



Red Hat Virtualization 4.1

REST API ガイド

Red Hat Virtualization REST アプリケーションプログラミングインターフェイスの使用

Red Hat Virtualization 4.1 REST API ガイド

Red Hat Virtualization REST アプリケーションプログラミングインターフェイスの使用

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/REST_API_Guide.ent file |.

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, the Red Hat 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 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.

概要

本ガイドでは、Red Hat Virtualization Manager Representational State Transfer Application Programming Interface について説明します。このガイドは、ovirt-engine-api-model コードにあるドキュメントのコメントから生成されており、現在、未完状態です。本書の更新バージョンは、新しいコンテンツが利用可能になると公開されます。

目次

第1章 はじめに	35
1.1. REPRESENTATIONAL STATE TRANSFER	35
1.2. API の前提条件	35
第2章 認証とセキュリティー	37
2.1. TLS/SSL 証明書	37
2.1.1. CA 証明書の取得	37
2.1.2. クライアントへの証明書のインポート	39
2.2. 認証	39
2.2.1. OAuth 認証	39
2.2.2. Basic 認証	41
2.2.3. 認証セッション	42
2.2.3.1. 認証されたセッションの要求	42
第3章 一般的な概念	44
3.1. タイプ	44
3.2. 特定タイプ	44
3.3. オブジェクト	44
3.4. コレクション	45
3.5. 表現	45
3.5.1. XML 表現	45
3.5.2. JSON 表現	45
3.6. サービス	46
3.7. 検索	49
3.7.1. 最大結果パラメーター	49
3.7.2. ケースの機密性	49
3.7.3. 検索構文	50
3.7.4. ワイルドカード	50
3.7.5. ページネーション	50
3.8. パーミッション	51
3.9. エラーの処理	52
第4章 クイックスタートの例	53
4.1. 例：API のエントリーポイントへのアクセス	53
4.2. 例：データセンターの一覧表示	55
4.3. 例：ホストクラスターの一覧表示	56
4.4. 例：論理ネットワークの一覧表示	57
4.5. 例：ホストの一覧表示	58
4.6. 例：NFS データストレージの作成	59
4.7. 例：NFS ISO ストレージの作成	60
4.8. 例：ストレージドメインのデータセンターへの割り当て	62
4.9. 例：仮想マシンの作成	63
4.10. 例：仮想マシン NIC の作成	65
4.11. 例：仮想マシンディスクの作成	65
4.12. 例：ISO イメージの仮想マシンへの割り当て	66
4.13. 例：仮想マシンの起動	68
第5章 要求	70
第6章 サービス	99
6.1. AFFINITYGROUP	99
6.1.1. get GET	99

6.1.2. remove DELETE	99
6.1.3. update PUT	100
6.2. AFFINITYGROUPVMM	100
6.2.1. remove DELETE	100
6.3. AFFINITYGROUPVMS	100
6.3.1. add POST	100
6.3.2. list GET	101
6.3.2.1. max	101
6.4. AFFINITYGROUPS	101
6.4.1. add POST	101
6.4.2. list GET	102
6.4.2.1. max	102
6.5. AFFINITYLABEL	102
6.5.1. get GET	103
6.5.2. remove DELETE	103
6.5.3. update PUT	103
6.6. AFFINITYLABELHOST	103
6.6.1. get GET	103
6.6.2. remove DELETE	104
6.7. AFFINITYLABELHOSTS	104
6.7.1. add POST	104
6.7.2. list GET	104
6.8. AFFINITYLABELVMM	104
6.8.1. get GET	105
6.8.2. remove DELETE	105
6.9. AFFINITYLABELVMS	105
6.9.1. add POST	105
6.9.2. list GET	105
6.10. AFFINITYLABELS	106
6.10.1. add POST	106
6.10.2. list GET	106
6.10.2.1. max	106
6.11. ASSIGNEDAFFINITYLABEL	106
6.11.1. get GET	107
6.11.2. remove DELETE	107
6.12. ASSIGNEDAFFINITYLABELS	107
6.12.1. add POST	107
6.12.2. list GET	107
6.13. ASSIGNEDCPUPROFILE	108
6.13.1. get GET	108
6.13.2. remove DELETE	108
6.14. ASSIGNEDCPUPROFILES	108
6.14.1. add POST	108
6.14.2. list GET	109
6.14.2.1. max	109
6.15. ASSIGNEDDISKPROFILE	109
6.15.1. get GET	109
6.15.2. remove DELETE	109
6.16. ASSIGNEDDISKPROFILES	110
6.16.1. add POST	110
6.16.2. list GET	110
6.16.2.1. max	110
6.17. ASSIGNEDPERMISSIONS	110

6.17.1. add POST	111
6.17.2. list GET	112
6.18. ASSIGNEDROLES	112
6.18.1. list GET	112
6.18.1.1. max	112
6.19. ASSIGNEDTAG	113
6.19.1. get GET	113
6.19.2. remove DELETE	113
6.20. ASSIGNEDTAGS	114
6.20.1. add POST	114
6.20.2. list GET	114
6.20.2.1. max	115
6.21. ASSIGNEDVNICPROFILE	115
6.21.1. get GET	115
6.21.2. remove DELETE	115
6.22. ASSIGNEDVNICPROFILES	115
6.22.1. add POST	116
6.22.2. list GET	116
6.22.2.1. max	116
6.23. ATTACHEDSTORAGEDOMAIN	116
6.23.1. activate POST	116
6.23.2. deactivate POST	117
6.23.3. get GET	117
6.23.4. remove DELETE	117
6.24. ATTACHEDSTORAGEDOMAINS	117
6.24.1. add POST	118
6.24.2. list GET	118
6.24.2.1. max	118
6.25. バランス	118
6.25.1. get GET	118
6.25.2. remove DELETE	119
6.26. バランス	119
6.26.1. add POST	119
6.26.2. list GET	119
6.26.2.1. max	120
6.27. ブックマーク	120
6.27.1. get GET	120
6.27.2. remove DELETE	120
6.27.3. update PUT	121
6.28. ブックマーク	121
6.28.1. add POST	121
6.28.2. list GET	122
6.28.2.1. max	122
6.29. CLUSTER	122
6.29.1. get GET	123
6.29.2. remove DELETE	124
6.29.3. resetemulatedmachine POST	125
6.29.4. update PUT	125
6.30. CLUSTERLEVEL	125
6.30.1. get GET	126
6.31. CLUSTERLEVELS	126
6.31.1. list GET	127
6.32. CLUSTERNETWORK	127

6.32.1. get GET	127
6.32.2. remove DELETE	128
6.32.3. update PUT	128
6.33. CLUSTERNETWORKS	128
6.33.1. add POST	128
6.33.2. list GET	129
6.33.2.1. max	129
6.34. CLUSTERS	129
6.34.1. add POST	129
6.34.2. list GET	130
6.34.2.1. case_sensitive	130
6.34.2.2. max	130
6.35. コピー可能	130
6.35.1. copy POST	131
6.36. CPUPROFILE	131
6.36.1. get GET	131
6.36.2. remove DELETE	131
6.36.3. update PUT	131
6.37. CPUPROFILES	132
6.37.1. add POST	132
6.37.2. list GET	132
6.37.2.1. max	132
6.38. DATACENTER	132
6.38.1. get GET	133
6.38.2. remove DELETE	133
6.38.2.1. force	134
6.38.3. update PUT	134
6.39. DATACENTERNETWORK	134
6.39.1. get GET	135
6.39.2. remove DELETE	135
6.39.3. update PUT	135
6.40. DATACENTERNETWORKS	135
6.40.1. add POST	136
6.40.2. list GET	136
6.40.2.1. max	136
6.41. DATACENTERS	136
6.41.1. add POST	137
6.41.2. list GET	137
6.41.2.1. case_sensitive	138
6.41.2.2. max	138
6.42. ディスク	139
6.42.1. copy POST	139
6.42.2. export POST	140
6.42.3. get GET	140
6.42.4. move POST	140
6.42.5. remove DELETE	141
6.42.6. sparsify POST	141
6.43. DISKATTACHMENT	141
6.43.1. get GET	141
6.43.2. remove DELETE	142
6.43.2.1. detach_only	142
6.43.3. update PUT	142
6.44. DISKATTACHMENTS	143

6.44.1. add POST	143
6.44.2. list GET	144
6.45. DISKPROFILE	144
6.45.1. get GET	144
6.45.2. remove DELETE	145
6.45.3. update PUT	145
6.46. DISKPROFILES	145
6.46.1. add POST	145
6.46.2. list GET	145
6.46.2.1. max	146
6.47. DISKSNAPSHOT	146
6.47.1. get GET	146
6.47.2. remove DELETE	146
6.48. DISKSNAPSHOTS	146
6.48.1. list GET	147
6.48.1.1. max	147
6.49. ディスク	147
6.49.1. add POST	147
6.49.2. list GET	149
6.49.2.1. case_sensitive	150
6.49.2.2. max	150
6.49.2.3. unregistered	150
6.50. DOMAIN	150
6.50.1. get GET	151
6.51. DOMAINGROUP	151
6.51.1. get GET	151
6.52. DOMAINGROUPS	151
6.52.1. list GET	152
6.52.1.1. case_sensitive	152
6.52.1.2. max	152
6.53. DOMAINUSER	152
6.53.1. get GET	152
6.54. DOMAINUSERS	153
6.54.1. list GET	153
6.54.1.1. case_sensitive	154
6.54.1.2. max	154
6.55. ドメイン	154
6.55.1. list GET	154
6.55.1.1. max	155
6.56. ENGINEKATELLOERRATA	155
6.56.1. list GET	155
6.56.1.1. max	156
6.57. イベント	156
6.57.1. get GET	156
6.57.2. remove DELETE	157
6.58. イベント	157
6.58.1. add POST	157
6.58.2. list GET	158
6.58.2.1. case_sensitive	159
6.58.2.2. from	159
6.58.2.3. max	159
6.58.2.4. search	160
6.58.3. undelete POST	160

6.59. EXTERNALCOMPUTERESOURCE	161
6.59.1. get GET	161
6.60. EXTERNALCOMPUTERESOURCES	161
6.60.1. list GET	161
6.60.1.1. max	161
6.61. EXTERNALDISCOVEREDHOST	162
6.61.1. get GET	162
6.62. EXTERNALDISCOVEREDHOSTS	162
6.62.1. list GET	162
6.62.1.1. max	162
6.63. EXTERNALHOST	162
6.63.1. get GET	163
6.64. EXTERNALHOSTGROUP	163
6.64.1. get GET	163
6.65. EXTERNALHOSTGROUPS	163
6.65.1. list GET	163
6.65.1.1. max	164
6.66. EXTERNALHOSTPROVIDER	164
6.66.1. get GET	164
6.66.2. importcertificates POST	164
6.66.3. remove DELETE	164
6.66.4. testconnectivity POST	165
6.66.5. update PUT	165
6.67. EXTERNALHOSTPROVIDERS	165
6.67.1. add POST	165
6.67.2. list GET	165
6.67.2.1. max	166
6.68. EXTERNALHOSTS	166
6.68.1. list GET	166
6.68.1.1. max	166
6.69. EXTERNALPROVIDER	166
6.69.1. importcertificates POST	166
6.69.2. testconnectivity POST	167
6.70. EXTERNALPROVIDERCERTIFICATE	167
6.70.1. get GET	167
6.71. EXTERNALPROVIDERCERTIFICATES	167
6.71.1. list GET	167
6.71.1.1. max	168
6.72. EXTERNALVMIMPORTS	168
6.72.1. add POST	168
6.73. FENCEAGENT	169
6.73.1. get GET	169
6.73.2. remove DELETE	169
6.73.3. update PUT	169
6.74. FENCEAGENTS	169
6.74.1. add POST	170
6.74.2. list GET	170
6.74.2.1. max	170
6.75. FILE	170
6.75.1. get GET	170
6.76. ファイル	171
6.76.1. list GET	171
6.76.1.1. case_sensitive	171

6.76.1.2. max	171
6.77. フィルター	171
6.77.1. get GET	172
6.77.2. remove DELETE	172
6.78. フィルター	172
6.78.1. add POST	172
6.78.2. list GET	172
6.78.2.1. max	173
6.79. GLUSTERBRICK	173
6.79.1. get GET	173
6.79.2. remove DELETE	174
6.79.3. replace POST	174
6.80. GLUSTERBRICKS	175
6.80.1. activate POST	175
6.80.2. add POST	176
6.80.3. list GET	177
6.80.3.1. max	177
6.80.4. migrate POST	177
6.80.5. remove DELETE	178
6.80.6. stopmigrate POST	179
6.80.6.1. bricks	179
6.81. GLUSTERHOOK	179
6.81.1. disable POST	180
6.81.2. enable POST	180
6.81.3. get GET	180
6.81.4. remove DELETE	180
6.81.5. resolve POST	181
6.82. GLUSTERHOOKS	181
6.82.1. list GET	181
6.82.1.1. max	181
6.83. GLUSTERVOLUME	182
6.83.1. get GET	182
6.83.2. getprofilestatistics POST	183
6.83.3. rebalance POST	183
6.83.3.1. fix_layout	184
6.83.3.2. force	184
6.83.4. remove DELETE	184
6.83.5. resetalloptions POST	185
6.83.6. resetoption POST	185
6.83.7. setoption POST	185
6.83.8. start POST	186
6.83.8.1. force	186
6.83.9. startprofile POST	186
6.83.10. stop POST	187
6.83.11. stopprofile POST	187
6.83.12. stoprebalance POST	187
6.84. GLUSTERVOLUMES	188
6.84.1. add POST	188
6.84.2. list GET	189
6.84.2.1. case_sensitive	189
6.84.2.2. max	189
6.85. グループ	189
6.85.1. get GET	190

6.85.2. remove DELETE	190
6.86. グループ	190
6.86.1. add POST	190
6.86.2. list GET	191
6.86.2.1. case_sensitive	191
6.86.2.2. max	191
6.87. ホスト	191
6.87.1. activate POST	192
6.87.2. approve POST	193
6.87.3. commitnetconfig POST	193
6.87.4. deactivate POST	194
6.87.4.1. stop_gluster_service	194
6.87.5. enrollcertificate POST	194
6.87.6. fence POST	194
6.87.7. forceselectspm POST	195
6.87.8. get GET	195
6.87.9. install POST	196
6.87.9.1. deploy_hosted_engine	197
6.87.9.2. undeploy_hosted_engine	197
6.87.10. iscsidiscover POST	197
6.87.11. iscsilogin POST	198
6.87.12. refresh POST	198
6.87.13. remove DELETE	198
6.87.14. setupnetworks POST	199
6.87.15. unregisteredstoragedomainsdiscover POST	202
6.87.16. update PUT	203
6.87.17. upgrade POST	203
6.87.18. upgradecheck POST	203
6.88. HOSTDEVICE	204
6.88.1. get GET	204
6.89. HOSTDEVICES	204
6.89.1. list GET	204
6.89.1.1. max	205
6.90. HOSTHOOK	205
6.90.1. get GET	205
6.91. HOSTHOOKS	205
6.91.1. list GET	205
6.91.1.1. max	206
6.92. HOSTNIC	206
6.92.1. get GET	206
6.92.2. updatevirtualfunctionsconfiguration POST	206
6.93. HOSTNICS	206
6.93.1. list GET	207
6.93.1.1. max	207
6.94. HOSTNUMANODE	207
6.94.1. get GET	207
6.95. HOSTNUMANODES	207
6.95.1. list GET	208
6.95.1.1. max	208
6.96. HOSTSTORAGE	208
6.96.1. list GET	208
6.96.1.1. report_status	209
6.97. ホスト	209

6.97.1. add POST	210
6.97.1.1. deploy_hosted_engine	211
6.97.1.2. undeploy_hosted_engine	211
6.97.2. list GET	211
6.97.2.1. case_sensitive	212
6.97.2.2. max	212
6.98. アイコン	212
6.98.1. get GET	212
6.99. ICONS	213
6.99.1. list GET	213
6.99.1.1. max	213
6.100. IMAGE	213
6.100.1. get GET	213
6.100.2. import POST	214
6.101. IMAGETRANSFER	214
6.101.1. extend POST	217
6.101.2. finalize POST	218
6.101.3. get GET	218
6.101.4. pause POST	218
6.101.5. resume POST	218
6.102. IMAGETRANSFERS	218
6.102.1. add POST	219
6.102.2. list GET	219
6.103. イメージ	219
6.103.1. list GET	219
6.103.1.1. max	219
6.104. INSTANCETYPE	219
6.104.1. get GET	220
6.104.2. remove DELETE	220
6.104.3. update PUT	220
6.105. INSTANCETYPEGRAPHICSCONSOLE	221
6.105.1. get GET	221
6.105.2. remove DELETE	221
6.106. INSTANCETYPEGRAPHICSCONSOLES	222
6.106.1. add POST	222
6.106.2. list GET	222
6.106.2.1. max	222
6.107. INSTANCETYPENIC	222
6.107.1. get GET	223
6.107.2. remove DELETE	223
6.107.3. update PUT	223
6.108. INSTANCETYPENICS	223
6.108.1. add POST	224
6.108.2. list GET	224
6.108.2.1. max	224
6.109. INSTANCETYPEWATCHDOG	224
6.109.1. get GET	224
6.109.2. remove DELETE	225
6.109.3. update PUT	225
6.110. INSTANCETYPEWATCHDOGS	225
6.110.1. add POST	225
6.110.2. list GET	226
6.110.2.1. max	226

6.111. INSTANCETYPES	226
6.111.1. add POST	226
6.111.2. list GET	228
6.111.2.1. case_sensitive	228
6.111.2.2. max	228
6.112. ISCSIBOND	228
6.112.1. get GET	229
6.112.2. remove DELETE	229
6.112.3. update PUT	229
6.113. ISCSIBONDS	230
6.113.1. add POST	230
6.113.2. list GET	230
6.113.2.1. max	231
6.114. JOB	231
6.114.1. clear POST	231
6.114.2. end POST	231
6.114.2.1. succeeded	232
6.114.3. get GET	232
6.115. ジョブ	233
6.115.1. add POST	233
6.115.2. list GET	234
6.115.2.1. max	234
6.116. KATELLOERRATA	234
6.116.1. list GET	235
6.116.1.1. max	235
6.117. KATELLOERRATUM	235
6.117.1. get GET	236
6.118. MACPOOL	236
6.118.1. get GET	236
6.118.2. remove DELETE	237
6.118.3. update PUT	237
6.119. MACPOOLS	238
6.119.1. add POST	238
6.119.2. list GET	238
6.119.2.1. max	239
6.120. MEASURABLE	239
6.121. MOVEABLE	239
6.121.1. move POST	239
6.122. NETWORK	239
6.122.1. get GET	239
6.122.2. remove DELETE	240
6.122.3. update PUT	241
6.123. NETWORKATTACHMENT	241
6.123.1. get GET	241
6.123.2. remove DELETE	242
6.123.3. update PUT	242
6.124. NETWORKATTACHMENTS	242
6.124.1. add POST	242
6.124.2. list GET	242
6.124.2.1. max	243
6.125. NETWORKFILTER	243
6.125.1. get GET	243
6.126. NETWORKFILTERS	243

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

6.140. OPENSTACKVOLUMEAUTHENTICATIONKEY	258
6.140.1. get GET	258
6.140.2. remove DELETE	259
6.140.3. update PUT	259
6.141. OPENSTACKVOLUMEAUTHENTICATIONKEYS	259
6.141.1. add POST	259
6.141.2. list GET	259
6.141.2.1. max	260
6.142. OPENSTACKVOLUMEPROVIDER	260
6.142.1. get GET	260
6.142.2. importcertificates POST	260
6.142.3. remove DELETE	261
6.142.4. testconnectivity POST	261
6.142.5. update PUT	261
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	263
6.145. OPENSTACKVOLUMETYPES	263
6.145.1. list GET	263
6.145.1.1. max	263
6.146. OPERATINGSYSTEM	263
6.146.1. get GET	263
6.147. OPERATINGSYSTEMS	264
6.147.1. list GET	264
6.147.1.1. max	264
6.148. パーミッション	264
6.148.1. get GET	264
6.148.2. remove DELETE	265
6.149. PERMIT	265
6.149.1. get GET	265
6.149.2. remove DELETE	266
6.150. PERMITS	266
6.150.1. add POST	266
6.150.2. list GET	266
6.150.2.1. max	267
6.151. QOS	267
6.151.1. get GET	267
6.151.2. remove DELETE	268
6.151.3. update PUT	268
6.152. QOSS	268
6.152.1. add POST	268
6.152.2. list GET	268
6.152.2.1. max	269
6.153. クォータ	269
6.153.1. get GET	269
6.153.2. remove DELETE	269
6.153.3. update PUT	270
6.154. QUOTA_CLUSTERLIMIT	270
6.154.1. get GET	270
6.154.2. remove DELETE	271

6.155. QUOTA_CLUSTERLIMITS	271
6.155.1. add POST	271
6.155.2. list GET	271
6.155.2.1. max	271
6.156. QUOTA_STORAGELIMIT	272
6.156.1. get GET	272
6.156.2. remove DELETE	272
6.157. QUOTA_STORAGELIMITS	272
6.157.1. add POST	272
6.157.2. list GET	273
6.157.2.1. max	273
6.158. QUOTAS	273
6.158.1. add POST	273
6.158.2. list GET	273
6.158.2.1. max	274
6.159. コール	274
6.159.1. get GET	274
6.159.2. remove DELETE	274
6.159.3. update PUT	275
6.160. コール	275
6.160.1. add POST	276
6.160.2. list GET	276
6.160.2.1. max	277
6.161. SCHEDULINGPOLICIES	277
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	278
6.162.2. remove DELETE	278
6.162.3. update PUT	278
6.163. SCHEDULINGPOLICYUNIT	278
6.163.1. get GET	279
6.163.2. remove DELETE	279
6.164. SCHEDULINGPOLICYUNITS	279
6.164.1. list GET	279
6.164.1.1. max	280
6.165. スナップショット	280
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	282
6.168. SNAPSHOTDISK	282
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	283
6.170.1. get GET	283

6.171. SNAPSHOTNICS	283
6.171.1. list GET	283
6.171.1.1. max	283
6.172. スナップショット	283
6.172.1. add POST	284
6.172.2. list GET	284
6.172.2.1. max	285
6.173. SSHPUBLICKEY	285
6.173.1. get GET	285
6.173.2. remove DELETE	285
6.173.3. update PUT	285
6.174. SSHPUBLICKEYS	286
6.174.1. add POST	286
6.174.2. list GET	286
6.174.2.1. max	286
6.175. 統計	286
6.175.1. get GET	287
6.176. 統計	287
6.176.1. list GET	287
6.176.1.1. max	288
6.177. STEP	288
6.177.1. end POST	288
6.177.1.1. succeeded	289
6.177.2. get GET	289
6.178. STEPS	289
6.178.1. add POST	290
6.178.2. list GET	290
6.178.2.1. max	291
6.179. ストレージ	291
6.179.1. get GET	291
6.179.1.1. report_status	292
6.180. STORAGEDOMAIN	293
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	295
6.180.5.1. destroy	295
6.180.5.2. host	296
6.180.6. update PUT	296
6.180.7. updateovfstore POST	296
6.181. STORAGEDOMAINCONTENTDISK	297
6.181.1. get GET	297
6.182. STORAGEDOMAINCONTENTDISKS	297
6.182.1. list GET	297
6.182.1.1. case_sensitive	298
6.182.1.2. max	298
6.183. STORAGEDOMAINSERVERCONNECTION	298
6.183.1. get GET	298
6.183.2. remove DELETE	298
6.184. STORAGEDOMAINSERVERCONNECTIONS	299
6.184.1. add POST	299
6.184.2. list GET	299

6.184.2.1. max	299
6.185. STORAGEDOMAINTEMPLATE	299
6.185.1. get GET	300
6.185.2. import POST	300
6.185.2.1. clone	301
6.185.3. register POST	301
6.185.3.1. allow_partial_import	301
6.185.4. remove DELETE	302
6.186. STORAGEDOMAINTEMPLATES	302
6.186.1. list GET	302
6.186.1.1. max	302
6.186.1.2. unregistered	302
6.187. STORAGEDOMAINVM	303
6.187.1. get GET	303
6.187.2. import POST	303
6.187.2.1. clone	304
6.187.2.2. collapse_snapshots	305
6.187.3. register POST	305
6.187.3.1. allow_partial_import	305
6.187.3.2. reassign_bad_mac	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	307
6.190.1. list GET	307
6.190.1.1. max	308
6.190.1.2. unregistered	308
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	311
6.192.2.1. host	311
6.192.3. update PUT	311
6.192.3.1. force	312
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. システム	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. タグ	319
6.198.1. get GET	319
6.198.2. remove DELETE	320
6.198.3. update PUT	320
6.199. タグ	321
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	324
6.200.4. update PUT	324
6.201. TEMPLATECDROM	325
6.201.1. get GET	325
6.201.1.1. cdrom	325
6.202. TEMPLATECDROMS	326
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	327
6.203.3. get GET	327
6.203.4. remove DELETE	327
6.204. TEMPLATEDISKATTACHMENT	327
6.204.1. get GET	328
6.204.2. remove DELETE	328
6.205. TEMPLATEDISKATTACHMENTS	328
6.205.1. list GET	328
6.206. TEMPLATEDISKS	329
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	330
6.208. TEMPLATEGRAPHICSCONSOLES	330
6.208.1. add POST	330
6.208.2. list GET	330
6.208.2.1. max	331
6.209. TEMPLATENIC	331
6.209.1. get GET	331
6.209.2. remove DELETE	331
6.209.3. update PUT	331
6.210. TEMPLATENICS	332
6.210.1. add POST	332
6.210.2. list GET	332
6.210.2.1. max	332

6.211. TEMPLATEWATCHDOG	332
6.211.1. get GET	332
6.211.2. remove DELETE	333
6.211.3. update PUT	333
6.212. TEMPLATEWATCHDOGS	333
6.212.1. add POST	333
6.212.2. list GET	333
6.212.2.1. max	334
6.213. テンプレート	334
6.213.1. add POST	334
6.213.1.1. clone_permissions	335
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	337
6.215. UNMANAGEDNETWORKS	337
6.215.1. list GET	337
6.215.1.1. max	337
6.216. USER	337
6.216.1. get GET	338
6.216.2. remove DELETE	338
6.217. USERS	339
6.217.1. add POST	339
6.217.2. list GET	340
6.217.2.1. case_sensitive	340
6.217.2.2. max	341
6.218. VIRTUALFUNCTIONALLOWEDNETWORK	341
6.218.1. get GET	341
6.218.2. remove DELETE	341
6.219. VIRTUALFUNCTIONALLOWEDNETWORKS	341
6.219.1. add POST	341
6.219.2. list GET	342
6.219.2.1. max	342
6.220. VM	342
6.220.1. cancelmigration POST	343
6.220.2. clone POST	343
6.220.3. commitsnapshot POST	344
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	349

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	351
6.220.16.1. pause	351
6.220.16.2. use_cloud_init	351
6.220.16.3. use_sysprep	352
6.220.16.4. vm	352
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	356
6.222.1. list GET	356
6.222.1.1. applications	356
6.222.1.2. max	357
6.223. VMCDROM	357
6.223.1. get GET	357
6.223.1.1. current	358
6.223.2. update PUT	358
6.223.2.1. current	359
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	361
6.225.5. move POST	361
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	363
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	365

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	366
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	372
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	374
6.233.2. remove DELETE	374
6.233.3. update PUT	374
6.234. VMNUMANODES	375
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	381
6.241.1. get GET	382
6.241.1.1. watchdog	382
6.241.2. remove DELETE	382

6.241.3. update PUT	382
6.241.3.1. watchdog	383
6.242. VMWATCHDOGS	383
6.242.1. add POST	383
6.242.1.1. watchdog	384
6.242.2. list GET	384
6.242.2.1. max	384
6.242.2.2. watchdogs	384
6.243. VMS	385
6.243.1. add POST	385
6.243.1.1. clone	388
6.243.1.2. clone_permissions	388
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	390
6.244.3. update PUT	390
6.245. VNICPROFILES	391
6.245.1. add POST	391
6.245.2. list GET	392
6.245.2.1. max	392
6.246. 重み	392
6.246.1. get GET	393
6.246.2. remove DELETE	393
6.247. 重み	393
6.247.1. add POST	393
6.247.2. list GET	393
6.247.2.1. max	394
第7章 タイプ	395
7.1. ACCESSPROTOCOL ENUM	395
7.2. ACTION 構造体	395
7.3. AFFINITYGROUP STRUCT	398
7.3.1. enforcing	399
7.3.2. positive	399
7.4. AFFINITYLABEL 構造体	400
7.4.1. read_only	400
7.5. AFFINITYRULE 構造体	401
7.5.1. enabled	401
7.5.2. enforcing	401
7.5.3. positive	401
7.6. AGENT 構造体	402
7.6.1. host	402
7.7. AGENTCONFIGURATION 構造体	402
7.8. API 構造体	403
7.9. APISUMMARY 構造体	404
7.10. APISUMMARYITEM STRUCT	404
7.11. APPLICATION 構造体	405
7.12. アーキテクチャー ENUM	405
7.13. AUTHORIZEDKEY 構造体	405
7.14. AUTONUMASTATUS ENUM	406

7.15. BALANCE 構造体	406
7.16. BIOS 構造体	407
7.17. BLOCKSTATISTIC 構造体	407
7.18. BONDING 構造体	407
7.18.1. ad_partner_mac	407
7.18.2. options	408
7.18.3. slaves	408
7.18.4. active_slave	408
7.19. BOOKMARK 構造体	408
7.20. BOOT 構造体	409
7.21. BOOTDEVICE ENUM	409
7.22. BOOTMENU STRUCT	409
7.23. BOOTPROTOCOL ENUM	409
7.23.1. autoconf	410
7.23.2. dhcp	410
7.24. BRICKPROFILEDETAIL 構造体	410
7.25. CDROM 構造体	410
7.25.1. vms	411
7.26. CERTIFICATE 構造体	411
7.27. CLOUDINIT STRUCT	412
7.28. CLUSTER 構造体	412
7.28.1. custom_scheduling_policy_properties	416
7.28.2. fencing_policy	416
7.28.3. gluster_tuned_profile	417
7.28.4. required_rng_sources	417
7.28.5. version	417
7.28.6. scheduling_policy	418
7.29. CLUSTERLEVEL 構造体	419
7.30. CONFIGURATION 構造体	419
7.30.1. data	419
7.31. CONFIGURATIONTYPE ENUM	421
7.32. CONSOLE 構造体	421
7.33. CORE 構造体	421
7.34. CPU 構造体	421
7.35. CPUMODE ENUM	422
7.36. CUPROFILE 構造体	422
7.37. CPUTOPOLOGY 構造体	423
7.38. CPUTUNE 構造体	423
7.39. CPUTYPE 構造体	423
7.40. CREATIONSTATUS ENUM	424
7.41. CUSTOMPROPERTY 構造体	424
7.42. DATACENTER 構造体	424
7.42.1. version	425
7.43. DATACENTERSTATUS ENUM	426
7.44. DEVICE 構造体	426
7.44.1. vms	427
7.45. DISK 構造体	427
7.45.1. active	429
7.45.2. actual_size	429
7.45.3. bootable	429
7.45.4. initial_size	429
7.45.5. interface	429
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	431
7.45.12. vms	432
7.46. DISKATTACHMENT 構造体	432
7.46.1. active	432
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 構造体	435
7.50.1. active	436
7.50.2. actual_size	436
7.50.3. bootable	436
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	437
7.50.9. wipe_after_delete	437
7.50.10. statistics	438
7.50.11. storage_domains	439
7.50.12. vms	439
7.51. DISKSTATUS ENUM	439
7.52. DISKSTORAGETYPE ENUM	439
7.53. DISKTYPE ENUM	440
7.54. DISPLAY 構造体	440
7.55. DISPLAYTYPE ENUM	441
7.56. DNS 構造体	441
7.57. DNSRESOLVERCONFIGURATION 構造体	441
7.57.1. name_servers	441
7.58. DOMAIN 構造体	442
7.58.1. users	442
7.59. ENTITYEXTERNALSTATUS ENUM	442
7.59.1. error	443
7.59.2. failure	443
7.60. ENTITYPROFILEDETAIL STRUCT	443
7.61. ERRORHANDLING 構造体	443
7.62. EVENT 構造体	443
7.62.1. correlation_id	444
7.62.2. flood_rate	444
7.62.3. cluster	444
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 構造体	446
7.66. EXTERNALHOSTGROUP STRUCT	447
7.67. EXTERNALHOSTPROVIDER STRUCT	447
7.67.1. requires_authentication	448
7.68. EXTERNALPROVIDER 構造体	449
7.68.1. requires_authentication	449
7.69. EXTERNALSTATUS ENUM	449
7.70. EXTERNALSYSTEMTYPE ENUM	450
7.71. EXTERNALVMIMPORT STRUCT	450
7.71.1. url	450
7.71.2. cpu_profile	451
7.71.3. drivers_iso	451
7.71.4. host	451
7.71.5. quota	451
7.71.6. vm	451
7.72. EXTERNALVMPROVIDERTYPE ENUM	451
7.73. FAULT 構造体	452
7.74. FENCETYPE ENUM	452
7.75. FENCINGPOLICY 構造体	452
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	453
7.76. FILE 構造体	453
7.77. FILTER 構造体	454
7.78. FLOPPY 構造体	454
7.78.1. vms	455
7.79. FOPSTATISTIC STRUCT	455
7.80. GLUSTERBRICK 構造体	455
7.80.1. vms	457
7.81. GLUSTERBRICKADVANCEDDETAILS STRUCT	457
7.81.1. vms	457
7.82. GLUSTERBRICKMEMORYINFO STRUCT	458
7.83. GLUSTERBRICKSTATUS ENUM	458
7.84. GLUSTERCLIENT 構造体	458
7.85. GLUSTERHOOK STRUCT	458
7.86. GLUSTERHOOKSTATUS ENUM	459
7.87. GLUSTERMEMORYPOOL STRUCT	460
7.88. GLUSTERSERVERHOOK STRUCT	460
7.89. GLUSTERSTATE ENUM	461
7.90. GLUSTERVOLUME 構造体	461
7.91. GLUSTERVOLUMEPROFILEDETAILS STRUCT	462
7.92. GLUSTERVOLUMESTATUS ENUM	462
7.93. GLUSTERVOLUMETYPE ENUM	463
7.93.1. disperse	463
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	464
7.93.7. replicate	464
7.93.8. stripe	464

7.93.9. striped_replicate	465
7.94. GRACEPERIOD 構造体	465
7.95. GRAPHICSCONSOLE 構造体	465
7.96. GRAPHICSTYPE ENUM	466
7.97. GROUP 構造体	466
7.97.1. roles	467
7.98. GUESTOPERATINGSYSTEM STRUCT	467
7.99. HARDWAREINFORMATION 構造体	467
7.100. HIGHAVAILABILITY 構造体	468
7.100.1. priority	468
7.101. HOOK 構造体	469
7.102. HOOKCONTENTTYPE ENUM	469
7.103. HOOKSTAGE ENUM	470
7.104. HOOKSTATUS ENUM	470
7.105. HOST 構造体	470
7.105.1. external_status	472
7.105.2. kdump_status	472
7.105.3. ksm	472
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	473
7.105.9. status_detail	473
7.105.10. transparent_huge_pages	473
7.105.11. version	474
7.105.12. katello_errata	475
7.105.13. statistics	476
7.106. HOSTDEVICE 構造体	477
7.106.1. driver	478
7.107. HOSTDEVICEPASSTHROUGH 構造体	478
7.108. HOSTNIC STRUCT	478
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	482
7.109.2. number_of_virtual_functions	482
7.110. HOSTPROTOCOL ENUM	482
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	484
7.111.6. non_operational	484
7.111.7. non_responsive	484
7.111.8. pending_approval	485
7.111.9. preparing_for_maintenance	485
7.112. HOSTSTORAGE 構造体	485
7.112.1. nfs_retrans	486
7.112.2. nfs_timeo	486

7.113. HOSTTYPE ENUM	486
7.113.1. ovirt_node	486
7.114. HOSTEDENGINE 構造体	487
7.115. ICON 構造体	487
7.115.1. media_type	487
7.116. IDENTIFIED 構造体	488
7.117. IMAGE 構造体	488
7.118. IMAGETRANSFER 構造体	488
7.118.1. direction	489
7.118.2. phase	489
7.118.3. proxy_url	489
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	491
7.120.6. paused_system	491
7.120.7. resuming	491
7.120.8. unknown	491
7.121. INHERITABLEBOOLEAN ENUM	492
7.122. INITIALIZATION 構造体	492
7.123. INSTANCETYPE 構造体	493
7.123.1. cpu	496
7.123.2. custom_compatibility_version	497
7.123.3. high_availability	497
7.123.4. large_icon	497
7.123.5. lease	497
7.123.6. memory	497
7.123.7. migration_downtime	498
7.123.8. origin	498
7.123.9. small_icon	498
7.123.10. sso	498
7.124. IO 構造体	499
7.125. IP 構造体	499
7.125.1. address	500
7.125.2. netmask	500
7.125.3. version	500
7.126. IPADDRESSASSIGNMENT STRUCT	500
7.127. IPVERSION ENUM	501
7.128. ISCSIBOND STRUCT	501
7.129. ISCSIDetails STRUCT	501
7.130. JOB 構造体	502
7.130.1. external	503
7.131. JOBSTATUS ENUM	503
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	505
7.134. KERNEL 構造体	505
7.135. KSM 構造体	506
7.136. LOGSEVERITY ENUM	506
7.136.1. alert	506
7.136.2. error	506
7.136.3. normal	506
7.136.4. warning	506
7.137. LOGICALUNIT 構造体	507
7.137.1. discard_max_size	508
7.137.2. discard_zeroes_data	508
7.138. LUNSTATUS ENUM	508
7.139. MAC 構造体	508
7.140. MACPOOL 構造体	508
7.140.1. allow_duplicates	509
7.140.2. default_pool	509
7.140.3. ranges	509
7.141. MEMORYOVERCOMMIT STRUCT	509
7.142. MEMORYPOLICY 構造体	509
7.142.1. max	510
7.143. MESSAGEBROKERTYPE ENUM	510
7.144. METHOD 構造体	510
7.145. MIGRATEONERROR ENUM	510
7.146. MIGRATIONBANDWIDTH 構造体	511
7.146.1. custom_value	511
7.147. MIGRATIONBANDWIDTHASSIGNMENTMETHOD ENUM	511
7.147.1. auto	511
7.148. MIGRATIONOPTIONS 構造体	512
7.149. MIGRATIONPOLICY 構造体	512
7.150. NETWORK 構造体	512
7.150.1. usages	514
7.151. NETWORKATTACHMENT 構造体	514
7.151.1. dns_resolver_configuration	517
7.151.2. properties	517
7.152. NETWORKCONFIGURATION 構造体	519
7.153. NETWORKFILTER 構造体	519
7.153.1. version	520
7.154. NETWORKLABEL 構造体	520
7.155. NETWORKPLUGINTYPE ENUM	520
7.156. NETWORKSTATUS ENUM	520
7.157. NETWORKUSAGE ENUM	521
7.157.1. management	521
7.158. NFSPROFILEDETAIL STRUCT	521
7.159. NFSVERSION ENUM	521
7.159.1. v4_2	522
7.160. NIC 構造体	522
7.160.1. network	523
7.160.2. vms	523
7.161. NICCONFIGURATION 構造体	523
7.162. NICINTERFACE ENUM	524
7.163. NICSTATUS ENUM	524
7.164. NUMANODE STRUCT	524
7.164.1. statistics	525

7.165. NUMANODEPIN STRUCT	526
7.165.1. host_numa_node	527
7.165.2. pinned	527
7.166. NUMATUNEMODE ENUM	527
7.167. OPENSTACKIMAGE 構造体	527
7.168. OPENSTACKIMAGEPROVIDER STRUCT	527
7.168.1. requires_authentication	528
7.169. OPENSTACKNETWORK 構造体	528
7.170. OPENSTACKNETWORKPROVIDER 構造体	529
7.170.1. read_only	530
7.170.2. requires_authentication	530
7.171. OPENSTACKNETWORKPROVIDERTYPE ENUM	530
7.171.1. external	531
7.171.2. neutron	531
7.172. OPENSTACKPROVIDER 構造体	531
7.172.1. requires_authentication	532
7.173. OPENSTACKSUBNET 構造体	532
7.173.1. ip_version	532
7.174. OPENSTACKVOLUMEPROVIDER 構造体	532
7.174.1. requires_authentication	533
7.175. OPENSTACKVOLUMETYPE STRUCT	534
7.176. OPENSTACKVOLUMEAUTHENTICATIONKEY STRUCT	534
7.177. OPENSTACKVOLUMEAUTHENTICATIONKEYUSAGETYPE ENUM	535
7.178. OPERATINGSYSTEM STRUCT	535
7.178.1. custom_kernel_cmdline	535
7.178.2. reported_kernel_cmdline	536
7.179. OPERATINGSYSTEMINFO STRUCT	536
7.180. OPTION 構造体	536
7.181. OSTYPE ENUM	537
7.182. PACKAGE STRUCT	538
7.183. PAYLOAD 構造体	538
7.184. PAYLOADENCODING ENUM	538
7.185. PERMISSION STRUCT	539
7.186. PERMIT 構造体	539
7.187. PMPROXY 構造体	540
7.188. PMPROXYTYPE ENUM	540
7.189. POLICYUNITTYPE ENUM	540
7.190. PORTMIRRORING STRUCT	541
7.191. POWERMANAGEMENT STRUCT	541
7.191.1. agents	541
7.191.2. automatic_pm_enabled	542
7.191.3. kdump_detection	542
7.191.4. type	542
7.192. POWERMANAGEMENTSTATUS ENUM	542
7.193. PRODUCT 構造体	542
7.194. PRODUCTINFO STRUCT	543
7.194.1. vendor	543
7.195. PROFILEDETAIL STRUCT	543
7.196. PROPERTY 構造体	544
7.197. PROXYTICKET STRUCT	544
7.198. QCOWVERSION ENUM	544
7.198.1. qcow2_v3	544
7.199. QOS STRUCT	545

7.199.1. cpu_limit	546
7.199.2. inbound_average	546
7.199.3. inbound_burst	546
7.199.4. inbound_peak	546
7.199.5. max_iops	547
7.199.6. max_read_iops	547
7.199.7. max_read_throughput	547
7.199.8. max_throughput	547
7.199.9. max_write_iops	547
7.199.10. max_write_throughput	547
7.199.11. outbound_average	547
7.199.12. outbound_average_linkshare	548
7.199.13. outbound_average_realtime	548
7.199.14. outbound_average_upperlimit	548
7.199.15. outbound_burst	548
7.199.16. outbound_peak	548
7.200. QOSTYPE ENUM	549
7.201. QUOTA 構造体	549
7.202. QUOTA_CLUSTERLIMIT STRUCT	550
7.203. QUOTA_MODETYPE ENUM	551
7.204. QUOTA_STORAGELIMIT STRUCT	551
7.205. RANGE 構造体	552
7.206. RATE 構造体	552
7.207. REPORTEDCONFIGURATION 構造体	552
7.208. REPORTEDDEVICE 構造体	552
7.209. REPORTEDDEVICETYPE ENUM	553
7.210. RESOLUTIONTYPE ENUM	553
7.211. RNGDEVICE 構造体	553
7.212. RNGSOURCE ENUM	554
7.212.1. urandom	554
7.213. ROLE 構造体	554
7.213.1. mutable	555
7.214. ROLETYPE ENUM	555
7.215. SCHEDULINGPOLICY STRUCT	555
7.216. SCHEDULINGPOLICYUNIT STRUCT	556
7.217. SCSIGENERICIO ENUM	556
7.218. SELINUX 構造体	557
7.219. SELINUXMODE ENUM	557
7.220. SERIALNUMBER 構造体	557
7.221. SERIALNUMBERPOLICY ENUM	557
7.222. SESSION 構造体	558
7.222.1. console_user	558
7.222.2. ip	558
7.222.3. protocol	558
7.222.4. user	559
7.223. SKIPIFCONNECTIVITYBROKEN STRUCT	559
7.223.1. enabled	559
7.223.2. threshold	559
7.224. SKIPIFSDACTIVE STRUCT	559
7.224.1. enabled	559
7.225. SNAPSHOT 構造体	560
7.225.1. cpu	563
7.225.2. custom_compatibility_version	564

7.225.3. high_availability	564
7.225.4. large_icon	564
7.225.5. lease	564
7.225.6. memory	564
7.225.7. migration_downtime	565
7.225.8. next_run_configuration_exists	565
7.225.9. origin	565
7.225.10. persist_memorystate	565
7.225.11. placement_policy	565
7.225.12. small_icon	566
7.225.13. sso	566
7.225.14. stop_reason	566
7.225.15. affinity_labels	568
7.225.16. katello_errata	568
7.225.17. original_template	569
7.226. SNAPSHOTSTATUS ENUM	569
7.227. SNAPSHOTTYPE ENUM	569
7.228. SPECIALOBJECTS 構造体	569
7.229. SPM 構造体	570
7.230. SPMSTATUS ENUM	570
7.231. SSH 構造体	570
7.232. SSHAUTHENTICATIONMETHOD ENUM	571
7.233. SSHPUBLICKEY STRUCT	571
7.234. SSO 構造体	571
7.235. SSOMETHOD ENUM	572
7.236. 統計 STRUCT	572
7.237. STATISTICKIND ENUM	574
7.238. STATISTICUNIT ENUM	574
7.239. STEP 構造体	574
7.239.1. external	575
7.240. STEPENUM ENUM	575
7.240.1. executing	576
7.240.2. finalizing	576
7.240.3. rebalancing_volume	576
7.240.4. removing_bricks	576
7.240.5. unknown	576
7.240.6. validating	576
7.241. STEPSTATUS ENUM	576
7.241.1. aborted	577
7.241.2. finished	577
7.241.3. started	577
7.241.4. unknown	577
7.242. STORAGECONNECTION 構造体	577
7.243. STORAGECONNECTIONEXTENSION 構造体	578
7.244. STORAGEDOMAIN 構造体	579
7.244.1. discard_after_delete	581
7.244.2. supports_discard	581
7.244.3. supports_discard_zeroes_data	581
7.244.4. wipe_after_delete	581
7.244.5. data_center	582
7.245. STORAGEDOMAINLEASE 構造体	582
7.246. STORAGEDOMAINSTATUS ENUM	583
7.247. STORAGEDOMAINTYPE ENUM	583

7.248. STORAGEFORMAT ENUM	584
7.249. STORAGETYPE ENUM	584
7.249.1. cinder	584
7.249.2. glance	584
7.249.3. glusterfs	584
7.250. SWITCHTYPE ENUM	585
7.251. TAG 構造体	585
7.252. TEMPLATE 構造体	585
7.252.1. cpu	588
7.252.2. custom_compatibility_version	588
7.252.3. high_availability	588
7.252.4. large_icon	589
7.252.5. lease	589
7.252.6. memory	589
7.252.7. migration_downtime	589
7.252.8. origin	589
7.252.9. small_icon	590
7.252.10. sso	590
7.253. TEMPLATESTATUS ENUM	591
7.254. TEMPLATEVERSION 構造体	591
7.254.1. version_number	591
7.255. TICKET 構造体	592
7.256. TIMEZONE 構造体	592
7.256.1. utc_offset	592
7.257. TRANSPARENTHUGEPAGES 構造体	592
7.258. TRANSPORTTYPE ENUM	592
7.259. UNMANAGEDNETWORK 構造体	593
7.260. USB 構造体	593
7.261. USBTYPE ENUM	593
7.261.1. legacy	594
7.262. USER 構造体	594
7.262.1. namespace	595
7.262.2. principal	595
7.262.3. user_name	595
7.263. VALUE 構造体	595
7.264. VALUETYPE ENUM	596
7.265. VCPUPIN 構造体	596
7.266. VENDOR 構造体	596
7.267. VERSION 構造体	596
7.268. VIRTIO SCSI STRUCT	597
7.269. VIRTUALNUMANODE STRUCT	597
7.269.1. statistics	598
7.270. VLAN 構造体	599
7.271. VM 構造体	599
7.271.1. cpu	602
7.271.2. custom_compatibility_version	603
7.271.3. high_availability	603
7.271.4. large_icon	603
7.271.5. lease	603
7.271.6. memory	603
7.271.7. migration_downtime	604
7.271.8. next_run_configuration_exists	604
7.271.9. origin	604

7.271.10. placement_policy	605
7.271.11. small_icon	605
7.271.12. sso	605
7.271.13. stop_reason	605
7.271.14. affinity_labels	607
7.271.15. katello_errata	607
7.271.16. original_template	608
7.272. VMAFFINITY ENUM	608
7.273. VMBASE 構造体	608
7.273.1. cpu	610
7.273.2. custom_compatibility_version	611
7.273.3. high_availability	611
7.273.4. large_icon	611
7.273.5. lease	611
7.273.6. memory	611
7.273.7. migration_downtime	612
7.273.8. origin	612
7.273.9. small_icon	612
7.273.10. sso	612
7.274. VMDEVICETYPE ENUM	613
7.275. VMPLACEMENTPOLICY 構造体	613
7.276. VMPOOL 構造体	613
7.276.1. auto_storage_select	614
7.276.2. stateful	614
7.276.3. instance_type	615
7.277. VMPOOLTYPE ENUM	615
7.278. VMSTATUS ENUM	615
7.278.1. paused	616
7.278.2. powering_up	616
7.278.3. restoring_state	616
7.278.4. saving_state	617
7.278.5. suspended	617
7.278.6. unknown	617
7.278.7. up	617
7.278.8. wait_for_launch	617
7.279. VMSUMMARY 構造体	617
7.280. VMTYPE ENUM	617
7.280.1. desktop	618
7.280.2. server	618
7.281. VNICPASSTHROUGH 構造体	618
7.282. VNICPASSTHROUGHMODE ENUM	618
7.283. VNICPROFILE 構造体	618
7.283.1. migratable	619
7.283.2. pass_through	619
7.283.3. port_mirroring	619
7.283.4. network_filter	620
7.283.5. qos	620
7.284. VNICPROFILEMAPPING 構造体	620
7.285. VOLUMEGROUP STRUCT	621
7.286. WATCHDOG 構造体	622
7.286.1. model	622
7.286.2. vms	622
7.287. WATCHDOGACTION ENUM	623

7.287.1. none	623
7.288. WATCHDOGMODEL ENUM	623
7.289. WEIGHT 構造体	623
付録A プリミティブ型	625
A.1. 文字列 プリミティブ	625
A.2. ブール値 プリミティブ	625
A.3. 整数 プリミティブ	625
A.4. 10 進 プリミティブ型	626
A.5. DATE プリミティブ	626
付録B API バージョン 4 の変更点	628
B.1. YAML サポートの削除	628
B.2. 複合型の名前変更	628
B.3. STATUS タイプの ENUM 型への置き換え	629
B.4. NIC NETWORK と PORT_MIRRORING プロパティの削除	630
B.5. NIC ACTIVE プロパティの削除	630
B.6. ディスク TYPE プロパティの削除	631
B.7. ディスク SIZE プロパティの削除	631
B.8. VM を単一のホストに固定するためのサポートを削除	631
B.9. CAPABILITIES.PERMITS 要素の削除	631
B.10. STORAGE_MANAGER 要素の削除	632
B.11. データセンターの STORAGE_TYPE 要素の削除	632
B.12. TIMEZONE 要素の削除	632
B.13. GUEST_INFO 要素の削除	632
B.14. CPU ID 属性の TYPE 要素への置き換え	633
B.15. CPU トポロジーで属性の代わりに要素を使用する	634
B.16. VCPU ピンで属性の代わりに要素を使用する	634
B.17. VCPU ピンで属性の代わりに要素を使用する	634
B.18. メモリーのオーバーコミットで属性の代わりに要素を使用する	635
B.19. CONSOLE で属性の代わりに要素を使用する	635
B.20. VIRTIO SCSI で属性の代わりに要素を使用する	635
B.21. 電源管理エージェント TYPE に属性ではなく要素を使用する	636
B.22. 電源管理エージェントオプションで属性の代わりに要素を使用する	636
B.23. IP アドレスで属性の代わりに要素を使用する	636
B.24. MAC アドレスで属性の代わりに要素を使用する	637
B.25. ブートデバイスで属性の代わりに要素を使用する	637
B.26. オペレーティングシステム TYPE に属性の代わりに要素を使用する	637
B.27. ホストを取得するリクエストからの FORCE パラメーターの削除	637
B.28. 非推奨のホスト電源管理設定の削除	638
B.29. 複数の BOOT の代わりに複数の BOOT.DEVICES.DEVICE を使用する	638
B.30. DISKS.CLONE と DISKS.DETACH_ONLY 要素の削除	639
B.31. 要素 VMPOOL の名前を VM_POOL に変更する	640
B.32. 複数の LOGICAL_UNIT の代わりに LOGICAL_UNITS を使用する	640
B.33. SNAPSHOTS.COLLAPSE_SNAPSHOTS 要素の削除	641
B.34. STORAGE と HOST_STORAGE 要素の名前の変更	641
B.35. PERMISSIONS.CLONE 要素の削除	642
B.36. 乱数ジェネレーター SOURCE 要素の名前の変更	643
B.37. 中間の TAG.PARENT 要素の削除	644
B.38. スケジューリングの組み込み名としきい値の削除	644
B.39. BRICKS.REPLICA_COUNT および BRICKS.STRIPE_COUNT の削除	645
B.40. 統計 TYPE プロパティの名前を KIND に変更する	646
B.41. 複数の VCPU_PIN の代わりに複数の VCPU_PINS.VCPU_PIN を使用する	646

B.42. FORCE パラメーターを使用してデータセンターを強制的に削除する	647
B.43. FORCE パラメーターを使用してホストを強制的に削除する	647
B.44. ストレージドメインの強制削除にパラメーターを使用する	647
B.45. HOST パラメーターを使用したストレージサーバー接続の削除	648
B.46. FORCE と STORAGE_DOMAIN パラメーターを使用したテンプレートディスクの削除	648
B.47. VM ディスク API を介してディスクを削除しないでください。	649
B.48. FORCE クエリーパラメーターを使用して、仮想マシンを強制的に削除する	649
B.49. 複数のブリックを削除するには、DELETE の代わりに POST を使用する	649
B.50. SCHEDULING_POLICY.POLICY 要素の削除	650
B.51. SNAPSHOT.SNAPSHOT_TYPE の追加	650
B.52. VM からの MOVE アクションの削除	650
B.53. REPORTED_CONFIGURATIONS.IN_SYNC を NETWORK_ATTACHMENT に移動する	650
B.54. CAPABILITIES を CLUSTERLEVELS に置き換える	651
B.55. DISKS を DISKATTACHMENTS に置き換える	652
B.56. ISCSI_TARGETS 要素を使用して、未登録のストレージを検出する	653

第1章 はじめに

Red Hat Virtualization Manager には、**Representational State Transfer (REST) API**が含まれています。ソフトウェア開発者やシステム管理者は、この API を使用することで、標準の Web インターフェイス以外で Red Hat Virtualization 環境を制御できるようになります。API は、開発者および管理者が Red Hat Virtualization 環境の機能を標準の Hypertext Transfer Protocol (HTTP) 経由で API にアクセスする外部アプリケーションやカスタムスクリプトと統合する場合に便利です。

API の利点は以下のとおりです。

- 幅広いクライアントサポート: HTTP プロトコルをサポートする各種プログラミング言語、フレームワークまたはシステムで API を使用できます。
- 自己記述型: 実行時に多くの詳細が発見されるため、クライアントアプリケーションでは、仮想化インフラストラクチャーの情報は最小限で済みます。
- リソースベースのモデル: リソースベースの REST モデルにより仮想化プラットフォームを自然な形で管理することが可能です。

これにより、開発者および管理者は以下のような作業を行うことができます。

- エンタープライズ IT システムとの統合
- サードパーティーの仮想化ソフトウェアとの統合
- 自動メンテナンスやエラーチェックなどのタスクの実行
- スクリプトを使って、Red Hat Virtualization 環境の反復タスクを自動化します。

本書は、Red Hat Virtualization API のリファレンスとしての役割を果たします。本ガイドでは、開発者および管理者を対象に、指定の SDK を使用するか、直接、API を経由して Red Hat Virtualization 環境の利用方法のステップと実例を記載しています。

1.1. REPRESENTATIONAL STATE TRANSFER

Representational State Transfer (REST) は、特定サービスとその表現にフォーカスする設計アーキテクチャーです。リソース表現は、情報の主要な抽象化で、サーバー上の特定の1つの管理要素に対応します。クライアントは Uniform Resource Identifier (URI) にある server 要素に要求を送信し、**GET**、**POST**、**PUT**、**DELETE** などの標準の HTTP メソッドで操作を実行します。これにより、クライアントとサーバー間でステートレスに通信が行われ、各要求が他の要求とは独立して機能し、要求完了必要なすべての情報が含まれます。

1.2. API の前提条件

Red Hat Virtualization API を使用するための前提条件

- API を含む Red Hat Virtualization Manager のネットワークインストール。
- API サーバーから HTTP 要求を開始および受信するクライアントまたはプログラミングライブラリー。以下に例を示します。
 - [oVirt Python SDK](#)
 - [oVirt Ruby SDK](#)
 - [oVirt Java SDK](#)

- [cURL](#) コマンドラインツール。
- [RESTClient](#) (RESTful Web サービスのデバッガー)
- HTTP (Hypertext Transfer Protocol) の知識 (REST API の対話に使用されるプロトコル)。Internet Engineering Task Force (IETF) は、<http://www.ietf.org/rfc/rfc2616.txt> でハイパーテキスト転送プロトコルを説明する Request for Comments (RFC) を提供しています。
- API がリソース表現の構築に使用する Extensible Markup Language(XML) または JavaScript Object Notation(JSON) の知識。W3C は、<http://www.w3.org/TR/xml> で XML の完全な仕様を提供します。ECMA International は、<http://www.ecma-international.org> で JSON で無料公開を提供します。

第2章 認証とセキュリティ

2.1. TLS/SSL 証明書

Red Hat Virtualization API には Hypertext Transfer Protocol Secure (HTTPS) が必要です。[1] SDK や CLI コンポーネントなどのクライアントソフトウェアとのセキュアな対話これには、サーバーが使用する CA 証明書を取得して、クライアントの証明書ストアにインポートする必要があります。

2.1.1. CA 証明書の取得

Red Hat Virtualization Manager から CA 証明書を取得し、以下のいずれかの方法でクライアントマシンに転送できます。

方法 1

CA 証明書の取得で推奨の方法は、**openssl s_client** コマンドラインツールを使用してサーバーとの実際の TLS ハンドシェイクを実行し、サーバーが提示する証明書を抽出する方法です。以下のようなコマンドを実行します。

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

このコマンドはサーバーに接続し、以下のような出力を表示します。

```
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-----
MIIEaTCCA1GgAwIBAgICEAQwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMCMVVMx
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW5lbnEudXhhdXBs
SVlJe7e5FTEtHJGTAEWWM6dGbsFhip5VXM0gfqg=
-----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-----
MIIDxjCCAq6gAwIBAgICEAAwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMCMVVMx
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW5lbnEudXhhdXBs
Pkyg1rQHR6ebGQ==
-----END CERTIFICATE-----
```

-----BEGIN CERTIFICATE----- と -----END CERTIFICATE----- のマークの間のテキストは、サーバーで提示された証明書が表示されます。最初のものはサーバー自体の証明書で、最後の証明書は CA の証明書です。これらのマークを含む CA 証明書を **ca.crt** ファイルにコピーします。結果は以下ようになります。

```
-----BEGIN CERTIFICATE-----
MIIDxjCCAq6gAwIBAgICEAAwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMCMVVMx
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW5lbnEudXhhdXBs
```

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



重要

これは、サーバーが使用する CA 証明書を取得する最も信頼性の高い方法です。ここで説明されている残りの方法はほとんどの場合に機能しますが、サーバーの管理者が手動で置き換えられた場合は、正しい CA 証明書は取得されません。

方法 2

上記の `openssl s_client` メソッドを使用できない場合は、コマンドラインツールを使用して Red Hat Virtualization Manager から CA 証明書をダウンロードできます。

コマンドラインツールの例には、`curl` と `wget` があります。これらはいずれも、複数のプラットフォームで利用できます。

`curl` を使用している場合:

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

`wget` を使用している場合:

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

方法 3

Web ブラウザーを使用して、配置されている証明書に移動します。

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

選択したブラウザーに応じて、証明書はブラウザーのキーストアにダウンロードまたはインポートします。

1. **ブラウザーにより証明書がダウンロードされた場合** には、ファイルを `ca.crt` として保存します。
2. **ブラウザーにより証明書がインポートされた場合** には、ブラウザーの認定オプションから証明書をエクスポートし、`ca.crt` として保存します。

方法 4

Red Hat Virtualization Manager にログインし、トラストストアから証明書をエクスポートして、クライアントマシンにコピーします。

1. Red Hat Virtualization Manager マシンに `root` としてログインします。
2. Java `keytool` 管理ユーティリティーを使用して、トラストストアから証明書をエクスポートします。

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

これにより、**ca.crt** という名前の証明書ファイルが作成されます。

3. **scp** コマンドを使用して、証明書をクライアントマシンにコピーします。

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

これらの各メソッドにより、クライアントマシンの **ca.crt** という名前の証明書ファイルが作成されます。次に、このファイルをクライアントの証明書ストアにインポートする必要があります。

2.1.2. クライアントへの証明書のインポート

クライアントへの証明書のインポートは、クライアントが証明書を保存し、解釈する方法に依存します。証明書のインポートに関する詳細は、クライアントのドキュメントを参照してください。

2.2. 認証

Red Hat Virtualization Manager アカウントを持つユーザーは、API にアクセスできます。すべての要求は、以下で説明するように **OAuth** または **Basic** 認証を使用して認証する必要があります。

2.2.1. OAuth 認証

Red Hat Virtualization バージョン 4.0 以降、[RFC 6749](#) に説明されているように、推奨の認証メカニズムは **OAuth 2.0** です。

OAuth は高性能なプロトコルであり、認可およびアクセストークンを取得するメカニズムがいくつかあります。Red Hat Virtualization API と合わせて使用する場合には、[RFC 6749](#) の [セクション 4.3](#) で説明されているように **Resource Owner Password Credentials Grant** (リソース所有者のパスワード認証情報の付与) のみがサポートされます。

最初に **トークン** を取得し、ユーザー名とパスワードを Red Hat Virtualization Manager のシングルサインオンサービスに送信する必要があります。

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

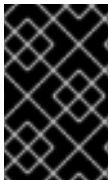
リクエスト本文には、**grant_type**、**scope**、**username**、および **password** パラメーターが含まれている必要があります。

表2.1 OAuth トークン要求パラメーター

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

これらのパラメーターは URL でエンコード されている必要があります。たとえば、ユーザー名の @ 文字を %40 としてエンコードする必要があります。結果のリクエスト本文は以下のようになります。

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



重要

scope パラメーターは OAuth RFC で任意として説明されていますが、Red Hat Virtualization API と併用する場合は必須であり、この値は **ovirt-app-api** である必要があります。

ユーザー名とパスワードが有効な場合、Red Hat Virtualization Manager のシングルサインオンサービスは、以下のような JSON ドキュメントで応答します。

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

API 認証の目的で、関連する唯一の名前/値のペアは **access_token** です。これは操作しないでください。SSO サービスが提供するとおりに使用してください。

トークンを取得すると、HTTP **Authorization** ヘッダーにトークンを追加して、**Bearer** スキームで API への要求を実行するために使用できます。たとえば、仮想マシンの一覧を取得するには、以下のような要求を送信します。

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

トークンは、複数の要求に対して複数回使用できますが、最終的に期限切れになります。期限が切れると、サーバーは 401 HTTP 応答コードで要求を拒否します。

```
HTTP/1.1 401 Unauthorized
```

これが生じる場合は、Red Hat Virtualization Manager のシングルサインオンサービスは現在トークンの更新をサポートしていないため、新しいトークンが必要です。上記と同じ方法を使用して新しいトークンを要求できます。

2.2.2. Basic 認証



重要

Basic 認証は後方互換性としてのみサポートされます。Red Hat Virtualization のバージョン 4.0 以降は非推奨となり、今後削除されます。

各要求は、HTTP Basic 認証を使用して [2] 認証情報をエンコードします。要求に適切な **Authorization** ヘッダーが含まれていない場合には、サーバーは **401 Authorization Required** 応答を送信します。

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

HTTP/1.1 401 Authorization Required
```

要求は、指定したレルムの **Authorization** ヘッダーで発行されます。 **username@domain:password** 規則を使用して、提供された認証情報で適切な Red Hat Virtualization Manager ドメインおよびユーザーをエンコードします。

下記の表には、Base64 認証情報をエンコードするためのプロセスをまとめています。

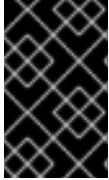
表2.2 API アクセスの認証情報のエンコーディング

項目	値
ユーザー名	admin
Domain	internal
Password	mypassword
エンコードされていない認証情報	admin@internal:mypassword
Base64 エンコードされた認証情報	YWRtaW5AaW50ZXJuYWw6bXlwYXNzd29yZA==

Base64 でエンコードされた認証情報を以下のように指定します。

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

HTTP/1.1 200 OK
```

**重要**

Basic 認証では、パスワードなどの機密情報がプレーンテキストで送信される可能性があります。API では、プレーンテキスト要求のトランスポートレベルの暗号化に Hypertext Transfer Protocol Secure (HTTPS) が必要です。

**重要**

Base64 ライブラリーによっては、結果を複数行に分割して、行末には改行文字をつけます。これによりヘッダーが破損し、要求に問題が発生します。**Authorization** ヘッダーには、エンコードされた認証情報がヘッダー内に一行で記載されている必要があります。

2.2.3. 認証セッション

API は認証セッションサポートを提供します。認証情報で最初の要求を送信し、次にセッションクッキーを使用して後続のすべての要求を送信し、認証を行います。

2.2.3.1. 認証されたセッションの要求

1. **Authorization** および **Prefer: persistent-auth** ヘッダーを使用して要求を送信します。

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

```
HTTP/1.1 200 OK
...
```

これは、以下のヘッダーで応答を返します。

```
Set-Cookie: JSESSIONID=5dQja5ubr4yvl2MM2z+LZxrK; Path=/ovirt-engine/api; Secure
```

JSESSIONID= の値を書き留めておきます。この例では、値は **5dQja5ubr4yvl2MM2z+LZxrK** です。

2. **JSESSIONID=** の値で **Prefer: persistent-auth** および **Cookie** ヘッダーで後続のすべての要求を送信します。認証セッションを使用する場合は、**Authorization** ヘッダーが不要になりました。

```
HEAD /ovirt-engine/api HTTP/1.1
Host: myengine.example.com
Prefer: persistent-auth
Cookie: JSESSIONID=5dQja5ubr4yvl2MM2z+LZxrK
```

```
HTTP/1.1 200 OK
...
```

3. セッションがなくなっただけの場合は、**Prefer: persistent-auth** ヘッダーなしでサーバーへの要求を実行します。

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

HTTP/1.1 200 OK

...

[1] HTTPS は、[RFC 2818 HTTP over TLS](#)で説明されています。

[2] Basic 認証については、[RFC 2617 HTTP Authentication: Basic and Digest Access Authentication](#)で説明されています。

第3章 一般的な概念

3.1. タイプ

API は **タイプ** の概念を使用して、受け入れおよび返されたさまざまな種類のオブジェクトを記述します。

関連するタイプには次の 3 種類があります。

プリミティブ型

文字列 や **整数** などの単純な種類のオブジェクトを記述します。

列挙型

VmStatus または **DiskFormat** などの有効な値の一覧を記述します。

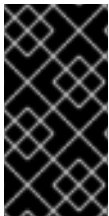
構造化型

Vm や **Disk** などの複数の属性とリンクを使用して、構造化されたオブジェクトを記述します。

3.2. 特定タイプ

API によって使用されるタイプの多くは、**identified** オブジェクトを表します。このオブジェクトは、一意の識別子が割り当てられており、他のオブジェクトとは独立して存在します。これらのオブジェクトの記述に使用されるタイプは、以下の共通属性のセットが含まれる **identified** タイプを拡張します。

属性	タイプ	Description
id	String	仮想化インフラストラクチャーの各オブジェクトには、一意の ID として機能する id が含まれます。
href	String	絶対パスとしてのオブジェクトの正規の場所。
name	String	ユーザーが指定する、人間が判読可能なオブジェクト名。 name 名は、同じタイプのすべてのオブジェクトの中で一意です。
description	String	ユーザーが指定する、人間が判読可能なオブジェクトの説明 (フリーフォーム)。



重要

現時点で、ほとんどの種類のオブジェクトでは、**id** 属性は実際には無作為に生成された **UUID** ですが、これは実装の情報であり、今後変更される可能性があるため、この情報に依存しないようにしてください。代わりにユーザーは、これらの識別子が文字列であると仮定する必要があります。

3.3. オブジェクト

オブジェクトは、API でサポートされるタイプの個別インスタンスです。たとえば、識別子が **123** の仮想マシンは、**Vm** タイプのオブジェクトです。

3.4. コレクション

コレクションは、同じタイプのオブジェクトのセットです。

3.5. 表現

オブジェクトの状態は、クライアントとサーバーを転送する時に表現する必要があります。API は、入力と出力の両方で、オブジェクトの状態を表す XML と JSON をサポートします。

3.5.1. XML 表現

オブジェクトの XML 表現は、オブジェクトのタイプに対応する XML 要素、**id** 属性と **href** 属性に対応する XML 属性、残りの属性に対応する XML 要素のネストで設定されます。たとえば、仮想マシンの XML 表現は以下のようになります。

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

オブジェクトのコレクションの XML 表現は、オブジェクトのタイプの名前が複数ある XML 要素で設定されます。これには、コレクションのオブジェクトの表現が含まれます。たとえば、仮想マシンのコレクションに対する XML 表現は以下のようになります。

```
<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>
```



重要

オブジェクトの XML 表現では、**id** および **href** 属性は XML 属性として表現される唯一の属性で、残りはネストされた XML 要素として表されます。

3.5.2. JSON 表現

オブジェクトの JSON 表現は、各属性の名前/値のペア (**id** および **href** など) を含む JSON ドキュメントで設定されます。たとえば、仮想マシンの JSON 表現は以下のようになります。

```
{
```

```

    "id": "123",
    "href": "/ovirt-engine/api/vms/123",
    "name": "myvm",
    "description": "My VM",
    "memory": 1073741824,
    ...
  }

```

オブジェクトコレクションの JSON 表現には、名前/値のペア (単数でオブジェクトのタイプの名前) などの JSON ドキュメントで設定され、その中にコレクションのオブジェクトの表現が指定されたアレイが含まれます。たとえば、仮想マシンのコレクションの JSON 表現は以下のようになります。

```

{
  "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. サービス

サービスは、API がサポートするオブジェクトで取得、更新、削除、および実行を行うサーバーの一部です。

関連するサービスには、以下の 2 つのタイプがあります。

オブジェクトのコレクションを管理するサービス

これらのサービスは、既存のオブジェクトの一覧を表示し、新規オブジェクトを追加できます。たとえば、**Vms** サービスは、システムで利用可能な仮想マシンのコレクションを管理します。

特定のオブジェクトを管理するサービス

これらのサービスは、特定のオブジェクトでアクションの取得、更新、および削除、および実行を行います。たとえば、**Vm** サービスは特定の仮想マシンを管理します。

各サービスは、サーバー内の特定のパスからアクセスできます。たとえば、システムで利用可能な仮想マシンのコレクションを管理するサービスは、パス **/vms** に、仮想マシン **123** を管理するサービスは、パス **/vms/123** にあります。

種類が何であってもサービスには、実行できる操作を表す **メソッド** のセットがあります。オブジェクトのコレクションを管理するサービスは通常、**list** と **add** メソッドをサポートします。特定のオブジェクトを管理するサービスには、通常、**get**、**update** メソッド、および **remove** メソッドがあります。さ

らに、サービスには、あまり一般的ではない操作を表す **action** メソッドも含まれることがあります。たとえば、**Vm** サービスには、仮想マシンの起動に使用される **start** メソッドがあります。

通常のメソッドでは、メソッドの名前と HTTP メソッドの名前の間で直接マッピングされます。

メソッド名	HTTP メソッド
add	POST
get	GET
list	GET
更新	PUT
remove	DELETE

HTTP 要求で使用されるパスは、接頭辞が **/ovirt-engine/api** のサービスのパスです。

たとえば、仮想マシンを一覧表示するリクエストは、HTTP **GET** メソッドとパス **/vms** を使用して、以下のように指定します。

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

アクションメソッドの場合には、HTTP メソッドは常に **POST** になり、メソッドの名前が接尾辞としてパスに追加されます。たとえば、仮想マシン **123** を起動する要求は、HTTP **POST** メソッドとパス **/vms/123/start** を使用して以下のように指定します。

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

メソッドごとにパラメーターセットがあります。

パラメーターは2つのカテゴリーに分類されます。

主なパラメーター

主要なパラメーターは、取得、追加、または更新するオブジェクトまたはコレクションに対応します。これは **add**、**get**、**list**、**update** メソッドにのみ適用され、1メソッドごとにこのような主要パラメーターは1つだけ存在します。

セカンダリーパラメーター

残りのパラメーター。

たとえば、仮想マシンを追加する操作 ([こちら](#) を参照) には、**vm**、**clone**、および **clone_permissions** の3つのパラメーターがあります。追加するオブジェクトを記述するため、主要パラメーターは **vm** です。**clone** および **clone_permissions** パラメーターはセカンダリーパラメーターです。

入力に使用する主要パラメーターは、HTTP 要求の本文に含める必要があります。たとえば、仮想マシンを追加する場合は、**Vm** タイプの **vm** パラメーターを要求本文に含める必要があります。つまり、すべての HTTP の情報が含まれる、仮想マシン追加の完全な要求は、以下のようになります。

```
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>
```

出力に使用すると、主要なパラメーターは応答ボディに含まれます。たとえば、仮想マシンを追加する場合、**vm** パラメーターは応答ボディに含まれます。そのため、完全な応答ボディは以下のようになります。

```
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>
```

セカンダリーパラメーターは(後に説明している action メソッドを除く)入力のみが可能で、クエリーパラメーターとして組み込む必要があります。たとえば、**clone** パラメーターが **true** に設定された仮想マシンを追加する場合に、完全な要求は以下のようになります。

```
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 メソッドにはセカンダリーパラメーターだけが含まれます。これらは入出力に使用でき、**action** 要素でラップされたリクエスト本文に含める必要があります。仮想マシンの起動に使用されるアクションメソッド([こちら](#)を参照)には、仮想マシンの起動方法を説明する **vm** パラメーターと、ゲストオペレーティングシステムの設定に **cloud-init** を使用するかどうかを指定する **use_cloud_init** パラメーターがあります。そのため、XML を使用する場合には、**cloud-init** を使用して仮想マシン **123** を起動する完全な要求は以下のようになります。

```
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. 検索

一部のサービスの **list** メソッドには、**検索** 条件を指定するために使用できる `search` パラメーターがあります。これを使用すると、サーバーはこれらの条件を満たすコレクション内のオブジェクトのみを返します。たとえば、以下の要求では **myvm** という名前の仮想マシンのみを返します。

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

3.7.1. 最大結果パラメーター

max パラメーターを使用して、返されるオブジェクトの数を制限します。たとえば、以下の要求は、システムで利用可能なものに関係なく、仮想マシンが1つだけ返されます。

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

max パラメーターのない検索要求は、すべてのオブジェクトを返します。システムの全体的なパフォーマンスにおける要求の影響を減らすために、**max** パラメーターを指定することが推奨されます。

3.7.2. ケースの機密性

デフォルトでは、クエリーは大文字と小文字を区別しません。たとえば、以下の要求は、**myvm**、**MyVM**、および **MYVM** という名前の仮想マシンを返します。

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

この動作を変更するには、任意の **case_sensitive** boolean パラメーターを使用できます。たとえば、**MyHost** または **MYHOST** ではなく **myhost** という名前の仮想マシンを取得するには、以下のよう
に要求を送信します。

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

3.7.3. 検索構文

検索 パラメーターは、Red Hat Virtualization クエリ言語と同じ形式を使用します。

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

sortby 句はオプションで、結果を順序付ける場合にのみ必要です。

検索クエリーの例

コレクション	基準	結果
hosts	vms.status=up	稼働 中の仮想マシンを実行しているすべてのホスト一覧を返します。
vms	domain=example.com	指定されたドメインで稼働しているすべての仮想マシンの一覧を返します。
vms	users.name=mary	ユーザー名が Mary のユーザーに属する全仮想マシンの一覧を返します。
events	severity > normal sortby time	重大度が 通常 よりも高いすべてのイベントの一覧を返します。 time 属性の値でソートされます。
events	severity > normal sortby time desc	重大度が 通常 よりも高いすべてのイベントの一覧を返します。 time 属性の値を降順で並び替えます。

search パラメーターの値は、演算子やスペースなどの予約済み文字を変換するために [URL エンコード](#) されている必要があります。たとえば、等号記号は **%3D** としてエンコードする必要があります。

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

3.7.4. ワイルドカード

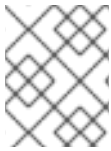
アスタリスクは値の一部として使用して、empty の文字列が含まれる文字列を検索できます。たとえば、以下の要求は、myvm、**myvm2**、**myvma**、または **myvm-webserver** などの **myvm** で始まる名前を持つすべての仮想マシンを返します。

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

3.7.5. ページネーション

一部の Red Hat Virtualization 環境には、大規模なオブジェクトが含まれます。一度のリクエストですべてを取得するのは現実的ではなく、パフォーマンスも悪くなります。ページ単位での検索を可能にするため、**search** パラメーターはオプションで **page** 句をサポートしています。これは、**max** パラメーターと組み合わせてページングの基礎となります。たとえば、ページサイズが 10 の仮想マシンの最初のページを取得するには、以下のように要求を送信します。

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



注記

search パラメーターは URL エンコードされており、エンコード前の実際の **search** パラメーターの値は **page 1** なので、実際には 1 ページ目を要求していることになります。

page 値を増やして、次のページを取得します。

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

page 句は、**search** パラメーター内の他の句と組み合わせて使用できます。たとえば、以下の要求は仮想マシンの 2 ページ目を返しますが、名前が並べ替えます。

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



重要

API はステートレスで、すべての要求が互いに独立しているため、異なる要求間で状態を維持することはできません。その結果、要求間でステータスが変更されると、ページの結果に一貫性がなくなることがあります。

たとえば、仮想マシンの一覧から特定のページを要求し、次のページを要求する前に仮想マシンが作成または削除された場合には、結果の一部が欠落しているか、または重複が含まれる可能性があります。

3.8. パーミッション

単一オブジェクトを管理するサービスの多くは、そのオブジェクトに割り当てられたパーミッションを管理する **permissions** サービスへの参照を提供します。各パーミッションには、ユーザーまたはグループ、ロール、およびオブジェクトへのリンクが含まれます。たとえば、特定の仮想マシンに割り当てられているパーミッションは、以下のような要求を送信して取得できます。

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

応答ボディは以下ようになります。

```
<permissions>
  <permission id="456" href="/ovirt-engien/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>
```

パーミッションは、このサービスへのパーミッション表現を含む **POST** 要求を送信するオブジェクトに追加されます。それぞれの新規パーミッションには、ロールとユーザーが必要です。

3.9. エラーの処理

エラーによっては、標準的な HTTP ステータスコード以外の詳しい説明が必要です。たとえば、API は応答ボディに **問題** があるオブジェクトの状態の更新またはアクションを報告します。障害には **reason** および **detail** 属性が含まれます。たとえば、サーバーが必須 **name** 属性なしで仮想マシンを作成する要求を受信すると、以下の HTTP 応答行が返されます。

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

そして、応答のボディは以下のとおりです。

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


第4章 クイックスタートの例

本章では、基本的な Red Hat Virtualization 環境を設定し、仮想マシンを作成する REST API の機能を示す例を紹介します。この例では、標準の前提条件に加えて、以下が必要です。

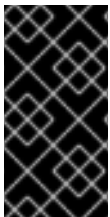
- ネットワークされ設定された Red Hat Virtualization インストール。
- インストールする仮想マシンオペレーティングシステムを含む ISO ファイル。本章では、インストール ISO の例に [CentOS 7](#) を使用します。
- 選択したオペレーティングシステムの ISO ファイルをアップロードする Red Hat Virtualization の **engine-iso-uploader** ツール。

この例では、**curl** を使用してクライアントアプリケーションでの API 要求を示しています。HTTP リクエストが可能なすべてのアプリケーションは **curl** を置き換えることができることに注意してください。



重要

分かりやすくするため、この例では **Host** および **Authorization** ヘッダーを省略します。ただし、これらのフィールドは必須であり、Red Hat Virtualization のインストールに固有のデータを必要とします。



重要

すべての **curl** の例では、ユーザー名に **admin@internal**、パスワードには **mypassword**、証明書の場所に **/etc/pki/ovirt-engine/ca.pem**、ホスト名に **myengine.example.com** を使用しています。これらは単なる例です。お使いの環境の有効な値に置き換えてください。



注記

Red Hat Virtualization は、各リソースの **id** 属性に一意的識別子を生成します。この例の識別子コードは、お使いの Red Hat Virtualization 環境の識別子コードとは異なる場合があります。



注記

本セクションの多くの例では、API によって返される結果の属性の一部が省略され、短くしています。参照を使用すると、常に属性の完全なリストを検索できます。たとえば、クラスタータイプの属性の完全なリストを表示する場合は、[ここ](#)に進んでください。

4.1. 例：API のエントリーポイントへのアクセス

以下の要求は、API のバージョン 4 のメインエントリーポイントの表現を取得します。

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

同じリクエストですが、**Version** ヘッダーの代わりに **/v4** URL 接頭辞を使用します。

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

curl コマンドを使用して、同じ要求を行います。

```
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
```

結果は、[Api](#) タイプのオブジェクトになります。

```
<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.el7</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>
```

重要

ヘッダーと URL 接頭辞が使用されていない場合には、サーバーは自動的にバージョンを選択します。デフォルトはバージョン 4 です。**ENGINE_API_DEFAULT_VERSION** 設定パラメーターを使用してデフォルトのバージョンを変更できます。

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

このパラメーターを変更すると、バージョンを明示的に指定しない API のすべてのユーザーに影響します。

エントリーポイントは、仮想化環境のコレクションへのリンクをユーザーに提供します。各コレクションリンクの **rel** 属性は、各リンクの参照ポイントを提供します。この例の次のステップは、データセンターコレクションを検証し、これは **datacenters** リンクから入手できます。

エントリーポイントには、[product_info](#)、[special_objects](#)、[summary](#) などの他のデータも含まれます。このデータは、この例以外の章で説明しています。

4.2. 例：データセンターの一覧表示

Red Hat Virtualization は、インストール時に **Default** のデータセンターを作成します。この例では、仮想環境のベースとして **Default** データセンターを使用します。

以下の要求は、データセンターの表現を取得します。

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

curl コマンドを使用して、同じ要求を行います。

```
# 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
```

結果として、[DataCenter](#) タイプのオブジェクト一覧が作成されます。

```
<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>

```

Default データ センターの ID をメモします。仮想環境の他のリソースに関連して、このデータセンターを特定します。

データセンターには、データセンターにアタッチされたストレージドメインを管理する [サービス](#) へのリンクも含まれています。

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

このサービスは、メインの **storagedomains** コレクションからストレージドメインを割り当てるために使用されます。この例では、この点について後ほど説明します。

4.3. 例：ホストクラスタの一覧表示

Red Hat Virtualization は、インストール時に **Default** ホストクラスタを作成します。この例では、**Default** クラスタを使用して Red Hat Virtualization 環境のリソースをグループ化します。

以下の要求は、クラスタコレクションの表現を取得します。

```

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

```

curl コマンドを使用して、同じ要求を行います。

```

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

```

結果として、[Cluster](#) タイプのオブジェクトの一覧が表示されます。

```

<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"/>
    ...
  <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>

```

Default ホストのクラスターの **ID** をメモします。仮想環境の他のリソースと照合して、このホストクラスターを特定します。

Default クラスターは、**data_center** リンクの **id** および **href** 属性を使用する関係により **Default** データセンターに関連付けられます。

```

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

```

networks リンクは、このクラスターに関連付けられたネットワークを管理する [サービス](#) への参照です。次のセクションでは、ネットワークコレクションの詳細を説明します。

4.4. 例：論理ネットワークの一覧表示

Red Hat Virtualization は、インストール時にデフォルトの **ovirtmgmt** ネットワークを作成します。このネットワークは、Red Hat Virtualization Manager がホストにアクセスするための管理ネットワークとして機能します。

このネットワークは **Default** クラスターに関連付けられており、**Default** データセンターのメンバーです。この例では、**ovirtmgmt** ネットワークを使用して仮想マシンに接続します。

以下の要求は、論理ネットワークの一覧を取得します。

```

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

```

curl コマンドを使用して、同じ要求を行います。

```

# 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

```

結果として、**Network** タイプのオブジェクトの一覧が表示されます。

```

<networks>
  <network href="/ovirt-engine/api/networks/003" id="003">
    <name>ovirtmgmt</name>
    <description>Management Network</description>
    <link href="/ovirt-engine/api/networks/003/permissions" rel="permissions"/>
    <link href="/ovirt-engine/api/networks/003/vnicprofiles" rel="vnicprofiles"/>
  </network>
</networks>

```

```

<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>

```

ovirtmgmt ネットワークは、データセンターの **ID** を使用した関係により **Default** データセンターに割り当てられます。

ovirtmgmt ネットワークは、クラスターのネットワークサブコレクションの関係を介して **Default** クラスターにも割り当てられます。

4.5. 例：ホストの一覧表示

以下の例では、ホストの一覧を取得し、仮想化環境に登録されている **myhost** という名前のホストを表示します。

```

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

```

curl コマンドを使用して、同じ要求を行います。

```

# 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

```

結果として、**Host** タイプのオブジェクトの一覧が表示されます。

```

<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>
      <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>

```

ホストの **ID** をメモします。仮想環境の他のリソースに関連して、このホストを特定します。

このホストは **Default** クラスターのメンバーで、**nics** サブコレクションにアクセスすると、このホストが **ovirtmgmt** ネットワークに接続されていることが分かります。

4.6. 例 : NFS データストレージの作成

NFS データストレージドメインは、データセンターに接続されたエクスポートされた NFS 共有であり、仮想化ゲストイメージのストレージを提供します。新規ストレージドメインを作成するには、ストレージドメイン表現が含まれる **POST** 要求をストレージドメインコレクションの URL に送信する必要があります。

ストレージドメインでは、デフォルトで削除後にワイプオプションを有効にできます。これを設定するには、POST 要求で **wipe_after_delete** を指定します。このオプションは、ドメインの作成後に編集することが可能ですが、その場合にはすでに存在していたディスクの削除後にワイププロパティは変更されません。

要求は以下のようになります。

```

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

```

また、リクエスト本文は以下のようになります。

```

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

```

curl コマンドを使用した同じ要求。

```
# 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

```

サーバーはホスト **myhost** を使用して、**my nfs.example.com:/exports/mydata** のエクスポートパスで **mydata** という名前の NFS データストレージドメインを作成します。API は ([StorageDomain](#) タイプ) の新規作成されたストレージドメインリソースについて以下の表現も返します。

```

<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. 例 : NFS ISO ストレージの作成

NFS ISO ストレージドメインは、データセンターにアタッチされ、マウントされた NFS 共有で、DVD/CD-ROM ISO および仮想フロッピーディスク (VFD) イメージファイルのストレージを提供します。新規ストレージドメインを作成するには、ストレージドメイン表現が含まれる **POST** 要求をストレージドメインコレクションの URL に送信する必要があります。

要求は以下ようになります。


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

また、リクエスト本文は以下のようになります。

```
<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>
```

curl コマンドを使用した同じ要求。

```
# 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
```

サーバーは、ホスト **myhost** を使用して、エクスポートパスが **my nfs.example.com:/exports/myisos** で、**myisos** という名前の NFS ISO ストレージドメインを作成します。API は ([StorageDomain](#) タイプの) 新規作成されたストレージドメインリソースについて以下の表現も返します。

```
<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. 例：ストレージドメインのデータセンターへの割り当て

以下の例では、**mydata** および **myisos** ストレージドメインを **Default** データセンターにアタッチします。

mydata ストレージドメインをアタッチするには、以下のように要求を送信します。

```

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

```

リクエスト本文は以下のようになります。

```

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

```

curl コマンドを使用して、同じ要求を行います。

```

# 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

```

myisos ストレージドメインをアタッチするには、以下のような要求を送信します。

```

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

```

リクエスト本文は以下のようになります。

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

`curl` コマンドを使用して、同じ要求を行います。

```
# 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. 例：仮想マシンの作成

以下の例では、仮想化環境の **Blank** テンプレートをベースとして使用し、**Default** クラスタに **myvm** という名前の仮想マシンを作成します。また、要求は仮想マシンのメモリーを 512 MiB として定義し、ブートデバイスを仮想ハードディスクに設定します。

要求には、作成する仮想マシンを記述する `Vm` タイプのオブジェクトが含まれている必要があります。

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

また、リクエスト本文は以下のようになります。

```
<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>
```

`curl` コマンドを使用して、同じ要求を行います。

```
# 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
```

応答ポディーは、[Vm](#) タイプのオブジェクトです。

```
<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. 例：仮想マシン NIC の作成

以下の例では、サンプルの仮想マシンを **ovirtmgmt** ネットワークに接続するための仮想ネットワークインターフェイスを作成します。

要求は以下のようになります。

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

リクエスト本文には、作成する NIC を記述する **Nic** タイプのオブジェクトが含まれている必要があります。

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

curl コマンドを使用して、同じ要求を行います。

```
# 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. 例：仮想マシンディスクの作成

以下の例では、サンプルの仮想マシン用に 8 GiB の **コピーオンライト** ディスクを作成します。

要求は以下のようになります。

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

リクエスト本文は **DiskAttachment** 型のオブジェクトで、ディスクとその仮想マシンへのアタッチ方法を記述してください。

```

<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>

```

curl コマンドを使用して、同じ要求を行います。

```

# 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

```

storage_domains 属性は、API に対して、ディスクを **mydata** ストレージドメインに保存するように指示します。

4.12. 例 : ISO イメージの仮想マシンへの割り当て

この例の仮想マシンのブートメディアには、オペレーティングシステムのインストールに CD-ROM または DVD ISO イメージが必要です。この例では、インストールに CentOS 7 イメージを使用します。

仮想マシンが使用するには、**myisos** ISO ドメインで ISO イメージが利用可能でなければなりません。Red Hat Virtualization は、正しいユーザーパーミッションで ISO イメージが適切なディレクトリーパスにアップロードされるように uploader ツールを提供します。

ISO をアップロードしたら、API を使用して ISO ストレージドメインからファイルの一覧を要求できません。

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

curl コマンドを使用して、同じ要求を行います。

```
# 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
```

サーバーは、利用可能な ISO(または floppy) イメージごとに1つずつ、**File** タイプのオブジェクト一覧を返します。

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

API ユーザーは **CentOS-7-x86_64-Minimal.iso** をサンプル仮想マシンに割り当てます。ISO イメージの割り当ては、管理またはユーザーポータルアプリケーションで **CD** ボタンを使用するのと同じです。

要求は以下のようになります。

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

リクエスト本文は、ISO(または floppy) イメージの識別子を示す内部 **ファイル** 属性が含まれる **Cdrom** タイプのオブジェクトである必要があります。

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

curl コマンドを使用して、同じ要求を行います。

```
# 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
```

詳細は、仮想マシンの CD-ROMS を管理する [サービス](#) のドキュメントを参照してください。

4.13. 例：仮想マシンの起動

仮想環境が完了し、仮想マシンには正常に動作させるのに必要なすべてのコンポーネントが含まれます。この例では、`start` メソッドを使用して仮想マシンを起動します。

要求は以下のようになります。

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

リクエスト本文は以下のようになります。

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

`curl` コマンドを使用して、同じ要求を行います。

```
# 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
```

追加のリクエスト本文は、今回の起動のみ仮想マシンのブートデバイスを CD-ROM に設定します。これにより、仮想マシンが割り当てられた ISO イメージからオペレーティングシステムをインストールできます。ブートデバイスは、その後の起動時にディスクに戻ります。

第5章 要求

本セクションでは、API で利用可能な全要求を列挙します。

- [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/activ
- POST
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/migr
- POST
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/stop
- GET
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{bric
- DELETE
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{bric
- POST
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{bric
- GET
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{bric
- GET
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{bric
- 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}
- 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/{lab
- DELETE
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/networklabels/{lab
- 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/{perr
- DELETE
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/permissions/{perr
- 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/{profil
- DELETE
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profil
- POST
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profil
- GET
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profil
- GET
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profil

- DELETE
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profil
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections
- GET
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections/{storageconne
- PUT
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections/{storageconne
- DELETE
/datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections/{storageconne
- 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 /datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks
- GET /datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks
- GET /datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}
- DELETE /datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}
- POST /datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/copy
- POST /datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/export
- POST /datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/move
- POST
/datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions
- GET
/datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions
- GET
/datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permission:id}
- DELETE
/Datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permission:id}

- POST
/datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/sparsify
- GET /datacenter/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/statistics
- GET
/datacenter/{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}/ディスク
- 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}](#)

第6章 サービス

本セクションでは、API で利用可能なすべてのサービスを説明します。

6.1. AFFINITYGROUP

このサービスは単一のアフィニティグループを管理します。

表6.1 メソッドの概要

Name	概要
get	アフィニティグループの詳細を取得します。
remove	アフィニティグループを削除します。
update	アフィニティグループを更新します。

6.1.1. get GET

アフィニティグループの詳細を取得します。

```
<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>
```

表6.2 パラメーターの概要

Name	タイプ	方向	概要
group	AffinityGroup	Out	アフィニティグループ。

6.1.2. remove DELETE

アフィニティグループを削除します。

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

表6.3 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.1.3. update PUT

アフィニティーグループを更新します。

表6.4 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
group	AffinityGroup	In/Out	アフィニティーグループ。

6.2. AFFINITYGROUPVPM

このサービスは単一の仮想マシンを管理し、アフィニティーグループの割り当てを管理します。

表6.5 メソッドの概要

Name	概要
remove	この仮想マシンをアフィニティーグループから削除します。

6.2.1. remove DELETE

この仮想マシンをアフィニティーグループから削除します。

表6.6 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.3. AFFINITYGROUPVMS

このサービスは、アフィニティーグループに割り当てられたすべての仮想マシンのコレクションを管理します。

表6.7 メソッドの概要

Name	概要
add	仮想マシンをアフィニティーグループに追加します。
list	このアフィニティーグループに割り当てられたすべての仮想マシンを一覧表示します。

6.3.1. add POST

仮想マシンをアフィニティグループに追加します。

たとえば、仮想マシン 000-000 をアフィニティグループ 123-456 に追加するには、以下に要求を送信します。

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

以下のボディを使用します。

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

表6.8 パラメーターの概要

Name	タイプ	方向	概要
vm	Vm	In/Out	

6.3.2. list GET

このアフィニティグループに割り当てられたすべての仮想マシンを一覧表示します。

表6.9 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す仮想マシンの最大数を設定します。
vms	Vm[]	Out	

6.3.2.1. max

返す仮想マシンの最大数を設定します。指定されていない場合には、すべての仮想マシンが返されません。

6.4. AFFINITYGROUPS

アフィニティグループサービスは、仮想マシンの関係と依存関係を管理します。

表6.10 メソッドの概要

Name	概要
add	新しいアフィニティグループを作成します。
list	既存のアフィニティグループを一覧表示します。

6.4.1. add POST

新しいアフィニティグループを作成します。

以下の例のような Pos 要求を送信して、新しいアフィニティーグループを作成します。

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

そして、以下の例をその本文で使用します。

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

表6.11 パラメーターの概要

Name	タイプ	方向	概要
group	AffinityGroup	In/Out	作成するアフィニティーグループオブジェクト。

6.4.2. list GET

既存のアフィニティーグループを一覧表示します。

表6.12 パラメーターの概要

Name	タイプ	方向	概要
groups	AffinityGroup[]	Out	既存のアフィニティーグループの一覧。
max	Integer	In	返すアフィニティーグループの最大数を設定します。

6.4.2.1. max

返すアフィニティーグループの最大数を設定します。指定されていない場合は、すべてのアフィニティーグループが返されます。

6.5. AFFINITYLABEL

単一のアフィニティーラベルの詳細。

表6.13 メソッドの概要

Name	概要
get	ラベルの詳細を取得します。
remove	システムからラベルを削除し、削除されたラベルの割り当てを消去します。

Name	概要
update	ラベルを更新します。

6.5.1. get GET

ラベルの詳細を取得します。

表6.14 パラメーターの概要

Name	タイプ	方向	概要
ラベル	AffinityLabel	Out	

6.5.2. remove DELETE

システムからラベルを削除し、削除されたラベルの割り当てを消去します。

6.5.3. update PUT

ラベルを更新します。この呼び出しは、名前や説明などのメタデータをすべて更新します。

表6.15 パラメーターの概要

Name	タイプ	方向	概要
ラベル	AffinityLabel	In/Out	

6.6. AFFINITYLABELHOST

このサービスは、affinitylabels/hosts サブコレクションでアクセスする際に特定のラベルを持つホストを表します。

表6.16 メソッドの概要

Name	概要
get	このラベルが割り当てられたホストの詳細を取得します。
remove	ホストからラベルを削除します。

6.6.1. get GET

このラベルが割り当てられたホストの詳細を取得します。

表6.17 パラメーターの概要

Name	タイプ	方向	概要
host	ホスト	Out	

6.6.2. remove DELETE

ホストからラベルを削除します。

6.7. AFFINITYLABELHOSTS

このサービスは、affinitylabels/hosts サブコレクションでアクセスする際に特定のラベルを持つホストの一覧を表します。

表6.18 メソッドの概要

Name	概要
add	Add a label to a host.
list	ラベルの付いたホストをすべて一覧表示します。

6.7.1. add POST

Add a label to a host.

表6.19 パラメーターの概要

Name	タイプ	方向	概要
host	ホスト	In/Out	

6.7.2. list GET

ラベルの付いたホストをすべて一覧表示します。

表6.20 パラメーターの概要

Name	タイプ	方向	概要
hosts	Host[]	Out	

6.8. AFFINITYLABELVM

このサービスは、affinitylabels/vms サブコレクションでアクセスする際に特定のラベルを持つ仮想マシンを表します。

表6.21 メソッドの概要

Name	概要
get	このラベルが割り当てられた仮想マシンの詳細を取得します。
remove	仮想マシンからラベルを削除します。

6.8.1. get GET

このラベルが割り当てられた仮想マシンの詳細を取得します。

表6.22 パラメーターの概要

Name	タイプ	方向	概要
vm	Vm	Out	

6.8.2. remove DELETE

仮想マシンからラベルを削除します。

6.9. AFFINITYLABELVMS

このサービスは、affinitylabels/vms サブコレクションでアクセスする際に特定のラベルを持つ仮想マシンの一覧を表します。

表6.23 メソッドの概要

Name	概要
add	仮想マシンにラベルを追加します。
list	ラベルの付いたすべての vms を一覧表示します。

6.9.1. add POST

仮想マシンにラベルを追加します。

表6.24 パラメーターの概要

Name	タイプ	方向	概要
vm	Vm	In/Out	

6.9.2. list GET

ラベルの付いたすべての vms を一覧表示します。

表6.25 パラメーターの概要

Name	タイプ	方向	概要
vms	Vm[]	Out	

6.10. AFFINITYLABELS

システムで利用可能なアフィニティーラベルを管理します。

表6.26 メソッドの概要

Name	概要
add	新しいラベルを作成します。
list	システムに存在するすべてのラベルを一覧表示します。

6.10.1. add POST

新しいラベルを作成します。ラベルは、vms または hosts 一覧にあるすべてのエンティティーに自動的に割り当てられます。

表6.27 パラメーターの概要

Name	タイプ	方向	概要
ラベル	AffinityLabel	In/Out	

6.10.2. list GET

システムに存在するすべてのラベルを一覧表示します。

表6.28 パラメーターの概要

Name	タイプ	方向	概要
labels	AffinityLabel []	Out	
max	Integer	In	返すラベルの最大数を設定します。

6.10.2.1. max

返すラベルの最大数を設定します。指定されていない場合、すべてのラベルが返されます。

6.11. ASSIGNEDAFFINITYLABEL

このサービスは、Entity/affinitylabels サブコレクションを使用してアクセスすると、エンティティー割り当てに対する1つのラベルを表します。

表6.29 メソッドの概要

Name	概要
get	割り当てられたラベルの詳細を取得します。
remove	エンティティーからラベルを削除します。

6.11.1. get GET

割り当てられたラベルの詳細を取得します。

表6.30 パラメーターの概要

Name	タイプ	方向	概要
ラベル	AffinityLabel	Out	

6.11.2. remove DELETE

エンティティーからラベルを削除します。ラベル自体には触れません。

6.12. ASSIGNEDAFFINITYLABELS

このサービスは、Entity/affinitylabels を使用してアクセスする際にサポートされているエンティティーに割り当てられるアフィニティーラベルを一覧表示し、操作するために使用されます。

表6.31 メソッドの概要

Name	概要
add	ラベルをエンティティーに割り当てます。
list	エンティティーに割り当てられたすべてのラベルを一覧表示します。

6.12.1. add POST

ラベルをエンティティーに割り当てます。

表6.32 パラメーターの概要

Name	タイプ	方向	概要
ラベル	AffinityLabel	In/Out	

6.12.2. list GET

エンティティーに割り当てられたすべてのラベルを一覧表示します。

表6.33 パラメーターの概要

Name	タイプ	方向	概要
label	AffinityLabel []	Out	

6.13. ASSIGNEDCPUPROFILE

表6.34 メソッドの概要

Name	概要
get	
remove	

6.13.1. get GET

表6.35 パラメーターの概要

Name	タイプ	方向	概要
profile	CpuProfile	Out	

6.13.2. remove DELETE

表6.36 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.14. ASSIGNEDCPUPROFILES

表6.37 メソッドの概要

Name	概要
add	
list	

6.14.1. add POST

表6.38 パラメーターの概要

Name	タイプ	方向	概要
profile	CpuProfile	In/Out	

6.14.2. list GET

表6.39 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロファイルの最大数を設定します。
profiles	CpuProfile[]	Out	

6.14.2.1. max

返すプロファイルの最大数を設定します。指定されていない場合は、すべてのプロファイルが返されます。

6.15. ASSIGNEDDISKPROFILE

表6.40 メソッドの概要

Name	概要
get	
remove	

6.15.1. get GET

表6.41 パラメーターの概要

Name	タイプ	方向	概要
disk_profile	DiskProfile	Out	

6.15.2. remove DELETE

表6.42 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.16. ASSIGNEDDISKPROFILES

表6.43 メソッドの概要

Name	概要
add	
list	

6.16.1. add POST

表6.44 パラメーターの概要

Name	タイプ	方向	概要
profile	DiskProfile	In/Out	

6.16.2. list GET

表6.45 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロファイルの最大数を設定します。
profiles	DiskProfile[]	Out	

6.16.2.1. max

返すプロファイルの最大数を設定します。指定されていない場合は、すべてのプロファイルが返されます。

6.17. ASSIGNEDPERMISSIONS

ユーザー、グループ、またはエンティティタイプ別にスコープ指定されたパーミッションサブコレクションを表します。

表6.46 メソッドの概要

Name	概要
add	特定のエンティティのユーザーまたはグループに新しいパーミッションを割り当てます。
list	特定のエンティティのすべてのパーミッションを一覧表示します。

6.17.1. add POST

特定のエンティティのユーザーまたはグループに新しいパーミッションを割り当てます。

たとえば、**UserVmManager** ロールを ID が **123** の仮想マシン、id が **456** のユーザーに割り当てるには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

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

id が **456** のユーザーに **SuperUser** ロールを割り当てるには、以下のように要求を送信します。

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

リクエスト本文は以下のようになります。

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

ユーザーの代わりにグループにパーミッションを割り当てる場合は、**user** 要素を **group** の適切な **ID** に置き換えます。たとえば、**UserRole** ロールを ID が **123** のクラスター、ID が **789** のグループに割り当てるには、以下のような要求を送信します。

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

リクエスト本文は以下のようになります。

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

表6.47 パラメーターの概要

Name	タイプ	方向	概要
permission	パーミ ッション	In/Out	パーミッション。

6.17.2. list GET

特定のエンティティのすべてのパーミッションを一覧表示します。

たとえば、id **123** のクラスターのすべての権限を一覧表示するには、以下のように要求を送信します。

```
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>
```

表6.48 パラメーターの概要

Name	タイプ	方向	概要
permissions	Permission[]	Out	パーミッションのリスト

6.18. ASSIGNEDROLES

ロールサブコレクション (例: ユーザーごとにスコープ設定) を表します。

表6.49 メソッドの概要

Name	概要
list	

6.18.1. list GET

表6.50 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すロールの最大数を設定します。
roles	Role[]	Out	

6.18.1.1. max

返すロールの最大数を設定します。指定のない場合は、すべてのロールが返されます。

6.19. ASSIGNEDTAG

システムの特定制エンティティへの特定のタグの割り当てを管理するサービス

表6.51 メソッドの概要

Name	概要
get	割り当てられたタグに関する情報を取得します。
remove	システムの特定制のエンティティからタグの割り当てを解除します。

6.19.1. get GET

割り当てられたタグに関する情報を取得します。

たとえば、id **123** の仮想マシンに割り当てられる id **456** のタグに関する情報を取得するには、以下のよう
に要求を送信します。

```
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>
```

表6.52 パラメーターの概要

Name	タイプ	方向	概要
tag	タグ	Out	割り当てられたタグ。

6.19.2. remove DELETE

システムの特定制のエンティティからタグの割り当てを解除します。

たとえば、id **123** の仮想マシンから id **456** のタグの割り当てを解除するには、以下のよう
に要求を送信
します。

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

表6.53 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.20. ASSIGNEDTAGS

システムの特定のエンティティへのタグの割り当てを管理するサービス。

表6.54 メソッドの概要

Name	概要
add	システムの特定のエンティティにタグを割り当てます。
list	特定のエンティティに割り当てられたすべてのタグを一覧表示します。

6.20.1. add POST

システムの特定のエンティティにタグを割り当てます。

たとえば、ID が **123** の仮想マシンにタグ **mytag** を割り当てるには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

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

表6.55 パラメーターの概要

Name	タイプ	方向	概要
tag	タグ	In/Out	割り当てられたタグ。

6.20.2. list GET

特定のエンティティに割り当てられたすべてのタグを一覧表示します。

たとえば、id **123** の仮想マシンのタグの一覧を表示するには、以下のようリクエストを送信します。

```
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>
```

表6.56 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すタグの最大数を設定します。
tags	Tag[]	Out	割り当てられたタグの一覧。

6.20.2.1. max

返すタグの最大数を設定します。指定されていない場合は、すべてのタグが返されます。

6.21. ASSIGNEDVNICPROFILE

表6.57 メソッドの概要

Name	概要
get	
remove	

6.21.1. get GET

表6.58 パラメーターの概要

Name	タイプ	方向	概要
profile	VnicProfile	Out	

6.21.2. remove DELETE

表6.59 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.22. ASSIGNEDVNICPROFILES

表6.60 メソッドの概要

Name	概要
add	
list	

6.22.1. add POST

表6.61 パラメーターの概要

Name	タイプ	方向	概要
profile	VnicProfile	In/Out	

6.22.2. list GET

表6.62 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロファイルの最大数を設定します。
profiles	VnicProfile[]	Out	

6.22.2.1. max

返すプロファイルの最大数を設定します。指定されていない場合は、すべてのプロファイルが返されます。

6.23. ATTACHEDSTORAGEDOMAIN

表6.63 メソッドの概要

Name	概要
activate	この操作により、割り当てられたストレージドメインがアクティベートされます。
deactivate	この操作により、接続されたストレージドメインが非アクティブになります。
get	
remove	

6.23.1. activate POST

この操作により、割り当てられたストレージドメインがアクティベートされます。ストレージドメインがアクティブ化されると、データセンターで使用できるようになります。

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

アクティブ化アクションはアクション固有のパラメーターを実行しないため、リクエストの本文には空の **action** が含まれている必要があります。

```
<action/>
```


表6.64 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクティベーションを非同期で実行する必要があるかどうかを示します。

6.23.2. deactivate POST

この操作により、接続されたストレージドメインが非アクティブになります。ストレージドメインが非アクティブ化されると、データセンターでは使用されなくなります。

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

deactivate アクションはアクション固有のパラメーターを取りません。したがって、リクエスト本文には空の **action** が含まれている必要があります。

<action/>

表6.65 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	非アクティブ化を非同期で実行する必要があるかどうかを示します。

6.23.3. get GET

表6.66 パラメーターの概要

Name	タイプ	方向	概要
storage_domain	StorageDomain	Out	

6.23.4. remove DELETE

表6.67 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.24. ATTACHEDSTORAGEDOMAINS

表6.68 メソッドの概要

Name	概要
add	
list	

6.24.1. add POST

表6.69 パラメーターの概要

Name	タイプ	方向	概要
storage_domain	StorageDomain	In/Out	

6.24.2. list GET

表6.70 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すストレージドメインの最大数を設定します。
storage_domains	StorageDomain[]	Out	

6.24.2.1. max

返すストレージドメインの最大数を設定します。指定されていない場合は、すべてのストレージドメインが返されます。

6.25. バランス

表6.71 メソッドの概要

Name	概要
get	
remove	

6.25.1. get GET

表6.72 パラメーターの概要

Name	タイプ	方向	概要
balance	バランス	Out	
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

6.25.2. remove DELETE

表6.73 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.26. バランス

表6.74 メソッドの概要

Name	概要
add	
list	

6.26.1. add POST

表6.75 パラメーターの概要

Name	タイプ	方向	概要
balance	バランス	In/Out	

6.26.2. list GET

表6.76 パラメーターの概要

Name	タイプ	方向	概要
balances	Balance[]	Out	
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返すバランスの最大数を設定します。

6.26.2.1. max

返すバランスの最大数を設定します。指定されていない場合は、すべてのバランスが返されます。

6.27. ブックマーク

ブックマークを管理するサービス

表6.77 メソッドの概要

Name	概要
get	ブックマークを取得します。
remove	ブックマークを削除します。
update	ブックマークを更新します。

6.27.1. get GET

ブックマークを取得します。

ブックマークを取得する例:

```
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>
```

表6.78 パラメーターの概要

Name	タイプ	方向	概要
bookmark	ブックマーク	Out	要求されたブックマーク。

6.27.2. remove DELETE

ブックマークを削除します。

ブックマークを削除する例:

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

表6.79 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.27.3. update PUT

ブックマークを更新します。

ブックマークを更新する例:

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

リクエスト本文:

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

表6.80 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
bookmark	ブックマーク	In/Out	更新されたブックマーク。

6.28. ブックマーク

ブックマークを管理するサービス

表6.81 メソッドの概要

Name	概要
add	新規ブックマークの追加
list	利用可能なブックマークをすべて表示します。

6.28.1. add POST

新規ブックマークの追加

ブックマークの追加例:

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

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

表6.82 パラメーターの概要

Name	タイプ	方向	概要
bookmark	ブックマーク	In/Out	追加されたブックマーク。

6.28.2. list GET

利用可能なブックマークをすべて表示します。

ブックマークの一覧表示例:

```
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>
```

表6.83 パラメーターの概要

Name	タイプ	方向	概要
bookmarks	bookmark[]	Out	利用可能なブックマークの一覧。
max	Integer	In	返すブックマークの最大数を設定します。

6.28.2.1. max

返すブックマークの最大数を設定します。指定されていない場合は、すべてのブックマークが返されます。

6.29. CLUSTER

特定のクラスターを管理するサービス。

表6.84 メソッドの概要

Name	概要
get	クラスターに関する情報を取得します。
remove	システムからクラスターを削除します。
resetemulatedmachine	
update	クラスターに関する情報を更新します。

6.29.1. get GET

クラスターに関する情報を取得します。

クラスターを取得する例:

```
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>

```

表6.85 パラメーターの概要

Name	タイプ	方向	概要
cluster	Cluster	Out	
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

6.29.2. remove DELETE

システムからクラスターを削除します。

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

表6.86 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.29.3. resetemulatedmachine POST

表6.87 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	リセットを非同期で実行する必要があるかどうかを示します。

6.29.4. update PUT

クラスターに関する情報を更新します。

指定されたフィールドのみが更新されます。その他は変更されません。

たとえば、クラスターの CPU を更新します。

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

リクエスト本文は以下のようになります。

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

表6.88 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
cluster	Cluster	In/Out	

6.30. CLUSTERLEVEL

特定のクラスターレベルに関する情報を提供します。詳細は、[ClusterLevels](#) サービスを参照してください。

表6.89 メソッドの概要

Name	概要
get	このサービスが管理する特定のクラスターレベルの機能に関する情報を提供します。

6.30.1. get GET

このサービスが管理する特定のクラスターレベルの機能に関する情報を提供します。

たとえば、レベル 3.6 でサポートされる CPU タイプを確認するには、以下のように要求を送信できます。

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

これにより、サポートされる CPU タイプとクラスターレベルを記述する他の情報が含まれる `ClusterLevel` オブジェクトが返されます。

```
<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>
```

表6.90 パラメーターの概要

Name	タイプ	方向	概要
level	<code>ClusterLevel</code>	Out	取得したクラスターレベル。

6.31. CLUSTERLEVELS

エンジンがサポートするさまざまなクラスターレベルの機能に関する情報を提供します。エンジンのバージョン 4.0 は、レベル 4.0 および 3.6 をサポートします。これらのレベルのそれぞれは、さまざまな CPU タイプのセットをサポートします。以下に例を示します。このサービスは、その情報を提供します。

表6.91 メソッドの概要

Name	概要
list	システムがサポートするクラスターレベルを一覧表示します。

6.31.1. list GET

システムがサポートするクラスターレベルを一覧表示します。

GET /ovirt-engine/api/clusterlevels

これにより、利用可能なクラスターレベルのリストが返されます。

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

表6.92 パラメーターの概要

Name	タイプ	方向	概要
levels	ClusterLevel []	Out	取得されるクラスターレベル。

6.32. CLUSTERNETWORK

特定のクラスターネットワークを管理するサービス

表6.93 メソッドの概要

Name	概要
get	クラスターネットワークの詳細を取得します。
remove	クラスターからネットワークの割り当てを解除します。
update	クラスターのネットワークを更新します。

6.32.1. get GET

クラスターネットワークの詳細を取得します。

表6.94 パラメーターの概要

Name	タイプ	方向	概要
network	Network	Out	クラスターネットワーク

6.32.2. remove DELETE

クラスターからネットワークの割り当てを解除します。

6.32.3. update PUT

クラスターのネットワークを更新します。

表6.95 パラメーターの概要

Name	タイプ	方向	概要
network	Network	In/Out	クラスターネットワーク

6.33. CLUSTER NETWORKS

クラスターネットワークを管理するサービス

表6.96 メソッドの概要

Name	概要
add	ネットワークをクラスターに割り当てます。
list	クラスターに割り当てられているネットワークを一覧表示します。

6.33.1. add POST

ネットワークをクラスターに割り当てます。

以下の例のような Post 要求を送信して、ネットワークをクラスターに割り当てます。

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

ボディで以下の例を使用します。

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

表6.97 パラメーターの概要

Name	タイプ	方向	概要
------	-----	----	----

Name	タイプ	方向	概要
network	Network	In/Out	クラスターに割り当てるネットワークオブジェクト。

6.33.2. list GET

クラスターに割り当てられているネットワークを一覧表示します。

表6.98 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すネットワークの最大数を設定します。
networks	Network[]	Out	クラスターに割り当てられているネットワークの一覧。

6.33.2.1. max

返すネットワークの最大数を設定します。指定されていない場合には、すべてのネットワークが返されます。

6.34. CLUSTERS

クラスターを管理するサービス

表6.99 メソッドの概要

Name	概要
add	新規クラスターを作成します。
list	

6.34.1. add POST

新規クラスターを作成します。

これには、**name** 属性、**cpu.type** 属性、および **data_center** 属性が必要です。**id** または **name** 属性のいずれかでデータセンターを特定します。

POST /ovirt-engine/api/clusters

リクエスト本文は以下のようになります。

```
<cluster>
  <name>mycluster</name>
  <cpu>
```

```

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

```

表6.100 パラメーターの概要

Name	タイプ	方向	概要
cluster	Cluster	In/Out	

6.34.2. list GET

表6.101 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
clusters	Cluster[]	Out	
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返すことのできるクラスタの最大数。
search	String	In	返されたクラスタを制限するために使用されるクエリー文字列。

6.34.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.34.2.2. max

返すことのできるクラスタの最大数。指定されていない場合は、すべてのクラスタが返されます。

6.35. コピー可能

表6.102 メソッドの概要

Name	概要
copy	

6.35.1. copy POST

表6.103 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	コピーを非同期的に実行するかどうかを指定します。

6.36. CPUPROFILE

表6.104 メソッドの概要

Name	概要
get	
remove	
update	

6.36.1. get GET

表6.105 パラメーターの概要

Name	タイプ	方向	概要
profile	CpuProfile	Out	

6.36.2. remove DELETE

表6.106 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.36.3. update PUT

表6.107 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
profile	CpuProfile	In/Out	

6.37. CPUPROFILES

表6.108 メソッドの概要

Name	概要
add	
list	

6.37.1. add POST

表6.109 パラメーターの概要

Name	タイプ	方向	概要
profile	CpuProfile	In/Out	

6.37.2. list GET

表6.110 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロファイルの最大数を設定します。
profile	CpuProfile[]	Out	

6.37.2.1. max

返すプロファイルの最大数を設定します。指定されていない場合は、すべてのプロファイルが返されます。

6.38. DATACENTER

データセンターを管理するサービス

表6.111 メソッドの概要

Name	概要
get	データセンターを取得します。
remove	データセンターを削除します。
update	データセンターを更新します。

6.38.1. get GET

データセンターを取得します。

データセンターを取得する例:

```
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>
```

表6.112 パラメーターの概要

Name	タイプ	方向	概要
data_center	DataCenter	Out	
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

6.38.2. remove DELETE

データセンターを削除します。

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

特別なパラメーターがないと、データセンターにアタッチされたストレージドメインは切り離され、ストレージから削除されます。この操作の実行中に何かが失敗した場合、たとえば、ストレージからストレージドメインを削除するために使用できるホストがない場合、操作全体が失敗します。

force パラメーターが **true** の場合、たとえば1つのストレージドメインの削除中に何かが失敗した場合でも、操作は常に成功します。障害は無視され、データセンターがデータベースから削除されます。

表6.113 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。
force	ブール値	In	操作中に何かが失敗した場合でも、操作が成功し、ストレージドメインがデータベースから削除されるかどうかを示します。

6.38.2.1. force

操作中に何かが失敗した場合でも、操作が成功し、ストレージドメインがデータベースから削除されるかどうかを示します。

このパラメーターはオプションであり、デフォルト値は **false** です。

6.38.3. update PUT

データセンターを更新します。

name、**description**、**storage_type**、**version**、**storage_format**、および **mac_pool** 要素は、作成後に更新可能です。たとえば、データセンター **123** の名前と説明を変更するには、以下のようにリクエストを送信します。

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

リクエスト本文は以下のようになります。

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

表6.114 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
data_center	DataCenter	In/Out	更新されたデータセンター。

6.39. DATACENTERNETWORK

特定のデータセンターネットワークを管理するサービス

表6.115 メソッドの概要

Name	概要
get	データセンターネットワークの詳細を取得します。
remove	ネットワークを削除します。
update	データセンター内のネットワークを更新します。

6.39.1. get GET

データセンターネットワークの詳細を取得します。

表6.116 パラメーターの概要

Name	タイプ	方向	概要
network	Network	Out	データセンターネットワーク。

6.39.2. remove DELETE

ネットワークを削除します。

6.39.3. update PUT

データセンター内のネットワークを更新します。

表6.117 パラメーターの概要

Name	タイプ	方向	概要
network	Network	In/Out	データセンターネットワーク。

6.40. DATACENTERNETWORKS

データセンターネットワークを管理するサービス

表6.118 メソッドの概要

Name	概要
add	データセンターに新しいネットワークを作成します。
list	データセンター内のネットワークを一覧表示します。

6.40.1. add POST

データセンターに新しいネットワークを作成します。

以下の例のようなリクエストの後に、ID が **123** のデータセンターに新しいネットワークを作成します。

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

ボディで以下の例を使用します。

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

表6.119 パラメーターの概要

Name	タイプ	方向	概要
network	Network	In/Out	データセンターで作成されるネットワークオブジェクト。

6.40.2. list GET

データセンター内のネットワークを一覧表示します。

表6.120 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すネットワークの最大数を設定します。
networks	Network[]	Out	データセンターにあるネットワークの一覧。

6.40.2.1. max

返すネットワークの最大数を設定します。指定されていない場合には、すべてのネットワークが返されます。

6.41. DATACENTERS

データセンターを管理するサービス

表6.121 メソッドの概要

Name	概要
add	新しいデータセンターを作成します。
list	データセンターを一覧表示します。

6.41.1. add POST

新しいデータセンターを作成します。

新しいデータセンターを作成するには、**name** および **local** 要素が必要です。たとえば、共有ストレージ(NFS、iSCSI、またはファイバーチャネル)を使用する **mydc** という名前のデータセンターを作成するには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

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

表6.122 パラメーターの概要

Name	タイプ	方向	概要
data_center	DataCenter	In/Out	追加するデータセンター。

6.41.2. list GET

データセンターを一覧表示します。

以下の要求は、データセンターの表現を取得します。

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

上記のリクエストは **curl** で実行されました:

```
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
```

応答の例を以下に示します。

```
<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>

```

Default データセンターの ID コードに注意してください。このコードは、仮想環境の他のリソースに関連して、このデータセンターを特定します。

データセンターには、ストレージドメインコレクションへのリンクも含まれています。データセンターはこのコレクションを使用して、ストレージドメインのメインコレクションからストレージドメインを割り当てます。

表6.123 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
data_centers	DataCenter[]	Out	
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返すデータセンターの最大数を設定します。
search	String	In	返されたデータセンターを制限するために使用されるクエリー文字列。

6.41.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.41.2.2. max

返すデータセンターの最大数を設定します。指定されていない場合は、すべてのデータセンターが返されます。

6.42. ディスク

単一ディスクを管理します。

表6.124 メソッドの概要

Name	概要
copy	この操作により、指定されたストレージドメインにディスクをコピーします。
export	
get	
move	ディスクを別のストレージドメインに移動します。
remove	
sparsify	ディスクをスパース化します。

6.42.1. copy POST

この操作により、指定されたストレージドメインにディスクをコピーします。

たとえば、以下の要求を使用してディスクのコピーを容易にすることができます。

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

リクエスト本文は以下のようになります。

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

表6.125 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	コピーを非同期的に実行するかどうかを指定します。
disk	ディスク	In	
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

Name	タイプ	方向	概要
storage_domain	StorageDomain	In	

6.42.2. export POST

表6.126 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	エクスポートを非同期で実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
storage_domain	StorageDomain	In	

6.42.3. get GET

表6.127 パラメーターの概要

Name	タイプ	方向	概要
disk	ディスク	Out	

6.42.4. move POST

ディスクを別のストレージドメインに移動します。

たとえば、識別子 **123** のディスクを、識別子 **456** が指定されたストレージドメインに移動するには、以下のリクエストを送信します。

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

リクエスト本文は、以下のようになります。

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

表6.128 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	移動を非同期で実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
storage_domain	StorageDomain	In	

6.42.5. remove DELETE

表6.129 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.42.6. sparsify POST

ディスクをスパース化します。

Sparsification は、ファイルシステムで使用されていないディスクイメージのスペースを解放します。その結果、イメージはストレージの領域を減らします。

現在、スパース化はスナップショットのないディスクでのみ機能します。派生ディスクを持つディスクも許可されていません。

6.43. DISKATTACHMENT

このサービスは、仮想マシンへのディスクの接続を管理します。

表6.130 メソッドの概要

Name	概要
get	起動フラグやディスクへのリンクなど、添付ファイルの詳細を返します。
remove	ディスクアタッチメントを削除します。
update	ディスクアタッチメントとその中のディスクプロパティを更新します。

6.43.1. get GET

起動フラグやディスクへのリンクなど、添付ファイルの詳細を返します。

ディスク接続を取得する例:

■

```
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>
```

表6.131 パラメーターの概要

Name	タイプ	方向	概要
attachment	DiskAttachment	Out	

6.43.2. remove DELETE

ディスクアタッチメントを削除します。

これにより、仮想マシンからディスクがデタッチされるだけで、**detach_only** パラメーターが **false** でない限り、システムからディスクが削除されることはありません。

ディスクアタッチメントを削除する例:

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

表6.132 パラメーターの概要

Name	タイプ	方向	概要
detach_only	ブール値	In	ディスクを仮想マシンからのみ切り離す必要があり、システムからは切り離さないようにする必要があるかどうかを示します。

6.43.2.1. detach_only

ディスクを仮想マシンからのみ切り離す必要があり、システムからは切り離さないようにする必要があるかどうかを示します。デフォルト値は **true** で、システムからディスクを削除しません。

6.43.3. update PUT

ディスクアタッチメントとその中のディスクプロパティを更新します。

```
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>

```

表6.133 パラメーターの概要

Name	タイプ	方向	概要
disk_attachment	DiskAttachment	In/Out	

6.44. DISKATTACHMENTS

このサービスは、仮想マシンにアタッチされている一連のディスクを管理します。アタッチされている各ディスクは、起動可能フラグ、ディスクインターフェイス、およびディスクへの参照を含む `DiskAttachment` で表されます。

表6.134 メソッドの概要

Name	概要
add	仮想マシンに新しいディスクアタッチメントを追加します。
list	仮想マシンに接続されているディスクを一覧表示します。

6.44.1. add POST

仮想マシンに新しいディスクアタッチメントを追加します。ディスクがすでに存在する場合は、**attachment** パラメーターには参照のみを含めることができます。

```

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

```

または、ディスクがまだ存在しない場合には、ディスクの完全な表現を含めることができます。

```

<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>

```

```
...
</disk>
</disk_attachment>
```

この場合、ディスクが作成され、仮想マシンに割り当てられます。

いずれの場合も、ID **345** の仮想マシンに以下の URL を使用します。

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



重要

サーバーは **active** 属性を含まないリクエストを受け入れますが、その効果は定義されていません。場合によっては、ディスクが自動的にアクティベートされ、その他の場合は自動的にアクティブになりません。問題を回避するには、希望の値で **active** 属性を常に含めることを強く推奨します。

表6.135 パラメーターの概要

Name	タイプ	方向	概要
attachment	DiskAttachment	In/Out	

6.44.2. list GET

仮想マシンに接続されているディスクを一覧表示します。

表6.136 パラメーターの概要

Name	タイプ	方向	概要
attachments	DiskAttachment[]	Out	

6.45. DISKPROFILE

表6.137 メソッドの概要

Name	概要
get	
remove	
update	

6.45.1. get GET

表6.138 パラメーターの概要

Name	タイプ	方向	概要
profile	DiskProfile	Out	

6.45.2. remove DELETE

表6.139 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.45.3. update PUT

表6.140 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
profile	DiskProfile	In/Out	

6.46. DISKPROFILES

表6.141 メソッドの概要

Name	概要
add	
list	

6.46.1. add POST

表6.142 パラメーターの概要

Name	タイプ	方向	概要
profile	DiskProfile	In/Out	

6.46.2. list GET

表6.143 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロファイルの最大数を設定します。
profile	DiskProfile[]	Out	

6.46.2.1. max

返すプロファイルの最大数を設定します。指定されていない場合は、すべてのプロファイルが返されます。

6.47. DISKSNAPSHOT

表6.144 メソッドの概要

Name	概要
get	
remove	

6.47.1. get GET

表6.145 パラメーターの概要

Name	タイプ	方向	概要
snapshot	DiskSnapshot	Out	

6.47.2. remove DELETE

表6.146 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.48. DISKSNAPSHOTS

表6.147 メソッドの概要

Name	概要
list	

6.48.1. list GET

表6.148 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すスナップショットの最大数を設定します。
snapshots	DiskSnapshot[]	Out	

6.48.1.1. max

返すスナップショットの最大数を設定します。指定されていない場合は、すべてのスナップショットが返されます。

6.49. ディスク

システムで利用可能なディスクのコレクションを管理します。

表6.149 メソッドの概要

Name	概要
add	新しいフローティングディスクを追加します。
list	ディスクの一覧を取得します。

6.49.1. add POST

新しいフローティングディスクを追加します。

追加できるディスクには、ディスクイメージ、ダイレクト LUN、および Cinder ディスクの 3 つのタイプがあります。

新しいイメージディスクの追加:

新しいフローティングイメージ `Disk` を作成する場合、API には `storage_domain`、`provisioned_size`、および `format` 属性が必要です。

ID `123` のストレージドメインに `provisioned_size`、`format`、`name` を指定して新規のフローティングイメージディスクを作成するには、以下のように要求を送信します。

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

リクエスト本文の場合は、以下のようになります。

```
<disk>
  <storage_domains>
    <storage_domain id="123"/>
  </storage_domains>
```

```
<name>mydisk</name>
<provisioned_size>1048576</provisioned_size>
<format>cow</format>
</disk>
```

新規ダイレクト LUN ディスクの追加:

API を介して新しいフローティングダイレクト LUN を追加する場合、使用できるフレーバーは 2 つあります。

1. **host** 要素の場合: この場合、ホストは健全性チェック (たとえば、LUN が表示されていること) および LUN に関する基本情報 (たとえば、サイズやシリアル) を取得するために使用されません。
2. **host** 要素がない場合: この場合、操作はデータベースのみの操作となり、ストレージにはアクセスされません。

ID が **123**、指定された **alias**、**type**、および ID が **456** の **logical_unit** (属性 **address**、**port**、**target** を持つ) の **host** 要素を持つ新しいフローティングダイレクト LUN ディスクを作成するには、次のようにリクエストを送信します。

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

リクエスト本文の場合は、以下のようになります。

```
<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>
```

ホストを使用せずに新しい Floating ダイレクト LUN ディスクを作成するには、**host** 要素を削除します。

新しい Cinder ディスクを追加します。

新規 Floating Cinder ディスクを作成するには、以下のようによりを送信します。

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

リクエスト本文の場合は、以下のようになります。

```
<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>

```

表6.150 パラメーターの概要

Name	タイプ	方向	概要
disk	ディスク	In/Out	ディスク。

6.49.2. list GET

ディスクの一覧を取得します。

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

以下のような XML 応答を取得します。

```

<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>

```

表6.151 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。

Name	タイプ	方向	概要
disks	Disk[]	Out	取得されたディスクの一覧。
max	Integer	In	返すディスクの最大数を設定します。
search	String	In	返されたディスクを制限するために使用されるクエリー文字列。
unregistered	ブール値	In	ストレージドメイン内の登録済みディスクまたは未登録ディスクのリストを取得するかどうかを示します。

6.49.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.49.2.2. max

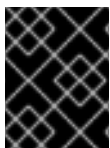
返すディスクの最大数を設定します。指定しない場合、すべてのディスクが返されます。

6.49.2.3. unregistered

ストレージドメイン内の登録済みディスクまたは未登録ディスクのリストを取得するかどうかを示します。ストレージドメイン内の未登録ディスクのリストを取得するには、呼び出しで未登録フラグを示す必要があります。たとえば、未登録ディスクの一覧を取得するための REST API 呼び出しは、以下のようになります。

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

未登録フラグのデフォルト値は **false** です。



重要

このパラメーターは、割り当てられたストレージドメインの **disks** サブコレクションにのみ適用され、最上位の **ディスク コレクション** には適用されません。

6.50. DOMAIN

システム内の認証ドメインの詳細を表示するサービス

表6.152 メソッドの概要

Name	概要
get	認証ドメイン情報を取得します。

6.50.1. get GET

認証ドメイン情報を取得します。

使用方法

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

ドメイン情報を返します。

```
<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>
```

表6.153 パラメーターの概要

Name	タイプ	方向	概要
domain	Domain	Out	認証ドメイン。

6.51. DOMAINGROUP

表6.154 メソッドの概要

Name	概要
get	

6.51.1. get GET

表6.155 パラメーターの概要

Name	タイプ	方向	概要
get	グループ	Out	

6.52. DOMAINGROUPS

表6.156 メソッドの概要

Name	概要
list	

6.52.1. list GET

表6.157 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
groups	Group[]	Out	
max	Integer	In	返すグループの最大数を設定します。
search	String	In	返されるグループを制限するために使用されるクエリ文字列。

6.52.1.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.52.1.2. max

返すグループの最大数を設定します。指定のない場合は、すべてのグループが返されます。

6.53. DOMAINUSER

システム内のドメインユーザーを表示するサービス

表6.158 メソッドの概要

Name	概要
get	ドメインユーザー情報を取得します。

6.53.1. get GET

ドメインユーザー情報を取得します。

使用方法

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

ドメインユーザー情報を返します。

```
<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>

```

表6.159 パラメーターの概要

Name	タイプ	方向	概要
user	User	Out	ドメインユーザー。

6.54. DOMAINUSERS

システム内のすべてのドメインユーザーを一覧表示するサービス。

表6.160 メソッドの概要

Name	概要
list	ドメイン内のすべてのユーザーを一覧表示します。

6.54.1. list GET

ドメイン内のすべてのユーザーを一覧表示します。

使用方法

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

ドメイン内のユーザーの一覧を返します。

```

<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>

```

表6.161 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
max	Integer	In	返すユーザーの最大数を設定します。
search	String	In	返されるユーザーを制限するために使用されるクエリー文字列。
users	User[]	Out	ドメイン内のユーザーのリスト。

6.54.1.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.54.1.2. max

返すユーザーの最大数を設定します。指定しない場合、すべてのユーザーが返されます。

6.55. ドメイン

システム内のすべての認証ドメインを一覧表示するサービス。

表6.162 メソッドの概要

Name	概要
list	システム内のすべての認証ドメインを一覧表示します。

6.55.1. list GET

システム内のすべての認証ドメインを一覧表示します。

使用方法

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

ドメインの一覧を返します。

```
<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>

```

表6.163 パラメーターの概要

Name	タイプ	方向	概要
domains	Domain[]	Out	ドメインの一覧。
max	Integer	In	返すドメインの最大数を設定します。

6.55.1.1. max

返すドメインの最大数を設定します。指定されていない場合は、すべてのドメインが返されます。

6.56. ENGINEKATELLOERRATA

エンジンに割り当てられた Katello エラータを管理するサービスこの情報は Katello から取得されます。

表6.164 メソッドの概要

Name	概要
list	Katello エラータの表現を取得します。

6.56.1. list GET

Katello エラータの表現を取得します。

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

以下のような XML で応答を受け取ります。

```

<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>

```

```

</katello_erratum>
...
</katello_errata>

```

表6.165 パラメーターの概要

Name	タイプ	方向	概要
errata	KatelloErratum[]	Out	Katello エラータの表現。
max	Integer	In	返すエラータの最大数を設定します。

6.56.1.1. max

返すエラータの最大数を設定します。指定されていない場合は、すべてのエラータが返されます。

6.57. イベント

システムのイベントを管理するサービス。

表6.166 メソッドの概要

Name	概要
get	イベントを取得します。
remove	内部監査ログからイベントを削除します。

6.57.1. get GET

イベントを取得します。

イベントの取得例:

```
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>

```


イベントにある情報に応じてフィールドの数が変わることに注意してください。たとえば、ストレージドメイン関連のイベントの場合、ストレージドメインの参照と、このストレージドメインが置かれているデータセンターのリファレンスを取得します。

表6.167 パラメーターの概要

Name	タイプ	方向	概要
event	イベント	Out	

6.57.2. remove DELETE

内部監査ログからイベントを削除します。

以下のリクエストを送信するとイベントを削除できます。

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

表6.168 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.58. イベント

システムのイベントを管理するサービス。

表6.169 メソッドの概要

Name	概要
add	外部イベントを内部監査ログに追加します。
list	イベントの一覧を取得します。
undelete	

6.58.1. add POST

外部イベントを内部監査ログに追加します。

これは、システムの管理者に関連するイベントを検出または生成する外部システムとの統合を目的としています。たとえば、外部監視ツールは、仮想マシンのゲストオペレーティングシステム内でファイルシステムがいっぱいであることを検出できる場合があります。このイベントは、以下のような要求を送信する内部監査ログに追加できます。

```
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>
```

イベントは特定のオブジェクトにリンクすることもできます。たとえば、上記のイベントは、**vm** リンクを使用して、発生した特定の仮想マシンにリンクできます。

```
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>
```



注記

前述の例の **vm** のようにリンクを使用する場合は、**id** 属性のみが許可されます。**name** 属性 (指定されている場合) は無視されます。

表6.170 パラメーターの概要

Name	タイプ	方向	概要
event	イベント	In/Out	

6.58.2. list GET

イベントの一覧を取得します。

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

上記のリクエストに対して、以下のレスポンスを受け取ります。

```
<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>

```

以下のイベントが発生します。

- id="1": 管理ユーザーアカウントの API ログイン。
- id="2": API は admin ユーザーアカウントからログアウトします。

表6.171 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
events	Event[]	Out	
from	Integer	In	返される最初のイベントの識別子を示します。
max	Integer	In	返すイベントの最大数を設定します。
search	String	In	イベントサービスは、他のリソースサービスと同様に検索クエリーを提供します。

6.58.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.58.2.2. from

返される最初のイベントの識別子を示します。イベントの識別子は厳密に増加しているため、このパラメーターを使用すると、その識別子が指定の値以上のイベントのみが返されます。たとえば、以下のリクエストは **123** 以上の識別子のイベントのみを返します。

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

このパラメーターは任意です。指定されていない場合には、返される最初のイベントが最後に生成されます。

6.58.2.3. max

返すイベントの最大数を設定します。指定されていない場合は、すべてのイベントが返されます。

6.58.2.4. search

イベントサービスは、他のリソースサービスと同様に検索クエリーを提供します。

特定の重大度を指定して検索できます。

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

上記のリクエストに対して、重大度が **normal** と等しいイベントのリストを取得します。

```
<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>
```

仮想化環境は、一定期間後に大量のイベントを生成します。ただし、API は1つの検索クエリーのデフォルトイベントのみを表示します。デフォルト以外のものを表示するには、API は結果を検索クエリーで page コマンドを使用してページに分割します。次の検索クエリーは、sortby 句と組み合わせてページ値を使用して結果をページ分割するように API に指示します。

```
sortby time asc page 1
```

以下の例は、イベントリソースをページ分割します。URL でエンコードされたリクエストは以下のとおりです。

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

ページ値を増やして、結果の次のページを表示します。

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

6.58.3. undelete POST

表6.172 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除解除を非同期的に実行するかどうかを示します。

6.59. EXTERNALCOMPUTERESOURCE

表6.173 メソッドの概要

Name	概要
get	

6.59.1. get GET

表6.174 パラメーターの概要

Name	タイプ	方向	概要
resource	ExternalComputeResource	Out	

6.60. EXTERNALCOMPUTERESOURCES

表6.175 メソッドの概要

Name	概要
list	

6.60.1. list GET

表6.176 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すリソースの最大数を設定します。
resources	ExternalComputeResource[]	Out	

6.60.1.1. max

返すリソースの最大数を設定します。指定されていない場合は、すべてのリソースが返されます。

6.61. EXTERNALDISCOVEREDHOST

表6.177 メソッドの概要

Name	概要
get	

6.61.1. get GET

表6.178 パラメーターの概要

Name	タイプ	方向	概要
host	ExternalDisc overedHost	Out	

6.62. EXTERNALDISCOVEREDHOSTS

表6.179 メソッドの概要

Name	概要
list	

6.62.1. list GET

表6.180 パラメーターの概要

Name	タイプ	方向	概要
hosts	ExternalDisc overedHost[]	Out	
max	Integer	In	返すホストの最大数を設定します。

6.62.1.1. max

返すホストの最大数を設定します。指定されていない場合は、すべてのホストが返されます。

6.63. EXTERNALHOST

表6.181 メソッドの概要

Name	概要
get	

6.63.1. get GET

表6.182 パラメーターの概要

Name	タイプ	方向	概要
host	ExternalHost	Out	

6.64. EXTERNALHOSTGROUP

表6.183 メソッドの概要

Name	概要
get	

6.64.1. get GET

表6.184 パラメーターの概要

Name	タイプ	方向	概要
group	ExternalHost Group	Out	

6.65. EXTERNALHOSTGROUPS

表6.185 メソッドの概要

Name	概要
list	

6.65.1. list GET

表6.186 パラメーターの概要

Name	タイプ	方向	概要
groups	ExternalHost Group[]	Out	

Name	タイプ	方向	概要
max	Integer	In	返すグループの最大数を設定します。

6.65.1.1. max

返すグループの最大数を設定します。指定のない場合は、すべてのグループが返されます。

6.66. EXTERNALHOSTPROVIDER

表6.187 メソッドの概要

Name	概要
get	
importcertificates	
remove	
testconnectivity	
update	

6.66.1. get GET

表6.188 パラメーターの概要

Name	タイプ	方向	概要
provider	ExternalHost Provider	Out	

6.66.2. importcertificates POST

表6.189 パラメーターの概要

Name	タイプ	方向	概要
certificates	Certificate[]	In	

6.66.3. remove DELETE

表6.190 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.66.4. testconnectivity POST

表6.191 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	テストを非同期で実行する必要があるかどうかを示します。

6.66.5. update PUT

表6.192 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
provider	ExternalHost Provider	In/Out	

6.67. EXTERNALHOSTPROVIDERS

表6.193 メソッドの概要

Name	概要
add	
list	

6.67.1. add POST

表6.194 パラメーターの概要

Name	タイプ	方向	概要
provider	ExternalHost Provider	In/Out	

6.67.2. list GET

表6.195 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロバイダーの最大数を設定します。
providers	ExternalHost Provider[]	Out	

6.67.2.1. max

返すプロバイダーの最大数を設定します。指定しない場合は、すべてのプロバイダーが返されます。

6.68. EXTERNALHOSTS

表6.196 メソッドの概要

Name	概要
list	

6.68.1. list GET

表6.197 パラメーターの概要

Name	タイプ	方向	概要
hosts	ExternalHost []	Out	
max	Integer	In	返すホストの最大数を設定します。

6.68.1.1. max

返すホストの最大数を設定します。指定されていない場合は、すべてのホストが返されます。

6.69. EXTERNALPROVIDER

表6.198 メソッドの概要

Name	概要
importcertificates	
testconnectivity	

6.69.1. importcertificates POST

表6.199 パラメーターの概要

表6.199 パラメーターの概要

Name	タイプ	方向	概要
certificates	Certificate[]	In	

6.69.2. testconnectivity POST

表6.200 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	テストを非同期で実行する必要があるかどうかを示します。

6.70. EXTERNALPROVIDERCERTIFICATE

表6.201 メソッドの概要

Name	概要
get	

6.70.1. get GET

表6.202 パラメーターの概要

Name	タイプ	方向	概要
certificate	証明書	Out	

6.71. EXTERNALPROVIDERCERTIFICATES

表6.203 メソッドの概要

Name	概要
list	

6.71.1. list GET

表6.204 パラメーターの概要

Name	タイプ	方向	概要
certificates	Certificate[]	Out	

Name	タイプ	方向	概要
max	Integer	In	返す証明書の最大数を設定します。

6.71.1.1. max

返す証明書の最大数を設定します。指定しない場合は、すべての証明書が返されます。

6.72. EXTERNALVMIMPORTS

外部仮想マシンをインポートする機能を提供します。

表6.205 メソッドの概要

Name	概要
add	この操作は、KVM、XEN、VMware などの外部ハイパーバイザーから仮想マシンをインポートするために使用されます。

6.72.1. add POST

この操作は、KVM、XEN、VMware などの外部ハイパーバイザーから仮想マシンをインポートするために使用されます。

たとえば、VMware からの仮想マシンのインポートは、次のリクエストを使用して簡単に行うことができます。

POST /externalvmimports

タイプ [ExternalVmImport](#) のリクエスト本文の場合、次に例を示します。

```
<external_vm_import>
  <vm>
    <name>my_vm</name>
  </vm>
  <cluster id="360014051136c20574f743bdbd28177fd" />
  <storage_domain id="8bb5ade5-e988-4000-8b93-dbfc6717fe50" />
  <name>vm_name_as_is_in_vmware</name>
  <sparse>true</sparse>
  <username>vmware_user</username>
  <password>123456</password>
  <provider>VMWARE</provider>
  <url>vpx://wmware_user@vcenter-host/DataCenter/Cluster/esxi-host?no_verify=1</url>
  <drivers_iso id="virtio-win-1.6.7.iso" />
</external_vm_import>
```

表6.206 パラメーターの概要

Name	タイプ	方向	概要
import	ExternalVml mport	In/Out	

6.73. FENCEAGENT

表6.207 メソッドの概要

Name	概要
get	
remove	
update	

6.73.1. get GET

表6.208 パラメーターの概要

Name	タイプ	方向	概要
agent	Agent	Out	

6.73.2. remove DELETE

表6.209 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.73.3. update PUT

表6.210 パラメーターの概要

Name	タイプ	方向	概要
agent	Agent	In/Out	
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。

6.74. FENCEAGENTS

表6.211 メソッドの概要

Name	概要
add	
list	

6.74.1. add POST

表6.212 パラメーターの概要

Name	タイプ	方向	概要
agent	Agent	In/Out	

6.74.2. list GET

表6.213 パラメーターの概要

Name	タイプ	方向	概要
agents	Agent[]	Out	
max	Integer	In	返すエージェントの最大数を設定します。

6.74.2.1. max

返すエージェントの最大数を設定します。指定しない場合、すべてのエージェントが返されます。

6.75. FILE

表6.214 メソッドの概要

Name	概要
get	

6.75.1. get GET

表6.215 パラメーターの概要

Name	タイプ	方向	概要
file	File	Out	

6.76. ファイル

クライアントが利用可能なファイルを一覧表示する方法を提供します。

このサービスは、管理者がアップロードする ISO イメージおよび仮想フロッピーディスク(VFD)を含む ISO ストレージドメインを特別に対象としています。

CDROM デバイスを仮想マシンに追加するには、ISO ストレージドメインのファイルからの ISO イメージが必要です。

表6.216 メソッドの概要

Name	概要
list	

6.76.1. list GET

表6.217 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
file	File[]	Out	
max	Integer	In	返すファイルの最大数を設定します。
search	String	In	返されたファイルを制限するために使用されるクエリ文字列。

6.76.1.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.76.1.2. max

返すファイルの最大数を設定します。指定されていない場合は、すべてのファイルが返されます。

6.77. フィルター

表6.218 メソッドの概要

Name	概要
get	

Name	概要
remove	

6.77.1. get GET

表6.219 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
result	フィルター	Out	

6.77.2. remove DELETE

表6.220 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.78. フィルター

表6.221 メソッドの概要

Name	概要
add	
list	

6.78.1. add POST

表6.222 パラメーターの概要

Name	タイプ	方向	概要
filter	フィルター	In/Out	

6.78.2. list GET

表6.223 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
filters	Filter[]	Out	
max	Integer	In	返すフィルターの最大数を設定します。

6.78.2.1. max

返すフィルターの最大数を設定します。指定しない場合、すべてのエージェントが返されます。

6.79. GLUSTERBRICK

このサービスは単一の gluster ブリックを管理します。

表6.224 メソッドの概要

Name	概要
get	ブリックの詳細を取得します。
remove	ブリックを削除します。
replace	このブリックを新しいものに置き換えます。

6.79.1. get GET

ブリックの詳細を取得します。

ヘッダー **All-Content** が **true** に設定された基礎となる gluster ボリュームから、ステータスの詳細を取得します。これは、**gluster volume status <volumename> <brickname> detail** を実行するのと同じです。

たとえば、gluster ボリューム **123** の **brick234** の詳細を取得するには、次のようなリクエストを送信します。

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

これにより、以下のような応答ボディが返されます。

```
<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>xfs</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>

```

表6.225 パラメーターの概要

Name	タイプ	方向	概要
brick	GlusterBrick	Out	

6.79.2. remove DELETE

ブリックを削除します。

基礎となる gluster ボリュームからブリックを削除し、データベースからエントリーを削除します。これは、データ移行なしで単一のブリックを削除する場合にのみ使用できます。複数のブリックとデータ移行を削除するには、代わりに [migrate](#) を使用します。

たとえば、gluster ボリューム **123** からブリック **234** を削除するには、以下のように要求を送信します。

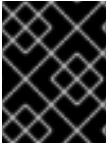
```
DELETE /ovirt-engine/api/clusters/567/glustervolumes/123/glusterbricks/234
```

表6.226 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.79.3. replace POST

このブリックを新しいものに置き換えます。



重要

この操作はエンジンのバージョン 3.5 以降非推奨となり、今後削除されます。代わりに、[add brick](#) を使用してブリックを [migrate brick](#) します。

表6.227 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	代替を非同期的に実行するかどうかを指定します。
force	ブール値	In	

6.80. GLUSTERBRICKS

このサービスは、gluster ボリュームで gluster ブリックを管理します。

表6.228 メソッドの概要

Name	概要
activate	削除操作のデータ移行の後のデータ移行をアクティベートします。
add	gluster ボリュームに、ブリックの一覧を追加します。
list	gluster ボリュームのブリックを一覧表示します。
migrate	ブリックを削除する前にデータの移行を開始します。
remove	gluster ボリュームからブリックを削除します。
stopmigrate	削除のブリック操作についてのデータのブリックからの移行を停止します。

6.80.1. activate POST

削除操作のデータ移行の後のデータ移行をアクティベートします。

ブリックからのデータ移行が完了し、ユーザーがブリックを削除したくないと、ブリックをアクティベートするために使用されます。削除用に以前にマークされていたブリックは、通常のブリックとして使用されるようになりました。

たとえば、データの移行元である `glustervolume 123` のブリックを保持するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

-

```

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

```

表6.229 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクティベーションを非同期で実行する必要があるかどうかを示します。
bricks	GlusterBrick[]	In	再アクティベートする必要があるブリックの一覧。

6.80.2. add POST

gluster ボリュームに、ブリックの一覧を追加します。

ブリックを追加して gluster ボリュームを拡張するのに使用します。複製されたボリュームタイプの場合は、**replica_count** パラメーターを渡す必要があります。レプリカ数が増える場合、ブリックの数はレプリカセットの数と同じである必要があります。

たとえば、gluster ボリューム **123** にブリックを追加するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```

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

```

表6.230 パラメーターの概要

Name	タイプ	方向	概要
bricks	GlusterBrick[]	In/Out	ボリュームに追加するブリックの一覧
replica_count	Integer	In	ボリューム後の追加操作のレプリカ数。
stripe_count	Integer	In	追加後の操作のストライプ数。

6.80.3. list GET

gluster ボリュームのブリックを一覧表示します。

たとえば、gluster ボリューム **123** のブリックを一覧表示するには、以下のように要求を送信します。

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

以下のような出力を提供します。

```
<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>
```

表6.231 パラメーターの概要

Name	タイプ	方向	概要
bricks	GlusterBrick[]	Out	
max	Integer	In	返すブリックの最大数を設定します。

6.80.3.1. max

返すブリックの最大数を設定します。指定されていない場合は、すべてのブリックが返されます。

6.80.4. migrate POST

ブリックを削除する前にデータの移行を開始します。

ブリックの削除は2つのステップで、削除されるブリック上のデータが最初に残りのブリックに移行します。移行が完了すると、API [remove](#) を介してブリックの削除が確定されます。いずれの時点でも、[stopmigrate](#) をキャンセルするアクションを呼び出す必要があります。

たとえば、ID が **123** の gluster ボリュームからブリックを削除するには、次のリクエストを送信します。

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

リクエスト本文は以下のようになります。

■

```

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

```

移行プロセスは、ジョブを使用して API から返されたジョブ ID と、ステップを使用して `job` の `step` から追跡できます。

表6.232 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	移行を非同期的に実行するかどうかを指定します。
bricks	GlusterBrick[]	In	データ移行を開始する必要があるブリックの一覧。

6.80.5. remove DELETE

gluster ボリュームからブリックを削除します。

データ損失なしでブリックを削除する場合は、最初に `stopmigrate` を使用してそれらのデータを削除してからそれらを削除する方法が推奨されます。削除前にイメージストリームで移行が呼び出されなかった場合、データ移行なしにブリックが削除され、データが失われる可能性があります。

たとえば、gluster ボリューム **123** からブリックを削除するには、以下のように要求を送信します。

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

リクエスト本文は以下のようになります。

```

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

```

表6.233 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。
bricks	GlusterBrick[]	In	削除するブリックの一覧
replica_count	Integer	In	ボリューム後の追加操作のレプリカ数。

6.80.6. stopmigrate POST

削除のブリック操作についてのデータのブリックからの移行を停止します。

ユーザーがブリックの使用を継続したい場合に備えて、2ステップのブリック削除プロセスの一部として開始されたデータ移行をキャンセルするには。削除用にマーク付けされていたブリックは、この操作の後に通常のブリックとして機能します。

たとえば、gluster ボリューム **123** のブリックからデータの移行を停止するには、以下のように要求を送信します。

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

リクエスト本文は以下のようになります。

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

表6.234 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。
bricks	GlusterBrick[]	In	データ移行を停止する必要があるブリックの一覧。

6.80.6.1. bricks

データ移行を停止する必要があるブリックの一覧。この一覧は、[migrate](#) のために渡された引数と一致する必要があります。

6.81. GLUSTERHOOK

表6.235 メソッドの概要

Name	概要
disable	クラスターの全サーバーで Gluster フックを無効にすることにより、クラスターのサーバー間でフックのステータスの競合を解決します。
enable	クラスターの全サーバーで Gluster フックを無効にすることにより、クラスターのサーバー間でフックのステータスの競合を解決します。
get	

Name	概要
remove	クラスターのすべてのサーバーからこの Gluster フックを削除し、これをデータベースから削除します。
resolve	解決のタイプに応じて、不足しているフックの競合を解決します。

6.81.1. disable POST

クラスターの全サーバーで Gluster フックを無効にすることにより、クラスターのサーバー間でフックのステータスの競合を解決します。これにより、データベースでフックのステータスが **DISABLED** に更新されました。

表6.236 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.81.2. enable POST

クラスターの全サーバーで Gluster フックを無効にすることにより、クラスターのサーバー間でフックのステータスの競合を解決します。これにより、データベースでフックのステータスが **DISABLED** に更新されました。

表6.237 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.81.3. get GET

表6.238 パラメーターの概要

Name	タイプ	方向	概要
hook	GlusterHook	Out	

6.81.4. remove DELETE

クラスターのすべてのサーバーからこの Gluster フックを削除し、これをデータベースから削除します。

表6.239 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.81.5. resolve POST

解決のタイプに応じて、不足しているフックの競合を解決します。

ADD の場合は、フックがないすべてのサーバーに、エンジンデータベースに保存されているフックをコピーして解決します。エンジンは、フックがないすべてのサーバーの一覧を維持します。

COPY の場合、フックが欠落しているすべてのサーバーにエンジンデータベースに保存されているフックをコピーして、フックの内容で競合を解決します。エンジンは、コンテンツが競合するサーバーの一覧を維持します。ホスト ID がパラメーターとして渡される場合、サーバーのフックコンテンツがマスターとして使用され、クラスター内の他のサーバーにコピーすることができます。

表6.240 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。
host	ホスト	In	
resolution_type	String	In	

6.82. GLUSTERHOOKS

表6.241 メソッドの概要

Name	概要
list	

6.82.1. list GET

表6.242 パラメーターの概要

Name	タイプ	方向	概要
hooks	GlusterHook []	Out	
max	Integer	In	返すフックの最大数を設定します。

6.82.1.1. max

返すフックの最大数を設定します。指定されていない場合は、すべてのフックが返されます。

6.83. GLUSTERVOLUME

このサービスは単一の gluster ボリュームを管理します。

表6.243 メソッドの概要

Name	概要
get	gluster ボリュームの詳細を取得します。
getprofilestatistics	gluster ボリュームプロファイルの統計を取得します。
rebalance	gluster ボリュームをリバランスします。
remove	gluster ボリュームを削除します。
resetalloptions	gluster ボリュームに設定されたすべてのオプションをリセットします。
resetoption	gluster ボリュームで特定のオプションをリセットします。
setoption	gluster ボリュームに特定のオプションを設定します。
start	gluster ボリュームを起動します。
startprofile	gluster ボリュームのプロファイリングを開始します。
stop	gluster ボリュームを停止します。
stopprofile	gluster ボリュームのプロファイリングを停止します。
stoprebalance	gluster ボリュームのリバランスを停止します。

6.83.1. get GET

gluster ボリュームの詳細を取得します。

たとえば、クラスター **456** で識別子 **123** の gluster ボリュームの詳細を取得するには、以下のように要求を送信します。

```
GET /ovirt-engine/api/clusters/456/glustervolumes/123
```

この GET リクエストは以下の出力を返します。

```
<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>

```

表6.244 パラメーターの概要

Name	タイプ	方向	概要
volume	GlusterVolume	Out	gluster ボリュームを表します。

6.83.2. getprofilestatistics POST

gluster ボリュームプロファイルの統計を取得します。

たとえば、クラスター **456** で識別子 **123** の gluster ボリュームのプロファイル統計を取得するには、以下のように要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/getprofilestatistics
```

表6.245 パラメーターの概要

Name	タイプ	方向	概要
details	GlusterVolumeProfileDetails	Out	アクションから返される Gluster ボリュームのプロファイリング情報。

6.83.3. rebalance POST

gluster ボリュームをリバランスします。

gluster ボリュームのリバランスは、すべてのブリックにデータを均等に分散するのに役立ちます。(データの移行なし) gluster ボリュームを拡張または縮小した後、ブリック間でデータのリバランスを行う必要があります。複製されていないボリュームでは、リバランス操作を実行するために、すべてのブリックをオンラインにする必要があります。複製されたボリュームでは、レプリカ内の少なくとも1つのブリックがオンラインである必要があります。

たとえば、クラスター **456** で識別子 **123** の gluster ボリュームをリバランスするには、以下のように要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/rebalance
```

表6.246 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	リバランスを非同期的に実行するかどうかを指定します。
fix_layout	ブール値	In	true に設定すると、リバランスによってレイアウトが修正され、ボリュームに追加された新規データがすべてのホストに分散されます。
force	ブール値	In	リバランスを強制的に起動するかどうかを示します。

6.83.3.1. fix_layout

true に設定すると、リバランスによってレイアウトが修正され、ボリュームに追加された新規データがすべてのホストに分散されます。ただし、既存のデータは移行/リバランスされません。デフォルトは **false** です。

6.83.3.2. force

リバランスを強制的に起動するかどうかを示します。rebalance コマンドは、古いクライアントがクラスターに接続されている場合でも、force オプションを使用して実行できます。ただし、これにより、データが失われる可能性があります。デフォルトは **false** です。

6.83.4. remove DELETE

gluster ボリュームを削除します。

たとえば、クラスター **456** で識別子 **123** のボリュームを削除するには、以下のように要求を送信します。

```
DELETE /ovirt-engine/api/clusters/456/glustervolumes/123
```

表6.247 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.83.5. resetalloptions POST

gluster ボリュームに設定されたすべてのオプションをリセットします。

たとえば、クラスター **456** で識別子 **123** を使用して gluster ボリューム内のすべてのオプションをリセットするには、以下のように要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/resetalloptions
```

表6.248 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	リセットを非同期で実行する必要があるかどうかを示します。

6.83.6. resetoption POST

gluster ボリュームで特定のオプションをリセットします。

たとえば、クラスター **456** で識別子 **123** の gluster ボリュームで特定の **option1** をリセットするには、以下のように要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/resetoption
```

リクエスト本文は、以下のようになります。

```
<action>
  <option name="option1"/>
</action>
```

表6.249 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	リセットを非同期で実行する必要があるかどうかを示します。
force	ブール値	In	
option	オプション	In	リセットするオプション。

6.83.7. setoption POST

gluster ボリュームに特定のオプションを設定します。

たとえば、クラスター **456** で識別子 **123** で gluster ボリュームに **option1** の値 **value1** を設定するには、以下のように要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/setoption
```

リクエスト本文は、以下のようになります。

```
<action>
  <option name="option1" value="value1"/>
</action>
```

表6.250 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。
option	オプション	In	設定するオプション。

6.83.8. start POST

gluster ボリュームを起動します。

Gluster ボリュームは、データの読み取り/書き込みを開始する必要があります。たとえば、クラスター **456** で識別子 **123** で gluster ボリュームを起動するには、以下のように要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/start
```

表6.251 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。
force	ブール値	In	ボリュームが強制的に起動するかどうかを示します。

6.83.8.1. force

ボリュームが強制的に起動するかどうかを示します。gluster ボリュームがすでに開始されているが、いくつか/すべてのブリックがダウンしている場合は、強制開始を使用してすべてのブリックを起動できます。デフォルトは **false** です。

6.83.9. startprofile POST

gluster ボリュームのプロファイリングを開始します。

たとえば、クラスター **456** で識別子 **123** の gluster ボリュームのプロファイリングを開始するには、以下のような要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/startprofile
```

表6.252 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.83.10. stop POST

gluster ボリュームを停止します。

ボリュームを停止すると、そのデータにはアクセスできなくなります。

たとえば、クラスター **456** で識別子 **123** の gluster ボリュームを停止するには、以下のように要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/stop
```

表6.253 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。
force	ブール値	In	

6.83.11. stopprofile POST

gluster ボリュームのプロファイリングを停止します。

たとえば、クラスター **456** で識別子 **123** の gluster ボリュームのプロファイリングを停止するには、以下のように要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/stopprofile
```

表6.254 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.83.12. stoprebalance POST

gluster ボリュームのリバランスを停止します。

たとえば、クラスター **456** で識別子 **123** の gluster ボリュームの再調整を停止するには、以下のように要求を送信します。

```
POST /ovirt-engine/api/clusters/456/glustervolumes/123/stoprebalance
```

表6.255 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.84. GLUSTERVOLUMES

このサービスは、クラスターで利用可能な gluster ボリュームのコレクションを管理します。

表6.256 メソッドの概要

Name	概要
add	新しい gluster ボリュームを作成します。
list	クラスター内のすべての gluster ボリュームを一覧表示します。

6.84.1. add POST

新しい gluster ボリュームを作成します。

ボリュームは、**volume** パラメーターのプロパティに基づいて作成されます。プロパティ **name**、**volume_type**、および **bricks** が必要です。

たとえば、**myvolume** という名前のボリュームをクラスター **123** に追加するには、以下の要求を送信します。

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

リクエスト本文は、以下のようになります。

```
<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>
```


表6.257 パラメーターの概要

Name	タイプ	方向	概要
volume	GlusterVolume	In/Out	ボリュームを作成する gluster ボリュームの定義が入力として渡され、新規に作成されるボリュームが返されます。

6.84.2. list GET

クラスター内のすべての gluster ボリュームを一覧表示します。

たとえば、クラスター **456** のすべての Gluster ボリュームを一覧表示するには、以下のように要求を送信します。

```
GET /ovirt-engine/api/clusters/456/glustervolumes
```

表6.258 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
max	Integer	In	返すボリュームの最大数を設定します。
search	String	In	返されたボリュームを制限するために使用されるクエリー文字列です。
volumes	GlusterVolume[]	Out	

6.84.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.84.2.2. max

返すボリュームの最大数を設定します。指定されていない場合は、すべてのボリュームが返されます。

6.85. グループ

表6.259 メソッドの概要

Name	概要
get	
remove	

6.85.1. get GET

表6.260 パラメーターの概要

Name	タイプ	方向	概要
get	グループ	Out	

6.85.2. remove DELETE

表6.261 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.86. グループ

表6.262 メソッドの概要

Name	概要
add	ディレクトリーサービスからグループを追加します。
list	

6.86.1. add POST

ディレクトリーサービスからグループを追加します。ドメイン名は、認証プロバイダーの名前であることに注意してください。

たとえば、**internal-authz** 認証プロバイダーから **Developers** グループを追加するには、以下のような要求を送信します。

```
POST /ovirt-engine/api/groups
```

リクエスト本文は以下のようになります。

```
<group>
  <name>Developers</name>
  <domain>
```

```
<name>internal-authz</name>
</domain>
</group>
```

表6.263 パラメーターの概要

Name	タイプ	方向	概要
group	グループ	In/Out	

6.86.2. list GET

表6.264 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
groups	Group[]	Out	
max	Integer	In	返すグループの最大数を設定します。
search	String	In	返されるグループを制限するために使用されるクエリ文字列。

6.86.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.86.2.2. max

返すグループの最大数を設定します。指定のない場合は、すべてのグループが返されます。

6.87. ホスト

ホストを管理するサービス

表6.265 メソッドの概要

Name	概要
activate	仮想マシンの実行など、使用するホストをアクティブにします。
approve	仮想化環境で使用するために事前にインストールされたハイパーバイザーホストを承認します。

Name	概要
commitnetconfig	ネットワーク設定を正常としてマークし、ホスト内にその設定を持続させます。
deactivate	メンテナンスタスクを実行するためにホストを無効にします。
enrollcertificate	ホストの証明書を登録します。
fence	ホストの電源管理デバイスを制御します。
forceselectspm	ホストをストレージプールマネージャー(SPM)として手動で設定します。
get	ホストの詳細を取得します。
install	VDSM と関連ソフトウェアをホストにインストールします。
iscsidiscover	イニシエーターの詳細を使用して、ホスト上の iSCSI ターゲットを検出します。
iscsilogin	ターゲットの詳細を使用して、ホストの iSCSI ターゲットにログインします。
refresh	ホストデバイスと機能を更新します。
remove	システムからホストを削除します。
setupnetworks	この方法は、ホストのネットワークインターフェイスの設定を変更するために使用されます。
unregisteredstoragedomainsdiscover	
update	ホストのプロパティを更新します。
upgrade	ホストで VDSM と選択したソフトウェアをアップグレードします。
upgradecheck	ホストにアップグレードが利用可能であるかどうかを確認します。

6.87.1. activate POST

仮想マシンの実行など、使用するホストをアクティブにします。

表6.266 パラメーターの概要

Name	タイプ	方向	概要
------	-----	----	----

Name	タイプ	方向	概要
async	ブール値	In	アクティベーションを非同期で実行する必要があるかどうかを示します。

6.87.2. approve POST

仮想化環境で使用するために事前にインストールされたハイパーバイザーホストを承認します。

このアクションは、このホストのターゲットクラスターを定義するためのオプションのクラスター要素も受け入れます。

表6.267 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	承認を非同期的に実行するかどうかを指定します。
cluster	Cluster	In	

6.87.3. commitnetconfig POST

ネットワーク設定を正常としてマークし、ホスト内にその設定を持続させます。

API ユーザーは、ネットワーク設定をコミットして、ホストネットワークインターフェイスのアタッチメントまたはデタッチメントを永続化するか、ボンディングされたインターフェイスの作成と削除を永続化します。



重要

ネットワーク設定は、設定の変更によってホスト接続が失われないことをエンジンが確認した後にのみ、コミットされます。ホストの接続が失われた場合、ホストを再起動する必要があります、自動的に以前のネットワーク設定に戻ります。

たとえば、ID が **123** のホストのネットワーク設定をコミットするには、次のようなリクエストを送信します。

```
POST /ovirt-engine/api/hosts/123/commitnetconfig
```

リクエスト本文は以下のようになります。

```
<action/>
```

表6.268 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.87.4. deactivate POST

メンテナンスタスクを実行するためにホストを無効にします。

表6.269 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	非アクティブ化を非同期で実行する必要があるかどうかを示します。
reason	String	In	
stop_gluster_service	ブール値	In	ホストの非アクティブ化の一環として gluster サービスを停止する必要があるかどうかを示します。

6.87.4.1. stop_gluster_service

ホストの非アクティブ化の一環として gluster サービスを停止する必要があるかどうかを示します。gluster ホストのメンテナンスタスク中に使用できます。この変数のデフォルト値は **false** です。

6.87.5. enrollcertificate POST

ホストの証明書を登録します。この警告が近づいている、またはすでに有効期限が切れているという警告が発生した場合に役に立ちます。

表6.270 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	登録を非同期で実行する必要があるかどうかを示します。

6.87.6. fence POST

ホストの電源管理デバイスを制御します。

たとえば、ホストを起動すると仮定します。これは、以下の方法で実行できます。

```
#!/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"
```

表6.271 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	フェンシングを非同期で実行する必要があるかどうかを示します。
fence_type	String	In	
power_management	PowerManagement	Out	

6.87.7. forceselectspm POST

ホストをストレージプールマネージャー(SPM)として手動で設定します。

```
POST /ovirt-engine/api/hosts/123/forceselectspm
```

リクエスト本文は以下のようになります。

```
<action/>
```

表6.272 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.87.8. get GET

ホストの詳細を取得します。

表6.273 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
host	ホスト	Out	

6.87.9. install POST

VDSM と関連ソフトウェアをホストにインストールします。ホストタイプは、アクションの追加パラメーターを定義します。

curl と JSON を使用してホストをインストールする例、プレーン:

```
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"
```

ホスト型エンジンコンポーネントと共に **curl** および JSON を使用してホストをインストールする例:

```
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"
```



重要

ホストの再インストール時にエンジンのバージョン 4.1.2 以降、デフォルトでホストファイアウォールの定義を上書きします。

表6.274 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	インストールを非同期で実行する必要があるかどうかを示します。

Name	タイプ	方向	概要
deploy_hosted_engine	ブール値	In	true に設定すると、このホストもホスト型エンジンコンポーネントをデプロイする必要があることを意味します。
host	ホスト	In	この override_iptables プロパティは、ファイアウォール設定をデフォルト設定に置き換える必要があるかどうかを示すために使用されます。
image	String	In	oVirt ノードのインストール時に、イメージの ISO ファイルが必要になります。
root_password	String	In	SSH 経由でホストに接続するために使用される root ユーザーのパスワード。
ssh	Ssh	In	ホストへの接続に使用される SSH の詳細。
undeploy_hosted_engine	ブール値	In	true に設定すると、このホストはホストエンジンコンポーネントをアンデプロイする必要があり、このホストは高可用性クラスターの一部として機能しません。

6.87.9.1. deploy_hosted_engine

true に設定すると、このホストもホスト型エンジンコンポーネントをデプロイする必要があることを意味します。欠落している値は **true**、つまりデプロイとして扱われます。このパラメーターを省略すると、**false** を意味し、ホスト型エンジン領域で操作を実行しません。

6.87.9.2. undeploy_hosted_engine

true に設定すると、このホストはホストエンジンコンポーネントをアンデプロイする必要があり、このホストは高可用性クラスターの一部として機能しません。欠落している値は **true** として扱われます。つまり、デプロイされていない省略は **false** を意味し、ホスト型エンジン領域で操作を実行しません。

6.87.10. iscsidiscover POST

イニシエーターの詳細を使用して、ホスト上の iSCSI ターゲットを検出します。

表6.275 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	検出を非同期で実行する必要があるかどうかを示します。
iscsi	IscsiDetails	In	ターゲット iSCSI デバイス。

Name	タイプ	方向	概要
iscsi_targets	String[]	Out	iSCSI ターゲット。

6.87.11. iscsilogin POST

ターゲットの詳細を使用して、ホストの iSCSI ターゲットにログインします。

表6.276 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	ログインを非同期で実行する必要があるかどうかを示します。
iscsi	IscsiDetails	In	ターゲット iSCSI デバイス。

6.87.12. refresh POST

ホストデバイスと機能を更新します。

表6.277 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	リフレッシュを非同期で実行する必要があるかどうかを示します。

6.87.13. remove DELETE

システムからホストを削除します。

```
#!/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"
```

表6.278 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.87.14. setupnetworks POST

この方法は、ホストのネットワークインターフェイスの設定を変更するために使用されます。

たとえば、3つのネットワークインターフェイス **eth0**、**eth1**、および **eth2** を持つホストがあり、**eth0** および **eth1** を使用して新しいボンディングを設定し、その上に VLAN を配置するとします。簡単なシェルスクリプトと **curl** コマンドライン HTTP クライアントを使用すると、以下のように実行できます。

```
#!/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"

```

これは API のバージョン 4 で有効であることに注意してください。以前のバージョンでは、一部の要素は XML 要素ではなく XML 属性として表されていました。特に、**options** と **ip** 要素は次のように表されました。

```

<options name="mode" value="4"/>
<options name="miimon" value="100"/>
<ip address="192.168.122.10" netmask="255.255.255.0"/>

```

Python SDK を使用すると、以下のコードでも同じことができます。

```

# 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()
```

**重要**

ネットワーク設定がホストに保存されていること、およびホストの再起動時に適用されることを確認するには、[commitnetconfig](#) を呼び出すことを忘れないでください。

表6.279 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。
check_connectivity	ブール値	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

表6.280 パラメーターの概要

Name	タイプ	方向	概要
------	-----	----	----

Name	タイプ	方向	概要
async	ブール値	In	検出を非同期で実行する必要があるかどうかを示します。
iscsi	iscsiDetails	In	
storage_domain_mains	StorageDomainain[]	Out	

6.87.16. update PUT

ホストのプロパティを更新します。

たとえば、ホストのカーネルコマンドラインを更新するには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<host>
  <os>
    <custom_kernel_cmdline>vfio_iommu_type1.allow_unsafe_interrupts=1</custom_kernel_cmdline>
  </os>
</host>
```

表6.281 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
host	ホスト	In/Out	

6.87.17. upgrade POST

ホストで VDSM と選択したソフトウェアをアップグレードします。

表6.282 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アップグレードを非同期で実行する必要があるかどうかを示します。

6.87.18. upgradecheck POST

ホストにアップグレードが利用可能であるかどうかを確認します。利用可能なアップグレードがある場合は、webadmin のホストステータスアイコンの横にアイコンが表示されます。また、アップグレード

の可否を示す監査ログメッセージも追加されます。アップグレードは、webadmin から、または [アップグレード](#) ホストアクションを使用して開始できます。

6.88. HOSTDEVICE

ホストの特定のデバイスにアクセスするためのサービス。

表6.283 メソッドの概要

Name	概要
get	特定のホストのデバイスに関する情報を取得します。

6.88.1. get GET

特定のホストのデバイスに関する情報を取得します。

ホストデバイスを取得する例:

```
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>
```

表6.284 パラメーターの概要

Name	タイプ	方向	概要
device	HostDevice	Out	

6.89. HOSTDEVICES

ホストデバイスにアクセスするためのサービス。

表6.285 メソッドの概要

Name	概要
list	ホストのデバイスを一覧表示します。

6.89.1. list GET

ホストのデバイスを一覧表示します。

表6.286 パラメーターの概要

Name	タイプ	方向	概要
devices	HostDevice[]	Out	
max	Integer	In	返すデバイスの最大数を設定します。

6.89.1.1. max

返すデバイスの最大数を設定します。指定しない場合、すべてのデバイスが返されます。

6.90. HOSTHOOK

表6.287 メソッドの概要

Name	概要
get	

6.90.1. get GET

表6.288 パラメーターの概要

Name	タイプ	方向	概要
hook	Hook	Out	

6.91. HOSTHOOKS

表6.289 メソッドの概要

Name	概要
list	

6.91.1. list GET

表6.290 パラメーターの概要

Name	タイプ	方向	概要
hooks	Hook[]	Out	
max	Integer	In	返すフックの最大数を設定します。

6.91.1.1. max

返すフックの最大数を設定します。指定されていない場合は、すべてのフックが返されます。

6.92. HOSTNIC

ホストのネットワークインターフェイスを管理するサービス。

表6.291 メソッドの概要

Name	概要
get	
updatevirtualfunctionsconfiguration	現在のリソースが SR-IOV 対応 NIC を表す場合、アクションは仮想機能設定を更新します。

6.92.1. get GET

表6.292 パラメーターの概要

Name	タイプ	方向	概要
nic	HostNic	Out	

6.92.2. updatevirtualfunctionsconfiguration POST

現在のリソースが SR-IOV 対応 NIC を表す場合、アクションは仮想機能設定を更新します。入力、以下のプロパティの少なくとも1つで設定されている必要があります。

- **allNetworksAllowed**
- **numberOfVirtualFunctions**

プロパティの意味については、**HostNicVirtualFunctionsConfiguration** タイプを参照してください。

表6.293 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
virtual_functions_configuration	HostNicVirtualFunctionsConfiguration	In	

6.93. HOSTNICS

ホストのネットワークインターフェイスを管理するサービス。

表6.294 メソッドの概要

Name	概要
list	

6.93.1. list GET

表6.295 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す NIC の最大数を設定します。
nics	HostNic[]	Out	

6.93.1.1. max

返す NIC の最大数を設定します。指定されていない場合は、すべての NIC が返されます。

6.94. HOSTNUMANODE

表6.296 メソッドの概要

Name	概要
get	

6.94.1. get GET

表6.297 パラメーターの概要

Name	タイプ	方向	概要
node	NumaNode	Out	

6.95. HOSTNUMANODES

表6.298 メソッドの概要

Name	概要
list	

6.95.1. list GET

表6.299 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すノードの最大数を設定します。
nodes	NumaNode[]	Out	

6.95.1.1. max

返すノードの最大数を設定します。指定されていない場合は、すべてのノードが返されます。

6.96. HOSTSTORAGE

ホストストレージを管理するサービス。

表6.300 メソッドの概要

Name	概要
list	ストレージのリストを取得します。

6.96.1. list GET

ストレージのリストを取得します。

```
GET /ovirt-engine/api/hosts/123/storage
```

取得する XML 応答は次のようになります。

```
<host_storages>
  <host_storage id="123">
    ...
  </host_storage>
  ...
</host_storages>
```

表6.301 パラメーターの概要

Name	タイプ	方向	概要
report_status	ブール値	In	ストレージ内の LUN のステータスを確認する必要があるかどうかを示します。
storages	HostStorage []	Out	ストレージのリストを取得しました。

6.96.1.1. report_status

ストレージ内の LUN のステータスを確認する必要があるかどうかを示します。LUN のステータスの確認は非常に重要な操作であり、このデータは必ずしもユーザーが必要とするものではありません。このパラメーターは、LUN のステータスチェックを実行しないオプションを提供します。

デフォルトは、後方互換性を確保するために **true** です。

LUN ステータスの例を次に示します。

```
<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>
```

これは、LUN ステータスのない例です。

```
<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. ホスト

ホストを管理するサービス。

表6.302 メソッドの概要

Name	概要
add	新しいホストを作成します。

Name	概要
list	利用可能なすべてのホストのリストを取得します。

6.97.1. add POST

新しいホストを作成します。

ホストは、**host** パラメーターの属性に基づいて作成されます。**名前**、**address** プロパティ、および **root_password** プロパティが必要です。

たとえば、ホストを追加するには、以下のリクエストを送信します。

```
POST /ovirt-engine/api/hosts
```

リクエスト本文は、以下のようになります。

```
<host>
  <name>myhost</name>
  <address>myhost.example.com</address>
  <root_password>myrootpassword</root_password>
</host>
```



注記

root_password 要素は、クライアントが提供する初期表現にのみ含まれ、後続の要求から返される表現には公開されません。



重要

ホストが新たに追加された場合のエンジンのバージョン 4.1.2 以降、デフォルトでホストファイアウォールの定義を上書きします。

ホスト型エンジンホストを追加するには、オプションの **deploy_hosted_engine** パラメーターを使用します。

```
POST /ovirt-engine/api/hosts?deploy_hosted_engine=true
```

表6.303 パラメーターの概要

Name	タイプ	方向	概要
deploy_hosted_engine	ブール値	In	true に設定すると、このホストもホスト型エンジンコンポーネントをデプロイする必要があることを意味します。
host	ホスト	In/Out	新規ホストの作成元となるホスト定義がパラメーターとして渡され、新しく作成されたホストが返されます。

Name	タイプ	方向	概要
undeploy_hosted_engine	ブール値	In	true に設定すると、このホストはホストエンジンコンポーネントをアンデプロイする必要があり、このホストは高可用性クラスターの一部として機能しません。

6.97.1.1. deploy_hosted_engine

true に設定すると、このホストもホスト型エンジンコンポーネントをデプロイする必要があることを意味します。欠落している値は **true**、つまりデプロイとして扱われます。このパラメーターを省略すると、**false** を意味し、ホスト型エンジン領域で操作を実行しません。

6.97.1.2. undeploy_hosted_engine

true に設定すると、このホストはホストエンジンコンポーネントをアンデプロイする必要があり、このホストは高可用性クラスターの一部として機能しません。欠落している値は **true**（デプロイ解除）として処理されます。このパラメーターを省略すると、**false** を意味し、ホスト型エンジン領域で操作を実行しません。

6.97.2. list GET

利用可能なすべてのホストのリストを取得します。

たとえば、ホストを一覧表示するには、次のリクエストを送信します。

```
GET /ovirt-engine/api/hosts
```

応答本文は以下のようになります。

```
<hosts>
  <host href="/ovirt-engine/api/hosts/123" id="123">
    ...
  </host>
  <host href="/ovirt-engine/api/hosts/456" id="456">
    ...
  </host>
  ...
</hosts>
```

表6.304 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

Name	タイプ	方向	概要
hosts	Host[]	Out	
max	Integer	In	返すホストの最大数を設定します。
search	String	In	返されたホストを制限するために使用されるクエリ文字列。

6.97.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.97.2.2. max

返すホストの最大数を設定します。指定されていない場合は、すべてのホストが返されます。

6.98. アイコン

アイコンを管理するサービス (読み取り専用)。

表6.305 メソッドの概要

Name	概要
get	アイコンを取得します。

6.98.1. get GET

アイコンを取得します。

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

次のような XML 応答が得られます。

```
<icon id="123">
  <data>Some binary data here</data>
  <media_type>image/png</media_type>
</icon>
```

表6.306 パラメーターの概要

Name	タイプ	方向	概要
icon	アイコン	Out	取得したアイコン。

6.99. ICONS

アイコンを管理するサービス。

表6.307 メソッドの概要

Name	概要
list	アイコンのリストを取得します。

6.99.1. list GET

アイコンのリストを取得します。

GET /ovirt-engine/api/icons

次のような XML 応答が得られます。

```
<icons>
  <icon id="123">
    <data>...</data>
    <media_type>image/png</media_type>
  </icon>
  ...
</icons>
```

表6.308 パラメーターの概要

Name	タイプ	方向	概要
icons	icon[]	Out	アイコンのリストを取得しました。
max	Integer	In	返すアイコンの最大数を設定します。

6.99.1.1. max

返すアイコンの最大数を設定します。指定しない場合、すべてのアイコンが返されます。

6.100. IMAGE

表6.309 メソッドの概要

Name	概要
get	
import	イメージをインポートします。

6.100.1. get GET

表6.310 パラメーターの概要

Name	タイプ	方向	概要
image	Image	Out	

6.100.2. import POST

イメージをインポートします。

import_as_template パラメーターが **true** の場合、イメージはテンプレートとしてインポートされません。それ以外の場合は、ディスクとしてインポートされます。

テンプレートとしてインポートする場合、テンプレートの名前はオプションの **template.name** パラメーターで指定することができます。このパラメーターが指定されていない場合、テンプレートの名前はエンジンによって **GlanceTemplate-x** として自動的に割り当てられます (**x** は7つのランダムな16進数文字になります)。

ディスクとしてインポートする場合、ディスクの名前はオプションの **disk.name** パラメーターで指定することができます。このパラメーターが指定されていない場合、ディスクの名前はエンジンによって **GlanceDisk-x** として自動的に割り当てられます (**x** はイメージ識別子の7つの16進数文字になります)。

エンジンによって自動的に生成されるこれらの名前を回避するために、テンプレートまたはディスク名を常に明示的に指定することをお勧めします。

表6.311 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	インポートを非同期で実行する必要があるかどうかを示します。
cluster	Cluster	In	import_as_template パラメーターが true に設定されている場合に、イメージのインポート先となるクラスター。
disk	ディスク	In	インポートするディスク。
import_as_template	ブール値	In	インポートしたディスクからテンプレートを作成するかどうかを指定します。
storage_domain	StorageDomain	In	ディスクのインポート先となるストレージドメイン。
template	Template	In	import_as_template パラメーターが true に設定されている場合に作成されるテンプレートの名前。

6.101. IMAGETRANSFER

このサービスは、イメージ転送を制御するメカニズムを提供します。クライアントは、「[ImageTransfers](#)」サービスの [追加](#) を使用して転送を作成し、データを転送するイメージを示す必要があります。

その後、転送はこのサービスによって管理されます。

たとえば、ID **52cb593f-837c-4633-a444-35a0a0383706** でディスクイメージにアップロードする場合、クライアントは以下のように oVirt の Python の SDK を使用できます。

```
transfers_service = system_service.image_transfers_service()
transfer = transfers_service.add(
    types.ImageTransfer(
        image=types.Image(
            id='52cb593f-837c-4633-a444-35a0a0383706'
        )
    )
)
```

ユーザーがディスクをアップロードではなくダウンロードしたい場合は、転送の `direction` 属性として **download** を指定する必要があります。これにより、書き込み権限ではなく、イメージからの読み取り権限が付与されます。

例:

```
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
    )
)
```

転送には、アップロード/ダウンロードのフローを管理するフェーズがあります。このようなフローを実装するクライアントは、転送のフェーズをポーリング/チェックし、それに応じて動作する必要があります。想定されるすべてのフェーズは、[ImageTransferPhase](#) にあります。

新しい転送を追加した後、そのフェーズは [初期化](#) されます。クライアントは、転送が変更されるまで、転送のフェーズをポーリングする必要があります。フェーズが [転送中](#) になると、セッションは転送を開始できる状態になります。

以下に例を示します。

```
transfer_service = transfers_service.image_transfer_service(transfer.id)
while transfer.phase == types.ImageTransferPhase.INITIALIZING:
    time.sleep(3)
transfer = transfer_service.get()
```

その段階で、転送のフェーズが [paused_system](#) である場合は、セッションが正常に確立されていないこととなります。その原因として考えられるのは、転送対象として選択されたホストで `ovirt-imageio-daemon` が動作していないことです。転送は、それを管理するサービスの [resume](#) を呼び出すことによって再開できます。

セッションが正常に確立された場合：返される転送エンティティには [proxy_url](#) および [signed_ticket](#)

属性が含まれます。これは、クライアントが必要なデータを転送するために使用する必要があります。クライアントは、イメージのデータで HTTPS 要求を送信するための手法およびツールを選択できます。

- **proxy_url** は、I/O を行うイメージへのプロキシサーバーのアドレスです。
- **signed_ticket** は、信頼できる通信を実行するために HTTPS リクエストの **Authentication** ヘッダーに追加する必要があるコンテンツです。

たとえば、転送の実行には Python の `HTTPSConnection` を使用することができます。そのため、転送する転送には **transfer_headers** dict が設定されます。

```
transfer_headers = {
    'Authorization': transfer.signed_ticket,
}
```

Python の **HTTPSConnection** を使用すると、新しい接続が確立されます。

```
# 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)
)
```

アップロードするには、送信される特定のコンテンツ範囲を **Content-Range** HTTPS ヘッダーにメモする必要があります。これは、転送を複数のリクエストに分割し、より柔軟なプロセスを可能にするために使用できます。

これには、クライアントはチャンネルを開いたままにするために転送セッションを繰り返し拡張する必要があります。そうしないと、セッションが終了し、転送は **paused_system** フェーズになり、サーバーへの HTTPS 要求は拒否されます。

たとえば、クライアントはファイルのチャンクを繰り返し処理し、サービスをセッションを拡張するように要求しながらプロキシサーバーに送信できます。

```
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
```

同様に、ダウンロード転送では **Range** ヘッダーを送信し、ディスクをチャンクにダウンロードすることでダウンロードプロセスをより簡単に管理できるようにする必要があります。

たとえば、クライアントはディスクイメージのチャンクを繰り返し処理しますが、今回は独自のファイルをイメージにアップロードするのではなく、ローカルファイルにダウンロードします。

```
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
```

転送を終了すると、ユーザーは `finalize` を呼び出す必要があります。これにより、転送プロセスを完了する最終の調整と検証が行われます。

以下に例を示します。

```
transfer_service.finalize()
```

エラーの場合は、転送のフェーズが `finished_failure` に変更され、ディスクのステータスが **Illegal** に変更されます。それ以外の場合は、`finished_success` に変更され、ディスクを使用する準備が整います。どちらの場合も、転送エンティティはすぐに削除されます。

表6.312 メソッドの概要

Name	概要
extend	イメージ転送セッションを延長します。
finalize	データの転送が終了したら、転送を完了します。
get	イメージ転送エンティティを取得します。
pause	イメージ転送セッションを一時停止します。
再開	イメージ転送セッションを再開します。

6.101.1. extend POST

イメージ転送セッションを延長します。

6.101.2. finalize POST

データの転送が終了したら、転送を完了します。

これにより、転送されるデータが有効であり、転送の対象となったイメージエンティティーに適合していることが確認されます。具体的には、イメージエンティティーが QCOW ディスクの場合、アップロードされたデータが実際に QCOW ファイルであり、イメージにバックアップファイルがないことを確認します。

6.101.3. get GET

イメージ転送エンティティーを取得します。

表6.313 パラメーターの概要

Name	タイプ	方向	概要
image_transfer	ImageTransfer	Out	

6.101.4. pause POST

イメージ転送セッションを一時停止します。

6.101.5. resume POST

イメージ転送セッションを再開します。クライアントは、転送のフェーズが **resuming** と異なるまで、ポーリングする必要があります。以下に例を示します。

```
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

このサービスは、oVirt で Image I/O API を実行するためのイメージ転送を管理します。詳細は、[イメージ転送](#) を参照してください。

表6.314 メソッドの概要

Name	概要
add	新しいイメージ転送を追加します。
list	現在実行されているイメージ転送のリストを取得します。

6.102.1. add POST

新しいイメージ転送を追加します。新しい転送を行うには、イメージを指定する必要があります。

表6.315 パラメーターの概要

Name	タイプ	方向	概要
image_transfer	ImageTransfer	In/Out	

6.102.2. list GET

現在実行されているイメージ転送のリストを取得します。

表6.316 パラメーターの概要

Name	タイプ	方向	概要
image_transfer	imagetransfer[]	Out	

6.103. イメージ

表6.317 メソッドの概要

Name	概要
list	

6.103.1. list GET

表6.318 パラメーターの概要

Name	タイプ	方向	概要
images	Image[]	Out	
max	Integer	In	返すイメージの最大数を設定します。

6.103.1.1. max

返すイメージの最大数を設定します。指定しない場合、すべてのイメージが返されます。

6.104. INSTANCETYPE

表6.319 メソッドの概要

Name	概要
get	特定のインスタンスタイプとその属性を取得します。
remove	システムから特定のインスタンスタイプを削除します。
update	特定のインスタンスタイプとその属性を更新します。

6.104.1. get GET

特定のインスタンスタイプとその属性を取得します。

GET /ovirt-engine/api/instancetypees/123

表6.320 パラメーターの概要

Name	タイプ	方向	概要
instance_type	InstanceType	Out	

6.104.2. remove DELETE

システムから特定のインスタンスタイプを削除します。

インスタンスタイプの削除後にインスタンスタイプXを使用して仮想マシンが作成された場合、仮想マシンのインスタンスタイプは **custom** に設定されます。

DELETE /ovirt-engine/api/instancetypees/123

表6.321 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.104.3. update PUT

特定のインスタンスタイプとその属性を更新します。

すべての属性は、作成後に編集可能です。インスタンスタイプXを使用して仮想マシンが作成され、インスタンスタイプXの一部の設定が更新された場合、仮想マシンの設定はエンジンによって自動的に更新されます。

PUT /ovirt-engine/api/instancetypees/123

たとえば、インスタンスタイプ **123** のメモリーを1GiBに更新し、CPUトポロジーを2ソケットと1コアに設定するには、以下のようなリクエストを送信します。

■


```

<instance_type>
  <memory>1073741824</memory>
  <cpu>
    <topology>
      <cores>1</cores>
      <sockets>2</sockets>
      <threads>1</threads>
    </topology>
  </cpu>
</instance_type>

```

表6.322 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
instance_type	InstanceType	In/Out	

6.105. INSTANCETYPEGRAPHICSCONSOLE

表6.323 メソッドの概要

Name	概要
get	インスタンスタイプのグラフィックコンソール設定を取得します。
remove	インスタンスタイプからグラフィックコンソールを削除します。

6.105.1. get GET

インスタンスタイプのグラフィックコンソール設定を取得します。

表6.324 パラメーターの概要

Name	タイプ	方向	概要
console	GraphicsConsole	Out	インスタンスタイプのグラフィックコンソールに関する情報。

6.105.2. remove DELETE

インスタンスタイプからグラフィックコンソールを削除します。

表6.325 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.106. INSTANCETYPEGRAPHICSCONSOLES

表6.326 メソッドの概要

Name	概要
add	インスタンスタイプに新しいグラフィックコンソールを追加します。
list	インスタンスタイプの設定済みグラフィックコンソールをすべて一覧表示します。

6.106.1. add POST

インスタンスタイプに新しいグラフィックコンソールを追加します。

表6.327 パラメーターの概要

Name	タイプ	方向	概要
console	GraphicsConsole	In/Out	

6.106.2. list GET

インスタンスタイプの設定済みグラフィックコンソールをすべて一覧表示します。

表6.328 パラメーターの概要

Name	タイプ	方向	概要
consoles	GraphicsConsole[]	Out	インスタンスタイプのグラフィックコンソールのリスト。
max	Integer	In	返すコンソールの最大数を設定します。

6.106.2.1. max

返すコンソールの最大数を設定します。指定しない場合、すべてのコンソールが返されます。

6.107. INSTANCETYPENIC

表6.329 メソッドの概要

Name	概要
get	インスタンスタイプのネットワークインターフェイス設定を取得します。
remove	インスタンスタイプからネットワークインターフェイスを削除します。
update	インスタンスタイプのネットワークインターフェイス設定を更新します。

6.107.1. get GET

インスタンスタイプのネットワークインターフェイス設定を取得します。

表6.330 パラメーターの概要

Name	タイプ	方向	概要
nic	Nic	Out	

6.107.2. remove DELETE

インスタンスタイプからネットワークインターフェイスを削除します。

表6.331 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.107.3. update PUT

インスタンスタイプのネットワークインターフェイス設定を更新します。

表6.332 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
nic	Nic	In/Out	

6.108. INSTANCETYPENICS

表6.333 メソッドの概要

Name	概要
add	インスタンスタイプに新しいネットワークインターフェイスを追加します。

Name	概要
list	インスタンスタイプの設定済みネットワークインターフェイスをすべて一覧表示します。

6.108.1. add POST

インスタンスタイプに新しいネットワークインターフェイスを追加します。

表6.334 パラメーターの概要

Name	タイプ	方向	概要
nic	Nic	In/Out	

6.108.2. list GET

インスタンスタイプの設定済みネットワークインターフェイスをすべて一覧表示します。

表6.335 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す NIC の最大数を設定します。
nics	Nic[]	Out	
search	String	In	返されたボリュームを制限するために使用されるクエリー文字列です。

6.108.2.1. max

返す NIC の最大数を設定します。指定されていない場合は、すべての NIC が返されます。

6.109. INSTANCETYPEWATCHDOG

表6.336 メソッドの概要

Name	概要
get	インスタンスタイプのウォッチドッグ設定を取得します。
remove	インスタンスタイプからウォッチドッグを削除します。
update	インスタンスタイプのウォッチドッグ設定を更新します。

6.109.1. get GET

インスタンスタイプのウォッチドッグ設定を取得します。

表6.337 パラメーターの概要

Name	タイプ	方向	概要
watchdog	Watchdog	Out	

6.109.2. remove DELETE

インスタンスタイプからウォッチドッグを削除します。

表6.338 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.109.3. update PUT

インスタンスタイプのウォッチドッグ設定を更新します。

表6.339 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
watchdog	Watchdog	In/Out	

6.110. INSTANCETYPEWATCHDOGS

表6.340 メソッドの概要

Name	概要
add	インスタンスタイプに新しいウォッチドッグを追加します。
list	インスタンスタイプの設定済みウォッチドッグをすべて一覧表示します。

6.110.1. add POST

インスタンスタイプに新しいウォッチドッグを追加します。

表6.341 パラメーターの概要

Name	タイプ	方向	概要
watchdog	Watchdog	In/Out	

6.110.2. list GET

インスタンスタイプの設定済みウォッチドッグをすべて一覧表示します。

表6.342 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すウォッチドッグの最大数を設定します。
search	String	In	返されたボリュームを制限するために使用されるクエリー文字列です。
watchdogs	Watchdog[]	Out	

6.110.2.1. max

返すウォッチドッグの最大数を設定します。指定しない場合、すべてのウォッチドッグが返されます。

6.111. INSTANCETYPES

表6.343 メソッドの概要

Name	概要
add	新しいインスタンスタイプを作成します。
list	システム内の既存のすべてのインスタンスタイプを一覧表示します。

6.111.1. add POST

新しいインスタンスタイプを作成します。

これには name 属性のみが必要であり、仮想マシンのすべてのハードウェア設定を含めることができます。

```
POST /ovirt-engine/api/instancetypes
```

リクエスト本文は以下のようになります。

```
<instance_type>
  <name>myinstancetype</name>
</template>
```

次のようなリクエスト本文を使用して、すべてのハードウェア設定でインスタンスタイプを作成します。

```
<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>

```

表6.344 パラメーターの概要

Name	タイプ	方向	概要
instance_type	InstanceType	In/Out	

6.111.2. list GET

システム内の既存のすべてのインスタンスタイプを一覧表示します。

表6.345 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
instance_type	InstanceType[]	Out	
max	Integer	In	返すインスタンスタイプの最大数を設定します。
search	String	In	返されたボリュームを制限するために使用されるクエリー文字列です。

6.111.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.111.2.2. max

返すインスタンスタイプの最大数を設定します。指定しない場合、すべてのインスタンスタイプが返されます。

6.112. ISCSIBOND

表6.346 メソッドの概要

Name	概要
get	
remove	既存の iSCSI ボンディングを削除します。
update	iSCSI ボンディングを更新します。

6.112.1. get GET

表6.347 パラメーターの概要

Name	タイプ	方向	概要
bond	iscsiBond	Out	

6.112.2. remove DELETE

既存の iSCSI ボンディングを削除します。

たとえば、iSCSI ボンディング **456** を削除するには、次のようなリクエストを送信します。

```
DELETE /ovirt-engine/api/datacenters/123/iscsibonds/456
```

表6.348 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.112.3. update PUT

iSCSI ボンディングを更新します。

iSCSI ボンディングの更新は、**name** 属性と **description** 属性でのみ実行できます。たとえば、データセンター **123** の iSCSI ボンディング **456** を更新するには、次のようなリクエストを送信します。

```
PUT /ovirt-engine/api/datacenters/123/iscsibonds/1234
```

リクエスト本文は以下のようになります。

```
<iscsi_bond>
  <name>mybond</name>
  <description>My iSCSI bond</description>
</iscsi_bond>
```

表6.349 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
bond	iscsiBond	In/Out	

6.113. ISCSIBONDS

表6.350 メソッドの概要

Name	概要
add	データセンターに新しい iSCSI ボンディングを作成します。
list	

6.113.1. add POST

データセンターに新しい iSCSI ボンディングを作成します。

たとえば、ストレージ接続 **456** および **789** を使用してデータセンター **123** に新しい iSCSI ボンディングを作成するには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<iscsi_bond>
  <name>mybond</name>
  <storage_connections>
    <storage_connection id="456"/>
    <storage_connection id="789"/>
  </storage_connections>
  <networks>
    <network id="abc"/>
  </networks>
</iscsi_bond>
```

表6.351 パラメーターの概要

Name	タイプ	方向	概要
bond	iscsiBond	In/Out	

6.113.2. list GET

表6.352 パラメーターの概要

Name	タイプ	方向	概要
bonds	lscsiBond[]	Out	
max	Integer	In	返すボンディングの最大数を設定します。

6.113.2.1. max

返すボンディングの最大数を設定します。指定しない場合、すべてのボンディングが返されます。

6.114. JOB

ジョブを管理するサービス

表6.353 メソッドの概要

Name	概要
clear	システムによってクリアされる外部ジョブの実行を設定します。
end	外部ジョブの実行を終了としてマークします。
get	ジョブを取得します。

6.114.1. clear POST

システムによってクリアされる外部ジョブの実行を設定します。

たとえば、識別子 **123** でジョブを設定するには、次の要求を送信します。

```
POST /ovirt-engine/api/jobs/clear
```

リクエスト本文は、以下のようになります。

```
<action/>
```

表6.354 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.114.2. end POST

外部ジョブの実行を終了としてマークします。

たとえば、識別子 **123** でジョブを終了するには、次の要求を送信します。

-

POST /ovirt-engine/api/jobs/end

リクエスト本文は、以下のようになります。

```
<action>
  <force>true</force>
  <status>finished</status>
</action>
```

表6.355 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。
force	ブール値	In	ジョブを強制終了する必要があるかどうかを示します。
succeeded	ブール値	In	ジョブを正常終了または失敗としてマークする必要があるかどうかを示します。

6.114.2.1. succeeded

ジョブを正常終了または失敗としてマークする必要があるかどうかを示します。

このパラメーターはオプションであり、デフォルト値は **true** です。

6.114.3. get GET

ジョブを取得します。

GET /ovirt-engine/api/jobs/123

以下のような XML で応答を受け取ります。

```
<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>
```

表6.356 パラメーターの概要

Name	タイプ	方向	概要
job	Job	Out	ジョブの表現を取得します。

6.115. ジョブ

ジョブを管理するサービス。

表6.357 メソッドの概要

Name	概要
add	外部ジョブを追加します。
list	ジョブの表現を取得します。

6.115.1. add POST

外部ジョブを追加します。

たとえば、次のリクエストでジョブを追加する場合で、

```
POST /ovirt-engine/api/jobs
```

リクエスト本文は、以下のようになります。

```
<job>
  <description>Doing some work</description>
  <auto_cleared>true</auto_cleared>
</job>
```

応答は以下のようになります。

```
<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>
```

表6.358 パラメーターの概要

Name	タイプ	方向	概要
job	Job	In/Out	追加されるジョブ。

6.115.2. list GET

ジョブの表現を取得します。

GET /ovirt-engine/api/jobs

以下のような XML で応答を受け取ります。

```
<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>
```

表6.359 パラメーターの概要

Name	タイプ	方向	概要
jobs	Job[]	Out	ジョブの表現。
max	Integer	In	返すジョブの最大数を設定します。

6.115.2.1. max

返すジョブの最大数を設定します。指定されていない場合は、すべてのジョブが返されます。

6.116. KATELLOERRATA

Katello エラータを管理するサービスです。この情報は Katello から取得されます。

表6.360 メソッドの概要

Name	概要
list	Katello エラータの表現を取得します。

6.116.1. list GET

Katello エラータの表現を取得します。

GET /ovirt-engine/api/katelloerrata

以下のような XML で応答を受け取ります。

```
<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>
```

表6.361 パラメーターの概要

Name	タイプ	方向	概要
errata	KatelloErratum[]	Out	Katello エラータの表現。
max	Integer	In	返すエラータの最大数を設定します。

6.116.1.1. max

返すエラータの最大数を設定します。指定されていない場合は、すべてのエラータが返されます。

6.117. KATELLOERRATUM

Katello エラータを管理するサービスです。

表6.362 メソッドの概要

Name	概要
get	Katello エラータを取得します。

6.117.1. get GET

Katello エラータを取得します。

GET /ovirt-engine/api/katelloerrata/123

以下のような XML で応答を受け取ります。

```
<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>
```

表6.363 パラメーターの概要

Name	タイプ	方向	概要
erratum	KatelloErratum	Out	Katello エラータの表現を取得します。

6.118. MACPOOL

表6.364 メソッドの概要

Name	概要
get	
remove	MAC アドレスプールを削除します。
update	MAC アドレスプールを更新します。

6.118.1. get GET

表6.365 パラメーターの概要

Name	タイプ	方向	概要
pool	MacPool	Out	

6.118.2. remove DELETE

MAC アドレスプールを削除します。

たとえば、ID が **123** の MAC アドレスプールを削除するには、次のようなリクエストを送信します。

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

表6.366 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.118.3. update PUT

MAC アドレスプールを更新します。

name、**description**、**allow_duplicates**、および **ranges** 属性を更新できます。

たとえば、ID **123** の MAC アドレスプールを更新するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<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>
```

表6.367 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
pool	MacPool	In/Out	

6.119. MACPOOLS

表6.368 メソッドの概要

Name	概要
add	新しい MAC アドレスプールを作成します。
list	

6.119.1. add POST

新しい MAC アドレスプールを作成します。

MAC アドレスプールを作成するには、**name** 属性と **range** 属性の値が必要です。

たとえば、MAC アドレスプールを作成するには、以下のようなリクエストを送信します。

```
POST /ovirt-engine/api/macpools
```

リクエスト本文は以下のようになります。

```
<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>
```

表6.369 パラメーターの概要

Name	タイプ	方向	概要
pool	MacPool	In/Out	

6.119.2. list GET

表6.370 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプールの最大数を設定します。
pools	MacPool[]	Out	

6.119.2.1. max

返すプールの最大数を設定します。指定されていない場合は、すべてのプールが返されます。

6.120. MEASURABLE

6.121. MOVEABLE

表6.371 メソッドの概要

Name	概要
move	

6.121.1. move POST

表6.372 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	移動を非同期で実行する必要があるかどうかを示します。

6.122. NETWORK

ネットワークを管理するサービス

表6.373 メソッドの概要

Name	概要
get	論理ネットワークを取得します。
remove	論理ネットワーク、または論理ネットワークのデータセンターへの関連付けを削除します。
update	論理ネットワークを更新します。

6.122.1. get GET

論理ネットワークを取得します。

以下に例を示します。

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

以下を応答します。

```
<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>
```

表6.374 パラメーターの概要

Name	タイプ	方向	概要
network	Network	Out	

6.122.2. remove DELETE

論理ネットワーク、または論理ネットワークのデータセンターへの関連付けを削除します。

たとえば、論理ネットワーク **123** を削除するには、以下のような要求を送信します。

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

各ネットワークは、1つのデータセンターに厳密にバインドされています。したがって、ネットワークとデータセンターの関連付けを解除すると、そのネットワークを単に削除した場合と同じ結果になります。ただし、データセンター **123** のネットワーク **456** を削除するという方がより具体的かもしれません。

たとえば、ネットワーク **456** とデータセンター **123** の関連付けを削除するには、以下のような要求を送信します。

```
DELETE /ovirt-engine/api/datacenters/123/networks/456
```

表6.375 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.122.3. update PUT

論理ネットワークを更新します。

name、**description**、**ip**、**vlan**、**stp**、および **display** 属性を更新できます。

たとえば、論理ネットワーク **123** の説明を更新するには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<network>
  <description>My updated description</description>
</network>
```

ネットワークの最大伝送単位は、**mtu** 属性の整数値を指定する PUT 要求を使用して設定されます。

たとえば、最大伝送単位を設定するには、以下のようなリクエストを送信します。

```
PUT /ovirt-engine/api/datacenters/123/networks/456
```

リクエスト本文は以下のようになります。

```
<network>
  <mtu>1500</mtu>
</network>
```

表6.376 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
network	Network	In/Out	

6.123. NETWORKATTACHMENT

表6.377 メソッドの概要

Name	概要
get	
remove	
update	

6.123.1. get GET

表6.378 パラメーターの概要

Name	タイプ	方向	概要
attachment	NetworkAttachment	Out	

6.123.2. remove DELETE

表6.379 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.123.3. update PUT

表6.380 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
attachment	NetworkAttachment	In/Out	

6.124. NETWORKATTACHMENTS

表6.381 メソッドの概要

Name	概要
add	
list	

6.124.1. add POST

表6.382 パラメーターの概要

Name	タイプ	方向	概要
attachment	NetworkAttachment	In/Out	

6.124.2. list GET

表6.383 パラメーターの概要

Name	タイプ	方向	概要
attachments	NetworkAttachment[]	Out	
max	Integer	In	返すアタッチメントの最大数を設定します。

6.124.2.1. max

返すアタッチメントの最大数を設定します。指定しない場合、すべてのアタッチメントが返されます。

6.125. NETWORKFILTER

ネットワークフィルターを管理します。

```
<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>
```

バージョンは、特定のフィルターの最小サポートバージョンを参照することに注意してください。

表6.384 メソッドの概要

Name	概要
get	ネットワークフィルターの表現を取得します。

6.125.1. get GET

ネットワークフィルターの表現を取得します。

表6.385 パラメーターの概要

Name	タイプ	方向	概要
network_filter	NetworkFilter	Out	

6.126. NETWORKFILTERS

読み取り専用のネットワークフィルターサブコレクションを表します。

ネットワークフィルターを使用すると、定義されたルールに従って、VM の nic との間で送受信されるパケットをフィルター処理できます。詳細は、[NetworkFilter](#) サービスドキュメントを参照してください。

ネットワークフィルターは、バージョン 3.0 以降、さまざまなバージョンでサポートされています。

ネットワークフィルターは、vnic プロファイルごとに定義されます。

vnic プロファイルは、特定のネットワークに対して定義されます。

ネットワークは、複数の異なるクラスターに割り当てることができます。将来的には、各ネットワークはクラスターレベルで定義されます。

現在、各ネットワークはデータセンターレベルで定義されています。各ネットワークの潜在的なネットワークフィルターは、ネットワークのデータセンター互換バージョン V によって決定されます。特定のネットワークに対してこのネットワークフィルターを設定するには、V がネットワークフィルターバージョン以上である必要があります。ネットワークフィルターをサポートするバージョンのクラスターにネットワークが割り当てられている場合、データセンターのバージョンがネットワークフィルターのバージョンより小さいため、フィルターを使用できない場合があることに注意してください。

特定のクラスターに対してサポートされているすべてのネットワークフィルターを一覧表示する例:

```
GET http://localhost:8080/ovirt-engine/api/clusters/{cluster:id}/networkfilters
```

出力:

```
<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>
```


表6.386 メソッドの概要

Name	概要
list	ネットワークフィルターの表現を取得します。

6.126.1. list GET

ネットワークフィルターの表現を取得します。

表6.387 パラメーターの概要

Name	タイプ	方向	概要
filters	NetworkFilter[]	Out	

6.127. NETWORKLABEL

表6.388 メソッドの概要

Name	概要
get	
remove	論理ネットワークからラベルを削除します。

6.127.1. get GET

表6.389 パラメーターの概要

Name	タイプ	方向	概要
label	NetworkLabel	Out	

6.127.2. remove DELETE

論理ネットワークからラベルを削除します。

たとえば、ID **123** を持つ論理ネットワークからラベル **exemplary** を削除するには、以下の要求を送信します。

```
DELETE /ovirt-engine/api/networks/123/labels/exemplary
```

表6.390 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.128. NETWORKLABELS

表6.391 メソッドの概要

Name	概要
add	論理ネットワークにラベルをアタッチします。
list	

6.128.1. add POST

論理ネットワークにラベルをアタッチします。

ラベルを論理ネットワークにアタッチして、その論理ネットワークと、同じラベルがアタッチされた物理ホストネットワークインターフェイスとの関連付けを自動化することができます。

たとえば、ラベル **mylabel** を ID **123** を持つ論理ネットワークにアタッチするには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<label id="mylabel"/>
```

表6.392 パラメーターの概要

Name	タイプ	方向	概要
label	NetworkLabel	In/Out	

6.128.2. list GET

表6.393 パラメーターの概要

Name	タイプ	方向	概要
labels	NetworkLabel[]	Out	
max	Integer	In	返すラベルの最大数を設定します。

6.128.2.1. max

返すラベルの最大数を設定します。指定されていない場合、すべてのラベルが返されます。

6.129. ネットワーク

論理ネットワークを管理します。

エンジンは、インストール時にデフォルトの **ovirtmgmt** ネットワークを作成します。このネットワークは、ハイパーバイザーホストにアクセスするための管理ネットワークとして機能します。このネットワークは **Default** クラスタに関連付けられており、**Default** データセンターのメンバーです。

表6.394 メソッドの概要

Name	概要
add	新しい論理ネットワークを作成するか、既存のネットワークをデータセンターに関連付けます。
list	論理ネットワークを一覧表示します。

6.129.1. add POST

新しい論理ネットワークを作成するか、既存のネットワークをデータセンターに関連付けます。

新しいネットワークの作成には、**name** 要素と **data_center** 要素が必要です。

たとえば、データセンター **123** 用に **mynetwork** という名前のネットワークを作成するには、以下のような要求を送信します。

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

リクエスト本文は以下のようになります。

```
<network>
  <name>mynetwork</name>
  <data_center id="123"/>
</network>
```

既存のネットワーク **456** をデータセンター **123** に関連付けるには、以下のような要求を送信します。

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

リクエスト本文は以下のようになります。

```
<network>
  <name>ovirtmgmt</name>
</network>
```

表6.395 パラメーターの概要

Name	タイプ	方向	概要
network	Network	In/Out	

6.129.2. list GET

論理ネットワークを一覧表示します。

以下に例を示します。

```
GET /ovirt-engine/api/networks
```

以下を応答します。

```
<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>
```

表6.396 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
max	Integer	In	返すネットワークの最大数を設定します。
networks	Network[]	Out	
search	String	In	返されたホストを制限するために使用されるクエリ文字列。

6.129.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.129.2.2. max

返すネットワークの最大数を設定します。指定されていない場合には、すべてのネットワークが返されます。

6.130. OPENSTACKIMAGE

表6.397 メソッドの概要

Name	概要
get	
import	Glance イメージストレージドメインから仮想マシンをインポートします。

6.130.1. get GET

表6.398 パラメーターの概要

Name	タイプ	方向	概要
image	Openstackimage	Out	

6.130.2. import POST

Glance イメージストレージドメインから仮想マシンをインポートします。

たとえば、識別子 **123** のストレージドメインから識別子 **456** のイメージをインポートするには、以下のような要求を送信します。

```
POST /ovirt-engine/api/openstackimageproviders/123/images/456/import
```

リクエスト本文は以下のようになります。

```
<action>
  <storage_domain>
    <name>images0</name>
  </storage_domain>
  <cluster>
    <name>images0</name>
  </cluster>
</action>
```

表6.399 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	インポートを非同期で実行する必要があるかどうかを示します。

Name	タイプ	方向	概要
cluster	Cluster	In	このパラメーターは、 import_as_template を使用する場合は必須であり、テンプレートとして glance イメージをインポートする際に使用するクラスターを示します。
disk	ディスク	In	
import_as_template	ブール値	In	イメージをテンプレートとしてインポートする必要があるかどうかを示します。
storage_domain	StorageDomain	In	
template	Template	In	

6.131. OPENSTACKIMAGEPROVIDER

表6.400 メソッドの概要

Name	概要
get	
importcertificates	
remove	
testconnectivity	
update	

6.131.1. get GET

表6.401 パラメーターの概要

Name	タイプ	方向	概要
provider	OpenStackImageProvider	Out	

6.131.2. importcertificates POST

表6.402 パラメーターの概要

Name	タイプ	方向	概要
certificates	Certificate[]	In	

6.131.3. remove DELETE

表6.403 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.131.4. testconnectivity POST

表6.404 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	テストを非同期で実行する必要があるかどうかを示します。

6.131.5. update PUT

表6.405 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
provider	OpenStackImageProvider	In/Out	

6.132. OPENSTACKIMAGEPROVIDERS

表6.406 メソッドの概要

Name	概要
add	
list	

6.132.1. add POST

表6.407 パラメーターの概要

Name	タイプ	方向	概要
provider	OpenStackImageProvider	In/Out	

6.132.2. list GET

表6.408 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロバイダーの最大数を設定します。
providers	OpenStackImageProvider[]	Out	

6.132.2.1. max

返すプロバイダーの最大数を設定します。指定しない場合は、すべてのプロバイダーが返されます。

6.133. OPENSTACKIMAGES

表6.409 メソッドの概要

Name	概要
list	Glance イメージストレージドメインのイメージを一覧表示します。

6.133.1. list GET

Glance イメージストレージドメインのイメージを一覧表示します。

表6.410 パラメーターの概要

Name	タイプ	方向	概要
images	OpenStackImage[]	Out	
max	Integer	In	返すイメージの最大数を設定します。

6.133.1.1. max

返すイメージの最大数を設定します。指定しない場合、すべてのイメージが返されます。

6.134. OPENSTACKNETWORK

表6.411 メソッドの概要

Name	概要
get	
import	この操作は、外部ネットワークを oVirt にインポートします。

6.134.1. get GET

表6.412 パラメーターの概要

Name	タイプ	方向	概要
network	OpenStackNetwork	Out	

6.134.2. import POST

この操作は、外部ネットワークを oVirt にインポートします。指定したデータセンターにネットワークが追加されます。

表6.413 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	インポートを非同期で実行する必要があるかどうかを示します。
data_center	DataCenter	In	ネットワークのインポート先のデータセンター。

6.134.2.1. data_center

ネットワークのインポート先のデータセンター。データセンターは必須であり、**id** または **name** 属性を使用して指定できます。残りの属性は無視されます。

6.135. OPENSTACKNETWORKPROVIDER

このサービスは、OpenStack ネットワークプロバイダーを管理します。

表6.414 メソッドの概要

Name	概要
get	このサービスによって管理されるオブジェクトの表現を返します。

Name	概要
importcertificates	
remove	プロバイダーを削除します。
testconnectivity	
update	プロバイダーを更新します。

6.135.1. get GET

このサービスによって管理されるオブジェクトの表現を返します。

たとえば、識別子が **1234** の OpenStack ネットワークプロバイダーを取得するには、以下のようなリクエストを送信します。

```
GET /ovirt-engine/api/openstacknetworkproviders/1234
```

表6.415 パラメーターの概要

Name	タイプ	方向	概要
provider	OpenStackNetworkProvider	Out	

6.135.2. importcertificates POST

表6.416 パラメーターの概要

Name	タイプ	方向	概要
certificates	Certificate[]	In	

6.135.3. remove DELETE

プロバイダーを削除します。

たとえば、識別子が **1234** の OpenStack ネットワークプロバイダーを削除するには、以下のようなリクエストを送信します。

```
DELETE /ovirt-engine/api/openstacknetworkproviders/1234
```

表6.417 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.135.4. testconnectivity POST

表6.418 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	テストを非同期で実行する必要があるかどうかを示します。

6.135.5. update PUT

プロバイダーを更新します。

たとえば、識別子 **1234** を持つ OpenStack ネットワークプロバイダーの **provider_name**、**requires_authentication**、**url**、**tenant_name**、および **type** プロパティを更新するには、以下のようなリクエストを送信します。

```
PUT /ovirt-engine/api/openstacknetworkproviders/1234
```

リクエスト本文は以下のようになります。

```
<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>
```

表6.419 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
provider	OpenStackNetworkProvider	In/Out	更新するプロバイダー。

6.136. OPENSTACKNETWORKPROVIDERS

このサービスは、OpenStack ネットワークプロバイダーを管理します。

表6.420 メソッドの概要

Name	概要
add	この操作により、新しいネットワークプロバイダーがシステムに追加されます。
list	

6.136.1. add POST

この操作により、新しいネットワークプロバイダーがシステムに追加されます。**type** プロパティーが存在しない場合、デフォルト値の **NEUTRON** が使用されます。

表6.421 パラメーターの概要

Name	タイプ	方向	概要
provider	OpenStackNetworkProvider	In/Out	

6.136.2. list GET

表6.422 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロバイダーの最大数を設定します。
providers	OpenStackNetworkProvider[]	Out	

6.136.2.1. max

返すプロバイダーの最大数を設定します。指定しない場合は、すべてのプロバイダーが返されます。

6.137. OPENSTACKNETWORKS

表6.423 メソッドの概要

Name	概要
list	

6.137.1. list GET

表6.424 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すネットワークの最大数を設定します。
networks	OpenStackNetwork[]	Out	

6.137.1.1. max

返すネットワークの最大数を設定します。指定されていない場合には、すべてのネットワークが返されます。

6.138. OPENSTACKSUBNET

表6.425 メソッドの概要

Name	概要
get	
remove	

6.138.1. get GET

表6.426 パラメーターの概要

Name	タイプ	方向	概要
subnet	OpenStackSubnet	Out	

6.138.2. remove DELETE

表6.427 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.139. OPENSTACKSUBNETS

表6.428 メソッドの概要

Name	概要
add	

Name	概要
list	

6.139.1. add POST

表6.429 パラメーターの概要

Name	タイプ	方向	概要
subnet	OpenStackS ubnet	In/Out	

6.139.2. list GET

表6.430 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すサブネットワークの最大数を設定します。
subnets	OpenStackS ubnet[]	Out	

6.139.2.1. max

返すサブネットワークの最大数を設定します。指定されていない場合には、すべてのサブネットワークが返されます。

6.140. OPENSTACKVOLUMEAUTHENTICATIONKEY

表6.431 メソッドの概要

Name	概要
get	
remove	
update	

6.140.1. get GET

表6.432 パラメーターの概要

Name	タイプ	方向	概要
key	OpenstackVolumeAuthenticationKey	Out	

6.140.2. remove DELETE

表6.433 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.140.3. update PUT

表6.434 パラメーターの概要

Name	タイプ	方向	概要
key	OpenstackVolumeAuthenticationKey	In/Out	

6.141. OPENSTACKVOLUMEAUTHENTICATIONKEYS

表6.435 メソッドの概要

Name	概要
add	
list	

6.141.1. add POST

表6.436 パラメーターの概要

Name	タイプ	方向	概要
key	OpenstackVolumeAuthenticationKey	In/Out	

6.141.2. list GET

表6.437 パラメーターの概要

Name	タイプ	方向	概要
keys	OpenstackV olumeAuthe nticationKey[]	Out	
max	Integer	In	返すことのできるキーの最大数。

6.141.2.1. max

返すことのできるキーの最大数。指定されていない場合は、すべてのキーが返されます。

6.142. OPENSTACKVOLUMEPROVIDER

表6.438 メソッドの概要

Name	概要
get	
importcertificates	
remove	
testconnectivity	
update	

6.142.1. get GET

表6.439 パラメーターの概要

Name	タイプ	方向	概要
provider	OpenStackV olumeProvid er	Out	

6.142.2. importcertificates POST

表6.440 パラメーターの概要

Name	タイプ	方向	概要
certificates	Certificate[]	In	

6.142.3. remove DELETE

表6.441 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.142.4. testconnectivity POST

表6.442 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	テストを非同期で実行する必要があるかどうかを示します。

6.142.5. update PUT

表6.443 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
provider	OpenStackVolumeProvider	In/Out	

6.143. OPENSTACKVOLUMEPROVIDERS

表6.444 メソッドの概要

Name	概要
add	新しいボリュームプロバイダーを追加します。
list	ボリュームプロバイダーのリストを取得します。

6.143.1. add POST

新しいボリュームプロバイダーを追加します。

以下に例を示します。

```
POST /ovirt-engine/api/openstackvolumeproviders
```

リクエスト本文は以下のようになります。

```
<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>
```

表6.445 パラメーターの概要

Name	タイプ	方向	概要
provider	OpenStackVolumeProvider	In/Out	

6.143.2. list GET

ボリュームプロバイダーのリストを取得します。

表6.446 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロバイダーの最大数を設定します。
providers	OpenStackVolumeProvider[]	Out	

6.143.2.1. max

返すプロバイダーの最大数を設定します。指定しない場合は、すべてのプロバイダーが返されます。

6.144. OPENSTACKVOLUMETYPE

表6.447 メソッドの概要

Name	概要
get	

6.144.1. get GET

表6.448 パラメーターの概要

Name	タイプ	方向	概要
type	OpenStackV olumeType	Out	

6.145. OPENSTACKVOLUMETYPES

表6.449 メソッドの概要

Name	概要
list	

6.145.1. list GET

表6.450 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すボリュームタイプの最大数を設定します。
types	OpenStackV olumeType[]	Out	

6.145.1.1. max

返すボリュームタイプの最大数を設定します。指定されていない場合は、すべてのボリュームタイプが返されます。

6.146. OPERATINGSYSTEM

表6.451 メソッドの概要

Name	概要
get	

6.146.1. get GET

表6.452 パラメーターの概要

Name	タイプ	方向	概要
operating_system	OperatingSystemInfo	Out	

6.147. OPERATINGSYSTEMS

表6.453 メソッドの概要

Name	概要
list	

6.147.1. list GET

表6.454 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すネットワークの最大数を設定します。
operating_system	OperatingSystemInfo[]	Out	

6.147.1.1. max

返すネットワークの最大数を設定します。指定されていない場合には、すべてのネットワークが返されます。

6.148. パーミッション

表6.455 メソッドの概要

Name	概要
get	
remove	

6.148.1. get GET

表6.456 パラメーターの概要

Name	タイプ	方向	概要
permission	パーミッション	Out	

6.148.2. remove DELETE

表6.457 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.149. PERMIT

ロールの特定の permit を管理するサービス。

表6.458 メソッドの概要

Name	概要
get	ロールの permit に関する情報を取得します。
remove	ロールから permit を削除します。

6.149.1. get GET

ロールの permit に関する情報を取得します。

たとえば、ID **123** のロールの ID **456** の permit に関する情報を取得するには、以下のようなリクエストを送信します。

```
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>
```

表6.459 パラメーターの概要

Name	タイプ	方向	概要
permit	Permit	Out	ロールの permit。

6.149.2. remove DELETE

ロールから permit を削除します。

たとえば、ID **123** のロールから ID **456** の permit を削除するには、以下のようなリクエストを送信します。

```
DELETE /ovirt-engine/api/roles/123/permits/456
```

表6.460 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.150. PERMITS

特定のロールの permits サブコレクションを表します。

表6.461 メソッドの概要

Name	概要
add	ロールに permit を追加します。
list	ロールの permits を一覧表示します。

6.150.1. add POST

ロールに permit を追加します。permit 名は、「[ClusterLevels](#)」サービスから取得できます。

たとえば、ID **123** のロールに permit **create_vm** を割り当てるには、以下のようなリクエストを送信します。

```
POST /ovirt-engine/api/roles/123/permits
```

リクエスト本文は以下のようになります。

```
<permit>
  <name>create_vm</name>
</permit>
```

表6.462 パラメーターの概要

Name	タイプ	方向	概要
permit	Permit	In/Out	追加する permit。

6.150.2. list GET

ロールの permits を一覧表示します。

たとえば、ID **123** のロールの permits を一覧表示するには、以下のようなリクエストを送信します。

```
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>
```

表6.463 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す permits の最大数を設定します。
permits	Permit[]	Out	permits のリスト。

6.150.2.1. max

返す permits の最大数を設定します。指定されていない場合は、すべての permits が返されます。

6.151. QOS

表6.464 メソッドの概要

Name	概要
get	
remove	
update	

6.151.1. get GET

表6.465 パラメーターの概要

Name	タイプ	方向	概要
qos	Qos	Out	

6.151.2. remove DELETE

表6.466 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.151.3. update PUT

表6.467 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
qos	Qos	In/Out	

6.152. QOSS

表6.468 メソッドの概要

Name	概要
add	
list	

6.152.1. add POST

表6.469 パラメーターの概要

Name	タイプ	方向	概要
qos	Qos	In/Out	

6.152.2. list GET

表6.470 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す QoS 記述子の最大数を設定します。
qoss	qos[]	Out	

6.152.2.1. max

返す QoS 記述子の最大数を設定します。指定されていない場合は、すべての記述子が返されます。

6.153. クォータ

表6.471 メソッドの概要

Name	概要
get	クォータを取得します。
remove	クォータを削除します。
update	クォータを更新します。

6.153.1. get GET

クォータを取得します。

クォータを取得する例:

```
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>
```

表6.472 パラメーターの概要

Name	タイプ	方向	概要
quota	クォータ	Out	

6.153.2. remove DELETE

クォータを削除します。

クォータを削除する例:

```
DELETE /ovirt-engine/api/datacenters/123-456/quotas/654-321-0472718ab224 HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

表6.473 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.153.3. update PUT

クォータを更新します。

クォータを更新する例:

```
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>
```

表6.474 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
quota	クォータ	In/Out	

6.154. QUOTA_CLUSTERLIMIT

表6.475 メソッドの概要

Name	概要
get	
remove	

6.154.1. get GET

表6.476 パラメーターの概要

Name	タイプ	方向	概要
limit	QuotaClusterLimit	Out	

6.154.2. remove DELETE

表6.477 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.155. QUOTA_CLUSTER_LIMITS

表6.478 メソッドの概要

Name	概要
add	
list	

6.155.1. add POST

表6.479 パラメーターの概要

Name	タイプ	方向	概要
limit	QuotaClusterLimit	In/Out	

6.155.2. list GET

表6.480 パラメーターの概要

Name	タイプ	方向	概要
limits	QuotaClusterLimit[]	Out	
max	Integer	In	返す制限の最大数を設定します。

6.155.2.1. max

返す制限の最大数を設定します。指定されていない場合は、すべての制限が返されます。

6.156. QUOTASTORAGELIMIT

表6.481 メソッドの概要

Name	概要
get	
remove	

6.156.1. get GET

表6.482 パラメーターの概要

Name	タイプ	方向	概要
limit	QuotaStorageLimit	Out	

6.156.2. remove DELETE

表6.483 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。

6.157. QUOTASTORAGELIMITS

表6.484 メソッドの概要

Name	概要
add	
list	

6.157.1. add POST

表6.485 パラメーターの概要

Name	タイプ	方向	概要
limit	QuotaStorageLimit	In/Out	

6.157.2. list GET

表6.486 パラメーターの概要

Name	タイプ	方向	概要
limits	QuotaStorageLimit[]	Out	
max	Integer	In	返す制限の最大数を設定します。

6.157.2.1. max

返す制限の最大数を設定します。指定されていない場合は、すべての制限が返されます。

6.158. QUOTAS

表6.487 メソッドの概要

Name	概要
add	新しいクォータを作成します。
list	データセンターのクォータを一覧表示します。

6.158.1. add POST

新しいクォータを作成します。

新しいクォータを作成する例:

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

```
<quota>
  <name>myquota</name>
  <description>My new quota for virtual machines</description>
</quota>
```

表6.488 パラメーターの概要

Name	タイプ	方向	概要
quota	クォータ	In/Out	

6.158.2. list GET

データセンターのクォータを一覧表示します。

表6.489 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すクォータ記述子の最大数を設定します。
quotas	Quota[]	Out	

6.158.2.1. max

返すクォータ記述子の最大数を設定します。指定されていない場合は、すべての記述子が返されます。

6.159. ロール

表6.490 メソッドの概要

Name	概要
get	ロールを取得します。
remove	ロールを削除します。
update	ロールを更新します。

6.159.1. get GET

ロールを取得します。

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

以下のような XML 応答を受け取ります。

```
<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>
```

表6.491 パラメーターの概要

Name	タイプ	方向	概要
role	ロール	Out	ロールを取得しました。

6.159.2. remove DELETE

ロールを削除します。

ロールを削除するには、その ID を知る必要があります。以下のようにリクエストを送信します。

```
DELETE /ovirt-engine/api/roles/{role_id}
```

表6.492 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.159.3. update PUT

ロールを更新します。ロールの作成後に、**name**、**description**、および **administrative** 属性を更新できます。このエンドポイント内では、ロールの permits を管理する [サービス](#) を使用するために、必要なロールの permits を追加または削除することはできません。

たとえば、ロールの **name**、**description**、および **administrative** 属性を更新するには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<role>
  <name>MyNewRoleName</name>
  <description>My new description of the role</description>
  <administrative>true</administrative>
</group>
```

表6.493 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
role	ロール	In/Out	ロールを更新しました。

6.160. ロール

ロールのグローバルセットへの読み取り専用アクセスを提供します。

表6.494 メソッドの概要

Name	概要
add	新規ロールを作成します。
list	ロールを一覧表示します。

6.160.1. add POST

新規ロールを作成します。ロールは、管理者または非管理者であることができ、異なる permits を持つことができます。

たとえば、ログインして仮想マシンを作成する permits を持つ **MyRole** 非管理ロールを追加するには、以下のようなリクエストを送信します (permit ID を渡す必要があることに注意してください)。

```
POST /ovirt-engine/api/roles
```

リクエスト本文は以下のようになります。

```
<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>
```

表6.495 パラメーターの概要

Name	タイプ	方向	概要
role	ロール	In/Out	追加されるロール。

6.160.2. list GET

ロールを一覧表示します。

```
GET /ovirt-engine/api/roles
```

以下のような XML で応答を受け取ります。

```
<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>
```

表6.496 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すロールの最大数を設定します。

Name	タイプ	方向	概要
roles	Role[]	Out	ロールの一覧を取得しました。

6.160.2.1. max

返すロールの最大数を設定します。指定のない場合は、すべてのロールが返されます。

6.161. SCHEDULINGPOLICIES

表6.497 メソッドの概要

Name	概要
add	
list	

6.161.1. add POST

表6.498 パラメーターの概要

Name	タイプ	方向	概要
policy	SchedulingPolicy	In/Out	

6.161.2. list GET

表6.499 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返すポリシーの最大数を設定します。
policies	SchedulingPolicy[]	Out	

6.161.2.1. max

返すポリシーの最大数を設定します。指定されていない場合は、すべてのポリシーが返されます。

6.162. SCHEDULINGPOLICY

表6.500 メソッドの概要

Name	概要
get	
remove	
update	

6.162.1. get GET

表6.501 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルタする必要があるかどうかを示します。
policy	SchedulingPolicy	Out	

6.162.2. remove DELETE

表6.502 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.162.3. update PUT

表6.503 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
policy	SchedulingPolicy	In/Out	

6.163. SCHEDULINGPOLICYUNIT

表6.504 メソッドの概要

Name	概要
get	
remove	

6.163.1. get GET

表6.505 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルタする必要があるかどうかを示します。
unit	SchedulingPolicyUnit	Out	

6.163.2. remove DELETE

表6.506 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.164. SCHEDULINGPOLICYUNITS

表6.507 メソッドの概要

Name	概要
list	

6.164.1. list GET

表6.508 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルタする必要があるかどうかを示します。
max	Integer	In	返すポリシーユニットの最大数を設定します。

Name	タイプ	方向	概要
units	SchedulingPolicyUnit[]	Out	

6.164.1.1. max

返すポリシーユニットの最大数を設定します。指定されていない場合は、すべてのポリシーユニットが返されます。

6.165. スナップショット

表6.509 メソッドの概要

Name	概要
get	
remove	
restore	仮想マシンのスナップショットを復元します。

6.165.1. get GET

表6.510 パラメーターの概要

Name	タイプ	方向	概要
snapshot	スナップショット	Out	

6.165.2. remove DELETE

表6.511 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.165.3. restore POST

仮想マシンのスナップショットを復元します。

たとえば、識別子が **123** の仮想マシンの識別子が **456** のスナップショットを復元するには、以下のようなリクエストを送信します。

POST /ovirt-engine/api/vms/123/snapshots/456/restore

ボディに空の **action** がある場合:

<action/>

表6.512 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	リセットを非同期で実行する必要があるかどうかを示します。
disks	Disk[]	In	
restore_memory	ブール値	In	

6.166. SNAPSHOTCDROM

表6.513 メソッドの概要

Name	概要
get	

6.166.1. get GET

表6.514 パラメーターの概要

Name	タイプ	方向	概要
cdrom	Cdrom	Out	

6.167. SNAPSHOTCDROMS

表6.515 メソッドの概要

Name	概要
list	

6.167.1. list GET

表6.516 パラメーターの概要

Name	タイプ	方向	概要
cdroms	Cdrom[]	Out	
max	Integer	In	返す CDROM の最大数を設定します。

6.167.1.1. max

返す CDROM の最大数を設定します。指定されていない場合は、すべての CDROM が返されます。

6.168. SNAPSHOTDISK

表6.517 メソッドの概要

Name	概要
get	

6.168.1. get GET

表6.518 パラメーターの概要

Name	タイプ	方向	概要
disk	ディスク	Out	

6.169. SNAPSHOTDISKS

表6.519 メソッドの概要

Name	概要
list	

6.169.1. list GET

表6.520 パラメーターの概要

Name	タイプ	方向	概要
disks	Disk[]	Out	
max	Integer	In	返すディスクの最大数を設定します。

6.169.1.1. max

返すディスクの最大数を設定します。指定しない場合、すべてのディスクが返されます。

6.170. SNAPSHOTNIC

表6.521 メソッドの概要

Name	概要
get	

6.170.1. get GET

表6.522 パラメーターの概要

Name	タイプ	方向	概要
nic	Nic	Out	

6.171. SNAPSHOTNICS

表6.523 メソッドの概要

Name	概要
list	

6.171.1. list GET

表6.524 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す NIC の最大数を設定します。
nics	Nic[]	Out	

6.171.1.1. max

返す NIC の最大数を設定します。指定されていない場合は、すべての NIC が返されます。

6.172. スナップショット

表6.525 メソッドの概要

Name	概要
add	仮想マシンのスナップショットを作成します。
list	

6.172.1. add POST

仮想マシンのスナップショットを作成します。

たとえば、仮想マシン **123** の新しいスナップショットを作成するには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<snapshot>
  <description>My snapshot</description>
</snapshot>
```

重要

スナップショットが作成されると、`persistence_memorystate` 属性のデフォルト値は **true** になります。これは、仮想マシンのメモリーの内容がスナップショットに含まれることを意味し、仮想マシンが長時間一時停止されることも意味します。これは、時間的な制約が厳しいアプリケーション (NTP サーバーなど) に悪影響を与える可能性があります。そのような場合は、属性を **false** に設定してください。

```
<snapshot>
  <description>My snapshot</description>
  <persist_memorystate>>false</persist_memorystate>
</snapshot>
```

表6.526 パラメーターの概要

Name	タイプ	方向	概要
snapshot	スナップ ショット	In/Out	

6.172.2. list GET

表6.527 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すスナップショットの最大数を設定します。

Name	タイプ	方向	概要
snapshots	Snapshot[]	Out	

6.172.2.1. max

返すスナップショットの最大数を設定します。指定されていない場合は、すべてのスナップショットが返されます。

6.173. SSHPUBLICKEY

表6.528 メソッドの概要

Name	概要
get	
remove	
update	

6.173.1. get GET

表6.529 パラメーターの概要

Name	タイプ	方向	概要
key	SshPublicKey	Out	

6.173.2. remove DELETE

表6.530 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.173.3. update PUT

表6.531 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。

Name	タイプ	方向	概要
key	SshPublicKey	In/Out	

6.174. SSHPUBLICKEYS

表6.532 メソッドの概要

Name	概要
add	
list	

6.174.1. add POST

表6.533 パラメーターの概要

Name	タイプ	方向	概要
key	SshPublicKey	In/Out	

6.174.2. list GET

表6.534 パラメーターの概要

Name	タイプ	方向	概要
keys	SshPublicKey[]	Out	
max	Integer	In	返すことのできるキーの最大数。

6.174.2.1. max

返すことのできるキーの最大数。指定されていない場合は、すべてのキーが返されます。

6.175. 統計

表6.535 メソッドの概要

Name	概要
get	

6.175.1. get GET

表6.536 パラメーターの概要

Name	タイプ	方向	概要
statistic	統計	Out	

6.176. 統計

表6.537 メソッドの概要

Name	概要
list	統計のリストを取得します。

6.176.1. list GET

統計のリストを取得します。

たとえば、仮想マシン **123** の統計を取得するには、以下のようなリクエストを送信します。

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

結果は以下のようになります。

```
<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>
```

URI の末尾に ID を指定することで、統計の一部のみを取得できます。つまり、以下のようになります。

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

出力:

```
<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>

```

表6.538 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す統計の最大数を設定します。
statistics	Statistic[]	Out	

6.176.1.1. max

返す統計の最大数を設定します。指定されていない場合は、すべての統計が返されます。

6.177. STEP

ステップを管理するサービス。

表6.539 メソッドの概要

Name	概要
end	外部ステップの実行を終了としてマークします。
get	ステップを取得します。

6.177.1. end POST

外部ステップの実行を終了としてマークします。

たとえば、識別子 **123** の **job** に属する識別子 **456** のステップを終了するには、以下のリクエストを送信します。

```
POST /ovirt-engine/api/jobs/123/steps/456/end
```

リクエスト本文は、以下のようになります。

```

<action>
  <force>true</force>
  <succeeded>true</succeeded>

```

```
</action>
```

表6.540 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。
force	ブール値	In	ステップを強制終了する必要があるかどうかを示します。
succeeded	ブール値	In	ステップを正常終了または失敗としてマークする必要があるかどうかを示します。

6.177.1.1. succeeded

ステップを正常終了または失敗としてマークする必要があるかどうかを示します。

このパラメーターはオプションであり、デフォルト値は **true** です。

6.177.2. get GET

ステップを取得します。

```
GET /ovirt-engine/api/jobs/123/steps/456
```

以下のような XML で応答を受け取ります。

```
<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>
```

表6.541 パラメーターの概要

Name	タイプ	方向	概要
step	Step	Out	ステップの表現を取得します。

6.178. STEPS

ステップを管理するサービスです。

表6.542 メソッドの概要

Name	概要
add	外部ステップを既存のジョブまたは既存のステップに追加します。
list	ステップの表現を取得します。

6.178.1. add POST

外部ステップを既存のジョブまたは既存のステップに追加します。

たとえば、識別子が **123** の **job** にステップを追加するには、以下のリクエストを送信します。

```
POST /ovirt-engine/api/jobs/123/steps
```

リクエスト本文は、以下のようになります。

```
<step>
  <description>Validating</description>
  <start_time>2016-12-12T23:07:26.605+02:00</start_time>
  <status>started</status>
  <type>validating</type>
</step>
```

応答は以下のようになります。

```
<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>
```

表6.543 パラメーターの概要

Name	タイプ	方向	概要
step	Step	In/Out	追加されるステップ。

6.178.2. list GET

ステップの表現を取得します。

```
GET /ovirt-engine/api/job/123/steps
```

以下のような XML で応答を受け取ります。

```
<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>
  ...
</steps>
```

表6.544 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すステップの最大数を設定します。
steps	step[]	Out	ステップの表現。

6.178.2.1. max

返すステップの最大数を設定します。指定されていない場合は、すべてのステップが返されます。

6.179. ストレージ

表6.545 メソッドの概要

Name	概要
get	

6.179.1. get GET

表6.546 パラメーターの概要

Name	タイプ	方向	概要
report_status	ブール値	In	ストレージ内の LUN のステータスを確認する必要があるかどうかを示します。
storage	HostStorage	Out	

6.179.1.1. report_status

ストレージ内の LUN のステータスを確認する必要があるかどうかを示します。LUN のステータスの確認は非常に重要な操作であり、このデータは必ずしもユーザーが必要とするものではありません。このパラメーターは、LUN のステータスチェックを実行しないオプションを提供します。

デフォルトは、後方互換性を確保するために **true** です。

LUN ステータスの例を次に示します。

```
<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>
      <status>used</status>
      <vendor_id>LIO-ORG</vendor_id>
      <volume_group_id>O9Du7I-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>
```

これは、LUN ステータスのない例です。

```
<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>O9Du7I-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. STORAGEDOMAIN

表6.547 メソッドの概要

Name	概要
get	
isattached	
reduceluns	この操作により、ストレージドメインから論理ユニットが削減されます。
refreshluns	この操作により、LUN サイズが更新されます。
remove	ストレージドメインを削除します。
update	ストレージドメインを更新します。
updateovfstore	この操作により、このストレージドメインの OVF_STORE が強制的に更新されます。

6.180.1. get GET

表6.548 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
storage_domain	StorageDomain	Out	

6.180.2. isattached POST

表6.549 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。
host	ホスト	In	
is_attached	ブール値	Out	

6.180.3. reduceluns POST

この操作により、ストレージドメインから論理ユニットが削減されます。

そのために、提供された論理ユニットに保存されているデータは、ストレージドメインの他の論理ユニットに移動され、その後、ストレージドメインから削減されます。

たとえば、ストレージドメインから2つの論理ユニットを削減するには、次のようなリクエストを送信します。

```
POST /ovirt-engine/api/storagedomains/123/reduceluns
```

リクエスト本文は以下のようになります。

```
<action>
  <logical_units>
    <logical_unit id="1IET_00010001"/>
    <logical_unit id="1IET_00010002"/>
  </logical_units>
</action>
```

表6.550 パラメーターの概要

Name	タイプ	方向	概要
logical_units	LogicalUnit[]	In	ストレージドメインから削減する必要がある論理ユニット。

6.180.4. refreshluns POST

この操作により、LUN サイズが更新されます。

ストレージサーバーの基盤となる LUN のサイズを増やした後、ユーザーは LUN サイズを更新できます。このアクションは、提供された LUN の再スキャンを強制し、必要に応じて新しいサイズでデータベースを更新します。

たとえば、2つの LUN のサイズを更新するには、以下のようなリクエストを送信します。

```
POST /ovirt-engine/api/storagedomains/262b056b-aede-40f1-9666-b883eff59d40/refreshluns
```

リクエスト本文は以下のようになります。

```
<action>
  <logical_units>
    <logical_unit id="1IET_00010001"/>
    <logical_unit id="1IET_00010002"/>
  </logical_units>
</action>
```

表6.551 パラメーターの概要

Name	タイプ	方向	概要
------	-----	----	----

Name	タイプ	方向	概要
async	ブール値	In	リフレッシュを非同期で実行する必要があるかどうかを示します。
logical_units	LogicalUnit[]	In	更新が必要な LUN。

6.180.5. remove DELETE

ストレージドメインを削除します。

特別なパラメーターがない場合、ストレージドメインはシステムから切り離され、データベースから削除されます。ストレージドメインは、同じまたは異なるセットアップに、すべてのデータを使用してインポートできます。ストレージにアクセスできない場合、操作は失敗します。

destroy パラメーターが **true** の場合、ストレージにアクセスできない場合でも操作は常に成功し、失敗は無視され、ストレージドメインはデータベースから削除されます。

format パラメーターが **true** の場合、実際のストレージがフォーマットされ、メタデータが LUN またはディレクトリーから削除されるため、同じセットアップまたは別のセットアップにインポートできなくなります。

表6.552 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。
destroy	ブール値	In	ストレージにアクセスできない場合でも、操作が成功し、ストレージドメインがデータベースから削除されるかどうかを示します。
format	ブール値	In	<p>実際のストレージをフォーマットし、基盤となる LUN またはディレクトリーからすべてのメタデータを削除するかどうかを示します。</p> <p>[source] ---- DELETE /ovirt-engine/api/storagedomains/123?format=true ----</p> <p>このパラメーターはオプションであり、デフォルト値は false です。</p>
host	String	In	ストレージドメインの削除に使用するホストを示します。

6.180.5.1. destroy

ストレージにアクセスできない場合でも、操作が成功し、ストレージドメインがデータベースから削除されるかどうかを示します。

```
DELETE /ovirt-engine/api/storagedomains/123?destroy=true
```

このパラメーターはオプションであり、デフォルト値は **false** です。

6.180.5.2. host

ストレージドメインの削除に使用するホストを示します。

このパラメーターは必須であり、ホストの名前または識別子を含めることができます。たとえば、**myhost** という名前のホストを使用して、識別子 **123** のストレージドメインを削除するには、以下のようなリクエストを送信します。

```
DELETE /ovirt-engine/api/storagedomains/123?host=myhost
```

6.180.6. update PUT

ストレージドメインを更新します。

`StorageDomain` の属性すべてが作成後に更新可能である訳ではありません。更新できるものは、**name**、**description**、**comment**、**warning_low_space_indicator**、**critical_space_action_blocker**、および **wipe_after_delete** です (**wipe_after_delete** 属性を変更しても、すでに存在するディスクの wipe after delete プロパティは変更されません)。

識別子 **123** を持つストレージドメインの **name** 属性と **wipe_after_delete** 属性を更新するには、以下のようにリクエストを送信する。

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

リクエスト本文の場合は、以下ようになります。

```
<storage_domain>
  <name>data2</name>
  <wipe_after_delete>true</wipe_after_delete>
</storage_domain>
```

表6.553 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
storage_domain	<code>StorageDomain</code>	In/Out	

6.180.7. updateovfstore POST

この操作により、このストレージドメインの **OVF_STORE** が強制的に更新されます。

OVF_STORE は、ストレージドメインにある仮想マシンとディスクのメタデータを含むディスクイメージです。このメタデータは、ドメインが別のデータセンターまたは別のインストールとの間でインポートまたはエクスポートされる場合に使用されます。

デフォルトでは、**OVF_STORE** は定期的に更新されますが (デフォルトでは 60 分に設定)、重要な変更後や **OVF_STORE** が破損していると思われる場合に、ユーザーは更新を強制する必要があることもあります。

ユーザーによって開始された場合、**OVF_STORE** 更新は、更新の必要性の有無にかかわらず実行されます。

表6.554 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	OVF_STORE 更新を非同期的に実行するかどうかを示します。

6.181. STORAGEDOMAINCONTENTDISK

表6.555 メソッドの概要

Name	概要
get	

6.181.1. get GET

表6.556 パラメーターの概要

Name	タイプ	方向	概要
disk	ディスク	Out	
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

6.182. STORAGEDOMAINCONTENTDISKS

表6.557 メソッドの概要

Name	概要
list	

6.182.1. list GET

表6.558 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
disks	Disk[]	Out	
max	Integer	In	返すディスクの最大数を設定します。
search	String	In	返されたディスクを制限するために使用されるクエリ文字列。

6.182.1.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.182.1.2. max

返すディスクの最大数を設定します。指定しない場合、すべてのディスクが返されます。

6.183. STORAGEDOMAINSERVERCONNECTION

表6.559 メソッドの概要

Name	概要
get	
remove	ストレージ接続をストレージから切り離します。

6.183.1. get GET

表6.560 パラメーターの概要

Name	タイプ	方向	概要
connection	StorageConnection	Out	

6.183.2. remove DELETE

ストレージ接続をストレージから切り離します。

表6.561 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.184. STORAGEDOMAINSERVERCONNECTIONS

表6.562 メソッドの概要

Name	概要
add	
list	

6.184.1. add POST

表6.563 パラメーターの概要

Name	タイプ	方向	概要
connection	StorageConnection	In/Out	

6.184.2. list GET

表6.564 パラメーターの概要

Name	タイプ	方向	概要
connections	StorageConnections[]	Out	
max	Integer	In	返す接続の最大数を設定します。

6.184.2.1. max

返す接続の最大数を設定します。指定しない場合、すべての接続が返されます。

6.185. STORAGEDOMAINTEMPLATE

表6.565 メソッドの概要

Name	概要
get	
import	エクスポートストレージドメインからテンプレートをインポートするアクション。
register	
remove	

6.185.1. get GET

表6.566 パラメーターの概要

Name	タイプ	方向	概要
template	Template	Out	

6.185.2. import POST

エクスポートストレージドメインからテンプレートをインポートするアクション。

たとえば、ストレージドメイン **123** からテンプレート **456** をインポートするには、次のリクエストを送信します。

```
POST /ovirt-engine/api/storagedomains/123/templates/456/import
```

リクエスト本文は、以下のようになります。

```
<action>
  <storage_domain>
    <name>myexport</name>
  </storage_domain>
  <cluster>
    <name>mycluster</name>
  </cluster>
</action>
```

表6.567 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	インポートを非同期で実行する必要があるかどうかを示します。
clone	ブール値	In	オプションの clone パラメーターを使用して、インポートされたテンプレートとそのエンティティの新しい UUID を生成します。

Name	タイプ	方向	概要
cluster	Cluster	In	
exclusive	ブール値	In	
storage_domain main	StorageDomain	In	
template	Template	In	
vm	Vm	In	

6.185.2.1. clone

オプションの **clone** パラメーターを使用して、インポートされたテンプレートとそのエンティティの新しい UUID を生成します。

別の Red Hat Virtualization 環境によってエクスポートされたテンプレートを使用して、エクスポートドメインからテンプレートをインポートする際に、**クローン** パラメーターを **false** に設定して、ユーザーはテンプレートをインポートする場合があります。

6.185.3. register POST

表6.568 パラメーターの概要

Name	タイプ	方向	概要
allow_partial_import	ブール値	In	一部のディスクのみにテンプレートを登録できるかどうかを示します。
async	ブール値	In	登録を非同期的に実行するかどうかを指定します。
clone	ブール値	In	
cluster	Cluster	In	
exclusive	ブール値	In	
template	Template	In	

6.185.3.1. allow_partial_import

一部のディスクのみにテンプレートを登録できるかどうかを示します。

このフラグが **true** の場合、イメージが見つからなかった場合、エンジンは検証プロセスで失敗しませんが、代わりに、不足しているディスクなしでテンプレートを登録できます。これは主に、一部のストレージドメインが使用できない場合のテンプレートの登録中に使用されます。デフォルト値は **false** で

す。

6.185.4. remove DELETE

表6.569 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.186. STORAGEDOMAINTEMPLATES

表6.570 メソッドの概要

Name	概要
list	

6.186.1. list GET

表6.571 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すテンプレートの最大数を設定します。
templates	Template[]	Out	
unregistered	ブール値	In	ストレージドメイン上のディスクを含む登録済みまたは未登録のテンプレートのリストを取得するかどうかを示します。

6.186.1.1. max

返すテンプレートの最大数を設定します。指定されていない場合は、すべてのテンプレートが返されません。

6.186.1.2. unregistered

ストレージドメイン上のディスクを含む登録済みまたは未登録のテンプレートのリストを取得するかどうかを示します。未登録テンプレートのリストを取得するには、呼び出しで未登録フラグを指定する必要があります。たとえば、未登録テンプレートの一覧を取得するための REST API 呼び出しは、以下のようになります。

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

未登録フラグのデフォルト値は **false** です。リクエストは、アタッチされているストレージドメインにのみ適用されます。

6.187. STORAGEDOMAINVM

表6.572 メソッドの概要

Name	概要
get	
import	エクスポートストレージドメインから仮想マシンをインポートします。
register	
remove	エクスポートストレージドメインから仮想マシンを削除します。

6.187.1. get GET

表6.573 パラメーターの概要

Name	タイプ	方向	概要
vm	Vm	Out	

6.187.2. import POST

エクスポートストレージドメインから仮想マシンをインポートします。

たとえば、次のようなリクエストを送信します。

```
POST /ovirt-engine/api/storagedomains/123/vms/456/import
```

リクエスト本文は以下のようになります。

```
<action>
  <storage_domain>
    <name>mydata</name>
  </storage_domain>
  <cluster>
    <name>mycluster</name>
  </cluster>
</action>
```

仮想マシンを新しいエンティティとしてインポートするには、**clone** パラメーターを追加します。

```
<action>
  <storage_domain>
    <name>mydata</name>
  </storage_domain>
  <cluster>
    <name>mycluster</name>
  </cluster>
```

```
<clone>true</clone>
<vm>
  <name>myvm</name>
</vm>
</action>
```

インポートするディスクを選択するために、オプションの **disks** パラメーターを含めます。たとえば、**123** と **456** の識別子を持つテンプレートのディスクをインポートするには、以下のリクエスト本文を送信します。

```
<action>
  <cluster>
    <name>mycluster</name>
  </cluster>
  <vm>
    <name>myvm</name>
  </vm>
  <disks>
    <disk id="123"/>
    <disk id="456"/>
  </disks>
</action>
```

表6.574 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	インポートを非同期で実行する必要があるかどうかを示します。
clone	ブール値	In	インポートされた仮想マシンの識別子を再生成する必要があるかどうかを示します。
cluster	Cluster	In	
collapse_snapshots	ブール値	In	スナップショットのない仮想マシンになるように、インポートされた仮想マシンのスナップショットを折りたたむ必要があることを示します。
storage_domain	StorageDomain	In	
vm	Vm	In	

6.187.2.1. clone

インポートされた仮想マシンの識別子を再生成する必要があるかどうかを示します。

デフォルトでは、仮想マシンがインポートされると、識別子が保持されます。つまり、識別子は一意である必要があるため、同じ仮想マシンを複数回インポートすることはできません。同じマシンを複数回インポートできるようにするには、このパラメーターを **true** に設定します。デフォルトは **false** です。

6.187.2.2. collapse_snapshots

スナップショットのない仮想マシンになるように、インポートされた仮想マシンのスナップショットを折りたたむ必要があることを示します。

このパラメーターはオプションであり、明示的に指定されていない場合、デフォルト値は **false** です。

6.187.3. register POST

表6.575 パラメーターの概要

Name	タイプ	方向	概要
allow_partial_import	ブール値	In	仮想マシンをその一部のディスクのみに登録できるかどうかを示します。
async	ブール値	In	登録を非同期的に実行するかどうかを指定します。
clone	ブール値	In	
cluster	Cluster	In	
reassign_bad_macs	ブール値	In	エンジンによるインポートプロセス中に、問題のある MAC アドレスを再割り当てする必要があるかどうかを示します。
vm	Vm	In	
vnic_profile_mappings	VnicProfileMapping[]	In	インポートプロセス中に適用される仮想 NIC プロファイルのマッピングルール。

6.187.3.1. allow_partial_import

仮想マシンをその一部のディスクのみに登録できるかどうかを示します。

このフラグが **true** の場合、イメージが見つからなくても、エンジンは検証プロセスで失敗しませんが、代わりに、欠落しているディスクなしで仮想マシンを登録できます。これは主に、一部のストレージドメインが使用できない場合の仮想マシンの登録中に使用されます。デフォルト値は **false** です。

6.187.3.2. reassign_bad_macs

エンジンによるインポートプロセス中に、問題のある MAC アドレスを再割り当てする必要があるかどうかを示します。

次のいずれかに該当する場合、MAC アドレスは問題のあるものと見なされます。

- ターゲット環境の仮想マシンに既に割り当てられている MAC アドレスと競合する。
- 対象 MAC アドレスプールの範囲外である。

6.187.4. remove DELETE

エクスポートストレージドメインから仮想マシンを削除します。

たとえば、仮想マシン **456** をストレージドメイン **123** から削除するには、以下のようなリクエストを送信します。

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

表6.576 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.188. STORAGEDOMAINVMDISKATTACHMENT

エクスポートドメイン内の仮想マシンにアタッチされているディスクの詳細を返します。

表6.577 メソッドの概要

Name	概要
get	すべてのプロパティとディスクへのリンクを含むアタッチメントの詳細を返します。

6.188.1. get GET

すべてのプロパティとディスクへのリンクを含むアタッチメントの詳細を返します。

表6.578 パラメーターの概要

Name	タイプ	方向	概要
attachment	DiskAttachment	Out	ディスクアタッチメント。

6.189. STORAGEDOMAINVMDISKATTACHMENTS

エクスポートドメイン内の仮想マシンにアタッチされているディスクの詳細を返します。

表6.579 メソッドの概要

Name	概要
list	仮想マシンに接続されているディスクを一覧表示します。

6.189.1. list GET

仮想マシンに接続されているディスクを一覧表示します。

表6.580 パラメーターの概要

Name	タイプ	方向	概要
attachments	DiskAttachment[]	Out	

6.190. STORAGEDOMAINVMS

エクスポートストレージドメインの仮想マシンを一覧表示します。

たとえば、識別子が **123** のストレージドメインで利用可能な仮想マシンを取得するには、以下のリクエストを送信します。

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

これにより、以下のレスポンス本文が返されます。

```
<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>
```

これらのコレクションの仮想マシンとテンプレートは、[StorageDomain](#) 参照と [インポート](#) アクションも含まれていることを除き、最上位の [Vm](#) コレクションと [Template](#) コレクションの対応するものと同様の表現をしています。

表6.581 メソッドの概要

Name	概要
list	

6.190.1. list GET

表6.582 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す仮想マシンの最大数を設定します。
unregistered	ブール値	In	ストレージドメイン上のディスクを含む登録済みまたは未登録の仮想マシンのリストを取得するかどうかを示します。

Name	タイプ	方向	概要
vm	Vm[]	Out	

6.190.1.1. max

返す仮想マシンの最大数を設定します。指定されていない場合は、すべての仮想マシンが返されます。

6.190.1.2. unregistered

ストレージドメイン上のディスクを含む登録済みまたは未登録の仮想マシンのリストを取得するかどうかを示します。未登録の仮想マシンのリストを取得するには、呼び出しで未登録フラグを示す必要があります。たとえば、未登録の仮想マシンのリストを取得するには、REST API 呼び出しは次のようになります。

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

未登録フラグのデフォルト値は **false** です。リクエストは、アタッチされているストレージドメインにのみ適用されます。

6.191. STORAGEDOMAINS

表6.583 メソッドの概要

Name	概要
add	新しいストレージドメインを追加します。
list	

6.191.1. add POST

新しいストレージドメインを追加します。

新しい [StorageDomain](#) を作成するには、名前、タイプ、ホスト、およびストレージ属性が必要です。**id** または **name** 属性で **host** 属性を特定します。oVirt 3.6 以降では、ストレージドメインでデフォルトで削除後にワイプオプションを有効にできます。これを設定するには、POST 要求で **wipe_after_delete** を指定します。このオプションは、ドメインの作成後に編集することが可能ですが、その場合にはすでに存在していたディスクの削除後にワイププロパティは変更されません。

指定の **名前**、**タイプ**、**storage.type**、**storage.address**、**storage.path**、および ID **123** のホストを使用して新しいストレージドメインを追加するには、以下のように要求を送信します。

```
POST /ovirt-engine/api/storagedomains
```

リクエスト本文の場合は、以下のようになります。

```
<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>

```

新しい NFS ISO ストレージドメインを作成するには、次のような要求を送信します。

```

<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>

```

新しい iSCSI ストレージドメインを作成するには、次のような要求を送信します。

```

<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>

```

表6.584 パラメーターの概要

Name	タイプ	方向	概要
storage_domain	StorageDomain	In/Out	

6.191.2. list GET

表6.585 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返すストレージドメインの最大数を設定します。
search	String	In	返されるストレージドメインを制限するために使用されるクエリー文字列。
storage_domains	StorageDomain[]	Out	

6.191.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.191.2.2. max

返すストレージドメインの最大数を設定します。指定されていない場合は、すべてのストレージドメインが返されます。

6.192. STORAGESEVERCONNECTION

表6.586 メソッドの概要

Name	概要
get	
remove	ストレージ接続を削除します。
update	ストレージ接続を更新します。

6.192.1. get GET

表6.587 パラメーターの概要

Name	タイプ	方向	概要
connection	StorageConnection	Out	

6.192.2. remove DELETE

ストレージ接続を削除します。

ストレージ接続は、ストレージドメインも LUN ディスクも参照していない場合にのみ削除できます。ホスト名または ID はオプションです。それを提供すると、そのホストからの接続が切断 (アンマウント) されます。

表6.588 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。
host	String	In	接続がアンマウント (切断) されるホストの名前または識別子。

6.192.2.1. host

接続がアンマウント (切断) されるホストの名前または識別子。指定しない場合、ホストは切断されません。

たとえば、識別子 **456** のホストを使用して、識別子 **123** のストレージ接続を削除するには、以下のようなリクエストを送信します。

```
DELETE /ovirt-engine/api/storageconnections/123?host=456
```

6.192.3. update PUT

ストレージ接続を更新します。

たとえば、ストレージサーバーのアドレスを変更するには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<storage_connection>
  <address>mynewnfs.example.com</address>
  <host>
    <name>myhost</name>
  </host>
</storage_connection>
```

表6.589 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。

Name	タイプ	方向	概要
connection	StorageConnection	In/Out	
force	ブール値	In	関連するストレージドメインのステータスに関係なく、操作が成功するかどうかを示します。

6.192.3.1. force

関連するストレージドメインのステータスに関係なく、操作が成功するかどうかを示します (つまり、ストレージドメインのステータスがメンテナンスではない場合にも更新が適用されます)。

このパラメーターはオプションであり、デフォルト値は **false** です。

6.193. STORAGE SERVER CONNECTION EXTENSION

表6.590 メソッドの概要

Name	概要
get	
remove	
update	指定されたホストのストレージサーバー接続エクステンションを更新します。

6.193.1. get GET

表6.591 パラメーターの概要

Name	タイプ	方向	概要
extension	StorageConnectionExtension	Out	

6.193.2. remove DELETE

表6.592 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.193.3. update PUT

指定されたホストのストレージサーバー接続エクステンションを更新します。

ホスト **123** のストレージ接続 **456** を更新するには、以下のようなリクエストを送信します。

```
PUT /ovirt-engine/api/hosts/123/storageconnectionextensions/456
```

リクエスト本文は以下のようになります。

```
<storage_connection_extension>
  <target>iqn.2016-01.com.example:mytarget</target>
  <username>myuser</username>
  <password>mypassword</password>
</storage_connection_extension>
```

表6.593 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
extension	StorageConnectionExtension	In/Out	

6.194. STORAGESEVERCONNECTIONEXTENSIONS

表6.594 メソッドの概要

Name	概要
add	指定されたホストの新しいストレージサーバー接続エクステンションを作成します。
list	

6.194.1. add POST

指定されたホストの新しいストレージサーバー接続エクステンションを作成します。

このエクステンションにより、ユーザーは特定のホストの iSCSI ターゲットのクレデンシャルを定義できます。たとえば、ホスト **123** から iSCSI ターゲットに接続するときに **myuser** および **mypassword** をクレデンシャルとして使用するには、次のような要求を送信します。

```
POST /ovirt-engine/api/hosts/123/storageconnectionextensions
```

リクエスト本文は以下のようになります。

```
<storage_connection_extension>
  <target>iqn.2016-01.com.example:mytarget</target>
  <username>myuser</username>
```

```
<password>mypassword</password>
</storage_connection_extension>
```

表6.595 パラメーターの概要

Name	タイプ	方向	概要
extension	StorageConnectionExtension	In/Out	

6.194.2. list GET

表6.596 パラメーターの概要

Name	タイプ	方向	概要
extensions	StorageConnectionExtension[]	Out	
max	Integer	In	返すエクステンションの最大数を設定します。

6.194.2.1. max

返すエクステンションの最大数を設定します。指定されていない場合は、すべてのエクステンションが返されます。

6.195. STORAGESEVERCONNECTIONS

表6.597 メソッドの概要

Name	概要
add	新しいストレージ接続を作成します。
list	

6.195.1. add POST

新しいストレージ接続を作成します。

たとえば、NFS サーバー **my nfs.example.com** と NFS 共有 **/export/mydata** の新しいストレージ接続を作成するには、以下のようなリクエストを送信します。

```
POST /ovirt-engine/api/storageconnections
```

リクエスト本文は以下のようになります。

```

<storage_connection>
  <type>nfs</type>
  <address>my nfs.example.com</address>
  <path>/export/mydata</path>
  <host>
    <name>myhost</name>
  </host>
</storage_connection>

```

表6.598 パラメーターの概要

Name	タイプ	方向	概要
connection	StorageConnection	In/Out	

6.195.2. list GET

表6.599 パラメーターの概要

Name	タイプ	方向	概要
connections	StorageConnection[]	Out	
max	Integer	In	返す接続の最大数を設定します。

6.195.2.1. max

返す接続の最大数を設定します。指定しない場合、すべての接続が返されます。

6.196. システム

表6.600 メソッドの概要

Name	概要
get	製品名、バージョン番号、関連オブジェクト数の概要など、API を説明する基本情報を返します。
reloadconfigurations	

6.196.1. get GET

製品名、バージョン番号、関連オブジェクト数の概要など、API を説明する基本情報を返します。

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

次の応答が得られます。

```

<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>

```

エントリーポイントは、仮想化環境のコレクションへのリンクをユーザーに提供します。各コレクションリンクの **rel** 属性は、各リンクの参照ポイントを提供します。

エントリーポイントには、**product_info**、**special_objects**、および **summary** などの他のデータも含まれます。

表6.601 パラメーターの概要

Name	タイプ	方向	概要
api	Api	Out	

6.196.2. reloadconfigurations POST

表6.602 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	リロードを非同期で実行する必要があるかどうかを示します。

6.197. SYSTEMPERMISSIONS

このサービスは新しいメソッドを追加しません。これは、システムオブジェクトに割り当てられたパーミッションを管理するリソースのパスを指定するアノテーションのプレースホルダーに過ぎません。

表6.603 メソッドの概要

Name	概要
add	特定のエンティティのユーザーまたはグループに新しいパーミッションを割り当てます。
list	特定のエンティティのすべてのパーミッションを一覧表示します。

6.197.1. add POST

特定のエンティティのユーザーまたはグループに新しいパーミッションを割り当てます。

たとえば、**UserVmManager** ロールを ID が **123** の仮想マシン、id が **456** のユーザーに割り当てるには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

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

id が **456** のユーザーに **SuperUser** ロールを割り当てるには、以下のように要求を送信します。

POST /ovirt-engine/api/permissions

リクエスト本文は以下のようになります。

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

ユーザーの代わりにグループにパーミッションを割り当てる場合は、**user** 要素を **group** の適切な **ID** に置き換えます。たとえば、**UserRole** ロールを ID が **123** のクラスター、ID が **789** のグループに割り当てるには、以下のような要求を送信します。

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

リクエスト本文は以下のようになります。

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

表6.604 パラメーターの概要

Name	タイプ	方向	概要
permission	パーミ ッション	In/Out	パーミッション。

6.197.2. list GET

特定のエンティティーのすべてのパーミッションを一覧表示します。

たとえば、id **123** のクラスターのすべての権限を一覧表示するには、以下のように要求を送信します。

```
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>
```

表6.605 パラメーターの概要

Name	タイプ	方向	概要
permissions	Permission[]	Out	パーミッションのリスト

6.198. タグ

システム内の特定のタグを管理するサービス。

表6.606 メソッドの概要

Name	概要
get	タグに関する情報を取得します。
remove	システムからタグを削除します。
update	タグエンティティを更新します。

6.198.1. get GET

タグに関する情報を取得します。

たとえば、ID **123** のタグに関する情報を取得するには、以下のようなリクエストを送信します。

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

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

表6.607 パラメーターの概要

Name	タイプ	方向	概要
tag	タグ	Out	タグ。

6.198.2. remove DELETE

システムからタグを削除します。

たとえば、ID **123** のタグを削除するには、以下のようなリクエストを送信します。

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

表6.608 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.198.3. update PUT

タグエンティティを更新します。

たとえば、親タグを ID **123** のタグの ID **456** のタグに更新するには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<tag>
  <parent id="456"/>
</tag>
```

ID の代わりにタグ名を指定することもできます。たとえば、親タグを ID **123** のタグの **mytag** という名前のタグに更新するには、以下のようなリクエストを送信します。

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

表6.609 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
tag	タグ	In/Out	更新されたタグ。

6.199. タグ

システム内のタグのコレクションを管理するサービスを表します。

表6.610 メソッドの概要

Name	概要
add	システムに新しいタグを追加します。
list	システム内のタグを一覧表示します。

6.199.1. add POST

システムに新しいタグを追加します。

たとえば、**mytag** という名前の新しいタグをシステムに追加するには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

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



注記

root タグは、親タグが指定されていない場合にデフォルトの親タグと見なされる特別な疑似タグになります。root タグを削除したり、親タグを割り当てたりすることはできません。

特定の親タグを持つ新しいタグを作成するには、以下のようなリクエスト本文を送信します。

```
<tag>
  <name>mytag</name>
  <parent>
    <name>myparenttag</name>
  </parent>
</tag>
```

表6.611 パラメーターの概要

Name	タイプ	方向	概要
tag	タグ	In/Out	追加されたタグ。

6.199.2. list GET

システム内のタグを一覧表示します。

たとえば、システム内のタグの完全な階層を一覧表示するには、以下のようなリクエストを送信します。

```
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>
```

前の XML 出力では、以下のようなタグの階層を確認できます。

```
root:    (id: 111)
  - root2 (id: 222)
  - root3 (id: 333)
```

表6.612 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すタグの最大数を設定します。
tags	Tag[]	Out	システム内のすべてのタグの一覧。

6.199.2.1. max

返すタグの最大数を設定します。指定されていない場合は、すべてのタグが返されます。

6.200. TEMPLATE

仮想マシンのテンプレートとテンプレートのバージョンを管理します。

表6.613 メソッドの概要

Name	概要
export	テンプレートをデータセンターのエクスポートドメインにエクスポートします。

Name	概要
get	このテンプレートまたはテンプレートバージョンに関する情報を返します。
remove	仮想マシンテンプレートを削除します。
update	テンプレートを更新します。

6.200.1. export POST

テンプレートをデータセンターのエクスポートドメインにエクスポートします。

たとえば、以下のリクエストを使用すると操作が容易になります。

```
POST /ovirt-engine/api/templates/123/export
```

リクエスト本文は以下のようになります。

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

表6.614 パラメーターの概要

Name	タイプ	方向	概要
exclusive	ブール値	In	同じ名前の既存のテンプレートを上書きする必要があるかどうかを示します。
storage_domain	StorageDomain	In	宛先のエクスポートストレージドメインを指定します。

6.200.1.1. exclusive

同じ名前の既存のテンプレートを上書きする必要があるかどうかを示します。

宛先ドメインに同じ名前のテンプレートが存在する場合、エクスポートアクションは失敗したアクションを報告します。このパラメーターを **true** に設定すると、この動作が変更され、既存のテンプレートがすべて上書きされます。

6.200.2. get GET

このテンプレートまたはテンプレートバージョンに関する情報を返します。

表6.615 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
template	Template	Out	テンプレートまたはテンプレートバージョンに関する情報。

6.200.3. remove DELETE

仮想マシンテンプレートを削除します。

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

表6.616 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.200.4. update PUT

テンプレートを更新します。

名前、説明、タイプ、メモリー、CPU、トポロ

ジー、**os**、**high_availability**、**display**、**stateless**、**usb**、および **timezone** 要素は、テンプレートの作成後に更新できます。

たとえば、1 GiB のメモリーを持つようにテンプレートを更新するには、以下のようなリクエストを送信します。

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

リクエスト本文は、以下のようになります。

```
<template>
  <memory>1073741824</memory>
</template>
```

version_name name 属性は、テンプレートバージョンに使用される **version** 属性内で更新できる唯一の属性です。

```
<template>
  <version>
    <version_name>mytemplate_2</version_name>
  </version>
</template>
```

表6.617 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
template	Template	In/Out	

6.201. TEMPLATECDROM

テンプレートで CD-ROM デバイスを管理するサービス。

表6.618 メソッドの概要

Name	概要
get	この CDROM デバイスに関する情報を返します。

6.201.1. get GET

この CDROM デバイスに関する情報を返します。

たとえば、テンプレート **123** の CD-ROM デバイスに関する情報を取得するには、以下のようなリクエストを送信します。

```
GET /ovirt-engine/api/templates/123/cdroms/
```

表6.619 パラメーターの概要

Name	タイプ	方向	概要
cdrom	Cdrom	Out	CDROM デバイスに関する情報。

6.201.1.1. cdrom

CDROM デバイスに関する情報。

情報は、CD-ROM デバイス、テンプレート、およびオプションで挿入されたディスクへの参照を含む **cdrom** 属性で設定されています。

ディスクが挿入されている場合は、**file** 属性には ISO イメージへの参照が含まれます。

```
<cdrom href="..." id="00000000-0000-0000-0000-000000000000">
  <template href="/ovirt-engine/api/templates/123" id="123"/>
  <file id="mycd.iso"/>
</cdrom>
```

ディスクが挿入されていない場合は、**file** 属性は報告されません。

```
<cdrom href="..." id="00000000-0000-0000-0000-000000000000">
  <template href="/ovirt-engine/api/templates/123" id="123"/>
</cdrom>
```

6.202. TEMPLATECDROMS

テンプレートの CD-ROM デバイスを一覧表示します。

表6.620 メソッドの概要

Name	概要
list	

6.202.1. list GET

表6.621 パラメーターの概要

Name	タイプ	方向	概要
cdroms	Cdrom[]	Out	テンプレートの CD-ROM デバイスの一覧。
max	Integer	In	返す CD-ROM の最大数を設定します。

6.202.1.1. max

返す CD-ROM の最大数を設定します。指定されていない場合は、すべての CD-ROM が返されます。

6.203. TEMPLATEDISK

表6.622 メソッドの概要

Name	概要
copy	
export	
get	
remove	

6.203.1. copy POST

表6.623 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	コピーを非同期的に実行するかどうかを指定します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

6.203.2. export POST

表6.624 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	エクスポートを非同期で実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

6.203.3. get GET

表6.625 パラメーターの概要

Name	タイプ	方向	概要
disk	ディスク	Out	

6.203.4. remove DELETE

表6.626 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.204. TEMPLATEDISKATTACHMENT

このサービスは、テンプレートへのディスクの割り当てを管理します。

表6.627 メソッドの概要

Name	概要
get	アタッチメントの詳細を返します。

Name	概要
remove	テンプレートからディスクを削除します。

6.204.1. get GET

アタッチメントの詳細を返します。

表6.628 パラメーターの概要

Name	タイプ	方向	概要
attachment	DiskAttachment	Out	

6.204.2. remove DELETE

テンプレートからディスクを削除します。ディスクは、他のストレージドメインにディスクのコピーが存在する場合にのみ削除されます。

どのコピーを削除するかを決定するには、ストレージドメインを指定する必要があります (テンプレートディスクは複数のストレージドメインにコピーを持つことができます)。

```
DELETE /ovirt-engine/api/templates/{template:id}/diskattachments/{attachment:id}?
storage_domain=072fbaa1-08f3-4a40-9f34-a5ca22dd1d74
```

表6.629 パラメーターの概要

Name	タイプ	方向	概要
force	ブール値	In	
storage_domain	String	In	削除するイメージが存在するストレージドメインの識別子を指定します。

6.205. TEMPLATEDISKATTACHMENTS

このサービスは、テンプレートに割り当てられた一連のディスクを管理します。アタッチされた各ディスクは、[DiskAttachment](#) で表されます。

表6.630 メソッドの概要

Name	概要
list	テンプレートにアタッチされているディスクを一覧表示します。

6.205.1. list GET

テンプレートにアタッチされているディスクを一覧表示します。

表6.631 パラメーターの概要

Name	タイプ	方向	概要
attachments	DiskAttachment[]	Out	

6.206. TEMPLATEDISKS

表6.632 メソッドの概要

Name	概要
list	

6.206.1. list GET

表6.633 パラメーターの概要

Name	タイプ	方向	概要
disks	Disk[]	Out	
max	Integer	In	返すディスクの最大数を設定します。

6.206.1.1. max

返すディスクの最大数を設定します。指定しない場合、すべてのディスクが返されます。

6.207. TEMPLATEGRAPHICSCONSOLE

表6.634 メソッドの概要

Name	概要
get	テンプレートのグラフィックコンソール設定を取得します。
remove	テンプレートからグラフィックコンソールを削除します。

6.207.1. get GET

テンプレートのグラフィックコンソール設定を取得します。

表6.635 パラメーターの概要

Name	タイプ	方向	概要
console	GraphicsConsole	Out	テンプレートのグラフィックコンソールに関する情報。

6.207.2. remove DELETE

テンプレートからグラフィックコンソールを削除します。

表6.636 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.208. TEMPLATEGRAPHICSCONSOLES

表6.637 メソッドの概要

Name	概要
add	テンプレートに新しいグラフィックコンソールを追加します。
list	テンプレートの設定済みグラフィックコンソールをすべて一覧表示します。

6.208.1. add POST

テンプレートに新しいグラフィックコンソールを追加します。

表6.638 パラメーターの概要

Name	タイプ	方向	概要
console	GraphicsConsole	In/Out	

6.208.2. list GET

テンプレートの設定済みグラフィックコンソールをすべて一覧表示します。

表6.639 パラメーターの概要

Name	タイプ	方向	概要
consoles	GraphicsConsole[]	Out	テンプレートのグラフィックコンソールのリスト。

Name	タイプ	方向	概要
max	Integer	In	返すコンソールの最大数を設定します。

6.208.2.1. max

返すコンソールの最大数を設定します。指定しない場合、すべてのコンソールが返されます。

6.209. TEMPLATENIC

表6.640 メソッドの概要

Name	概要
get	
remove	
update	

6.209.1. get GET

表6.641 パラメーターの概要

Name	タイプ	方向	概要
nic	Nic	Out	

6.209.2. remove DELETE

表6.642 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.209.3. update PUT

表6.643 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
nic	Nic	In/Out	

6.210. TEMPLATENICS

表6.644 メソッドの概要

Name	概要
add	
list	

6.210.1. add POST

表6.645 パラメーターの概要

Name	タイプ	方向	概要
nic	Nic	In/Out	

6.210.2. list GET

表6.646 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す NIC の最大数を設定します。
nics	Nic[]	Out	

6.210.2.1. max

返す NIC の最大数を設定します。指定されていない場合は、すべての NIC が返されます。

6.211. TEMPLATEWATCHDOG

表6.647 メソッドの概要

Name	概要
get	
remove	
update	

6.211.1. get GET

表6.648 パラメーターの概要

Name	タイプ	方向	概要
watchdog	Watchdog	Out	

6.211.2. remove DELETE

表6.649 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.211.3. update PUT

表6.650 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
watchdog	Watchdog	In/Out	

6.212. TEMPLATEWATCHDOGS

表6.651 メソッドの概要

Name	概要
add	
list	

6.212.1. add POST

表6.652 パラメーターの概要

Name	タイプ	方向	概要
watchdog	Watchdog	In/Out	

6.212.2. list GET

表6.653 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すウォッチドッグの最大数を設定します。
watchdogs	Watchdog[]	Out	

6.212.2.1. max

返すウォッチドッグの最大数を設定します。指定しない場合、すべてのウォッチドッグが返されます。

6.213. テンプレート

このサービスは、システムで使用可能な仮想マシンテンプレートを管理します。

表6.654 メソッドの概要

Name	概要
add	新しいテンプレートを作成します。
list	仮想マシンテンプレートの一覧を返します。

6.213.1. add POST

新しいテンプレートを作成します。

これには、**name** 要素と **vm** 要素が必要です。 **id name** 属性で仮想マシンを特定します。

```
POST /ovirt-engine/api/templates
```

リクエスト本文は以下のようになります。

```
<template>
  <name>mytemplate</name>
  <vm id="123"/>
</template>
```

テンプレートは、既存のテンプレートのサブバージョンとして作成できます。これには、新しいテンプレートに **name** および **vm** 属性と、新しいテンプレートバージョンの **base_template** および **version_name** 属性が必要です。 **base_template** および **version_name** 属性は、**template** セクションにある **version** セクション内で指定する必要があります。 **id** または **name** 属性で仮想マシンを識別します。

```
<template>
  <name>mytemplate</name>
  <vm id="123"/>
  <version>
    <base_template id="456"/>
  </version>
</template>
```

```
<version_name>mytemplate_001</version_name>
</version>
</template>
```

表6.655 パラメーターの概要

Name	タイプ	方向	概要
clone_permissions	ブール値	In	仮想マシンの権限をテンプレートにコピーするかどうかを指定します。
seal	ブール値	In	テンプレートをシールします。
template	Template	In/Out	テンプレートまたはテンプレートバージョンに関する情報。

6.213.1.1. clone_permissions

仮想マシンの権限をテンプレートにコピーするかどうかを指定します。

このオプションのパラメーターが指定され、その値が **true** の場合、仮想マシンのパーミッション（継承されたものではなく、直接のパーミッションのみ）が作成されたテンプレートにコピーされます。たとえば、パーミッションをコピーして **myvm** 仮想マシンからテンプレートを作成するには、次のよう
なリクエストを送信します。

```
POST /ovirt-engine/api/templates?clone_permissions=true
```

リクエスト本文は以下のようになります。

```
<template>
  <name>mytemplate</name>
  <vm>
    <name>myvm</name>
  </vm>
</template>
```

6.213.1.2. seal

テンプレートをシールします。

このオプションのパラメーターが指定され、その値が **true** の場合、テンプレートは作成後にシールされます。

シーリングにより、ホスト固有の設定すべてがファイルシステムから削除されます。SSH キー、UDEV ルール、MAC アドレス、システム ID、ホスト名などにより、テンプレートを使用して手動の介入なしに複数の仮想マシンを簡単に作成できます。

現在、シーリングは Linux OS でのみサポートされています。

6.213.2. list GET

仮想マシンテンプレートの一覧を返します。

以下に例を示します。

```
GET /ovirt-engine/api/templates
```

仮想マシンと仮想マシンテンプレートの一覧を返します。

表6.656 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返すテンプレートの最大数を設定します。
search	String	In	返されたボリュームを制限するために使用されるクエリー文字列です。
templates	Template[]	Out	仮想マシンテンプレートのリスト。

6.213.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.213.2.2. max

返すテンプレートの最大数を設定します。指定されていない場合は、すべてのテンプレートが返されます。

6.214. UNMANAGEDNETWORK

表6.657 メソッドの概要

Name	概要
get	
remove	

6.214.1. get GET

表6.658 パラメーターの概要

Name	タイプ	方向	概要
network	Unmanaged Network	Out	

6.214.2. remove DELETE

表6.659 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.215. UNMANAGEDNETWORKS

表6.660 メソッドの概要

Name	概要
list	

6.215.1. list GET

表6.661 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すネットワークの最大数を設定します。
networks	Unmanaged Network[]	Out	

6.215.1.1. max

返すネットワークの最大数を設定します。指定されていない場合には、すべてのネットワークが返されます。

6.216. USER

システム内のユーザーを管理するサービス。このサービスを使用して、ユーザーの詳細を取得するか、ユーザーを削除します。新しいユーザーを追加するには、「Users」を使用します。

表6.662 メソッドの概要

Name	概要
get	システムユーザー情報を取得します。

Name	概要
remove	システムユーザーを削除します。

6.216.1. get GET

システムユーザー情報を取得します。

使用方法

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

ユーザー情報を返します。

```
<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>
```

表6.663 パラメーターの概要

Name	タイプ	方向	概要
user	User	Out	システムユーザー。

6.216.2. remove DELETE

システムユーザーを削除します。

使用方法

```
DELETE /ovirt-engine/api/users/1234
```

表6.664 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.217. USERS

システム内のユーザーを管理するサービス。

表6.665 メソッドの概要

Name	概要
add	ディレクトリーサービスからユーザーを追加します。
list	システム内のすべてのユーザーを一覧表示します。

6.217.1. add POST

ディレクトリーサービスからユーザーを追加します。

たとえば、**myextension-authz** 承認プロバイダーから **myuser** ユーザーを追加するには、次のようなリクエストを送信します。

```
POST /ovirt-engine/api/users
```

リクエスト本文は以下のようになります。

```
<user>
  <user_name>myuser@myextension-authz</user_name>
  <domain>
    <name>myextension-authz</name>
  </domain>
</user>
```

Active Directory を使用している場合は、ユーザープリンシパル名 (UPN) を **username** として渡し、その後承認プロバイダー名を渡す必要があります。[バグ 1147900](#) のため、ユーザーの UPN に設定された **principal** パラメーターも提供する必要があります。

たとえば、UPN **myuser@mysubdomain.mydomain.com** を持つユーザーを **myextension-authz** 承認プロバイダーから追加するには、以下のようなリクエスト本文を送信します。

```
<user>
  <principal>myuser@mysubdomain.mydomain.com</principal>
  <user_name>myuser@mysubdomain.mydomain.com@myextension-authz</user_name>
  <domain>
    <name>myextension-authz</name>
  </domain>
</user>
```

表6.666 パラメーターの概要

Name	タイプ	方向	概要
user	User	In/Out	

6.217.2. list GET

システム内のすべてのユーザーを一覧表示します。

使用方法

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

ユーザーの一覧を返します。

```
<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>
```

表6.667 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
max	Integer	In	返すユーザーの最大数を設定します。
search	String	In	返されるユーザーを制限するために使用されるクエリ文字列。
users	User[]	Out	ユーザーのリスト。

6.217.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.217.2.2. max

返すユーザーの最大数を設定します。指定しない場合、すべてのユーザーが返されます。

6.218. VIRTUALFUNCTIONALLOWEDNETWORK

表6.668 メソッドの概要

Name	概要
get	
remove	

6.218.1. get GET

表6.669 パラメーターの概要

Name	タイプ	方向	概要
network	Network	Out	

6.218.2. remove DELETE

表6.670 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.219. VIRTUALFUNCTIONALLOWEDNETWORKS

表6.671 メソッドの概要

Name	概要
add	
list	

6.219.1. add POST

表6.672 パラメーターの概要

Name	タイプ	方向	概要
network	Network	In/Out	

6.219.2. list GET

表6.673 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すネットワークの最大数を設定します。
networks	Network[]	Out	

6.219.2.1. max

返すネットワークの最大数を設定します。指定されていない場合には、すべてのネットワークが返されます。

6.220. VM

表6.674 メソッドの概要

Name	概要
cancelmigration	この操作は、別の物理ホストへの仮想マシンの移行を停止します。
clone	
commitsnapshot	仮想マシンをプレビューされたスナップショットの状態に永続的に復元します。
detach	プールから仮想マシンをデタッチします。
export	仮想マシンをエクスポートドメインにエクスポートします。
freezefilesystems	仮想マシンのファイルシステムをフリーズします。
get	仮想マシンの説明を取得します。
logon	外部コンソールから仮想マシンにアクセスするための自動ユーザーログオンを開始します。
maintenance	ホスト型エンジン仮想マシンのグローバルメンテナナンスモードを設定します。
migrate	仮想マシンを別の物理ホストに移行します。
previewsnapshot	仮想マシンをスナップショットの状態に一時的に復元します。
reboot	仮想マシンに再起動リクエストを送信します。

Name	概要
remove	アタッチされている仮想ディスクを含む仮想マシンを削除します。
reordermacaddresses	
shutdown	この操作は、シャットダウンリクエストを仮想マシンに送信します。
start	仮想マシンを起動します。
stop	この操作により、仮想マシンの電源が強制的にオフになります。
suspend	この操作により、仮想マシンの状態がディスクに保存され、停止されます。
thawfilesystems	仮想マシンのファイルシステムを解凍します。
ticket	仮想マシンのディスプレイにアクセスするための時間制約のある認証トークンを生成します。
undosnapshot	スナップショットをプレビューする前の状態に仮想マシンを復元します。
update	

6.220.1. cancelmigration POST

この操作は、別の物理ホストへの仮想マシンの移行を停止します。

POST /ovirt-engine/api/vms/123/cancelmigration

cancel migration アクションは、アクション固有のパラメーターを取りません。したがって、リクエスト本文には空の **action** が含まれている必要があります。

<action/>

表6.675 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	移行を非同期的にキャンセルする必要があるかどうかを示します。

6.220.2. clone POST

表6.676 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	クローンを非同期で実行するかどうかを示します。
vm	Vm	In	

6.220.3. commitsnapshot POST

仮想マシンをプレビューされたスナップショットの状態に永続的に復元します。

詳細は、[preview_snapshot](#) 操作を参照してください。

表6.677 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	スナップショットを非同期でコミットする必要があるかどうかを示します。

6.220.4. detach POST

プールから仮想マシンをデタッチします。

POST /ovirt-engine/api/vms/123/detach

detach アクションはアクション固有のパラメーターを取りません。したがって、リクエスト本文は空の **action** を含む必要があります。

<action/>

表6.678 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	デタッチアクションを非同期で実行する必要があるかどうかを示します。

6.220.5. export POST

仮想マシンをエクスポートドメインにエクスポートします。

たとえば、仮想マシン **123** をエクスポートドメイン **myexport** にエクスポートするには、以下を実行します。

POST /ovirt-engine/api/vms/123/export

リクエスト本文は以下のようになります。

<action>

```

<storage_domain>
  <name>myexport</name>
</storage_domain>
<exclusive>true</exclusive>
<discard_snapshots>true</discard_snapshots>
</action>

```

表6.679 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	エクスポートを非同期で実行する必要があるかどうかを示します。
discard_snapshots	ブール値	In	すべてのスナップショットを折りたたんで仮想マシンをエクスポートする必要がある場合は、 discard_snapshots パラメーターを使用します。
exclusive	ブール値	In	exclusive パラメーターは、仮想マシンの別のコピーがすでにエクスポートドメインに存在する場合でも、エクスポートする必要がある場合に使用しません (オーバーライド)。
storage_domain	StorageDomain	In	

6.220.6. freezefilesystems POST

仮想マシンのファイルシステムをフリーズします。

この操作は、実行中の仮想マシンのライブスナップショットを作成するときに、QEMU ゲストエージェントを使用して仮想マシンのファイルシステムをフリーズします。通常、これはマネージャーによって自動的に行われますが、OpenStack Volume (Cinder) ディスクを使用する仮想マシンでは API を使用して手動で実行する必要があります。

以下に例を示します。

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

```
<action/>
```

表6.680 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	フリーズを非同期で実行する必要があるかどうかを示します。

6.220.7. get GET

仮想マシンの説明を取得します。

表6.681 パラメーターの概要

Name	タイプ	方向	概要
all_content	ブール値	In	仮想マシンのすべての属性を応答に含める必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
next_run	ブール値	In	返された結果が現在実行中の仮想マシンを記述しているか、それとも既に実行されているが仮想マシンが再起動された場合にのみ有効になる変更を含む仮想マシンを記述しているかを示します。
vm	Vm	Out	仮想マシンの説明。

6.220.7.1. all_content

仮想マシンのすべての属性を応答に含める必要があるかどうかを示します。

デフォルトでは、以下の属性が除外されます。

- **console**
- **initialization.configuration.data** - 仮想マシンを説明する OVF ドキュメント。
- **rng_source**
- **soundcard**
- **virtio_scsi**

たとえば、仮想マシン '123' の完全な表現を取得するには、以下を実行します。

```
GET /ovirt-engine/api/vms/123?all_content=true
```



注記

これらの属性はパフォーマンスを低下させるため、デフォルトでは含まれていません。これらの属性はめったに使用されず、データベースへの追加のクエリーを必要とします。このパラメーターは、パフォーマンスが低下するため、必要な場合にのみ使用してください。

6.220.7.2. next_run

返された結果が現在実行中の仮想マシンを記述しているか、それとも既に実行されているが仮想マシンが再起動された場合にのみ有効になる変更を含む仮想マシンを記述しているかを示します。デフォルト値は **false** です。

パラメーターがリクエストに含まれているが値がない場合、値は **true** であると見なされます。以下のリクエストは、

```
GET /vms/{vm:id};next_run
```

値 **true** を使用することと同等です。

```
GET /vms/{vm:id};next_run=true
```

6.220.8. logon POST

外部コンソールから仮想マシンにアクセスするための自動ユーザーログオンを開始します。

このアクションには、**ovirt-guest-agent-gdm-plugin** および **ovirt-guest-agent-pam-module** パッケージがインストールされ、**ovirt-guest-agent** サービスが仮想マシンで実行されている必要があります。

ユーザーが外部コンソールから仮想マシンにアクセスするには、仮想マシンに対する適切なユーザー権限が必要です。

以下に例を示します。

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

リクエスト本文:

```
<action/>
```

表6.682 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	ログオンを非同期で実行する必要があるかどうかを示します。

6.220.9. maintenance POST

ホスト型エンジン仮想マシンのグローバルメンテナンスモードを設定します。

このアクションは、他の仮想マシンには影響しません。

以下に例を示します。

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

```
<action>
  <maintenance_enabled>true<maintenance_enabled/>
</action>
```

表6.683 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	グローバルメンテナンスアクションを非同期で実行する必要があるかどうかを示します。
maintenance_enabled	ブール値	In	グローバルメンテナンスを有効にするか無効にするかを示します。

6.220.10. migrate POST

仮想マシンを別の物理ホストに移行します。

以下に例を示します。

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

仮想マシンの移行先のホストを指定するには、以下を実行します。

```
<action>
  <host id="2ab5e1da-b726-4274-bbf7-0a42b16a0fc3"/>
</action>
```

表6.684 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	移行を非同期的に実行するかどうかを指定します。
cluster	Cluster	In	仮想マシンの移行先のクラスターを指定します。
force	ブール値	In	仮想マシンが移行不可として定義されている場合でも、仮想マシンを移行する必要があることを指定します。
host	ホスト	In	仮想マシンの移行先となる特定のホストを指定します。

6.220.10.1. cluster

仮想マシンの移行先のクラスターを指定します。これはオプションのパラメーターです。デフォルトでは、仮想マシンは同じクラスター内の別のホストに移行されます。

6.220.10.2. force

仮想マシンが移行不可として定義されている場合でも、仮想マシンを移行する必要があることを指定します。これはオプションのパラメーターです。デフォルトでは **false** に設定されます。

6.220.10.3. host

仮想マシンの移行先となる特定のホストを指定します。これはオプションのパラメーターです。デフォ

ルトでは、Red Hat Virtualization Manager は移行用のデフォルトホストを同じクラスター内から自動的に選択します。API ユーザーが特定のホストを必要とする場合、ユーザーは **id** または **name** パラメーターを使用してホストを指定できます。

6.220.11. previewsnapshot POST

仮想マシンをスナップショットの状態に一時的に復元します。

スナップショットは、**snapshot.id** パラメーターで示されます。内容を確認できるように、一時的に復元されます。検査が完了すると、仮想マシンの状態を **commit_snapshot** メソッドを使用して永続的にするか、**undo_snapshot** メソッドを使用して破棄できます。

表6.685 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	プレビューを非同期で実行する必要があるかどうかを示します。
disks	Disk[]	In	
restore_memory	ブール値	In	
snapshot	スナップショット	In	
vm	Vm	In	

6.220.12. reboot POST

仮想マシンに再起動リクエストを送信します。

以下に例を示します。

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

reboot アクションはアクション固有のパラメーターを取りません。したがって、リクエスト本文は空の **action** を含む必要があります。

```
<action/>
```

表6.686 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	再起動を非同期で実行する必要があるかどうかを示します。

6.220.13. remove DELETE

アタッチされている仮想ディスクを含む仮想マシンを削除します。

たとえば、識別子が **123** の仮想マシンを削除するには、以下を実行します。

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

表6.687 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。
detach_only	ブール値	In	アタッチされた仮想ディスクを削除する代わりに、最初にデタッチして保持する必要があるかどうかを示しています。
force	ブール値	In	仮想マシンを強制的に削除する必要があるかどうかを示します。

6.220.13.1. force

仮想マシンを強制的に削除する必要があるかどうかを示します。

ロックされた仮想マシンおよびロックされたディスクイメージを持つ仮想マシンは、このフラグを true に設定しないと削除できません。

6.220.14. reordermacaddresses POST

表6.688 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクションを非同期で実行する必要があるかどうかを示します。

6.220.15. shutdown POST

この操作は、シャットダウンリクエストを仮想マシンに送信します。

以下に例を示します。

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

shutdown アクションはアクション固有のパラメーターを取らないので、リクエスト本文は空の **action** を含む必要があります。

```
<action/>
```

表6.689 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	シャットダウンを非同期で実行する必要があるかどうかを示します。

6.220.16. start POST

仮想マシンを起動します。

仮想環境が完成し、機能するために必要なすべてのコンポーネントが仮想マシンに含まれている場合は、仮想マシンを起動できます。

この例では、仮想マシンを起動します。

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

リクエスト本文は以下のようになります。

```
<action/>
```

表6.690 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	起動アクションを非同期で実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
pause	ブール値	In	true に設定されている場合、仮想マシンを一時停止モードで起動します。
use_cloud_init	ブール値	In	true に設定すると、初期化タイプは cloud-init に設定されます。
use_sysprep	ブール値	In	true に設定すると、初期化タイプは Sysprep に設定されます。
vm	Vm	In	この特定の実行のための仮想マシンの定義。

6.220.16.1. pause

true に設定されている場合、仮想マシンを一時停止モードで起動します。デフォルトは **false** です。

6.220.16.2. use_cloud_init

true に設定すると、初期化タイプは **cloud-init** に設定されます。デフォルト値は **false** です。詳細は、[こちら](#) を参照してください。

6.220.16.3. use_sysprep

true に設定すると、初期化タイプは **Sysprep** に設定されます。デフォルト値は **false** です。詳細は、[こちら](#) を参照してください。

6.220.16.4. vm

この特定の実行のための仮想マシンの定義。

以下に例を示します。

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

これにより、この特定の起動に対してのみブートデバイスが CDROM に設定されます。仮想マシンの電源がオフになると、この定義は元に戻ります。

6.220.17. stop POST

この操作により、仮想マシンの電源が強制的にオフになります。

以下に例を示します。

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

stop アクションはアクション固有のパラメーターを取りません。したがって、リクエスト本文は空の **action** を含む必要があります。

```
<action/>
```

表6.691 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	stop アクションを非同期で実行する必要があるかどうかを示します。

6.220.18. suspend POST

この操作により、仮想マシンの状態がディスクに保存され、停止されます。停止された仮想マシンを起動し、起動アクションで仮想マシンの状態を復元します。

以下に例を示します。

POST /ovirt-engine/api/vms/123/suspend

suspend アクションはアクション固有のパラメーターを取らないので、リクエスト本文は空の **action** を含む必要があります。

<action/>

表6.692 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	suspend アクションを非同期で実行する必要があるかどうかを示します。

6.220.19. thawfilesystems POST

仮想マシンのファイルシステムを解凍します。

この操作は、実行中の仮想マシンのライブスナップショットを作成するときに、QEMU ゲストエージェントを使用して仮想マシンのファイルシステムを解凍します。通常、これはマネージャーによって自動的に行われますが、OpenStack Volume (Cinder) ディスクを使用する仮想マシンでは API を使用して手動で実行する必要があります。

以下に例を示します。

POST /api/vms/123/thawfilesystems

<action/>

表6.693 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	ファイルシステムの解凍アクションを非同期で実行する必要があるかどうかを示します。

6.220.20. ticket POST

仮想マシンのディスプレイにアクセスするための時間制約のある認証トークンを生成します。

以下に例を示します。

POST /ovirt-engine/api/vms/123/ticket

クライアント指定のアクションには、必要に応じて、必要なチケット値および/または有効期限 (秒単位) が含まれます。

応答は、使用された実際のチケットの値と有効期限を指定します。

<action>

```
<ticket>
  <value>abcd12345</value>
  <expiry>120</expiry>
</ticket>
</action>
```

重要

仮想マシンが1つのグラフィックスプロトコルのみをサポートするように設定されている場合、生成された認証トークンはそのプロトコルに対して有効になります。ただし、仮想マシンが複数のプロトコル (VNC と SPICE) をサポートするように設定されている場合、認証トークンは SPICE プロトコルに対してのみ有効になります。

VNC など特定のプロトコルの認証トークンを取得するには、仮想マシンのグラフィックスコンソールを管理する [サービス](#) の **ticket** メソッドを使用して、リクエストを送信します。

```
POST /ovirt-engine/api/vms/123/graphicsconsoles/456/ticket
```

表6.694 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	チケットの生成を非同期で実行する必要があるかどうかを示します。
ticket	Ticket	In/Out	

6.220.21. undosnapshot POST

スナップショットをプレビューする前の状態に仮想マシンを復元します。

詳細は、[preview_snapshot](#) 操作を参照してください。

表6.695 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	スナップショットをもとに戻すアクションを非同期で実行する必要があるかどうかを示します。

6.220.22. update PUT

表6.696 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。

Name	タイプ	方向	概要
next_run	ブール値	In	更新を仮想マシンにすぐに適用するか、仮想マシンの再起動時にのみ適用するかを示します。
vm	Vm	In/Out	

6.220.22.1. next_run

更新を仮想マシンにすぐに適用するか、仮想マシンの再起動時にのみ適用するかを示します。デフォルト値は **false** で、デフォルトでは変更が即座に適用されます。

6.221. VMAPPLICATION

仮想マシンにインストールされたアプリケーションに関する情報を提供するサービス。

表6.697 メソッドの概要

Name	概要
get	アプリケーションに関する情報を返します。

6.221.1. get GET

アプリケーションに関する情報を返します。

表6.698 パラメーターの概要

Name	タイプ	方向	概要
application	アプリケーション	Out	アプリケーションに関する情報。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

6.221.1.1. application

アプリケーションに関する情報。

この情報には、アプリケーションの名前を含む **name** 属性 (バージョンなどの追加情報を含む場合がある任意の文字列) と、仮想マシンを識別する **vm** 属性でが含まれます。

たとえば、以下のようなリクエストです。

```
GET /ovirt-engine/api/vms/123/applications/789
```

次のような情報が返される場合があります。

```
<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

仮想マシンにインストールされているアプリケーションに関する情報を提供するサービス。

表6.699 メソッドの概要

Name	概要
list	仮想マシンにインストールされているアプリケーションの一覧を返します。

6.222.1. list GET

仮想マシンにインストールされているアプリケーションの一覧を返します。

表6.700 パラメーターの概要

Name	タイプ	方向	概要
applications	Application[]	Out	仮想マシンにインストールされているアプリケーションのリスト。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返すアプリケーションの最大数を設定します。

6.222.1.1. applications

仮想マシンにインストールされているアプリケーションのリスト。

たとえば、以下のようなリクエストです。

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

次のようなリストが返される場合があります。

```
<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

返すアプリケーションの最大数を設定します。指定されていない場合は、すべてのアプリケーションが返されます。

6.223. VMCDROM

仮想マシンの CDROM デバイスを管理します。

ディスクの変更と取り出しは、**file** 属性の値を変更するために、常に **update** メソッドを使用して行われます。

表6.701 メソッドの概要

Name	概要
get	この CDROM デバイスに関する情報を返します。
update	この CDROM デバイスに関する情報を更新します。

6.223.1. get GET

この CDROM デバイスに関する情報を返します。

この情報は、CDROM デバイス、仮想マシン、およびオプションで挿入されたディスクへの参照を含む **cdrom** 属性で設定されます。

ディスクが挿入されている場合は、**file** 属性には ISO イメージへの参照が含まれます。

```
<cdrom href="..." id="00000000-0000-0000-0000-000000000000">
  <file id="mycd.iso"/>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</cdrom>
```

ディスクが挿入されていない場合は、**file** 属性は報告されません。

```
<cdrom href="..." id="00000000-0000-0000-0000-000000000000">
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</cdrom>
```

表6.702 パラメーターの概要

Name	タイプ	方向	概要
cdrom	Cdrom	Out	CDROM デバイスに関する情報。

Name	タイプ	方向	概要
current	ブール値	In	操作が現在実行中の仮想マシンの情報を返す必要があるかどうかを示します。

6.223.1.1. current

操作が現在実行中の仮想マシンの情報を返す必要があるかどうかを示します。このパラメーターはオプションであり、デフォルト値は **false** です。

6.223.2. update PUT

この CDROM デバイスに関する情報を更新します。

file 属性の値を変更することにより、ディスクを変更またはイジェクトすることができます。たとえば、ディスクを挿入または変更するには、次のようなリクエストを送信します。

```
PUT /ovirt-engine/api/vms/123/cdroms/00000000-0000-0000-0000-000000000000
```

本文には、**file** 属性の新しい値が含まれている必要があります。

```
<cdrom>
  <file id="mycd.iso"/>
</cdrom>
```

id 属性の値 (この例では **mycd.iso**) は、アタッチされた ISO ストレージドメインで使用可能なファイルに対応している必要があります。

ディスクのイジェクトには、空の **id** を持つ **file** を使用します。

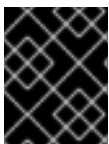
```
<cdrom>
  <file id=""/>
</cdrom>
```

デフォルトでは、上記の操作は次回の起動後に仮想マシンに表示されるディスクを永続的に変更しますが、現在実行中の仮想マシンには影響しません。現在実行中の仮想マシンに表示されるディスクを変更する場合は、**current=true** パラメーターを追加します。たとえば、現在のディスクを取り出すには、次のようなリクエストを送信します。

```
PUT /ovirt-engine/api/vms/123/cdroms/00000000-0000-0000-0000-000000000000?current=true
```

リクエスト本文は以下ようになります。

```
<cdrom>
  <file id=""/>
</cdrom>
```



重要

current=true パラメーターで行われた変更は永続化されないため、仮想マシンの再起動後は効果がありません。

表6.703 パラメーターの概要

Name	タイプ	方向	概要
cdrom	Cdrom	In/Out	CDROM デバイスに関する情報。
current	ブール値	In	現在実行中の仮想マシンに更新を適用するか、次回の起動後の仮想マシンに適用するかを指定します。

6.223.2.1. current

現在実行中の仮想マシンに更新を適用するか、次回の起動後の仮想マシンに適用するかを指定します。このパラメーターはオプションであり、デフォルト値は **false** です。これは、デフォルトでは次回の起動後にのみ更新が有効になることを意味します。

6.224. VMCDROMS

仮想マシンの CDROM デバイスを管理します。

現在、仮想マシンには CDROM デバイスが1つだけあります。新しいデバイスを追加することはできず、既存のデバイスを削除することもできないため、**add** または **remove** のメソッドはありません。CDROM ディスクの変更と取り出しは、CDROM デバイスを管理する [service](#) の [update](#) メソッドで行います。

表6.704 メソッドの概要

Name	概要
list	仮想マシンの CDROM デバイスの一覧を返します。

6.224.1. list GET

仮想マシンの CDROM デバイスの一覧を返します。

表6.705 パラメーターの概要

Name	タイプ	方向	概要
cdroms	Cdrom[]	Out	仮想マシンの CDROM デバイスのリスト。
max	Integer	In	返す CDROM の最大数を設定します。

6.224.1.1. max

返す CDROM の最大数を設定します。指定されていない場合は、すべての CDROM が返されます。

6.225. VMDISK

表6.706 メソッドの概要

Name	概要
activate	
deactivate	
export	
get	
move	
remove	仮想マシンからディスクを切り離します。
update	

6.225.1. activate POST

表6.707 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクティベーションを非同期で実行する必要があるかどうかを示します。

6.225.2. deactivate POST

表6.708 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	非アクティブ化を非同期で実行する必要があるかどうかを示します。

6.225.3. export POST

表6.709 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	エクスポートを非同期で実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

6.225.4. get GET

表6.710 パラメーターの概要

Name	タイプ	方向	概要
disk	ディスク	Out	

6.225.5. move POST

表6.711 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	移動を非同期で実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。

6.225.6. remove DELETE

仮想マシンからディスクを切り離します。



注記

API のバージョン 3 では、これによりディスクがシステムから完全に削除されていましたが、バージョン 4 以降では削除されません。完全に削除する必要がある場合は、[最上位のディスクサービスの remove メソッド](#)を使用します。

表6.712 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.225.7. update PUT

表6.713 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
disk	ディスク	In/Out	

6.226. VMDISKS

表6.714 メソッドの概要

Name	概要
add	
list	

6.226.1. add POST

表6.715 パラメーターの概要

Name	タイプ	方向	概要
disk	ディスク	In/Out	

6.226.2. list GET

表6.716 パラメーターの概要

Name	タイプ	方向	概要
disks	Disk[]	Out	
max	Integer	In	返すディスクの最大数を設定します。

6.226.2.1. max

返すディスクの最大数を設定します。指定しない場合、すべてのディスクが返されます。

6.227. VMGRAPHICSCONSOLE

表6.717 メソッドの概要

Name	概要
get	仮想マシンのグラフィックコンソール設定を取得します。
proxyticket	
remoteviewerconnectionfile	remote-viewer クライアントと互換性のあるファイルを生成します。
remove	仮想マシンからグラフィックコンソールを削除します。
ticket	この仮想マシンのコンソールにアクセスするための時間制約のある認証トークンを生成します。

6.227.1. get GET

仮想マシンのグラフィックコンソール設定を取得します。

表6.718 パラメーターの概要

Name	タイプ	方向	概要
console	GraphicsConsole	Out	仮想マシンのグラフィックコンソールに関する情報。
current	ブール値	In	以下のクエリーを使用して、グラフィックコンソールの現在のランタイム設定を取得します。

6.227.1.1. current

以下のクエリーを使用して、グラフィックコンソールの現在のランタイム設定を取得します。

```
GET /ovirt-engine/api/vms/123/graphicsconsoles/456?current=true
```

デフォルト値は **false** です。

6.227.2. proxyticket POST

表6.719 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	チケットの生成を非同期で実行する必要があるかどうかを示します。
proxy_ticket	ProxyTicket	Out	

6.227.3. remoteviewerconnectionfile POST

remote-viewer クライアントと互換性のあるファイルを生成します。

次のリクエストを使用して、グラフィックコンソールのリモートビューアー接続ファイルを生成します。このアクションは、仮想マシンが実行されている場合にのみファイルを生成することに注意してください。

```
POST /ovirt-engine/api/vms/123/graphicsconsoles/456/remoteviewerconnectionfile
```

remoteviewerconnectionfile アクションはアクション固有のパラメーターを取らないので、リクエスト本文は空の **action** を含む必要があります。

```
<action/>
```

応答には、**remote-viewer** クライアントで使用できるファイルが含まれています。

```

<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=O=local,CN=example.com
    ca=...
  </remote_viewer_connection_file>
</action>

```

たとえば、リモートビューアー接続ファイルのコンテンツを取得して一時ファイルに保存するには、ユーザーは次のように oVirt Python SDK を使用できます。

```

# 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)

```

リモートビューアー接続ファイルを作成すると、次のように仮想マシングラフィックコンソールに接続できます。

```

#!/bin/sh -ex

remote-viewer --ovirt-ca-file=/etc/pki/ovirt-engine/ca.pem /tmp/remote_viewer_connection_file.vv

```

表6.720 パラメーターの概要

Name	タイプ	方向	概要
remote_viewer_connection_file	String	Out	remote-viewer クライアントと互換性のあるファイルが含まれています。

6.227.3.1. remote_viewer_connection_file

remote-viewer クライアントと互換性のあるファイルが含まれています。

ユーザーはこの属性の内容を使用してファイルを作成し、**remote-viewer** クライアントに渡して仮想マシンのグラフィックコンソールに接続することができます。

6.227.4. remove DELETE

仮想マシンからグラフィックコンソールを削除します。

表6.721 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.227.5. ticket POST

この仮想マシンのコンソールにアクセスするための時間制約のある認証トークンを生成します。

```
POST /ovirt-engine/api/vms/123/graphicsconsoles/456/ticket
```

クライアント指定のアクションには、必要に応じて、必要なチケット値および/または有効期限 (秒単位) が含まれます。

いずれの場合も、応答は実際に使用されたチケットの値と有効期限を指定します。

```
<action>
  <ticket>
    <value>abcd12345</value>
    <expiry>120</expiry>
  </ticket>
</action>
```

表6.722 パラメーターの概要

Name	タイプ	方向	概要
ticket	Ticket	In/Out	このコンソールへのアクセスに使用できる、生成されたチケット。

6.228. VMGRAPHICSCONSOLES

表6.723 メソッドの概要

Name	概要
add	新しいグラフィックコンソールを仮想マシンに追加します。
list	仮想マシンの設定済みグラフィックコンソールをすべて一覧表示します。

6.228.1. add POST

新しいグラフィックコンソールを仮想マシンに追加します。

表6.724 パラメーターの概要

Name	タイプ	方向	概要
console	GraphicsConsole	In/Out	

6.228.2. list GET

仮想マシンの設定済みグラフィックコンソールをすべて一覧表示します。

表6.725 パラメーターの概要

Name	タイプ	方向	概要
consoles	GraphicsConsole[]	Out	仮想マシンのグラフィックコンソールのリスト。
current	ブール値	In	以下のクエリーを使用して、グラフィックコンソールの現在のランタイム設定を取得します。
max	Integer	In	返すコンソールの最大数を設定します。

6.228.2.1. current

以下のクエリーを使用して、グラフィックコンソールの現在のランタイム設定を取得します。

```
GET /ovirt-engine/api/vms/123/graphicsconsoles?current=true
```

デフォルト値は **false** です。

6.228.2.2. max

返すコンソールの最大数を設定します。指定しない場合、すべてのコンソールが返されます。

6.229. VMHOSTDEVICE

仮想マシンにアタッチされた個々のホストデバイスを管理するサービス。

表6.726 メソッドの概要

Name	概要
get	特定の仮想マシンにアタッチされた特定のホストデバイスに関する情報を取得します。
remove	このホストデバイスの接続を特定の仮想マシンから削除します。

6.229.1. get GET

特定の仮想マシンにアタッチされた特定のホストデバイスに関する情報を取得します。

以下に例を示します。

```
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>
```

表6.727 パラメーターの概要

Name	タイプ	方向	概要
device	HostDevice	Out	特定の仮想マシンにアタッチされたホストデバイスに関する情報を取得しました。

6.229.2. remove DELETE

このホストデバイスの接続を特定の仮想マシンから削除します。



注記

このデバイスが IOMMU プレースホルダーとして機能する場合、削除することはできません (削除すると、**placeholder** フラグが **true** に設定されます)。すべての IOMMU プレースホルダーデバイスは、非プレースホルダーデバイスがなくなると同時に自動的に削除されることに注意してください (指定された IOMMU グループのすべてのデバイスが切り離されます)。

DELETE /ovirt-engine/api/vms/123/hostdevices/456

表6.728 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.230. VMHOSTDEVICES

仮想マシンにアタッチされたホストデバイスを管理するサービス。

表6.729 メソッドの概要

Name	概要
add	ターゲットデバイスを指定された仮想マシンにアタッチします。
list	指定された仮想マシンに割り当てられているホストデバイスを一覧表示します。

6.230.1. add POST

ターゲットデバイスを指定された仮想マシンにアタッチします。

以下に例を示します。

POST /ovirt-engine/api/vms/123/hostdevices

HostDevice タイプのリクエスト本文の例

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



注記

ホストデバイスを正常にアタッチするために必要な前提条件は、仮想マシンが **必ず** 1つのホストにピンングされていることです。その場合、このホストに関連してデバイス ID が取得されます。



注記

より大きな IOMMU グループの一部である PCI デバイスをアタッチすると、その IOMMU グループの残りのデバイスがプレースホルダーとしてアタッチされます。その後、これらのデバイスは、**true** に設定された **HostDevice** タイプの **placeholder** 属性を使用して識別されます。

すでに IOMMU プレースホルダーとして機能しているデバイスをアタッチする場合は、そのデバイスに対して明示的な Add 操作を発行するだけで、その **placeholder** フラグがクリアされ、デバイスが仮想マシンにアクセスできるようになります。

表6.730 パラメーターの概要

Name	タイプ	方向	概要
device	HostDevice	In/Out	指定された仮想マシンにアタッチするホストデバイス。

6.230.2. list GET

指定された仮想マシンに割り当てられているホストデバイスを一覧表示します。

表6.731 パラメーターの概要

Name	タイプ	方向	概要
device	HostDevice[]	Out	取得した、指定された仮想マシンにアタッチされているホストデバイスのリスト。
max	Integer	In	返すデバイスの最大数を設定します。

6.230.2.1. max

返すデバイスの最大数を設定します。指定しない場合、すべてのデバイスが返されます。

6.231. VMNIC

表6.732 メソッドの概要

Name	概要
activate	
deactivate	
get	
remove	NIC を削除します。

Name	概要
update	NIC を更新します。

6.231.1. activate POST

表6.733 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	アクティベーションを非同期で実行する必要があるかどうかを示します。

6.231.2. deactivate POST

表6.734 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	非アクティブ化を非同期で実行する必要があるかどうかを示します。

6.231.3. get GET

表6.735 パラメーターの概要

Name	タイプ	方向	概要
nic	Nic	Out	

6.231.4. remove DELETE

NIC を削除します。

たとえば、ID **123** の仮想マシンから ID **456** の NIC を削除するには、以下のようなリクエストを送信します。

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

 重要

ホットプラグ機能は、ホットプラグ操作を行う仮想マシンオペレーティングシステムのみをサポートします。オペレーティングシステムの例は次のとおりです。

- Red Hat Enterprise Linux 6
- Red Hat Enterprise Linux 5
- Windows Server 2008 および
- Windows Server 2003

表6.736 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.231.5. update PUT

NIC を更新します。

たとえば、ID **123** の仮想マシンに属する **456** を持つ NIC を更新するには、以下のようなリクエストを送信します。

```
PUT /ovirt-engine/api/vms/123/nics/456
```

リクエスト本文は以下のようになります。

```
<nic>
  <name>myNIC</name>
  <interface>e1000</interface>
  <vnic_profile id='789' />
</nic>
```

 重要

ホットプラグ機能は、ホットプラグ操作を行う仮想マシンオペレーティングシステムのみをサポートします。オペレーティングシステムの例は次のとおりです。

- Red Hat Enterprise Linux 6
- Red Hat Enterprise Linux 5
- Windows Server 2008 および
- Windows Server 2003

表6.737 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
nic	Nic	In/Out	

6.232. VMNICS

表6.738 メソッドの概要

Name	概要
add	仮想マシンに NIC を追加します。
list	

6.232.1. add POST

仮想マシンに NIC を追加します。

以下の例では、**virtio** と **ovirtmgmt** ネットワークを使用して、**mynic** という名前のネットワークインターフェイスを仮想マシンに追加します。

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

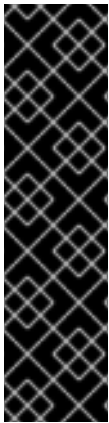
```
<nic>
  <interface>virtio</interface>
  <name>mynic</name>
  <network>
    <name>ovirtmgmt</name>
  </network>
</nic>
```

次の例では、そのリクエストを **curl** を使用して送信しています。

```
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>

重要

ホットプラグ機能は、ホットプラグ操作を行う仮想マシンオペレーティングシステムのみをサポートします。オペレーティングシステムの例は次のとおりです。

- Red Hat Enterprise Linux 6
- Red Hat Enterprise Linux 5
- Windows Server 2008 および
- Windows Server 2003

表6.739 パラメーターの概要

Name	タイプ	方向	概要
nic	Nic	In/Out	

6.232.2. list GET

表6.740 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返す NIC の最大数を設定します。
nics	Nic[]	Out	

6.232.2.1. max

返す NIC の最大数を設定します。指定されていない場合は、すべての NIC が返されます。

6.233. VMNUMANODE

表6.741 メソッドの概要

Name	概要
get	
remove	仮想 NUMA ノードを削除します。
update	仮想 NUMA ノードを更新します。

6.233.1. get GET

表6.742 パラメーターの概要

Name	タイプ	方向	概要
node	VirtualNuma Node	Out	

6.233.2. remove DELETE

仮想 NUMA ノードを削除します。

仮想 NUMA ノードを削除する例:

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

表6.743 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.233.3. update PUT

仮想 NUMA ノードを更新します。

仮想 NUMA ノードをホスト上の物理 NUMA ノードにピンニングする例:

```
PUT /ovirt-engine/api/vms/123/numanodes/456
```

リクエスト本文には、以下がが含まれている必要があります。

```
<vm_numa_node>
  <numa_node_pins>
    <numa_node_pin>
      <index>0</index>
    </numa_node_pin>
  </numa_node_pins>
</vm_numa_node>
```

表6.744 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
node	VirtualNuma Node	In/Out	

6.234. VMNUMANODES

表6.745 メソッドの概要

Name	概要
add	仮想マシンの新しい仮想 NUMA ノードを作成します。
list	仮想マシンの仮想 NUMA ノードを一覧表示します。

6.234.1. add POST

仮想マシンの新しい仮想 NUMA ノードを作成します。

NUMA ノードの作成例:

```
POST /ovirt-engine/api/vms/c7ecd2dc/numanodes
Accept: application/xml
Content-type: application/xml
```

リクエスト本文には、以下を含めることができます。

```
<vm_numa_node>
  <cpu>
    <cores>
      <core>
        <index>0</index>
      </core>
    </cores>
  </cpu>
  <index>0</index>
  <memory>1024</memory>
</vm_numa_node>
```

表6.746 パラメーターの概要

Name	タイプ	方向	概要
node	VirtualNuma Node	In/Out	

6.234.2. list GET

仮想マシンの仮想 NUMA ノードを一覧表示します。

表6.747 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すノードの最大数を設定します。

Name	タイプ	方向	概要
nodes	VirtualNuma Node[]	Out	

6.234.2.1. max

返すノードの最大数を設定します。指定されていない場合は、すべてのノードが返されます。

6.235. VMPOOL

仮想マシンプールを管理するためのサービス。

表6.748 メソッドの概要

Name	概要
allocatevm	この操作により、仮想マシンが仮想マシンプールに割り当てられます。
get	仮想マシンプールを取得します。
remove	仮想マシンプールを削除します。
update	仮想マシンプールを更新します。

6.235.1. allocatevm POST

この操作により、仮想マシンが仮想マシンプールに割り当てられます。

POST /ovirt-engine/api/vmpools/123/allocatevm

仮想マシンの割り当てアクションはアクション固有のパラメーターを実行しないため、リクエストの本文には空の **action** が含まれている必要があります。

<action/>

表6.749 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	割り当てを非同期で実行する必要があるかどうかを示します。

6.235.2. get GET

仮想マシンプールを取得します。

GET /ovirt-engine/api/vmpools/123

次のような XML 応答が返されます。

```
<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>
```

表6.750 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
pool	VmPool	Out	取得した仮想マシンプール。

6.235.3. remove DELETE

仮想マシンプールを削除します。

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

表6.751 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.235.4. update PUT

仮想マシンプールを更新します。

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

name、**description**、**size**、**prestarted_vms**、および **max_user_vms** 属性は、仮想マシンプールの作成後に更新できます。

```
<vm_pool>
```

```

<name>VM_Pool_B</name>
<description>Virtual Machine Pool B</description>
<size>3</size>
<prestarted_vms>1</size>
<max_user_vms>2</size>
</vmpool>

```

表6.752 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
pool	VmPool	In/Out	更新中の仮想マシンプール。

6.236. VMPOOLS

仮想マシンプールへの読み取り/書き込みアクセスを提供します。

表6.753 メソッドの概要

Name	概要
add	新しい仮想マシンプールを作成します。
list	使用可能な仮想マシンプールのリストを取得します。

6.236.1. add POST

新しい仮想マシンプールを作成します。

新しいプールには、**name**、**cluster**、**template** の属性が必要です。**id** または **name** のネストされた属性でクラスターとテンプレートを特定します。

```
POST /ovirt-engine/api/vmpools
```

以下のボディを使用します。

```

<vmpool>
  <name>mypool</name>
  <cluster id="123"/>
  <template id="456"/>
</vmpool>

```

表6.754 パラメーターの概要

Name	タイプ	方向	概要
pool	VmPool	In/Out	追加するプール。

6.236.2. list GET

使用可能な仮想マシンプールのリストを取得します。

```
GET /ovirt-engine/api/vmpools
```

以下の応答が返されます。

```
<vm_pools>
  <vm_pool id="123">
    ...
  </vm_pool>
  ...
</vm_pools>
```

表6.755 パラメーターの概要

Name	タイプ	方向	概要
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返すプールの最大数を設定します。
pools	VmPool[]	Out	取得したプール。
search	String	In	返されたプールを制限するために使用されるクエリー文字列。

6.236.2.1. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.236.2.2. max

返すプールの最大数を設定します。この値が指定されていない場合、すべてのプールが返されます。

6.237. VMREPORTEDDEVICE

表6.756 メソッドの概要

Name	概要
get	

6.237.1. get GET

表6.757 パラメーターの概要

Name	タイプ	方向	概要
reported_device	ReportedDevice	Out	

6.238. VMREPORTEDDEVICES

表6.758 メソッドの概要

Name	概要
list	

6.238.1. list GET

表6.759 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すデバイスの最大数を設定します。
reported_device	ReportedDevice[]	Out	

6.238.1.1. max

返すデバイスの最大数を設定します。指定しない場合、すべてのデバイスが返されます。

6.239. VMSESSION

表6.760 メソッドの概要

Name	概要
get	

6.239.1. get GET

表6.761 パラメーターの概要

Name	タイプ	方向	概要
session	Session	Out	

6.240. VMSESSIONS

仮想マシンのユーザーセッションに関する情報を提供します。

表6.762 メソッドの概要

Name	概要
list	この仮想マシンのすべてのユーザーセッションを一覧表示します。

6.240.1. list GET

この仮想マシンのすべてのユーザーセッションを一覧表示します。

たとえば、仮想マシン **123** のセッション情報を取得するには、以下のようなリクエストを送信します。

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

応答本文には、次のような内容が含まれます。

```
<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>
```

表6.763 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すセッションの最大数を設定します。
sessions	Session[]	Out	

6.240.1.1. max

返すセッションの最大数を設定します。指定されていない場合は、すべてのセッションが返されます。

6.241. VMWATCHDOG

仮想マシンのウォッチドッグを管理するサービス。

表6.764 メソッドの概要

Name	概要
get	ウォッチドッグに関する情報を返します。
remove	仮想マシンからウォッチドッグを削除します。
update	ウォッチドッグの情報を更新します。

6.241.1. get GET

ウォッチドッグに関する情報を返します。

表6.765 パラメーターの概要

Name	タイプ	方向	概要
watchdog	Watchdog	Out	ウォッチドッグに関する情報です。

6.241.1.1. watchdog

ウォッチドッグに関する情報です。

この情報は、**model** 要素、**action** 要素、および仮想マシンへの参照で設定されます。たとえば、以下のようになります。

```
<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

仮想マシンからウォッチドッグを削除します。

たとえば、仮想マシンからウォッチドッグを削除するには、次のようなリクエストを送信します。

```
DELETE /ovirt-engine/api/vms/123/watchdogs/00000000-0000-0000-0000-000000000000
```

表6.766 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.241.3. update PUT

ウォッチドッグの情報を更新します。

action 要素と **model** 要素を使用して情報を更新できます。

たとえば、ウォッチドッグを更新するには、次のようなリクエストを送信します。

```
PUT /ovirt-engine/api/vms/123/watchdogs
<watchdog>
  <action>reset</action>
</watchdog>
```

応答本文を使用:

```
<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>
```

表6.767 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
watchdog	Watchdog	In/Out	ウォッチドッグに関する情報です。

6.241.3.1. watchdog

ウォッチドッグに関する情報です。

リクエストデータには、**model** と **action** の少なくともどちらかの要素が必要です。応答データには、更新されたウォッチドッグに関する完全な情報が含まれています。

6.242. VMWATCHDOGS

仮想マシンのウォッチドッグを一覧表示します。

表6.768 メソッドの概要

Name	概要
add	仮想マシンに新しいウォッチドッグを追加します。
list	仮想マシンのウォッチドッグのリスト。

6.242.1. add POST

仮想マシンに新しいウォッチドッグを追加します。

たとえば、ウォッチドッグを仮想マシンに追加するには、次のようなリクエストを送信します。

```
POST /ovirt-engine/api/vms/123/watchdogs
<watchdog>
  <action>poweroff</action>
  <model>i6300esb</model>
</watchdog>
```

応答本文を使用:

```
<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>
```

表6.769 パラメーターの概要

Name	タイプ	方向	概要
watchdog	Watchdog	In/Out	ウォッチドッグに関する情報です。

6.242.1.1. watchdog

ウォッチドッグに関する情報です。

リクエストデータには、**model** 要素 (**i6300esb** など) と **action** 要素 (**none**、**reset**、**poweroff**、**dump**、**pause** のいずれか) が含まれている必要があります。応答データには、追加されたウォッチドッグおよび仮想マシンへの参照が含まれます。

6.242.2. list GET

仮想マシンのウォッチドッグのリスト。

表6.770 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すウォッチドッグの最大数を設定します。
watchdogs	Watchdog[]	Out	ウォッチドッグに関する情報です。

6.242.2.1. max

返すウォッチドッグの最大数を設定します。指定しない場合、すべてのウォッチドッグが返されます。

6.242.2.2. watchdogs

ウォッチドッグに関する情報です。

この情報は、**model** 要素、**action** 要素、および仮想マシンへの参照で設定されます。たとえば、以下のようになります。

```
<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

表6.771 メソッドの概要

Name	概要
add	新しい仮想マシンを作成します。
list	

6.243.1. add POST

新しい仮想マシンを作成します。

仮想マシンはさまざまな方法で作成できます。

- テンプレートから。この場合、テンプレートの識別子または名前を指定する必要があります。たとえば、プレーンシェルスクリプトと XML を使用すると、次のようになります。

```
#!/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"
```

- スナップショットから。この場合、スナップショットの識別子を指定する必要があります。たとえば、プレーンシェルスクリプトと XML を使用すると、次のようになります。

```
#!/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"
```

テンプレートまたはスナップショットから仮想マシンを作成する場合、通常は、仮想マシンのディスクを作成するストレージドメインを明示的に指定すると便利です。仮想マシンがテンプレートから作成された場合、マッピングを示す一連の **disk_attachment** 要素を渡すことで、これを行うことができます。

```
<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>
```

仮想マシンがスナップショットから作成された場合、この一連のディスクは少し異なり、**id** の代わりに **image_id** 属性を使用します。

```
<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>

```

XML 記述で追加の仮想マシンパラメーターを指定することができます。たとえば、2 GiB の RAM を備えた **desktop** タイプの仮想マシンでは、次のようなリクエスト本文を送信することで記述を追加できます。

```

<vm>
  <name>myvm</name>
  <description>My Desktop Virtual Machine</description>
  <type>desktop</type>
  <memory>2147483648</memory>
  ...
</vm>

```

起動可能な CDROM デバイスは、次のように設定できます。

```

<vm>
  ...
  <os>
    <boot dev="cdrom"/>
  </os>
</vm>

```

CDROM から起動するには、[CDROM サービス](#) で説明されているように、まずディスクを挿入する必要があります。次に、**os.boot.devices** 属性を使用して、その CDROM からの起動を指定できます。

```

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

```

いずれの場合も、仮想マシンが作成されるクラスターの名前または識別子は必須です。

表6.772 パラメーターの概要

Name	タイプ	方向	概要
------	-----	----	----

Name	タイプ	方向	概要
clone	ブール値	In	仮想マシンをテンプレートから独立させる必要があるかどうかを指定します。
clone_permissions	ブール値	In	テンプレートの権限を仮想マシンにコピーするかどうかを指定します。
vm	Vm	In/Out	

6.243.1.1. clone

仮想マシンをテンプレートから独立させる必要があるかどうかを指定します。

仮想マシンがテンプレートからデフォルトで作成される場合、仮想マシンのディスクはテンプレートのディスクに依存し、**copy on write** メカニズムを使用して、テンプレートとの違いだけが実際のストレージスペースを使用するようにします。このパラメーターが指定され、値が **true** の場合、作成された仮想マシンのディスクは複製され、テンプレートとは無関係になります。たとえば、独立した仮想マシンを作成するには、次のようなリクエストを送信します。

```
POST /ovirt-engine/vms?clone=true
```

リクエスト本文は以下のようになります。

```
<vm>
  <name>myvm</name>
  <template>
    <name>mytemplate</name>
  </template>
  <cluster>
    <name>mycluster</name>
  </cluster>
</vm>
```



注記

このパラメーターが **true** の場合、**clone_permissions=true** を使用する場合と同様に、テンプレートのパーミッションもコピーされます。

6.243.1.2. clone_permissions

テンプレートの権限を仮想マシンにコピーするかどうかを指定します。

このオプションのパラメーターが指定され、その値が **true** の場合、テンプレートのパーミッション (継承されたものではなく、直接のパーミッションのみ) が作成された仮想マシンにコピーされます。たとえば、**mytemplate** テンプレートから仮想マシンを作成し、そのパーミッションをコピーするには、次のようなリクエストを送信します。

```
POST /ovirt-engine/api/vms?clone_permissions=true
```

リクエスト本文は以下のようになります。


```

<vm>
  <name>myvm</name>
  <template>
    <name>mytemplate</name>
  </template>
  <cluster>
    <name>mycluster</name>
  </cluster>
</vm>

```

6.243.2. list GET

表6.773 パラメーターの概要

Name	タイプ	方向	概要
all_content	ブール値	In	仮想マシンのすべての属性を応答に含める必要があるかどうかを示します。
case_sensitive	ブール値	In	search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返すことのできる結果の最大数。
search	String	In	返される仮想マシンを制限するために使用されるクエリー文字列。
vms	Vm[]	Out	

6.243.2.1. all_content

仮想マシンのすべての属性を応答に含める必要があるかどうかを示します。

デフォルトでは、以下の属性が除外されます。

- **console**
- **initialization.configuration.data** - 仮想マシンを説明する OVF ドキュメント。
- **rng_source**
- **soundcard**
- **virtio_scsi**

たとえば、仮想マシンの完全な表現を取得するには、次のようなリクエストを送信します。

```
GET /ovirt-engine/api/vms?all_content=true
```



注記

これらの属性を含めない理由はパフォーマンスです。ほとんど使用されず、データベースへの追加のクエリーが必要です。そのため、本当に必要な場合にのみ、このパラメーターを使用するようにしてください。

6.243.2.2. case_sensitive

search パラメーターを使用して実行する検索を、大文字と小文字を区別して実行する必要があるかどうかを示します。デフォルト値は **true** です。つまり、大文字と小文字の区別が考慮されます。ケースを無視する場合は、**false** に設定します。

6.244. VNICPROFILE

このサービスは、vNIC プロファイルを管理します。

表6.774 メソッドの概要

Name	概要
get	vNIC プロファイルの詳細を取得します。
remove	vNIC プロファイルを削除します。
update	vNIC プロファイルの詳細を更新します。

6.244.1. get GET

vNIC プロファイルの詳細を取得します。

表6.775 パラメーターの概要

Name	タイプ	方向	概要
profile	VnicProfile	Out	

6.244.2. remove DELETE

vNIC プロファイルを削除します。

表6.776 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.244.3. update PUT

vNIC プロファイルの詳細を更新します。

表6.777 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	更新を非同期的に実行するかどうかを指定します。
profile	VnicProfile	In/Out	更新中の vNIC プロファイル。

6.245. VNICPROFILES

このサービスは、すべての vNIC プロファイルのコレクションを管理します。

表6.778 メソッドの概要

Name	概要
add	vNIC プロファイルを追加します。
list	すべての vNIC プロファイルを一覧表示します。

6.245.1. add POST

vNIC プロファイルを追加します。

たとえば、vNIC プロファイル **123** をネットワーク **456** に追加するには、次の宛先にリクエストを送信します。

```
POST /ovirt-engine/api/networks/456/vnicprofiles
```

以下のボディを使用します。

```
<vnic_profile id="123">
  <name>new_vNIC_name</name>
  <pass_through>
    <mode>disabled</mode>
  </pass_through>
  <port_mirroring>false</port_mirroring>
</vnic_profile>
```

各 VNIC プロファイルにはデフォルトのネットワークフィルターがあることに注意してください。デフォルトのネットワークフィルターの計算方法は、[NetworkFilters](#) のドキュメントを参照してください。

新しい VNIC プロファイル作成の出力は、指定された body 引数によって異なります。ネットワークフィルターが指定されていない場合は、デフォルトのネットワークフィルターが設定されます。以下に例を示します。

```
<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>

```

空のネットワークフィルターが指定された場合、VNIC プロファイルのデフォルトネットワークフィルターに関係なく、特定の VNIC プロファイルに対してネットワークフィルターは設定されません。以下に例を示します。

```

<vnic_profile>
  <name>no_network_filter</name>
  <network_filter/>
</vnic_profile>

```

特定の有効なネットワークフィルター ID が指定された場合、VNIC プロファイルのデフォルトネットワークフィルターに関係なく、指定されたネットワークフィルターを使用して VNIC プロファイルが設定されます。以下に例を示します。

```

<vnic_profile>
  <name>user_choice_network_filter</name>
  <network_filter id="0000001b-001b-001b-001b-0000000001d5"/>
</vnic_profile>

```

表6.779 パラメーターの概要

Name	タイプ	方向	概要
profile	VnicProfile	In/Out	追加される vNIC プロファイル。

6.245.2. list GET

すべての vNIC プロファイルを一覧表示します。

表6.780 パラメーターの概要

Name	タイプ	方向	概要
max	Integer	In	返すプロファイルの最大数を設定します。
profiles	VnicProfile[]	Out	すべての vNIC プロファイルのリスト。

6.245.2.1. max

返すプロファイルの最大数を設定します。指定されていない場合は、すべてのプロファイルが返されます。

6.246. 重み

表6.781 メソッドの概要

Name	概要
get	
remove	

6.246.1. get GET

表6.782 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルタする必要があるかどうかを示します。
weight	重み	Out	

6.246.2. remove DELETE

表6.783 パラメーターの概要

Name	タイプ	方向	概要
async	ブール値	In	削除を非同期的に実行するかどうかを指定します。

6.247. 重み

表6.784 メソッドの概要

Name	概要
add	
list	

6.247.1. add POST

表6.785 パラメーターの概要

Name	タイプ	方向	概要
weight	重み	In/Out	

6.247.2. list GET

表6.786 パラメーターの概要

Name	タイプ	方向	概要
filter	ブール値	In	ユーザーのパーミッションにしたがって、結果をフィルターする必要があるかどうかを示します。
max	Integer	In	返す重みの最大数を設定します。
weights	Weight[]	Out	

6.247.2.1. max

返す重みの最大数を設定します。指定しない場合、すべての重みが返されます。

第7章 タイプ

このセクションでは、API で使用できるすべてのデータ型を列挙します。

7.1. ACCESSPROTOCOL ENUM

表7.1 値の概要

Name	概要
cifs	
gluster	
nfs	

7.2. ACTION 構造体

表7.2 属性の概要

Name	タイプ	概要
allow_partial_import	ブール値	
async	ブール値	
bricks	GlusterBrick[]	
certificates	Certificate[]	
check_connectivity	ブール値	
clone	ブール値	
cluster	Cluster	
collapse_snapshots	ブール値	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
connectivity_timeout	Integer	
data_center	DataCenter	

Name	タイプ	概要
deploy_hosted_engine	ブール値	
description	String	プレーンテキストでの人間が判読できる説明。
details	GlusterVolumePro fileDetails	
discard_snapshots	ブール値	
disk	ディスク	
disks	Disk[]	
exclusive	ブール値	
fault	異常	
fence_type	String	
filter	ブール値	
fix_layout	ブール値	
force	ブール値	
grace_period	GracePeriod	
host	ホスト	
id	String	一意の ID
image	String	
import_as_template	ブール値	
is_attached	ブール値	
iscsi	IscsiDetails	
iscsi_targets	String[]	
job	Job	

Name	タイプ	概要
logical_units	LogicalUnit[]	
maintenance_enabled	ブール値	
modified_bonds	HostNic[]	
modified_labels	NetworkLabel[]	
modified_network_attachments	NetworkAttachment[]	
name	String	人間が判読できるプレーンテキストでの名前。
option	オプション	
pause	ブール値	
power_management	PowerManagement	
proxy_ticket	ProxyTicket	
reason	String	
reassign_bad_macs	ブール値	
remote_viewer_connection_file	String	
removed_bonds	HostNic[]	
removed_labels	NetworkLabel[]	
removed_network_attachments	NetworkAttachment[]	
resolution_type	String	
restore_memory	ブール値	
root_password	String	

Name	タイプ	概要
snapshot	スナップショット	
ssh	Ssh	
status	String	
stop_gluster_service	ブール値	
storage_domain	StorageDomain	
storage_domains	StorageDomain[]	
succeeded	ブール値	
synchronized_network_attachments	NetworkAttachment[]	
template	Template	
ticket	Ticket	
undeploy_hosted_engine	ブール値	
use_cloud_init	ブール値	
use_sysprep	ブール値	
virtual_functions_configuration	HostNicVirtualFunctionsConfiguration	
vm	Vm	
vnic_profile_mappings	VnicProfileMapping[]	

7.3. AFFINITYGROUP STRUCT

アフィニティーグループは、定義された関係を持つ仮想マシンのグループを表します。

表7.3 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
enforcing	ブール値	アフィニティーグループが、そのアフィニティーグループのメンバーである仮想マシンに適用されるアフィニティーのハード強制とソフト強制のどちらを使用するかを指定します。
hosts_rule	AffinityRule	このアフィニティーグループのメンバーである仮想マシンとホストの間に適用されるアフィニティールールを指定します。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
positive	ブール値	アフィニティーグループが、そのアフィニティーグループのメンバーである仮想マシンに正のアフィニティーと負のアフィニティーのどちらを適用するか指定します。
vms_rule	AffinityRule	このアフィニティーグループのメンバーである仮想マシンに適用されるアフィニティールールを指定します。

7.3.1. enforcing

アフィニティーグループが、そのアフィニティーグループのメンバーである仮想マシンに適用されるアフィニティーのハード強制とソフト強制のどちらを使用するかを指定します。



警告

この属性はエンジンのバージョン 4.1以降非推奨となり、将来削除されることに注意してください。今後は **vms_rule** 属性を使用してください。

7.3.2. positive

アフィニティーグループが、そのアフィニティーグループのメンバーである仮想マシンに正のアフィニティーと負のアフィニティーのどちらを適用するか指定します。

**警告**

この属性はエンジンのバージョン 4.1以降非推奨となり、将来削除されることに注意してください。今後は **vms_rule** 属性を使用してください。

表7.4 リンクの概要

Name	タイプ	概要
cluster	Cluster	アフィニティーグループが適用されるクラスターへの参照。
hosts	Host[]	このアフィニティーグループに割り当てられたすべてのホストのリスト。
vms	Vm[]	このアフィニティーグループに割り当てられたすべての仮想マシンのリスト。

7.4. AFFINITYLABEL 構造体

アフィニティーラベルは、仮想マシンのスケジューリングに影響を与える可能性があります。利用可能なホストからサブクラスターを作成するために最も頻繁に使用されます。

表7.5 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
read_only	ブール値	read_only プロパティは、変更できないラベルをマークします。

7.4.1. read_only

read_only プロパティは、変更できないラベルをマークします。これは通常、内部で生成されたラベルをリストする場合に該当します。

表7.6 リンクの概要

Name	タイプ	概要
hosts	Host[]	このスケジューリングラベルを使用してラベル付けされたホストのリスト。
vms	Vm[]	このスケジューリングラベルを使用してラベル付けされた仮想マシンのリスト。

7.5. AFFINITYRULE 構造体

アフィニティグループの汎用ルール定義。サポートされている各リソースタイプ(仮想マシン、ホスト)は、個別のルールによって制御されます。そのため、定義された仮想マシン間にアフィニティはないが、定義された仮想マシンと仮想ホストの間にはハードアフィニティがあるなどのルールを表現できます。

表7.7 属性の概要

Name	タイプ	概要
enabled	ブール値	アフィニティグループがこのルールを使用するかどうかを指定します。
enforcing	ブール値	アフィニティグループが、このルールで制御されるリソースに適用されるアフィニティのハード強制とソフト強制のどちらを使用するかを指定します。
positive	ブール値	アフィニティグループが、このルールによって制御されるリソースに正のアフィニティと負のアフィニティのどちらを適用するか指定します。

7.5.1. enabled

アフィニティグループがこのルールを使用するかどうかを指定します。この属性は作成時に任意であり、提供されない場合は **true** とみなされます。この属性が更新操作で指定されない場合、AffinityGroup **positive** 属性も設定されていれば **true** と見なされます。**enabled** 属性と **positive** 属性の両方が欠落している場合、バックエンドの **enabled** 値は保持されます。

7.5.2. enforcing

アフィニティグループが、このルールで制御されるリソースに適用されるアフィニティのハード強制とソフト強制のどちらを使用するかを指定します。この引数は、ルールが有効な場合は必須であり、ルールが無効な場合は無視されます。

7.5.3. positive

アフィニティグループが、このルールによって制御されるリソースに正のアフィニティと負のアフィニティのどちらを適用するか指定します。この引数は、ルールが有効な場合は必須であり、ルールが無効な場合は無視されます。

7.6. AGENT 構造体

フェンスエージェントを表すタイプ。

表7.8 属性の概要

Name	タイプ	概要
address	String	フェンスエージェントのアドレス。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
concurrent	ブール値	エージェントを同時に使用するか、順次使用するかを指定します。
description	String	プレーンテキストでの人間が判読できる説明。
encrypt_options	ブール値	オプションを暗号化するかどうかを指定します。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
options	Option[]	フェンスエージェントオプション (キーと値のペアのコンマ区切りリスト)。
order	Integer	他のエージェントと一緒に使用する場合のこのエージェントの順序。
password	String	フェンスエージェントのパスワード。
port	Integer	フェンスエージェントのポート。
type	String	フェンスエージェントのタイプ。
username	String	フェンスエージェントのユーザー名。

表7.9 リンクの概要

Name	タイプ	概要
host	ホスト	ホストサービスへの参照。

7.6.1. host

ホストサービスへの参照。各フェンスエージェントは、1つのホストに属します。

7.7. AGENTCONFIGURATION 構造体

表7.10 属性の概要

Name	タイプ	概要
address	String	
broker_type	MessageBrokerType	
network_mappings	String	
password	String	
port	Integer	
username	String	

7.8. API 構造体

このタイプには、API の root サービスによって返される情報が含まれます。

その情報を取得するには、次のようなリクエストを送信します。

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

結果は以下のようになります。

```
<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>

```

表7.11 属性の概要

Name	タイプ	概要
product_info	ProductInfo	製品名、ベンダー名、バージョンなど、製品に関する情報。
special_objects	SpecialObjects	空のテンプレートやタグ階層のルートなどの、特別なオブジェクトへの参照。
summary	ApiSummary	仮想マシン、ホスト、ストレージドメインなどの関連オブジェクトの総数を含む概要。
time	日付	この情報が生成された日時。

7.9. APISUMMARY 構造体

仮想マシン、ホスト、ストレージドメインなどの関連オブジェクトの総数を含む概要。

表7.12 属性の概要

Name	タイプ	概要
hosts	ApiSummaryItem	ホストの概要。
storage_domains	ApiSummaryItem	ストレージドメインの概要。
users	ApiSummaryItem	ユーザーの概要。
vms	ApiSummaryItem	仮想マシンの概要。

7.10. APISUMMARYITEMSTRUCT

このタイプには、API 概要の項目が含まれます。各項目には、ある種類のオブジェクトの合計数とアクティブな数が含まれています。

表7.13 属性の概要

Name	タイプ	概要
active	Integer	アクティブなオブジェクトの総数。
total	Integer	オブジェクトの総数。

7.11. APPLICATION 構造体

表7.14 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.15 リンクの概要

Name	タイプ	概要
vm	Vm	

7.12. アーキテクチャー ENUM

表7.16 値の概要

Name	概要
ppc64	
undefined	
x86_64	

7.13. AUTHORIZEDKEY 構造体

表7.17 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
key	String	
name	String	人間が判読できるプレーンテキストでの名前。

表7.18 リンクの概要

Name	タイプ	概要
user	User	

7.14. AUTONUMASTATUS ENUM

表7.19 値の概要

Name	概要
disable	
enable	
unknown	

7.15. BALANCE 構造体

表7.20 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.21 リンクの概要

Name	タイプ	概要
scheduling_policy	SchedulingPolicy	
scheduling_policy_unit	SchedulingPolicyUnit	

7.16. BIOS 構造体

表7.22 属性の概要

Name	タイプ	概要
boot_menu	BootMenu	

7.17. BLOCKSTATISTIC 構造体

表7.23 属性の概要

Name	タイプ	概要
statistics	Statistic[]	

7.18. BONDING 構造体

ネットワークインターフェイスボンドを表します。

表7.24 属性の概要

Name	タイプ	概要
ad_partner_mac	Mac	モード 4 のパートナーボンディングの ad_partner_mac プロパティ。
options	Option[]	ボンディングインターフェイスの option 要素のリスト。
slaves	HostNic[]	結合されたインターフェイスの slave NIC のリスト。

7.18.1. ad_partner_mac

モード 4 のパートナーボンディングの **ad_partner_mac** プロパティ。ボンディングモード 4 は 802.3ad 標準であり、動的リンクアグリゲーションとも呼ばれます。詳細は、[Wikipedia](#) および [Presentation](#) を参照してください。**ad_partner_mac** は、ボンディングのもう一方の端にあるシステム (スイッチ) の MAC アドレスです。このパラメーターは読み取り専用です。設定してもボンディングには影響しません。これは、ボンディングが配置されているシステムの `/sys/class/net/bondX/bonding/ad_partner_mac` ファイルから取得されます。

7.18.2. options

ボンディングインターフェイスの option 要素のリスト。各オプションには、プロパティ名と値の属性が含まれています。ボンディングインターフェイスを追加する場合にのみ必要です。

7.18.3. slaves

結合されたインターフェイスの slave NIC のリスト。ボンディングインターフェイスを追加する場合にのみ必要です。

表7.25 リンクの概要

Name	タイプ	概要
active_slave	HostNic	それをサポートするモード (active-backup、balance-alb、balance-tlb) でのボンディングの active_slave プロパティ。

7.18.4. active_slave

それをサポートするモード (active-backup、balance-alb、balance-tlb) でのボンディングの **active_slave** プロパティ。詳細は、[Linux ドキュメント](#) を参照してください。このパラメーターは読み取り専用です。設定してもボンディングには影響しません。これは、ボンディングが配置されているシステムの `/sys/class/net/bondX/bonding/active_slave` ファイルから取得されます。

以下に例を示します。

```
GET /ovirt-engine/api/hosts/123/nics/321
```

以下を応答します。

```
<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 構造体

表7.26 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。

Name	タイプ	概要
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
value	String	

7.20. BOOT 構造体

表7.27 属性の概要

Name	タイプ	概要
devices	BootDevice[]	

7.21. BOOTDEVICE ENUM

表7.28 値の概要

Name	概要
cdrom	
hd	
network	

7.22. BOOTMENU STRUCT

表7.29 属性の概要

Name	タイプ	概要
enabled	ブール値	

7.23. BOOTPROTOCOL ENUM

NIC への IP アドレス割り当て方法のオプションを定義します。

表7.30 値の概要

Name	概要
autoconf	ステートレスアドレスの自動設定。

Name	概要
dhcp	Dynamic host configuration protocol (DHCP)
none	アドレス設定なし。
static	静的に定義されたアドレス、マスク、およびゲートウェイ。

7.23.1. autoconf

ステートレスアドレスの自動設定。

このメカニズムは [RFC 4862](#) で定義されています。詳細は、[こちらの Wikipedia の記事](#) を参照してください。



注記

この値は、IPv6 アドレスに対してのみ有効です。

7.23.2. dhcp

Dynamic host configuration protocol (DHCP)

詳細は、[こちらの Wikipedia の記事](#) を参照してください。

7.24. BRICKPROFILEDETAIL 構造体

表7.31 属性の概要

Name	タイプ	概要
profile_details	ProfileDetail[]	

表7.32 リンクの概要

Name	タイプ	概要
brick	GlusterBrick	

7.25. CDROM 構造体

表7.33 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。

Name	タイプ	概要
description	String	プレーンテキストでの人間が判読できる説明。
file	File	
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.34 リンクの概要

Name	タイプ	概要
instance_type	InstanceType	オプションで、デバイスが使用するインスタンスタイプへの参照。
template	Template	オプションで、デバイスが使用するテンプレートへの参照。
vm	Vm	この要素は使用せず、代わりに vms を使用してください。
vms	Vm[]	このデバイスを使用する仮想マシンへの参照。

7.25.1. vms

このデバイスを使用する仮想マシンへの参照。デバイスは複数の仮想マシンで使用できます。たとえば、共有ディスクは2つ以上の仮想マシンで同時に使用することができます。

7.26. CERTIFICATE 構造体

表7.35 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
content	String	
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
organization	String	

Name	タイプ	概要
subject	String	

7.27. CLOUDINIT STRUCT

表7.36 属性の概要

Name	タイプ	概要
authorized_keys	AuthorizedKey[]	
files	File[]	
host	ホスト	
network_configuration	NetworkConfiguration	
regenerate_ssh_keys	ブール値	
timezone	String	
users	User[]	

7.28. CLUSTER 構造体

クラスターのタイプの表現。

クラスターの JSON 表現

```
{
  "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"
} ]
} ]
} ]
}

```

表7.37 属性の概要

Name	タイプ	概要
ballooning_enabled	ブール値	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
cpu	Cpu	

Name	タイプ	概要
custom_scheduling_policy_properties	Property[]	クラスターのカスタムスケジューリングポリシーのプロパティ。
description	String	プレーンテキストでの人間が判読できる説明。
display	表示	
error_handling	ErrorHandling	
fencing_policy	FencingPolicy	クラスターに対してカスタムフェンシングポリシーを定義できます。
gluster_service	ブール値	
gluster_tuned_profile	String	https://fedorahosted の名前。
ha_reservation	ブール値	
id	String	一意の ID
ksm	Ksm	
maintenance_reason_required	ブール値	
memory_policy	MemoryPolicy	
migration	MigrationOptions	
name	String	人間が判読できるプレーンテキストでの名前。
optional_reason	ブール値	
required_rng_sources	RngSource[]	クラスター内の各ホストに必要な乱数ジェネレーター (RNG) ソースのセット。
serial_number	SerialNumber	
supported_versions	Version[]	
switch_type	SwitchType	指定されたクラスター内のすべてのネットワークで使用されるスイッチのタイプ。

Name	タイプ	概要
threads_as_cores	ブール値	
trusted_service	ブール値	
tunnel_migration	ブール値	
version	バージョン	クラスターの互換バージョン。
virt_service	ブール値	

7.28.1. custom_scheduling_policy_properties

クラスターのカスタムスケジューリングポリシーのプロパティ。これらのオプションのプロパティは、**scheduling_policy** リンクで指定されたスケジューリングポリシーのプロパティをオーバーライドし、この特定のクラスターにのみ適用されます。

たとえば、クラスターのカスタムプロパティを更新するには、以下のリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<cluster>
  <custom_scheduling_policy_properties>
    <property>
      <name>HighUtilization</name>
      <value>70</value>
    </property>
  </custom_scheduling_policy_properties>
</cluster>
```

custom_scheduling_policy_properties 属性を使用した更新操作では、**scheduling_policy** リンクで指定されたスケジューリングポリシーのプロパティは更新されず、この特定のクラスターにのみ反映されます。

7.28.2. fencing_policy

クラスターに対してカスタムフェンシングポリシーを定義できます。

以下に例を示します。

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

リクエスト本文：

```
<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

クラスター内のすべてのホストに設定する `tuned` プロファイルの名前。これは必須ではなく、gluster サービスを使用するクラスターにのみ関連します。

7.28.4. required_rng_sources

クラスター内の各ホストに必要な乱数ジェネレーター (RNG) ソースのセット。

読み取られると、暗黙的な **urandom** (クラスターバージョン 4.1 以降の場合) または **random** (クラスターバージョン 4.0 以前の場合) に加えて、選択された RNG ソースが追加で返されます。書き込まれると、暗黙的な **urandom** と **random** RNG ソースは削除できません。



重要

エンジンのバージョン 4.1 より前では、必要な乱数ジェネレーターのセットは管理者によって完全に制御可能で、**random** ソースを含む任意のソースを追加または削除できます。しかし、バージョン 4.1 以降では、**urandom** と **random** ソースは常にセットの一部となり、削除することはできません。



重要

エンジンバージョン 4.1 では、クラスター内の **random** RNG ソースを互換バージョン 4.1 以降に置き換える新しい RNG ソース **urandom** が導入されています。

7.28.5. version

クラスターの互換バージョン。

このクラスター内のすべてのホストは、少なくともこの互換バージョンをサポートする必要があります。

以下に例を示します。

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

以下を応答します。

```

<cluster>
  ...
  <version>
    <major>4</major>
    <minor>0</minor>

```

```

</version>
...
</cluster>

```

互換バージョンを更新するには、以下を使用します。

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

リクエスト本文は以下のようになります。

```

<cluster>
  <version>
    <major>4</major>
    <minor>1</minor>
  </version>
</cluster>

```

クラスター互換バージョンを更新するには、クラスター内のすべてのホストが新しい互換バージョンをサポートする必要があります。

表7.38 リンクの概要

Name	タイプ	概要
affinity_groups	AffinityGroup[]	
cpu_profiles	CpuProfile[]	
data_center	DataCenter	
gluster_hooks	GlusterHook[]	
gluster_volumes	GlusterVolume[]	
mac_pool	MacPool	このクラスターによって使用される MAC プールへの参照。
management_network	Network	
network_filters	NetworkFilter[]	
networks	Network[]	
permissions	Permission[]	
scheduling_policy	SchedulingPolicy	このクラスターによって使用されるデフォルトのスケジューリングポリシーへの参照。

7.28.6. scheduling_policy

このクラスターによって使用されるデフォルトのスケジューリングポリシーへの参照。



注記

スケジューリングポリシープロパティは、デフォルトで参照されるスケジューリングポリシーから取得されますが、このクラスターの **custom_scheduling_policy_properties** 属性で指定されたプロパティで上書きされません。

7.29. CLUSTERLEVEL 構造体

特定のクラスターレベルでサポートされる機能について説明します。

表7.39 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
cpu_types	CpuType[]	このクラスターレベルでサポートされている CPU タイプ。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
permits	Permit[]	このクラスターレベルでサポートされている permits。

7.30. CONFIGURATION 構造体

表7.40 属性の概要

Name	タイプ	概要
data	String	仮想マシンを記述するドキュメント。
type	ConfigurationType	

7.30.1. data

仮想マシンを記述するドキュメント。

OVF ドキュメントの例:

```
<?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>
  <IsInitalized>>false</IsInitalized>
  <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:Instanceld>1</rasd:Instanceld>
      <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:Instanceld>2</rasd:Instanceld>
<rasd:ResourceType>4</rasd:ResourceType>
<rasd:AllocationUnits>MegaBytes</rasd:AllocationUnits>
<rasd:VirtualQuantity>1024</rasd:VirtualQuantity>
</Item>
<Item>
<rasd:Caption>USB Controller</rasd:Caption>
<rasd:Instanceld>3</rasd:Instanceld>
<rasd:ResourceType>23</rasd:ResourceType>
<rasd:UsbPolicy>DISABLED</rasd:UsbPolicy>
</Item>
</Section>
</Content>
</ovf:Envelope>

```

7.31. CONFIGURATIONTYPE ENUM

表7.41 値の概要

Name	概要
ovf	

7.32. CONSOLE 構造体

表7.42 属性の概要

Name	タイプ	概要
enabled	ブール値	

7.33. CORE 構造体

表7.43 属性の概要

Name	タイプ	概要
index	Integer	
socket	Integer	

7.34. CPU 構造体

表7.44 属性の概要

Name	タイプ	概要
architecture	アーキテクチャー	
cores	Core[]	
cpu_tune	CpuTune	
level	Integer	
mode	CpuMode	
name	String	
speed	10 進数	
topology	CpuTopology	
type	String	

7.35. CPUMODE ENUM

表7.45 値の概要

Name	概要
custom	
host_model	
host_passthrough	

7.36. CPUPROFILE 構造体

表7.46 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID

Name	タイプ	概要
name	String	人間が判読できるプレーンテキストでの名前。

表7.47 リンクの概要

Name	タイプ	概要
cluster	Cluster	
permissions	Permission[]	
qos	Qos	

7.37. CPUTOPOLOGY 構造体

表7.48 属性の概要

Name	タイプ	概要
cores	Integer	
sockets	Integer	
threads	Integer	

7.38. CPUTUNE 構造体

表7.49 属性の概要

Name	タイプ	概要
vcpu_pins	VcpuPin[]	

7.39. CPUTYPE 構造体

サポートされている CPU タイプを記述します。

表7.50 属性の概要

Name	タイプ	概要
architecture	アーキテクチャー	CPU のアーキテクチャー。
level	Integer	CPU タイプのレベル。

Name	タイプ	概要
name	String	CPU タイプの名前（例： Intel Conroe Family ）。

7.40. CREATIONSTATUS ENUM

表7.51 値の概要

Name	概要
完了	
failed	
in_progress	
pending	

7.41. CUSTOMPROPERTY 構造体

表7.52 属性の概要

Name	タイプ	概要
name	String	
regexp	String	
value	String	

7.42. DATACENTER 構造体

表7.53 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
local	ブール値	
name	String	人間が判読できるプレーンテキストでの名前。

Name	タイプ	概要
quota_mode	QuotaModeType	
status	DataCenterStatus	
storage_format	StorageFormat	
supported_versions	Version[]	
version	バージョン	データセンターの互換バージョン。

7.42.1. version

データセンターの互換バージョン。

このデータセンター内のすべてのクラスターは、少なくともこの互換バージョンに設定されている必要があります。

以下に例を示します。

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

以下を応答します。

```
<data_center>
...
<version>
  <major>4</major>
  <minor>0</minor>
</version>
...
</data_center>
```

互換バージョンを更新するには、以下を使用します。

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

リクエスト本文は以下のようになります。

```
<data_center>
  <version>
    <major>4</major>
    <minor>1</minor>
  </version>
</data_center>
```

表7.54 リンクの概要

Name	タイプ	概要
clusters	Cluster[]	このデータセンター内のクラスターへの参照。
iscsi_bonds	IscsiBond[]	このデータセンターによって使用される ISCSI ボンディングへの参照。
mac_pool	MacPool	このデータセンターによって使用される MAC プールへの参照。
networks	Network[]	このデータセンターに接続されたネットワークへの参照。
permissions	Permission[]	このデータセンターに割り当てられたパーミッションへの参照。
qoss	qos[]	このデータセンターが使用する QoS への参照。
quotas	Quota[]	このデータセンターに割り当てられたクォータへの参照。
storage_domains	StorageDomain[]	このデータセンターに割り当てられたストレージドメインへの参照。

7.43. DATACENTERSTATUS ENUM

表7.55 値の概要

Name	概要
contend	
maintenance	
not_operational	
problematic	
uninitialized	
up	

7.44. DEVICE 構造体

デバイスは、デバイスの潜在的な親へのリンクをラップします。

表7.56 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.57 リンクの概要

Name	タイプ	概要
instance_type	InstanceType	オプションで、デバイスが使用するインスタンスタイプへの参照。
template	Template	オプションで、デバイスが使用するテンプレートへの参照。
vm	Vm	この要素は使用せず、代わりに vms を使用してください。
vms	Vm[]	このデバイスを使用する仮想マシンへの参照。

7.44.1. vms

このデバイスを使用する仮想マシンへの参照。デバイスは複数の仮想マシンで使用できます。たとえば、共有ディスクは2つ以上の仮想マシンで同時に使用することができます。

7.45. DISK 構造体

仮想ディスクデバイスを表します。

表7.58 属性の概要

Name	タイプ	概要
active	ブール値	ディスクが仮想マシンから見えるかどうかを示します。
actual_size	Integer	ディスクの実際のサイズ (バイト単位)。
alias	String	
bootable	ブール値	ディスクが起動可能としてマークされているかどうかを示します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。

Name	タイプ	概要
description	String	プレーンテキストでの人間が判読できる説明。
format	DiskFormat	基礎となるストレージフォーマット。
id	String	一意の ID
image_id	String	
initial_size	Integer	ブロックストレージで作成された sparse イメージディスクの初期サイズ (バイト単位)。
interface	DiskInterface	ディスクデバイスを仮想マシンに接続するために使用されるインターフェイスドライバのタイプ。
logical_name	String	
lun_storage	HostStorage	
name	String	人間が判読できるプレーンテキストでの名前。
propagate_errors	ブール値	ディスクエラーによって仮想マシンを一時停止すべきでないかどうかを示します。代わりに、ディスクエラーをゲストオペレーティングシステムに伝播する必要があるかどうかを示します。
provisioned_size	Integer	ディスクの仮想サイズ (バイト単位)。
qcow_version	QcowVersion	QCOW ボリュームの基礎となる QCOW バージョン。
read_only	ブール値	ディスクが読み取り専用モードかどうかを示します。
sgio	ScsiGenericIO	
shareable	ブール値	ディスクを複数の仮想マシンに接続できるかどうかを示します。
sparse	ブール値	ディスクの物理ストレージの事前割り当てを避けるべきかどうかを示します。
status	DiskStatus	ディスクデバイスのステータス。
storage_type	DiskStorageType	

Name	タイプ	概要
uses_scsi_reservation	ブール値	
wipe_after_delete	ブール値	削除後にディスクのブロックがゼロとして読み戻されるかどうかを示します。 - ブロックストレージでは、ディスクはゼロにならなければ削除できません。

7.45.1. active

ディスクが仮想マシンから見えるかどうかを示します。



重要

仮想マシンにディスクアタッチメントを追加する場合、サーバーはこの属性を含まない要求を受け入れますが、効果は定義されません。場合によっては、ディスクが自動的にアクティベートされ、その他の場合は自動的にアクティブになりません。問題を回避するには、常に希望の値でこの属性をふくめることを強く推奨します。

7.45.2. actual_size

ディスクの実際のサイズ (バイト単位)。

実際のサイズは、ディスクによって実際に使用されるバイト数であり、**cow** 形式を使用するディスクのプロビジョニングされたサイズよりも小さくなります。

7.45.3. bootable

ディスクが起動可能としてマークされているかどうかを示します。



重要

この属性は、実際に仮想マシンに接続されているディスクに対してのみ意味があり、API のバージョン 4 では、[DiskAttachment](#) タイプに移動されています。ここでは後方互換性のためにのみ保持されており、将来的には削除される予定です。

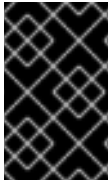
7.45.4. initial_size

ブロックストレージで作成された sparse イメージディスクの初期サイズ (バイト単位)。

初期サイズは、sparse ディスクがブロックストレージ上に作成されたときに最初に割り当てられるバイト数です。初期サイズは、プロビジョニングされたサイズよりも小さくなります。指定されていない場合は、システムで使用されるデフォルトの初期サイズが割り当てられます。

7.45.5. interface

ディスクデバイスを仮想マシンに接続するために使用されるインターフェイスドライバーのタイプ。



重要

この属性は、実際に仮想マシンに接続されているディスクに対してのみ意味があり、API のバージョン 4 では、[DiskAttachment](#) タイプに移動されています。ここでは後方互換性のためにのみ保持されており、将来的には削除される予定です。

7.45.6. provisioned_size

ディスクの仮想サイズ (バイト単位)。

新しいディスクを作成する場合、この属性は必須です。

7.45.7. qcow_version

QCOW ボリュームの基礎となる QCOW バージョン。QCOW バージョンは、ボリュームがサポートする qemu バージョンを qemu に指定します。このフィールドは更新 API を使用して更新でき、QCOW ボリュームについてのみ報告されます。これは、ディスクが作成されたストレージドメインのバージョンによって決まります。V4 未満のバージョンのストレージドメインは QCOW2 ボリュームをサポートしますが、V4 ストレージドメインは QCOW2v3 もサポートしています。異なる QCOW バージョンの機能の詳細は、[こちら](#) を参照してください。

7.45.8. shareable

ディスクを複数の仮想マシンに接続できるかどうかを示します。



重要

ディスクが複数の仮想マシンに接続されている場合、[GlusterFS](#) や [GFS](#) などの共有ファイルシステムを使用するなど、データの破損を回避するために、それらの仮想マシンのゲストオペレーティングシステムがアクセスを調整する責任があります。

7.45.9. wipe_after_delete

削除後にディスクのブロックがゼロとして読み戻されるかどうかを示します。

- ブロックストレージでは、ディスクはゼロにならないと削除できません。
- ファイルストレージでは、以前に削除されたブロックがゼロとして読み戻されることがファイルシステムによって既に保証されているため、ディスクはすぐに削除されます。

表7.59 リンクの概要

Name	タイプ	概要
disk_profile	DiskProfile	
instance_type	InstanceType	オプションで、デバイスが使用するインスタンスタイプへの参照。
openstack_volume_type	OpenStackVolumeType	
permissions	Permission[]	

Name	タイプ	概要
quota	クォータ	
snapshot	スナップショット	
statistics	Statistic[]	ディスクによって公開される統計。
storage_domain	StorageDomain	
storage_domains	StorageDomain[]	このディスクに関連付けられているストレージドメイン。
template	Template	オプションで、デバイスが使用するテンプレートへの参照。
vm	Vm	この要素は使用せず、代わりに vms を使用してください。
vms	Vm[]	このデバイスを使用する仮想マシンへの参照。

7.45.10. statistics

ディスクによって公開される統計。以下に例を示します。

```
<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>
```

これらの統計は、ディスクの取得時に直接含まれず、リンクのみが含まれます。統計を取得するには、以下のリンクを参照してください。

```
GET /ovirt-engine/api/disks/123/statistics
```

7.45.11. storage_domains

このディスクに関連付けられているストレージドメイン。



注記

テンプレートから作成されたのではない仮想マシンに最初のディスクを追加する場合にのみ必要です。

7.45.12. vms

このデバイスを使用する仮想マシンへの参照。デバイスは複数の仮想マシンで使用できます。たとえば、共有ディスクは2つ以上の仮想マシンで同時に使用することができます。

7.46. DISKATTACHMENT 構造体

ディスクを仮想マシンに接続する方法について記述します。

表7.60 属性の概要

Name	タイプ	概要
active	ブール値	ディスクが接続されている仮想マシンでディスクがアクティブかどうかを定義します。
bootable	ブール値	ディスクが起動可能かどうかを定義します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
interface	DiskInterface	ディスクデバイスを仮想マシンに接続するために使用されるインターフェイスドライバーのタイプ。
logical_name	String	仮想マシンの内部から見た、仮想マシンのディスクの論理名。
name	String	人間が判読できるプレーンテキストでの名前。
pass_discard	ブール値	仮想マシンが破棄コマンドをストレージに渡すかどうかを定義します。
read_only	ブール値	ディスクが読み取り専用として仮想マシンに接続されているかどうかを示します。
uses_scsi_reservation	ブール値	このディスクに対して SCSI 予約を有効にするかどうかを定義します。

7.46.1. active

ディスクが接続されている仮想マシンでディスクがアクティブかどうかを定義します。

アクティブステータスの仮想マシンにアタッチされているディスクは、実行時に仮想マシンに接続され、使用することができます。

7.46.2. logical_name

仮想マシンの内部から見た、仮想マシンのディスクの論理名。

ディスクの論理名は、ゲストエージェントが仮想マシン内にインストールされて実行されている場合にのみ報告されます。

たとえば、ゲストオペレーティングシステムが Linux で、ディスクが VirtIO インターフェイスを介して接続されている場合、論理名は `/dev/vda` として報告されます。

```
<disk_attachment>
...
<logical_name>/dev/vda</logical_name>
</disk_attachment>
```

ゲストオペレーティングシステムが Windows の場合、論理名は `\\.\PHYSICALDRIVE0` として報告されます。

7.46.3. read_only

ディスクが読み取り専用として仮想マシンに接続されているかどうかを示します。

新しいディスクアタッチメントを追加する場合、デフォルト値は **false** です。

```
<disk_attachment>
...
<read_only>true</read_only>
</disk_attachment>
```

7.46.4. uses_scsi_reservation

このディスクに対して SCSI 予約を有効にするかどうかを定義します。

VIRTIO-SCSI パススルーが有効になっている仮想マシンは、ディスクに永続的な SCSI 予約を設定できます。永続的な SCSI 予約を設定すると、それらの仮想マシンを別のホストに移行できなくなります。これは、SCSI 予約が SCSI イニシエーター、つまりホストに固有であるため、ディスクへのアクセスが失われるからです。このシナリオは自動的に検出できません。これらの仮想マシンが移行されないようにするために、ユーザーはこの属性を **true** に設定して、仮想マシンが SCSI 予約を使用していることを示すことができます。

表7.61 リンクの概要

Name	タイプ	概要
disk	ディスク	ディスクへの参照。
template	Template	テンプレートへの参照。
vm	Vm	仮想マシンへの参照。

7.47. DISKFORMAT ENUM

ディスクの基礎となるストレージフォーマット。

表7.62 値の概要

Name	概要
cow	Copy On Write フォーマットでは、パフォーマンスのオーバーヘッドが小さいスナップショットが可能です。
raw	raw フォーマットではスナップショットは許可されませんが、パフォーマンスが向上します。

7.48. DISKINTERFACE ENUM

表7.63 値の概要

Name	概要
ide	
spapr_vscsi	
virtio	
virtio_scsi	

7.49. DISKPROFILE STRUCT

表7.64 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.65 リンクの概要

Name	タイプ	概要
permissions	Permission[]	

Name	タイプ	概要
qos	Qos	
storage_domain	StorageDomain	

7.50. DISKSNAPSHOT 構造体

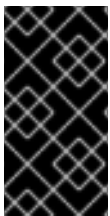
表7.66 属性の概要

Name	タイプ	概要
active	ブール値	ディスクが仮想マシンから見えるかどうかを示します。
actual_size	Integer	ディスクの実際のサイズ (バイト単位)。
alias	String	
bootable	ブール値	ディスクが起動可能としてマークされているかどうかを示します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
format	DiskFormat	基礎となるストレージフォーマット。
id	String	一意の ID
image_id	String	
initial_size	Integer	ブロックストレージで作成された sparse イメージディスクの初期サイズ (バイト単位)。
interface	DiskInterface	ディスクデバイスを仮想マシンに接続するために使用されるインターフェイスドライバのタイプ。
logical_name	String	
lun_storage	HostStorage	
name	String	人間が判読できるプレーンテキストでの名前。
propagate_errors	ブール値	ディスクエラーによって仮想マシンを一時停止すべきでないかどうかを示します。代わりに、ディスクエラーをゲストオペレーティングシステムに伝播する必要があるかどうかを示します。

Name	タイプ	概要
provisioned_size	Integer	ディスクの仮想サイズ (バイト単位)。
qcow_version	QcowVersion	QCOW ボリュームの基礎となる QCOW バージョン。
read_only	ブール値	ディスクが読み取り専用モードかどうかを示します。
sgio	ScsiGenericIO	
shareable	ブール値	ディスクを複数の仮想マシンに接続できるかどうかを示します。
sparse	ブール値	ディスクの物理ストレージの事前割り当てを避けるべきかどうかを示します。
status	DiskStatus	ディスクデバイスのステータス。
storage_type	DiskStorageType	
uses_scsi_reservation	ブール値	
wipe_after_delete	ブール値	削除後にディスクのブロックがゼロとして読み戻されるかどうかを示します。 - ブロックストレージでは、ディスクはゼロにならなければ削除できません。

7.50.1. active

ディスクが仮想マシンから見えるかどうかを示します。



重要

仮想マシンにディスクアタッチメントを追加する場合、サーバーはこの属性を含まない要求を受け入れますが、効果は定義されません。場合によっては、ディスクが自動的にアクティベートされ、その他の場合は自動的にアクティブになりません。問題を回避するには、常に希望の値でこの属性をふくめることを強く推奨します。

7.50.2. actual_size

ディスクの実際のサイズ (バイト単位)。

実際のサイズは、ディスクによって実際に使用されるバイト数であり、**cow** 形式を使用するディスクのプロビジョニングされたサイズよりも小さくなります。

7.50.3. bootable

ディスクが起動可能としてマークされているかどうかを示します。



重要

この属性は、実際に仮想マシンに接続されているディスクに対してのみ意味があり、API のバージョン 4 では、[DiskAttachment](#) タイプに移動されています。ここでは後方互換性のためにのみ保持されており、将来的には削除される予定です。

7.50.4. initial_size

ブロックストレージで作成された sparse イメージディスクの初期サイズ (バイト単位)。

初期サイズは、sparse ディスクがブロックストレージ上に作成されたときに最初に割り当てられるバイト数です。初期サイズは、プロビジョニングされたサイズよりも小さくなります。指定されていない場合は、システムで使用されるデフォルトの初期サイズが割り当てられます。

7.50.5. interface

ディスクデバイスを仮想マシンに接続するために使用されるインターフェイスドライバーのタイプ。



重要

この属性は、実際に仮想マシンに接続されているディスクに対してのみ意味があり、API のバージョン 4 では、[DiskAttachment](#) タイプに移動されています。ここでは後方互換性のためにのみ保持されており、将来的には削除される予定です。

7.50.6. provisioned_size

ディスクの仮想サイズ (バイト単位)。

新しいディスクを作成する場合、この属性は必須です。

7.50.7. qcow_version

QCOW ボリュームの基礎となる QCOW バージョン。QCOW バージョンは、ボリュームがサポートする qemu バージョンを qemu に指定します。このフィールドは更新 API を使用して更新でき、QCOW ボリュームについてのみ報告されます。これは、ディスクが作成されたストレージドメインのバージョンによって決まります。V4 未満のバージョンのストレージドメインは QCOW2 ボリュームをサポートしますが、V4 ストレージドメインは QCOW2v3 もサポートしています。異なる QCOW バージョンの機能の詳細は、[こちら](#) を参照してください。

7.50.8. shareable

ディスクを複数の仮想マシンに接続できるかどうかを示します。



重要

ディスクが複数の仮想マシンに接続されている場合、[GlusterFS](#) や [GFS](#) などの共有ファイルシステムを使用するなど、データの破損を回避するために、それらの仮想マシンのゲストオペレーティングシステムがアクセスを調整する責任があります。

7.50.9. wipe_after_delete

削除後にディスクのブロックがゼロとして読み戻されるかどうかを示します。

- ブロックストレージでは、ディスクはゼロにならないと削除できません。
- ファイルストレージでは、以前に削除されたブロックがゼロとして読み戻されることがファイルシステムによって既に保証されているため、ディスクはすぐに削除されます。

表7.67 リンクの概要

Name	タイプ	概要
disk	ディスク	
disk_profile	DiskProfile	
instance_type	InstanceType	オプションで、デバイスが使用するインスタンスタイプへの参照。
openstack_volume_type	OpenStackVolumeType	
permissions	Permission[]	
quota	クォータ	
snapshot	スナップショット	
statistics	Statistic[]	ディスクによって公開される統計。
storage_domain	StorageDomain	
storage_domains	StorageDomain[]	このディスクに関連付けられているストレージドメイン。
template	Template	オプションで、デバイスが使用するテンプレートへの参照。
vm	Vm	この要素は使用せず、代わりに vms を使用してください。
vms	Vm[]	このデバイスを使用する仮想マシンへの参照。

7.50.10. statistics

ディスクによって公開される統計。以下に例を示します。

```
<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>

```

これらの統計は、ディスクの取得時に直接含まれず、リンクのみが含まれます。統計を取得するには、以下のリンクを参照してください。

```
GET /ovirt-engine/api/disks/123/statistics
```

7.50.11. storage_domains

このディスクに関連付けられているストレージドメイン。



注記

テンプレートから作成されたのではない仮想マシンに最初のディスクを追加する場合にのみ必要です。

7.50.12. vms

このデバイスを使用する仮想マシンへの参照。デバイスは複数の仮想マシンで使用できます。たとえば、共有ディスクは2つ以上の仮想マシンで同時に使用することができます。

7.51. DISKSTATUS ENUM

表7.68 値の概要

Name	概要
illegal	
locked	
ok	

7.52. DISKSTORAGETYPE ENUM

表7.69 値の概要

Name	概要
cinder	

Name	概要
image	
lun	

7.53. DISKTYPE ENUM

表7.70 値の概要

Name	概要
data	
system	

7.54. DISPLAY 構造体

表7.71 属性の概要

Name	タイプ	概要
address	String	
allow_override	ブール値	
certificate	証明書	
copy_paste_enabled	ブール値	
disconnect_action	String	
file_transfer_enabled	ブール値	
keyboard_layout	String	
monitors	Integer	
port	Integer	
proxy	String	

Name	タイプ	概要
secure_port	Integer	
single_qxl_pci	ブール値	
smartcard_enabled	ブール値	
type	DisplayType	

7.55. DISPLAYTYPE ENUM

表7.72 値の概要

Name	概要
spice	
vnc	

7.56. DNS 構造体

DNS リゾルバー設定を表します。

表7.73 属性の概要

Name	タイプ	概要
search_domains	Host[]	検索ドメインとして機能するホストの配列。
servers	Host[]	DNS サーバーとして機能するホストの配列。

7.57. DNSRESOLVERCONFIGURATION 構造体

DNS リゾルバー設定を表します。

表7.74 属性の概要

Name	タイプ	概要
name_servers	String[]	ネームサーバーのアドレスの配列。

7.57.1. name_servers

ネームサーバーのアドレスの配列。IPv4 または IPv6 アドレスのいずれかを指定できます。

7.58. DOMAIN 構造体

このタイプは、ディレクトリーサービスドメインを表します。

表7.75 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
user	User	

表7.76 リンクの概要

Name	タイプ	概要
groups	Group[]	ディレクトリーサービス内のすべてのグループへの参照。
users	User[]	ディレクトリーサービス内のすべてのユーザー一覧への参照。

7.58.1. users

ディレクトリーサービス内のすべてのユーザー一覧への参照。この情報は、新しいユーザーを Red Hat Virtualization 環境に追加するために使用されます。

7.59. ENTITYEXTERNALSTATUS ENUM

外部エンティティのステータスを表すタイプ。

表7.77 値の概要

Name	概要
error	外部エンティティのステータスに誤りがあります。
failure	外部エンティティに障害の原因となる問題があります。
info	外部エンティティのステータスは問題ありませんが、関連する可能性のある情報がいくつかあります。

Name	概要
ok	外部エンティティのステータスは問題ありません。
warning	外部エンティティのステータスは問題ありませんが、注意が必要な問題があります。

7.59.1. error

外部エンティティのステータスに誤りがあります。これには、適度な注意が必要な場合があります。

7.59.2. failure

外部エンティティに障害の原因となる問題があります。すぐに注意が必要な場合があります。

7.60. ENTITYPROFILEDETAIL STRUCT

表7.78 属性の概要

Name	タイプ	概要
profile_details	ProfileDetail[]	

7.61. ERRORHANDLING 構造体

表7.79 属性の概要

Name	タイプ	概要
on_error	MigrateOnError	

7.62. EVENT 構造体

イベントを表すタイプ。

表7.80 属性の概要

Name	タイプ	概要
code	Integer	イベントコード。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
correlation_id	String	イベント相関識別子。
custom_data	String	カスタムイベントデータを表すフリーテキスト。

Name	タイプ	概要
custom_id	Integer	カスタムイベント識別子。
description	String	プレーンテキストでの人間が判読できる説明。
flood_rate	Integer	フラッドレートを定義します。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
origin	String	イベントの発生源を特定するフリーテキスト。
severity	LogSeverity	イベントの重大度。
time	日付	イベント時間。

7.62.1. correlation_id

イベント相関識別子。複数のイベントを相互に関連付けるために使用されます。

7.62.2. flood_rate

フラッドレートを定義します。これにより、定義されたレートでイベントが複数回発生した場合のフラッシングが防止されます。デフォルトは 30 秒です。

表7.81 リンクの概要

Name	タイプ	概要
cluster	Cluster	クラスターサービスへの参照。
data_center	DataCenter	データセンターサービスへの参照。
host	ホスト	ホストサービスへの参照。
storage_domain	StorageDomain	ストレージドメインサービスへの参照。
template	Template	テンプレートサービスへの参照。
user	User	ユーザーサービスへの参照。
vm	Vm	仮想マシンサービスへの参照。

7.62.3. cluster

クラスターサービスへの参照。イベントはクラスターに関連付けることができます。

7.62.4. data_center

データセンターサービスへの参照。イベントはデータセンターに関連付けることができます。

7.62.5. host

ホストサービスへの参照。イベントはホストに関連付けることができます。

7.62.6. storage_domain

ストレージドメインサービスへの参照。イベントは、ストレージドメインに関連付けることができます。

7.62.7. template

テンプレートサービスへの参照。イベントはテンプレートに関連付けることができます。

7.62.8. user

ユーザーサービスへの参照。イベントはユーザーに関連付けることができます。

7.62.9. vm

仮想マシンサービスへの参照。イベントは仮想マシンに関連付けることができます。

7.63. EXTERNALCOMPUTERESOURCE STRUCT

表7.82 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
provider	String	
url	String	
user	String	

表7.83 リンクの概要

Name	タイプ	概要
external_host_provider	ExternalHostProvider	

7.64. EXTERNALDISCOVEREDHOST STRUCT

表7.84 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
ip	String	
last_report	String	
mac	String	
name	String	人間が判読できるプレーンテキストでの名前。
subnet_name	String	

表7.85 リンクの概要

Name	タイプ	概要
external_host_provider	ExternalHostProvider	

7.65. EXTERNALHOST 構造体

表7.86 属性の概要

Name	タイプ	概要
address	String	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。

Name	タイプ	概要
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.87 リンクの概要

Name	タイプ	概要
external_host_provider	ExternalHostProvider	

7.66. EXTERNALHOSTGROUP STRUCT

表7.88 属性の概要

Name	タイプ	概要
architecture_name	String	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
domain_name	String	
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
operating_system_name	String	
subnet_name	String	

表7.89 リンクの概要

Name	タイプ	概要
external_host_provider	ExternalHostProvider	

7.67. EXTERNALHOSTPROVIDER STRUCT

表7.90 属性の概要

Name	タイプ	概要
authentication_url	String	外部プロバイダー認証 URL アドレスを定義します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
password	String	認証プロセス中にユーザーのパスワードを定義します。
properties	Property[]	プロバイダー名/値のプロパティの配列。
requires_authentication	ブール値	プロバイダー認証が必要かどうかを定義します。
url	String	外部プロバイダーの URL アドレスを定義します。
username	String	認証プロセス中に使用されるユーザー名を定義します。

7.67.1. requires_authentication

プロバイダー認証が必要かどうかを定義します。

認証が必要な場合、**username** と **password** の両方の属性が認証時に使用されます。

表7.91 リンクの概要

Name	タイプ	概要
certificates	Certificate[]	
compute_resources	ExternalComputeResource[]	
discovered_hosts	ExternalDiscoveredHost[]	
host_groups	ExternalHostGroup[]	
hosts	Host[]	

7.68. EXTERNALPROVIDER 構造体

外部プロバイダーを表します。

表7.92 属性の概要

Name	タイプ	概要
authentication_url	String	外部プロバイダー認証 URL アドレスを定義します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
password	String	認証プロセス中にユーザーのパスワードを定義します。
properties	Property[]	プロバイダー名/値のプロパティの配列。
requires_authentication	ブール値	プロバイダー認証が必要かどうかを定義します。
url	String	外部プロバイダーの URL アドレスを定義します。
username	String	認証プロセス中に使用されるユーザー名を定義します。

7.68.1. requires_authentication

プロバイダー認証が必要かどうかを定義します。

認証が必要な場合、**username** と **password** の両方の属性が認証時に使用されます。

7.69. EXTERNALSTATUS ENUM

表7.93 値の概要

Name	概要
error	
failure	
info	

Name	概要
ok	
warning	

7.70. EXTERNALSYSTEMTYPE ENUM

step に関連する外部システムの種類を表す。

表7.94 値の概要

Name	概要
gluster	step に関連する外部システムとして Gluster を表現する。
vdsm	step に関連する外部システムとして VDSM を表現する。

7.71. EXTERNALVMIMPORT STRUCT

外部システムからの仮想マシンのインポート操作のパラメーターについて説明します。

表7.95 属性の概要

Name	タイプ	概要
name	String	外部システム内で定義されている、インポートする仮想マシンの名前。
password	String	外部ハイパーバイザーシステムに対して認証するためのパスワード。
provider	ExternalVmProviderType	外部仮想マシンプロバイダーのタイプ。
sparse	ブール値	結果の仮想マシンのディスク割り当てポリシーを指定します。sparse の場合は true 、事前割り当ての場合は false になります。
url	String	変換のために virt-v2v ツールに渡される URL。
username	String	外部ハイパーバイザーシステムに対して認証するためのユーザー名。

7.71.1. url

変換のために **virt-v2v** ツールに渡される URL。

以下に例を示します。

```
vpix://wmware_user@vcenter-host/DataCenter/Cluster/esxi-host?no_verify=1
```

その他の例は、<http://libguestfs.org/virt-v2v.1.html> を参照してください。

表7.96 リンクの概要

Name	タイプ	概要
cluster	Cluster	結果の仮想マシンのターゲットクラスターを指定します。
cpu_profile	CpuProfile	オプション。
drivers_iso	File	オプション。
host	ホスト	オプション。
quota	クォータ	オプション。
storage_domain	StorageDomain	変換されたディスクのターゲットストレージドメインを指定します。
vm	Vm	新しく作成された仮想マシンの名前を指定するために使用される仮想マシンエンティティ。

7.71.2. cpu_profile

オプション。結果の仮想マシンの CPU プロファイルを指定します。

7.71.3. drivers_iso

オプション。**virt-v2v** 変換プロセス中に使用できるドライバーを含む ISO の名前。

7.71.4. host

オプション。変換プロセスに使用するホストを (ホスト ID を使用して) 指定します。指定されていない場合は、自動的に選択されます。

7.71.5. quota

オプション。結果の仮想マシンに適用されるクォータを指定します。

7.71.6. vm

新しく作成された仮想マシンの名前を指定するために使用される仮想マシンエンティティ。

名前が指定されていない場合は、ソース仮想マシン名が使用されます。

7.72. EXTERNALVMPROVIDERTYPE ENUM

外部ハイパーバイザーシステムのタイプについて説明します。

表7.97 値の概要

Name	概要
kvm	
vmware	
xen	

7.73. FAULT 構造体

表7.98 属性の概要

Name	タイプ	概要
detail	String	
reason	String	

7.74. FENCETYPE ENUM

フェンス操作のタイプを表すタイプ。

表7.99 値の概要

Name	概要
manual	電源管理を介した手動のホストフェンシング。
restart	電源管理を介してホストを再起動します。
start	電源管理を介してホストを起動します。
status	電源管理を介してホストの電源ステータスを確認します。
stop	電源管理を介してホストを停止します。

7.75. FENCINGPOLICY 構造体

クラスターフェンシングポリシーを表すタイプ。

表7.100 属性の概要

Name	タイプ	概要
enabled	ブール値	このクラスターでフェンシングを有効または無効にします。
skip_if_connectivity_broken	SkipIfConnectivity Broken	有効にすると、クラスター内の設定可能なパーセンテージを超えるホストが接続を失ってもホストのフェンシングは実行されません。
skip_if_gluster_bricks_up	ブール値	フェンシングされているホストで Gluster ブリックが稼働している場合、フェンシングをスキップする必要があるかどうかを示すフラグ。
skip_if_gluster_quorum_not_met	ブール値	Gluster ブリックが稼働中であり、Gluster クォーラムがそれらのブリックなしでは満たされない場合にフェンシングをスキップする必要があるかどうかを示すフラグ。
skip_if_sd_active	SkipIfSdActive	有効にすると、ホストがストレージでリースを維持している場合にフェンシングをスキップします。

7.75.1. skip_if_connectivity_broken

有効にすると、クラスター内の設定可能なパーセンテージを超えるホストが接続を失ってもホストのフェンシングは実行されません。これは、クラスター内でグローバルネットワークの問題が発生した場合にフェンシングストームを防ぐためです。

7.75.2. skip_if_gluster_bricks_up

フェンシングされているホストで Gluster ブリックが稼働している場合、フェンシングをスキップする必要があるかどうかを示すフラグ。このフラグはオプションで、デフォルト値は **false** です。

7.75.3. skip_if_gluster_quorum_not_met

Gluster ブリックが稼働中であり、Gluster クォーラムがそれらのブリックなしでは満たされない場合にフェンシングをスキップする必要があるかどうかを示すフラグ。このフラグはオプションで、デフォルト値は **false** です。

7.75.4. skip_if_sd_active

有効にすると、ホストがストレージでリースを維持している場合にフェンシングをスキップします。これは、ホストがまだストレージにアクセスできる場合、フェンシングされないことを意味します。

7.76. FILE 構造体

表7.101 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。

Name	タイプ	概要
content	String	
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
type	String	

表7.102 リンクの概要

Name	タイプ	概要
storage_domain	StorageDomain	

7.77. FILTER 構造体

表7.103 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
position	Integer	

表7.104 リンクの概要

Name	タイプ	概要
scheduling_policy_unit	SchedulingPolicyUnit	

7.78. FLOPPY 構造体

表7.105 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
file	File	
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.106 リンクの概要

Name	タイプ	概要
instance_type	InstanceType	オプションで、デバイスが使用するインスタンスタイプへの参照。
template	Template	オプションで、デバイスが使用するテンプレートへの参照。
vm	Vm	この要素は使用せず、代わりに vms を使用してください。
vms	Vm[]	このデバイスを使用する仮想マシンへの参照。

7.78.1. vms

このデバイスを使用する仮想マシンへの参照。デバイスは複数の仮想マシンで使用できます。たとえば、共有ディスクは2つ以上の仮想マシンで同時に使用することができます。

7.79. FOPSTATISTICSTRUCT

表7.107 属性の概要

Name	タイプ	概要
name	String	
statistics	Statistic[]	

7.80. GLUSTERBRICK 構造体

表7.108 属性の概要

Name	タイプ	概要
brick_dir	String	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
device	String	
fs_name	String	
gluster_clients	GlusterClient[]	
id	String	一意の ID
memory_pools	GlusterMemoryPool[]	
mnt_options	String	
name	String	人間が判読できるプレーンテキストでの名前。
pid	Integer	
port	Integer	
server_id	String	
status	GlusterBrickStatus	

表7.109 リンクの概要

Name	タイプ	概要
gluster_volume	GlusterVolume	
instance_type	InstanceType	オプションで、デバイスが使用するインスタンスタイプへの参照。
statistics	Statistic[]	
template	Template	オプションで、デバイスが使用するテンプレートへの参照。
vm	Vm	この要素は使用せず、代わりに vms を使用してください。
vms	Vm[]	このデバイスを使用する仮想マシンへの参照。

7.80.1. vms

このデバイスを使用する仮想マシンへの参照。デバイスは複数の仮想マシンで使用できます。たとえば、共有ディスクは2つ以上の仮想マシンで同時に使用することができます。

7.81. GLUSTERBRICKADVANCEDDETAILS STRUCT

表7.110 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
device	String	
fs_name	String	
gluster_clients	GlusterClient[]	
id	String	一意の ID
memory_pools	GlusterMemoryPool[]	
mnt_options	String	
name	String	人間が判読できるプレーンテキストでの名前。
pid	Integer	
port	Integer	

表7.111 リンクの概要

Name	タイプ	概要
instance_type	InstanceType	オプションで、デバイスが使用するインスタンスタイプへの参照。
template	Template	オプションで、デバイスが使用するテンプレートへの参照。
vm	Vm	この要素は使用せず、代わりに vms を使用してください。
vms	Vm[]	このデバイスを使用する仮想マシンへの参照。

7.81.1. vms

このデバイスを使用する仮想マシンへの参照。デバイスは複数の仮想マシンで使用できます。たとえば、共有ディスクは2つ以上の仮想マシンで同時に使用することができます。

7.82. GLUSTERBRICKMEMORYINFO STRUCT

表7.112 属性の概要

Name	タイプ	概要
memory_pools	GlusterMemoryPool[]	

7.83. GLUSTERBRICKSTATUS ENUM

表7.113 値の概要

Name	概要
down	ブリックは down 状態です。データを保存したり、ブリックから取得したりできません。
unknown	ホストが応答していないため、ステータスを判別できない場合。
up	ブリックは up 状態にあり、データを保存したり、ブリックから取得したりできます。

7.84. GLUSTERCLIENT 構造体

表7.114 属性の概要

Name	タイプ	概要
bytes_read	Integer	
bytes_written	Integer	
client_port	Integer	
host_name	String	

7.85. GLUSTERHOOK STRUCT

表7.115 属性の概要

Name	タイプ	概要
checksum	String	

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
conflict_status	Integer	
conflicts	String	
content	String	
content_type	HookContentType	
description	String	プレーンテキストでの人間が判読できる説明。
gluster_command	String	
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
stage	HookStage	
status	GlusterHookStatus	

表7.116 リンクの概要

Name	タイプ	概要
cluster	Cluster	
server_hooks	GlusterServerHook[]	

7.86. GLUSTERHOOKSTATUS ENUM

表7.117 値の概要

Name	概要
disabled	クラスターでフックが無効になっています。
enabled	クラスターでフックが有効になっています。
missing	フックのステータスが不明/欠落しています。

7.87. GLUSTERMEMORYPOOL STRUCT

表7.118 属性の概要

Name	タイプ	概要
alloc_count	Integer	
cold_count	Integer	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
hot_count	Integer	
id	String	一意の ID
max_alloc	Integer	
max_stdalloc	Integer	
name	String	人間が判読できるプレーンテキストでの名前。
padded_size	Integer	
pool_misses	Integer	
type	String	

7.88. GLUSTERSERVERHOOK STRUCT

表7.119 属性の概要

Name	タイプ	概要
checksum	String	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
content_type	HookContentType	
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

Name	タイプ	概要
status	GlusterHookStatus	

表7.120 リンクの概要

Name	タイプ	概要
host	ホスト	

7.89. GLUSTERSTATE ENUM

表7.121 値の概要

Name	概要
down	
unknown	
up	

7.90. GLUSTERVOLUME 構造体

表7.122 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
disperse_count	Integer	
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
options	Option[]	
redundancy_count	Integer	
replica_count	Integer	

Name	タイプ	概要
status	GlusterVolumeStatus	
stripe_count	Integer	
transport_types	TransportType[]	
volume_type	GlusterVolumeType	

表7.123 リンクの概要

Name	タイプ	概要
bricks	GlusterBrick[]	
cluster	Cluster	
statistics	Statistic[]	

7.91. GLUSTERVOLUMEPROFILEDETAILS STRUCT

表7.124 属性の概要

Name	タイプ	概要
brick_profile_details	BrickProfileDetail[]	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
nfs_profile_details	NfsProfileDetail[]	

7.92. GLUSTERVOLUMESTATUS ENUM

表7.125 値の概要

Name	概要
down	クライアントがボリュームをマウントして使用できるようにするには、ボリュームを開始する必要があります。
unknown	ホストが応答していないため、ステータスを判別できない場合。
up	ボリュームが開始され、クライアントによるマウントおよび使用が可能です。

7.93. GLUSTERVOLUMETYPE ENUM

Gluster ボリュームのタイプを表すタイプ。

表7.126 値の概要

Name	概要
disperse	分散ボリュームは消去コードに基づいており、ディスクまたはサーバーの障害に対するスペース効率の高い保護を提供します。
distribute	分散ボリュームは、ボリューム内のブリック全体にファイルを分散します。
distributed_disperse	Distributed dispersed ボリュームは、分散サブボリューム全体にファイルを分散します。
distributed_replicate	Distributed replicated ボリュームは、ボリューム内の複製ブリック全体にファイルを分散します。
distributed_stripe	Distributed striped ボリュームは、クラスター内の2つ以上のノードにデータをストライプします。
distributed_stripped_replicate	Distributed striped replicated ボリュームは、クラスター内の複製ブリック全体にストライプデータを分散します。
replicate	複製されたボリュームは、ボリューム内のブリック全体でファイルをレプリケートします。
stripe	Striped ボリュームは、ボリューム内のブリック全体にデータをストライプします。
striped_replicate	Striped replicated ボリュームは、クラスター内のレプリケートされたブリック全体にデータをストライプします。

7.93.1. disperse

Dispersed ボリュームは消去コードに基づいており、ディスクまたはサーバーの障害に対するスペース効率の高い保護を提供します。

元のファイルのエンコードされたフラグメントを各ブリックに分散させ、元のファイルを復元するためにフラグメントのサブセットのみが必要になるようにします。データへのアクセスを失わずに喪失するブリックの数は、ボリュームの作成時に管理者によって設定されます。

7.93.2. distribute

分散ボリュームは、ボリューム内のブリック全体にファイルを分散します。

Distributed ボリュームは、ストレージのスケーリングを必要とし、冗長性が重要でないか、他のハードウェア/ソフトウェア層によって提供される場合に使用できます。

7.93.3. distributed_disperse

Distributed dispersed ボリュームは、分散サブボリューム全体にファイルを分散します。

これには、distribute replicate ボリュームと同じ利点がありますが、disperse を使用してデータをブリックに格納します。

7.93.4. distributed_replicate

Distributed replicated ボリュームは、ボリューム内の複製ブリック全体にファイルを分散します。

ストレージのスケーリングに必要で、信頼性を高く保つことが重要な環境で、分散レプリケーションボリュームを使用できます。また、分散レプリケートされたボリュームは、ほとんどの環境で読み取りパフォーマンスも向上します。

7.93.5. distributed_stripe

Distributed striped ボリュームは、クラスター内の2つ以上のノードにデータをストライプします。

Distributed striped ボリュームは、ストレージのスケーリングが必要で、非常に大きなファイルにアクセスする同時実行性の高い環境が不可欠な場合に使用する必要があります。

注記: Glusterfs 3.7 リリースでのシャーディングの導入により、striped ボリュームは非推奨となり、今後のリリースで削除される予定です。

7.93.6. distributed_striped_replicate

Distributed striped replicated ボリュームは、クラスター内の複製ブリック全体にストライプデータを分散します。

最良の結果を得るために、非常に大きなファイルへの並列アクセスとパフォーマンスが重要な同時実行性の高い環境では、distributed striped replicated ボリュームを使用する必要があります。

注記: Glusterfs 3.7 リリースでのシャーディングの導入により、striped ボリュームは非推奨となり、今後のリリースで削除される予定です。

7.93.7. replicate

複製されたボリュームは、ボリューム内のブリック全体でファイルをレプリケートします。

レプリケーションされたボリュームは、高可用性と高信頼性が重要な環境で使用できます。

7.93.8. stripe

Striped ボリュームは、ボリューム内のブリック全体にデータをストライプします。

最良の結果を得るには、striped ボリュームは、非常に大きなファイルにアクセスする同時実行性の高い環境でのみ使用する必要があります。

注記: Glusterfs 3.7 リリースでのシャーディングの導入により、striped ボリュームは非推奨となり、今後のリリースで削除される予定です。

7.93.9. striped_replicate

Striped replicated ボリュームは、クラスター内のレプリケートされたブリック全体にデータをストライプします。

最良の結果を得るために、非常に大きなファイルへの並列アクセスが発生し、パフォーマンスが重要な同時実行性の高い環境では、striped replicated ボリュームを使用する必要があります。

注記: Glusterfs 3.7 リリースでのシャーディングの導入により、striped ボリュームは非推奨となり、今後のリリースで削除される予定です。

7.94. GRACEPERIOD 構造体

表7.127 属性の概要

Name	タイプ	概要
expiry	Integer	

7.95. GRAPHICSCONSOLE 構造体

表7.128 属性の概要

Name	タイプ	概要
address	String	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
port	Integer	
protocol	GraphicsType	
tls_port	Integer	

表7.129 リンクの概要

Name	タイプ	概要
instance_type	InstanceType	
template	Template	
vm	Vm	

7.96. GRAPHICSTYPE ENUM

表7.130 値の概要

Name	概要
spice	
vnc	

7.97. GROUP 構造体

このタイプは、ディレクトリーサービス内のすべてのグループを表します。

表7.131 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
domain_entry_id	String	含まれるディレクトリーサービスドメイン ID。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
namespace	String	グループが存在する名前空間。

表7.132 リンクの概要

Name	タイプ	概要
domain	Domain	このグループを含むドメインへのリンク。

Name	タイプ	概要
permissions	Permission[]	このグループにアタッチされたパーミッションのパーミッションサブコレクションへのリンク。
roles	Role[]	このグループにアタッチされたロールのロールサブコレクションへのリンク。
tags	Tag[]	このグループにアタッチされたタグのタグサブコレクションへのリンク。

7.97.1. roles

このグループにアタッチされたロールのロールサブコレクションへのリンク。

新しいグループの最初のロール割り当てを表すためにのみ使用されます。その後、ロール割り当ての変更は、**roles** サブコレクションを介してのみサポートされます。

7.98. GUESTOPERATINGSYSTEM STRUCT

表7.133 属性の概要

Name	タイプ	概要
architecture	String	
codename	String	
distribution	String	
family	String	
kernel	カーネル	
version	バージョン	

7.99. HARDWAREINFORMATION 構造体

表7.134 属性の概要

Name	タイプ	概要
family	String	
manufacturer	String	

Name	タイプ	概要
product_name	String	
serial_number	String	
supported_rng_sources	RngSource[]	
uuid	String	
version	String	

7.100. HIGHAVAILABILITY 構造体

表7.135 属性の概要

Name	タイプ	概要
enabled	ブール値	
priority	Integer	実行および移行キュー内の仮想マシンの優先度を示します。

7.100.1. priority

実行および移行キュー内の仮想マシンの優先度を示します。

優先度の高い仮想マシンは、優先度の低い仮想マシンより先に開始および移行されます。

値は 0 から 100 の整数です。値が高いと優先順位が高くなります。

グラフィカルユーザーインターフェイス (GUI) では、可能なすべての値を指定することはできません。代わりに、**Low**、**Medium** または **High** のいずれかのみ選択できます。API を使用して値を設定すると、GUI は次のようにラベルを設定します。

API 値	GUI ラベル
0 - 25	低
26 - 74	中
75 - 100	高

GUI を使用してラベルを選択すると、API の値は次のように設定されます。

GUI ラベル	API 値
低	1
中	50
高	100

7.101. HOOK 構造体

フックを表します。

表7.136 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
event_name	String	フックを実行するイベントの名前。
id	String	一意の ID
md5	String	フックのチェックサム。
name	String	人間が判読できるプレーンテキストでの名前。

表7.137 リンクの概要

Name	タイプ	概要
host	ホスト	フックが属するホストへの参照。

7.102. HOOKCONTENTTYPE ENUM

フックスクリプトのコンテンツタイプを表します。

表7.138 値の概要

Name	概要
binary	フックのバイナリーコンテンツタイプ。
text	フックのテキストコンテンツタイプ。

7.103. HOOKSTAGE ENUM

表7.139 値の概要

Name	概要
post	
pre	

7.104. HOOKSTATUS ENUM

タイプはフックのステータスを表します。

表7.140 値の概要

Name	概要
disabled	フックが無効になっています。
enabled	フックが有効になっています。
missing	フックがありません。

7.105. HOST 構造体

ホストを表すタイプ。

表7.141 属性の概要

Name	タイプ	概要
address	String	ホストアドレス (FQDN/IP)。
auto_numa_status	AutoNumaStatus	ホストの自動 Non-Uniform Memory Access (NUMA) ステータス。
certificate	証明書	ホスト証明書。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
cpu	Cpu	このホストの CPU タイプ。
description	String	プレーンテキストでの人間が判読できる説明。
device_passthrough	HostDevicePassthrough	このホストでホストデバイスのパススルーが有効であるかどうかを指定します。

Name	タイプ	概要
display	表示	オプションで、このホストの表示アドレスを明示的に指定します。
external_status	ExternalStatus	ホストの外部ステータス。
hardware_information	HardwareInformation	ホストハードウェア情報。
hosted_engine	HostedEngine	このホストのホスト型エンジンのステータス。
id	String	一意の ID
iscsi	IscsiDetails	ホスト iSCSI の詳細。
kdump_status	KdumpStatus	ホストの KDUMP ステータス。
ksm	Ksm	Kernel SamePage Merging(KSM) は、複数の同一ページから1つのページ参照にメモリーページへの参照を削減します。
libvirt_version	バージョン	ホストの libvirt バージョン。
max_scheduling_memory	Integer	このホストの最大スケジューリングメモリー (バイト単位)。
memory	Integer	このホストの物理メモリーの量 (バイト単位)。
name	String	人間が判読できるプレーンテキストでの名前。
numa_supported	ブール値	このホストで Non-Uniform Memory Access (NUMA) をサポートしているかどうかを指定します。
os	OperatingSystem	このホストのオペレーティングシステム。
override_iptables	ブール値	ファイアウォール定義をオーバーライドする必要があるかどうかを指定します。
port	Integer	ホストポート。
power_management	PowerManagement	ホストの電源管理の定義。
protocol	HostProtocol	エンジンがホストとの通信に使用するプロトコル。

Name	タイプ	概要
root_password	String	新しいホストを作成するときに、パスワード認証方法を選択した場合は root パスワードが必要ですが、これはその後の表現には含まれません。
se_linux	SeLinux	ホストの SELinux ステータス。
spm	Spm	ホスト Storage Pool Manager (SPM) のステータスと定義。
ssh	Ssh	SSH 定義。
status	HostStatus	ホストのステータス。
status_detail	String	ホストステータスの詳細。
summary	VmSummary	仮想マシンの概要 - アクティブ数、移行数、および合計数。
transparent_huge_pages	TransparentHugePages	透過的な Huge Page のサポートにより、メモリーページのサイズが標準の 4 KiB の制限を超えて拡張されます。
type	HostType	ホストにオペレーティングシステムのフルインストールが含まれているか、仮想マシンをホストすることのみを目的とした縮小版が含まれているかを示します。
update_available	ブール値	このホストに oVirt 関連の更新があるかどうかを指定します。
version	バージョン	VDSM のバージョン。

7.105.1. external_status

ホストの外部ステータス。これは、サードパーティーのソフトウェアが、問題が発生した場合にホスト外部のステータスを変更するために使用することができます。サードパーティーのソフトウェアがこのステータスをチェックしてそれに応じて行動しない限り、これはホストのライフサイクルに影響を与えません。

7.105.2. kdump_status

ホストの KDUMP ステータス。KDUMP は、ホストカーネルがクラッシュし、メモリーダンプが実行されているときに発生します。

7.105.3. ksm

Kernel SamePage Merging(KSM) は、複数の同一ページから1つのページ参照にメモリーページへの参照を削減します。これは、メモリー密度の最適化に役立ちます。

たとえば、ホスト **123** に対して KSM を有効にするには、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<host>
  <ksm>
    <enabled>true</enabled>
  </ksm>
</host>
```

7.105.4. libvirt_version

ホストの libvirt バージョン。libvirt の詳細は、[libvirt](#) を参照してください。

7.105.5. override_iptables

ファイアウォール定義をオーバーライドする必要があるかどうかを指定します。これは、ホストがインストールまたは再インストールされた場合にのみ適用されます。

7.105.6. protocol

エンジンがホストとの通信に使用するプロトコル。



警告

エンジンのバージョン 4.1 以降、**xml** が削除されたため、プロトコルは常に **stomp** に設定されています。

7.105.7. se_linux

ホストの SELinux ステータス。**Security-Enhanced Linux (SELinux)** は、アクセス制御セキュリティポリシーをサポートするためのメカニズムを提供する Linux カーネルのコンポーネントです。

7.105.8. spm

ホスト **Storage Pool Manager (SPM)** のステータスと定義。これを使用して、このホストの SPM 優先度を設定し、これが現在の SPM であるかどうかを確認します。

7.105.9. status_detail

ホストステータスの詳細。Gluster ホストに関連します。

7.105.10. transparent_huge_pages

透過的な Huge Page のサポートにより、メモリーページのサイズが標準の 4 KiB の制限を超えて拡張されます。これにより、メモリー消費が削減され、ホストのパフォーマンスが向上します。

たとえば、ホスト **123** の透過的な Huge Page のサポートを有効にするには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<host>
  <transparent_hugepages>
    <enabled>true</enabled>
  </transparent_hugepages>
</host>
```

7.105.11. version

VDSM のバージョン。

以下に例を示します。

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

この **GET** リクエストは、以下の出力を返します。

```
<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>
```

表7.142 リンクの概要

Name	タイプ	概要
affinity_labels	AffinityLabel[]	
agents	Agent[]	
cluster	Cluster	
devices	Device[]	
external_host_provider	ExternalHostProvider	
hooks	Hook[]	
katello_errata	KatelloErratum[]	ホストに割り当てられているすべての Katello エラータを一覧表示します。

Name	タイプ	概要
network_attachments	NetworkAttachment[]	
nics	HostNic[]	
numa_nodes	NumaNode[]	
permissions	Permission[]	
statistics	Statistic[]	各ホストリソースは、ホスト固有の統計の統計サブコレクションを公開します。
storage_connection_extensions	StorageConnectionExtension[]	
storages	HostStorage[]	
tags	Tag[]	
unmanaged_networks	UnmanagedNetwork[]	

7.105.12. katello_errata

ホストに割り当てられているすべての Katello エラータを一覧表示します。

```
GET /ovirt-engine/api/hosts/123/katelloerrata
```

以下のような XML で応答を受け取ります。

```
<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

各ホストリソースは、ホスト固有の統計の統計サブコレクションを公開します。

XML 表現の例:

```

<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>

```



注記

この統計サブコレクションは読み取り専用です。

次のリストは、ホストの統計タイプを示しています。

Name	説明
memory.total	ホスト上の合計メモリー (バイト単位)。
memory.used	ホストで使用されるバイト単位のメモリー。
memory.free	ホスト上の空きメモリー (バイト単位)。
memory.shared	ホスト上で共有しているメモリー (バイト単位)。
memory.buffers	バイト単位の I/O バッファ。
memory.cached	OS のキャッシュ (バイト単位)。
swap.total	ホスト上の合計スワップメモリー (バイト単位)。
swap.free	ホスト上の空きスワップメモリー (バイト単位)。

Name	説明
swap.used	ホスト上で使用されているスワップメモリー (バイト単位)。
swap.cached	バイト単位のスワップメモリーもホストのメモリーにキャッシュされます。
ksm.cpu.current	カーネル SamePage マージの CPU 使用率。
cpu.current.user	ユーザースライスの CPU 使用率。
cpu.current.system	システムの CPU 使用率。
cpu.current.idle	アイドル状態の CPU 使用率。
cpu.load.avg.5m	5 分あたりの CPU 負荷の平均。
boot.time	マシンの起動時間。

7.106. HOSTDEVICE 構造体

表7.143 属性の概要

Name	タイプ	概要
capability	String	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
driver	String	このデバイスがバインドされているドライバーの名前。
id	String	一意の ID
iommu_group	Integer	
name	String	人間が判読できるプレーンテキストでの名前。
physical_function	HostDevice	
placeholder	ブール値	
product	製品	

Name	タイプ	概要
vendor	Vendor	
virtual_function_s	Integer	

7.106.1. driver

このデバイスがバインドされているドライバーの名前。

例: **pcieport** または **uhci_hcd**。

表7.144 リンクの概要

Name	タイプ	概要
host	ホスト	
parent_device	HostDevice	
vm	Vm	

7.107. HOSTDEVICEPASSTHROUGH 構造体

表7.145 属性の概要

Name	タイプ	概要
enabled	ブール値	

7.108. HOSTNIC STRUCT

ホスト NIC を表します。

たとえば、ホスト NIC の XML 表現は次のようになります。

```
<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>

```

ボンディングされたインターフェイスは、**bonding** 属性と **slaves** 属性を含む `HostNic` オブジェクトとして表されます。

たとえば、ボンディングされたホスト NIC の XML 表現は次のようになります。

```

<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>
      <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>

```

表7.146 属性の概要

Name	タイプ	概要
ad_aggregator_id	Integer	モード 4 のボンドの場合、ボンドまたはボンドスレーブの ad_aggregator_id プロパティ。

Name	タイプ	概要
base_interface	String	NIC のベースインターフェイス。
bonding	ボンディング	NIC のボンディングパラメーター。
boot_protocol	BootProtocol	NIC の IPv4 ブートプロトコル設定。
bridged	ブール値	ブリッジネットワークのステータスを定義します。
check_connectivity	ブール値	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
custom_configuration	ブール値	
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
ip	Ip	NIC の IPv4 アドレス。
ipv6	Ip	NIC の IPv6 アドレス。
ipv6_boot_protocol	BootProtocol	NIC の IPv6 ブートプロトコル設定。
mac	Mac	NIC の MAC アドレス。
mtu	Integer	インターフェイスの最大伝送単位。
name	String	人間が判読できるプレーンテキストでの名前。
network_labels	NetworkLabel[]	この NIC に適用されるラベル。
override_configuration	ブール値	
properties	Property[]	
speed	Integer	
statistics	Statistic[]	NIC の統計へのリンク。
status	NicStatus	

Name	タイプ	概要
virtual_function_s_configuration	HostNicVirtualFunctionsConfiguration	物理機能 NIC の仮想機能設定について説明します。
vlan	Vlan	

7.108.1. ad_aggregator_id

モード 4 のボンドの場合、ボンドまたはボンドスレーブの **ad_aggregator_id** プロパティ。ボンディングモード 4 は 802.3ad 標準であり、動的リンクアグリゲーションとも呼ばれます。(詳細は、[Wikipedia](#) および [プレゼンテーション](#) を参照してください。)これは、モード 4 のボンディング、またはボンディングの一部である NIC にも有効です。他のモードのボンディング、またはモード 4 のボンディングの一部ではない NIC には存在しません。**ad_aggregator_id** プロパティは、どのボンディングスレーブがアクティブであるかを示します。アクティブなスレーブの **ad_aggregator_id** の値は、ボンディングの **ad_aggregator_id** プロパティの値と同じです。このパラメータは読み取り専用です。設定してもボンディング/NIC には影響しません。これは、ボンディングの場合は `/sys/class/net/bondX/bonding/ad_aggregator` ファイルから、NIC の場合は `/sys/class/net/ensX/bonding_slave/ad_aggregator_id` ファイルから取得されます。

7.108.2. bridged

ブリッジネットワークのステータスを定義します。ブリッジネットワークの場合は **true** に設定し、ブリッジレスネットワークの場合は **false** に設定します。

7.108.3. statistics

NIC の統計へのリンク。

HostNic 統計値のデータタイプ：

- data.current.rx - 受信したデータの 1 秒あたりのバイト数。
- data.current.tx - 送信されたデータ速度 (バイト毎秒)。
- data.total.rx - 受信したデータの合計。
- data.total.tx - 送信されたデータの合計。
- errors.total.rx - データの受信によるエラーの合計。
- errors.total.tx - データの送信によるエラーの合計。

表7.147 リンクの概要

Name	タイプ	概要
host	ホスト	
network	Network	インターフェイスが接続するネットワークへの参照。

Name	タイプ	概要
physical_function	HostNic	SR-IOV Virtual Function NIC の Physical Function NIC への参照。
qos	Qos	インターフェイスの quality-of-service 設定へのリンク。

7.108.4. network

インターフェイスが接続するネットワークへの参照。空白のネットワーク ID を使用できます。

7.109. HOSTNICVIRTUALFUNCTIONSCONFIGURATION STRUCT

SR-IOV 対応の物理機能 NIC の仮想機能設定について説明します。

表7.148 属性の概要

Name	タイプ	概要
all_networks_authorized	ブール値	関連する仮想関数にすべてのネットワークを定義することを許可するか、指定したネットワークのみを許可するかを定義します。
max_number_of_virtual_functions	Integer	NIC がサポートする仮想機能の最大数。
number_of_virtual_functions	Integer	現在定義されている仮想関数の数。

7.109.1. max_number_of_virtual_functions

NIC がサポートする仮想機能の最大数。このプロパティは読み取り専用です。

7.109.2. number_of_virtual_functions

現在定義されている仮想関数の数。0 から **max_number_of_virtual_functions** までのユーザー定義値。

7.110. HOSTPROTOCOL ENUM

エンジンがホストと通信するために使用するプロトコル。



警告

エンジンのバージョン 4.1 以降、**xml** が削除されたため、プロトコルは常に **stomp** に設定されています。

表7.149 値の概要

Name	概要
stomp	STOMP 上の JSON-RPC プロトコル。
xml	XML-RPC プロトコル。

7.111. HOSTSTATUSENUM

ホストステータスを表すタイプ。

表7.150 値の概要

Name	概要
connecting	エンジンは特定のしきい値の場合はホストと通信できないので、現時点ではフェンシングを通過する前に接続を試みています。
down	ホストがダウンしています。
error	ホストはエラーステータスにあります。
initializing	ホストは初期化中です。
install_failed	ホストのインストールに失敗しました。
installing	ホストがインストールされています。
installing_os	これで、ホストオペレーティングシステムがインストールされます。
kdumping	ホストカーネルがクラッシュし、メモリーダンプが実行されています。
maintenance	ホストのステータスは maintenance です。
non_operational	ホストは動作しません。
non_responsive	ホストは応答しません。

Name	概要
pending_approval	ホストは管理者の承認を待っています。
preparing_for_maintenance	ホストはメンテナンスの準備をしています。
reboot	ホストは再起動されています。
unassigned	ホストはアクティブ化プロセス中です。
up	ホストが稼働しています。

7.111.1. error

ホストはエラーステータスにあります。これは、仮想マシンを何度か実行しようとして失敗した場合に発生します。

7.111.2. initializing

ホストは初期化中です。これは、ホストを 'up' ステータスに移行する前の中間のステップです。

7.111.3. install_failed

ホストのインストールに失敗しました。このような場合は、イベントログを参照して、インストールに失敗した原因を把握し、再インストールを実行してください。

7.111.4. installing_os

これで、ホストオペレーティングシステムがインストールされます。このステータスは、Satellite/Foreman プロバイダーを使用し、ベアメタルプロビジョニング (検出されたホストプロビジョニング) を発行する場合に関連します。

7.111.5. maintenance

ホストのステータスは maintenance です。ホストがメンテナンス中の場合、仮想マシンを実行できません。

7.111.6. non_operational

ホストは動作しません。これは、ストレージとの接続がない、必須ネットワークがサポートされていない、クラスターレベルがサポートされていないなど、さまざまな理由で発生する可能性があります。

7.111.7. non_responsive

ホストは応答しません。これは、エンジンがホストと通信できないことを意味します。

7.111.8. pending_approval

ホストは管理者の承認を待っています。これは、ヴィンテージの ovirt-node / RHV-H にのみ関連します。

7.111.9. preparing_for_maintenance

ホストはメンテナランスの準備をしています。この間、エンジンはすべての仮想マシンをこのホストから他のホストにライブマイグレーションします。すべての移行が完了すると、ホストは 'maintenance' ステータスに移行します。

7.112. HOSTSTORAGE 構造体

表7.151 属性の概要

Name	タイプ	概要
address	String	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
logical_units	LogicalUnit[]	
mount_options	String	
name	String	人間が判読できるプレーンテキストでの名前。
nfs_retrans	Integer	さらなるリカバリーアクションを試みる前に、リクエストを再試行する回数。
nfs_timeo	Integer	NFS 要求を再試行する前に応答を待つ時間 (10 分の 1 秒単位)。
nfs_version	NfsVersion	
override_luns	ブール値	
password	String	
path	String	
port	Integer	
portal	String	
target	String	

Name	タイプ	概要
type	StorageType	
username	String	
vfs_type	String	
volume_group	VolumeGroup	

7.112.1. nfs_retrans

さらなるリカバリーアクションを試みる前に、リクエストを再試行する回数。値は 0 から 65535 の範囲に指定する必要があります。詳細は、**nfs** の man ページの **retrans** マウントオプションの説明を参照してください。

7.112.2. nfs_timeo

NFS 要求を再試行する前に応答を待つ時間 (10 分の 1 秒単位)。値は 0 から 65535 の範囲に指定する必要があります。詳細は、**nfs** の man ページの **timeo** マウントオプションの説明を参照してください。

表7.152 リンクの概要

Name	タイプ	概要
host	ホスト	

7.113. HOSTTYPE ENUM

この列挙型は、ホストが使用するオペレーティングシステムのタイプを判別するために使用されます。

表7.153 値の概要

Name	概要
ovirt_node	ホストには、Red Hat Virtualization Host (RHVH) が含まれています。これは、Red Hat Enterprise Linux、CentOS、または Fedora と同じインストーラーを使用する Red Hat Enterprise Virtualization Hypervisor (RHEV-H) の新しい実装です。
rhel	ホストには、Red Hat Enterprise Linux、CentOS、または Fedora の完全なインストールが含まれています。
rhev_h	ホストには、Red Hat Enterprise Linux、CentOS、または Fedora の小規模バージョンである Red Hat Enterprise Virtualization Hypervisor (RHEV-H) が含まれており、仮想マシンをホストするためだけに使用されます。

7.113.1. ovirt_node

ホストには、Red Hat Virtualization Host (RHVH) が含まれています。これは、Red Hat Enterprise

Linux、CentOS、または Fedora と同じインストーラーを使用する Red Hat Enterprise Virtualization Hypervisor (RHEV-H) の新しい実装です。RHVH とレガシー RHEV-H の主な違いは、RHVH には書き込み可能なファイルシステムがあり、レガシー RHEV-H のように Manager が RPM をプッシュするのではなく、独自のインストールを処理することです。

7.114. HOSTEDENGINE 構造体

表7.154 属性の概要

Name	タイプ	概要
active	ブール値	
configured	ブール値	
global_maintenance	ブール値	
local_maintenance	ブール値	
score	Integer	

7.115. ICON 構造体

仮想マシンまたはテンプレートのアイコン。

表7.155 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
data	String	アイコンファイルの Base64 エンコードコンテンツ。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
media_type	String	アイコンファイルのフォーマット。
name	String	人間が判読できるプレーンテキストでの名前。

7.115.1. media_type

アイコンファイルのフォーマット。

以下のいずれかになります。

- `image/jpeg`
- `image/png`
- `image/gif`

7.116. IDENTIFIED 構造体

このインターフェイスは、識別子を持つオブジェクトを表すすべてのタイプの基本モデルです。

表7.156 属性の概要

Name	タイプ	概要
<code>comment</code>	<code>String</code>	このオブジェクトに関するコメントを含むフリーテキスト。
<code>description</code>	<code>String</code>	プレーンテキストでの人間が判読できる説明。
<code>id</code>	<code>String</code>	一意の ID
<code>name</code>	<code>String</code>	人間が判読できるプレーンテキストでの名前。

7.117. IMAGE 構造体

表7.157 属性の概要

Name	タイプ	概要
<code>comment</code>	<code>String</code>	このオブジェクトに関するコメントを含むフリーテキスト。
<code>description</code>	<code>String</code>	プレーンテキストでの人間が判読できる説明。
<code>id</code>	<code>String</code>	一意の ID
<code>name</code>	<code>String</code>	人間が判読できるプレーンテキストでの名前。

表7.158 リンクの概要

Name	タイプ	概要
<code>storage_domain</code>	<code>StorageDomain</code>	

7.118. IMAGETRANSFER 構造体

このタイプには、実行中のイメージ転送に関する情報が含まれます。

表7.159 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
direction	ImageTransferDirection	方向は、転送がイメージデータの送信 (upload) であるか、イメージデータの受信 (download) であるかを示します。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
phase	ImageTransferPhase	進行中のイメージ転送の現在のフェーズ。
proxy_url	String	ユーザーが入力または出力するプロキシサーバーの URL。
signed_ticket	String	入力または出力するプロキシサーバーの HTTPS リクエストの 認証 ヘッダーとしてアタッチする必要がある署名付きチケット (proxy_url 属性を参照)。

7.118.1. direction

方向は、転送がイメージデータの送信 (**upload**) であるか、イメージデータの受信 (**download**) であるかを示します。

新しい転送の追加時に方向が設定されていない場合、デフォルトの転送方向は **upload** になります。

7.118.2. phase

進行中のイメージ転送の現在のフェーズ。各転送には、管理されたセッションが必要です。このセッションは、ユーザーがイメージを入力または出力するために開かれている必要があります。詳細は、[イメージ転送](#) を参照してください。

7.118.3. proxy_url

ユーザーが入力または出力するプロキシサーバーの URL。この属性は、イメージ転送エンティティが **転送** フェーズにある場合にのみ利用できます。詳細は、**phase** を参照してください。

表7.160 リンクの概要

Name	タイプ	概要
host	ホスト	入力または出力の対象となるイメージへの書き込みに使用されるホスト。
image	Image	入力または出力の対象となるイメージ。

7.119. IMAGETRANSFERDIRECTION ENUM

転送時の [イメージ転送](#) 方向。

新しい転送を追加する場合、ユーザーは ImageTransferDirection として **upload** を選択してイメージに転送するか、**download** を選択してイメージから転送するか選択できます。

詳細は、[イメージ転送](#) を参照してください。

表7.161 値の概要

Name	概要
ダウンロード	イメージからデータをストリーミングする場合、ユーザーは download を択する必要があります。
upload	ユーザーは、イメージにデータをストリームしたいときに、 upload を選択することができます。

7.120. IMAGETRANSFERPHASE ENUM

[イメージ転送](#) エンティティのフェーズのリスト。これらの値はそれぞれ、転送フローの特定のポイントを定義します。

詳細は、[イメージ転送](#) を参照してください。

表7.162 値の概要

Name	概要
cancelled	このフェーズは、ユーザーが転送をキャンセルした結果として設定されます。
finalizing_failure	このフェーズは、管理ポータルでのみ設定でき、転送中にエラーが発生し、失敗してファイナライズされていることを示します。
finalizing_success	このフェーズは、ユーザーが finalize を呼び出したときに設定されます。
finished_failure	対象のイメージが検証に失敗し、使用できないことを示します。
finished_success	転送セッションが正常に終了し、対象のイメージが検証されて使用できる状態になったことを示します。
initializing	イメージ転送の初期段階。
paused_system	このフェーズは、セッションがタイムアウトしたか、この転送で他のエラーが発生した場合（例：選択したホストで <code>ovirt-imageio-daemon</code> が実行されていない）を意味します。
paused_user	このフェーズは、ユーザーが pause を使用して一時停止を呼び出した結果です。

Name	概要
resuming	クライアントが resume を呼び出して転送が再開されたフェーズ。
transferring	転送セッションが開始され、クライアントが任意のツールを使用して目的のイメージを入力または出力できるフェーズ。
unknown	未知のフェーズ。

7.120.1. cancelled

このフェーズは、ユーザーが転送をキャンセルした結果として設定されます。キャンセルは、管理ポータルでのみ実行できます。

7.120.2. finalizing_success

このフェーズは、ユーザーが [finalize](#) を呼び出したときに設定されます。転送セッションを終了し、対象のイメージの使用を終了するには、[finalize](#) の呼び出しが不可欠です。ファイナライズ後、フェーズは **finished_success** または **finished_failure** に変更されます。

詳細は、[イメージ転送](#) を参照してください。

7.120.3. finished_failure

対象のイメージが検証に失敗し、使用できないことを示します。このフェーズに達すると、イメージ転送エンティティが削除され、対象のイメージが **illegal** に設定されます。

7.120.4. finished_success

転送セッションが正常に終了し、対象のイメージが検証されて使用できる状態になったことを示します。このフェーズに到達すると、イメージ転送エンティティは削除されます。

7.120.5. initializing

イメージ転送の初期段階。転送セッションの確立中に設定されます。セッションが確立されると、フェーズが **transferring** に変更されます。

7.120.6. paused_system

このフェーズは、セッションがタイムアウトしたか、この転送で他のエラーが発生した場合（例：選択したホストで `ovirt-imageio-daemon` が実行されていない）を意味します。セッションを再開するには、クライアントは [resume](#) を呼び出す必要があります。再開後、フェーズは **resuming** に切り替わります。

7.120.7. resuming

クライアントが [resume](#) を呼び出して転送が再開されたフェーズ。再開すると新しいセッションが開始され、呼び出した後、フェーズは **transferring**、失敗した場合は **paused_system** に変更されます。

7.120.8. unknown

未知のフェーズ。これは、予期しないエラーが発生した場合にのみ設定されます。

7.121. INHERITABLEBOOLEAN ENUM

設定するか、上位レベルから継承できるブール値を表す enum。継承順は仮想マシン → クラスタ → engine-config です。

表7.163 値の概要

Name	概要
false	このレベルで値を false に設定します。
inherit	上位レベルから値を継承します。
true	このレベルで値を true に設定します。

7.122. INITIALIZATION 構造体

表7.164 属性の概要

Name	タイプ	概要
active_directory_ou	String	
authorized_ssh_keys	String	
cloud_init	CloudInit	
configuration	設定	
custom_script	String	
dns_search	String	
dns_servers	String	
domain	String	
host_name	String	
input_locale	String	
nic_configurations	NicConfiguration[]	

Name	タイプ	概要
org_name	String	
regenerate_ids	ブール値	
regenerate_ssh_keys	ブール値	
root_password	String	
system_locale	String	
timezone	String	
ui_language	String	
user_locale	String	
user_name	String	
windows_license_key	String	

7.123. INSTANCETYPE 構造体

仮想マシンのハードウェア設定について説明します。

たとえば、**medium** インスタンスタイプには、1つの仮想 CPU と 4GiB のメモリーが含まれています。これは最上位のエンティティです (たとえば、データセンターやクラスターにバインドされていません)。インスタンスタイプに使用され、仮想マシンおよびテンプレートタイプに共通する属性は次のとおりです。

- **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

インスタンスタイプとテンプレートの両方から仮想マシンを作成する場合、仮想マシンはインスタンスタイプからハードウェア設定を継承します。



注記

ほとんどのテンプレート属性はインスタンスタイプでは使用されませんが、インスタンスタイプはその属性をテンプレートエンティティから継承します。

表7.165 属性の概要

Name	タイプ	概要
bios	Bios	仮想マシンの BIOS 設定への参照。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
console	コンソール	この仮想マシン用に設定されたコンソール。
cpu	Cpu	仮想マシン CPU の設定。
cpu_shares	Integer	
creation_time	日付	仮想マシンの作成日。
custom_compatibility_version	バージョン	仮想マシンのカスタム互換性バージョン。
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	さまざまなフックを設定するために VDSM に送信されるプロパティ。

Name	タイプ	概要
delete_protected	ブール値	true の場合、仮想マシンは削除できません。
description	String	プレーンテキストでの人間が判読できる説明。
display	表示	仮想マシンの表示設定。
domain	Domain	この仮想マシン用に設定されたドメイン。
high_availability	高可用性	仮想マシンの高可用性設定。
id	String	一意の ID
initialization	初期化	仮想マシンの初期化設定への参照。
io	Io	IO スレッドのパフォーマンスチューニング用。
large_icon	アイコン	仮想マシンの大きなアイコン。
lease	StorageDomainLease	この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。
memory	Integer	仮想マシンのメモリー (バイト単位)。
memory_policy	MemoryPolicy	仮想マシンのメモリー管理設定への参照。
migration	MigrationOptions	実行中の仮想マシンの別のホストへの移行設定への参照。
migration_down_time	Integer	別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。
name	String	人間が判読できるプレーンテキストでの名前。
origin	String	この仮想マシンのオリジン。
os	OperatingSystem	仮想マシンにインストールされているオペレーティングシステムのタイプ。
rng_device	RngDevice	この仮想マシンの乱数ジェネレーターデバイスの設定。
serial_number	SerialNumber	クラスター内の仮想マシンのシリアル番号。
small_icon	アイコン	仮想マシンの小さなアイコン。

Name	タイプ	概要
soundcard_enabled	ブール値	true の場合、サウンドカードが仮想マシンに追加されます。
sso	Sso	この仮想マシンが設定されているシングルサインオン設定への参照。
start_paused	ブール値	true の場合、仮想マシンは起動後、最初は 'paused' 状態になります。
stateless	ブール値	true の場合、仮想マシンはステートレスで、シャットダウン後にその状態 (ディスク) がロールバックされます。
status	TemplateStatus	テンプレートのステータス
time_zone	TimeZone	oVirt によって設定された仮想マシンのタイムゾーン。
tunnel_migration	ブール値	true の場合、ネットワークデータ転送は仮想マシンのライブマイグレーション中に暗号化されます。
type	VmType	仮想マシンがデスクトップとサーバーのどちらに最適化されているかを決定します。
usb	Usb	この仮想マシンの USB デバイスの設定 (カウント、タイプ)。
version	TemplateVersion	これが別のテンプレートのベースバージョンかサブバージョンであることを示します。
virtio_scsi	VirtioScsi	VirtIO SCSI 設定への参照。
vm	Vm	このテンプレートに関連付けられている仮想マシンの設定。

7.123.1. cpu

仮想マシン CPU の設定。

ソケット設定は、仮想マシンを再起動せずに更新できます。コアとスレッドは再起動する必要があります。

たとえば、ソケットの数をすぐに 4 に変更し、再起動後にコアとスレッドの数を 2 に変更するには、以下のリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<vm>
  <cpu>
    <topology>
```

```
<sockets>4</sockets>
<cores>2</cores>
<threads>2</threads>
</topology>
</cpu>
</vm>
```

7.123.2. custom_compatibility_version

仮想マシンのカスタム互換性バージョン。

仮想マシンを独自の互換性バージョンにカスタマイズできるようにします。**custom_compatibility_version** が設定されている場合、この特定の仮想マシンのクラスター互換性バージョンをオーバーライドします。

仮想マシンの互換バージョンは、仮想マシンが格納されているデータセンターによって制限され、仮想マシンが実行される予定のホストの機能に対してチェックされます。

7.123.3. high_availability

仮想マシンの高可用性設定。設定されている場合、仮想マシンが予期せずダウンしたときに自動的に再起動されます。

7.123.4. large_icon

仮想マシンの大きなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.123.5. lease

この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。

リースを使用して実行している仮想マシンは、この仮想マシンの別のインスタンスが別のホストで実行されるのを防ぐために、実行中にリースが別のホストによって取得されていないことを確認する必要があります。これにより、高可用性の仮想マシンでスプリットブレインが保護されます。このテンプレートから作成された仮想マシンを、このストレージドメインをリースの場所として事前設定するために、テンプレートにリース用に定義されたストレージドメインを含めることもできます。

7.123.6. memory

仮想マシンのメモリー (バイト単位)。

たとえば、1ギビバイト (GiB) のメモリーを含むように仮想マシンを更新するには、次の要求を送信します。

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

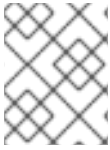
リクエスト本文は、以下のようになります。

```
<vm>
  <memory>1073741824</memory>
</vm>
```



注記

この例のメモリーは、次の式を使用してバイトに変換されます:
 $1 \text{ GiB} = 2^{30} \text{ バイト} = 1073741824 \text{ バイト}$ 。



注記

メモリーホットプラグは、Red Hat Virtualization 3.6 以降でサポートされています。上記の例では、仮想マシンの実行中にメモリーを増やすことができます。

7.123.7. migration_downtime

別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。

仮想マシンに対して明示的に設定するか、**engine-config -s DefaultMaximumMigrationDowntime=[value]** で設定します。

7.123.8. origin

この仮想マシンのオリジン。

値:

- **ovirt**
- **rhev**
- **vmware**
- **xen**
- **external**
- **hosted_engine**
- **managed_hosted_engine**
- **kvm**
- **physical_machine**
- **hyperv**

7.123.9. small_icon

仮想マシンの小さなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.123.10. sso

この仮想マシンが設定されているシングルサインオン設定への参照。コンソールを開くと、ユーザーは仮想マシンのオペレーティングシステムに自動的にサインインできます。

表7.166 リンクの概要

Name	タイプ	概要
cdroms	Cdrom[]	テンプレートにアタッチされた CD-ROM デバイスへの参照。
cluster	Cluster	仮想マシンが属するクラスターへの参照。
cpu_profile	CpuProfile	この仮想マシンによって使用される CPU プロファイルへの参照。
disk_attachments	DiskAttachment[]	テンプレートに割り当てられたディスクへの参照。
graphics_consoles	GraphicsConsole[]	テンプレートに接続されたグラフィックコンソールへの参照。
nics	Nic[]	テンプレートにアタッチされたネットワークインターフェイスへの参照。
permissions	Permission[]	テンプレートに割り当てられたユーザーパーミッションへの参照。
quota	クォータ	この仮想マシンに設定されたクォータ設定への参照。
storage_domain	StorageDomain	仮想マシンが属するストレージドメインへの参照。
tags	Tag[]	テンプレートにアタッチされたタグへの参照。
watchdogs	Watchdog[]	テンプレートにアタッチされたウォッチドッグデバイスへの参照。

7.124. IO 構造体

表7.167 属性の概要

Name	タイプ	概要
threads	Integer	

7.125. IP 構造体

ネットワークインターフェイスの IP 設定を表します。

表7.168 属性の概要

Name	タイプ	概要
address	String	IP アドレスのテキスト表現。

Name	タイプ	概要
gateway	String	デフォルトゲートウェイのアドレス。
netmask	String	ネットワークマスク。
version	IpVersion	IP プロトコルのバージョン。

7.125.1. address

IP アドレスのテキスト表現。

たとえば、IPv4 アドレスは以下のように表現されます。

```
<ip>
  <address>192.168.0.1</address>
  ...
</ip>
```

IPv6 アドレスは次のように表されます。

```
<ip>
  <address>2620:52:0:20f0:4216:7eff:feaa:1b50</address>
  ...
</ip>
```

7.125.2. netmask

ネットワークマスク。

IPv6 アドレスの場合、値は 0-128 の範囲の整数で、サブネット接頭辞を表します。

7.125.3. version

IP プロトコルのバージョン。



注記

Manager のバージョン 4.1 から、この属性はオプションになり、値が指定されていない場合は、**address** 属性の値から推測されます。

7.126. IPADDRESSASSIGNMENT STRUCT

ネットワークデバイスの IP アドレスの割り当てを表します。

静的ブートプロトコル割り当てを行うには、IP 設定でサブネットマスクと IP アドレス (およびオプションでデフォルトゲートウェイ) を指定する必要があります。

表7.169 属性の概要

Name	タイプ	概要
assignment_method	BootProtocol	ネットワークデバイスの IP 設定の割り当てに使用されるブートプロトコルを設定します。
ip	Ip	ネットワークデバイスの IP 設定を設定します。

7.127. IPVERSION ENUM

IP プロトコルバージョンの値を定義します。

表7.170 値の概要

Name	概要
v4	IPv4。
v6	IPv6。

7.128. ISCSIBOND STRUCT

表7.171 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.172 リンクの概要

Name	タイプ	概要
data_center	DataCenter	
networks	Network[]	
storage_connections	StorageConnections[]	

7.129. ISCSIDetails STRUCT

表7.173 属性の概要

Name	タイプ	概要
address	String	
disk_id	String	
initiator	String	
lun_mapping	Integer	
password	String	
paths	Integer	
port	Integer	
portal	String	
product_id	String	
serial	String	
size	Integer	
status	String	
storage_domain_id	String	
target	String	
username	String	
vendor_id	String	
volume_group_id	String	

7.130. JOB 構造体

システム内のフローの実行を監視するジョブを表します。ジョブには、階層構造の複数のステップを含めることができます。ステップは並行して処理できますが、フローの実装によって異なります。

表7.174 属性の概要

Name	タイプ	概要
auto_cleared	ブール値	システムによってジョブが完了した後、ジョブが自動的にクリアされるかどうかを示します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
end_time	日付	ジョブの終了時刻。
external	ブール値	ジョブが外部システムによって開始されたかどうかを示します。
id	String	一意の ID
last_updated	日付	ジョブの最終更新日。
name	String	人間が判読できるプレーンテキストでの名前。
start_time	日付	ジョブの開始時刻。
status	JobStatus	ジョブのステータス。

7.130.1. external

ジョブが外部システムによって開始されたかどうかを示します。外部ジョブは、ジョブの作成者によって外部で管理されます。

表7.175 リンクの概要

Name	タイプ	概要
owner	User	ジョブの所有者であるユーザー。
steps	step[]	ジョブのステップ。

7.131. JOBSTATUS ENUM

ステップのステータスを表します。

表7.176 値の概要

Name	概要
aborted	中止されたジョブのステータス。

Name	概要
failed	失敗したジョブのステータス。
finished	終了したジョブのステータス。
started	開始されたジョブのステータス。
unknown	不明なジョブのステータス。

7.131.1. aborted

中止されたジョブのステータス。このステータスは、強制的に中止された外部ジョブに適用されます。

7.131.2. finished

終了したジョブのステータス。このステータスは、ジョブの実行が完了したことを示します。

7.131.3. started

開始されたジョブのステータス。このステータスは、現在実行中のジョブを表します。

7.131.4. unknown

不明なジョブのステータス。このステータスは、解決方法が不明なジョブ、つまり、システムが予期せず再起動される前に実行されたジョブを表します。

7.132. KATELLOERRATUM STRUCT

Katello erratum を表すタイプ。

表7.177 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
issued	日付	Katello エラータが発行された日付。
name	String	人間が判読できるプレーンテキストでの名前。
packages	Package[]	Katello エラータによって報告された問題を解決するパッケージのリスト。

Name	タイプ	概要
severity	String	Katello エラータの重大度。
solution	String	Katello エラータで記述されている問題のソリューション。
summary	String	Katello エラータの要約。
title	String	Katello エラータのタイトル。
type	String	Katello エラータのタイプ。

7.132.1. severity

Katello エラータの重大度。

サポートされる重要度は、**moderate**、**important**、**critical** です。

7.132.2. type

Katello エラータのタイプ。

サポートされるタイプは、**bugfix**、**enhancement**、**security** です。

表7.178 リンクの概要

Name	タイプ	概要
host	ホスト	Katello エラータが割り当てられているホストへの参照。
vm	Vm	Katello エラータが割り当てられている仮想マシンへの参照。

7.133. KDUMPSTATUSENUM

表7.179 値の概要

Name	概要
disabled	
enabled	
unknown	

7.134. KERNEL 構造体

表7.180 属性の概要

Name	タイプ	概要
version	バージョン	

7.135. KSM 構造体

表7.181 属性の概要

Name	タイプ	概要
enabled	ブール値	
merge_across_nodes	ブール値	

7.136. LOGSEVERITY ENUM

イベントの重大度を表す列挙型。

表7.182 値の概要

Name	概要
alert	アラートの重大度。
error	エラーの重大度。
normal	通常の重大度。
warning	警告の重大度。

7.136.1. alert

アラートの重大度。すぐに対処する必要がある状態を指定するために使用されます。

7.136.2. error

エラーの重大度。調査が必要なエラーがあることを指定するために使用されます。

7.136.3. normal

通常の重大度。情報イベントに使用されます。

7.136.4. warning

警告の重大度。何かが間違っている可能性があることを警告するために使用されます。

7.137. LOGICALUNIT 構造体

表7.183 属性の概要

Name	タイプ	概要
address	String	
discard_max_size	Integer	1回の操作で論理ユニットの基礎のストレージによって破棄できる最大バイト数。
discard_zeroes_data	ブール値	論理ユニットの基礎となるストレージ内の以前に破棄されたブロックがゼロとして読み戻される場合は True。
disk_id	String	
id	String	
lun_mapping	Integer	
password	String	
paths	Integer	
port	Integer	
portal	String	
product_id	String	
serial	String	
size	Integer	
status	LunStatus	
storage_domain_id	String	
target	String	
username	String	
vendor_id	String	
volume_group_id	String	

7.137.1. discard_max_size

1回の操作で論理ユニットの基礎のストレージによって破棄できる最大バイト数。値 0 は、デバイスが破棄機能をサポートしていないことを意味します。



注記

これは、**discard_max_bytes** の **queue-sysfs** の [ドキュメント](#) に記載されているように、ハードウェアの制限ではなく、ソフトウェアの制限です。

7.137.2. discard_zeroes_data

論理ユニットの基礎となるストレージ内の以前に破棄されたブロックがゼロとして読み戻される場合は True。詳細は、**discard_zeroes_data** の **queue-sysfs** の [ドキュメント](#) を参照してください。

7.138. LUNSTATUS ENUM

表7.184 値の概要

Name	概要
free	
unusable	
used	

7.139. MAC 構造体

表7.185 属性の概要

Name	タイプ	概要
address	String	

7.140. MACPOOL 構造体

MAC アドレスプールを表します。

MAC アドレスプールの XML 表現の例:

```
<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>
```



```

</range>
</ranges>
</mac_pool>

```

表7.186 属性の概要

Name	タイプ	概要
allow_duplicates	ブール値	プールで重複する MAC アドレスを許可するかどうかを定義します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
default_pool	ブール値	これがデフォルトプールかどうかを定義します。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
ranges	Range[]	プールの MAC アドレスの範囲を定義します。

7.140.1. allow_duplicates

プールで重複する MAC アドレスを許可するかどうかを定義します。指定されていない場合、**false** にデフォルト設定されます。

7.140.2. default_pool

これがデフォルトプールかどうかを定義します。指定されていない場合、**false** にデフォルト設定されます。

7.140.3. ranges

プールの MAC アドレスの範囲を定義します。複数の範囲を定義できます。

7.141. MEMORYOVERCOMMIT STRUCT

表7.187 属性の概要

Name	タイプ	概要
percent	Integer	

7.142. MEMORYPOLICY 構造体

仮想マシンのようなエンティティのメモリー関連のプロパティの論理グループ。

表7.188 属性の概要

Name	タイプ	概要
ballooning	ブール値	
guaranteed	Integer	
max	Integer	仮想マシンの最大メモリー（バイト単位）。
over_commit	MemoryOverCommit	
transparent_huge_pages	TransparentHugePages	

7.142.1. max

仮想マシンの最大メモリー（バイト単位）。

ユーザーは値をバイト単位で指定し、エンジンは値を最も低い MiB 値に丸めます。

たとえば、ユーザーが 1073741825 (1 GiB + 1 byte) という値を入力すると、Red Hat Virtualization Manager はその値を最も近い下位 MiB 境界に切り詰めます。この場合は、1073741824 (1 GiB) になります。

7.143. MESSAGEBROKERTYPE ENUM

表7.189 値の概要

Name	概要
qpidd	
rabbitmq	

7.144. METHOD 構造体

表7.190 属性の概要

Name	タイプ	概要
id	SsoMethod	

7.145. MIGRATEONERROR ENUM

表7.191 値の概要

Name	概要
<code>do_not_migrate</code>	
<code>migrate</code>	
<code>migrate_highly_available</code>	

7.146. MIGRATIONBANDWIDTH 構造体

移行で使用される帯域幅を定義します。

表7.192 属性の概要

Name	タイプ	概要
<code>assignment_method</code>	MigrationBandwidthAssignmentMethod	帯域幅を割り当てるのに使用するメソッド。
<code>custom_value</code>	Integer	Mbps 単位のカスタム帯域幅。

7.146.1. custom_value

Mbps 単位のカスタム帯域幅。`assignmentMethod` 属性が `custom` の場合にのみ適用されます。

7.147. MIGRATIONBANDWIDTHASSIGNMENTMETHOD ENUM

移行帯域幅の割り当て方法を定義します。

表7.193 値の概要

Name	概要
<code>auto</code>	Quality of Service が定義されている場合、Quality of Service から帯域幅を取得します。
<code>custom</code>	カスタム定義の帯域幅 (Mbit/s)。
<code>hypervisor_default</code>	ハイパーバイザーで設定された値を取得します。

7.147.1. auto

Quality of Service が定義されている場合、Quality of Service から帯域幅を取得します。Quality of Service が定義されていない場合、帯域幅は使用中の検出されたリンク速度から取得されます。何も検出されない場合、帯域幅は `hypervisor_default` 値に戻ります。

7.148. MIGRATIONOPTIONS 構造体

移行オプションのタイプ。

表7.194 属性の概要

Name	タイプ	概要
auto_converge	InheritableBoolean	
bandwidth	MigrationBandwidth	移行で使用できる帯域幅。
compressed	InheritableBoolean	

表7.195 リンクの概要

Name	タイプ	概要
policy	MigrationPolicy	engine-config を使用して定義された、移行ポリシーへの参照。

7.149. MIGRATIONPOLICY 構造体

収束や許可される並列移行の数など、移行の処理方法を説明するポリシー。

表7.196 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

7.150. NETWORK 構造体

論理ネットワークのタイプ。

論理ネットワークの JSON 表現の例:

```
{
  "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"
    } ]
  } ]
} ]
}

```

同じ論理ネットワークの XML 表現の例:

```

<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>

```

表7.197 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
display	ブール値	
dns_resolver_configuration	DnsResolverConfiguration	
id	String	一意の ID

Name	タイプ	概要
ip	Ip	
mtu	Integer	ネットワークの最大伝送単位を指定します。
name	String	人間が判読できるプレーンテキストでの名前。
profile_required	ブール値	
必須	ブール値	
status	NetworkStatus	
stp	ブール値	ネットワークでスパニングツリープロトコルが有効かどうかを指定します。
usages	NetworkUsage[]	ネットワークの使用要素のセットを定義します。
vlan	Vlan	

7.150.1. usages

ネットワークの使用要素のセットを定義します。

たとえば、ユーザーは、**vm** および **display** の値を使用して、ネットワークが仮想マシントラフィックに使用され、表示トラフィックにも使用されるように指定できます。

表7.198 リンクの概要

Name	タイプ	概要
cluster	Cluster	
data_center	DataCenter	ネットワークがメンバーになっているデータセンターへの参照。
network_labels	NetworkLabel[]	ネットワークに割り当てられたラベルへの参照。
permissions	Permission[]	ネットワークのパーミッションへの参照。
qos	Qos	
vnic_profiles	VnicProfile[]	ネットワークのプロファイルへの参照。

7.151. NETWORKATTACHMENT 構造体

ホストがネットワークに接続する方法について説明します。

ホスト上のネットワークアタッチメントのXML表現:

```
<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>
```

ネットワークをネットワークインターフェイスカード (NIC) にアタッチするには、**name** または **id** を持つネットワーク要素が必要です。

たとえば、ネットワークをホストネットワークインターフェイスカードにアタッチするには、以下のようなリクエストを送信します。

```
POST /ovirt-engine/api/hosts/123/nics/456/networkattachments
```

リクエスト本文は以下のようになります。

```
<networkattachment>
  <network id="234"/>
</networkattachment>
```

ネットワークをホストにアタッチするには、次のようなリクエストを送信します。

```
POST /ovirt-engine/api/hosts/123/networkattachments
```

リクエスト本文は以下のようになります。

```
<network_attachment>
  <network id="234"/>
  <host_nic id="456"/>
</network_attachment>
```

ip_address_assignments および **properties** 要素は、作成後に更新可能です。

たとえば、ネットワークアタッチメントを更新するには、以下のようなリクエストを送信します。

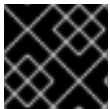
```
PUT /ovirt-engine/api/hosts/123/nics/456/networkattachments/789
```

リクエスト本文は以下ようになります。

```
<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>
```

ネットワークインターフェイスカードからネットワークをデタッチするには、以下のようなリクエストを送信します。

```
DELETE /ovirt-engine/api/hosts/123/nics/456/networkattachments/789
```



重要

ネットワークアタッチメント設定への変更は、明示的にコミットする必要があります。

ネットワークアタッチメントの **properties** サブコレクションの XML 表現。

```
<network_attachment>
  <properties>
    <property>
      <name>bridge_opts</name>
      <value>
        forward_delay=1500 group_fwd_mask=0x0 multicast_snooping=1
      </value>
    </property>
  </properties>
  ...
</network_attachment>
```

表7.199 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
dns_resolver_configuration	DnsResolverConfiguration	GET を使用してネットワークアタッチメントを取得すると、DNS リゾルバーの設定が報告されます。
id	String	一意の ID
in_sync	ブール値	
ip_address_assignments	IpAddressAssignment[]	ネットワークの IP 設定。
name	String	人間が判読できるプレーンテキストでの名前。
properties	Property[]	ネットワーク設定のカスタムプロパティを定義します。
reported_configurations	ReportedConfiguration[]	設定プロパティの読み取り専用リスト。

7.151.1. dns_resolver_configuration

GET を使用してネットワークアタッチメントを取得すると、DNS リゾルバーの設定が報告されます。新しいネットワークアタッチメントを作成する場合、または既存のネットワークアタッチメントを更新する場合はオプションです。

7.151.2. properties

ネットワーク設定のカスタムプロパティを定義します。

ブリッジオプションのセット名は `bridge_opts` です。複数の項目を空白文字で区切ります。**bridge_opts** には、以下のキーが有効です。

Name	デフォルト値
forward_delay	1500
gc_timer	3765
group_addr	1:80:c2:0:0:0
group_fwd_mask	0x0
hash_elasticity	4

Name	デフォルト値
hash_max	512
hello_time	200
hello_timer	70
max_age	2000
multicast_last_member_count	2
multicast_last_member_interval	100
multicast_membership_interval	26000
multicast_querier	0
multicast_querier_interval	25500
multicast_query_interval	13000
multicast_query_response_interval	1000
multicast_query_use_ifaddr	0
multicast_router	1
multicast_snooping	1
multicast_startup_query_count	2
multicast_startup_query_interval	3125

表7.200 リンクの概要

Name	タイプ	概要
host	ホスト	
host_nic	HostNic	ホストネットワークインターフェイスへの参照。
network	Network	インターフェイスがアタッチされているネットワークへの参照。
qos	Qos	

7.152. NETWORKCONFIGURATION 構造体

表7.201 属性の概要

Name	タイプ	概要
dns	Dns	
nics	Nic[]	

7.153. NETWORKFILTER 構造体

ネットワークフィルターは、定義されたルールに従って、仮想マシンの NIC との間で送受信されるパケットをフィルタリングします。

libvirt に基づいてサポートされるネットワークフィルターには、いくつかのタイプがあります。さまざまなネットワークフィルターの詳細については、[こちら](#) を参照してください。

libvirt のネットワークフィルターに加えて、2つの追加のネットワークフィルターがあります。1つは **vdsml-no-mac-spoofing** と呼ばれ、**no-mac-spoofing** および **no-arp-mac-spoofing** で構成されています。もう1つは **ovirt-no-filter** と呼ばれ、仮想マシンの NIC にネットワークフィルターを定義しない場合に使用されます。**ovirt-no-filter** ネットワークフィルターは内部実装にのみ使用され、NIC には存在しません。

これは XML 表現の例です。

```
<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>
```

バージョンの一部が存在しない場合は、-1 で表されます。

表7.202 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
version	バージョン	特定の NetworkFilter の最小サポート対象バージョン。

7.153.1. version

特定の NetworkFilter の最小サポート対象バージョン。これは、NetworkFilter が最初に導入されたバージョンです。

7.154. NETWORKLABEL 構造体

ホストネットワークインターフェイスに追加できるラベルを表します。

表7.203 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.204 リンクの概要

Name	タイプ	概要
host_nic	HostNic	
network	ネットワーク	

7.155. NETWORKPLUGINTYPE ENUM

表7.205 値の概要

Name	概要
open_vswitch	

7.156. NETWORKSTATUS ENUM

表7.206 値の概要

Name	概要
non_operational	
operational	

7.157. NETWORKUSAGE ENUM

このタイプは、クラスター内でネットワークが使用される目的を示します。

表7.207 値の概要

Name	概要
display	ネットワークは、SPICE および VNC トラフィックに使用されます。
gluster	ネットワークは、Gluster (ブリック) データトラフィックに使用されます。
management	ネットワークは、Red Hat Virtualization Manager とノード間の通信に使用されます。
migration	このネットワークは、仮想マシンの移行に使用されます。
vm	

7.157.1. management

ネットワークは、Red Hat Virtualization Manager とノード間の通信に使用されます。これは、ovirtmgmt ブリッジが作成されるネットワークです。

7.158. NFSPROFILEDETAIL STRUCT

表7.208 属性の概要

Name	タイプ	概要
nfs_server_ip	String	
profile_details	ProfileDetail[]	

7.159. NFSVERSION ENUM

表7.209 値の概要

Name	概要
auto	
v3	
v4	
v4_1	

Name	概要
v4_2	NFS 4.

7.159.1. v4_2

NFS 4.2.

7.160. NIC 構造体

仮想マシン NIC を表します。

たとえば、NIC の XML 表現は以下のようになります。

```
<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>
```

表7.210 属性の概要

Name	タイプ	概要
boot_protocol	BootProtocol	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
interface	NicInterface	NIC に使用されるドライバーのタイプ。
linked	ブール値	NIC が仮想マシンにリンクされているかどうかを定義します。
mac	Mac	インターフェイスの MAC アドレス。
name	String	人間が判読できるプレーンテキストでの名前。
on_boot	ブール値	
plugged	ブール値	NIC が仮想マシンに接続されているかどうかを定義します。

表7.211 リンクの概要

Name	タイプ	概要
instance_type	InstanceType	オプションで、デバイスが使用するインスタンスタイプへの参照。
network	Network	インターフェイスが接続されるべきネットワークへの参照。
network_attachments	NetworkAttachment[]	
network_labels	NetworkLabel[]	
reported_devices	ReportedDevice[]	
statistics	Statistic[]	NIC の統計へのリンク。
template	Template	オプションで、デバイスが使用するテンプレートへの参照。
virtual_function_allowed_labels	NetworkLabel[]	
virtual_function_allowed_networks	Network[]	
vm	Vm	この要素は使用せず、代わりに vms を使用してください。
vms	Vm[]	このデバイスを使用する仮想マシンへの参照。
vnic_profile	VnicProfile	

7.160.1. network

インターフェイスが接続されるべきネットワークへの参照。空白のネットワーク ID を使用できます。

NIC の作成または更新におけるこの要素の使用は非推奨になりました。代わりに **vnic_profile** を使用してください。これは、IP アドレスやその他のネットワーク詳細のホルダーとして、**initialization** 要素によってまだ使用されているため保存されます。

7.160.2. vms

このデバイスを使用する仮想マシンへの参照。デバイスは複数の仮想マシンで使用できます。たとえば、共有ディスクは2つ以上の仮想マシンで同時に使用することができます。

7.161. NICCONFIGURATION 構造体

表7.212 属性の概要

Name	タイプ	概要
boot_protocol	BootProtocol	
ip	Ip	
name	String	
on_boot	ブール値	

7.162. NICINTERFACE ENUM

表7.213 値の概要

Name	概要
e1000	
pci_passthrough	
rtl8139	
rtl8139_virtio	
spapr_vlan	
virtio	

7.163. NICSTATUS ENUM

表7.214 値の概要

Name	概要
down	
up	

7.164. NUMANODE STRUCT

物理 NUMA ノードを表します。

XML 表現の例:

```
<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>

```

表7.215 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
cpu	Cpu	
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
index	Integer	
memory	Integer	NUMA ノードのメモリー (MB)。
name	String	人間が判読できるプレーンテキストでの名前。
node_distance	String	

表7.216 リンクの概要

Name	タイプ	概要
host	ホスト	
statistics	Statistic[]	各ホスト NUMA ノードリソースは、ホスト NUMA ノード固有の統計の統計サブコレクションを公開します。

7.164.1. statistics

各ホスト NUMA ノードリソースは、ホスト NUMA ノード固有の統計の統計サブコレクションを公開します。

XML 表現の例:

```

<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>

```



注記

この統計サブコレクションは読み取り専用です。

次のリストは、ホスト NUMA ノードの統計タイプを示しています。

Name	説明
memory.total	NUMA ノードの合計メモリー (バイト単位)。
memory.used	NUMA ノードで使用されるメモリー (バイト単位)。
memory.free	NUMA ノードの空きメモリー (バイト単位)。
cpu.current.user	ユーザースライスの CPU 使用率。
cpu.current.system	システムの CPU 使用率。
cpu.current.idle	アイドル状態の CPU 使用率。

7.165. NUMANODEPIN STRUCT

仮想 NUMA ノードの物理 NUMA ノードへのピンングを表します。

表7.217 属性の概要

Name	タイプ	概要
host_numa_node	NumaNode	非推奨。
index	Integer	仮想 NUMA ノードがピンングされている物理 NUMA ノードのインデックス。

Name	タイプ	概要
pinned	ブール値	非推奨。

7.165.1. host_numa_node

非推奨。機能はありません。

7.165.2. pinned

非推奨。常に **true** である必要があります。

7.166. NUMATUNEMODE ENUM

表7.218 値の概要

Name	概要
interleave	
preferred	
strict	

7.167. OPENSTACKIMAGE 構造体

表7.219 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.220 リンクの概要

Name	タイプ	概要
openstack_image_provider	OpenStackImageProvider	

7.168. OPENSTACKIMAGEPROVIDER STRUCT

表7.221 属性の概要

Name	タイプ	概要
authentication_url	String	外部プロバイダー認証 URL アドレスを定義します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
password	String	認証プロセス中にユーザーのパスワードを定義します。
properties	Property[]	プロバイダー名/値のプロパティの配列。
requires_authentication	ブール値	プロバイダー認証が必要かどうかを定義します。
tenant_name	String	
url	String	外部プロバイダーの URL アドレスを定義します。
username	String	認証プロセス中に使用されるユーザー名を定義します。

7.168.1. requires_authentication

プロバイダー認証が必要かどうかを定義します。

認証が必要な場合、**username** と **password** の両方の属性が認証時に使用されます。

表7.222 リンクの概要

Name	タイプ	概要
certificates	Certificate[]	
images	OpenStackImage[]	

7.169. OPENSTACKNETWORK 構造体

表7.223 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.224 リンクの概要

Name	タイプ	概要
openstack_net work_provider	OpenStackNetworkProvider	

7.170. OPENSTACKNETWORKPROVIDER 構造体

表7.225 属性の概要

Name	タイプ	概要
agent_configuration	AgentConfiguration	エージェントの設定。
authentication_url	String	外部プロバイダー認証 URL アドレスを定義します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
password	String	認証プロセス中にユーザーのパスワードを定義します。
plugin_type	NetworkPluginType	ネットワークプラグインのタイプ。
properties	Property[]	プロバイダー名/値のプロパティの配列。
read_only	ブール値	プロバイダーが読み取り専用かどうかを示します。

Name	タイプ	概要
requires_authentication	ブール値	プロバイダー認証が必要かどうかを定義します。
tenant_name	String	
type	OpenStackNetworkProviderType	プロバイダーのタイプ。
url	String	外部プロバイダーの URL アドレスを定義します。
username	String	認証プロセス中に使用されるユーザー名を定義します。

7.170.1. read_only

プロバイダーが読み取り専用かどうかを示します。

読み取り専用プロバイダーでは、ネットワークまたはサブネットの追加、変更、削除は許可されません。ポート関連の操作は、仮想 NIC のプロビジョニングに必要なため、許可されます。

7.170.2. requires_authentication

プロバイダー認証が必要かどうかを定義します。

認証が必要な場合、**username** と **password** の両方の属性が認証時に使用されます。

表7.226 リンクの概要

Name	タイプ	概要
certificates	Certificate[]	証明書リストへの参照。
networks	OpenStackNetwork[]	OpenStack ネットワークリストへの参照。
subnets	OpenStackSubnet[]	OpenStack ネットワークのサブネットリストへの参照。

7.171. OPENSTACKNETWORKPROVIDERTYPE ENUM

OpenStack ネットワークプロバイダーは、OpenStack Neutron によって実装できます。この場合、Neutron エージェントはホストに自動的にインストールされます。または、OpenStack API を実装する外部プロバイダーにすることもできます。この場合、仮想インターフェイスドライバーはカスタムソリューションであり、手動でインストールされます。

表7.227 値の概要

Name	概要
external	プロバイダーが OpenStack Neutron API を実装する外部プロバイダーであることを示します。
neutron	プロバイダーが OpenStack Neutron であることを示します。

7.171.1. external

プロバイダーが OpenStack Neutron API を実装する外部プロバイダーであることを示します。この場合の仮想インターフェイスドライバーは、外部プロバイダーによって実装されます。

7.171.2. neutron

プロバイダーが OpenStack Neutron であることを示します。標準の OpenStack Neutron エージェントは、仮想インターフェイスドライバーとして使用されます。

7.172. OPENSTACKPROVIDER 構造体

表7.228 属性の概要

Name	タイプ	概要
authentication_url	String	外部プロバイダー認証 URL アドレスを定義します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
password	String	認証プロセス中にユーザーのパスワードを定義します。
properties	Property[]	プロバイダー名/値のプロパティの配列。
requires_authentication	ブール値	プロバイダー認証が必要かどうかを定義します。
tenant_name	String	
url	String	外部プロバイダーの URL アドレスを定義します。
username	String	認証プロセス中に使用されるユーザー名を定義します。

7.172.1. requires_authentication

プロバイダー認証が必要かどうかを定義します。

認証が必要な場合、**username** と **password** の両方の属性が認証時に使用されます。

7.173. OPENSTACKSUBNET 構造体

表7.229 属性の概要

Name	タイプ	概要
cidr	String	ネットワーク CIDR を定義します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
dns_servers	String[]	DNS サーバーのリストを定義します。
gateway	String	IP ゲートウェイを定義します。
id	String	一意の ID
ip_version	String	IP バージョンを定義します。
name	String	人間が判読できるプレーンテキストでの名前。

7.173.1. ip_version

IP バージョンを定義します。

IPv6 では、値は **v4' for IPv4 or `v6** になります。

表7.230 リンクの概要

Name	タイプ	概要
openstack_net work	OpenStackNetwork	OpenStack ネットワークを管理するサービスへの参照。

7.174. OPENSTACKVOLUMEPROVIDER 構造体

表7.231 属性の概要

Name	タイプ	概要
authentication_url	String	外部プロバイダー認証 URL アドレスを定義します。

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
password	String	認証プロセス中にユーザーのパスワードを定義します。
properties	Property[]	プロバイダー名/値のプロパティの配列。
requires_authentication	ブール値	プロバイダー認証が必要かどうかを定義します。
tenant_name	String	
url	String	外部プロバイダーの URL アドレスを定義します。
username	String	認証プロセス中に使用されるユーザー名を定義します。

7.174.1. requires_authentication

プロバイダー認証が必要かどうかを定義します。

認証が必要な場合、**username** と **password** の両方の属性が認証時に使用されます。

表7.232 リンクの概要

Name	タイプ	概要
authentication_keys	OpenstackVolumeAuthenticationKey[]	
certificates	Certificate[]	
data_center	DataCenter	
volume_types	OpenStackVolumeType[]	

7.175. OPENSTACKVOLUMETYPE STRUCT

表7.233 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
properties	Property[]	

表7.234 リンクの概要

Name	タイプ	概要
openstack_volumeprovider	OpenStackVolumeProvider	

7.176. OPENSTACKVOLUMEAUTHENTICATIONKEY STRUCT

表7.235 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
creation_date	日付	
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
usage_type	OpenstackVolumeAuthenticationKeyUsageType	
uuid	String	
value	String	

表7.236 リンクの概要

Name	タイプ	概要
openstack_volume_provider	OpenStackVolume Provider	

7.177. OPENSTACKVOLUMEAUTHENTICATIONKEYUSAGETYPE ENUM

表7.237 値の概要

Name	概要
ceph	

7.178. OPERATINGSYSTEM STRUCT

オペレーティングシステムを説明する情報。これは、仮想マシンとホストの両方に使用されます。

表7.238 属性の概要

Name	タイプ	概要
boot	Boot	
cmdline	String	
custom_kernel_cmdline	String	ホストカーネルコマンドラインのカスタム部分。
initrd	String	
kernel	String	
reported_kernel_cmdline	String	実行中のホストによって報告されるホストカーネルコマンドライン。
type	String	
version	バージョン	

7.178.1. custom_kernel_cmdline

ホストカーネルコマンドラインのカスタム部分。これは、既存のカーネルコマンドラインと統合されます。

この属性によって実装された変更を適用するには、ホストを再インストールしてから再起動する必要があります。

各ホストデプロイの手順を実行中に、前のホストデプロイの手順で追加されたカーネルパラメーターは **grubby --update-kernel DEFAULT --remove-args <previous_custom_params>** を使用して削除され、現在のカーネルコマンドラインのカスタマイズは **grubby --update-kernel DEFAULT --args <custom_params>** を使用して適用されます。Manager は、最後に適用されたカーネルパラメーターのカスタマイズを内部的に追跡します。



注記

この属性は現在、ホストに対してのみ使用されます。

7.178.2. reported_kernel_cmdline

実行中のホストによって報告されるホストカーネルコマンドライン。

これは読み取り専用の属性です。この属性を変更しようとしても、黙って無視されます。



注記

この属性は現在、ホストに対してのみ使用されます。

7.179. OPERATINGSYSTEMINFO STRUCT

表7.239 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
large_icon	アイコン	
name	String	人間が判読できるプレーンテキストでの名前。
small_icon	アイコン	

7.180. OPTION 構造体

表7.240 属性の概要

Name	タイプ	概要
name	String	
type	String	
value	String	

7.181. OSTYPE ENUM

表7.241 値の概要

Name	概要
other	
other_linux	
rhel_3	
rhel_3x64	
rhel_4	
rhel_4x64	
rhel_5	
rhel_5x64	
rhel_6	
rhel_6x64	
unassigned	
windows_2003	
windows_2003x 64	
windows_2008	
windows_2008r 2x64	
windows_2008x 64	
windows_2012x 64	
windows_7	
windows_7x64	

Name	概要
windows_8	
windows_8x64	
windows_xp	

7.182. PACKAGE STRUCT

パッケージを表すタイプ。

これは、package 要素の例です。

```
<package>
  <name>libipa_hbac-1.9.2-82.11.el6_4.i686</name>
</package>
```

表7.242 属性の概要

Name	タイプ	概要
name	String	パッケージの名前

7.183. PAYLOAD 構造体

表7.243 属性の概要

Name	タイプ	概要
files	File[]	
type	VmDeviceType	
volume_id	String	

7.184. PAYLOADENCODING ENUM

表7.244 値の概要

Name	概要
base64	
plaintext	

7.185. PERMISSION STRUCT

表7.245 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.246 リンクの概要

Name	タイプ	概要
cluster	Cluster	
data_center	DataCenter	
disk	ディスク	
group	グループ	
host	ホスト	
role	ロール	
storage_domain	StorageDomain	
template	Template	
user	User	
vm	Vm	
vm_pool	VmPool	

7.186. PERMIT 構造体

タイプは permit を表します。

表7.247 属性の概要

Name	タイプ	概要
administrative	ブール値	permit が管理上のものかどうかを指定します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.248 リンクの概要

Name	タイプ	概要
role	ロール	permit が属するロールへの参照。

7.187. PMPROXY 構造体

表7.249 属性の概要

Name	タイプ	概要
type	PmProxyType	

7.188. PMPROXYTYPE ENUM

表7.250 値の概要

Name	概要
cluster	フェンスプロキシは、フェンスされたホストと同じクラスターから選択されます。
dc	フェンスプロキシは、フェンスされたホストと同じデータセンターから選択されません。
other_dc	フェンスプロキシは、フェンスされたホストとは異なるデータセンターから選択されます。

7.189. POLICYUNITTYPE ENUM

すべての内部ポリシーユニットタイプのタイプを保持します。

表7.251 値の概要

Name	概要
filter	
load_balancing	
weight	

7.190. PORTMIRRORING STRUCT

7.191. POWERMANAGEMENT STRUCT

表7.252 属性の概要

Name	タイプ	概要
address	String	ホストのホスト名または IP アドレスです。
agents	Agent[]	複数のフェンスが使用されている場合のフェンスエージェントオプションを指定します。
automatic_pm_enabled	ブール値	エネルギーを節約するために、ホストの自動電源制御を切り替えます。
enabled	ブール値	電源管理設定が有効か無効かを示します。
kdump_detection	ブール値	ホストをシャットダウンする前に、ホスト上で kdump が実行されているかどうかを判断するかどうかを切り替えます。
options	Option[]	オプション name="" および value="" 文字列で指定された、選択した type= のフェンシングオプション。
password	String	電源管理用の有効で堅牢なパスワード。
pm_proxies	PmProxy[]	電源管理プロキシを決定します。
status	PowerManagementStatus	ホストの電源ステータスを決定します。
type	String	フェンシングデバイスのコード。
username	String	電源管理用の有効なユーザー名。

7.191.1. agents

複数のフェンスが使用されている場合のフェンスエージェントオプションを指定します。

order サブ要素を使用して、フェンスエージェントに優先順位を付けます。エージェントは、フェンスアクションが成功するまで、その順序に従って順次実行されます。2つ以上のフェンスエージェントの順序が同じ場合、それらは同時に実行されます。その他のサブ要素には、type、ip、user、password、および options が含まれます。

7.191.2. automatic_pm_enabled

エネルギーを節約するために、ホストの自動電源制御を切り替えます。true に設定すると、クラスタの負荷が低い場合にホストの電源が自動的にオフになり、必要に応じて再び電源がオンになります。これは、ユーザーが無効にしない限り、ホスト作成時に true に設定されます。

7.191.3. kdump_detection

ホストをシャットダウンする前に、ホスト上で kdump が実行されているかどうかを判断するかどうかを切り替えます。true に設定すると、ホストは kdump プロセス中にシャットダウンしません。これは、ユーザーが無効にしない限り、ホストで電源管理が有効になっている場合に true に設定されます。

7.191.4. type

フェンシングデバイスのコード。

有効なフェンシングデバイスコードの一覧は、**capabilities** コレクションにあります。

7.192. POWERMANAGEMENTSTATUS ENUM

表7.253 値の概要

Name	概要
off	ホストは OFF です。
on	ホストが ON です。
unknown	不明なステータス。

7.193. PRODUCT 構造体

表7.254 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

7.194. PRODUCTINFO STRUCT

製品情報です。

エントリーポイントには、API ユーザーが Red Hat Virtualization 環境の正当性を判断するのに役立つ **product_info** 要素が含まれています。これには、製品名、**vendor**、**version** 含まれます。

本物の Red Hat Virtualization 環境を検証する

以下の要素は、本物の Red Hat Virtualization 環境を識別します。

```
<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>
```

表7.255 属性の概要

Name	タイプ	概要
name	String	oVirt Engine などの製品名です。
vendor	String	ベンダーの名前 (例: `ovirt)`。
version	バージョン	製品のバージョン番号。

7.194.1. vendor

ベンダーの名前 (例: **ovirt.org**)。

7.195. PROFILEDETAIL STRUCT

表7.256 属性の概要

Name	タイプ	概要
block_statistics	BlockStatistic[]	
duration	Integer	

Name	タイプ	概要
fop_statistics	FopStatistic[]	
profile_type	String	
statistics	Statistic[]	

7.196. PROPERTY 構造体

表7.257 属性の概要

Name	タイプ	概要
name	String	
value	String	

7.197. PROXYTICKET STRUCT

表7.258 属性の概要

Name	タイプ	概要
value	String	

7.198. QCOWVERSION ENUM

QCOW バージョンは、ボリュームがサポートする qemu バージョンを qemu に指定します。

このフィールドは更新 API を使用して更新でき、QCOW ボリュームについてのみ報告されます。これは、ディスクが作成されたストレージドメインのバージョンによって決まります。V4 より前のバージョンのストレージドメインは QCOW2 バージョン 2 ボリュームをサポートしますが、V4 ストレージドメインは QCOW2 バージョン 3 もサポートします。異なる QCOW バージョンの機能の詳細は、[こちら](#)を参照してください。

表7.259 値の概要

Name	概要
qcow2_v2	Copy On Write のデフォルトの互換性バージョン。これは、どの QEMU でも使えるということを意味します。
qcow2_v3	QEMU1 で導入された Copy On Write 互換バージョン。

7.198.1. qcow2_v3

QEMU 1.1 で導入された Copy On Write 互換バージョン。これは、新しいフォーマットが使用されていることを意味します。

7.199. QOS STRUCT

このタイプは、サービスの品質 (QoS) を定義する属性を表します。

ストレージの **type** は `storage` で、属性 **max_throughput**、**max_read_throughput**、**max_write_throughput**、**max_iops**、**max_read_iops** および **max_write_iops** が関連しています。

コンピューティング機能を持つリソースの場合、**type** は `cpu` で、属性 **cpu_limit** が関連しています。

仮想マシンネットワークの **type** は `network` で、属性 **inbound_average**、**inbound_peak**、**inbound_burst**、**outbound_average**、**outbound_peak**、および **outbound_burst** が関連しています。

ホストネットワークの場合、**type** は `hostnetwork` で、属性 **outbound_average_linkshare**、**outbound_average_upperlimit**、および **outbound_average_realtime** が関連しています。

表7.260 属性の概要

Name	タイプ	概要
comment	<code>String</code>	このオブジェクトに関するコメントを含むフリーテキスト。
cpu_limit	<code>Integer</code>	最大処理能力 (単位: %)。
description	<code>String</code>	プレーンテキストでの人間が判読できる説明。
id	<code>String</code>	一意の ID
inbound_average	<code>Integer</code>	Mbps 単位の望ましい平均インバウンドビットレート。
inbound_burst	<code>Integer</code>	1回のバーストで配信できるデータ量(MiB 単位)。
inbound_peak	<code>Integer</code>	最大インバウンドレート(Mbps)。
max_iops	<code>Integer</code>	1秒あたりの最大許容入出力操作数。
max_read_iops	<code>Integer</code>	1秒あたりの最大許容入力操作数。
max_read_throughput	<code>Integer</code>	読み取り操作の最大許容スループット。
max_throughput	<code>Integer</code>	最大許容総スループット。
max_write_iops	<code>Integer</code>	1秒あたりの最大許容出力操作数。

Name	タイプ	概要
max_write_throughput	Integer	書き込み操作の最大許容スループット。
name	String	人間が判読できるプレーンテキストでの名前。
outbound_average	Integer	Mbps 単位の望ましい平均アウトバウンドビットレート。
outbound_average_linkshare	Integer	重み共有。
outbound_average_realtime	Integer	Mbps 単位のコミットされたレート。
outbound_average_upperlimit	Integer	ネットワークが使用する最大帯域幅(Mbps 単位)。
outbound_burst	Integer	MiB 単位の 1 回のバーストで送信できるデータの量。
outbound_peak	Integer	Mbps 単位の最大アウトバウンドレート。
type	QosType	このエントリーに割り当てることができるリソースの種類。

7.199.1. cpu_limit

最大処理能力 (単位: %)。

コンピューティングリソースの設定に使用されます。

7.199.2. inbound_average

Mbps 単位の望ましい平均インバウンドビットレート。

仮想マシンのネットワークを設定するために使用します。定義されている場合、**inbound_peak** と **inbound_burst** も設定される必要があります。

詳細は、[Libvirt-QOS](#) を参照してください。

7.199.3. inbound_burst

1 回のバーストで配信できるデータ量(MiB 単位)。

仮想マシンのネットワークを設定するために使用します。定義されている場合、**inbound_average** と **inbound_peak** も設定する必要があります。

詳細は、[Libvirt-QOS](#) を参照してください。

7.199.4. inbound_peak

最大インバウンドレート(Mbps)。

仮想マシンのネットワークを設定するために使用します。定義されている場合、**inbound_average** と **inbound_burst** も設定する必要があります。

詳細は、[Libvirt-QOS](#) を参照してください。

7.199.5. max_iops

1秒あたりの最大許容入出力操作数。

ストレージの設定に使用されます。**max_read_iops** または **max_write_iops** が設定されている場合、設定することはできません。

7.199.6. max_read_iops

1秒あたりの最大許容入力操作数。

ストレージの設定に使用されます。**max_iops** が設定されている場合、設定することはできません。

7.199.7. max_read_throughput

読み取り操作の最大許容スループット。

ストレージの設定に使用されます。**max_throughput** が設定されている場合は、設定することはできません。

7.199.8. max_throughput

最大許容総スループット。

ストレージの設定に使用されます。**max_read_throughput** または **max_write_throughput** が設定されている場合、設定することはできません。

7.199.9. max_write_iops

1秒あたりの最大許容出力操作数。

ストレージの設定に使用されます。**max_iops** が設定されている場合、設定することはできません。

7.199.10. max_write_throughput

書き込み操作の最大許容スループット。

ストレージの設定に使用されます。**max_throughput** が設定されている場合は、設定することはできません。

7.199.11. outbound_average

Mbps 単位の望ましい平均アウトバウンドビットレート。

仮想マシンのネットワークを設定するために使用します。定義されている場合、**outbound_peak** と **outbound_burst** も設定する必要があります。

詳細は、[Libvirt-QOS](#) を参照してください。

7.199.12. outbound_average_linkshare

重み共有。

ホストネットワークの設定に使用されます。同じ論理リンクリンクにアタッチされた他のネットワークと比較して、特定のネットワークに割り当てる必要がある論理リンクの容量を指定します。正確な共有は、そのリンクの全ネットワークの共有の合計によって異なります。デフォルトでは、この値は 1-100 の範囲の数字になります。

7.199.13. outbound_average_realtime

Mbps 単位のコミットされたレート。

ホストネットワークの設定に使用されます。ネットワークに必要な最小帯域幅。要求される Committed Rate は保証されず、ネットワークインフラストラクチャーおよび同じ論理リンクの他のネットワークによって要求される Committed Rate によって異なります。

7.199.14. outbound_average_upperlimit

ネットワークが使用する最大帯域幅(Mbps 単位)。

ホストネットワークの設定に使用されます。**outboundAverageUpperlimit** と **outbound_average_realtime** が提供される場合、**outbound_averageUpperlimit** は **outbound_average_realtime** より低くすることはできません。

詳細は、[Libvirt-QOS](#) を参照してください。

7.199.15. outbound_burst

MiB 単位の 1 回のバーストで送信できるデータの量。

仮想マシンのネットワークを設定するために使用します。定義されている場合、**outbound_average** と **outbound_peak** も設定する必要があります。

詳細は、[Libvirt-QOS](#) を参照してください。

7.199.16. outbound_peak

Mbps 単位の最大アウトバウンドレート。

仮想マシンのネットワークを設定するために使用します。定義されている場合、**outbound_average** と **outbound_burst** も設定する必要があります。

詳細は、[Libvirt-QOS](#) を参照してください。

表7.261 リンクの概要

Name	タイプ	概要
data_center	DataCenter	QoS が関連付けられているデータセンター。

7.200. QOSTYPE ENUM

このタイプは、[Quality of service \(QoS\)](#) を割り当てることができるリソースのタイプを表します。

表7.262 値の概要

Name	概要
cpu	Quality of service (QoS) は、コンピューティング機能を持つリソースに割り当てることができます。
hostnetwork	Quality of service (QoS) は、ホストネットワークに割り当てることができます。
network	Quality of service (QoS) は、仮想マシンネットワークに割り当てることができます。
storage	Quality of service (QoS) は、ストレージに割り当てることができます。

7.201. QUOTA 構造体

クォータオブジェクトを表します。

クォータのXML表現の例:

```
<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>
```

表7.263 属性の概要

Name	タイプ	概要
cluster_hard_limit_pct	Integer	
cluster_soft_limit_pct	Integer	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
data_center	DataCenter	
description	String	プレーンテキストでの人間が判読できる説明。
disks	Disk[]	

Name	タイプ	概要
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
storage_hard_limit_pct	Integer	
storage_soft_limit_pct	Integer	
users	User[]	
vms	Vm[]	

表7.264 リンクの概要

Name	タイプ	概要
permissions	Permission[]	
quota_cluster_limits	QuotaClusterLimit[]	
quota_storage_limits	QuotaStorageLimit[]	

7.202. QUOTA_CLUSTER_LIMIT STRUCT

表7.265 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
memory_limit	10 進数	
memory_usage	10 進数	
name	String	人間が判読できるプレーンテキストでの名前。

Name	タイプ	概要
vcpu_limit	Integer	
vcpu_usage	Integer	

表7.266 リンクの概要

Name	タイプ	概要
cluster	Cluster	
quota	クォータ	

7.203. QUOTAMODETYPE ENUM

表7.267 値の概要

Name	概要
audit	
disabled	
enabled	

7.204. QUOTASTORAGELIMIT STRUCT

表7.268 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
limit	Integer	
name	String	人間が判読できるプレーンテキストでの名前。
使用方法	10 進数	

表7.269 リンクの概要

Name	タイプ	概要
quota	クォータ	
storage_domain	StorageDomain	

7.205. RANGE 構造体

表7.270 属性の概要

Name	タイプ	概要
from	String	
上記を以下のように変更します。	String	

7.206. RATE 構造体

乱数ジェネレーターデバイスからの最大バイト消費速度を決定します。

表7.271 属性の概要

Name	タイプ	概要
bytes	Integer	期間ごとに消費できるバイト数。
period	Integer	1期間の長さ (ミリ秒単位)。

7.207. REPORTEDCONFIGURATION 構造体

表7.272 属性の概要

Name	タイプ	概要
actual_value	String	
expected_value	String	
in_sync	ブール値	ネットワークアタッチメントにコミットされていないネットワーク設定が含まれている場合は false 。
name	String	

7.208. REPORTEDDEVICE 構造体

表7.273 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
ips	Ip[]	
mac	Mac	
name	String	人間が判読できるプレーンテキストでの名前。
type	ReportedDeviceType	

表7.274 リンクの概要

Name	タイプ	概要
vm	Vm	

7.209. REPORTEDDEVICETYPE ENUM

表7.275 値の概要

Name	概要
network	

7.210. RESOLUTIONTYPE ENUM

表7.276 値の概要

Name	概要
add	
copy	

7.211. RNGDEVICE 構造体

乱数ジェネレーター (RNG) デバイスモデル。

表7.277 属性の概要

Name	タイプ	概要
rate	Rate	乱数ジェネレーターデバイスからの最大バイト消費速度を決定します。
source	RngSource	乱数ジェネレーターデバイスのバックエンド。

7.212. RNGSOURCE ENUM

乱数ジェネレーターのバックエンドタイプを表します。

表7.278 値の概要

Name	概要
hwrng	<code>/dev/hwrng</code> (通常は特殊な HW ジェネレーター) デバイスからランダムデータを取得します。
random	<code>/dev/random</code> デバイスからランダムデータを取得します。
urandom	<code>/dev/urandom</code> デバイスからランダムデータを取得します。

7.212.1. urandom

`/dev/urandom` デバイスからランダムデータを取得します。

この RNG ソースは、クラスターを認識しないエンティティ (つまり、空白のテンプレートとインスタンスタイプ) の **rbdom** RNG ソースと、互換性バージョン 4.1以降のクラスターに関連付けられたエンティティを置き換えることを目的としています。

7.213. ROLE 構造体

システムロールを表します。

表7.279 属性の概要

Name	タイプ	概要
administrative	ブール値	ロールを管理専用にするかどうかを定義します。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
mutable	ブール値	ロールを更新または削除する機能を定義します。

Name	タイプ	概要
name	String	人間が判読できるプレーンテキストでの名前。

7.213.1. mutable

ロールを更新または削除する機能を定義します。

mutable が **false** に設定されているロールは、定義済みのロールです。

表7.280 リンクの概要

Name	タイプ	概要
permits	Permit[]	ロール permits の permits サブコレクションへのリンク。
user	User	

7.214. ROLETYPE ENUM

ロールが管理ロールかどうかを表すタイプ。少なくとも1つの管理ロールを付与されたユーザーは、管理者と見なされます。

表7.281 値の概要

Name	概要
admin	管理ロール。
user	ユーザーロール。

7.215. SCHEDULINGPOLICY STRUCT

表7.282 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
default_policy	ブール値	
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
locked	ブール値	

Name	タイプ	概要
name	String	人間が判読できるプレーンテキストでの名前。
properties	Property[]	

表7.283 リンクの概要

Name	タイプ	概要
balances	Balance[]	
filters	Filter[]	
weight	Weight[]	

7.216. SCHEDULINGPOLICYUNIT STRUCT

表7.284 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
enabled	ブール値	
id	String	一意の ID
internal	ブール値	
name	String	人間が判読できるプレーンテキストでの名前。
properties	Property[]	
type	PolicyUnitType	

7.217. SCSIGENERICIO ENUM

表7.285 値の概要

Name	概要
filtered	

Name	概要
unfiltered	

7.218. SELINUX 構造体

表7.286 属性の概要

Name	タイプ	概要
mode	SeLinuxMode	

7.219. SELINUXMODE ENUM

表7.287 値の概要

Name	概要
disabled	
enforcing	
permissive	

7.220. SERIALNUMBER 構造体

表7.288 属性の概要

Name	タイプ	概要
policy	SerialNumberPolicy	
value	String	

7.221. SERIALNUMBERPOLICY ENUM

表7.289 値の概要

Name	概要
custom	
host	

Name	概要
vm	

7.222. SESSION 構造体

仮想マシンへのユーザーセッションについて説明します。

表7.290 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
console_user	ブール値	これがコンソールセッションであるかどうかを示します。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
ip	Ip	ユーザーの接続元 IP アドレス。
name	String	人間が判読できるプレーンテキストでの名前。
protocol	String	セッションで使用されるプロトコル。

7.222.1. console_user

これがコンソールセッションであるかどうかを示します。

コンソールユーザー (SPICE または VNC) の場合の値は **true**、その他 (RDP または SSH など) の場合は **false** になります。

7.222.2. ip

ユーザーの接続元 IP アドレス。

現在、コンソールユーザーのみが利用できます。

7.222.3. protocol

セッションで使用されるプロトコル。

現在使用されていません。SPICE、VNC、SSH、RDP など、ユーザーの接続方法に関する情報を対象としています。

表7.291 リンクの概要

Name	タイプ	概要
user	User	このセッションに関連するユーザー。
vm	Vm	このセッションに関連する仮想マシンへのリンク。

7.222.4. user

このセッションに関連するユーザー。

ユーザーがコンソールユーザーの場合、これは実際の Red Hat Virtualization ユーザーへのリンクです。それ以外の場合は、ユーザー名のみが提供されます。

7.223. SKIPIFCONNECTIVITYBROKEN STRUCT

表7.292 属性の概要

Name	タイプ	概要
enabled	ブール値	有効にすると、クラスター内の設定可能なパーセンテージを超えるホストが接続を失ってもホストのフェンシングは実行されません。
threshold	Integer	接続テストのしきい値。

7.223.1. enabled

有効にすると、クラスター内の設定可能なパーセンテージを超えるホストが接続を失ってもホストのフェンシングは実行されません。これは、クラスター内でグローバルネットワークの問題が発生した場合にフェンシングストームを防ぐためです。

7.223.2. threshold

接続テストのしきい値。少なくともクラスター内のホストのしきい値パーセンテージで接続が失われた場合、フェンシングは行われません。

7.224. SKIPIFSDACTIVE STRUCT

このタイプは、フェンシングポリシーのストレージ関連の設定を表します。

表7.293 属性の概要

Name	タイプ	概要
enabled	ブール値	有効にすると、ホストがストレージでリースを維持している場合にフェンシングをスキップします。

7.224.1. enabled

有効にすると、ホストがストレージでリースを維持している場合にフェンシングをスキップします。これは、ホストがまだストレージにアクセスできる場合、フェンシングされないことを意味します。

7.225. SNAPSHOT 構造体

スナップショットオブジェクトを表します。

XML 表現の例:

```
<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>
```

表7.294 属性の概要

Name	タイプ	概要
bios	Bios	仮想マシンの BIOS 設定への参照。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
console	コンソール	この仮想マシン用に設定されたコンソール。
cpu	Cpu	仮想マシン CPU の設定。
cpu_shares	Integer	
creation_time	日付	仮想マシンの作成日。
custom_compatibility_version	バージョン	仮想マシンのカスタム互換性バージョン。
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	さまざまなフックを設定するために VDSM に送信されるプロパティ。
date	日付	

Name	タイプ	概要
delete_protected	ブール値	true の場合、仮想マシンは削除できません。
description	String	プレーンテキストでの人間が判読できる説明。
display	表示	仮想マシンの表示設定。
domain	Domain	この仮想マシン用に設定されたドメイン。
fqdn	String	仮想マシンの完全修飾ドメイン名。
guest_operating_system	GuestOperatingSystem	仮想マシンにインストールされているオペレーティングシステム。
guest_time_zone	TimeZone	仮想マシンが使用するタイムゾーン (ゲストエージェントによって返されます)。
high_availability	高可用性	仮想マシンの高可用性設定。
id	String	一意の ID
initialization	初期化	仮想マシンの初期化設定への参照。
io	Io	IO スレッドのパフォーマンスチューニング用。
large_icon	アイコン	仮想マシンの大きなアイコン。
lease	StorageDomainLease	この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。
memory	Integer	仮想マシンのメモリー (バイト単位)。
memory_policy	MemoryPolicy	仮想マシンのメモリー管理設定への参照。
migration	MigrationOptions	実行中の仮想マシンの別のホストへの移行設定への参照。
migration_down_time	Integer	別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。
name	String	人間が判読できるプレーンテキストでの名前。
next_run_configuration_exists	ブール値	仮想マシンの設定が変更されたため、仮想マシンの再起動が必要です。

Name	タイプ	概要
numa_tune_mode	NumaTuneMode	NUMA トポロジーの適用方法。
origin	String	この仮想マシンのオリジン。
os	OperatingSystem	仮想マシンにインストールされているオペレーティングシステムのタイプ。
payloads	Payload[]	仮想マシンのオプションのペイロード。ISO が仮想マシンを設定するために使用されます。
persist_memory_state	ブール値	仮想マシンのメモリーの内容がスナップショットに含まれているかどうかを示します。
placement_policy	VmPlacementPolicy	仮想マシンの配置ポリシーの設定。
rng_device	RngDevice	この仮想マシンの乱数ジェネレーターデバイスの設定。
run_once	ブール値	true の場合、仮想マシンは run once コマンドを使用して開始されています。つまり、この1回の実行のために保存されている設定とは異なる可能性があります。
serial_number	SerialNumber	クラスター内の仮想マシンのシリアル番号。
small_icon	アイコン	仮想マシンの小さなアイコン。
snapshot_status	SnapshotStatus	
snapshot_type	SnapshotType	
soundcard_enabled	ブール値	true の場合、サウンドカードが仮想マシンに追加されます。
sso	Sso	この仮想マシンが設定されているシングルサインオン設定への参照。
start_paused	ブール値	true の場合、仮想マシンは起動後、最初は 'paused' 状態になります。
start_time	日付	仮想マシンが起動された日付。
stateless	ブール値	true の場合、仮想マシンはステートレスで、シャットダウン後にその状態 (ディスク) がロールバックされます。

Name	タイプ	概要
status	VmStatus	仮想マシンの現在の状態
status_detail	String	人間が読める現在のステータスの詳細。
stop_reason	String	仮想マシンが停止した理由。
stop_time	日付	仮想マシンが停止された日付。
time_zone	TimeZone	oVirt によって設定された仮想マシンのタイムゾーン。
tunnel_migration	ブール値	true の場合、ネットワークデータ転送は仮想マシンのライブマイグレーション中に暗号化されます。
type	VmType	仮想マシンがデスクトップとサーバーのどちらに最適化されているかを決定します。
usb	Usb	この仮想マシンの USB デバイスの設定 (カウント、タイプ)。
use_latest_template_version	ブール値	true の場合、仮想マシンは起動時にテンプレートの最新バージョンに再設定されます。
virtio_scsi	VirtioScsi	VirtIO SCSI 設定への参照。

7.225.1. cpu

仮想マシン CPU の設定。

ソケット設定は、仮想マシンを再起動せずに更新できます。コアとスレッドは再起動する必要があります。

たとえば、ソケットの数をすぐに 4 に変更し、再起動後にコアとスレッドの数を 2 に変更するには、以下のリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

7.225.2. custom_compatibility_version

仮想マシンのカスタム互換性バージョン。

仮想マシンを独自の互換性バージョンにカスタマイズできるようにします。**custom_compatibility_version** が設定されている場合、この特定の仮想マシンのクラスター互換性バージョンをオーバーライドします。

仮想マシンの互換バージョンは、仮想マシンが格納されているデータセンターによって制限され、仮想マシンが実行される予定のホストの機能に対してチェックされます。

7.225.3. high_availability

仮想マシンの高可用性設定。設定されている場合、仮想マシンが予期せずダウンしたときに自動的に再起動されます。

7.225.4. large_icon

仮想マシンの大きなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.225.5. lease

この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。

リースを使用して実行している仮想マシンは、この仮想マシンの別のインスタンスが別のホストで実行されるのを防ぐために、実行中にリースが別のホストによって取得されていないことを確認する必要があります。これにより、高可用性の仮想マシンでスプリットブレインが保護されます。このテンプレートから作成された仮想マシンを、このストレージドメインをリースの場所として事前設定するために、テンプレートにリース用に定義されたストレージドメインを含めることもできます。

7.225.6. memory

仮想マシンのメモリー (バイト単位)。

たとえば、1ギビバイト (GiB) のメモリーを含むように仮想マシンを更新するには、次の要求を送信します。

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

リクエスト本文は、以下のようになります。

```
<vm>  
<memory>1073741824</memory>  
</vm>
```



注記

この例のメモリーは、次の式を使用してバイトに変換されます:
 $1 \text{ GiB} = 2^{30} \text{ バイト} = 1073741824 \text{ バイト}$ 。



注記

メモリーホットプラグは、Red Hat Virtualization 3.6以降でサポートされています。上記の例では、仮想マシンの実行中にメモリーを増やすことができます。

7.225.7. migration_downtime

別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。

仮想マシンに対して明示的に設定するか、**engine-config -s DefaultMaximumMigrationDowntime=[value]** で設定します。

7.225.8. next_run_configuration_exists

仮想マシンの設定が変更されたため、仮想マシンの再起動が必要です。変更された設定は、仮想マシンのシャットダウン処理時に適用されます。

7.225.9. origin

この仮想マシンのオリジン。

値:

- **ovirt**
- **rhev**
- **vmware**
- **xen**
- **external**
- **hosted_engine**
- **managed_hosted_engine**
- **kvm**
- **physical_machine**
- **hyperv**

7.225.10. persist_memorystate

仮想マシンのメモリーの内容がスナップショットに含まれているかどうかを示します。

スナップショットが作成されるときデフォルト値は **true** です。

7.225.11. placement_policy

仮想マシンの配置ポリシーの設定。

この設定を更新して、仮想マシンを1つ以上のホストにピンングできます。



注記

複数のホストにピンングされた仮想マシンはライブマイグレーションできませんが、ホストに障害が発生した場合、高可用性になるように設定された仮想マシンは、仮想マシンがピンングされている他のホストの1つで自動的に再起動されます。

たとえば、仮想マシンを2つのホストに固定するには、以下のリクエストを送信します。

```
PUT /api/vms/123
```

リクエスト本文は以下のようになります。

```
<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

仮想マシンの小さなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.225.13. sso

この仮想マシンが設定されているシングルサインオン設定への参照。コンソールを開くと、ユーザーは仮想マシンのオペレーティングシステムに自動的にサインインできます。

7.225.14. stop_reason

仮想マシンが停止した理由。オプションで、仮想マシンをシャットダウンするときにユーザーが設定します。

表7.295 リンクの概要

Name	タイプ	概要
affinity_labels	AffinityLabel[]	オプション。

Name	タイプ	概要
applications	Application[]	仮想マシンにインストールされているアプリケーションのリスト。
cdroms	Cdrom[]	CDROM にマウントされた ISO への参照。
cluster	Cluster	仮想マシンが属するクラスターへの参照。
cpu_profile	CpuProfile	この仮想マシンによって使用される CPU プロファイルへの参照。
disk_attachments	DiskAttachment[]	仮想マシンにアタッチされているディスクへの参照。
external_host_provider	ExternalHostProvider	
floppies	Floppy[]	フロッピーにマウントされた ISO への参照。
graphics_consoles	GraphicsConsole[]	この仮想マシン用に設定されたグラフィックコンソールのリスト。
host	ホスト	仮想マシンが実行されているホストへの参照。
host_devices	HostDevice[]	この仮想マシンに関連付けられているデバイスへの参照。
instance_type	InstanceType	仮想マシンの設定は、インスタンスタイプの1つを介してオプションで事前定義できます。
katello_errata	KatelloErratum[]	仮想マシンに割り当てられたすべての Katello エラータを一覧表示します。
nics	Nic[]	仮想マシン上のネットワークインターフェイスデバイスのリストを参照します。
numa_nodes	NumaNode[]	この仮想マシンで使用される NUMA ノード設定を参照します。
original_template	Template	仮想マシンの作成に使用された元のテンプレートを参照します。
permissions	Permission[]	この仮想マシンに設定されたパーミッション。
quota	クォータ	この仮想マシンに設定されたクォータ設定への参照。
reported_devices	ReportedDevice[]	

Name	タイプ	概要
sessions	Session[]	この仮想マシンに対して開かれているユーザーセッションのリスト。
snapshots	Snapshot[]	仮想マシンから取得されたすべてのスナップショットを参照します。
statistics	Statistic[]	この仮想マシンから収集された統計データ。
storage_domain	StorageDomain	仮想マシンが属するストレージドメインへの参照。
tags	Tag[]	
template	Template	仮想マシンがベースとするテンプレートへの参照。
vm	Vm	
vm_pool	VmPool	仮想マシンが任意のメンバーであるプールへの参照。
watchdogs	Watchdog[]	ウォッチドッグ設定への参照。

7.225.15. affinity_labels

オプション。サブクラスターのラベル付けに使用されます。

7.225.16. katello_errata

仮想マシンに割り当てられたすべての Katello エラータを一覧表示します。

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

以下のような XML で応答を受け取ります。

```
<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

仮想マシンの作成に使用された元のテンプレートを参照します。

テンプレートまたは別の仮想マシンから仮想マシンをクローンした場合、**template** はブランクテンプレートにリンクし、履歴の追跡には **original_template** が使用されます。

それ以外の場合は、**template** と **original_template** は同じです。

7.226. SNAPSHOTSTATUS ENUM

表7.296 値の概要

Name	概要
in_preview	
locked	
ok	

7.227. SNAPSHOTTYPE ENUM

表7.297 値の概要

Name	概要
active	
preview	
regular	
stateless	

7.228. SPECIALOBJECTS 構造体

このタイプには、空白のテンプレートやタグ階層のルートなど、特別なオブジェクトへの参照が含まれます。

表7.298 リンクの概要

Name	タイプ	概要
blank_template	Template	空白のテンプレートへの参照。
root_tag	タグ	タグ階層のルートへの参照。

7.229. SPM 構造体

表7.299 属性の概要

Name	タイプ	概要
priority	Integer	
status	SpmStatus	

7.230. SPMSTATUS ENUM

表7.300 値の概要

Name	概要
contending	
none	
spm	

7.231. SSH 構造体

表7.301 属性の概要

Name	タイプ	概要
authentication_method	SshAuthentication Method	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
fingerprint	String	
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

Name	タイプ	概要
port	Integer	
user	User	

7.232. SSHAUTHENTICATIONMETHOD ENUM

表7.302 値の概要

Name	概要
password	
publickey	

7.233. SSHPUBLICKEY STRUCT

表7.303 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
content	String	
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.304 リンクの概要

Name	タイプ	概要
user	User	

7.234. SSO 構造体

表7.305 属性の概要

Name	タイプ	概要
メソッド	Method[]	

7.235. SSOMETHOD ENUM

表7.306 値の概要

Name	概要
guest_agent	

7.236. 統計 STRUCT

すべての種類の統計に使用されるジェネリックタイプ。

統計には、さまざまなエンティティの統計値が含まれています。次のオブジェクトには統計が含まれています:

- ディスク
- ホスト
- HostNic
- NumaNode
- Nic
- Vm
- GlusterBrick
- Step
- GlusterVolume

XML 表現の例:

```
<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>
  </statistic id="1234" href="/ovirt-engine/api/hosts/1234/nics/1234"/>
```


</statistic>

...

</statistics>



注記

この統計サブコレクションは読み取り専用です。

表7.307 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
kind	StatisticKind	統計測定値のタイプ。
name	String	人間が判読できるプレーンテキストでの名前。
type	ValueType	後続の統計値のデータタイプ。
unit	StatisticUnit	統計値を測定する単位または割合。
値	Value[]	datum を含むデータセット。

表7.308 リンクの概要

Name	タイプ	概要
brick	GlusterBrick	
disk	ディスク	含まれる disk リソースとの関係。
gluster_volume	GlusterVolume	
host	ホスト	
host_nic	HostNic	ホスト NIC への参照。
host_numa_node	NumaNode	
nic	Nic	

Name	タイプ	概要
step	Step	
vm	Vm	

7.237. STATISTICKIND ENUM

表7.309 値の概要

Name	概要
カウンター	
ゲージ	

7.238. STATISTICUNIT ENUM

表7.310 値の概要

Name	概要
bits_per_second	
bytes	
bytes_per_second	
count_per_second	
none	
percent	
秒	

7.239. STEP 構造体

job 実行の一部であるステップを表します。ステップは、より広いシーケンスの一部である特定の実行ユニットを記述および追跡するために使用されます。一部のステップでは、進行状況の報告がサポートされています。

表7.311 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
end_time	日付	ステップの終了時刻。
external	ブール値	ステップが外部システムによって開始されたかどうかを示します。
external_type	ExternalSystemType	ステップが参照する外部システム。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
number	Integer	現在の階層レベルでのステップの順序。
progress	Integer	ステップの進行状況 (報告されている場合) のパーセンテージ。
start_time	日付	ステップの開始時間。
status	StepStatus	ステップのステータス。
type	StepEnum	ステップのタイプ。

7.239.1. external

ステップが外部システムによって開始されたかどうかを示します。外部ステップは、ステップの作成者によって外部で管理されます。

表7.312 リンクの概要

Name	タイプ	概要
execution_host	ホスト	ステップの実行に使用されるホスト (オプション)。
job	Job	現在のステップ階層の最上位にある job を参照します。
parent_step	Step	階層内の現在のステップの親ステップを参照します。
statistics	Statistic[]	

7.240. STEPENUM ENUM

ステップタイプを表すタイプ。

表7.313 値の概要

Name	概要
executing	実行中のステップタイプ。
finalizing	最終処理中のステップタイプ。
rebalancing_volume	rebalancing volume ステップタイプ。
removing_bricks	removing bricks ステップタイプ。
unknown	不明なステップタイプ。
validating	検証ステップタイプ。

7.240.1. executing

実行中のステップタイプ。ジョブのメイン実行ブロックを追跡するために使用されます。通常、これは実行ステップの一部を説明するいくつかのサブステップの親ステップになります。

7.240.2. finalizing

最終処理中のステップタイプ。**job** を完了するために必要な実行後のステップについて説明します。

7.240.3. rebalancing_volume

rebalancing volume ステップタイプ。**Gluster** フローの一部であるステップタイプについて説明します。

7.240.4. removing_bricks

removing bricks ステップタイプ。**Gluster** フローの一部であるステップタイプについて説明します。

7.240.5. unknown

不明なステップタイプ。起源が不明なステップタイプについて説明します。

7.240.6. validating

検証ステップタイプ。実行前に、パラメーターの正常性とパラメーターの有効性を検証するために使用されます。

7.241. STEPSTATUS ENUM

ステップのステータスを表します。

表7.314 値の概要

Name	概要
aborted	中止されたステップのステータス。
failed	失敗したステップのステータス。
finished	終了したステップのステータス。
started	開始されたステップのステータス。
unknown	不明なステップステータス。

7.241.1. aborted

中止されたステップのステータス。このステータスは、強制的に中止された外部ステップに適用されません。

7.241.2. finished

終了したステップのステータス。このステータスは、完了したステップの実行を示しています。

7.241.3. started

開始されたステップのステータス。このステータスは、現在実行中のステップを表します。

7.241.4. unknown

不明なステップステータス。このステータスは、解決方法が不明なステップ、つまり、システムが予期せず再起動される前に実行されたステップを表します。

7.242. STORAGECONNECTION 構造体

ストレージサーバー接続を表します。

XML 表現の例:

```
<storage_connection id="123">
  <address>my nfs.example.com</address>
  <type>nfs</type>
  <path>/exports/mydata</path>
</storage_connection>
```

表7.315 属性の概要

Name	タイプ	概要
address	String	

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
mount_options	String	
name	String	人間が判読できるプレーンテキストでの名前。
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	

表7.316 リンクの概要

Name	タイプ	概要
host	ホスト	

7.243. STORAGECONNECTIONEXTENSION 構造体

表7.317 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。
password	String	
target	String	
username	String	

表7.318 リンクの概要

Name	タイプ	概要
host	ホスト	

7.244. STORAGEDOMAIN 構造体

ストレージドメイン。

識別子 **123** を持つ NFS ストレージドメインの XML 表現です。

```
<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>

```

表7.319 属性の概要

Name	タイプ	概要
available	Integer	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
committed	Integer	
critical_space_action_blocker	Integer	
description	String	プレーンテキストでの人間が判読できる説明。
discard_after_delete	ブール値	ブロック storage domain 上の disk のブロックが削除される直前に破棄されるかどうかを示します。
external_status	ExternalStatus	
id	String	一意の ID
import	ブール値	
マスター	ブール値	
name	String	人間が判読できるプレーンテキストでの名前。
status	StorageDomainStatus	
storage	HostStorage	
storage_format	StorageFormat	
supports_discard	ブール値	ブロックストレージドメインが破棄操作をサポートしているかどうかを示します。
supports_discard zeroes_data	ブール値	ブロックストレージドメインがデータを破棄するプロパティをサポートしているかどうかを示します。
type	StorageDomainType	
used	Integer	

Name	タイプ	概要
warning_low_space_indicator	Integer	
wipe_after_delete	ブール値	このストレージドメイン上のディスクの wipe_after_delete のデフォルト値として機能します。

7.244.1. discard_after_delete

ブロック `storage domain` 上の `disk` のブロックが削除される直前に破棄されるかどうかを示します。

`true` に設定され、このストレージドメインのディスクで **wipe_after_delete** 値が有効になっている場合、ディスクが削除されると次のようになります。

1. まず消去されます。
2. その後、そのブロックは破棄されます。
3. 最後に削除されます。

以下の点に留意してください。

- 削除後の破棄は、非ブロックストレージタイプでは常に **false** になります。
- 削除後に破棄を **true** に設定できるのは、ストレージドメインが **破棄をサポート** している場合のみです。

7.244.2. supports_discard

ブロックストレージドメインが破棄操作をサポートしているかどうかを示します。 **ストレージドメイン** は、構築元のすべての **論理ユニット** が破棄をサポートしている場合、つまり各論理ユニットの **discard_max_size** 値が 0 より大きい場合にのみ破棄をサポートします。これは、このストレージドメイン内の仮想ディスクの **pass_discard** 属性を有効にするために必要な条件の1つです。エンジンは、基礎となるブロックデバイスがファイルストレージドメインの破棄をサポートしているかどうかを確認できないため、この属性はまったく報告されません。

7.244.3. supports_discard_zeroes_data

ブロックストレージドメインがデータを破棄するプロパティをサポートしているかどうかを示します。 **ストレージドメイン** は、構築元のすべての **論理ユニット** がサポートする場合、つまり各論理ユニットの **discard_zeroes_data** 値が `true` の場合にのみ、データを破棄してゼロにするプロパティをサポートします。これは、このストレージドメインの仮想ディスクで **wipe_after_delete** 属性と **pass_discard** 属性の両方を有効にするために必要な条件の1つです。エンジンは、基礎となるブロックデバイスがファイルストレージドメインのデータを破棄するプロパティをサポートしているかどうかをチェックすることができないため、この属性はまったく報告されません。

7.244.4. wipe_after_delete

この **ストレージドメイン** 上の **ディスク** の **wipe_after_delete** のデフォルト値として機能します。

つまり、新しく作成されたディスクは、デフォルトでストレージドメインから **wipe_after_delete** 値を取得します。なお、設定値 **SANWipeAfterDelete** は、ブロックストレージドメインの **wipe_after_delete** のデフォルト値として機能します。

表7.320 リンクの概要

Name	タイプ	概要
data_center	DataCenter	ストレージドメインがアタッチされているデータセンターへのリンク。
data_centers	DataCenter[]	ストレージドメインがアタッチされているデータセンターへの一連のリンク。
disk_profiles	DiskProfile[]	
disk_snapshots	DiskSnapshot[]	
disks	Disk[]	
files	File[]	
host	ホスト	ホストは作成時にのみ関連します。
images	Image[]	
permissions	Permission[]	
storage_connections	StorageConnection[]	
templates	Template[]	
vms	Vm[]	

7.244.5. data_center

ストレージドメインがアタッチされているデータセンターへのリンク。ストレージドメインは複数のデータセンターにアタッチされる可能性があるため (ISO ドメインの場合)、これは後方互換性のためにのみ保持されます。代わりに **dataCenters** 要素を使用します。

7.245. STORAGEDOMAINLEASE 構造体

ストレージドメインに存在するリースを表します。

リースは、ストレージドメインの特別なボリュームに存在する [Sanlock](#) リソースです。この Sanlock リソースは、ストレージベースのロックを提供するために使用されます。

表7.321 リンクの概要

Name	タイプ	概要
storage_domain	StorageDomain	ロックが存在するストレージドメインへの参照。

7.246. STORAGEDOMAINSTATUS ENUM

表7.322 値の概要

Name	概要
activating	
active	
detaching	
inactive	
locked	
maintenance	
mixed	
preparing_for_maintenance	
unattached	
unknown	

7.247. STORAGEDOMAINSTYPE ENUM

表7.323 値の概要

Name	概要
data	
export	
image	
iso	
volume	

7.248. STORAGEFORMAT ENUM

表7.324 値の概要

Name	概要
v1	
v2	
v3	
v4	ストレージドメインフォーマットのバージョン 4。

7.249. STORAGETYPE ENUM

ストレージドメインのタイプを表すタイプ。

表7.325 値の概要

Name	概要
cinder	Cinder ストレージドメイン。
fcp	Fibre-Channel ストレージドメイン。
glance	Glance ストレージドメイン。
glusterfs	Gluster-FS ストレージドメイン。
iscsi	iSCSI ストレージドメイン。
localfs	ローカルストレージ上のストレージドメイン。
nfs	NFS ストレージドメイン。
posixfs	POSIX-FS ストレージドメイン。

7.249.1. cinder

Cinder ストレージドメイン。Cinder の詳細については、[Cinder](#) を参照してください。

7.249.2. glance

Glance ストレージドメイン。Glance の詳細については、[Glance](#) を参照してください。

7.249.3. glusterfs

Gluster-FS ストレージドメイン。Gluster の詳細については、[Gluster](#) を参照してください。

7.250. SWITCHTYPE ENUM

Manager がサポートするすべてのスイッチタイプについて説明します。

表7.326 値の概要

Name	概要
legacy	ネイティブスイッチタイプ。
ovs	Open vSwitch タイプ。

7.251. TAG 構造体

システム内のタグを表します。

表7.327 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.328 リンクの概要

Name	タイプ	概要
group	グループ	このタグが割り当てられているグループへの参照。
host	ホスト	このタグが割り当てられているホストへの参照。
parent	タグ	このタグの親タグへの参照。
template	Template	このタグが割り当てられたテンプレートへの参照。
user	User	このタグが割り当てられているユーザーへの参照。
vm	Vm	このタグが割り当てられている仮想マシンへの参照。

7.252. TEMPLATE 構造体

仮想マシンテンプレートを表すタイプ。これにより、共通の設定とディスク状態での仮想マシンの迅速な設定が可能になります。

表7.329 属性の概要

Name	タイプ	概要
bios	Bios	仮想マシンの BIOS 設定への参照。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
console	コンソール	この仮想マシン用に設定されたコンソール。
cpu	Cpu	仮想マシン CPU の設定。
cpu_shares	Integer	
creation_time	日付	仮想マシンの作成日。
custom_compatibility_version	バージョン	仮想マシンのカスタム互換性バージョン。
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	さまざまなフックを設定するために VDSM に送信されるプロパティ。
delete_protected	ブール値	true の場合、仮想マシンは削除できません。
description	String	プレーンテキストでの人間が判読できる説明。
display	表示	仮想マシンの表示設定。
domain	Domain	この仮想マシン用に設定されたドメイン。
high_availability	高可用性	仮想マシンの高可用性設定。
id	String	一意の ID
initialization	初期化	仮想マシンの初期化設定への参照。
io	io	IO スレッドのパフォーマンスチューニング用。
large_icon	アイコン	仮想マシンの大きなアイコン。

Name	タイプ	概要
lease	StorageDomainLease	この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。
memory	Integer	仮想マシンのメモリー (バイト単位)。
memory_policy	MemoryPolicy	仮想マシンのメモリー管理設定への参照。
migration	MigrationOptions	実行中の仮想マシンの別のホストへの移行設定への参照。
migration_down_time	Integer	別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。
name	String	人間が判読できるプレーンテキストでの名前。
origin	String	この仮想マシンのオリジン。
os	OperatingSystem	仮想マシンにインストールされているオペレーティングシステムのタイプ。
rng_device	RngDevice	この仮想マシンの乱数ジェネレーターデバイスの設定。
serial_number	SerialNumber	クラスター内の仮想マシンのシリアル番号。
small_icon	アイコン	仮想マシンの小さなアイコン。
soundcard_enabled	ブール値	true の場合、サウンドカードが仮想マシンに追加されます。
sso	Sso	この仮想マシンが設定されているシングルサインオン設定への参照。
start_paused	ブール値	true の場合、仮想マシンは起動後、最初は 'paused' 状態になります。
stateless	ブール値	true の場合、仮想マシンはステートレスで、シャットダウン後にその状態 (ディスク) がロールバックされます。
status	TemplateStatus	テンプレートのステータス
time_zone	TimeZone	oVirt によって設定された仮想マシンのタイムゾーン。
tunnel_migration	ブール値	true の場合、ネットワークデータ転送は仮想マシンのライブマイグレーション中に暗号化されます。
type	VmType	仮想マシンがデスクトップとサーバーのどちらに最適化されているかを決定します。

Name	タイプ	概要
usb	Usb	この仮想マシンの USB デバイスの設定 (カウント、タイプ)。
version	TemplateVersion	これが別のテンプレートのベースバージョンかサブバージョンであるかを示します。
virtio_scsi	VirtioScsi	VirtIO SCSI 設定への参照。
vm	Vm	このテンプレートに関連付けられている仮想マシンの設定。

7.252.1. cpu

仮想マシン CPU の設定。

ソケット設定は、仮想マシンを再起動せずに更新できます。コアとスレッドは再起動する必要があります。

たとえば、ソケットの数をすぐに 4 に変更し、再起動後にコアとスレッドの数を 2 に変更するには、以下のリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

7.252.2. custom_compatibility_version

仮想マシンのカスタム互換性バージョン。

仮想マシンを独自の互換性バージョンにカスタマイズできるようにします。**custom_compatibility_version** が設定されている場合、この特定の仮想マシンのクラスター互換性バージョンをオーバーライドします。

仮想マシンの互換バージョンは、仮想マシンが格納されているデータセンターによって制限され、仮想マシンが実行される予定のホストの機能に対してチェックされます。

7.252.3. high_availability

仮想マシンの高可用性設定。設定されている場合、仮想マシンが予期せずダウンしたときに自動的に再起動されます。

7.252.4. large_icon

仮想マシンの大きなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.252.5. lease

この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。

リースを使用して実行している仮想マシンは、この仮想マシンの別のインスタンスが別のホストで実行されるのを防ぐために、実行中にリースが別のホストによって取得されていないことを確認する必要があります。これにより、高可用性の仮想マシンでスプリットブレインが保護されます。このテンプレートから作成された仮想マシンを、このストレージドメインをリースの場所として事前設定するために、テンプレートにリース用に定義されたストレージドメインを含めることもできます。

7.252.6. memory

仮想マシンのメモリー (バイト単位)。

たとえば、1ギビバイト (GiB) のメモリーを含むように仮想マシンを更新するには、次の要求を送信します。

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

リクエスト本文は、以下のようになります。

```
<vm>
  <memory>1073741824</memory>
</vm>
```



注記

この例のメモリーは、次の式を使用してバイトに変換されます:
1 GiB = 2^{30} バイト = 1073741824 バイト。



注記

メモリーホットプラグは、Red Hat Virtualization 3.6 以降でサポートされています。上記の例では、仮想マシンの実行中にメモリーを増やすことができます。

7.252.7. migration_downtime

別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。

仮想マシンに対して明示的に設定するか、**engine-config -s DefaultMaximumMigrationDowntime=[value]** で設定します。

7.252.8. origin

この仮想マシンのオリジン。

値:

- **ovirt**
- **rhev**
- **vmware**
- **xen**
- **external**
- **hosted_engine**
- **managed_hosted_engine**
- **kvm**
- **physical_machine**
- **hyperv**

7.252.9. small_icon

仮想マシンの小さなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.252.10. sso

この仮想マシンが設定されているシングルサインオン設定への参照。コンソールを開くと、ユーザーは仮想マシンのオペレーティングシステムに自動的にサインインできます。

表7.330 リンクの概要

Name	タイプ	概要
cdroms	Cdrom[]	テンプレートにアタッチされた CD-ROM デバイスへの参照。
cluster	Cluster	仮想マシンが属するクラスターへの参照。
cpu_profile	CpuProfile	この仮想マシンによって使用される CPU プロファイルへの参照。
disk_attachments	DiskAttachment[]	テンプレートに割り当てられたディスクへの参照。
graphics_consoles	GraphicsConsole[]	テンプレートに接続されたグラフィックコンソールへの参照。
nics	Nic[]	テンプレートにアタッチされたネットワークインターフェイスへの参照。
permissions	Permission[]	テンプレートに割り当てられたユーザーパーミッションへの参照。

Name	タイプ	概要
quota	クォータ	この仮想マシンに設定されたクォータ設定への参照。
storage_domain	StorageDomain	仮想マシンが属するストレージドメインへの参照。
tags	Tag[]	テンプレートにアタッチされたタグへの参照。
watchdogs	Watchdog[]	テンプレートにアタッチされたウォッチドッグデバイスへの参照。

7.253. TEMPLATESTATUS ENUM

仮想マシンテンプレートのステータスを表すタイプ。

表7.331 値の概要

Name	概要
illegal	このステータスは、テンプレートの少なくとも1つのディスクが不正であることを示します。
locked	このステータスは、テンプレートでの他の操作を妨げる何らかの操作が実行されていることを示します。
ok	このステータスは、テンプレートが有効で使用できる状態であることを示します。

7.254. TEMPLATEVERSION 構造体

仮想マシンテンプレートのバージョンを表すタイプ。

表7.332 属性の概要

Name	タイプ	概要
version_name	String	このバージョンの名前。
version_number	Integer	テンプレートのバージョン階層におけるこのバージョンのインデックス。

7.254.1. version_number

テンプレートのバージョン階層におけるこのバージョンのインデックス。インデックス1は、ベースバージョンとも呼ばれるテンプレートの元のバージョンを表します。

表7.333 リンクの概要

Name	タイプ	概要
base_template	Template	このバージョンが関連付けられているテンプレートを参照します。

7.255. TICKET 構造体

仮想マシンへのアクセスを許可するチケットを表すタイプ。

表7.334 属性の概要

Name	タイプ	概要
expiry	Integer	チケットの有効期間 (秒単位)。
value	String	仮想マシンのアクセスチケット。

7.256. TIMEZONE 構造体

タイムゾーンの表現。

表7.335 属性の概要

Name	タイプ	概要
name	String	タイムゾーンの名前。
utc_offset	String	https://en からのオフセット。

7.256.1. utc_offset

[UTC](#) からのオフセット。

7.257. TRANSPARENTHUGEPAGES 構造体

Transparent huge pages (THP) のサポートを表すタイプ。

表7.336 属性の概要

Name	タイプ	概要
enabled	ブール値	THP サポートを有効にします。

7.258. TRANSPORTTYPE ENUM

Gluster ボリュームへのアクセスに使用されるプロトコル。

表7.337 値の概要

Name	概要
rdma	Remote direct memory access。
tcp	TCP。

7.259. UNMANAGEDNETWORK 構造体

表7.338 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.339 リンクの概要

Name	タイプ	概要
host	ホスト	
host_nic	HostNic	

7.260. USB 構造体

表7.340 属性の概要

Name	タイプ	概要
enabled	ブール値	
type	UsbType	

7.261. USBTYPE ENUM

表7.341 値の概要

Name	概要
legacy	レガシー USB タイプ。
native	

7.261.1. legacy

レガシー USB タイプ。

この USB タイプは、エンジンのバージョン 3.6 から非推奨となり、バージョン 4.1 で完全に削除されました。これは、既存のスクリプトの構文エラーを回避するためだけに保持されています。使用した場合は、自動的に **native** に置き換わります。

7.262. USER 構造体

システム内のユーザーを表します。

表7.342 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
department	String	
description	String	プレーンテキストでの人間が判読できる説明。
domain_entry_id	String	
email	String	
id	String	一意の ID
last_name	String	
logged_in	ブール値	
name	String	人間が判読できるプレーンテキストでの名前。
namespace	String	ユーザーが存在する名前空間。
password	String	
principal	String	user_name と同様です。

Name	タイプ	概要
user_name	String	ユーザーのユーザー名。

7.262.1. namespace

ユーザーが存在する名前空間。ユーザーを LDAP サーバーに格納する認可プロバイダーを使用する場合、この属性は LDAP サーバーのネーミングコンテキストと等しくなります。詳細は、<https://github.com/oVirt/ovirt-engine-extension-aaa-ldap> を参照してください。ユーザーをデータベースに格納するビルトイン認可プロバイダーを使用する場合、この属性は無視されます。詳細は、<https://github.com/oVirt/ovirt-engine-extension-aaa-jdbc> を参照してください。

7.262.2. principal

user_name と同様です。フォーマットは LDAP プロバイダーによって異なります。ほとんどの LDAP プロバイダーでは、**uid** LDAP 属性の値です。Active Directory の場合は、ユーザープリンシパル名 (UPN) です。

7.262.3. user_name

ユーザーのユーザー名。フォーマットは、認可プロバイダーのタイプによって異なります。ほとんどの LDAP プロバイダーでは、**uid** LDAP 属性の値です。Active Directory では、ユーザープリンシパル名 (UPN) です。**UPN** または **uid** の後には、認証プロバイダー名が続く必要があります。たとえば、LDAP の **uid** 属性の場合、**myuser@myextension-authz** です。**UPN** を使用する Active Directory の場合、**myuser@mysubdomain.mydomain.com@myextension-authz** です。この属性は、新しいユーザーを追加する際の必須パラメーターです。

表7.343 リンクの概要

Name	タイプ	概要
domain	Domain	
groups	Group[]	
permissions	Permission[]	
roles	Role[]	ユーザーリソースのロールサブコレクションへのリンク。
ssh_public_keys	SshPublicKey[]	
tags	Tag[]	ユーザーリソースのタグサブコレクションへのリンク。

7.263. VALUE 構造体

表7.344 属性の概要

Name	タイプ	概要
datum	10 進数	
detail	String	

7.264. VALUETYPE ENUM

表7.345 値の概要

Name	概要
decimal	
integer	
string	

7.265. VCPUPIN 構造体

表7.346 属性の概要

Name	タイプ	概要
cpu_set	String	
vcpu	Integer	

7.266. VENDOR 構造体

表7.347 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

7.267. VERSION 構造体

表7.348 属性の概要

Name	タイプ	概要
build	Integer	
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
full_version	String	
id	String	一意の ID
major	Integer	
minor	Integer	
name	String	人間が判読できるプレーンテキストでの名前。
revision	Integer	

7.268. VIRTIO SCSI STRUCT

virtio-SCSI のサポートを表すタイプ。サポートされている場合は、SCSI ゲストデバイスに virtio ドライバーを使用します。

表7.349 属性の概要

Name	タイプ	概要
enabled	ブール値	Virtio SCSI サポートを有効にします。

7.269. VIRTUALNUMANODE STRUCT

仮想 NUMA ノードを表します。

XML 表現の例:

```
<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>

```

表7.350 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
cpu	Cpu	
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
index	Integer	
memory	Integer	NUMA ノードのメモリー (MB)。
name	String	人間が判読できるプレーンテキストでの名前。
node_distance	String	
numa_node_pins	NumaNodePin[]	

表7.351 リンクの概要

Name	タイプ	概要
host	ホスト	
statistics	Statistic[]	各ホスト NUMA ノードリソースは、ホスト NUMA ノード固有の統計の統計サブコレクションを公開します。
vm	Vm	

7.269.1. statistics

各ホスト NUMA ノードリソースは、ホスト NUMA ノード固有の統計の統計サブコレクションを公開します。

XML 表現の例:

```

<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>

```



注記

この統計サブコレクションは読み取り専用です。

次のリストは、ホスト NUMA ノードの統計タイプを示しています。

Name	説明
memory.total	NUMA ノードの合計メモリー (バイト単位)。
memory.used	NUMA ノードで使用されるメモリー (バイト単位)。
memory.free	NUMA ノードの空きメモリー (バイト単位)。
cpu.current.user	ユーザースライス of CPU 使用率。
cpu.current.system	システムの CPU 使用率。
cpu.current.idle	アイドル状態の CPU 使用率。

7.270. VLAN 構造体

仮想 LAN (VLAN) タイプを表すタイプ。

表7.352 属性の概要

Name	タイプ	概要
id	Integer	仮想 LAN ID。

7.271. VM 構造体

仮想マシンを表します。

表7.353 属性の概要

Name	タイプ	概要
bios	Bios	仮想マシンの BIOS 設定への参照。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
console	コンソール	この仮想マシン用に設定されたコンソール。
cpu	Cpu	仮想マシン CPU の設定。
cpu_shares	Integer	
creation_time	日付	仮想マシンの作成日。
custom_compatibility_version	バージョン	仮想マシンのカスタム互換性バージョン。
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	さまざまなフックを設定するために VDSM に送信されるプロパティ。
delete_protected	ブール値	true の場合、仮想マシンは削除できません。
description	String	プレーンテキストでの人間が判読できる説明。
display	表示	仮想マシンの表示設定。
domain	Domain	この仮想マシン用に設定されたドメイン。
fqdn	String	仮想マシンの完全修飾ドメイン名。
guest_operating_system	GuestOperatingSystem	仮想マシンにインストールされているオペレーティングシステム。
guest_time_zone	TimeZone	仮想マシンが使用するタイムゾーン (ゲストエージェントによって返されます)。
high_availability	高可用性	仮想マシンの高可用性設定。
id	String	一意の ID

Name	タイプ	概要
initialization	初期化	仮想マシンの初期化設定への参照。
io	Io	IO スレッドのパフォーマンスチューニング用。
large_icon	アイコン	仮想マシンの大きなアイコン。
lease	StorageDomainLe ase	この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。
memory	Integer	仮想マシンのメモリー (バイト単位)。
memory_policy	MemoryPolicy	仮想マシンのメモリー管理設定への参照。
migration	MigrationOptions	実行中の仮想マシンの別のホストへの移行設定への参照。
migration_down time	Integer	別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。
name	String	人間が判読できるプレーンテキストでの名前。
next_run_config uration_exists	ブール値	仮想マシンの設定が変更されたため、仮想マシンの再起動が必要です。
numa_tune_mo de	NumaTuneMode	NUMA トポロジーの適用方法。
origin	String	この仮想マシンのオリジン。
os	OperatingSystem	仮想マシンにインストールされているオペレーティングシステムのタイプ。
payloads	Payload[]	仮想マシンのオプションのペイロード。ISO が仮想マシンを設定するために使用されます。
placement_poli cy	VmPlacementPolic y	仮想マシンの配置ポリシーの設定。
rng_device	RngDevice	この仮想マシンの乱数ジェネレーターデバイスの設定。
run_once	ブール値	true の場合、仮想マシンは run once コマンドを使用して開始されています。つまり、この1回の実行のために保存されている設定とは異なる可能性があります。
serial_number	SerialNumber	クラスター内の仮想マシンのシリアル番号。
small_icon	アイコン	仮想マシンの小さなアイコン。

Name	タイプ	概要
soundcard_enabled	ブール値	true の場合、サウンドカードが仮想マシンに追加されます。
sso	Sso	この仮想マシンが設定されているシングルサインオン設定への参照。
start_paused	ブール値	true の場合、仮想マシンは起動後、最初は 'paused' 状態になります。
start_time	日付	仮想マシンが起動された日付。
stateless	ブール値	true の場合、仮想マシンはステートレスで、シャットダウン後にその状態 (ディスク) がロールバックされます。
status	VmStatus	仮想マシンの現在の状態
status_detail	String	人間が読める現在のステータスの詳細。
stop_reason	String	仮想マシンが停止した理由。
stop_time	日付	仮想マシンが停止された日付。
time_zone	TimeZone	oVirt によって設定された仮想マシンのタイムゾーン。
tunnel_migration	ブール値	true の場合、ネットワークデータ転送は仮想マシンのライブマイグレーション中に暗号化されます。
type	VmType	仮想マシンがデスクトップとサーバーのどちらに最適化されているかを決定します。
usb	Usb	この仮想マシンの USB デバイスの設定 (カウント、タイプ)。
use_latest_template_version	ブール値	true の場合、仮想マシンは起動時にテンプレートの最新バージョンに再設定されます。
virtio_scsi	VirtioScsi	VirtIO SCSI 設定への参照。

7.271.1. cpu

仮想マシン CPU の設定。

ソケット設定は、仮想マシンを再起動せずに更新できます。コアとスレッドは再起動する必要があります。

たとえば、ソケットの数をすぐに 4 に変更し、再起動後にコアとスレッドの数を 2 に変更するには、以下のリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

7.271.2. custom_compatibility_version

仮想マシンのカスタム互換性バージョン。

仮想マシンを独自の互換性バージョンにカスタマイズできるようにします。**custom_compatibility_version** が設定されている場合、この特定の仮想マシンのクラスター互換性バージョンをオーバーライドします。

仮想マシンの互換バージョンは、仮想マシンが格納されているデータセンターによって制限され、仮想マシンが実行される予定のホストの機能に対してチェックされます。

7.271.3. high_availability

仮想マシンの高可用性設定。設定されている場合、仮想マシンが予期せずダウンしたときに自動的に再起動されます。

7.271.4. large_icon

仮想マシンの大きなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.271.5. lease

この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。

リースを使用して実行している仮想マシンは、この仮想マシンの別のインスタンスが別のホストで実行されるのを防ぐために、実行中にリースが別のホストによって取得されていないことを確認する必要があります。これにより、高可用性の仮想マシンでスプリットブレインが保護されます。このテンプレートから作成された仮想マシンを、このストレージドメインをリースの場所として事前設定するために、テンプレートにリース用に定義されたストレージドメインを含めることもできます。

7.271.6. memory

仮想マシンのメモリー (バイト単位)。

たとえば、1ギビバイト (GiB) のメモリーを含むように仮想マシンを更新するには、次の要求を送信します。

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

リクエスト本文は、以下のようになります。

```
<vm>
  <memory>1073741824</memory>
</vm>
```



注記

この例のメモリーは、次の式を使用してバイトに変換されます:
 $1 \text{ GiB} = 2^{30} \text{ バイト} = 1073741824 \text{ バイト}$ 。



注記

メモリーホットプラグは、Red Hat Virtualization 3.6 以降でサポートされています。上記の例では、仮想マシンの実行中にメモリーを増やすことができます。

7.271.7. migration_downtime

別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。

仮想マシンに対して明示的に設定するか、**engine-config -s DefaultMaximumMigrationDowntime=[value]** で設定します。

7.271.8. next_run_configuration_exists

仮想マシンの設定が変更されたため、仮想マシンの再起動が必要です。変更された設定は、仮想マシンのシャットダウン処理時に適用されます。

7.271.9. origin

この仮想マシンのオリジン。

値:

- **ovirt**
- **rhev**
- **vmware**
- **xen**
- **external**
- **hosted_engine**
- **managed_hosted_engine**
- **kvm**
- **physical_machine**
- **hyperv**

7.271.10. placement_policy

仮想マシンの配置ポリシーの設定。

この設定を更新して、仮想マシンを1つ以上のホストにピンングできます。



注記

複数のホストにピンングされた仮想マシンはライブマイグレーションできませんが、ホストに障害が発生した場合、高可用性になるように設定された仮想マシンは、仮想マシンがピンングされている他のホストの1つで自動的に再起動されます。

たとえば、仮想マシンを2つのホストに固定するには、以下のリクエストを送信します。

```
PUT /api/vms/123
```

リクエスト本文は以下のようになります。

```
<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

仮想マシンの小さなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.271.12. sso

この仮想マシンが設定されているシングルサインオン設定への参照。コンソールを開くと、ユーザーは仮想マシンのオペレーティングシステムに自動的にサインインできます。

7.271.13. stop_reason

仮想マシンが停止した理由。オプションで、仮想マシンをシャットダウンするときにユーザーが設定します。

表7.354 リンクの概要

Name	タイプ	概要
affinity_labels	AffinityLabel[]	オプション。
applications	Application[]	仮想マシンにインストールされているアプリケーションのリスト。
cdroms	Cdrom[]	CDROM にマウントされた ISO への参照。
cluster	Cluster	仮想マシンが属するクラスターへの参照。
cpu_profile	CpuProfile	この仮想マシンによって使用される CPU プロファイルへの参照。
disk_attachments	DiskAttachment[]	仮想マシンにアタッチされているディスクへの参照。
external_host_provider	ExternalHostProvider	
floppies	Floppy[]	フロッピーにマウントされた ISO への参照。
graphics_consoles	GraphicsConsole[]	この仮想マシン用に設定されたグラフィックコンソールのリスト。
host	ホスト	仮想マシンが実行されているホストへの参照。
host_devices	HostDevice[]	この仮想マシンに関連付けられているデバイスへの参照。
instance_type	InstanceType	仮想マシンの設定は、インスタンスタイプの1つを介してオプションで事前定義できます。
katello_errata	KatelloErratum[]	仮想マシンに割り当てられたすべての Katello エラータを一覧表示します。
nics	Nic[]	仮想マシン上のネットワークインターフェイスデバイスのリストを参照します。
numa_nodes	NumaNode[]	この仮想マシンで使用される NUMA ノード設定を参照します。
original_template	Template	仮想マシンの作成に使用された元のテンプレートを参照します。
permissions	Permission[]	この仮想マシンに設定されたパーミッション。
quota	クォータ	この仮想マシンに設定されたクォータ設定への参照。

Name	タイプ	概要
reported_devices	ReportedDevice[]	
sessions	Session[]	この仮想マシンに対して開かれているユーザーセッションのリスト。
snapshots	Snapshot[]	仮想マシンから取得されたすべてのスナップショットを参照します。
statistics	Statistic[]	この仮想マシンから収集された統計データ。
storage_domain	StorageDomain	仮想マシンが属するストレージドメインへの参照。
tags	Tag[]	
template	Template	仮想マシンがベースとするテンプレートへの参照。
vm_pool	VmPool	仮想マシンが任意のメンバーであるプールへの参照。
watchdogs	Watchdog[]	ウォッチドッグ設定への参照。

7.271.14. affinity_labels

オプション。サブクラスターのラベル付けに使用されます。

7.271.15. katello_errata

仮想マシンに割り当てられたすべての Katello エラータを一覧表示します。

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

以下のような XML で応答を受け取ります。

```
<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>
```

```

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

```

7.271.16. original_template

仮想マシンの作成に使用された元のテンプレートを参照します。

テンプレートまたは別の仮想マシンから仮想マシンをクローンした場合、**template** はブランクテンプレートにリンクし、履歴の追跡には **original_template** が使用されます。

それ以外の場合は、**template** と **original_template** は同じです。

7.272. VMAFFINITY ENUM

表7.355 値の概要

Name	概要
migratable	
pinned	
user_migratable	

7.273. VMBASE 構造体

基本的な仮想マシン設定を表します。これは、仮想マシン、テンプレート、およびインスタンスタイプで使用されます。

表7.356 属性の概要

Name	タイプ	概要
bios	Bios	仮想マシンの BIOS 設定への参照。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
console	コンソール	この仮想マシン用に設定されたコンソール。
cpu	Cpu	仮想マシン CPU の設定。
cpu_shares	Integer	
creation_time	日付	仮想マシンの作成日。
custom_compatibility_version	バージョン	仮想マシンのカスタム互換性バージョン。

Name	タイプ	概要
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	さまざまなフックを設定するために VDSM に送信されるプロパティ。
delete_protected	ブール値	true の場合、仮想マシンは削除できません。
description	String	プレーンテキストでの人間が判読できる説明。
display	表示	仮想マシンの表示設定。
domain	Domain	この仮想マシン用に設定されたドメイン。
high_availability	高可用性	仮想マシンの高可用性設定。
id	String	一意の ID
initialization	初期化	仮想マシンの初期化設定への参照。
io	io	IO スレッドのパフォーマンスチューニング用。
large_icon	アイコン	仮想マシンの大きなアイコン。
lease	StorageDomainLease	この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。
memory	Integer	仮想マシンのメモリー (バイト単位)。
memory_policy	MemoryPolicy	仮想マシンのメモリー管理設定への参照。
migration	MigrationOptions	実行中の仮想マシンの別のホストへの移行設定への参照。
migration_downtime	Integer	別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。
name	String	人間が判読できるプレーンテキストでの名前。
origin	String	この仮想マシンのオリジン。
os	OperatingSystem	仮想マシンにインストールされているオペレーティングシステムのタイプ。

Name	タイプ	概要
rng_device	RngDevice	この仮想マシンの乱数ジェネレーターデバイスの設定。
serial_number	SerialNumber	クラスター内の仮想マシンのシリアル番号。
small_icon	アイコン	仮想マシンの小さなアイコン。
soundcard_enabled	ブール値	true の場合、サウンドカードが仮想マシンに追加されます。
ss0	Sso	この仮想マシンが設定されているシングルサインオン設定への参照。
start_paused	ブール値	true の場合、仮想マシンは起動後、最初は 'paused' 状態になります。
stateless	ブール値	true の場合、仮想マシンはステートレスで、シャットダウン後にその状態 (ディスク) がロールバックされます。
time_zone	TimeZone	oVirt によって設定された仮想マシンのタイムゾーン。
tunnel_migration	ブール値	true の場合、ネットワークデータ転送は仮想マシンのライブマイグレーション中に暗号化されます。
type	VmType	仮想マシンがデスクトップとサーバーのどちらに最適化されているかを決定します。
usb	Usb	この仮想マシンの USB デバイスの設定 (カウント、タイプ)。
virtio_scsi	VirtioScsi	VirtIO SCSI 設定への参照。

7.273.1. cpu

仮想マシン CPU の設定。

ソケット設定は、仮想マシンを再起動せずに更新できます。コアとスレッドは再起動する必要があります。

たとえば、ソケットの数をすぐに 4 に変更し、再起動後にコアとスレッドの数を 2 に変更するには、以下のリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<vm>
  <cpu>
    <topology>
```

```

<sockets>4</sockets>
<cores>2</cores>
<threads>2</threads>
</topology>
</cpu>
</vm>

```

7.273.2. custom_compatibility_version

仮想マシンのカスタム互換性バージョン。

仮想マシンを独自の互換性バージョンにカスタマイズできるようにします。**custom_compatibility_version** が設定されている場合、この特定の仮想マシンのクラスター互換性バージョンをオーバーライドします。

仮想マシンの互換バージョンは、仮想マシンが格納されているデータセンターによって制限され、仮想マシンが実行される予定のホストの機能に対してチェックされます。

7.273.3. high_availability

仮想マシンの高可用性設定。設定されている場合、仮想マシンが予期せずダウンしたときに自動的に再起動されます。

7.273.4. large_icon

仮想マシンの大きなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.273.5. lease

この仮想マシン/テンプレートリースが存在するストレージドメインへの参照。

リースを使用して実行している仮想マシンは、この仮想マシンの別のインスタンスが別のホストで実行されるのを防ぐために、実行中にリースが別のホストによって取得されていないことを確認する必要があります。これにより、高可用性の仮想マシンでスプリットブレインが保護されます。このテンプレートから作成された仮想マシンを、このストレージドメインをリースの場所として事前設定するために、テンプレートにリース用に定義されたストレージドメインを含めることもできます。

7.273.6. memory

仮想マシンのメモリー (バイト単位)。

たとえば、1ギビバイト (GiB) のメモリーを含むように仮想マシンを更新するには、次の要求を送信します。

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

リクエスト本文は、以下のようになります。

```

<vm>
  <memory>1073741824</memory>
</vm>

```

**注記**

この例のメモリーは、次の式を使用してバイトに変換されます:
 $1 \text{ GiB} = 2^{30} \text{ バイト} = 1073741824 \text{ バイト}$ 。

**注記**

メモリーホットプラグは、Red Hat Virtualization 3.6 以降でサポートされています。上記の例では、仮想マシンの実行中にメモリーを増やすことができます。

7.273.7. migration_downtime

別のホストへのライブマイグレーション中に仮想マシンが応答しないでいられる最大時間 (ミリ秒)。

仮想マシンに対して明示的に設定するか、**engine-config -s DefaultMaximumMigrationDowntime=[value]** で設定します。

7.273.8. origin

この仮想マシンのオリジン。

値:

- **ovirt**
- **rhev**
- **vmware**
- **xen**
- **external**
- **hosted_engine**
- **managed_hosted_engine**
- **kvm**
- **physical_machine**
- **hyperv**

7.273.9. small_icon

仮想マシンの小さなアイコン。ユーザーが設定するか、オペレーティングシステムに従って設定されたイメージを参照します。

7.273.10. sso

この仮想マシンが設定されているシングルサインオン設定への参照。コンソールを開くと、ユーザーは仮想マシンのオペレーティングシステムに自動的にサインインできます。

表7.357 リンクの概要

Name	タイプ	概要
cluster	Cluster	仮想マシンが属するクラスターへの参照。
cpu_profile	CpuProfile	この仮想マシンによって使用される CPU プロファイルへの参照。
quota	クォータ	この仮想マシンに設定されたクォータ設定への参照。
storage_domain	StorageDomain	仮想マシンが属するストレージドメインへの参照。

7.274. VMDEVICETYPE ENUM

表7.358 値の概要

Name	概要
cdrom	
floppy	

7.275. VMPLACEMENTPOLICY 構造体

表7.359 属性の概要

Name	タイプ	概要
affinity	VmAffinity	

表7.360 リンクの概要

Name	タイプ	概要
hosts	Host[]	

7.276. VMPOOL 構造体

表7.361 属性の概要

Name	タイプ	概要
auto_storage_select	ブール値	テンプレートがコピーされる複数のストレージドメイン間でプールが仮想マシンのディスクを自動的に分散するかどうかを示します。

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
display	表示	
id	String	一意の ID
max_user_vms	Integer	
name	String	人間が判読できるプレーンテキストでの名前。
prestarted_vms	Integer	
rng_device	RngDevice	
size	Integer	
soundcard_enabled	ブール値	
stateful	ブール値	仮想マシンプールのステートフルフラグ。
type	VmPoolType	
use_latest_template_version	ブール値	

7.276.1. auto_storage_select

テンプレートがコピーされる複数のストレージドメイン間でプールが仮想マシンのディスクを自動的に分散するかどうかを示します。

プールで使用されるテンプレートが複数のストレージドメインに存在する場合、プールの仮想マシンのディスクはそれらのストレージドメインの1つに作成されます。デフォルトまたはこの属性の値が **false** の場合、そのストレージドメインはプール作成時に選択され、すべての仮想マシンが同じものを使用します。この属性が **true** の場合、仮想マシンがプールに追加されると、より多くの空き容量を持つストレージドメインが選択されます。

7.276.2. stateful

仮想マシンプールのステートフルフラグ。

ステートフル仮想マシンプールの仮想マシンは、常にステートフルモードで起動されます (ステートレススナップショットは作成されません)。仮想マシンが別のユーザーに渡された場合でも、仮想マシンの状態は保持されます。

表7.362 リンクの概要

Name	タイプ	概要
cluster	Cluster	
instance_type	InstanceType	このプールのベースとなるインスタンスタイプへの参照。
permissions	Permission[]	
template	Template	
vm	Vm	

7.276.3. instance_type

このプールのベースとなるインスタンスタイプへの参照。プールの作成時にのみ設定でき、編集はできません。

7.277. VMPOOLTYPE ENUM

表7.363 値の概要

Name	概要
automatic	
manual	

7.278. VMSTATUS ENUM

仮想マシンのステータスの rereseting と入力します。

表7.364 値の概要

Name	概要
down	このステータスは、仮想マシンプロセスが実行されていないことを示します。
image_locked	このステータスは、仮想マシンプロセスが実行されておらず、仮想マシンのディスクで起動を妨げる操作が行われていることを示します。
migrating	このステータスは、仮想マシンプロセスが実行中であり、仮想マシンがホスト間で移行中であることを示します。
not_responding	このステータスは、仮想マシンが応答していないことをハイパーバイザーが検出したことを示します。

Name	概要
paused	このステータスは、仮想マシンプロセスが実行中であり、仮想マシンが一時停止していることを示します。
powering_down	このステータスは、仮想マシンプロセスが実行中であり、実行を停止しようとしていることを示します。
powering_up	このステータスは、仮想マシンプロセスが実行中で、ゲストオペレーティングシステムがロードされていることを示します。
reboot_in_progress	このステータスは、仮想マシンプロセスが実行中で、ゲストオペレーティングシステムが再起動中であることを示します。
restoring_state	このステータスは、仮想マシンプロセスが実行されようとしており、仮想マシンが休止状態から復帰しようとしていることを示します。
saving_state	このステータスは、仮想マシンプロセスが実行中であり、仮想マシンが休止状態にあることを示します。
suspended	このステータスは、仮想マシンプロセスが実行されておらず、仮想マシンの実行状態が保存されたことを示します。
unassigned	このステータスは、無効なステータスを受信したときに設定されます。
unknown	このステータスは、システムが仮想マシンのステータスを特定できなかったことを示します。
up	このステータスは、仮想マシンプロセスが実行中で、ゲストオペレーティングシステムがロードされていることを示します。
wait_for_launch	このステータスは、仮想マシンプロセスが実行されようとしていることを示します。

7.278.1. paused

このステータスは、仮想マシンプロセスが実行中であり、仮想マシンが一時停止していることを示します。これは、仮想マシンが一時停止モードで実行されている場合と、仮想マシンがエラーが原因で自動的に一時停止されている場合の2つのケースで発生する可能性があります。

7.278.2. powering_up

このステータスは、仮想マシンプロセスが実行中で、ゲストオペレーティングシステムがロードされていることを示します。ゲストエージェントがインストールされていない場合、このステータスは、仮想マシンの実行時に事前定義された期間（デフォルトでは 60 秒）に設定されることに注意してください。

7.278.3. restoring_state

このステータスは、仮想マシンプロセスが実行されようとしており、仮想マシンが休止状態から復帰しようとしていることを示します。この状態では、仮想マシンの実行状態が復元されています。

7.278.4. saving_state

このステータスは、仮想マシンプロセスが実行中であり、仮想マシンが休止状態にあることを示します。この状態では、仮想マシンの実行状態が保存されています。このステータスは、ゲスト OS が休止状態であることを意味するものではないことに注意してください。

7.278.5. suspended

このステータスは、仮想マシンプロセスが実行されておらず、仮想マシンの実行状態が保存されたことを示します。このステータスは Down に似ていますが、VM がこのステータスで起動されると、通常の手順を使用して起動される代わりに、保存された実行状態が復元されます。

7.278.6. unknown

このステータスは、システムが仮想マシンのステータスを特定できなかったことを示します。このステータスでは、仮想マシンプロセスが実行されているか、実行されていない可能性があります。たとえば、ホストが応答しなくなると、ホストで実行されていた仮想マシンにこのステータスが設定されます。

7.278.7. up

このステータスは、仮想マシンプロセスが実行中で、ゲストオペレーティングシステムがロードされていることを示します。ゲストエージェントがインストールされていない場合、このステータスは、仮想マシンの実行時に事前に定義された期間 (デフォルトでは 60 秒) 後に設定されることに注意してください。

7.278.8. wait_for_launch

このステータスは、仮想マシンプロセスが実行されようとしていることを示します。このステータスは、仮想マシンを実行する要求がホストに到着したときに設定されます。仮想マシンプロセスの実行に失敗する可能性があります。

7.279. VMSUMMARY 構造体

表7.365 属性の概要

Name	タイプ	概要
active	Integer	
migrating	Integer	
total	Integer	

7.280. VMATYPE ENUM

仮想マシンの最適化対象を表す型。

表7.366 値の概要

Name	概要
desktop	仮想マシンは、デスクトップとして使用することを目的としています。
server	仮想マシンは、サーバーとして使用することを目的としています。

7.280.1. desktop

仮想マシンは、デスクトップとして使用することを目的としています。現在、サウンドデバイスが仮想マシンに自動的に追加されることを意味します。

7.280.2. server

仮想マシンは、サーバーとして使用することを目的としています。現時点では、サウンドデバイスが仮想マシンに自動的に追加されないことを意味します。

7.281. VNICPASSTHROUGH 構造体

表7.367 属性の概要

Name	タイプ	概要
mode	VnicPassThrough Mode	vNIC を仮想デバイスとして実装するか、ホストデバイスへのパススルーとして実装するかを定義します。

7.282. VNICPASSTHROUGHMODE ENUM

vNIC をパススルーデバイスとして実装するか、仮想デバイスとして実装するかを記述します。

表7.368 値の概要

Name	概要
disabled	仮想デバイスとして実装されます。
enabled	パススルーデバイスとして実装されます。

7.283. VNICPROFILE 構造体

vNIC プロファイルは、個々の NIC に適用できる設定の集まりです。

表7.369 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。

Name	タイプ	概要
custom_properties	CustomProperty[]	vNIC プロファイルに適用されるカスタムプロパティ。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
migratable	ブール値	pass_through NIC が移行可能かどうか。
name	String	人間が判読できるプレーンテキストでの名前。
pass_through	VnicPassThrough	SR-IOV 対応の ホスト NIC へのパススルーを有効にします。
port_mirroring	ブール値	ポートミラーリングを有効にします。

7.283.1. migratable

pass_through NIC が移行可能かどうか。

pass_through.mode が **無効** に設定されている場合、このオプションは意味がなく、**true** と見なされます。要求からこのオプションを省略すると、デフォルトで **true** に設定されます。

仮想マシンの移行時に、この仮想マシンは、すべての **pass_through** NIC に **migratable** としてフラグが付けられている場合にのみ移行されます。

7.283.2. pass_through

SR-IOV 対応の **ホスト NIC** へのパススルーを有効にします。

パススルーが有効な場合、vNIC プロファイルを使用すると、SR-IOV が有効なホスト NIC の **virtual function (VF)** に NIC を直接接続できます。次に、NIC はソフトウェアによるネットワーク仮想化をバイパスして、VF に直接接続してデバイスを割り当てます。

vNIC プロファイルがすでに NIC に接続されている場合は、パススルーを有効にすることはできません。vNIC プロファイルでパススルーが有効になっている場合、vNIC プロファイルの **qos** と **port_mirroring** は無効になります。

7.283.3. port_mirroring

ポートミラーリングを有効にします。

ポートミラーリングは、特定の **論理ネットワークおよびホスト上のレイヤー3 ネットワーク** トラフィックを **仮想マシン** の NIC にコピーします。この仮想マシンは、ネットワークのデバッグとチューニング、侵入検知、および同じホストと論理ネットワーク上の他の仮想マシンの動作の監視に使用できます。コピーされる唯一のトラフィックは、1つのホスト上の1つの論理ネットワークの内部です。ホスト外のネットワークのトラフィックが増えることはありませんが、ポートミラーリングが有効になっている仮想マシンは、他の仮想マシンよりも多くのホスト CPU および RAM を使用します。

ポートミラーリングには次の制限があります。

- ポートミラーリングが有効になっている vNIC プロファイルを持つ NIC のホットプラグはサポートされていません。
- vNIC プロファイルが仮想マシンに接続されている場合は、ポートミラーリングを変更することができません。

上記の制限があるため、追加の専用 vNIC プロファイルでポートミラーリングを有効にすることが推奨されます。



重要

ポートミラーリングを有効にすると、他のネットワークユーザーのプライバシーが低下します。

表7.370 リンクの概要

Name	タイプ	概要
network	Network	vNIC プロファイルが適用されるネットワークへの参照。
network_filter	NetworkFilter	このプロファイルを使用する NIC に適用される最上位のネットワークフィルターへの参照。
permissions	Permission[]	vNIC プロファイルの使用を許可する権限。
qos	Qos	vNIC プロファイルに適用するサービス品質属性への参照。

7.283.4. network_filter

このプロファイルを使用する NIC に適用される最上位のネットワークフィルターへの参照。

ネットワークフィルターは、参加している仮想マシンとの間でネットワークパケットトラフィックを管理する管理者機能を強化します。ネットワークフィルターには、他のフィルターへの参照、トラフィックフィルターリングのルール、または両方の組み合わせを含めることができます。

7.283.5. qos

vNIC プロファイルに適用するサービス品質属性への参照。

Quality of Service 属性は、NIC のインバウンドおよびアウトバウンドのネットワークトラフィックを規制します。

7.284. VNICPROFILEMAPPING 構造体

外部仮想 NIC プロファイルを Red Hat Virtualization Manager に存在するものにマップします。

たとえば、目的の仮想 NIC プロファイルのマッピングに次の 2 行が含まれているとします。

ソースネットワーク名	ソースネットワークプロファイル名	ターゲット仮想 NIC プロファイル ID
red	gold	738dd914-8ec8-4a8b-8628-34672a5d449b
blue	silver	892a12ec-2028-4451-80aa-ff3bf55d6bac

以下の形式で表記する必要があります。

```
<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>
```

表7.371 属性の概要

Name	タイプ	概要
source_network_name	String	外部ネットワークの名前を指定します。
source_network_profile_name	String	外部ネットワークプロファイルの名前を指定します。

表7.372 リンクの概要

Name	タイプ	概要
target_vnic_profile	VnicProfile	既存の仮想 NIC プロファイルを参照します。

7.285. VOLUMEGROUP STRUCT

表7.373 属性の概要

Name	タイプ	概要
id	String	

Name	タイプ	概要
logical_units	LogicalUnit[]	
name	String	

7.286. WATCHDOG 構造体

このタイプは、ウォッチドッグ設定を表します。

表7.374 属性の概要

Name	タイプ	概要
action	WatchdogAction	ウォッチドッグがトリガーされたときに実行されるウォッチドッグアクション。
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
id	String	一意の ID
model	WatchdogModel	ウォッチドッグデバイスのモデル。
name	String	人間が判読できるプレーンテキストでの名前。

7.286.1. model

ウォッチドッグデバイスのモデル。現在、I6300ESB のみがサポートされています。

表7.375 リンクの概要

Name	タイプ	概要
instance_type	InstanceType	オプションで、デバイスが使用するインスタンスタイプへの参照。
template	Template	オプションで、デバイスが使用するテンプレートへの参照。
vm	Vm	この要素は使用せず、代わりに vms を使用してください。
vms	Vm[]	このデバイスを使用する仮想マシンへの参照。

7.286.2. vms

このデバイスを使用する仮想マシンへの参照。デバイスは複数の仮想マシンで使用できます。たとえば、共有ディスクは2つ以上の仮想マシンで同時に使用することができます。

7.287. WATCHDOGACTION ENUM

このタイプは、利用可能なウォッチドッグアクションを記述します。

表7.376 値の概要

Name	概要
dump	仮想マシンプロセスは、ホストのデフォルトパスにコアダンプされます。
none	ウォッチドッグアクションがトリガーされる場合、アクションは実行されません。
pause	ウォッチドッグアクションがトリガーされると、仮想マシンは一時停止します。
poweroff	ウォッチドッグアクションがトリガーされると、仮想マシンの電源がオフになります。
reset	ウォッチドッグアクションがトリガーされると、仮想マシンが再起動されます。

7.287.1. none

ウォッチドッグアクションがトリガーされる場合、アクションは実行されません。ただし、ログメッセージは引き続き生成されます。

7.288. WATCHDOGMODEL ENUM

このタイプはウォッチドッグモデルを表します。

表7.377 値の概要

Name	概要
i6300esb	現在サポートされるモデルのみがモデル I6300ESB です。

7.289. WEIGHT 構造体

表7.378 属性の概要

Name	タイプ	概要
comment	String	このオブジェクトに関するコメントを含むフリーテキスト。
description	String	プレーンテキストでの人間が判読できる説明。
factor	Integer	

Name	タイプ	概要
id	String	一意の ID
name	String	人間が判読できるプレーンテキストでの名前。

表7.379 リンクの概要

Name	タイプ	概要
scheduling_policy	SchedulingPolicy	
scheduling_policy_unit	SchedulingPolicyUnit	

付録A プリミティブ型

このセクションでは、API でサポートされているプリミティブデータ型について説明します。

A.1. 文字列 プリミティブ

Unicode 文字の有限シーケンス。

A.2. ブール値 プリミティブ

数理論理学で使用される `false` の概念と `true` の概念を表します。

有効な値は文字列 `false` と `true` です。

エンジンは大文字と小文字を区別しないため、たとえば `False` と `FALSE` はどちらも有効な値です。ただし、サーバーは常に小文字の値を返します。

古いバージョンのエンジンとの後方互換性のために、値 `0` と `1` も使用できます。値 `0` は `false` を、`1` は `true` を意味します。これらの値のサポートは今後削除される可能性があるため、これらの値の使用は避けてください。

A.3. 整数 プリミティブ

整数の数学的概念を表します。

有効な値は、10 進数の有限シーケンスです。

現在、エンジンはこの型を符号付き 32 ビット整数で実装しており、最小値は 2^{31} (-2147483648) で、最大値は $2^{31}-1$ (2147483647) となっています。

ただし、システムには、32 ビットで可能な値の範囲では不十分な属性がいくつかあります。これらの例外的なケースでは、エンジンは、特に次の属性に対して 64 ビット整数を使用します。

- `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**

これらの例外の場合、最小値は -2^{63} (-9223372036854775808) で、最大値は $2^{63}-1$ (9223372036854775807) です。



注記

今後、整数型は無制限の精度の整数を使用して実装されるため、上記の制限と例外は最終的にはなくなります。

A.4. 10 進プリミティブ型

実数の数学的概念を表します。

現在、エンジンは 32 ビット [IEEE 754](#) シングル精度浮動小数点数を使用してこのタイプを実装します。

一部の属性では、この精度では十分ではありません。これらの例外的なケースでは、エンジンは 64 ビットの倍精度浮動小数点数を、特に次の属性に使用します。

- **QuotaStorageLimit.usage**
- **QuotaStorageLimit.memory_limit**
- **QuotaStorageLimit.memory_usage**



注記

今後、10 進数型は無制限の精度の 10 進数を使用して実装されるため、上記の制限と例外は最終的にはなくなります。

A.5. DATE プリミティブ

日付と時刻を表します。

エンジンによって返される形式は、XML を要求するときに [XML スキーマ仕様](#) に記述されている形式です。たとえば、次のようなリクエストを送信して、仮想マシンの XML 表現を取得するとします。

```
GET /ovirt-engine/api/vms/123
Accept: application/xml
```

応答本文には、次の XML ドキュメントが含まれます。

```
<vm id="123" href="/ovirt-engine/api/vms/123">
...
<creation_time>2016-09-08T09:53:35.138+02:00</creation_time>
...
</vm>
```

JSON 表現を要求する場合、エンジンは別の形式を使用します。1970 年 1^月 1 日からの秒数を含む整数。これは **エポック時間** としても知られています。たとえば、次のようなリクエストを送信して、仮想マシンの JSON 表現を取得するとします。

```
GET /ovirt-engine/api/vms/123
Accept: application/json
```

応答本文には、次の JSON ドキュメントが含まれます。

```
{
  "id": "123",
  "href="/ovirt-engine/api/vms/123",
  ...
  "creation_time": 1472564909990,
  ...
}
```



注記

いずれの場合も、エンジンによって返される日付は、実行中のサーバーで設定されたタイムゾーンを使用します。上記の例では UTC+2 になります。

付録B API バージョン 4 の変更点

本セクションでは、API のバージョン 4 で導入された後方互換性の重大な変更を説明します。

B.1. YAML サポートの削除

YAML のサポートは完全に削除されました。

B.2. 複合型の名前変更

次の XML スキーマ複合型の名前が変更されました。

Version 3	Version 4
API	Api
CPU	Cpu
CPU の数	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
NIC	Nic

Version 3	Version 4
PreviewVMs	PreviewVms
QoS	Qos
QoSs	Qoss
RSDL	RsdI
SELinux	SeLinux
SPM	Spm
SSHPublicKey	SshPublicKey
SSHPublicKeys	SshPublicKeys
SSH	Ssh
SkipIfSDActive	SkipIfSdActive
Slaves	HostNics
ストレージ	HostStorage
SupportedVersions	バージョン
VCpuPin	VcpuPin
VLAN	Vlan
VM	Vm
仮想マシン	Vms
VirtIO_SCSI	VirtioScsi
WatchDog	Watchdog
WatchDogs	Watchdogs

B.3. STATUS タイプの ENUM 型への置き換え

現在、さまざまなオブジェクトのステータスは、ステータスを説明する **state** 文字列と追加の詳細を示す別の **detail** 文字列を含む **Status** タイプを使用して報告されます。たとえば、IO エラーが原因で一時停止している仮想マシンのステータスは、現在次のように報告されています。

```
<vm>
...
<status>
  <state>paused</state>
  <detail>eio</detail>
</status>
...
</vm>
```

API のバージョン 4 では、この **Status** 型が削除され、enum 型に置き換えられました。追加の **詳細** 文字列が必要な場合は、追加の **status_detail** 属性に置き換えられています。たとえば、同じ仮想マシンのステータスは次のように報告されます。

```
<vm>
...
<status>paused</status>
<status_detail>eio</status_detail>
...
</vm>
```

B.4. NIC NETWORK と PORT_MIRRORING プロパティの削除

NIC **network** および **port_mirroring** 要素は **vnic_profile** 要素に置き換えられたので、ネットワークおよびポートミラーリング設定を指定する代わりに NIC を作成または更新する場合には、これらは vNIC プロファイルの作成時に指定されていました。

```
POST /ovirt-engine/api/vnicprofiles
```

```
<vnic_profile>
  <name>myprofile</name>
  <network id="..."/>
  <port_mirroring>true</port_mirroring>
</vnic_profile>
```

次に、NIC が作成されるか、既存の vNIC プロファイルを参照します。

```
PUT /ovirt-engine/api/vms/123/nics/456
```

```
<nic>
  <vnic_profile id="/vnicprofiles/..."/>
</nic>
```

古い要素とその意味は後方互換性のために保持されていましたが、現在は完全に削除されています。

network 要素は **initialization** 要素でまだ使用されているため、XML スキーマから削除されていませんが、NIC の作成または更新時に指定された場合は完全に無視されることに注意してください。

B.5. NIC ACTIVE プロパティの削除

NIC の **active** プロパティは、しばらく前に **plugged** に置き換えられました。現在は完全に削除されています。

B.6. ディスク TYPE プロパティの削除

ディスクの **type** プロパティは削除されましたが、XML スキーマに保持され、無視されます。現在は完全に削除されています。

B.7. ディスク SIZE プロパティの削除

ディスク **size** プロパティは、だいぶ前に **provisioned_size** に置き換えられています。現在は完全に削除されています。

B.8. VM を単一のホストに固定するためのサポートを削除

バージョン 3.6 より前の API では、VM エンティティの **placement_policy** 要素を使用して、VM を単一のホストにピンングすることができました。

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

```
<vm>
  <placement_policy>
    <host id="456"/>
  </placement_policy>
</vm>
```

バージョン 3.6 では、複数のホストをサポートするようにこの機能が強化され、新しい **hosts** 要素が追加されました。

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

```
<vm>
  <placement_policy>
    <hosts>
      <host id="456"/>
      <host id="789"/>
      ...
    </hosts>
  </placement_policy>
</vm>
```

後方互換性を維持するために、単一の **host** 要素が保持されました。4.0 ではこれが削除されたため、単一のホストにピンングする場合でも、アプリケーションは **hosts** 要素を使用する必要があります。

B.9. CAPABILITIES.PERMITS 要素の削除

permits のリストは、クラスターレベルごとに異なる可能性があり、かなり前に **version** 要素に追加されましたが、後方互換性のために、**capabilities** 要素にも保持されています。

4.0 では、**capabilities** サービスが完全に削除され、新しい **clusterlevels** サービスに置き換えられました。クラスターレベル 4.0 でサポートされる permits をを見つけるには、以下のようなリクエストを使用する必要があります。

GET /ovirt-engine/api/clusterlevels/4.0

結果は、そのクラスターレベルに固有の情報、特にサポートされている一連の `permits` を含むドキュメントになります。

```

<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. STORAGE_MANAGER 要素の削除

`storage_manager` 要素は、しばらく前に `spm` 要素に置き換えられました。古いものは後方互換性のために保持されていましたが、現在は完全に削除されています。

B.11. データセンターの STORAGE_TYPE 要素の削除

データセンターは、以前は特定のストレージタイプ (NFS、ファイバーチャネル、iSCSI など) に関連付けられていましたが、しばらくして変更され、ローカルストレージと共有ストレージの2つのタイプのみになりました。これを示すために新しい `local` 要素が導入され、古い `storage_type` 要素は後方互換性のために保持されました。この古い要素は完全に削除されました。

B.12. TIMEZONE 要素の削除

タイムゾーンを表す `timezone` 要素を格納するために使用される VM リソース。この要素は文字列のみを許可していました。

```

<vm>
  <timezone>Europe/Madrid</timezone>
</vm>

```

これはエクステンションを許可せず、UTC オフセットを追加する必要があったため、新しい構造化された `time_zone` 要素に置き換えられました。

```

<vm>
  <time_zone>
    <name>Europe/Madrid</name>
    <utc_offset>GMT+1</utc_offset>
  </time_zone>
</vm>

```

古い `timezone` 要素は保持されていましたが、現在は完全に削除されています。

B.13. GUEST_INFO 要素の削除

guest_info 要素は、IP アドレスや完全修飾ホスト名など、ゲストエージェントによって収集された情報を保持するために使用されていました。この情報は他の場所でも入手できます。たとえば、IP アドレスは VM リソース内で使用できます。

```
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>
```

また、NIC リソース内で、新しい **reported_devices** 要素を使用します。

```
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>
```

さらに、この新しい **reported_devices** 要素は、複数の IP アドレス、MAC アドレスなど、より完全な情報を提供します。

この重複を取り除くために、**guest_info** 要素が削除されました。

完全修飾ドメイン名をサポートするために、新しい **fqdn** 要素が VM リソースに追加されました。

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

```
<vm>
  <fqdn>myvm.example.com</fqdn>
</vms>
```

これには、**guest_info.fqdn** に含まれていたものと同じ情報が含まれます。

B.14. CPU ID 属性の TYPE 要素への置き換え

cpu 要素には、CPU のタイプを示す **id** 属性がありました。

```
<cpu id="Intel Conroe Family">
```

```
<architecture>X86_64</architecture>
...
</cpu>
```

これは、**id** 属性が不透明な識別子に使用される API モデルの残りの要素と矛盾しています。この **id** 属性は新しい **type** 要素に置き換えられました。

```
<cpu>
  <type>Intel Conroe Family</type>
  <architecture>X86_64</architecture>
</cpu>
```

B.15. CPU トポロジーで属性の代わりに要素を使用する

以前は、CPU トポロジー要素はそのプロパティに属性を使用していました。

```
<cpu>
  <topology sockets="1" cores="1" threads="1"/>
  ...
</cpu>
```

これは、API の一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<cpu>
  <topology>
    <sockets>1</sockets>
    <cores>1</cores>
    <threads>1</threads>
  </topology>
  ...
</cpu>
```

B.16. VCPU ピンで属性の代わりに要素を使用する

以前は、VCPU ピン要素はそのプロパティに属性を使用していました。

```
<cpu_tune>
  <vcpu_pin vcpu="0" cpu_set="0"/>
</cpu_tune>
```

これは、API の一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<cpu_tune>
  <vcpu_pin>
    <vcpu>0</vcpu>
    <cpu_set>0</cpu_set>
  </vcpu_pin>
</cpu_tune>
```

B.17. VCPU ピンで属性の代わりに要素を使用する

以前は、**version** 要素はそのプロパティに属性を使用していました:

```
<version major="3" minor="5" ..>
```

これは、API の一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<version>
  <major>3</minor>
  <minor>5</minor>
  ...
</version>
```

B.18. メモリーのオーバーコミットで属性の代わりに要素を使用する

以前は、**overcommit** 要素はそのプロパティに属性を使用していました。

```
<memory_policy>
  <overcommit percent="100"/>
  ...
</memory_policy>
```

これは、API の一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<memory_policy>
  <overcommit>
    <percent>100</percent>
  </overcommit>
  ...
</memory_policy>
```

B.19. CONSOLE で属性の代わりに要素を使用する

以前は、**console** 要素はそのプロパティに属性を使用していました:

```
<console enabled="true"/>
```

これは、API の一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<console>
  <enabled>true</enabled>
</console>
```

B.20. VIRTIO SCSI で属性の代わりに要素を使用する

以前は、VIRTIO ISCSI 要素はそのプロパティに属性を使用していました。

```
<virtio_scsi enabled="true"/>
```

これは、API の一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<virtio_scsi>
  <enabled>true</enabled>
</virtio_scsi>
```

B.21. 電源管理エージェント TYPE に属性ではなく要素を使用する

電源管理 **type** プロパティは属性として表されていました。

```
<agent type="apc">
  <username>myuser</username>
  ...
</agent>
```

これは、API の一般的な慣行に反しています。内部要素に置き換えられました。

```
<agent>
  <type>apc</type>
  <username>myuser</username>
  ...
</agent>
```

B.22. 電源管理エージェントオプションで属性の代わりに要素を使用する

以前は、電源管理エージェントの option 要素は、そのプロパティに属性を使用していました。

```
<options>
  <option name="port" value="22"/>
  <option name="slot" value="5"/>
  ...
</options>
```

これは、API の一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<options>
  <option>
    <name>port</name>
    <value>22</value>
  </option>
  <option>
    <name>slot</name>
    <value>5</value>
  </option>
  ...
</options>
```

B.23. IP アドレスで属性の代わりに要素を使用する

以前は、IP アドレス要素はそのプロパティに属性を使用していました。

```
<ip address="192.168.122.1" netmask="255.255.255.0"/>
```


これは、APIの一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<ip>
  <address>192.168.122.1</address>
  <netmask>255.255.255.0</netmask>
</ip>
```

B.24. MAC アドレスで属性の代わりに要素を使用する

以前は、MAC アドレス要素はそのプロパティに属性を使用していました。

```
<mac address="66:f2:c5:5f:bb:8d"/>
```

これは、APIの一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<mac>
  <address>66:f2:c5:5f:bb:8d</address>
</mac>
```

B.25. ブートデバイスで属性の代わりに要素を使用する

以前は、ブートデバイス要素はそのプロパティに属性を使用していました。

```
<boot dev="cdrom"/>
```

これは、APIの一般的な慣行に反しています。それらは内部要素に置き換えられました:

```
<boot>
  <dev>cdrom</dev>
</boot>
```

B.26. オペレーティングシステム TYPE に属性の代わりに要素を使用する

オペレーティングシステム **type** プロパティは属性として表されていました。

```
<os type="other">
  ...
</os>
```

これは、APIの一般的な慣行に反しています。内部要素に置き換えられました。

```
<os>
  <type>other</type>
  ...
</os>
```

B.27. ホストを取得するリクエストからの FORCE パラメーターの削除

データベースからデータを取得する前に、ホストのデータを更新する (VDSM を呼び出してホストの機能とデバイスをリロードする) 必要があることを示す **force** マトリクスパラメーターをサポートするために使用されるホストを取得するリクエスト。

```
GET /ovirt-engine/api/hosts/123;force
```

この **force** パラメーターは、ホストの **refresh** アクションに取って代わられましたが、後方互換性のために保持されています。現在は完全に削除されています。この機能を必要とするアプリケーションは、2つのリクエストを実行する必要があります。1つ目は、ホストをリフレッシュするためのリクエストです。

```
POST /ovirt-engine/api/hosts/123/refresh
```

```
<action/>
```

2つ目は、**force** パラメーターなしでそれを取得するためのリクエストです。

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

B.28. 非推奨のホスト電源管理設定の削除

ホストの電源管理設定は、以前はホストリソースの一部であり、組み込みの設定要素を使用していました。

```
<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>
```

これは、複数の電源管理エージェントをサポートするために、少し前に新しい **/hosts/123/fenceagents** コレクションを導入することで変更されました。

古い **type** 属性、古い **address** 要素、**username** 要素、**password** 要素、および **power_management** の内部に直接ある **agents** 要素は、後方互換性のために保持されました。これらの要素はすべて完全に削除されたため、電源管理エージェントをクエリーまたは変更する方法は、現時点では **/hosts/123/fenceagents** サブコレクションのみになります。

B.29. 複数の BOOT の代わりに複数の BOOT.DEVICES.DEVICE を使用する

これまで、仮想マシンの起動時にブートシーケンスを指定する方法は、それぞれが **dev** 要素を含む複数の **boot** 要素を使用することでした。たとえば、仮想マシンが最初に CDROM から起動し、次にハードディスクから起動するように指定するには、以下のリクエストが使用されました。

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

```
<action>
```

```

<vm>
...
<boot>
  <dev>cdrom</dev>
</boot>
<boot>
  <dev>hd</dev>
</boot>
</vm>
</action>

```

APIの他の部分での一般的な方法は、配列をラッパー要素で表すことです。その場合、そのラッパー要素に **boots** という名前を付けることができますが、ここで複数の値を指定できるのはブートシーケンスではなくブートデバイスであるため、あまり意味がありません。この矛盾を修正するために、これは複数のデバイスを含むことができる単一の **boot** 要素に置き換えられました。

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

```

<action>
<vm>
...
<boot>
  <devices>
    <device>cdrom</device>
    <device>hd</device>
  </devices>
</boot>
</vm>
</action>

```

B.30. DISKS.CLONE と DISKS.DETACH_ONLY 要素の削除

これらの要素は実際にはディスクの表現の一部ではなく、仮想マシンを追加および削除する操作のパラメーターです。

新しい仮想マシンのディスクのクローンを作成する必要があることを示すために、**disks.clone** 要素が使用されました。

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

```

<vm>
...
<disks>
  <clone>true</clone>
</disks>
</vm>

```

これは現在削除され、新しい **clone** クエリーパラメーターに置き換えられています。

```
POST /ovirt-engine/api/vms?clone=true
```

```
<vm>
...
</vm>
```

disks.detach_only 要素は、仮想マシンを削除するときにディスクを削除する必要はありませんが、仮想マシンからデタッチされるだけです。

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

```
<action>
  <vm>
    <disks>
      <detach_only>true</detach_only>
    </disks>
  </vm>
</action>
```

これは削除され、新しい **detach_only** クエリーパラメーターに置き換えられました。

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

B.31. 要素 **VMPOOL** の名前を **VM_POOL** に変更する

仮想マシンのプールを表す要素の名前は、以前は **vmpool** および **vm_pools** でした。複合タイプ(この場合は **VmPool** と **VmPools**)と要素の名前間での対応を統一するために、**vm_pool** と **vm_pools** に改名されました。

B.32. 複数の **LOGICAL_UNIT** の代わりに **LOGICAL_UNITS** を使用する

ボリュームグループの一部である論理ユニットは、無制限数の **logical_unit** 要素として報告されていました。たとえば、ストレージドメインの詳細を報告する場合は、以下ようになります。

```
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>
```

要素のリストは常に要素でラップされるため、これは API の通常の慣行に反します。これは現在修正されているため、論理ユニットのリストは **logical_units** 要素でラップされます。

```
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. SNAPSHOTS.COLLAPSE_SNAPSHOTS 要素の削除

この要素は実際にはスナップショットの表現の一部ではありませんが、エクスポートストレージドメインから仮想マシンをインポートする操作のパラメーターです。

```
POST /ovirt-engine/api/storagedomains/123/vms/456/import
```

```
<action>
  <vm>
    <snapshots>
      <collapse_snapshots>true</collapse_snapshots>
    </snapshots>
  </vm>
</action>
```

これは削除され、新しい **collapse_snapshots** クエリーパラメーターに置き換えられました。

```
POST /ovirt-engine/api/storagedomains/123/vms/456/import?collapse_snapshots=true
```

```
<action/>
```

B.34. STORAGE と HOST_STORAGE 要素の名前の変更

ホストストレージコレクションは、**storage** 要素と **host_storage** 要素、および **Storage** と **HostStorage** 複合型を使用して、ホストに関連付けられたストレージを報告します。

```
GET /ovirt-engine/api/hosts/123/storage
```

```
<host_storage>
  <storage>
  ...
```

```

</storage>
<storage>
...
</storage>
...
</host_storage>

```

これは、API の残りの部分で使用されるパターンに従っていません。外側の要素は複数形で、内側の要素は同じ名前ですが単数形です。これは、外側の要素として **host_storages** を、内側の要素として **host_storage** を使用するように変更されました。

```
GET /ovirt-engine/api/hosts/123/storage
```

```

<host_storages>
  <host_storage>
    ...
  </host_storage>
  <host_storage>
    ...
  </host_storage>
  ...
</host_storage>

```

B.35. PERMISSIONS.CLONE 要素の削除

この要素は実際にはアクセス許可の表現の一部ではありませんが、仮想マシンまたはテンプレートを作成する操作のパラメーターです。

```
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>

```

これは削除され、新しい **clone_permissions** クエリーパラメーターに置き換えられました。

```
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. 乱数ジェネレーター **SOURCE** 要素の名前の変更

乱数ジェネレーターのソースは、その使用を反映した名前の要素でラップされた **source** 要素のコレクションを使用して報告されていました。たとえば、従来は、クラスターの必要な乱数ジェネレーターソースは以下のように報告されていました。

GET /ovirt-engine/api/clusters/123

```
<cluster>
  ...
  <required_rng_sources>
    <source>random</source>
  </required_rng_sources>
  ...
</cluster>
```

また、ホストによってサポートされる乱数ジェネレーターソースは、以下のように報告されていました。

GET /ovirt-engine/api/hosts/123

```
<host>
  ...
  <hardware_information>
    <supported_rng_sources>
      <source>random</source>
    </supported_rng_sources>
  </hardware_information>
  ...
</host>
```

これは、コレクションが複数形で名前でもラップされ、要素が単数形で同じ名前でもラップされる他の API と一貫していません。これは修正されました。必要な乱数ジェネレーターソースは、以下のように報告されるようになりました。

GET /ovirt-engine/api/clusters/123

```
<cluster>
  <required_rng_sources>
    <required_rng_source>random</required_rng_source>
```

```

</required_rng_sources>
...
</cluster>

```

また、ホストでサポートされる乱数ジェネレーターソースは、以下のように報告されます。

```
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>

```

source だけでなく、**required_rng_source** と **supported_rng_source** を使用していることに注意してください。

B.37. 中間の TAG.PARENT 要素の削除

タグとその親タグの関係は、中間の **parent** タグを使用して表されていましたが、これには別の **tag** 要素が含まれていました。

```

<tag>
  <name>mytag</name>
  <parent>
    <tag id="..." href="..."/>
  </parent>
</tag>

```

この構造体は単純化され、1つの **parent** 要素のみが使用されるようになりました。

```

<tag>
  <name>mytag</name>
  <parent id="..." href="..."/>
</tag>

```

B.38. スケジューリングの組み込み名としきい値の削除

これまで、クラスターのスケジューリングポリシーの仕様は、組み込み名としきい値に基づいていました。たとえば、**均等に分散**されたスケジューリングポリシーを使用するクラスターは、以下のように表されていました。

```

<cluster>
  <name>mycluster</name>
  <scheduling_policy>
    <policy>evenly_distributed</policy>
    <thresholds high="80" duration="120"/>

```



```

</scheduling_policy>
...
</cluster>

```

このメカニズムは、スケジューリングポリシーを任意の名前とプロパティで定義できるトップレベルの **/schedulingpolicies** コレクションに置き換えられました。たとえば、同じスケジューリングポリシーは、最上位コレクションでは次のように表されます。

```

<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>

```

クラスターの表現は、スケジューリングポリシーをその識別子で参照します。

```

<cluster>
  <name>mycluster</name>
  <scheduling_policy id="..." />
  ...
</cluster>

```

後方互換性を維持するために、古い **policy** 要素および **thresholds** 要素が保持されました。クラスター内に組み込まれたスケジューリングポリシー表現も保持されました。これらはすべて完全に削除されたため、クラスターを取得、作成、または更新するときにスケジューリングポリシーを参照する唯一の方法は、識別子を使用して既存のポリシーを参照することです。たとえば、クラスターを取得する場合、**id** (および **href**) のみが入力されます。

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

```

<cluster>
  ...
  <scheduling_policy id="..." href="..." />
  ...
</cluster>

```

クラスターを作成または更新する場合、**id** のみが受け入れられます。

B.39. BRICKS.REPLICA_COUNT および BRICKS.STRIPE_COUNT の削除

これらの要素は、実際にはブリックのコレクションの表現の一部ではなく、ブリックを追加および削除する操作のパラメーターです。これらは削除され、新しい **replica_count** および **stripe_count** パラメーターに置き換えられました。

```
POST .../bricks?replica_count=3&stripe_count=2
```

```
DELETE .../bricks?replica_count=3
```

B.40. 統計 TYPE プロパティの名前を KIND に変更する

統計は、統計の種類 (ゲージ、カウンターなど) を示す **type** 要素と、値の型 (整数、文字列など) を示す **type** 属性を使用して表されていました。

```
<statistic>
  <type>GAUGE</type>
  <values type="INTEGER">
    <value>...</value>
    <value>...</value>
    ...
  </values>
</statistic>
```

両方の 型 概念の使用を避けるために、最初のものが **kind** に置き換えられ、**kind** と **type** の両方が要素になりました。

```
<statistic>
  <kind>gauge</kind>
  <type>integer</type>
  <values>
    <value>...</value>
    <value>...</value>
    ...
  </values>
</statistic>
```

B.41. 複数の VCPU_PIN の代わりに複数の VCPU_PINS.VCPU_PIN を使用する

これまで、仮想マシンの仮想から物理への CPU ピニングを指定するには、複数の **vcpu_pin** 要素を使用する方法を使用していました。

```
<vm>
  <cpu>
    <cpu_tune>
      <vcpu_pin>...</vcpu_pin>
      <vcpu_pin>...</vcpu_pin>
      ...
    </cpu_tune>
  </cpu>
</vm>
```

API の他の部分の一般的な慣例に準拠するために、これはラッパー要素を使用するように変更され、今回の場合は **vcpu_pins** です。

```
<vm>
  <cpu>
    <cpu_tune>
      <vcpu_pins>
        <vcpu_pin>...</vcpu_pin>
      </vcpu_pins>
    </cpu_tune>
  </cpu>
</vm>
```

```

    <vcpu_pin>...</vcpu_pin>
    ...
  </vcpu_pins>
</cpu_tune>
</cpu>
</vm>

```

B.42. FORCE パラメーターを使用してデータセンターを強制的に削除する

データセンターを削除する操作では、**force** パラメーターがサポートされています。これを使用するために、**DELETE** オペレーションはオプションのアクションパラメーターをサポートしていました。

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

```

<action>
  <force>true</force>
</action>

```

このオプションのアクションパラメーターは、オプションのパラメーターに置き換えられました。

```
DELETE /ovirt-engine/api/datacenters/123?force=true
```

B.43. FORCE パラメーターを使用してホストを強制的に削除する

ホストを削除する操作では、**force** パラメーターがサポートされています。これを使用するために、**DELETE** オペレーションはオプションのアクションパラメーターをサポートしていました。

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

```

<action>
  <force>true</force>
</action>

```

このオプションのアクションパラメーターは、オプションのパラメーターに置き換えられました。

```
DELETE /ovirt-engine/api/host/123?force=true
```

B.44. ストレージドメインの強制削除にパラメーターを使用する

ストレージドメインを削除する操作は、**force**、**destroy**、および **host** パラメーターをサポートしています。これらのパラメーターは、本体としてストレージドメインの表現を使用して **DELETE** メソッドに渡されていました。

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

```

<storage_domain>
  <force>...</force>
  <destroy>...</destroy>
  <host id="...">

```

```
<name>...</name>
</host>
</storage_domain>
```

HTTP **DELETE** パラメーターに本文が含まれているべきではなく、ストレージドメインの表現にもストレージドメインの属性ではないものを含めず、操作のパラメーターを含める必要があるため、これには問題がありました。

force、**delete**、および **host** 属性は同等のパラメーターに置き換えられ、操作では本文を使用できなくなりました。たとえば、**force** パラメーターを使用してストレージドメインを正しく削除する方法は次のとおりです。

```
DELETE /ovirt-engine/api/storagedomain/123?host=myhost&force=true
```

destroy パラメーターを使用して削除するには以下を実行します。

```
DELETE /ovirt-engine/api/storagedomain/123?host=myhost&destroy=true
```

B.45. HOST パラメーターを使用したストレージサーバー接続の削除

ストレージサーバー接続を削除する操作は、**host** パラメーターをサポートします。これを使用するには、オプションのアクションパラメーターをサポートするために使用される **DELETE** メソッドを使用します。

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

```
<action>
  <host id="...">
    <name>...</name>
  </host>
</action>
```

このオプションのアクションパラメーターは、オプションのパラメーターに置き換えられました。

```
DELETE /ovirt-engine/api/storageconnections/123?host=myhost
```

B.46. FORCE と STORAGE_DOMAIN パラメーターを使用したテンプレートディスクの削除

テンプレートディスクを削除する操作では、**force** および **storage_domain** パラメーターがサポートされます。このパラメーターを使用するには、オプションのアクションパラメーターのサポートに使用される **DELETE** メソッドを使用します。

```
DELETE /ovirt-engine/api/templates/123/disks/456
```

```
<action>
  <force>...</force>
  <storage_domain id="..."/>
</action>
```

API のバージョン 4 では、この操作は新しい **diskattachments** コレクションに移動され、要求本文はクエリーパラメーター **force** および **storage_domain** に置き換えられました。

```
DELETE /ovirt-engine/api/templates/123/disksattachments/456?force=true
```

```
DELETE /ovirt-engine/api/templates/123/disksattachments/456?storage_domain=123
```

B.47. VM ディスク API を介してディスクを削除しないでください。

/vms/123/disks/456 を削除してエンティティを削除すると、VM とディスクの関係が削除されるので、この操作では VM からディスクを切り離す必要があります。この操作では、システムからディスクを完全に削除できなくなり、ユーザーが間違いを犯しやすく、元に戻せない結果が生じていました。ディスクを削除するには、代わりに **/disk/456** API を使用します。

```
DELETE /ovirt-engine/api/disks/456
```

B.48. FORCE クエリーパラメーターを使用して、仮想マシンを強制的に削除する

仮想マシンを削除する操作では、**force** パラメーターがサポートされています。これを使用するには、オプションのアクションパラメーターをサポートするために使用される **DELETE** メソッドを使用します。

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

```
<action>
  <force>true</force>
</action>
```

このオプションのアクションパラメーターは、オプションのクエリーパラメーターに置き換えられました。

```
DELETE /ovirt-engine/api/vms/123?force=true
```

B.49. 複数のブリックを削除するには、DELETE の代わりに POST を使用する

複数の Gluster ブリックを削除する操作は、**DELETE** メソッドを使用して実装され、ブリックのリストをリクエストの本文として渡します。

```
DELETE /ovirt-engine/api/clusters/123/glustervolumes/456/bricks
```

```
<bricks>
  <bricks id="..."/>
  <bricks id="..."/>
  ...
</bricks>
```

DELETE メソッドには本体がないため、これは問題であるため、**POST** メソッドを使用する新しい削除アクションに置き換えられました。

```
POST /ovirt-engine/api/clusters/123/glustervolumes/456/bricks/remove
```

```
<bricks>
  <bricks id="..." />
  <bricks id="..." />
  ...
</bricks>
```

B.50. SCHEDULING_POLICY.POLICY 要素の削除

この要素は後方互換性のために保持されていました。代わりに **Scheduling_policy.name** を使用してください。

```
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. SNAPSHOT.SNAPSHOT_TYPE の追加

Enum は徐々に API に導入されています。これまで文字列だった一部のフィールドは、適切な列挙型に置き換えられます。そのようなフィールドの1つが `vm.type` です。ただし、このフィールドはスナップショットによって継承され、スナップショットタイプは `vm` タイプとは異なります。そのため、新しいフィールド (**`snapshot.snapshot_type`**) がスナップショットエンティティに追加されました。

```
<snapshot>
  ...
  <snapshot_type>regular|active|stateless|preview</snapshot_type>
  ...
</snapshot>
```

B.52. VM からの MOVE アクションの削除

VM エンティティの非推奨の **`move`** アクションは削除されました。代わりに、個々のディスクを移動できます。

B.53. REPORTED_CONFIGURATIONS.IN_SYNC を NETWORK_ATTACHMENT に移動する

API のバージョン 3 では、XML スキーマタイプ **ReportedConfigurations** に **in_sync** プロパティがありました。

```
<network_attachment>
  <reported_configurations>
    <in_sync>true</in_sync>
    <reported_configuration>
      ...
    </reported_configuration>
    ...
  </reported_configurations>
</network_attachment>
```

リストタイプ (報告された設定のリスト) には属性を指定できないので、API のバージョン 4 で使用される仕様メカニズムでは、これを表現することはできません。それを表現できるようにするために、属性は **in_sync** を囲む **network_attachment** に移動されました。

```
<network_attachment>
  <in_sync>true</in_sync>
  <reported_configurations>
    <reported_configuration>
      ...
    </reported_configuration>
    ...
  </reported_configurations>
</network_attachment>
```

B.54. CAPABILITIES を CLUSTERLEVELS に置き換える

最上位の **capabilities** コレクションは、新しい **clusterlevels** コレクションに置き換えられました。この新しいコレクションには、各クラスターレベルで使用可能な CPU タイプのリストなど、モデルでは使用できない情報が含まれます。

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

これにより、システムでサポートされている全クラスターレベルの詳細を含め、**ClusterLevel** オブジェクトのリストが返されます。

```
<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>
```

特定の各クラスターレベルには、バージョン自体で識別される独自のサブリソースがあります。

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

これにより、そのバージョンの詳細が返されます。

```
<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. DISKS を DISKATTACHMENTS に置き換える

バージョン 3 の API 仮想マシンとテンプレートには、接続されているディスクのすべての情報を含む **disks** コレクションがありました。API のバージョン 4 では、これらの **disks** コレクションは削除され、新しい **diskattachments** コレクションに置き換えられました。このコレクションには、ディスクへの参照と、ディスクとディスクが接続されている仮想マシンまたはテンプレートとの関係に固有の属性のみ (**interface** と **bootable**) が含まれます。

たとえば、仮想マシンに接続されているディスクを見つけるには、以下のようなリクエストを送信します。

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

以下のような応答が返されます。

```
<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>
```

ディスクの残りの詳細を見つけるには、提供されたリンクを参照してください。

仮想マシンまたはテンプレートにディスクを追加すると、新しい **disk_attachment** 要素も使用されます。リクエストが以下ようになります。

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

ディスクが存在せず、作成する場合は、次の本文を使用します。

```
<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>

```

または、ディスクがすでに存在し、ディスクを仮想マシンにアタッチするだけの場合は、次の本文を使用します。

```

<disk_attachment>
  <bootable>>false</bootable>
  <interface>virtio</interface>
  <disk id="456"/>
</disk_attachment>

```

vm.disks および **template.disks** 属性には、すべての用途に対して **disk_attachments** があることを考慮してください。たとえば、テンプレートを作成するとき、テンプレートのディスクを作成するストレージドメインを指定するために **vm.disks** 要素が使用されていました。この使用方法も **vm.disk_attachments** に置き換えられたため、特定のストレージドメインにディスクを含むテンプレートを作成するリクエストは次のようになります。

```

<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. ISCSI_TARGETS 要素を使用して、未登録のストレージを検出する

API のバージョン 3 では、未登録のストレージドメインを検出する操作は、複数の **iscsi_target** を使用して iSCSI ターゲットの一覧を受信していました。

```
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>
```

API のバージョン 4 では、この場合の **iscsi_target** のようなすべての繰り返し要素は、別の要素 (この場合は **iscsi_targets**) でラップされます。したがって、同じリクエストは次のようになります。

```
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>
```