	Summary
filter	
load_balancing	

Name	Summary
weight	

7.190. PORTMIRRORING STRUCT

7.191. POWERMANAGEMENT STRUCT

Table 7.252. Attributes summary

Name	Type	Summary
address	String	The host name or IP address of the host.
agents	Agent[]	Specifies fence agent options when multiple fences are used.
automatic_pm_enabled	Boolean	Toggles the automated power control of the host in order to save energy.
enabled	Boolean	Indicates whether power management configuration is enabled or disabled.
kdump_detection	Boolean	Toggles whether to determine if kdump is running on the host before it is shut down.
options	Option[]	Fencing options for the selected type= specified with the option name="" and value="" strings.
password	String	A valid, robust password for power management.
pm_proxies	PmProxy[]	Determines the power management proxy.
status	PowerManagementStatus	Determines the power status of the host.
type	String	Fencing device code.
username	String	A valid user name for power management.

7.191.1. agents

Specifies fence agent options when multiple fences are used.

Use the order sub-element to prioritize the fence agents. Agents are run sequentially according to their order until the fence action succeeds. When two or more fence agents have the same order, they are run concurrently. Other sub-elements include type, ip, user, password, and options.

7.191.2. automatic_pm_enabled

Toggles the automated power control of the host in order to save energy. When set to `true`, the host will be automatically powered down if the cluster's load is low, and powered on again when required. This is set to `true` when a host is created, unless disabled by the user.

7.191.3. kdump_detection

Toggles whether to determine if `kdump` is running on the host before it is shut down. When set to `true`, the host will not shut down during a `kdump` process. This is set to `true` when a host has power management enabled, unless disabled by the user.

7.191.4. type

Fencing device code.

A list of valid fencing device codes are available in the `capabilities` collection.

7.192. POWERMANAGEMENTSTATUS ENUM

Table 7.253. Values summary

Name	Summary
<code>off</code>	Host is OFF.
<code>on</code>	Host is ON.
<code>unknown</code>	Unknown status.

7.193. PRODUCT STRUCT

Table 7.254. Attributes summary

Name	Type	Summary
<code>comment</code>	<code>String</code>	Free text containing comments about this object.
<code>description</code>	<code>String</code>	A human-readable description in plain text.
<code>id</code>	<code>String</code>	A unique identifier.
<code>name</code>	<code>String</code>	A human-readable name in plain text.

7.194. PRODUCTINFO STRUCT

Product information.

The entry point contains a **product_info** element to help an API user determine the legitimacy of the Red Hat Virtualization environment. This includes the name of the product, the **vendor** and the **version**.

Verify a genuine Red Hat Virtualization environment

The follow elements identify a genuine Red Hat Virtualization environment:

```
<api>
...
<product_info>
  <name>oVirt Engine</name>
  <vendor>ovirt.org</vendor>
  <version>
    <build>0</build>
    <full_version>4.1.0_master</full_version>
    <major>4</major>
    <minor>1</minor>
    <revision>0</revision>
  </version>
</product_info>
...
</api>
```

Table 7.255. Attributes summary

Name	Type	Summary
name	String	The name of the product, for example oVirt Engine .
vendor	String	The name of the vendor, for example <code>ovirt.org</code> .
version	Version	The version number of the product.

7.194.1. vendor

The name of the vendor, for example `ovirt.org`.

7.195. PROFILEDETAIL STRUCT

Table 7.256. Attributes summary

Name	Type	Summary
block_statistics	BlockStatistic[]	
duration	Integer	
fop_statistics	FopStatistic[]	

Name	Type	Summary
<code>profile_type</code>	String	
<code>statistics</code>	Statistic[]	

7.196. PROPERTY STRUCT

Table 7.257. Attributes summary

Name	Type	Summary
<code>name</code>	String	
<code>value</code>	String	

7.197. PROXYTICKET STRUCT

Table 7.258. Attributes summary

Name	Type	Summary
<code>value</code>	String	

7.198. QCOWVERSION ENUM

The QCOW version specifies to the qemu which qemu version the volume supports.

This field can be updated using the update API and will be reported only for QCOW volumes, it is determined by the storage domain's version which the disk is created on. Storage domains with version lower than V4 support QCOW2 version 2 volumes, while V4 storage domains also support QCOW2 version 3. For more information about features of the different QCOW versions, see [here](#).

Table 7.259. Values summary

Name	Summary
<code>qcow2_v2</code>	The <i>Copy On Write</i> default compatibility version It means that every QEMU can use it.
<code>qcow2_v3</code>	The <i>Copy On Write</i> compatibility version which was introduced in QEMU 1.

7.198.1. qcow2_v3

The *Copy On Write* compatibility version which was introduced in QEMU 1.1 It means that the new format is in use.

7.199. QOS STRUCT

This type represents the attributes to define Quality of service (QoS).

For storage the **type** is [storage](#), the attributes **max_throughput**, **max_read_throughput**, **max_write_throughput**, **max_iops**, **max_read_iops** and **max_write_iops** are relevant.

For resources with computing capabilities the **type** is [cpu](#), the attribute **cpu_limit** is relevant.

For virtual machines networks the **type** is [network](#), the attributes **inbound_average**, **inbound_peak**, **inbound_burst**, **outbound_average**, **outbound_peak** and **outbound_burst** are relevant.

For host networks the **type** is [hostnetwork](#), the attributes **outbound_average_linkshare**, **outbound_average_upperlimit** and **outbound_average_realtime** are relevant.

Table 7.260. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
cpu_limit	Integer	The maximum processing capability in %.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
inbound_average	Integer	The desired average inbound bit rate in Mbps.
inbound_burst	Integer	The amount of data that can be delivered in a single burst in MiB.
inbound_peak	Integer	The maximum inbound rate in Mbps.
max_iops	Integer	Maximum permitted number of input and output operations per second.
max_read_iops	Integer	Maximum permitted number of input operations per second.
max_read_throughput	Integer	Maximum permitted throughput for read operations.
max_throughput	Integer	Maximum permitted total throughput.
max_write_iops	Integer	Maximum permitted number of output operations per second.
max_write_throughput	Integer	Maximum permitted throughput for write operations.

Name	Type	Summary
name	String	A human-readable name in plain text.
outbound_ave rage	Integer	The desired average outbound bit rate in Mbps.
outbound_ave rage_linksha re	Integer	Weighted share.
outbound_ave rage_realtim e	Integer	The committed rate in Mbps.
outbound_ave rage_upperli mit	Integer	The maximum bandwidth to be used by a network in Mbps.
outbound_bur st	Integer	The amount of data that can be sent in a single burst in MiB.
outbound_pea k	Integer	The maximum outbound rate in Mbps.
type	QosType	The kind of resources this entry can be assigned.

7.199.1. cpu_limit

The maximum processing capability in %.

Used to configure computing resources.

7.199.2. inbound_average

The desired average inbound bit rate in Mbps.

Used to configure virtual machines networks. If defined, **inbound_peak** and **inbound_burst** also has to be set.

See [Libvirt-QOS](#) for further details.

7.199.3. inbound_burst

The amount of data that can be delivered in a single burst in MiB.

Used to configure virtual machines networks. If defined, **inbound_average** and **inbound_peak** also has to be set.

See [Libvirt-QOS](#) for further details.

7.199.4. **inbound_peak**

The maximum inbound rate in Mbps.

Used to configure virtual machines networks. If defined, **inbound_average** and **inbound_burst** also has to be set.

See [Libvirt-QOS](#) for further details.

7.199.5. **max_iops**

Maximum permitted number of input and output operations per second.

Used to configure storage. Must not be set if **max_read_iops** or **max_write_iops** is set.

7.199.6. **max_read_iops**

Maximum permitted number of input operations per second.

Used to configure storage. Must not be set if **max_iops** is set.

7.199.7. **max_read_throughput**

Maximum permitted throughput for read operations.

Used to configure storage. Must not be set if **max_throughput** is set.

7.199.8. **max_throughput**

Maximum permitted total throughput.

Used to configure storage. Must not be set if **max_read_throughput** or **max_write_throughput** is set.

7.199.9. **max_write_iops**

Maximum permitted number of output operations per second.

Used to configure storage. Must not be set if **max_iops** is set.

7.199.10. **max_write_throughput**

Maximum permitted throughput for write operations.

Used to configure storage. Must not be set if **max_throughput** is set.

7.199.11. **outbound_average**

The desired average outbound bit rate in Mbps.

Used to configure virtual machines networks. If defined, **outbound_peak** and **outbound_burst** also has to be set.

See [Libvirt-QOS](#) for further details.

7.199.12. `outbound_average_linkshare`

Weighted share.

Used to configure host networks. Signifies how much of the logical link's capacity a specific network should be allocated, relative to the other networks attached to the same logical link. The exact share depends on the sum of shares of all networks on that link. By default this is a number in the range 1-100.

7.199.13. `outbound_average_realtime`

The committed rate in Mbps.

Used to configure host networks. The minimum bandwidth required by a network. The committed rate requested is not guaranteed and will vary depending on the network infrastructure and the committed rate requested by other networks on the same logical link.

7.199.14. `outbound_average_upperlimit`

The maximum bandwidth to be used by a network in Mbps.

Used to configure host networks. If `outboundAverageUpperlimit` and `outbound_average_realtime` are provided, the `outbound_averageUpperlimit` must not be lower than the `outbound_average_realtime`.

See [Libvirt-QOS](#) for further details.

7.199.15. `outbound_burst`

The amount of data that can be sent in a single burst in MiB.

Used to configure virtual machines networks. If defined, `outbound_average` and `outbound_peak` also has to be set.

See [Libvirt-QOS](#) for further details.

7.199.16. `outbound_peak`

The maximum outbound rate in Mbps.

Used to configure virtual machines networks. If defined, `outbound_average` and `outbound_burst` also has to be set.

See [Libvirt-QOS](#) for further details.

Table 7.261. Links summary

Name	Type	Summary
<code>data_center</code>	DataCenter	The data center the QoS is associated to.

7.200. QOSTYPE ENUM

This type represents the kind of resource the [Quality of service \(QoS\)](#) can be assigned to.

Table 7.262. Values summary

Name	Summary
cpu	The Quality of service (QoS) can be assigned to resources with computing capabilities.
hostnetwork	The Quality of service (QoS) can be assigned to host networks.
network	The Quality of service (QoS) can be assigned to virtual machines networks.
storage	The Quality of service (QoS) can be assigned to storage.

7.201. QUOTA STRUCT

Represents a quota object.

An example XML representation of a quota:

```
<quota href="/ovirt-engine/api/datacenters/7044934e/quotas/dcad5ddc"
id="dcad5ddc">
  <name>My Quota</name>
  <description>A quota for my oVirt environment</description>
  <cluster_hard_limit_pct>0</cluster_hard_limit_pct>
  <cluster_soft_limit_pct>0</cluster_soft_limit_pct>
  <data_center href="/ovirt-engine/api/datacenters/7044934e"
id="7044934e"/>
  <storage_hard_limit_pct>0</storage_hard_limit_pct>
  <storage_soft_limit_pct>0</storage_soft_limit_pct>
</quota>
```

Table 7.263. Attributes summary

Name	Type	Summary
cluster_hard_limit_pct	Integer	
cluster_soft_limit_pct	Integer	
comment	String	Free text containing comments about this object.
data_center	DataCenter	
description	String	A human-readable description in plain text.
disks	Disk[]	
id	String	A unique identifier.

Name	Type	Summary
name	String	A human-readable name in plain text.
storage_hard_limit_pct	Integer	
storage_soft_limit_pct	Integer	
users	User[]	
vms	Vm[]	

Table 7.264. Links summary

Name	Type	Summary
permissions	Permission[]	
quota_cluster_limits	QuotaClusterLimit[]	
quota_storage_limits	QuotaStorageLimit[]	

7.202. QUOTA_CLUSTER_LIMIT STRUCT

Table 7.265. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
memory_limit	Decimal	
memory_usage	Decimal	
name	String	A human-readable name in plain text.
vcpu_limit	Integer	

Name	Type	Summary
vcpu_usage	Integer	

Table 7.266. Links summary

Name	Type	Summary
cluster	Cluster	
quota	Quota	

7.203. QUOTAMODETYPE ENUM

Table 7.267. Values summary

Name	Summary
audit	
disabled	
enabled	

7.204. QUOTASTORAGELIMIT STRUCT

Table 7.268. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
limit	Integer	
name	String	A human-readable name in plain text.
usage	Decimal	

Table 7.269. Links summary

Name	Type	Summary
<code>quota</code>	Quota	
<code>storage_domain</code>	StorageDomain	

7.205. RANGE STRUCT

Table 7.270. Attributes summary

Name	Type	Summary
<code>from</code>	String	
<code>to</code>	String	

7.206. RATE STRUCT

Determines maximum speed of consumption of bytes from random number generator device.

Table 7.271. Attributes summary

Name	Type	Summary
<code>bytes</code>	Integer	Number of bytes allowed to consume per period.
<code>period</code>	Integer	Duration of one period in milliseconds.

7.207. REPORTEDCONFIGURATION STRUCT

Table 7.272. Attributes summary

Name	Type	Summary
<code>actual_value</code>	String	
<code>expected_value</code>	String	
<code>in_sync</code>	Boolean	false when the network attachment contains uncommitted network configuration.
<code>name</code>	String	

7.208. REPORTEDDEVICE STRUCT

Table 7.273. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
ips	Ip[]	
mac	Mac	
name	String	A human-readable name in plain text.
type	ReportedDeviceType	

Table 7.274. Links summary

Name	Type	Summary
vm	Vm	

7.209. REPORTEDDEVICETYPE ENUM

Table 7.275. Values summary

Name	Summary
network	

7.210. RESOLUTIONTYPE ENUM

Table 7.276. Values summary

Name	Summary
add	
copy	

7.211. RNGDEVICE STRUCT

Random number generator (RNG) device model.

Table 7.277. Attributes summary

Name	Type	Summary
rate	Rate	Determines maximum speed of consumption of bytes from random number generator device.
source	RngSource	Backend of the random number generator device.

7.212. RNGSOURCE ENUM

Representing the random generator backend types.

Table 7.278. Values summary

Name	Summary
hwrng	Obtains random data from the /dev/hwrng (usually specialized HW generator) device.
random	Obtains random data from the /dev/random device.
urandom	Obtains random data from the /dev/urandom device.

7.212.1. urandom

Obtains random data from the **/dev/urandom** device.

This RNG source is meant to replace **random** RNG source for non-cluster-aware entities (i.e. Blank template and instance types) and entities associated with clusters with compatibility version 4.1 or higher.

7.213. ROLE STRUCT

Represents a system role.

Table 7.279. Attributes summary

Name	Type	Summary
administrative	Boolean	Defines the role as administrative-only or not.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.

Name	Type	Summary
mutable	Boolean	Defines the ability to update or delete the role.
name	String	A human-readable name in plain text.

7.213.1. mutable

Defines the ability to update or delete the role.

Roles with mutable set to **false** are predefined roles.

Table 7.280. Links summary

Name	Type	Summary
permits	Permit[]	A link to the permits sub-collection for role permits.
user	User	

7.214. ROLETYPE ENUM

Type representing whether a role is administrative or not. A user which was granted at least one administrative role is considered an administrator.

Table 7.281. Values summary

Name	Summary
admin	Administrative role.
user	User role.

7.215. SCHEDULINGPOLICY STRUCT

Table 7.282. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
default_policy	Boolean	
description	String	A human-readable description in plain text.
id	String	A unique identifier.

Name	Type	Summary
locked	Boolean	
name	String	A human-readable name in plain text.
properties	Property[]	

Table 7.283. Links summary

Name	Type	Summary
balances	Balance[]	
filters	Filter[]	
weight	Weight[]	

7.216. SCHEDULINGPOLICYUNIT STRUCT

Table 7.284. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
enabled	Boolean	
id	String	A unique identifier.
internal	Boolean	
name	String	A human-readable name in plain text.
properties	Property[]	
type	PolicyUnitType	

7.217. SCSIGENERICIO ENUM

Table 7.285. Values summary

Name	Summary
filtered	
unfiltered	

7.218. SELINUX STRUCT

Table 7.286. Attributes summary

Name	Type	Summary
mode	SeLinuxMode	

7.219. SELINUXMODE ENUM

Table 7.287. Values summary

Name	Summary
disabled	
enforcing	
permissive	

7.220. SERIALNUMBER STRUCT

Table 7.288. Attributes summary

Name	Type	Summary
policy	SerialNumberPolicy	
value	String	

7.221. SERIALNUMBERPOLICY ENUM

Table 7.289. Values summary

Name	Summary
custom	

Name	Summary
host	
vm	

7.222. SESSION STRUCT

Describes a user session to a virtual machine.

Table 7.290. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
console_user	Boolean	Indicates if this is a console session.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
ip	Ip	The IP address the user is connected from.
name	String	A human-readable name in plain text.
protocol	String	The protocol used by the session.

7.222.1. console_user

Indicates if this is a console session.

The value will be **true** for console users (SPICE or VNC), and **false** for others (such as RDP or SSH).

7.222.2. ip

The IP address the user is connected from.

Currently only available for console users.

7.222.3. protocol

The protocol used by the session.

Currently not used. Intended for info about how the user is connected: through SPICE, VNC, SSH, or RDP.

Table 7.291. Links summary

Name	Type	Summary
user	User	The user related to this session.
vm	Vm	A link to the virtual machine related to this session.

7.222.4. user

The user related to this session.

If the user is a console user, this is a link to the real Red Hat Virtualization user. Otherwise, only the user name is provided.

7.223. SKIPIFCONNECTIVITYBROKEN STRUCT

Table 7.292. Attributes summary

Name	Type	Summary
enabled	Boolean	If enabled, we will not fence a host in case more than a configurable percentage of hosts in the cluster lost connectivity as well.
threshold	Integer	Threshold for connectivity testing.

7.223.1. enabled

If enabled, we will not fence a host in case more than a configurable percentage of hosts in the cluster lost connectivity as well. This comes to prevent fencing *storm* in cases where there is a global networking issue in the cluster.

7.223.2. threshold

Threshold for connectivity testing. If at least the threshold percentage of hosts in the cluster lost connectivity then fencing will not take place.

7.224. SKIPIFSDACTIVE STRUCT

This type represents the storage related configuration in the fencing policy.

Table 7.293. Attributes summary

Name	Type	Summary
enabled	Boolean	If enabled, we will skip fencing in case the host maintains its lease in the storage.

7.224.1. enabled

If enabled, we will skip fencing in case the host maintains its lease in the storage. It means that if the host still has storage access then it won't get fenced.

7.225. SNAPSHOT STRUCT

Represents a snapshot object.

Example XML representation:

```
<snapshot id="456" href="/ovirt-engine/api/vms/123/snapshots/456">
  <actions>
    <link rel="restore" href="/ovirt-
engine/api/vms/123/snapshots/456/restore"/>
  </actions>
  <vm id="123" href="/ovirt-engine/api/vms/123"/>
  <description>Virtual Machine 1 - Snapshot A</description>
  <type>active</type>
  <date>2010-08-16T14:24:29</date>
  <persist_memorystate>>false</persist_memorystate>
</snapshot>
```

Table 7.294. Attributes summary

Name	Type	Summary
bios	Bios	Reference to virtual machine's BIOS configuration.
comment	String	Free text containing comments about this object.
console	Console	Console configured for this virtual machine.
cpu	Cpu	The configuration of the virtual machine CPU.
cpu_shares	Integer	
creation_time	Date	The virtual machine creation date.
custom_compatibility_version	Version	Virtual machine custom compatibility version.
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	Properties sent to VDSM to configure various hooks.

Name	Type	Summary
date	Date	
delete_protected	Boolean	If true , the virtual machine cannot be deleted.
description	String	A human-readable description in plain text.
display	Display	The virtual machine display configuration.
domain	Domain	Domain configured for this virtual machine.
fqdn	String	Fully qualified domain name of the virtual machine.
guest_operating_system	GuestOperatingSystem	What operating system is installed on the virtual machine.
guest_time_zone	TimeZone	What time zone is used by the virtual machine (as returned by guest agent).
high_availability	HighAvailability	The virtual machine high availability configuration.
id	String	A unique identifier.
initialization	Initialization	Reference to virtual machine's initialization configuration.
io	Io	For performance tuning of IO threading.
large_icon	Icon	Virtual machine's large icon.
lease	StorageDomainLease	Reference to the storage domain this virtual machine/template lease reside on.
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	Reference to virtual machine's memory management configuration.
migration	MigrationOptions	Reference to configuration of migration of running virtual machine to another host.
migration_downtime	Integer	Maximum time the virtual machine can be non responsive during its live migration to another host in ms.
name	String	A human-readable name in plain text.

Name	Type	Summary
next_run_configuration_exists	Boolean	Virtual machine configuration has been changed and requires restart of the virtual machine.
numa_tune_mode	NumaTuneMode	How the NUMA topology is applied.
origin	String	The origin of this virtual machine.
os	OperatingSystem	Operating system type installed on the virtual machine.
payloads	Payload[]	Optional payloads of the virtual machine, used for ISOs to configure it.
persist_memorystate	Boolean	Indicates if the content of the memory of the virtual machine is included in the snapshot.
placement_policy	VmPlacementPolicy	The configuration of the virtual machine's placement policy.
rng_device	RngDevice	Random Number Generator device configuration for this virtual machine.
run_once	Boolean	If true , the virtual machine has been started using the <i>run once</i> command, meaning it's configuration might differ from the stored one for the purpose of this single run.
serial_number	SerialNumber	Virtual machine's serial number in a cluster.
small_icon	Icon	Virtual machine's small icon.
snapshot_status	SnapshotStatus	
snapshot_type	SnapshotType	
soundcard_enabled	Boolean	If true , the sound card is added to the virtual machine.
sso	Sso	Reference to the Single Sign On configuration this virtual machine is configured for.
start_paused	Boolean	If true , the virtual machine will be initially in 'paused' state after start.

Name	Type	Summary
start_time	Date	The date in which the virtual machine was started.
stateless	Boolean	If true , the virtual machine is stateless - it's state (disks) are rolled-back after shutdown.
status	VmStatus	The current status of the virtual machine.
status_detail	String	Human readable detail of current status.
stop_reason	String	The reason the virtual machine was stopped.
stop_time	Date	The date in which the virtual machine was stopped.
time_zone	TimeZone	The virtual machine's time zone set by oVirt.
tunnel_migration	Boolean	If true , the network data transfer will be encrypted during virtual machine live migration.
type	VmType	Determines whether the virtual machine is optimized for desktop or server.
usb	Usb	Configuration of USB devices for this virtual machine (count, type).
use_latest_template_version	Boolean	If true , the virtual machine is reconfigured to the latest version of it's template when it is started.
virtio_scsi	VirtioScsi	Reference to VirtIO SCSI configuration.

7.225.1. cpu

The configuration of the virtual machine CPU.

The socket configuration can be updated without rebooting the virtual machine. The cores and the threads require a reboot.

For example, to change the number of sockets to 4 immediately, and the number of cores and threads to 2 after reboot, send the following request:

```
PUT /ovirt-engine/api/vms/123
```

With a request body:

```
<vm>
  <cpu>
    <topology>
```



```

        <sockets>4</sockets>
        <cores>2</cores>
        <threads>2</threads>
    </topology>
</cpu>
</vm>

```

7.225.2. custom_compatibility_version

Virtual machine custom compatibility version.

Enables a virtual machine to be customized to its own compatibility version. If **custom_compatibility_version** is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in, and is checked against capabilities of the host the virtual machine is planned to run on.

7.225.3. high_availability

The virtual machine high availability configuration. If set, the virtual machine will be automatically restarted when it unexpectedly goes down.

7.225.4. large_icon

Virtual machine's large icon. Either set by user or refers to image set according to operating system.

7.225.5. lease

Reference to the storage domain this virtual machine/template lease reside on.

A virtual machine running with a lease requires checking while running that the lease is not taken by another host, preventing another instance of this virtual machine from running on another host. This provides protection against split-brain in highly available virtual machines. A template can also have a storage domain defined for a lease in order to have the virtual machines created from this template to be preconfigured with this storage domain as the location of the leases.

7.225.6. memory

The virtual machine's memory, in bytes.

For example, to update a virtual machine to contain 1 Gibibyte (GiB) of memory, send the following request:

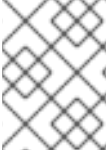
```
PUT /ovirt-engine/api/vms/123
```

With the following request body:

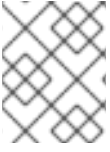
```

<vm>
  <memory>1073741824</memory>
</vm>

```

**NOTE**

Memory in the example is converted to bytes using the following formula:
 $1 \text{ GiB} = 2^{30} \text{ bytes} = 1073741824 \text{ bytes}.$

**NOTE**

Memory hot plug is supported from Red Hat Virtualization 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

7.225.7. migration_downtime

Maximum time the virtual machine can be non responsive during its live migration to another host in ms.

Set either explicitly for the virtual machine or by **engine-config -s**
DefaultMaximumMigrationDowntime=[value]

7.225.8. next_run_configuration_exists

Virtual machine configuration has been changed and requires restart of the virtual machine. Changed configuration is applied at processing the virtual machine's *shut down*.

7.225.9. origin

The origin of this virtual machine.

Possible values:

- **ovirt**
- **rhev**
- **vmware**
- **xen**
- **external**
- **hosted_engine**
- **managed_hosted_engine**
- **kvm**
- **physical_machine**
- **hyperv**

7.225.10. persist_memorystate

Indicates if the content of the memory of the virtual machine is included in the snapshot.

When a snapshot is created the default value is **true**.

7.225.11. placement_policy

The configuration of the virtual machine's placement policy.

This configuration can be updated to pin a virtual machine to one or more hosts.



NOTE

Virtual machines that are pinned to multiple hosts cannot be live migrated, but in the event of a host failure, any virtual machine configured to be highly available is automatically restarted on one of the other hosts to which the virtual machine is pinned.

For example, to pin a virtual machine to two hosts, send the following request:

```
PUT /api/vms/123
```

With a request body like this:

```
<vm>
  <high_availability>
    <enabled>>true</enabled>
    <priority>1</priority>
  </high_availability>
  <placement_policy>
    <hosts>
      <host>
        <name>Host1</name>
      </host>
      <host>
        <name>Host2</name>
      </host>
    </hosts>
    <affinity>pinned</affinity>
  </placement_policy>
</vm>
```

7.225.12. small_icon

Virtual machine's small icon. Either set by user or refers to image set according to operating system.

7.225.13. sso

Reference to the Single Sign On configuration this virtual machine is configured for. The user can be automatically signed in the virtual machine's operating system when console is opened.

7.225.14. stop_reason

The reason the virtual machine was stopped. Optionally set by user when shutting down the virtual machine.

Table 7.295. Links summary

Name	Type	Summary
affinity_labels	AffinityLabel[]	Optional.
applications	Application[]	List of applications installed on the virtual machine.
cdroms	Cdrom[]	Reference to the ISO mounted to the CDROM.
cluster	Cluster	Reference to cluster the virtual machine belongs to.
cpu_profile	CpuProfile	Reference to CPU profile used by this virtual machine.
disk_attachments	DiskAttachment[]	References the disks attached to the virtual machine.
external_host_provider	ExternalHostProvider	
floppies	Floppy[]	Reference to the ISO mounted to the floppy.
graphics_consoles	GraphicsConsole[]	List of graphics consoles configured for this virtual machine.
host	Host	Reference to the host the virtual machine is running on.
host_devices	HostDevice[]	References devices associated to this virtual machine.
instance_type	InstanceType	The virtual machine configuration can be optionally predefined via one of the instance types.
katello_errata	KatelloErratum[]	Lists all the Katello errata assigned to the virtual machine.
nics	Nic[]	References the list of network interface devices on the virtual machine.
numa_nodes	NumaNode[]	Refers to the NUMA Nodes configuration used by this virtual machine.
original_template	Template	References the original template used to create the virtual machine.
permissions	Permission[]	Permissions set for this virtual machine.
quota	Quota	Reference to quota configuration set for this virtual machine.

Name	Type	Summary
reported_devices	ReportedDevice[]	
sessions	Session[]	List of user sessions opened for this virtual machine.
snapshots	Snapshot[]	Refers to all snapshots taken from the virtual machine.
statistics	Statistic[]	Statistics data collected from this virtual machine.
storage_domain	StorageDomain	Reference to storage domain the virtual machine belongs to.
tags	Tag[]	
template	Template	Reference to the template the virtual machine is based on.
vm	Vm	
vm_pool	VmPool	Reference to the pool the virtual machine is optionally member of.
watchdogs	Watchdog[]	Refers to the Watchdog configuration.

7.225.15. affinity_labels

Optional. Used for labeling of sub-clusters.

7.225.16. katello_errata

Lists all the Katello errata assigned to the virtual machine.

```
GET /ovirt-engine/api/vms/123/katelloerrata
```

You will receive response in XML like this one:

```
<katello_errata>
  <katello_erratum href="/ovirt-engine/api/katelloerrata/456" id="456">
    <name>RHBA-2013:XYZ</name>
    <description>The description of the erratum</description>
    <title>some bug fix update</title>
    <type>bugfix</type>
    <issued>2013-11-20T02:00:00.000+02:00</issued>
    <solution>Few guidelines regarding the solution</solution>
    <summary>Updated packages that fix one bug are now available for
XYZ</summary>
    <packages>
      <package>
        <name>libipa_hbac-1.9.2-82.11.el6_4.i686</name>
```

```

        </package>
        ...
    </packages>
</katello_erratum>
...
</katello_errata>

```

7.225.17. original_template

References the original template used to create the virtual machine.

If the virtual machine is cloned from a template or another virtual machine, the **template** links to the Blank template, and the **original_template** is used to track history.

Otherwise the **template** and **original_template** are the same.

7.226. SNAPSHOTSTATUS ENUM

Table 7.296. Values summary

Name	Summary
in_preview	
locked	
ok	

7.227. SNAPSHOTTYPE ENUM

Table 7.297. Values summary

Name	Summary
active	
preview	
regular	
stateless	

7.228. SPECIALOBJECTS STRUCT

This type contains references to special objects, such as blank templates and the root of a hierarchy of tags.

Table 7.298. Links summary

Name	Type	Summary
blank_template	Template	A reference to a blank template.
root_tag	Tag	A reference to the root of a hierarchy of tags.

7.229. SPM STRUCT

Table 7.299. Attributes summary

Name	Type	Summary
priority	Integer	
status	SpmStatus	

7.230. SPMSTATUS ENUM

Table 7.300. Values summary

Name	Summary
contending	
none	
spm	

7.231. SSH STRUCT

Table 7.301. Attributes summary

Name	Type	Summary
authentication_method	SshAuthenticationMethod	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
fingerprint	String	
id	String	A unique identifier.

Name	Type	Summary
name	String	A human-readable name in plain text.
port	Integer	
user	User	

7.232. SSHAUTHENTICATIONMETHOD ENUM

Table 7.302. Values summary

Name	Summary
password	
publickey	

7.233. SSHPUBLICKEY STRUCT

Table 7.303. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
content	String	
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.304. Links summary

Name	Type	Summary
user	User	

7.234. SSO STRUCT

Table 7.305. Attributes summary

Name	Type	Summary
methods	Method[]	

7.235. SSOMETHOD ENUM

Table 7.306. Values summary

Name	Summary
guest_agent	

7.236. STATISTIC STRUCT

A generic type used for all kinds of statistics.

Statistic contains the statistics values for various entities. The following object contain statistics:

- Disk
- Host
- HostNic
- NumaNode
- Nic
- Vm
- GlusterBrick
- Step
- GlusterVolume

An example of a XML representation:

```
<statistics>
  <statistic id="1234" href="/ovirt-
engine/api/hosts/1234/nics/1234/statistics/1234">
    <name>data.current.rx</name>
    <description>Receive data rate</description>
    <values type="DECIMAL">
      <value>
        <datum>0</datum>
      </value>
    </values>
    <type>GAUGE</type>
    <unit>BYTES_PER_SECOND</unit>
    <host_nic id="1234" href="/ovirt-engine/api/hosts/1234/nics/1234"/>
  </statistic>
</statistics>
```

```

</statistic>
...
</statistics>

```

**NOTE**

This statistics sub-collection is read-only.

Table 7.307. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
kind	StatisticKind	The type of statistic measures.
name	String	A human-readable name in plain text.
type	ValueType	The data type for the statistical values that follow.
unit	StatisticUnit	The unit or rate to measure of the statistical values.
values	Value[]	A data set that contains datum .

Table 7.308. Links summary

Name	Type	Summary
brick	GlusterBrick	
disk	Disk	A relationship to the containing disk resource.
gluster_volume	GlusterVolume	
host	Host	
host_nic	HostNic	A reference to the host NIC.
host_numa_node	NumaNode	
nic	Nic	

Name	Type	Summary
step	Step	
vm	Vm	

7.237. STATISTICKIND ENUM

Table 7.309. Values summary

Name	Summary
counter	
gauge	

7.238. STATISTICUNIT ENUM

Table 7.310. Values summary

Name	Summary
bits_per_second	
bytes	
bytes_per_second	
count_per_second	
none	
percent	
seconds	

7.239. STEP STRUCT

Represents a step, which is part of **job** execution. Step is used to describe and track a specific execution unit which is part of a wider sequence. Some steps support reporting their progress.

Table 7.311. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
end_time	Date	The end time of the step.
external	Boolean	Indicates if the step is originated by an external system.
external_type	ExternalSystemType	The external system which is referenced by the step.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
number	Integer	The order of the step in current hierarchy level.
progress	Integer	The step progress (if reported) in percentages.
start_time	Date	The start time of the step.
status	StepStatus	The status of the step.
type	StepEnum	The type of the step.

7.239.1. external

Indicates if the step is originated by an external system. External steps are managed externally, by the creator of the step.

Table 7.312. Links summary

Name	Type	Summary
execution_host	Host	The host used for the step execution (optional).
job	Job	References the job which is the top of the current step hierarchy.
parent_step	Step	References the parent step of the current step in the hierarchy.
statistics	Statistic[]	

7.240. STEPENUM ENUM

Type representing a step type.

Table 7.313. Values summary

Name	Summary
executing	The executing step type.
finalizing	The finalizing step type.
rebalancing_ volume	The rebalancing volume step type.
removing_bri cks	The removing bricks step type.
unknown	The unknown step type.
validating	The validation step type.

7.240.1. executing

The executing step type. Used to track the main execution block of the job. Usually it will be a parent step of several sub-steps which describe portions of the execution step.

7.240.2. finalizing

The finalizing step type. Describes the post-execution steps requires to complete the **job**.

7.240.3. rebalancing_volume

The **rebalancing volume** step type. Describes a step type which is part of **Gluster** flow.

7.240.4. removing_bricks

The **removing bricks** step type. Describes a step type which is part of **Gluster** flow.

7.240.5. unknown

The unknown step type. Describes a step type which its origin is unknown.

7.240.6. validating

The validation step type. Used to verify the correctness of parameters and the validity of the parameters prior to the execution.

7.241. STEPSTATUS ENUM

Represents the status of the step.

Table 7.314. Values summary

Name	Summary
aborted	The aborted step status.
failed	The failed step status.
finished	The finished step status.
started	The started step status.
unknown	The unknown step status.

7.241.1. aborted

The aborted step status. This status is applicable for an external step that was forcibly aborted.

7.241.2. finished

The finished step status. This status describes a completed step execution.

7.241.3. started

The started step status. This status represents a step which is currently being executed.

7.241.4. unknown

The unknown step status. This status represents steps which their resolution is not known, i.e. steps that were executed before the system was unexpectedly restarted.

7.242. STORAGECONNECTION STRUCT

Represents a storage server connection.

Example XML representation:

```
<storage_connection id="123">
  <address>my nfs.example.com</address>
  <type>nfs</type>
  <path>/exports/mydata</path>
</storage_connection>
```

Table 7.315. Attributes summary

Name	Type	Summary
address	String	
comment	String	Free text containing comments about this object.

Name	Type	Summary
description	String	A human-readable description in plain text.
id	String	A unique identifier.
mount_options	String	
name	String	A human-readable name in plain text.
nfs_retrans	Integer	
nfs_timeo	Integer	
nfs_version	NfsVersion	
password	String	
path	String	
port	Integer	
portal	String	
target	String	
type	StorageType	
username	String	
vfs_type	String	

Table 7.316. Links summary

Name	Type	Summary
host	Host	

7.243. STORAGECONNECTIONEXTENSION STRUCT

Table 7.317. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.
password	String	
target	String	
username	String	

Table 7.318. Links summary

Name	Type	Summary
host	Host	

7.244. STORAGEDOMAIN STRUCT

Storage domain.

An XML representation of a NFS storage domain with identifier **123**:

```
<storage_domain href="/ovirt-engine/api/storagedomains/123" id="123">
  <name>mydata</name>
  <description>My data</description>
  <available>38654705664</available>
  <committed>1073741824</committed>
  <critical_space_action_blocker>5</critical_space_action_blocker>
  <external_status>ok</external_status>
  <master>true</master>
  <storage>
    <address>my nfs.example.com</address>
    <nfs_version>v3</nfs_version>
    <path>/exports/mydata</path>
    <type>nfs</type>
  </storage>
  <storage_format>v3</storage_format>
  <type>data</type>
  <used>13958643712</used>
  <warning_low_space_indicator>10</warning_low_space_indicator>
  <wipe_after_delete>>false</wipe_after_delete>
  <data_centers>
```



```

    <data_center href="/ovirt-engine/api/datacenters/456" id="456"/>
  </data_centers>
</storage_domain>

```

Table 7.319. Attributes summary

Name	Type	Summary
available	Integer	
comment	String	Free text containing comments about this object.
committed	Integer	
critical_space_action_blocker	Integer	
description	String	A human-readable description in plain text.
discard_after_delete	Boolean	Indicates whether disks' blocks on block storage domains will be discarded right before they are deleted.
external_status	ExternalStatus	
id	String	A unique identifier.
import	Boolean	
master	Boolean	
name	String	A human-readable name in plain text.
status	StorageDomainStatus	
storage	HostStorage	
storage_format	StorageFormat	
supports_discard	Boolean	Indicates whether a block storage domain supports discard operations.
supports_discard_zeroes_data	Boolean	Indicates whether a block storage domain supports the property that discard zeroes the data.

Name	Type	Summary
type	StorageDomainType	
used	Integer	
warning_low_space_indicator	Integer	
wipe_after_delete	Boolean	Serves as the default value of wipe_after_delete for disks on this storage domain .

7.244.1. discard_after_delete

Indicates whether **disks**' blocks on block **storage domains** will be discarded right before they are deleted.

If true, and a disk on this storage domain has its **wipe_after_delete** value enabled, then when the disk is deleted:

1. It is first wiped.
2. Then its blocks are discarded.
3. Finally it is deleted.

Note that:

- Discard after delete will always be **false** for non block storage types.
- Discard after delete can be set to **true** only if the storage domain **supports discard**.

7.244.2. supports_discard

Indicates whether a block storage domain supports discard operations. A **storage domain** only supports discard if all of the **logical units** that it is built from support discard; that is, if each logical unit's **discard_max_size** value is greater than 0. This is one of the conditions necessary for a virtual disk in this storage domain to have its **pass_discard** attribute enabled. Since the engine cannot check if the underlying block device supports discard for file storage domains, this attribute will not be reported for them at all.

7.244.3. supports_discard_zeroes_data

Indicates whether a block storage domain supports the property that discard zeroes the data. A **storage domain** only supports the property that discard zeroes the data if all of the **logical units** that it is built from support it; that is, if each logical unit's **discard_zeroes_data** value is true. This is one of the conditions necessary for a virtual disk in this storage domain to have both **wipe_after_delete** and **pass_discard** attributes enabled. Since the engine cannot check if the underlying block device supports the property that discard zeroes the data for file storage domains, this attribute will not be reported for them at all.

7.244.4. wipe_after_delete

Serves as the default value of `wipe_after_delete` for `disks` on this `storage domain`.

That is, newly created disks will get their `wipe_after_delete` value from their storage domains by default. Note that the configuration value `SANWipeAfterDelete` serves as the default value of block storage domains' `wipe_after_delete` value.

Table 7.320. Links summary

Name	Type	Summary
<code>data_center</code>	<code>DataCenter</code>	A link to the data center that the storage domain is attached to.
<code>data_centers</code>	<code>DataCenter[]</code>	A set of links to the data centers that the storage domain is attached to.
<code>disk_profiles</code>	<code>DiskProfile[]</code>	
<code>disk_snapshots</code>	<code>DiskSnapshot[]</code>	
<code>disks</code>	<code>Disk[]</code>	
<code>files</code>	<code>File[]</code>	
<code>host</code>	<code>Host</code>	Host is only relevant at creation time.
<code>images</code>	<code>Image[]</code>	
<code>permissions</code>	<code>Permission[]</code>	
<code>storage_connections</code>	<code>StorageConnection[]</code>	
<code>templates</code>	<code>Template[]</code>	
<code>vms</code>	<code>Vm[]</code>	

7.244.5. data_center

A link to the data center that the storage domain is attached to. This is preserved for backwards compatibility only, as the storage domain may be attached to multiple data centers (if it is an ISO domain). Use the `dataCenters` element instead.

7.245. STORAGEDOMAINLEASE STRUCT

Represents a lease residing on a storage domain.

A lease is a [Sanlock](#) resource residing on a special volume on the storage domain, this Sanlock resource is used to provide storage base locking.

Table 7.321. Links summary

Name	Type	Summary
<code>storage_domain</code>	StorageDomain	Reference to the storage domain on which the lock resides on.

7.246. STORAGEDOMAINSTATUS ENUM

Table 7.322. Values summary

Name	Summary
<code>activating</code>	
<code>active</code>	
<code>detaching</code>	
<code>inactive</code>	
<code>locked</code>	
<code>maintenance</code>	
<code>mixed</code>	
<code>preparing_for_maintenance</code>	
<code>unattached</code>	
<code>unknown</code>	

7.247. STORAGEDOMAINSTYPE ENUM

Table 7.323. Values summary

Name	Summary
<code>data</code>	
<code>export</code>	

Name	Summary
image	
iso	
volume	

7.248. STORAGEFORMAT ENUM

Table 7.324. Values summary

Name	Summary
v1	
v2	
v3	
v4	Version 4 of the storage domain format.

7.249. STORAGETYPE ENUM

Type representing a storage domain type.

Table 7.325. Values summary

Name	Summary
cinder	Cinder storage domain.
fcp	Fibre-Channel storage domain.
glance	Glance storage domain.
glusterfs	Gluster-FS storage domain.
iscsi	iSCSI storage domain.
localfs	Storage domain on Local storage.
nfs	NFS storage domain.
posixfs	POSIX-FS storage domain.

7.249.1. cinder

Cinder storage domain. For more details on Cinder please go to [Cinder](#).

7.249.2. glance

Glance storage domain. For more details on Glance please go to [Glance](#).

7.249.3. glusterfs

Gluster-FS storage domain. For more details on Gluster please go to [Gluster](#).

7.250. SWITCHTYPE ENUM

Describes all switch types supported by the Manager.

Table 7.326. Values summary

Name	Summary
legacy	The native switch type.
ovs	The Open vSwitch type.

7.251. TAG STRUCT

Represents a tag in the system.

Table 7.327. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.328. Links summary

Name	Type	Summary
group	Group	Reference to the group which has this tag assigned.
host	Host	Reference to the host which has this tag assigned.

Name	Type	Summary
parent	Tag	Reference to the parent tag of this tag.
template	Template	Reference to the template which has this tag assigned.
user	User	Reference to the user who has this tag assigned.
vm	Vm	Reference to the virtual machine which has this tag assigned.

7.252. TEMPLATE STRUCT

Type representing a virtual machine template. This allows a rapid instantiation of virtual machines with common configuration and disk states.

Table 7.329. Attributes summary

Name	Type	Summary
bios	Bios	Reference to virtual machine's BIOS configuration.
comment	String	Free text containing comments about this object.
console	Console	Console configured for this virtual machine.
cpu	Cpu	The configuration of the virtual machine CPU.
cpu_shares	Integer	
creation_time	Date	The virtual machine creation date.
custom_compatibility_version	Version	Virtual machine custom compatibility version.
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	Properties sent to VDSM to configure various hooks.
delete_protected	Boolean	If true , the virtual machine cannot be deleted.

Name	Type	Summary
description	String	A human-readable description in plain text.
display	Display	The virtual machine display configuration.
domain	Domain	Domain configured for this virtual machine.
high_availability	HighAvailability	The virtual machine high availability configuration.
id	String	A unique identifier.
initialization	Initialization	Reference to virtual machine's initialization configuration.
io	Io	For performance tuning of IO threading.
large_icon	Icon	Virtual machine's large icon.
lease	StorageDomainLease	Reference to the storage domain this virtual machine/template lease reside on.
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	Reference to virtual machine's memory management configuration.
migration	MigrationOptions	Reference to configuration of migration of running virtual machine to another host.
migration_downtime	Integer	Maximum time the virtual machine can be non responsive during its live migration to another host in ms.
name	String	A human-readable name in plain text.
origin	String	The origin of this virtual machine.
os	OperatingSystem	Operating system type installed on the virtual machine.
rng_device	RngDevice	Random Number Generator device configuration for this virtual machine.
serial_number	SerialNumber	Virtual machine's serial number in a cluster.
small_icon	Icon	Virtual machine's small icon.

Name	Type	Summary
soundcard_enabled	Boolean	If true , the sound card is added to the virtual machine.
sso	Sso	Reference to the Single Sign On configuration this virtual machine is configured for.
start_paused	Boolean	If true , the virtual machine will be initially in 'paused' state after start.
stateless	Boolean	If true , the virtual machine is stateless - it's state (disks) are rolled-back after shutdown.
status	TemplateStatus	The status of the template.
time_zone	TimeZone	The virtual machine's time zone set by oVirt.
tunnel_migration	Boolean	If true , the network data transfer will be encrypted during virtual machine live migration.
type	VmType	Determines whether the virtual machine is optimized for desktop or server.
usb	Usb	Configuration of USB devices for this virtual machine (count, type).
version	TemplateVersion	Indicates whether this is a base version or a sub version of another template.
virtio_scsi	VirtioScsi	Reference to VirtIO SCSI configuration.
vm	Vm	The virtual machine configuration associated with this template.

7.252.1. cpu

The configuration of the virtual machine CPU.

The socket configuration can be updated without rebooting the virtual machine. The cores and the threads require a reboot.

For example, to change the number of sockets to 4 immediately, and the number of cores and threads to 2 after reboot, send the following request:

```
PUT /ovirt-engine/api/vms/123
```

With a request body:

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

7.252.2. custom_compatibility_version

Virtual machine custom compatibility version.

Enables a virtual machine to be customized to its own compatibility version. If **custom_compatibility_version** is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in, and is checked against capabilities of the host the virtual machine is planned to run on.

7.252.3. high_availability

The virtual machine high availability configuration. If set, the virtual machine will be automatically restarted when it unexpectedly goes down.

7.252.4. large_icon

Virtual machine's large icon. Either set by user or refers to image set according to operating system.

7.252.5. lease

Reference to the storage domain this virtual machine/template lease reside on.

A virtual machine running with a lease requires checking while running that the lease is not taken by another host, preventing another instance of this virtual machine from running on another host. This provides protection against split-brain in highly available virtual machines. A template can also have a storage domain defined for a lease in order to have the virtual machines created from this template to be preconfigured with this storage domain as the location of the leases.

7.252.6. memory

The virtual machine's memory, in bytes.

For example, to update a virtual machine to contain 1 Gibibyte (GiB) of memory, send the following request:

```
PUT /ovirt-engine/api/vms/123
```

With the following request body:

```
<vm>
  <memory>1073741824</memory>
</vm>
```

**NOTE**

Memory in the example is converted to bytes using the following formula:
 1 GiB = 2^{30} bytes = 1073741824 bytes.

**NOTE**

Memory hot plug is supported from Red Hat Virtualization 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

7.252.7. migration_downtime

Maximum time the virtual machine can be non responsive during its live migration to another host in ms.

Set either explicitly for the virtual machine or by **engine-config -s DefaultMaximumMigrationDowntime=[value]**

7.252.8. origin

The origin of this virtual machine.

Possible values:

- **ovirt**
- **rhev**
- **vmware**
- **xen**
- **external**
- **hosted_engine**
- **managed_hosted_engine**
- **kvm**
- **physical_machine**
- **hyperv**

7.252.9. small_icon

Virtual machine's small icon. Either set by user or refers to image set according to operating system.

7.252.10. sso

Reference to the Single Sign On configuration this virtual machine is configured for. The user can be automatically signed in the virtual machine's operating system when console is opened.

Table 7.330. Links summary

Name	Type	Summary
cdroms	Cdrom[]	References to the CD-ROM devices attached to the template.
cluster	Cluster	Reference to cluster the virtual machine belongs to.
cpu_profile	CpuProfile	Reference to CPU profile used by this virtual machine.
disk_attachments	DiskAttachment[]	References to the disks attached to the template.
graphics_consoles	GraphicsConsole[]	References to the graphic consoles attached to the template.
nics	Nic[]	References to the network interfaces attached to the template.
permissions	Permission[]	References to the user permissions attached to the template.
quota	Quota	Reference to quota configuration set for this virtual machine.
storage_domain	StorageDomain	Reference to storage domain the virtual machine belongs to.
tags	Tag[]	References to the tags attached to the template.
watchdogs	Watchdog[]	References to the watchdog devices attached to the template.

7.253. TEMPLATESTATUS ENUM

Type representing a status of a virtual machine template.

Table 7.331. Values summary

Name	Summary
illegal	This status indicates that at least one of the disks of the template is illegal.
locked	This status indicates that some operation that prevents other operations with the template is being executed.
ok	This status indicates that the template is valid and ready for use.

7.254. TEMPLATEVERSION STRUCT

Type representing a version of a virtual machine template.

Table 7.332. Attributes summary

Name	Type	Summary
<code>version_name</code>	String	The name of this version.
<code>version_number</code>	Integer	The index of this version in the versions hierarchy of the template.

7.254.1. version_number

The index of this version in the versions hierarchy of the template. The index 1 represents the original version of a template that is also called base version.

Table 7.333. Links summary

Name	Type	Summary
<code>base_template</code>	Template	References the template that this version is associated with.

7.255. TICKET STRUCT

Type representing a ticket that allows virtual machine access.

Table 7.334. Attributes summary

Name	Type	Summary
<code>expiry</code>	Integer	Time to live for the ticket in seconds.
<code>value</code>	String	The virtual machine access ticket.

7.256. TIMEZONE STRUCT

Time zone representation.

Table 7.335. Attributes summary

Name	Type	Summary
<code>name</code>	String	Name of the time zone.
<code>utc_offset</code>	String	Offset from https://en .

7.256.1. utc_offset

Offset from [UTC](#).

7.257. TRANSPARENTHUGEPAGES STRUCT

Type representing a transparent huge pages (THP) support.

Table 7.336. Attributes summary

Name	Type	Summary
enabled	Boolean	Enable THP support.

7.258. TRANSPORTTYPE ENUM

Protocol used to access a Gluster volume.

Table 7.337. Values summary

Name	Summary
rdma	Remote direct memory access.
tcp	TCP.

7.259. UNMANAGEDNETWORK STRUCT

Table 7.338. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.339. Links summary

Name	Type	Summary
host	Host	
host_nic	HostNic	

7.260. USB STRUCT

Table 7.340. Attributes summary

Name	Type	Summary
enabled	Boolean	
type	UsbType	

7.261. USBTYPE ENUM

Table 7.341. Values summary

Name	Summary
legacy	A legacy USB type.
native	

7.261.1. legacy

A legacy USB type.

This USB type has been deprecated since version 3.6 of the engine, and has been completely removed in version 4.1. It is preserved only to avoid syntax errors in existing scripts. If it is used it will be automatically replaced by **native**.

7.262. USER STRUCT

Represents a user in the system.

Table 7.342. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
department	String	
description	String	A human-readable description in plain text.
domain_entry_id	String	
email	String	
id	String	A unique identifier.
last_name	String	

Name	Type	Summary
logged_in	Boolean	
name	String	A human-readable name in plain text.
namespace	String	Namespace where the user resides.
password	String	
principal	String	Similar to user_name .
user_name	String	The user's username.

7.262.1. namespace

Namespace where the user resides. When using the authorization provider that stores users in the LDAP server, this attribute equals the naming context of the LDAP server. See <https://github.com/oVirt/ovirt-engine-extension-aaa-ldap> for more information. When using the built-in authorization provider that stores users in the database this attribute is ignored. See <https://github.com/oVirt/ovirt-engine-extension-aaa-jdbc> for more information.

7.262.2. principal

Similar to **user_name**. The format depends on the LDAP provider. With most LDAP providers it is the value of the **uid** LDAP attribute. In the case of Active Directory it is the User Principal Name (UPN).

7.262.3. user_name

The user's username. The format depends on authorization provider type. In most LDAP providers it is the value of the **uid** LDAP attribute. In Active Directory it is the User Principal Name (UPN). **UPN** or **uid** must be followed by the authorization provider name. For example, in the case of LDAP's **uid** attribute it is: **myuser@myextension-authz**. In the case of Active Directory using **UPN** it is: **myuser@mysubdomain.mydomain.com@myextension-authz**. This attribute is a required parameter when adding a new user.

Table 7.343. Links summary

Name	Type	Summary
domain	Domain	
groups	Group[]	
permissions	Permission[]	
roles	Role[]	A link to the roles sub-collection for user resources.

Name	Type	Summary
ssh_public_keys	SshPublicKey[]	
tags	Tag[]	A link to the tags sub-collection for user resources.

7.263. VALUE STRUCT

Table 7.344. Attributes summary

Name	Type	Summary
datum	Decimal	
detail	String	

7.264. VALUETYPE ENUM

Table 7.345. Values summary

Name	Summary
decimal	
integer	
string	

7.265. VCPUPIN STRUCT

Table 7.346. Attributes summary

Name	Type	Summary
cpu_set	String	
vcpu	Integer	

7.266. VENDOR STRUCT

Table 7.347. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
name	String	A human-readable name in plain text.

7.267. VERSION STRUCT

Table 7.348. Attributes summary

Name	Type	Summary
build	Integer	
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
full_version	String	
id	String	A unique identifier.
major	Integer	
minor	Integer	
name	String	A human-readable name in plain text.
revision	Integer	

7.268. VIRTIO SCSI STRUCT

Type representing the support of virtio-SCSI. If it supported we use virtio driver for SCSI guest device.

Table 7.349. Attributes summary

Name	Type	Summary
enabled	Boolean	Enable Virtio SCSI support.

7.269. VIRTUALNUMANODE STRUCT

Represents the virtual NUMA node.

An example XML representation:

```
<vm_numa_node href="/ovirt-engine/api/vms/123/numanodes/456" id="456">
  <cpu>
    <cores>
      <core>
        <index>0</index>
      </core>
    </cores>
  </cpu>
  <index>0</index>
  <memory>1024</memory>
  <numa_node_pins>
    <numa_node_pin>
      <index>0</index>
    </numa_node_pin>
  </numa_node_pins>
  <vm href="/ovirt-engine/api/vms/123" id="123" />
</vm_numa_node>
```

Table 7.350. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
cpu	Cpu	
description	String	A human-readable description in plain text.
id	String	A unique identifier.
index	Integer	
memory	Integer	Memory of the NUMA node in MB.
name	String	A human-readable name in plain text.
node_distance	String	
numa_node_pins	NumaNodePin[]	

Table 7.351. Links summary

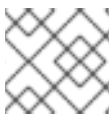
Name	Type	Summary
host	Host	
statistics	Statistic[]	Each host NUMA node resource exposes a statistics sub-collection for host NUMA node specific statistics.
vm	Vm	

7.269.1. statistics

Each host NUMA node resource exposes a statistics sub-collection for host NUMA node specific statistics.

An example of an XML representation:

```
<statistics>
  <statistic href="/ovirt-
engine/api/hosts/123/numanodes/456/statistics/789" id="789">
    <name>memory.total</name>
    <description>Total memory</description>
    <kind>gauge</kind>
    <type>integer</type>
    <unit>bytes</unit>
    <values>
      <value>
        <datum>25165824000</datum>
      </value>
    </values>
    <host_numa_node href="/ovirt-engine/api/hosts/123/numanodes/456"
id="456" />
  </statistic>
  ...
</statistics>
```



NOTE

This statistics sub-collection is read-only.

The following list shows the statistic types for a host NUMA node:

Name	Description
memory.total	Total memory in bytes on the NUMA node.
memory.used	Memory in bytes used on the NUMA node.
memory.free	Memory in bytes free on the NUMA node.

Name	Description
<code>cpu.current.user</code>	Percentage of CPU usage for user slice.
<code>cpu.current.system</code>	Percentage of CPU usage for system.
<code>cpu.current.idle</code>	Percentage of idle CPU usage.

7.270. VLAN STRUCT

Type representing a Virtual LAN (VLAN) type.

Table 7.352. Attributes summary

Name	Type	Summary
<code>id</code>	Integer	Virtual LAN ID.

7.271. VM STRUCT

Represents a virtual machine.

Table 7.353. Attributes summary

Name	Type	Summary
<code>bios</code>	Bios	Reference to virtual machine's BIOS configuration.
<code>comment</code>	String	Free text containing comments about this object.
<code>console</code>	Console	Console configured for this virtual machine.
<code>cpu</code>	Cpu	The configuration of the virtual machine CPU.
<code>cpu_shares</code>	Integer	
<code>creation_time</code>	Date	The virtual machine creation date.
<code>custom_compatibility_version</code>	Version	Virtual machine custom compatibility version.
<code>custom_cpu_model</code>	String	

Name	Type	Summary
custom_emulated_machine	String	
custom_properties	CustomProperty[]	Properties sent to VDSM to configure various hooks.
delete_protected	Boolean	If true , the virtual machine cannot be deleted.
description	String	A human-readable description in plain text.
display	Display	The virtual machine display configuration.
domain	Domain	Domain configured for this virtual machine.
fqdn	String	Fully qualified domain name of the virtual machine.
guest_operating_system	GuestOperatingSystem	What operating system is installed on the virtual machine.
guest_time_zone	TimeZone	What time zone is used by the virtual machine (as returned by guest agent).
high_availability	HighAvailability	The virtual machine high availability configuration.
id	String	A unique identifier.
initialization	Initialization	Reference to virtual machine's initialization configuration.
io	Io	For performance tuning of IO threading.
large_icon	Icon	Virtual machine's large icon.
lease	StorageDomainLease	Reference to the storage domain this virtual machine/template lease reside on.
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	Reference to virtual machine's memory management configuration.
migration	MigrationOptions	Reference to configuration of migration of running virtual machine to another host.

Name	Type	Summary
<code>migration_downtime</code>	Integer	Maximum time the virtual machine can be non responsive during its live migration to another host in ms.
<code>name</code>	String	A human-readable name in plain text.
<code>next_run_configuration_exists</code>	Boolean	Virtual machine configuration has been changed and requires restart of the virtual machine.
<code>numa_tune_mode</code>	NumaTuneMode	How the NUMA topology is applied.
<code>origin</code>	String	The origin of this virtual machine.
<code>os</code>	OperatingSystem	Operating system type installed on the virtual machine.
<code>payloads</code>	Payload[]	Optional payloads of the virtual machine, used for ISOs to configure it.
<code>placement_policy</code>	VmPlacementPolicy	The configuration of the virtual machine's placement policy.
<code>rng_device</code>	RngDevice	Random Number Generator device configuration for this virtual machine.
<code>run_once</code>	Boolean	If true , the virtual machine has been started using the <code>run once</code> command, meaning it's configuration might differ from the stored one for the purpose of this single run.
<code>serial_number</code>	SerialNumber	Virtual machine's serial number in a cluster.
<code>small_icon</code>	Icon	Virtual machine's small icon.
<code>soundcard_enabled</code>	Boolean	If true , the sound card is added to the virtual machine.
<code>sso</code>	Sso	Reference to the Single Sign On configuration this virtual machine is configured for.
<code>start_paused</code>	Boolean	If true , the virtual machine will be initially in 'paused' state after start.
<code>start_time</code>	Date	The date in which the virtual machine was started.

Name	Type	Summary
stateless	Boolean	If true , the virtual machine is stateless - it's state (disks) are rolled-back after shutdown.
status	VmStatus	The current status of the virtual machine.
status_detail	String	Human readable detail of current status.
stop_reason	String	The reason the virtual machine was stopped.
stop_time	Date	The date in which the virtual machine was stopped.
time_zone	TimeZone	The virtual machine's time zone set by oVirt.
tunnel_migration	Boolean	If true , the network data transfer will be encrypted during virtual machine live migration.
type	VmType	Determines whether the virtual machine is optimized for desktop or server.
usb	Usb	Configuration of USB devices for this virtual machine (count, type).
use_latest_template_version	Boolean	If true , the virtual machine is reconfigured to the latest version of it's template when it is started.
virtio_scsi	VirtioScsi	Reference to VirtIO SCSI configuration.

7.271.1. cpu

The configuration of the virtual machine CPU.

The socket configuration can be updated without rebooting the virtual machine. The cores and the threads require a reboot.

For example, to change the number of sockets to 4 immediately, and the number of cores and threads to 2 after reboot, send the following request:

```
PUT /ovirt-engine/api/vms/123
```

With a request body:

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
```



```

        <threads>2</threads>
    </topology>
</cpu>
</vm>

```

7.271.2. custom_compatibility_version

Virtual machine custom compatibility version.

Enables a virtual machine to be customized to its own compatibility version. If **custom_compatibility_version** is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in, and is checked against capabilities of the host the virtual machine is planned to run on.

7.271.3. high_availability

The virtual machine high availability configuration. If set, the virtual machine will be automatically restarted when it unexpectedly goes down.

7.271.4. large_icon

Virtual machine's large icon. Either set by user or refers to image set according to operating system.

7.271.5. lease

Reference to the storage domain this virtual machine/template lease reside on.

A virtual machine running with a lease requires checking while running that the lease is not taken by another host, preventing another instance of this virtual machine from running on another host. This provides protection against split-brain in highly available virtual machines. A template can also have a storage domain defined for a lease in order to have the virtual machines created from this template to be preconfigured with this storage domain as the location of the leases.

7.271.6. memory

The virtual machine's memory, in bytes.

For example, to update a virtual machine to contain 1 Gibibyte (GiB) of memory, send the following request:

```
PUT /ovirt-engine/api/vms/123
```

With the following request body:

```

<vm>
  <memory>1073741824</memory>
</vm>

```

**NOTE**

Memory in the example is converted to bytes using the following formula:
 $1 \text{ GiB} = 2^{30} \text{ bytes} = 1073741824 \text{ bytes}$.

**NOTE**

Memory hot plug is supported from Red Hat Virtualization 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

7.271.7. migration_downtime

Maximum time the virtual machine can be non responsive during its live migration to another host in ms.

Set either explicitly for the virtual machine or by **engine-config -s**
DefaultMaximumMigrationDowntime=[value]

7.271.8. next_run_configuration_exists

Virtual machine configuration has been changed and requires restart of the virtual machine. Changed configuration is applied at processing the virtual machine's *shut down*.

7.271.9. origin

The origin of this virtual machine.

Possible values:

- **ovirt**
- **rhev**
- **vmware**
- **xen**
- **external**
- **hosted_engine**
- **managed_hosted_engine**
- **kvm**
- **physical_machine**
- **hyperv**

7.271.10. placement_policy

The configuration of the virtual machine's placement policy.

This configuration can be updated to pin a virtual machine to one or more hosts.

**NOTE**

Virtual machines that are pinned to multiple hosts cannot be live migrated, but in the event of a host failure, any virtual machine configured to be highly available is automatically restarted on one of the other hosts to which the virtual machine is pinned.

For example, to pin a virtual machine to two hosts, send the following request:

```
PUT /api/vms/123
```

With a request body like this:

```
<vm>
  <high_availability>
    <enabled>>true</enabled>
    <priority>1</priority>
  </high_availability>
  <placement_policy>
    <hosts>
      <host>
        <name>Host1</name>
      </host>
      <host>
        <name>Host2</name>
      </host>
    </hosts>
    <affinity>pinned</affinity>
  </placement_policy>
</vm>
```

7.271.11. small_icon

Virtual machine's small icon. Either set by user or refers to image set according to operating system.

7.271.12. sso

Reference to the Single Sign On configuration this virtual machine is configured for. The user can be automatically signed in the virtual machine's operating system when console is opened.

7.271.13. stop_reason

The reason the virtual machine was stopped. Optionally set by user when shutting down the virtual machine.

Table 7.354. Links summary

Name	Type	Summary
affinity_labels	AffinityLabel[]	Optional.
applications	Application[]	List of applications installed on the virtual machine.

Name	Type	Summary
cdroms	Cdrom[]	Reference to the ISO mounted to the CDROM.
cluster	Cluster	Reference to cluster the virtual machine belongs to.
cpu_profile	CpuProfile	Reference to CPU profile used by this virtual machine.
disk_attachments	DiskAttachment[]	References the disks attached to the virtual machine.
external_host_provider	ExternalHostProvider	
floppies	Floppy[]	Reference to the ISO mounted to the floppy.
graphics_consoles	GraphicsConsole[]	List of graphics consoles configured for this virtual machine.
host	Host	Reference to the host the virtual machine is running on.
host_devices	HostDevice[]	References devices associated to this virtual machine.
instance_type	InstanceType	The virtual machine configuration can be optionally predefined via one of the instance types.
katello_errata	KatelloErratum[]	Lists all the Katello errata assigned to the virtual machine.
nics	Nic[]	References the list of network interface devices on the virtual machine.
numa_nodes	NumaNode[]	Refers to the NUMA Nodes configuration used by this virtual machine.
original_template	Template	References the original template used to create the virtual machine.
permissions	Permission[]	Permissions set for this virtual machine.
quota	Quota	Reference to quota configuration set for this virtual machine.
reported_devices	ReportedDevice[]	
sessions	Session[]	List of user sessions opened for this virtual machine.
snapshots	Snapshot[]	Refers to all snapshots taken from the virtual machine.

Name	Type	Summary
statistics	Statistic[]	Statistics data collected from this virtual machine.
storage_domain	StorageDomain	Reference to storage domain the virtual machine belongs to.
tags	Tag[]	
template	Template	Reference to the template the virtual machine is based on.
vm_pool	VmPool	Reference to the pool the virtual machine is optionally member of.
watchdogs	Watchdog[]	Refers to the Watchdog configuration.

7.271.14. affinity_labels

Optional. Used for labeling of sub-clusters.

7.271.15. katello_errata

Lists all the Katello errata assigned to the virtual machine.

```
GET /ovirt-engine/api/vms/123/katelloerrata
```

You will receive response in XML like this one:

```
<katello_errata>
  <katello_erratum href="/ovirt-engine/api/katelloerrata/456" id="456">
    <name>RHBA-2013:XYZ</name>
    <description>The description of the erratum</description>
    <title>some bug fix update</title>
    <type>bugfix</type>
    <issued>2013-11-20T02:00:00.000+02:00</issued>
    <solution>Few guidelines regarding the solution</solution>
    <summary>Updated packages that fix one bug are now available for
XYZ</summary>
    <packages>
      <package>
        <name>libipa_hbac-1.9.2-82.11.el6_4.i686</name>
      </package>
      ...
    </packages>
  </katello_erratum>
  ...
</katello_errata>
```

7.271.16. original_template

References the original template used to create the virtual machine.

If the virtual machine is cloned from a template or another virtual machine, the **template** links to the Blank template, and the **original_template** is used to track history.

Otherwise the **template** and **original_template** are the same.

7.272. VMAFFINITY ENUM

Table 7.355. Values summary

Name	Summary
migratable	
pinned	
user_migratable	

7.273. VMBASE STRUCT

Represents basic virtual machine configuration. This is used by virtual machines, templates and instance types.

Table 7.356. Attributes summary

Name	Type	Summary
bios	Bios	Reference to virtual machine's BIOS configuration.
comment	String	Free text containing comments about this object.
console	Console	Console configured for this virtual machine.
cpu	Cpu	The configuration of the virtual machine CPU.
cpu_shares	Integer	
creation_time	Date	The virtual machine creation date.
custom_compatibility_version	Version	Virtual machine custom compatibility version.
custom_cpu_model	String	

Name	Type	Summary
custom_emulated_machine	String	
custom_properties	CustomProperty[]	Properties sent to VDSM to configure various hooks.
delete_protected	Boolean	If true , the virtual machine cannot be deleted.
description	String	A human-readable description in plain text.
display	Display	The virtual machine display configuration.
domain	Domain	Domain configured for this virtual machine.
high_availability	HighAvailability	The virtual machine high availability configuration.
id	String	A unique identifier.
initialization	Initialization	Reference to virtual machine's initialization configuration.
io	Io	For performance tuning of IO threading.
large_icon	Icon	Virtual machine's large icon.
lease	StorageDomainLease	Reference to the storage domain this virtual machine/template lease reside on.
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	Reference to virtual machine's memory management configuration.
migration	MigrationOptions	Reference to configuration of migration of running virtual machine to another host.
migration_downtime	Integer	Maximum time the virtual machine can be non responsive during its live migration to another host in ms.
name	String	A human-readable name in plain text.
origin	String	The origin of this virtual machine.

Name	Type	Summary
os	OperatingSystem	Operating system type installed on the virtual machine.
rng_device	RngDevice	Random Number Generator device configuration for this virtual machine.
serial_number	SerialNumber	Virtual machine's serial number in a cluster.
small_icon	Icon	Virtual machine's small icon.
soundcard_enabled	Boolean	If true , the sound card is added to the virtual machine.
sso	Sso	Reference to the Single Sign On configuration this virtual machine is configured for.
start_paused	Boolean	If true , the virtual machine will be initially in 'paused' state after start.
stateless	Boolean	If true , the virtual machine is stateless - it's state (disks) are rolled-back after shutdown.
time_zone	TimeZone	The virtual machine's time zone set by oVirt.
tunnel_migration	Boolean	If true , the network data transfer will be encrypted during virtual machine live migration.
type	VmType	Determines whether the virtual machine is optimized for desktop or server.
usb	Usb	Configuration of USB devices for this virtual machine (count, type).
virtio_scsi	VirtioScsi	Reference to VirtIO SCSI configuration.

7.273.1. cpu

The configuration of the virtual machine CPU.

The socket configuration can be updated without rebooting the virtual machine. The cores and the threads require a reboot.

For example, to change the number of sockets to 4 immediately, and the number of cores and threads to 2 after reboot, send the following request:

```
PUT /ovirt-engine/api/vms/123
```


With a request body:

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

7.273.2. custom_compatibility_version

Virtual machine custom compatibility version.

Enables a virtual machine to be customized to its own compatibility version. If **custom_compatibility_version** is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in, and is checked against capabilities of the host the virtual machine is planned to run on.

7.273.3. high_availability

The virtual machine high availability configuration. If set, the virtual machine will be automatically restarted when it unexpectedly goes down.

7.273.4. large_icon

Virtual machine's large icon. Either set by user or refers to image set according to operating system.

7.273.5. lease

Reference to the storage domain this virtual machine/template lease reside on.

A virtual machine running with a lease requires checking while running that the lease is not taken by another host, preventing another instance of this virtual machine from running on another host. This provides protection against split-brain in highly available virtual machines. A template can also have a storage domain defined for a lease in order to have the virtual machines created from this template to be preconfigured with this storage domain as the location of the leases.

7.273.6. memory

The virtual machine's memory, in bytes.

For example, to update a virtual machine to contain 1 Gibibyte (GiB) of memory, send the following request:

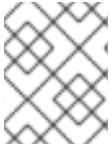
```
PUT /ovirt-engine/api/vms/123
```

With the following request body:

```
<vm>  
  <memory>1073741824</memory>  
</vm>
```

**NOTE**

Memory in the example is converted to bytes using the following formula:
1 GiB = 2^{30} bytes = 1073741824 bytes.

**NOTE**

Memory hot plug is supported from Red Hat Virtualization 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

7.273.7. migration_downtime

Maximum time the virtual machine can be non responsive during its live migration to another host in ms.

Set either explicitly for the virtual machine or by **engine-config -s DefaultMaximumMigrationDowntime=[value]**

7.273.8. origin

The origin of this virtual machine.

Possible values:

- **ovirt**
- **rhev**
- **vmware**
- **xen**
- **external**
- **hosted_engine**
- **managed_hosted_engine**
- **kvm**
- **physical_machine**
- **hyperv**

7.273.9. small_icon

Virtual machine's small icon. Either set by user or refers to image set according to operating system.

7.273.10. sso

Reference to the Single Sign On configuration this virtual machine is configured for. The user can be automatically signed in the virtual machine's operating system when console is opened.

Table 7.357. Links summary

Name	Type	Summary
c luster	Cluster	Reference to cluster the virtual machine belongs to.
cpu_profile	CpuProfile	Reference to CPU profile used by this virtual machine.
quota	Quota	Reference to quota configuration set for this virtual machine.
storage_domain	StorageDomain	Reference to storage domain the virtual machine belongs to.

7.274. VMDEVICETYPE ENUM

Table 7.358. Values summary

Name	Summary
cdrom	
floppy	

7.275. VMPLACEMENTPOLICY STRUCT

Table 7.359. Attributes summary

Name	Type	Summary
affinity	VmAffinity	

Table 7.360. Links summary

Name	Type	Summary
hosts	Host[]	

7.276. VMPOOL STRUCT

Table 7.361. Attributes summary

Name	Type	Summary
auto_storage_select	Boolean	Indicates if the pool should automatically distribute the disks of the virtual machines across the multiple storage domains where the template is copied.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
display	Display	
id	String	A unique identifier.
max_user_vms	Integer	
name	String	A human-readable name in plain text.
prestarted_vms	Integer	
rng_device	RngDevice	
size	Integer	
soundcard_enabled	Boolean	
stateful	Boolean	Virtual machine pool's stateful flag.
type	VmPoolType	
use_latest_template_version	Boolean	

7.276.1. auto_storage_select

Indicates if the pool should automatically distribute the disks of the virtual machines across the multiple storage domains where the template is copied.

When the template used by the pool is present in multiple storage domains, the disks of the virtual machines of the pool will be created in one of those storage domains. By default, or when the value of this attribute is **false**, that storage domain is selected when the pool is created, and all virtual machines will use the same. If this attribute is **true**, then, when a virtual machine is added to the pool, the storage domain that has more free space is selected.

7.276.2. stateful

Virtual machine pool's stateful flag.

Virtual machines from a stateful virtual machine pool are always started in stateful mode (stateless snapshot is not created). The state of the virtual machine is preserved even when the virtual machine is passed to a different user.

Table 7.362. Links summary

Name	Type	Summary
<code>cluster</code>	Cluster	
<code>instance_type</code>	InstanceType	Reference to the instance type on which this pool is based.
<code>permissions</code>	Permission[]	
<code>template</code>	Template	
<code>vm</code>	Vm	

7.276.3. instance_type

Reference to the instance type on which this pool is based. It can be set only on pool creation and cannot be edited.

7.277. VMPOOLTYPE ENUM

Table 7.363. Values summary

Name	Summary
<code>automatic</code>	
<code>manual</code>	

7.278. VMSTATUS ENUM

Type representing a status of a virtual machine.

Table 7.364. Values summary

Name	Summary
<code>down</code>	This status indicates that the virtual machine process is not running.

Name	Summary
image_locked	This status indicates that the virtual machine process is not running and there is some operation on the disks of the virtual machine that prevents it from being started.
migrating	This status indicates that the virtual machine process is running and the virtual machine is being migrated from one host to another.
not_responding	This status indicates that the hypervisor detected that the virtual machine is not responding.
paused	This status indicates that the virtual machine process is running and the virtual machine is paused.
powering_down	This status indicates that the virtual machine process is running and it is about to stop running.
powering_up	This status indicates that the virtual machine process is running and the guest operating system is being loaded.
reboot_in_progress	This status indicates that the virtual machine process is running and the guest operating system is being rebooted.
restoring_state	This status indicates that the virtual machine process is about to run and the virtual machine is going to awake from hibernation.
saving_state	This status indicates that the virtual machine process is running and the virtual machine is being hibernated.
suspended	This status indicates that the virtual machine process is not running and a running state of the virtual machine was saved.
unassigned	This status is set when an invalid status is received.
unknown	This status indicates that the system failed to determine the status of the virtual machine.
up	This status indicates that the virtual machine process is running and the guest operating system is loaded.
wait_for_launch	This status indicates that the virtual machine process is about to run.

7.278.1. paused

This status indicates that the virtual machine process is running and the virtual machine is paused. This may happen in two cases: when running a virtual machine is paused mode and when the virtual machine is being automatically paused due to an error.

7.278.2. powering_up

This status indicates that the virtual machine process is running and the guest operating system is being loaded. Note that if no guest-agent is installed, this status is set for a predefined period of time, that is by default 60 seconds, when running a virtual machine.

7.278.3. restoring_state

This status indicates that the virtual machine process is about to run and the virtual machine is going to awake from hibernation. In this status, the running state of the virtual machine is being restored.

7.278.4. saving_state

This status indicates that the virtual machine process is running and the virtual machine is being hibernated. In this status, the running state of the virtual machine is being saved. Note that this status does not mean that the guest operating system is being hibernated.

7.278.5. suspended

This status indicates that the virtual machine process is not running and a running state of the virtual machine was saved. This status is similar to Down, but when the VM is started in this status its saved running state is restored instead of being booted using the normal procedure.

7.278.6. unknown

This status indicates that the system failed to determine the status of the virtual machine. The virtual machine process may be running or not running in this status. For instance, when host becomes non-responsive the virtual machines that ran on it are set with this status.

7.278.7. up

This status indicates that the virtual machine process is running and the guest operating system is loaded. Note that if no guest-agent is installed, this status is set after a predefined period of time, that is by default 60 seconds, when running a virtual machine.

7.278.8. wait_for_launch

This status indicates that the virtual machine process is about to run. This status is set when a request to run a virtual machine arrives to the host. It is possible that the virtual machine process will fail to run.

7.279. VMSUMMARY STRUCT

Table 7.365. Attributes summary

Name	Type	Summary
active	Integer	
migrating	Integer	
total	Integer	

7.280. VMTYPE ENUM

Type representing what the virtual machine is optimized for.

Table 7.366. Values summary

Name	Summary
desktop	The virtual machine is intended to be used as a desktop.
server	The virtual machine is intended to be used as a server.

7.280.1. desktop

The virtual machine is intended to be used as a desktop. Currently, its implication is that a sound device will be automatically added to the virtual machine.

7.280.2. server

The virtual machine is intended to be used as a server. Currently, its implication is that a sound device will not be automatically added to the virtual machine.

7.281. VNICPASSTHROUGH STRUCT

Table 7.367. Attributes summary

Name	Type	Summary
mode	VnicPassThrough Mode	Defines whether the vNIC will be implemented as a virtual device, or as a pass-through to a host device.

7.282. VNICPASSTHROUGHMODE ENUM

Describes whether the vNIC is to be implemented as a pass-through device or a virtual one.

Table 7.368. Values summary

Name	Summary
disabled	To be implemented as a virtual device.
enabled	To be implemented as a pass-through device.

7.283. VNICPROFILE STRUCT

A vNIC profile is a collection of settings that can be applied to individual [NIC](#).

Table 7.369. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
custom_properties	CustomProperty[]	Custom properties applied to the vNIC profile.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
migratable	Boolean	Marks, whether pass_through NIC is migratable or not.
name	String	A human-readable name in plain text.
pass_through	VnicPassThrough	Enables the passthrough to a SR-IOV-enabled host NIC .
port_mirroring	Boolean	Enables port mirroring.

7.283.1. migratable

Marks, whether **pass_through** NIC is migratable or not.

If **pass_through.mode** is set to **disabled** this option has no meaning, and it will be considered to be **true**. If you omit this option from request, by default, this will be set to **true**.

When migrating virtual machine, this virtual machine will be migrated only if all **pass_through** NICs are flagged as **migratable**.

7.283.2. pass_through

Enables the passthrough to a SR-IOV-enabled [host NIC](#).

A vNIC profile enables a NIC to be directly connected to a [virtual function \(VF\)](#) of an SR-IOV-enabled host NIC, if passthrough is enabled. The NIC will then bypass the software network virtualization and connect directly to the VF for direct device assignment.

The passthrough cannot be enabled if the vNIC profile is already attached to a NIC. If a vNIC profile has passthrough enabled, **qos** and **port_mirroring** are disabled for the vNIC profile.

7.283.3. port_mirroring

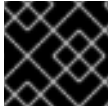
Enables port mirroring.

Port mirroring copies layer 3 network traffic on a given [logical network](#) and [host](#) to a NIC on a [virtual machine](#). This virtual machine can be used for network debugging and tuning, intrusion detection, and monitoring the behavior of other virtual machine on the same host and logical network. The only traffic copied is internal to one logical network on one host. There is no increase on traffic on the network external to the host; however a virtual machine with port mirroring enabled uses more host CPU and RAM than other virtual machines.

Port mirroring has the following limitations:

- Hot plugging NIC with a vNIC profile that has port mirroring enabled is not supported.
- Port mirroring cannot be altered when the vNIC profile is attached to a virtual machine.

Given the above limitations, it is recommended that you enable port mirroring on an additional, dedicated vNIC profile.



IMPORTANT

Enabling port mirroring reduces the privacy of other network users.

Table 7.370. Links summary

Name	Type	Summary
network	Network	Reference to the network that the vNIC profile is applied to.
network_filter	NetworkFilter	Reference to the top-level network filter that apply to the NICs that use this profile.
permissions	Permission[]	Permissions to allow usage of the vNIC profile.
qos	Qos	Reference to the quality of service attributes to apply to the vNIC profile.

7.283.4. network_filter

Reference to the top-level network filter that apply to the NICs that use this profile.

Network filters will enhance the admin ability to manage the network packets traffic from/to the participated virtual machines. The network filter may either contain a references to other filters, rules for traffic filtering, or hold a combination of both.

7.283.5. qos

Reference to the quality of service attributes to apply to the vNIC profile.

Quality of Service attributes regulate inbound and outbound network traffic of the NIC.

7.284. VNICPROFILEMAPPING STRUCT

Maps an external virtual NIC profile to one that exists in the Red Hat Virtualization Manager.

If, for example, the desired virtual NIC profile's mapping includes the following two lines:

Source network name	Source network profile name	Target virtual NIC profile ID
red	gold	738dd914-8ec8-4a8b-8628-34672a5d449b

Source network name	Source network profile name	Target virtual NIC profile ID
blue	silver	892a12ec-2028-4451-80aa-ff3bf55d6bac

It should be expressed in the following form:

```
<vnic_profile_mappings>
  <vnic_profile_mapping>
    <source_network_name>red</source_network_name>
    <source_network_profile_name>gold</source_network_profile_name>
    <target_vnic_profile id="738dd914-8ec8-4a8b-8628-34672a5d449b"/>
  </vnic_profile_mapping>
  <vnic_profile_mapping>
    <source_network_name>blue</source_network_name>
    <source_network_profile_name>silver</source_network_profile_name>
    <target_vnic_profile id="892a12ec-2028-4451-80aa-ff3bf55d6bac"/>
  </vnic_profile_mapping>
</vnic_profile_mappings>
```

Table 7.371. Attributes summary

Name	Type	Summary
source_network_name	String	Specifies the name of the external network.
source_network_profile_name	String	Specifies the name of the external network profile.

Table 7.372. Links summary

Name	Type	Summary
target_vnic_profile	VnicProfile	Points to an existing virtual NIC profile.

7.285. VOLUMEGROUP STRUCT

Table 7.373. Attributes summary

Name	Type	Summary
id	String	
logical_units	LogicalUnit[]	

Name	Type	Summary
name	String	

7.286. WATCHDOG STRUCT

This type represents a watchdog configuration.

Table 7.374. Attributes summary

Name	Type	Summary
action	WatchdogAction	Watchdog action to be performed when watchdog is triggered.
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
id	String	A unique identifier.
model	WatchdogModel	Model of watchdog device.
name	String	A human-readable name in plain text.

7.286.1. model

Model of watchdog device. Currently supported only I6300ESB.

Table 7.375. Links summary

Name	Type	Summary
instance_type	InstanceType	Optionally references to an instance type the device is used by.
template	Template	Optionally references to a template the device is used by.
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

7.286.2. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk may be used simultaneously by two or more virtual machines.

7.287. WATCHDOGACTION ENUM

This type describes available watchdog actions.

Table 7.376. Values summary

Name	Summary
dump	Virtual machine process will get core dumped to the default path on the host.
none	No action will be performed when watchdog action is triggered.
pause	Virtual machine will be paused when watchdog action is triggered.
poweroff	Virtual machine will be powered off when watchdog action is triggered.
reset	Virtual machine will be rebooted when watchdog action is triggered.

7.287.1. none

No action will be performed when watchdog action is triggered. However log message will still be generated.

7.288. WATCHDOGMODEL ENUM

This type represents the watchdog model.

Table 7.377. Values summary

Name	Summary
i6300esb	Currently only model supported is model I6300ESB.

7.289. WEIGHT STRUCT

Table 7.378. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human-readable description in plain text.
factor	Integer	
id	String	A unique identifier.
name	String	A human-readable name in plain text.

Table 7.379. Links summary

Name	Type	Summary
scheduling_policy	SchedulingPolicy	
scheduling_policy_unit	SchedulingPolicyUnit	

APPENDIX A. PRIMITIVE TYPES

This section describes the primitive data types supported by the API.

A.1. STRING PRIMITIVE

A finite sequence of [Unicode](#) characters.

A.2. BOOLEAN PRIMITIVE

Represents the *false* and *true* concepts used in mathematical logic.

The valid values are the strings **false** and **true**.

Case is ignored by the engine, so for example **False** and **FALSE** also valid values. However the server will always return lower case values.

For backwards compatibility with older versions of the engine, the values **0** and **1** are also accepted. The value **0** has the same meaning than **false**, and **1** has the same meaning than **true**. Try to avoid using these values, as support for them may be removed in the future.

A.3. INTEGER PRIMITIVE

Represents the mathematical concept of integer number.

The valid values are finite sequences of decimal digits.

Currently the engine implements this type using a signed 32 bit integer, so the minimum value is -2^{31} (-2147483648) and the maximum value is $2^{31}-1$ (2147483647).

However, there are some attributes in the system where the range of values possible with 32 bit isn't enough. In those exceptional cases the engine uses 64 bit integers, in particular for the following attributes:

- **Disk.actual_size**
- **Disk.provisioned_size**
- **GlusterClient.bytes_read**
- **GlusterClient.bytes_written**
- **Host.max_scheduling_memory**
- **Host.memory**
- **HostNic.speed**
- **LogicalUnit.size**
- **MemoryPolicy.guaranteed**
- **NumaNode.memory**

- `QuotaStorageLimit.limit`
- `StorageDomain.available`
- `StorageDomain.used`
- `StorageDomain.committed`
- `VmBase.memory`

For these exception cases the minimum value is -2^{63} (-9223372036854775808) and the maximum value is $2^{63}-1$ (9223372036854775807).



NOTE

In the future the integer type will be implemented using unlimited precision integers, so the above limitations and exceptions will eventually disappear.

A.4. DECIMAL PRIMITIVE

Represents the mathematical concept of real number.

Currently the engine implements this type using 32 bit [IEEE 754](#) single precision floating point numbers.

For some attributes this isn't enough precision. In those exceptional cases the engine uses 64 bit double precision floating point numbers, in particular for the following attributes:

- `QuotaStorageLimit.usage`
- `QuotaStorageLimit.memory_limit`
- `QuotaStorageLimit.memory_usage`



NOTE

In the future the decimal type will be implemented using unlimited precision decimal numbers, so the above limitations and exceptions will eventually disappear.

A.5. DATE PRIMITIVE

Represents a date and time.

The format returned by the engine is the one described in the [XML Schema specification](#) when requesting XML. For example, if you send a request like this to retrieve the XML representation of a virtual machine:

```
GET /ovirt-engine/api/vms/123
Accept: application/xml
```

The response body will contain the following XML document:

```
<vm id="123" href="/ovirt-engine/api/vms/123">
  ...
  <creation_time>2016-09-08T09:53:35.138+02:00</creation_time>
```



```
...  
</vm>
```

When requesting the JSON representation the engine uses a different, format: an integer containing the number of seconds since Jan 1st 1970, also know as *epoch time*. For example, if you send a request like this to retrieve the JSON representation of a virtual machine:

```
GET /ovirt-engine/api/vms/123  
Accept: application/json
```

The response body will contain the following JSON document:

```
{  
  "id": "123",  
  "href="/ovirt-engine/api/vms/123",  
  ...  
  "creation_time": 1472564909990,  
  ...  
}
```



NOTE

In both cases, the dates returned by the engine use the time zone configured in the server where it is running, in the above examples it is UTC+2.

APPENDIX B. CHANGES IN VERSION 4 OF THE API

This section enumerates the backwards compatibility breaking changes that have been introduced in version 4 of the API.

B.1. REMOVED YAML SUPPORT

The support for YAML has been completely removed.

B.2. RENAMED COMPLEX TYPES

The following XML schema complex types have been renamed:

Version 3	Version 4
API	Api
CPU	Cpu
CPUs	Cpus
CdRom	Cdrom
CdRoms	Cdroms
DNS	Dns
GuestNicConfiguration	NicConfiguration
GuestNicsConfiguration	NicConfigurations
HostNICStates	HostNicStates
HostNIC	HostNic
HostStorage	HostStorages
IO	Io
IP	Ip
IPs	Ips
KSM	Ksm
MAC	Mac

Version 3	Version 4
NIC	Nic
PreviewVMs	PreviewVms
QoS	Qos
QoSs	Qoss
RSDL	Rsd1
SELinux	SeLinux
SPM	Spm
SSHPublicKey	SshPublicKey
SSHPublicKeys	SshPublicKeys
SSH	Ssh
SkipIfSDActive	SkipIfSdActive
Slaves	HostNics
Storage	HostStorage
SupportedVersions	Versions
VCpuPin	VcpuPin
VLAN	Vlan
VM	Vm
VMs	Vms
VirtIO_SCSI	VirtioScsi
WatchDog	Watchdog
WatchDogs	Watchdogs

B.3. REPLACED THE `STATUS` TYPE WITH ENUM TYPES

Currently the status of different objects is reported using the **Status** type, which contains a **state** string describing the status and another **detail** string for additional details. For example, the status of a virtual machine that is paused due to an IO error is currently reported as follows:

```
<vm>
  ...
  <status>
    <state>paused</state>
    <detail>eio</detail>
  </status>
  ...
</vm>
```

In version 4 of the API this **Status** type has been removed and replaced by enum types. When the additional **detail** string is needed it has been replaced with an additional **status_detail** attribute. So, for example, the status of the same virtual machine will now be reported as follows:

```
<vm>
  ...
  <status>paused</status>
  <status_detail>eio</status_detail>
  ...
</vm>
```

B.4. REMOVE THE NIC NETWORK AND PORT_MIRRORING PROPERTIES

The NIC **network** and **port_mirroring** elements have been replaced by the **vnic_profile** element, so when creating or updating a NIC instead of specifying the network and port mirroring configuration, these are previously specified creating a vNIC profile:

```
POST /ovirt-engine/api/vnicprofiles
```

```
<vnic_profile>
  <name>myprofile</name>
  <network id="...">
  <port_mirroring>true</port_mirroring>
</vnic_profile>
```

And then the NIC is created or referencing the existing vNIC profile:

```
PUT /ovirt-engine/api/vms/123/nics/456
```

```
<nic>
  <vnic_profile id="/vnicprofiles/...">
</nic>
```

The old elements and their meaning were preserved for backwards compatibility, but they have now been completely removed.

Note that the **network** element hasn't been removed from the XML schema because it is still used by the **initialization** element, but it will be completely ignored if provided when creating or updating a NIC.

B.5. REMOVE THE NIC `ACTIVE` PROPERTY

The NIC `active` property was replaced by `plugged` some time ago. It has been completely removed now.

B.6. REMOVE THE DISK `TYPE` PROPERTY

The `type` property of disks has been removed, but kept in the XML schema and ignored. It has been completely removed now.

B.7. REMOVE THE DISK `SIZE` PROPERTY

The disk `size` property has been replaced by `provisioned_size` long ago. It has been completely removed now.

B.8. REMOVED SUPPORT FOR PINNING A VM TO A SINGLE HOST

Before version 3.6 the API had the possibility to pin a VM to a single host, using the `placement_policy` element of the VM entity:

```
PUT /ovirt-engine/api/vms/123
```

```
<vm>
  <placement_policy>
    <host id="456"/>
  </placement_policy>
</vm>
```

In version 3.6 this capability was enhanced to support multiple hosts, and to do so a new `hosts` element was added:

```
PUT /ovirt-engine/api/vms/123
```

```
<vm>
  <placement_policy>
    <hosts>
      <host id="456"/>
      <host id="789"/>
      ...
    </hosts>
  </placement_policy>
</vm>
```

To preserve backwards compatibility the single `host` element was preserved. In 4.0 this has been removed, so applications will need to use the `hosts` element even if when pinning to a single host.

B.9. REMOVED THE `CAPABILITIES.PERMITS` ELEMENT

The list of permits is potentially different for each cluster level, and it has been added to the **version** element long ago, but it has been kept into the **capabilities** element as well, just for backwards compatibility.

In 4.0 the **capabilities** service has been completely removed, and replaced by the new **clusterlevels** service. To find the permits supported by cluster level 4.0 a request like this should be used:

```
GET /ovirt-engine/api/clusterlevels/4.0
```

The result will be a document containing the information specific to that cluster level, in particular the set of supported permits:

```
<cluster_level id="4.0" href="/clusterlevels/4.0">
  ...
  <permits>
    <permit id="1">
      <name>create_vm</name>
      <administrative>>false</administrative>
    </permit>
    ...
  </permits>
</cluster_level>
```

B.10. REMOVED THE `STORAGE_MANAGER` ELEMENT

The **storage_manager** element was replaced by the **spm** element some time ago. The old one was kept for backwards compatibility, but it has been completely removed now.

B.11. REMOVED THE `DATA_CENTER STORAGE_TYPE` ELEMENT

Data centers used to be associated to a specific storage type (NFS, Fiber Channel, iSCSI, etc) but they have been changed some time so that there are only two types: with local storage and with shared storage. A new **local** element was introduced to indicate this, and the old **storage_type** element was preserved for backwards compatibility. This old element has now been completely removed.

B.12. REMOVE THE `TIMEZONE` ELEMENT

The VM resource used to contain a **timezone** element to represent the time zone. This element only allowed a string:

```
<vm>
  <timezone>Europe/Madrid</timezone>
</vm>
```

This doesn't allow extension, and as it was necessary to add the UTC offset, it was replaced with a new structured **time_zone** element:

```
<vm>
  <time_zone>
    <name>Europe/Madrid</name>
```

```

    <utc_offset>GMT+1</utc_offset>
  </time_zone>
</vm>

```

The old **timezone** element was preserved, but it has been completely removed now.

B.13. REMOVED THE `GUEST_INFO` ELEMENT

The **guest_info** element was used to hold information gathered by the guest agent, like the IP addresses and the fully qualified host name. This information is also available in other places. For example, the IP addresses are available within VM resource:

```
GET /ovirt-engine/api/vms/123
```

```

<vm>
  <guest_info>
    <ips>
      <ip address="192.168.122.30"/>
    </ips>
    <fqdn>myvm.example.com</fqdn>
  </guest_info>
</vm>

```

And also within the NIC resource, using the newer **reported_devices** element:

```
GET /ovirt-engine/api/vms/{vm:id}/nics/{nic:id}
```

```

<nic>
  <reported_devices>
    <reported_device>
      <name>eth0</name>
      <mac address="00:1a:4a:b5:4c:94"/>
      <ips>
        <ip address="192.168.1.115" version="v4"/>
        <ip address="fe80::21a:4aff:feb5:4c94" version="v6"/>
        <ip address "::1:21a:4aff:feb5:4c94" version="v6"/>
      </ips>
    </reported_device>
  </reported_devices>
</nic>

```

In addition this newer **reported_devices** element provides more complete information, like multiple IP addresses, MAC addresses, etc.

To remove this duplication the **guest_info** element has been removed.

To support the fully qualified domain name a new **fqdn** element has been added to the VM resource:

```
GET /ovirt-engine/api/vms/123
```

```

<vm>
  <fqdn>myvm.example.com</fqdn>
</vms>

```

This will contain the same information that `guest_info.fqdn` used to contain.

B.14. REPLACED CPU ID ATTRIBUTE WITH TYPE ELEMENT

The `cpu` element used to have an `id` attribute that indicates the type of CPU:

```
<cpu id="Intel Conroe Family">
  <architecture>X86_64</architecture>
  ...
</cpu>
```

This is in contradiction with the rest of the elements of the API model, where the `id` attribute is used for opaque identifiers. This `id` attribute has been replaced with a new `type` element:

```
<cpu>
  <type>Intel Conroe Family</type>
  <architecture>X86_64</architecture>
</cpu>
```

B.15. USE ELEMENTS INSTEAD OF ATTRIBUTES IN CPU TOPOLOGY

In the past the CPU topology element used attributes for its properties:

```
<cpu>
  <topology sockets="1" cores="1" threads="1"/>
  ...
</cpu>
```

This is contrary to the common practice in the API. They have been replaced by inner elements:

```
<cpu>
  <topology>
    <sockets>1</sockets>
    <cores>1</cores>
    <threads>1</threads>
  </topology>
  ...
</cpu>
```

B.16. USE ELEMENTS INSTEAD OF ATTRIBUTES IN VCPU PIN

In the past the VCPU pin element used attributes for its properties:

```
<cpu_tune>
  <vcpu_pin vcpu="0" cpu_set="0"/>
</cpu_tune>
```

This is contrary to the common practice in the API. They have been replaced by inner elements:

```
<cpu_tune>
```



```

<vcpu_pin>
  <vcpu>0</vcpu>
  <cpu_set>0</cpu_set>
</vcpu_pin>
</cpu_tune>

```

B.17. USE ELEMENTS INSTEAD OF ATTRIBUTES IN VCPU PIN

In the past the **version** element used attributes for its properties:

```
<version major="3" minor="5" ../>
```

This is contrary to the common practice in the API. They have been replaced by inner elements:

```

<version>
  <major>3</major>
  <minor>5</minor>
  ...
</version>

```

B.18. USE ELEMENTS INSTEAD OF ATTRIBUTES IN MEMORY OVERCOMMIT

In the past the **overcommit** element used attributes for its properties:

```

<memory_policy>
  <overcommit percent="100"/>
  ...
</memory_policy>

```

This is contrary to the common practice in the API. They have been replaced by inner elements:

```

<memory_policy>
  <overcommit>
    <percent>100</percent>
  </overcommit>
  ...
</memory_policy>

```

B.19. USE ELEMENTS INSTEAD OF ATTRIBUTES IN `console`

In the past the **console** element used attributes for its properties:

```
<console enabled="true"/>
```

This is contrary to the common practice in the API. They have been replaced by inner elements:

```

<console>
  <enabled>>true</enabled>
</console>

```

B.20. USE ELEMENTS INSTEAD OF ATTRIBUTES IN VIRTIO SCSI

In the past the VIRTIO ISCSI element used attributes for its properties:

```
<virtio_scsi enabled="true"/>
```

This is contrary to the common practice in the API. They have been replaced by inner elements:

```
<virtio_scsi>
  <enabled>true</enabled>
</virtio_scsi>
```

B.21. USE ELEMENT INSTEAD OF ATTRIBUTE FOR POWER MANAGEMENT AGENT TYPE

The power management **type** property was represented as an attribute:

```
<agent type="apc">
  <username>myuser</username>
  ...
</agent>
```

This is contrary to the common practice in the API. It has been replaced with an inner element:

```
<agent>
  <type>apc</type>
  <username>myuser</username>
  ...
</agent>
```

B.22. USE ELEMENTS INSTEAD OF ATTRIBUTES IN POWER MANAGEMENT AGENT OPTIONS

In the past the power management agent options element used attributes for its properties:

```
<options>
  <option name="port" value="22"/>
  <option name="slot" value="5"/>
  ...
</options>
```

This is contrary to the common practice in the API. They have been replaced with inner elements:

```
<options>
  <option>
    <name>port</name>
    <value>22</value>
  </option>
  <option>
    <name>slot</name>
    <value>5</value>
```

```

</option>
...
</options>

```

B.23. USE ELEMENTS INSTEAD OF ATTRIBUTES IN IP ADDRESS:

In the past the IP address element used attributes for its properties:

```
<ip address="192.168.122.1" netmask="255.255.255.0"/>
```

This is contrary to the common practice in the API. They have been replaced with inner elements:

```

<ip>
  <address>192.168.122.1</address>
  <netmask>255.255.255.0</netmask>
</ip>

```

B.24. USE ELEMENTS INSTEAD OF ATTRIBUTES IN MAC ADDRESS:

In the past the MAC address element used attributes for its properties:

```
<mac address="66:f2:c5:5f:bb:8d"/>
```

This is contrary to the common practice in the API. They have been replaced by inner elements:

```

<mac>
  <address>66:f2:c5:5f:bb:8d</address>
</mac>

```

B.25. USE ELEMENTS INSTEAD OF ATTRIBUTES IN BOOT DEVICE:

In the past the boot device element used attributes for its properties:

```
<boot dev="cdrom"/>
```

This is contrary to the common practice in the API. They have been replaced by inner elements:

```

<boot>
  <dev>cdrom</dev>
</boot>

```

B.26. USE ELEMENT INSTEAD OF ATTRIBUTE FOR OPERATING SYSTEM TYPE

The operating system **type** property was represented as an attribute:

```

<os type="other">
  ...
</os>

```

This is contrary to the common practice in the API. It has been replaced with an inner element:

```
<os>
  <type>other</type>
  ...
</os>
```

B.27. REMOVED THE `FORCE` PARAMETER FROM THE REQUEST TO RETRIEVE A HOST

The request to retrieve a host used to support a **force** matrix parameter to indicate that the data of the host should be refreshed (calling VDSM to reload host capabilities and devices) before retrieving it from the database:

```
GET /ovirt-engine/api/hosts/123;force
```

This **force** parameter has been superseded by the host **refresh** action, but kept for backwards compatibility. It has been completely removed now. Applications that require this functionality should perform two requests, first one to refresh the host:

```
POST /ovirt-engine/api/hosts/123/refresh
```

```
<action/>
```

And then one to retrieve it, without the **force** parameter:

```
GET /ovirt-engine/api/hosts/123
```

B.28. REMOVED DEPRECATED HOST POWER MANAGEMENT CONFIGURATION

The host power management configuration used to be part of the host resource, using embedded configuration elements:

```
<power_management type="apc">
  <enabled>true</enabled>
  <address>myaddress</address>
  <username>myaddress</username>
  <options>
    <option name="port" value="22"/>
    </option name="slot" value="5"/>
  </options>
  ...
</power_management>
```

This has been changed some time ago, in order to support multiple power management agents, introducing a new `/hosts/123/fenceagents` collection.

The old **type** attribute, the old **address**, **username** and **password** elements, and the inner **agents** element directly inside **power_management** were preserved for backwards compatibility. All these elements have been completely removed, so the only way to query or modify the power management

agents is now the `/hosts/123/fenceagents` sub-collection.

B.29. USE MULTIPLE `BOOT.DEVICES.DEVICE` INSTEAD OF MULTIPLE `BOOT`

In the past the way to specify the boot sequence when starting a virtual machine was to use multiple `boot` elements, each containing a `dev` element. For example, to specify that the virtual machine should first try to boot from CDROM and then from hard disk the following request was used:

```
POST /ovirt-engine/api/vms/123/start
```

```
<action>
  <vm>
    ...
    <boot>
      <dev>cdrom</dev>
    </boot>
    <boot>
      <dev>hd</dev>
    </boot>
  </vm>
</action>
```

The common practice in other parts of the API is to represent arrays with a wrapper element. In that case that wrapper element could be named `boots`, but that doesn't make much sense, as what can have multiple values here is the boot device, not the boot sequence. To fix this inconsistency this has been replaced with a single `boot` element that can contain multiple devices:

```
POST /ovirt-engine/api/vms/123/start
```

```
<action>
  <vm>
    ...
    <boot>
      <devices>
        <device>cdrom</device>
        <device>hd</device>
      </devices>
    </boot>
  </vm>
</action>
```

B.30. REMOVED THE `DISKS.CLONE` AND `DISKS.DETACH_ONLY` ELEMENTS

These elements aren't really part of the representation of disks, but parameters of the operations to add and remove virtual machines.

The `disks.clone` element was used to indicate that the disks of a new virtual machine have to be cloned:

```
POST /ovirt-engine/api/vms
```

```
<vm>
```

```

...
<disks>
  <clone>true</clone>
</disks>
</vm>

```

This has been now removed, and replaced by a new **clone** query parameter:

```
POST /ovirt-engine/api/vms?clone=true
```

```

<vm>
...
</vm>

```

The **disks.detach_only** element was used to indicate that when removing a virtual machine the disks don't have to be removed, but just detached from the virtual machine:

```
DELETE /ovirt-engine/api/vms/123
```

```

<action>
  <vm>
    <disks>
      <detach_only>true</detach_only>
    </disks>
  </vm>
</action>

```

This has been now removed, and replaced by a new **detach_only** query parameter:

```
DELETE /ovirt-engine/api/vms/123?detach_only=true
```

B.31. RENAME ELEMENT **VMPool** TO **VM_POOL**

The names of the elements that represent pools of virtual machines used to be **vmPool** and **vmPools**. They have been renamed to **vm_pool** and **vm_pools** in order to have a consistent correspondence between names of complex types (**VmPool** and **VmPools** in this case) and elements.

B.32. USE **LOGICAL_UNITS** INSTEAD OF MULTIPLE **LOGICAL_UNIT**

The logical units that are part of a volume group used to be reported as an unbounded number of **logical_unit** elements. For example, when reporting the details of a storage domain:

```
GET /ovirt-engine/api/storagedomains/123
```

```

<storage_domain>
...
  <storage>
    ...
    <volume_group>
      <logical_unit>
        <!-- First LU -->

```

```

    </logical_unit>
    <logical_unit>
      <!-- Second LU -->
    </logical_unit>
    ...
  </volume_group>
</storage>
</storage_domain>

```

This is contrary to the usual practice in the API, as list of elements are always wrapped with an element. This has been fixed now, so the list of logical units will be wrapped with the **logical_units** element:

```
GET /ovirt-engine/api/storagedomains/123
```

```

<storage_domain>
  ...
  <storage>
    ...
    <volume_group>
      <logical_units>
        <logical_unit>
          <!-- First LU -->
        </logical_unit>
        <logical_unit>
          <!-- Second LU -->
        </logical_unit>
        ...
      </logical_units>
    </volume_group>
  </storage>
</storage_domain>

```

B.33. REMOVED THE `SNAPSHOTS.COLLAPSE_SNAPSHOTS` ELEMENT

This element isn't really part of the representation of snapshots, but a parameter of the operation that imports a virtual machine from an export storage domain:

```
POST /ovirt-engine/api/storagedomains/123/vms/456/import
```

```

<action>
  <vm>
    <snapshots>
      <collapse_snapshots>true</collapse_snapshots>
    </snapshots>
  </vm>
</action>

```

This has been now removed, and replaced by a new **collapse_snapshots** query parameter:

```
POST /ovirt-engine/api/storagedomains/123/vms/456/import?
collapse_snapshots=true
```

```
<action/>
```

B.34. RENAMED STORAGE AND HOST_STORAGE ELEMENTS

The host storage collection used the **storage** and **host_storage** elements and the **Storage** and **HostStorage** complex types to report the storage associated to a host:

```
GET /ovirt-engine/api/hosts/123/storage
```

```
<host_storage>
  <storage>
    ...
  </storage>
  <storage>
    ...
  </storage>
  ...
</host_storage>
```

This doesn't follow the pattern used in the rest of the API, where the outer element is a plural name and the inner element is the same name but in singular. This has now been changed to use **host_storages** as the outer element and **host_storage** as the inner element:

```
GET /ovirt-engine/api/hosts/123/storage
```

```
<host_storages>
  <host_storage>
    ...
  </host_storage>
  <host_storage>
    ...
  </host_storage>
  ...
</host_storages>
```

B.35. REMOVED THE PERMISSIONS.CLONE ELEMENT

This element isn't really part of the representation of permissions, but a parameter of the operations to create virtual machines or templates:

```
POST /ovirt-engine/api/vms
```

```
<vm>
  <template id="...">
    <permissions>
      <clone>true</clone>
    </permissions>
  </template>
</action>
```



```
POST /ovirt-engine/api/templates
```

```
<template>
  <vm id="...">
    <permissions>
      <clone>true</clone>
    </permissions>
  </vm>
</template>
```

This has been now removed, and replaced by a new **clone_permissions** query parameter:

```
POST /ovirt-engine/api/vms?clone_permissions=true
```

```
<vm>
  <template id="..." />
</vm>
```

```
POST /ovirt-engine/api/templates?clone_permissions=true
```

```
<template>
  <vm id="..." />
</template>
```

B.36. RENAMED THE RANDOM NUMBER GENERATOR SOURCE ELEMENTS

The random number generator sources used to be reported using a collection of **source** elements wrapped by an element with a name reflecting its use. For example, the required random number generator sources of a cluster used to be reported as follows:

```
GET /ovirt-engine/api/clusters/123
```

```
<cluster>
  ...
  <required_rng_sources>
    <source>random</source>
  </required_rng_sources>
  ...
</cluster>
```

And the random number generator sources supported by a host used to be reported as follows:

```
GET /ovirt-engine/api/hosts/123
```

```
<host>
  ...
  <hardware_information>
    <supported_rng_sources>
      <source>random</source>
```

```

    </supported_rng_sources>
  </hardware_information>
  ...
</host>

```

This isn't consistent with the rest of the API, where collections are wrapped by a name in plural and elements by the same name in singular. This has been now fixed. The required random number generator sources will now be reported as follows:

```
GET /ovirt-engine/api/clusters/123
```

```

<cluster>
  <required_rng_sources>
    <required_rng_source>random</required_rng_source>
  </required_rng_sources>
  ...
</cluster>

```

And the random number generator sources supported by a host will be reported as follows:

```
GET /ovirt-engine/api/hosts/123
```

```

<host>
  ...
  <hardware_information>
    <supported_rng_sources>
      <supported_rng_source>random</supported_rng_source>
    </supported_rng_sources>
  </hardware_information>
  ...
</host>

```

Note the use of **required_rng_source** and **supported_rng_source** instead of just **source**.

B.37. REMOVED THE INTERMEDIATE TAG.PARENT ELEMENT

The relationship between a tag and its parent tag used to be represented using an intermediate **parent** tag, that in turn contains another **tag** element:

```

<tag>
  <name>mytag</name>
  <parent>
    <tag id="..." href="..." />
  </parent>
</tag>

```

This structure has been simplified so that only one **parent** element is used now:

```

<tag>
  <name>mytag</name>
  <parent id="..." href="..." />
</tag>

```

B.38. REMOVE SCHEDULING BUILT-IN NAMES AND THRESHOLDS

In the past the specification of scheduling policies for clusters was based in built-in names and thresholds. For example a cluster that used the **evenly distributed** scheduling policy was represented as follows:

```
<cluster>
  <name>mycluster</name>
  <scheduling_policy>
    <policy>evenly_distributed</policy>
    <thresholds high="80" duration="120"/>
  </scheduling_policy>
  ...
</cluster>
```

This mechanism was replaced with a top level **/schedulingpolicies** collection where scheduling policies can be defined with arbitrary names and properties. For example, the same scheduling policy is represented as follows in that top level collection:

```
<scheduling_policy>
  <name>evenly_distributed</name>
  <properties>
    <property>
      <name>CpuOverCommitDurationMinutes</name>
      <value>2</value>
    </property>
    <property>
      <name>HighUtilization</name>
      <value>80</value>
    </property>
  </properties>
</scheduling_policy>
```

The representation of the cluster references the scheduling policy with its identifier:

```
<cluster>
  <name>mycluster</name>
  <scheduling_policy id="..." />
  ...
</cluster>
```

To preserve backwards compatibility the old **policy** and **thresholds** elements were preserved. The scheduling policy representation embedded within the cluster was also preserved. All these things have been completely removed now, so the only way to reference a scheduling policy when retrieving, creating or updating a cluster is to reference an existing one using its identifier. For example, when retrieving a cluster only the **id** (and **href**) will be populated:

```
GET /ovirt-engine/api/clusters/123
```

```
<cluster>
  ...
  <scheduling_policy id="..." href="..." />
  ...
</cluster>
```

When creating or updating a cluster only the **id** will be accepted.

B.39. REMOVED THE `BRICKS.REPLICA_COUNT` AND `BRICKS.STRIPE_COUNT` ELEMENTS

These elements aren't really part of the representation of a collection of bricks, but parameters of the operations to add and remove bricks. They have now been removed, and replaced by new **replica_count** and **stripe_count** parameters:

```
POST .../bricks?replica_count=3&stripe_count=2
```

```
DELETE .../bricks?replica_count=3
```

B.40. RENAMED THE `STATISTICS` TYPE PROPERTY TO `KIND`

The statistics used to be represented using a **type** element that indicates the kind of statistic (gauge, counter, etc) and also a **type** attribute that indicates the type of the values (integer, string, etc):

```
<statistic>
  <type>GAUGE</type>
  <values type="INTEGER">
    <value>...</value>
    <value>...</value>
    ...
  </values>
</statistic>
```

To avoid the use of the **type** concept for both things the first has been replaced by **kind**, and both **kind** and **type** are now elements:

```
<statistic>
  <kind>gauge</kind>
  <type>integer</type>
  <values>
    <value>...</value>
    <value>...</value>
    ...
  </values>
</statistic>
```

B.41. USE MULTIPLE `VCPU_PINS.VCPU_PIN` INSTEAD OF MULTIPLE `VCPU_PIN`

In the past the way to specify the virtual to physical CPU pinning of a virtual machine was to use multiple **vcpu_pin** elements:

```
<vm>
  <cpu>
    <cpu_tune>
      <vcpu_pin>...</vcpu_pin>
      <vcpu_pin>...</vcpu_pin>
```

```

    ...
  </cpu_tune>
</cpu>
</vm>

```

In order to conform to the common practice in other parts of the API this has been changed to use a wrapper element, in this case **vcpu_pins**:

```

<vm>
  <cpu>
    <cpu_tune>
      <vcpu_pins>
        <vcpu_pin>...</vcpu_pin>
        <vcpu_pin>...</vcpu_pin>
        ...
      </vcpu_pins>
    </cpu_tune>
  </cpu>
</vm>

```

B.42. USE **FORCE** PARAMETER TO FORCE REMOVE A DATA CENTER

The operation that removes a data center supports a **force** parameter. In order to use it the **DELETE** operation used to support an optional action parameter:

```
DELETE /ovirt-engine/api/datacenters/123
```

```

<action>
  <force>true</force>
</action>

```

This optional action parameter has been replaced with an optional parameter:

```
DELETE /ovirt-engine/api/datacenters/123?force=true
```

B.43. USE **FORCE** PARAMETER TO FORCE REMOVE A HOST

The operation that removes a host supports a **force** parameter. In order to use it the **DELETE** operation used to support an optional action parameter:

```
DELETE /ovirt-engine/api/host/123
```

```

<action>
  <force>true</force>
</action>

```

This optional action parameter has been replaced with an optional parameter:

```
DELETE /ovirt-engine/api/host/123?force=true
```

B.44. USE PARAMETERS FOR FORCE REMOVE STORAGE DOMAIN

The operation that removes a storage domain supports the **force**, **destroy** and **host** parameters. These parameters were passed to the **DELETE** method using the representation of the storage domain as the body:

```
DELETE /ovirt-engine/api/storagedomains/123
```

```
<storage_domain>
  <force>...</force>
  <destroy>...</destroy>
  <host id="...">
    <name>...</name>
  </host>
</storage_domain>
```

This was problematic, as the HTTP **DELETE** parameters shouldn't have a body, and the representation of the storage domain shouldn't include things that aren't attributes of the storage domain, rather parameters of the operation.

The **force**, **delete** and **host** attributes have been replaced by equivalent parameters, and the operation doesn't now accept a body. For example, now the correct way to delete a storage domain with the **force** parameter is the following:

```
DELETE /ovirt-engine/api/storagedomain/123?host=myhost&force=true
```

To delete with the **destroy** parameter:

```
DELETE /ovirt-engine/api/storagedomain/123?host=myhost&destroy=true
```

B.45. USE `HOST` PARAMETER TO REMOVE STORAGE SERVER CONNECTION

The operation that removes a storage server connection supports a **host** parameter. In order to use it the **DELETE** method used to support an optional action parameter:

```
DELETE /ovirt-engine/api/storageconnections/123
```

```
<action>
  <host id="...">
    <name>...</name>
  </host>
</action>
```

This optional action parameter has been replaced with an optional parameter:

```
DELETE /ovirt-engine/api/storageconnections/123?host=myhost
```

B.46. USE `FORCE` AND `STORAGE_DOMAIN` PARAMETERS TO REMOVE TEMPLATE DISKS

The operation that removes a template disk supports the `force` and `storage_domain` parameters. In order to use it them the `DELETE` method used to support an optional action parameter:

```
DELETE /ovirt-engine/api/templates/123/disks/456
```

```
<action>
  <force>...</force>
  <storage_domain id="...">/>
</action>
```

In version 4 of the API this operation has been moved to the new `diskattachments` collection, and the request body has been replaced with the query parameters `force` and `storage_domain`:

```
DELETE /ovirt-engine/api/templates/123/disksattachments/456?force=true
```

```
DELETE /ovirt-engine/api/templates/123/disksattachments/456?
storage_domain=123
```

B.47. DON'T REMOVE DISKS VIA THE VM DISK API

Removing an entity by deleting `/vms/123/disks/456` means removing the relationship between the VM and the disk - i.e., this operation should just detach the disk from the VM. This operation is no longer able to remove disks completely from the system, which was prone to user errors and had unreversible consequences. To remove a disk, instead use the `/disk/456` API:

```
DELETE /ovirt-engine/api/disks/456
```

B.48. USE `FORCE` QUERY PARAMETER TO FORCE REMOVE A VIRTUAL MACHINE

The operation that removes a virtual machine supports a `force` parameter. In order to use it the `DELETE` method used to support an optional action parameter:

```
DELETE /ovirt-engine/api/vms/123
```

```
<action>
  <force>true</force>
</action>
```

This optional action parameter has been replaced with an optional query parameter:

```
DELETE /ovirt-engine/api/vms/123?force=true
```

B.49. USE `POST` INSTEAD OF `DELETE` TO REMOVE MULTIPLE BRICKS

The operation that removes multiple Gluster bricks was implemented using the **DELETE** method and passing the list of bricks as the body of the request:

```
DELETE /ovirt-engine/api/clusters/123/glustervolumes/456/bricks
```

```
<bricks>
  <bricks id="..."/>
  <bricks id="..."/>
  ...
</bricks>
```

This is problematic because the **DELETE** method shouldn't have a body, so it has been replaced with a new **remove** action that uses the **POST** method:

```
POST /ovirt-engine/api/clusters/123/glustervolumes/456/bricks/remove
```

```
<bricks>
  <bricks id="..."/>
  <bricks id="..."/>
  ...
</bricks>
```

B.50. REMOVED THE `SCHEDULING_POLICY.POLICY_ELEMENT`

The element was kept for backward compatibility. Use `scheduling_policy.name` instead.

```
POST /ovirt-engine/api/schedulingpolicies
```

```
<scheduling_policy>
  ...
  <name>policy_name</name>
  ...
</scheduling_policy>
```

```
PUT /ovirt-engine/api/schedulingpolicies/123
```

```
<scheduling_policy>
  ...
  <name>policy_name</name>
  ...
</scheduling_policy>
```

B.51. ADDED `SNAPSHOT.SNAPSHOT_TYPE`

Enums are being gradually introduced to the API. Some fields which were string until now, are replaced with an appropriate enum. One such field is `vm.type`. But this field is inherited by `snapshot`, and `snapshot` type is different than `vm` type. So a new field has been added to `snapshot` entity: `snapshot.snapshot_type`.

```
<snapshot>
```



```

...
<snapshot_type>regular|active|stateless|preview</snapshot_type>
...
</snapshot>

```

B.52. REMOVED `MOVE` ACTION FROM `VM`

The deprecated `move` action of the `VM` entity has been removed. Instead, you can move individual disks.

B.53. MOVED `REPORTED_CONFIGURATIONS.IN_SYNC` TO `NETWORK_ATTACHMENT`

In version 3 of the API the XML schema type `ReportedConfigurations` had a `in_sync` property:

```

<network_attachment>
  <reported_configurations>
    <in_sync>true</in_sync>
    <reported_configuration>
      ...
    </reported_configuration>
    ...
  </reported_configurations>
</network_attachment>

```

In the specification mechanism used by version 4 of the API this can't be expressed, because list types (the list of reported configurations) can't have attributes. To be able to represent it the attribute has been moved to the enclosing `network_attachment`:

```

<network_attachment>
  <in_sync>true</in_sync>
  <reported_configurations>
    <reported_configuration>
      ...
    </reported_configuration>
    ...
  </reported_configurations>
</network_attachment>

```

B.54. REPLACED `CAPABILITIES` WITH `CLUSTERLEVELS`

The top level `capabilities` collection has been replaced by the new `clusterlevels` collection. This new collection will contain the information that isn't available in the model, like the list of CPU types available for each cluster level:

```
GET /ovirt-engine/api/clusterlevels
```

This will return a list of `ClusterLevel` objects containing the details for all the cluster levels supported by the system:

```

<cluster_levels>
  <cluster_level id="3.6" href="/clusterlevels/3.6">
    <cpu_types>
      <cpu_type>

```

```

        <name>Intel Conroe Family</name>
        <level>2</level>
        <architecture>x86_64</architecture>
    </cpu_type>
    ...
</cpu_types>
...
</cluster_level>
</cluster_levels>

```

Each specific cluster level has its own subresource, identified by the version itself:

```
GET /ovirt-engine/api/clusterlevels/3.6
```

This will return the details of that version:

```

<cluster_level id="3.6" href="/clusterlevels/3.6">
  <cpu_types>
    <cpu_type>
      <name>Intel Conroe Family</name>
      <level>2</level>
      <architecture>x86_64</architecture>
    </cpu_type>
    ...
  </cpu_types>
  ...
</cluster_level>

```

B.55. REPLACED DISKS WITH DISKATTACHMENTS

In version 3 of the API virtual machines and templates had a **disks** collection containing all the information of the disks attached to them. In version 4 of the API these **disks** collections have been removed and replaced with a new **diskattachments** collection that will contain only the references to the disk and the attributes that are specific of the relationship between disks and the virtual machine or template that they are attached to: **interface** and **bootable**.

To find what disks are attached to a virtual machine, for example, send a request like this:

```
GET /ovirt-engine/api/vms/123/diskattachments
```

That will return a response like this:

```

<disk_attachments>
  <disk_attachment href="/vms/123/diskattachments/456" id="456">
    <bootable>>false</bootable>
    <interface>virtio</interface>
    <disk href="/disks/456" id="456"/>
    <vm href="/vms/123" id="123"/>
  </disk_attachment>
  ...
</disk_attachments>

```

To find the rest of the details of the disk, follow the link provided.

Adding disks to a virtual machine or template uses the new **disk_attachment** element as well: request like this:

```
POST /ovirt-engine/api/vms/123/diskattachments
```

With the following body if the disk doesn't exist and you want to create it:

```
<disk_attachment>
  <bootable>>false</bootable>
  <interface>virtio</interface>
  <disk>
    <description>My disk</description>
    <format>cow</format>
    <name>mydisk</name>
    <provisioned_size>1048576</provisioned_size>
    <storage_domains>
      <storage_domain>
        <name>mydata</name>
      </storage_domain>
    </storage_domains>
  </disk>
</disk_attachment>
```

Or with the following body if the disk already exists, and you just want to attach it to the virtual machine:

```
<disk_attachment>
  <bootable>>false</bootable>
  <interface>virtio</interface>
  <disk id="456"/>
</disk_attachment>
```

Take into account that the **vm.disks** and **template.disks** attributes have **disk_attachments** for all usages. For example, when creating a template the **vm.disks** element was used to indicate in which storage domain to create the disks of the template. This usage has also been replaced by **vm.disk_attachments**, so the request to create a template with disks in specific storage domains will now look like this:

```
<template>
  <name>mytemplate</name>
  <vm id="123">
    <disk_attachments>
      <disk_attachment>
        <disk id="456">
          <storage_domains>
            <storage_domain id="789"/>
          </storage_domains>
        </disk>
      </disk_attachment>
      ...
    </disk_attachments>
  </vm>
</template>
```

B.56. USE `ISCSI_TARGETS` ELEMENT TO DISCOVER UNREGISTERED STORAGE

In version 3 of the API the operation to discover unregistered storage domains used to receive a list of iSCSI targets, using multiple `iscsi_target` elements:

```
POST /ovirt-engine/api/hosts/123/unregisteredstoragedomaindiscover
```

```
<action>
  <iscsi>
    <address>myiscsiserver</address>
  </iscsi>
  <iscsi_target>iqn.2016-07.com.example:mytarget1</iscsi_target>
  <iscsi_target>iqn.2016-07.com.example:mytarget2</iscsi_target>
</action>
```

In version 4 of the API all repeating elements, like `iscsi_target` in this case, are wrapped with another element, `iscsi_targets` in case. So the same request should now look like this:

```
POST /ovirt-engine/api/hosts/123/unregisteredstoragedomaindiscover
```

```
<action>
  <iscsi>
    <address>myiscsiserver</address>
  </iscsi>
  <iscsi_targets>
    <iscsi_target>iqn.2016-07.com.example:mytarget1</iscsi_target>
    <iscsi_target>iqn.2016-07.com.example:mytarget2</iscsi_target>
  </iscsi_targets>
</action>
```