



Red Hat Virtualization 4.4

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 4.4 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 Documentation Team

rhev-docs@redhat.com

法律上の通知

Copyright © 2023 Red Hat, Inc.

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

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

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

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

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, 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 アプリケーションプログラミングインターフェイス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	15
1.1. REPRESENTATIONAL STATE TRANSFER	15
第2章 認証とセキュリティー	17
2.1. TLS/SSL 証明書	17
2.2. 認証	19
第3章 一般的な概念	24
3.1. タイプ	24
3.2. 特定タイプ	24
3.3. オブジェクト	24
3.4. コレクション	25
3.5. 表現	25
3.6. サービス	26
3.7. 検索	29
3.8. リンクの移動	31
3.9. パーミッション	33
3.10. エラーの処理	34
第4章 クイックスタートの例	35
4.1. API エントリーポイントへのアクセス	35
4.2. データセンターの一覧表示	37
4.3. ホストクラスターの一覧表示	38
4.4. 論理ネットワークの一覧表示	39
4.5. ホストを一覧表示します。	40
4.6. NFS データストレージの作成	41
4.7. NFS ISO ストレージの作成	42
4.8. データセンターへのストレージドメインの割り当て	43
4.9. 仮想マシンの作成	45
4.10. 仮想マシン NIC の作成	46
4.11. 仮想マシンディスクの作成	47
4.12. ISO イメージの仮想マシンへのアタッチ	48
4.13. 仮想マシンの起動	50
第5章 要求	51
第6章 サービス	83
6.1. AFFINITYGROUP	83
6.2. AFFINITYGROUPHOST	84
6.3. AFFINITYGROUPHOSTLABEL	84
6.4. AFFINITYGROUPHOSTLABELS	85
6.5. AFFINITYGROUPHOSTS	86
6.6. AFFINITYGROUPVPM	87
6.7. AFFINITYGROUPVVMLABEL	88
6.8. AFFINITYGROUPVVMLABELS	88
6.9. AFFINITYGROUPVMS	89
6.10. AFFINITYGROUPS	90
6.11. AFFINITYLABEL	92
6.12. AFFINITYLABELHOST	93
6.13. AFFINITYLABELHOSTS	93
6.14. AFFINITYLABELVM	94
6.15. AFFINITYLABELVMS	95

6.16. AFFINITYLABELS	96
6.17. エリア	97
6.18. ASSIGNEDAFFINITYLABEL	97
6.19. ASSIGNEDAFFINITYLABELS	98
6.20. ASSIGNEDCPUPROFILE	99
6.21. ASSIGNEDCPUPROFILES	100
6.22. ASSIGNEDDISKPROFILE	101
6.23. ASSIGNEDDISKPROFILES	101
6.24. ASSIGNEDPERMISSIONS	102
6.25. ASSIGNEDROLES	104
6.26. ASSIGNEDTAG	105
6.27. ASSIGNEDTAGS	106
6.28. ASSIGNEDVNICPROFILE	107
6.29. ASSIGNEDVNICPROFILES	108
6.30. ATTACHEDSTORAGEDOMAIN	109
6.31. ATTACHEDSTORAGEDOMAINDISK	111
6.32. ATTACHEDSTORAGEDOMAINDISKS	114
6.33. ATTACHEDSTORAGEDOMAINS	116
6.34. バランス	117
6.35. バランス	118
6.36. ブックマーク	119
6.37. ブックマーク	120
6.38. クラスタ	122
6.39. CLUSTERENABLEDFEATURE	127
6.40. CLUSTERENABLEDFEATURES	128
6.41. CLUSTEREXTERNALPROVIDERS	129
6.42. CLUSTERFEATURE	129
6.43. CLUSTERFEATURES	130
6.44. CLUSTERLEVEL	131
6.45. CLUSTERLEVELS	132
6.46. CLUSTERNETWORK	133
6.47. CLUSTERNETWORKS	134
6.48. CLUSTERS	135
6.49. コピー可能	137
6.50. CPUPROFILE	137
6.51. CPUPROFILES	138
6.52. DATACENTER	139
6.53. DATACENTERNETWORK	143
6.54. DATACENTERNETWORKS	144
6.55. DATACENTERS	145
6.56. ディスク	147
6.57. DISKATTACHMENT	155
6.58. DISKATTACHMENTS	156
6.59. DISKPROFILE	158
6.60. DISKPROFILES	159
6.61. DISKSNAPSHOT	160
6.62. DISKSNAPSHOTS	161
6.63. ディスク	162
6.64. DOMAIN	165
6.65. DOMAINGROUP	166
6.66. DOMAINGROUPS	167
6.67. DOMAINUSER	168
6.68. DOMAINUSERGROUPS	169

6.69. DOMAINUSERS	169
6.70. ドメイン	170
6.71. ENGINEKATELLOERRATA	171
6.72. イベント	173
6.73. EVENTSUBSCRIPTION	174
6.74. EVENTSUBSCRIPTIONS	175
6.75. イベント	177
6.76. EXTERNALCOMPUTERESOURCE	181
6.77. EXTERNALCOMPUTERESOURCES	182
6.78. EXTERNALDISCOVEREDHOST	183
6.79. EXTERNALDISCOVEREDHOSTS	184
6.80. EXTERNALHOST	185
6.81. EXTERNALHOSTGROUP	186
6.82. EXTERNALHOSTGROUPS	187
6.83. EXTERNALHOSTPROVIDER	188
6.84. EXTERNALHOSTPROVIDERS	190
6.85. EXTERNALHOSTS	191
6.86. EXTERNALNETWORKPROVIDERCONFIGURATION	192
6.87. EXTERNALNETWORKPROVIDERCONFIGURATIONS	192
6.88. EXTERNALPROVIDER	193
6.89. EXTERNALPROVIDERCERTIFICATE	194
6.90. EXTERNALPROVIDERCERTIFICATES	194
6.91. EXTERNALTEMPLATEIMPORTS	195
6.92. EXTERNALVMIMPORTS	196
6.93. FENCEAGENT	197
6.94. FENCEAGENTS	198
6.95. ファイル	201
6.96. ファイル	201
6.97. FILTER	203
6.98. フィルター	204
6.99. フォロー	205
6.100. GLUSTERBRICK	205
6.101. GLUSTERBRICKS	207
6.102. GLUSTERHOOK	211
6.103. GLUSTERHOOKS	213
6.104. GLUSTERVOLUME	214
6.105. GLUSTERVOLUMES	221
6.106. グループ	223
6.107. グループ	224
6.108. ホスト	226
6.109. HOSTCPUUNITS	243
6.110. HOSTDEVICE	244
6.111. HOSTDEVICES	245
6.112. HOSTHOOK	245
6.113. HOSTHOOKS	246
6.114. HOSTNIC	247
6.115. HOSTNICS	248
6.116. HOSTNUMANODE	249
6.117. HOSTNUMANODES	250
6.118. HOSTSTORAGE	250
6.119. ホスト	252
6.120. ICON	257
6.121. ICONS	257

6.122. イメージ	258
6.123. IMAGETRANSFER	260
6.124. IMAGETRANSFERS	262
6.125. IMAGES	264
6.126. INSTANCETYPE	265
6.127. INSTANCETYPEGRAPHICSCONSOLE	266
6.128. INSTANCETYPEGRAPHICSCONSOLES	267
6.129. INSTANCETYPENIC	268
6.130. INSTANCETYPENICS	269
6.131. INSTANCETYPEWATCHDOG	270
6.132. INSTANCETYPEWATCHDOGS	271
6.133. INSTANCETYPES	272
6.134. ISCSIBOND	275
6.135. ISCSIBONDS	276
6.136. JOB	277
6.137. ジョブ	279
6.138. KATELLOERRATA	281
6.139. KATELLOERRATUM	283
6.140. LINKLAYERDISCOVERYPROTOCOL	284
6.141. MACPOOL	285
6.142. MACPOOLS	286
6.143. MEASURABLE	287
6.144. MOVEABLE	287
6.145. NETWORK	288
6.146. NETWORKATTACHMENT	290
6.147. NETWORKATTACHMENTS	291
6.148. NETWORKFILTER	292
6.149. NETWORKFILTERS	293
6.150. NETWORKLABEL	295
6.151. NETWORKLABELS	296
6.152. NETWORKS	297
6.153. NICNETWORKFILTERPARAMETER	299
6.154. NICNETWORKFILTERPARAMETERS	301
6.155. OPENSTACKIMAGE	302
6.156. OPENSTACKIMAGEPROVIDER	303
6.157. OPENSTACKIMAGEPROVIDERS	305
6.158. OPENSTACKIMAGES	306
6.159. OPENSTACKNETWORK	307
6.160. OPENSTACKNETWORKPROVIDER	308
6.161. OPENSTACKNETWORKPROVIDERS	310
6.162. OPENSTACKNETWORKS	311
6.163. OPENSTACKSUBNET	312
6.164. OPENSTACKSUBNETS	312
6.165. OPENSTACKVOLUMEAUTHENTICATIONKEY	313
6.166. OPENSTACKVOLUMEAUTHENTICATIONKEYS	314
6.167. OPENSTACKVOLUMEPROVIDER	316
6.168. OPENSTACKVOLUMEPROVIDERS	317
6.169. OPENSTACKVOLUMETYPE	319
6.170. OPENSTACKVOLUMETYPES	319
6.171. OPERATINGSYSTEM	320
6.172. OPERATINGSYSTEMS	321
6.173. パーミッション	321
6.174. PERMIT	322

6.175. PERMITS	323
6.176. QOS	325
6.177. QOSS	327
6.178. クォータ	328
6.179. QUOTACLUSTERLIMIT	330
6.180. QUOTACLUSTERLIMITS	330
6.181. QUOTASTORAGELIMIT	331
6.182. QUOTASTORAGELIMITS	332
6.183. QUOTAS	334
6.184. ロール	335
6.185. ロール	336
6.186. SCHEDULINGPOLICIES	338
6.187. SCHEDULINGPOLICY	339
6.188. SCHEDULINGPOLICYUNIT	340
6.189. SCHEDULINGPOLICYUNITS	341
6.190. スナップショット	342
6.191. SNAPSHOTCDROM	344
6.192. SNAPSHOTCDROMS	344
6.193. SNAPSHOTDISK	345
6.194. SNAPSHOTDISKS	346
6.195. SNAPSHOTNIC	346
6.196. SNAPSHOTNICS	347
6.197. スナップショット	347
6.198. SHHPUBLICKEY	350
6.199. SHHPUBLICKEYS	351
6.200. 統計	352
6.201. 統計	353
6.202. STEP	354
6.203. STEPS	356
6.204. STORAGE	358
6.205. STORAGEDOMAIN	359
6.206. STORAGEDOMAINCONTENTDISK	364
6.207. STORAGEDOMAINCONTENTDISKS	364
6.208. STORAGEDOMAINDISK	365
6.209. STORAGEDOMAINDISKS	369
6.210. STORAGEDOMAINSERVERCONNECTION	371
6.211. STORAGEDOMAINSERVERCONNECTIONS	372
6.212. STORAGEDOMAINTEMPLATE	373
6.213. STORAGEDOMAINTEMPLATES	376
6.214. STORAGEDOMAINVM	377
6.215. STORAGEDOMAINVMDISKATTACHMENT	381
6.216. STORAGEDOMAINVMDISKATTACHMENTS	382
6.217. STORAGEDOMAINVMS	382
6.218. STORAGEDOMAINS	384
6.219. STORAGESERVERCONNECTION	386
6.220. STORAGESERVERCONNECTIONEXTENSION	388
6.221. STORAGESERVERCONNECTIONEXTENSIONS	389
6.222. STORAGESERVERCONNECTIONS	391
6.223. システム	392
6.224. SYSTEMOPTION	394
6.225. SYSTEMOPTIONS	396
6.226. SYSTEMPERMISSIONS	396
6.227. タグ	398

6.228. タグ	400
6.229. TEMPLATE	402
6.230. TEMPLATECDROM	404
6.231. TEMPLATECDROMS	405
6.232. TEMPLATEDISK	406
6.233. TEMPLATEDISKATTACHMENT	408
6.234. TEMPLATEDISKATTACHMENTS	409
6.235. TEMPLATEDISKS	409
6.236. TEMPLATEGRAPHICSCONSOLE	410
6.237. TEMPLATEGRAPHICSCONSOLES	411
6.238. TEMPLATEMEDIATEDDEVICE	412
6.239. TEMPLATEMEDIATEDDEVICES	414
6.240. TEMPLATENIC	415
6.241. TEMPLATENICS	416
6.242. TEMPLATEWATCHDOG	416
6.243. TEMPLATEWATCHDOGS	417
6.244. テンプレート	418
6.245. UNMANAGEDNETWORK	422
6.246. UNMANAGEDNETWORKS	423
6.247. USER	424
6.248. USEROPTION	426
6.249. USEROPTIONS	427
6.250. USERS	428
6.251. VIRTUALFUNCTIONALLOWEDNETWORK	430
6.252. VIRTUALFUNCTIONALLOWEDNETWORKS	431
6.253. VM	432
6.254. VMAPPLICATION	450
6.255. VMAPPLICATIONS	451
6.256. VMBACKUP	453
6.257. VMBACKUPDISK	454
6.258. VMBACKUPDISKS	454
6.259. VMBACKUPS	455
6.260. VMCDROM	458
6.261. VMCDROMS	460
6.262. VMCHECKPOINT	461
6.263. VMCHECKPOINTDISK	462
6.264. VMCHECKPOINTDISKS	462
6.265. VMCHECKPOINTS	463
6.266. VMDISK	464
6.267. VMDISKS	467
6.268. VMGRAPHICSCONSOLE	468
6.269. VMGRAPHICSCONSOLES	472
6.270. VMHOSTDEVICE	473
6.271. VMHOSTDEVICES	475
6.272. VMEDIATEDDEVICE	476
6.273. VMEDIATEDDEVICES	478
6.274. VMNIC	479
6.275. VMNICS	481
6.276. VMNUMANODE	483
6.277. VMNUMANODES	484
6.278. VMPOOL	486
6.279. VMPOOLS	489
6.280. VMREPORTEDDEVICE	491

6.281. VMREPORTEDDEVICES	491
6.282. VMSESSION	492
6.283. VMSESSIONS	493
6.284. VMWATCHDOG	494
6.285. VMWATCHDOGS	496
6.286. VMS	498
6.287. VNICPROFILE	505
6.288. VNICPROFILES	506
6.289. 重み	508
6.290. 重み	509
第7章 タイプ	511
7.1. ACCESSPROTOCOL ENUM	511
7.2. ACTION 構造体	511
7.3. AFFINITYGROUP STRUCT	516
7.4. AFFINITYLABEL 構造体	518
7.5. AFFINITYRULE 構造体	519
7.6. AGENT 構造体	520
7.7. AGENTCONFIGURATION 構造体	521
7.8. API 構造体	521
7.9. APISUMMARY 構造体	523
7.10. APISUMMARYITEM STRUCT	523
7.11. APPLICATION 構造体	524
7.12. アーキテクチャー ENUM	524
7.13. AUTHORIZEDKEY 構造体	525
7.14. AUTONUMASTATUS ENUM	525
7.15. AUTOPINNINGPOLICY ENUM	526
7.16. BACKUP 構造体	527
7.17. BACKUPPHASE ENUM	528
7.18. BALANCE 構造体	528
7.19. BIOS 構造体	529
7.20. BIOSTYPE ENUM	529
7.21. BLOCKSTATISTIC 構造体	529
7.22. BONDING 構造体	530
7.23. BOOKMARK 構造体	531
7.24. BOOT 構造体	531
7.25. BOOTDEVICE ENUM	531
7.26. BOOTMENU STRUCT	532
7.27. BOOTPROTOCOL ENUM	532
7.28. BRICKPROFILEDETAIL 構造体	533
7.29. CDROM 構造体	533
7.30. CERTIFICATE 構造体	534
7.31. CHECKPOINT 構造体	535
7.32. CHECKPOINTSTATE ENUM	535
7.33. CLOUDINIT STRUCT	536
7.34. CLOUDINITNETWORKPROTOCOL ENUM	536
7.35. CLUSTER 構造体	537
7.36. CLUSTERFEATURE 構造体	545
7.37. CLUSTERLEVEL 構造体	546
7.38. CLUSTERUPGRADEACTION ENUM	547
7.39. CONFIGURATION 構造体	547
7.40. CONFIGURATIONTYPE ENUM	549
7.41. CONSOLE 構造体	550

7.42. CORE 構造体	550
7.43. CPU 構造体	550
7.44. CPUMODE ENUM	550
7.45. CPUPINNINGPOLICY ENUM	551
7.46. CPUPROFILE 構造体	552
7.47. CPUTOPOLOGY 構造体	552
7.48. CPUTUNE 構造体	553
7.49. CPUTYPE 構造体	553
7.50. CREATIONSTATUS ENUM	553
7.51. CUSTOMPROPERTY 構造体	553
7.52. DATACENTER 構造体	554
7.53. DATACENTERSTATUS ENUM	555
7.54. DEVICE 構造体	556
7.55. DISK 構造体	557
7.56. DISKATTACHMENT 構造体	562
7.57. DISKBACKUP ENUM	564
7.58. DISKBACKUPMODE ENUM	564
7.59. DISKCONTENTTYPE ENUM	565
7.60. DISKFORMAT ENUM	566
7.61. DISKINTERFACE ENUM	566
7.62. DISKPROFILE STRUCT	567
7.63. DISKSNAPSHOT 構造体	567
7.64. DISKSTATUS ENUM	573
7.65. DISKSTORAGETYPE ENUM	573
7.66. DISKTYPE ENUM	574
7.67. DISPLAY 構造体	574
7.68. DISPLAYTYPE ENUM	577
7.69. DNS 構造体	577
7.70. DNSRESOLVERCONFIGURATION 構造体	577
7.71. DOMAIN 構造体	578
7.72. DYNAMICCPU STRUCT	578
7.73. ENTITYEXTERNALSTATUS ENUM	578
7.74. ENTITYPROFILEDETAIL STRUCT	579
7.75. ERRORHANDLING 構造体	579
7.76. EVENT 構造体	579
7.77. EVENTSUBSCRIPTION 構造体	582
7.78. EXTERNALCOMPUTERESOURCE STRUCT	583
7.79. EXTERNALDISCOVEREDHOST STRUCT	583
7.80. EXTERNALHOST 構造体	584
7.81. EXTERNALHOSTGROUP STRUCT	585
7.82. EXTERNALHOSTPROVIDER STRUCT	585
7.83. EXTERNALNETWORKPROVIDERCONFIGURATION 構造体	587
7.84. EXTERNALPROVIDER 構造体	587
7.85. EXTERNALSTATUS ENUM	588
7.86. EXTERNALSYSTEMTYPE ENUM	589
7.87. EXTERNALTEMPLATEIMPORT 構造体	589
7.88. EXTERNALVMIMPORT STRUCT	590
7.89. EXTERNALVMPROVIDERTYPE ENUM	592
7.90. FAULT 構造体	592
7.91. FENCETYPE ENUM	593
7.92. FENCINGPOLICY 構造体	593
7.93. FILE 構造体	594
7.94. FILTER 構造体	594

7.95. FIPSMODE ENUM	595
7.96. FIREWALLTYPE ENUM	596
7.97. FLOPPY 構造体	596
7.98. FOPSTATISTIC STRUCT	597
7.99. GLUSTERBRICK 構造体	597
7.100. GLUSTERBRICKADVANCEDDETAILS STRUCT	598
7.101. GLUSTERBRICKMEMORYINFO STRUCT	599
7.102. GLUSTERBRICKSTATUS ENUM	600
7.103. GLUSTERCLIENT 構造体	600
7.104. GLUSTERHOOK STRUCT	600
7.105. GLUSTERHOOKSTATUS ENUM	601
7.106. GLUSTERMEMORYPOOL STRUCT	601
7.107. GLUSTERSERVERHOOK STRUCT	602
7.108. GLUSTERSTATE ENUM	603
7.109. GLUSTERVOLUME 構造体	603
7.110. GLUSTERVOLUMEPROFILEDETAILS STRUCT	604
7.111. GLUSTERVOLUMESTATUS ENUM	604
7.112. GLUSTERVOLUMETYPE ENUM	605
7.113. GRACEPERIOD 構造体	607
7.114. GRAPHICSCONSOLE 構造体	607
7.115. GRAPHICSTYPE ENUM	608
7.116. GROUP 構造体	608
7.117. GUESTOPERATINGSYSTEM STRUCT	609
7.118. HARDWAREINFORMATION 構造体	610
7.119. HIGHAVAILABILITY 構造体	611
7.120. HOOK 構造体	612
7.121. HOOKCONTENTTYPE ENUM	612
7.122. HOOKSTAGE ENUM	613
7.123. HOOKSTATUS ENUM	613
7.124. HOST 構造体	613
7.125. HOSTCPUUNIT STRUCT	622
7.126. HOSTDEVICE 構造体	622
7.127. HOSTDEVICEPASSTHROUGH 構造体	623
7.128. HOSTNIC STRUCT	624
7.129. HOSTNICVIRTUALFUNCTIONSCONFIGURATION STRUCT	627
7.130. HOSTPROTOCOL ENUM	628
7.131. HOSTSTATUS ENUM	628
7.132. HOSTSTORAGE 構造体	630
7.133. HOSTTYPE ENUM	633
7.134. HOSTEDENGINE 構造体	634
7.135. ICON 構造体	635
7.136. IDENTIFIED 構造体	635
7.137. IMAGE 構造体	636
7.138. IMAGEFILETYPE ENUM	636
7.139. IMAGETRANSFER 構造体	637
7.140. IMAGETRANSFERDIRECTION ENUM	640
7.141. IMAGETRANSFERPHASE ENUM	640
7.142. IMAGETRANSFERTIMEOUTPOLICY ENUM	642
7.143. INHERITABLEBOOLEAN ENUM	643
7.144. INITIALIZATION 構造体	643
7.145. INSTANCETYPE 構造体	645
7.146. IO 構造体	654
7.147. IP 構造体	654

7.148. IPADDRESSASSIGNMENT STRUCT	655
7.149. IPVERSION ENUM	655
7.150. ISCSIBOND STRUCT	656
7.151. ISCSIDetails STRUCT	656
7.152. JOB 構造体	657
7.153. JOBSTATUS ENUM	658
7.154. KATELLOERRATUM STRUCT	659
7.155. KDUMPSTATUS ENUM	660
7.156. KERNEL 構造体	660
7.157. KSM 構造体	660
7.158. LINKLAYERDISCOVERYPROTOCOLELEMENT STRUCT	661
7.159. LOGMAXMEMORYUSEDTHRESHOLDTYPE ENUM	662
7.160. LOGSEVERITY ENUM	662
7.161. LOGICALUNIT 構造体	663
7.162. LUNSTATUS ENUM	665
7.163. MDEVTYPE STRUCT	665
7.164. MAC 構造体	665
7.165. MACPOOL 構造体	665
7.166. MEMORYOVERCOMMIT STRUCT	667
7.167. MEMORYPOLICY 構造体	667
7.168. MESSAGEBROKERTYPE ENUM	668
7.169. METHOD 構造体	668
7.170. MIGRATEONERROR ENUM	668
7.171. MIGRATIONBANDWIDTH 構造体	668
7.172. MIGRATIONBANDWIDTHASSIGNMENTMETHOD ENUM	669
7.173. MIGRATIONOPTIONS 構造体	669
7.174. MIGRATIONPOLICY 構造体	670
7.175. NETWORK 構造体	670
7.176. NETWORKATTACHMENT 構造体	674
7.177. NETWORKCONFIGURATION 構造体	678
7.178. NETWORKFILTER 構造体	678
7.179. NETWORKFILTERPARAMETER 構造体	679
7.180. NETWORKLABEL 構造体	680
7.181. NETWORKPLUGINTYPE ENUM	680
7.182. NETWORKSTATUS ENUM	681
7.183. NETWORKUSAGE ENUM	681
7.184. NFSPROFILEDETAIL STRUCT	682
7.185. NFSVERSION ENUM	682
7.186. NIC 構造体	682
7.187. NICCONFIGURATION 構造体	684
7.188. NICINTERFACE ENUM	685
7.189. NICSTATUS ENUM	685
7.190. NOTIFIABLEEVENT ENUM	686
7.191. NOTIFICATIONMETHOD ENUM	698
7.192. NUMANODE STRUCT	698
7.193. NUMANODEPIN STRUCT	700
7.194. NUMATUNEMODE ENUM	701
7.195. OPENSTACKIMAGE 構造体	701
7.196. OPENSTACKIMAGEPROVIDER STRUCT	702
7.197. OPENSTACKNETWORK 構造体	703
7.198. OPENSTACKNETWORKPROVIDER 構造体	703
7.199. OPENSTACKNETWORKPROVIDERTYPE ENUM	706
7.200. OPENSTACKPROVIDER 構造体	707

7.201. OPENSTACKSUBNET 構造体	707
7.202. OPENSTACKVOLUMEPROVIDER 構造体	708
7.203. OPENSTACKVOLUMETYPE STRUCT	709
7.204. OPENSTACKVOLUMEAUTHENTICATIONKEY STRUCT	710
7.205. OPENSTACKVOLUMEAUTHENTICATIONKEYUSAGETYPE ENUM	711
7.206. OPERATINGSYSTEM STRUCT	711
7.207. OPERATINGSYSTEMINFO STRUCT	713
7.208. OPTION 構造体	714
7.209. OSTYPE ENUM	714
7.210. PACKAGE STRUCT	715
7.211. PARALLELMIGRATIONSPOLICY ENUM	716
7.212. PAYLOAD STRUCT	716
7.213. PAYLOADENCODING ENUM	717
7.214. PERMISSION STRUCT	717
7.215. PERMIT 構造体	718
7.216. PMPROXY 構造体	718
7.217. PMPROXYTYPE ENUM	718
7.218. POLICYUNITTYPE ENUM	719
7.219. PORTMIRRORING STRUCT	719
7.220. POWERMANAGEMENT STRUCT	719
7.221. POWERMANAGEMENTSTATUS ENUM	720
7.222. PRODUCT 構造体	721
7.223. PRODUCTINFO STRUCT	721
7.224. PROFILEDETAIL STRUCT	722
7.225. PROPERTY STRUCT	722
7.226. PROXYTICKET STRUCT	722
7.227. QCOWVERSION ENUM	723
7.228. QOS STRUCT	723
7.229. QOSTYPE ENUM	727
7.230. QUOTA 構造体	727
7.231. QUOTA_CLUSTERLIMIT STRUCT	729
7.232. QUOTAMODETYPE ENUM	729
7.233. QUOTASTORAGELIMIT STRUCT	730
7.234. RANGE 構造体	730
7.235. RATE 構造体	731
7.236. REGISTRATIONAFFINITYGROUPMAPPING 構造体	731
7.237. REGISTRATIONAFFINITYLABELMAPPING 構造体	731
7.238. REGISTRATIONCLUSTERMAPPING 構造体	732
7.239. REGISTRATIONCONFIGURATION 構造体	733
7.240. REGISTRATIONDOMAINMAPPING 構造体	736
7.241. REGISTRATIONLUNMAPPING 構造体	737
7.242. REGISTRATIONROLEMAPPING 構造体	738
7.243. REGISTRATIONVNICPROFILEMAPPING STRUCT	738
7.244. REPORTEDCONFIGURATION 構造体	740
7.245. REPORTEDDEVICE 構造体	741
7.246. REPORTEDDEVICETYPE ENUM	741
7.247. RESOLUTIONTYPE ENUM	741
7.248. RNGDEVICE 構造体	742
7.249. RNGSOURCE ENUM	742
7.250. ROLE 構造体	742
7.251. ROLETYPE ENUM	743
7.252. SCHEDULINGPOLICY STRUCT	743
7.253. SCHEDULINGPOLICYUNIT STRUCT	744

7.254. SCSIGENERICIO ENUM	745
7.255. SELINUX 構造体	745
7.256. SELINUXMODE ENUM	745
7.257. SERIALNUMBER 構造体	745
7.258. SERIALNUMBERPOLICY ENUM	746
7.259. SESSION 構造体	746
7.260. SKIPIFCONNECTIVITYBROKEN STRUCT	747
7.261. SKIPIFSDACTIVE STRUCT	748
7.262. SNAPSHOT 構造体	748
7.263. SNAPSHOTSTATUS ENUM	759
7.264. SNAPSHOTTYPE ENUM	760
7.265. SPECIALOBJECTS 構造体	760
7.266. SPM 構造体	761
7.267. SPMSTATUS ENUM	761
7.268. SSH 構造体	761
7.269. SSHAUTHENTICATIONMETHOD ENUM	762
7.270. SHHPUBLICKEY STRUCT	762
7.271. SSO 構造体	763
7.272. SSOMETHOD ENUM	763
7.273. 統計 STRUCT	763
7.274. STATISTICKIND ENUM	765
7.275. STATISTICUNIT ENUM	765
7.276. STEP 構造体	766
7.277. STEPENUM ENUM	767
7.278. STEPSTATUS ENUM	768
7.279. STORAGECONNECTION 構造体	769
7.280. STORAGECONNECTIONEXTENSION 構造体	770
7.281. STORAGEDOMAIN 構造体	770
7.282. STORAGEDOMAINLEASE 構造体	774
7.283. STORAGEDOMAINSTATUS ENUM	774
7.284. STORAGEDOMAINTYPE ENUM	775
7.285. STORAGEFORMAT ENUM	776
7.286. STORAGETYPE ENUM	778
7.287. SWITCHTYPE ENUM	779
7.288. SYSTEMOPTION STRUCT	779
7.289. SYSTEMOPTIONVALUE STRUCT	779
7.290. TAG 構造体	779
7.291. TEMPLATE 構造体	780
7.292. TEMPLATESTATUS ENUM	788
7.293. TEMPLATEVERSION 構造体	788
7.294. TICKET 構造体	789
7.295. TIMEZONE 構造体	789
7.296. TPMSUPPORT ENUM	790
7.297. TRANSPARENTHUGE_PAGES 構造体	790
7.298. TRANSPORTTYPE ENUM	790
7.299. UNMANAGEDNETWORK 構造体	790
7.300. USB 構造体	791
7.301. USBTYPE ENUM	791
7.302. USER 構造体	792
7.303. USEROPTION 構造体	793
7.304. VALUE 構造体	794
7.305. VALUETYPE ENUM	794
7.306. VCPUPIN 構造体	794

7.307. VENDOR 構造体	795
7.308. VERSION 構造体	795
7.309. VGPUPLACEMENT ENUM	795
7.310. VIRTIO SCSI STRUCT	796
7.311. VIRTUALNUMANODE STRUCT	796
7.312. VLAN 構造体	798
7.313. VM 構造体	799
7.314. VMAFFINITY ENUM	809
7.315. VMBASE 構造体	809
7.316. VMDEVICETYPE ENUM	817
7.317. VM MEDIATED DEVICE STRUCT	817
7.318. VM PLACEMENT POLICY 構造体	818
7.319. VMPOOL 構造体	818
7.320. VMPOOLTYPE ENUM	821
7.321. VMSTATUS ENUM	821
7.322. VM STORAGE ERROR RESUME BEHAVIOUR ENUM	823
7.323. VMSUMMARY 構造体	824
7.324. VM TYPE ENUM	824
7.325. VNIC PASSTHROUGH 構造体	826
7.326. VNIC PASSTHROUGH MODE ENUM	826
7.327. VNIC PROFILE 構造体	826
7.328. VNIC PROFILE MAPPING 構造体	828
7.329. VOLUME GROUP STRUCT	830
7.330. WATCHDOG 構造体	830
7.331. WATCHDOG ACTION ENUM	831
7.332. WATCHDOG MODEL ENUM	831
7.333. WEIGHT 構造体	832
付録A プリミティブ型	833
A.1. 文字列 プリミティブ	833
A.2. ブール値 プリミティブ	833
A.3. 整数 プリミティブ	833
A.4. 10進 プリミティブ型	834
A.5. DATE プリミティブ	834
付録B API バージョン 4 の変更点	836
B.1. API バージョン 4.0 での変更点 (バージョン 3.6 の後継)	836
B.2. エンジンバージョン 4.5 の変更点	862
付録C 法的通知	863

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ロールを果たします。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 環境を利用する方法の手順と実例を記載しています。

1.1. REPRESENTATIONAL STATE TRANSFER

Representational State Transfer (REST) は、特定のサービスとその表現に焦点を当てた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。

1.1.1. 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 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)。 [RFC 2616: HTTP/1.1](#) を参照してください。
- 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 の [Extensible Markup Language \(XML\) 1.0](#) および [ECMA-404: JSON data interchange syntax](#) を参照してください。

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) 脚注が必要です:[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安全にやり取りするためには、[RFC 2818: HTTP Over TLS](#) を参照してください。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。

1. 以下の例のように、**openssl s_client** コマンドを実行します。

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

出力例

```
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-----  
MIIEaTCCA1GgAwIBAgICEAQwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMCVVMx  
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW50bG9uZS5jb20  
  
SVIJe7e5FTEtHJGTAEWWM6dGbsFhip5VXM0gfqg=  
-----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-----  
MIIDxjCCAq6gAwIBAgICEAAwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMCVVMx  
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW50bG9uZS5jb20  
  
Pkyg1rQHR6ebGQ==  
-----END CERTIFICATE-----
```

-----BEGIN CERTIFICATE----- 行と -----END CERTIFICATE----- 行の間のテキストは、サーバーで提示された証明書が表示されています。

最初の証明書は、サーバー自体の証明書です。2 番目の証明書は、CA の証明書です。

2. 以下の例のように、`-----BEGIN CERTIFICATE-----` 行と `-----END CERTIFICATE-----` 行を含む CA 証明書を `ca.crt` ファイルにコピーします。

```
-----BEGIN CERTIFICATE-----
MIIDxjCCAq6gAwIBAgICEAAwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMCVVMx
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW50INDEuZXhhbXBs
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、証明書がサーバー管理者によって手動で置き換えられた場合は、正しい CA 証明書を取得することはできません。

方法 2

`openssl s_client` を使用して証明書を取得できない場合は、`curl` や `wget` などの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することができます。`curl` と `wget` は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、`https://myengine.example.com/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA` にある証明書に移動します。

選択したブラウザーによって、証明書はダウンロードされるか、またはブラウザーのキーストアにインポートされます。

- ブラウザーで証明書をダウンロードした場合は、ファイルを `ca.crt` として保存します。
- ブラウザーが証明書をインポートする場合は、ブラウザーの証明書管理オプションを使用してエクスポートし、`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は、[RFC 6749](#) 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

名前	Value
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^[1] 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	Value
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せん。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] 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	文字列	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	文字列	絶対パスとしてのオブジェクトの正規の場所。
name	文字列	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	文字列	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** パラメーターをリクエスト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。たとえば、仮想マシンを起動する action メソッド ([こちら](#) を参照) には、仮想マシンの起動方法を記述する **vm** パラメーターと、ゲスト OS を設定するために **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** 条件の指定に使用できる **search** パラメーターがあります。これを使用すると、サーバーは上記の基準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

search パラメーターは、Red Hat Virtualization クエリ言語と同じ構文を使用します。

```
(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

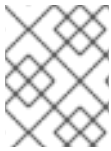
アスタリスクは値の一部として使用して、empty の文字列が含まれる文字列を検索できます。たとえば、以下のリクエストでは、**myvm**、**myvm2**、**myvma**、**myvm-webserver** など、**myvm** で始まる名前の仮想マシンをすべて返します。

```
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. リンクの移動

API は **リンク** として関連するオブジェクトへの参照を返します。たとえば、仮想マシンを取得すると、ディスクアタッチメントとネットワークインターフェイスカードへのリンクが含まれます。

```
<vm id="123" href="/ovirt-engine/api/vms/123">
...
  <link rel="diskattachments" href="/ovirt-engine/api/vms/123/diskattachments"/>
  <link rel="nics" href="/ovirt-engine/api/vms/123/nics"/>
...
</vm>
```

リンクしたオブジェクトの完全な説明は、別の要求を送信することで取得できます。

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

ただし、状況によっては、API を使用するアプリケーションが同じ要求でリンクされた情報を取得する方が便利な場合があります。追加のネットワークのラウンドトリップが原因でオーバーヘッドが許容できない範囲になったり、複数の要求が原因でアプリケーションのコードが許容できない範囲で複雑化する

場合に便利です。これらのユースケースでは、API は、アプリケーションが要求を1つだけ使用してリンクされた情報を取得できるようにする **follow** パラメーターを提供します。

follow パラメーターの値は、コンマで区切られた文字列のリストです。これらの各文字列は、リンクされたオブジェクトのパスです。たとえば、上記の例でディスクの割り当てと NIC を取得するには、要求は以下のようになります。

```
GET /ovirt-engine/api/vms/123?follow=disk_attachments,nics
```

これにより、以下のような応答が返されます。

```
<vm id="123" href="/ovirt-engine/api/vms/123">
...
<disk_attachments>
  <disk_attachment id="456" href="/ovirt-engine/api/vms/123/diskattachments/456">
    <active>true</active>
    <bootable>true</bootable>
    <interface>virtio_scsi</interface>
    <pass_discard>false</pass_discard>
    <read_only>false</read_only>
    <uses_scsi_reservation>false</uses_scsi_reservation>
    <disk id="789" href="/ovirt-engine/api/disks/789"/>
  </disk_attachment>
...
</disk_attachments>
<nics>
  <nic id="234" href="/ovirt-engine/api/vms/123/nics/234">
    <name>eth0</name>
    <interface>virtio</interface>
    <linked>true</linked>
    <mac>
      <address>00:1a:4a:16:01:00</address>
    </mac>
    <plugged>true</plugged>
  </nic>
...
</nics>
...
</vm>
```

リンクされたオブジェクトへのパスは、前の例のように1つの単語とすることも、ドットで区切った単語シーケンスで、入れ子データを要求することもできます。たとえば、前述の例では、ディスク割り当ての説明すべてを取得するために **disk_attachments** を使用しましたが、各ディスク割り当てにはディスクへのリンクが含まれていますが、各ディスク割り当てにはそのディスクへのリンクが含まれており、リンクは **フォロー** されていませんでした。また、ディスクへのリンクをたどるには、以下の要求を使用できます。

```
GET /ovirt-engine/api/vms/123?follow=disk_attachments.disk
```

これにより、以下の応答が生成されます。

```
<vm id="123" href="/ovirt-engine/api/vms/123">
  <disk_attachments>
    <disk_attachment id="456" href="/ovirt-engine/api/vms/123/diskattachments/456">
      <active>true</active>
```

```

<bootable>true</bootable>
<interface>virtio_scsi</interface>
<pass_discard>false</pass_discard>
<read_only>false</read_only>
<uses_scsi_reservation>false</uses_scsi_reservation>
<disk id="789" href="/ovirt-engine/api/disks/789">
  <name>mydisk</name>
  <description>My disk</description>
  <actual_size>0</actual_size>
  <format>raw</format>
  <sparse>true</sparse>
  <status>ok</status>
  <storage_type>image</storage_type>
  <total_size>0</total_size>
  ...
</disk>
</disk_attachment>
...
</disk_attachments>
...
</vm>

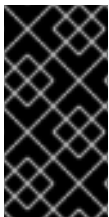
```

パスは、必要に応じて階層を深くすることができます。たとえば、ディスクの統計も取得するには、以下のコマンドを実行します。

```
GET /ovirt-engine/api/vms/123?follow=disk_attachments.disk.statistics
```

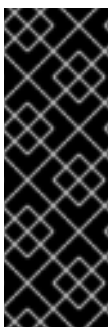
複数のパス要素と複数のパスを組み合わせることができます。たとえば、ディスクの割り当てとネットワークインターフェイスカードを、どちらも統計で取得するには、次のコマンドを実行します。

```
GET /ovirt-engine/api/vms/123?follow=disk_attachments.disk.statistics,nics.statistics
```



重要

オブジェクト取得の操作の大半は、**follow** のパラメーターをサポートしますが、一部の操作ではサポートされていない場合や、ドキュメントに、最適なパフォーマンスを得るためのアドバイスが含まれる場合があるので、参照ドキュメントを詳細に確認してください。



重要

follow パラメーターを使用すると、オーバーヘッドがクライアント側からサーバー側に移動します。追加のデータを要求する場合には、サーバーはその要求取得して基本データとマージする必要があります。これはサーバー側で CPU およびメモリーを消費し、ほとんどの場合に追加のデータベースクエリーが必要になります。これは、特に大規模な環境では、サーバーのパフォーマンスに悪影響を与える可能性があります。必ず実際の環境でアプリケーションをテストし、正当な理由がある場合にのみ **follow** パラメーターを使用してください。

3.9. 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
```

応答ボディー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。

3.10. 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 は応答ボディーに **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、応答のボディー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

本セクションの例では、REST API を使用して基本的な Red Hat Virtualization 環境を設定し、仮想マシンを作成する方法を説明します。これらの例では、標準の前提条件に加えて、以下が必要です。

- ネットワーク接続され、設定済みの Red Hat Virtualization ホスト。
- インストールする仮想マシンの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 のサンプルとして [CentOS 7](#) を使用します。

API 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 要求を紹介します。HTTP 要求を送信するアプリケーションを使用できます。



重要

この例の HTTP 要求ヘッダーでは、**Host** と **Authorization** ヘッダーが省略されています。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。

[curl](#) の例では、ユーザー名に **admin@internal**、パスワードに **mypassword**、証明書の場所には **/etc/pki/ovirt-engine/ca.pem**、ホスト名に **myengine.example.com** を使用します。ご使用の環境の正しい値に置き換える必要があります。

Red Hat Virtualization は、リソースごとに **id** 属性の一意的 ID を生成します。これら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ります。

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、簡潔さが保たれます。属性の完全なリストは、[クラスター](#) reference を参照してください。

4.1. API 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](#) コマンドを使用した同じ要求。

```
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です。

-

```

<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があります。

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. 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。以下の例では、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 コマンドを使用した同じ要求。

```
# 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. 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 コマンドを使用した同じ要求。

```
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 Nehalem 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. 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 コマンドを使用した同じ要求。

```
# 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"/>
    <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. ホストを一覧表示します。

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 コマンドを使用した同じ要求。

```
# 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. 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できますが、その場合はすでに存在している `wipe after delete` 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. 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>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) 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. データセンターへのストレージドメインの割り当て

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>
</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>mysos</name>
</storage_domain>
'\
https://myengine.example.com/ovirt-engine/api/datacenters/001/storagedomains

```

4.9. 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 '
<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

```

応答ボディは、`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. 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 '
<nic>
  <name>mynic</name>
  <description>My network interface card</description>
</nic>
'\
https://myengine.example.com/ovirt-engine/api/vms/007/nics
```

4.11. 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 '
<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. ISO イメージの仮想マシンへのアタッチ

次の仮想マシン例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、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。[ImageTransfer](#) を使用してイメージ転送を作成し、[ImageTransfers](#) を使用して ISO イメージをアップロードすることができます。

ISO イメージ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一覧を要求できます。

```

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

```

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 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 example 仮想マシンの例に割り当てます。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 コマンドを使用した同じ要求。

```
# 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. 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 '
<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}](#)
- [GET /clusterlevels/{level:id}/clusterfeatures](#)
- [GET /clusterlevels/{level:id}/clusterfeatures/{feature: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}/hostlabels](#)
- [GET /clusters/{cluster:id}/affinitygroups/{group:id}/hostlabels](#)
- [DELETE /clusters/{cluster:id}/affinitygroups/{group:id}/hostlabels/{label:id}](#)
- [POST /clusters/{cluster:id}/affinitygroups/{group:id}/hosts](#)
- [GET /clusters/{cluster:id}/affinitygroups/{group:id}/hosts](#)
- [DELETE /clusters/{cluster:id}/affinitygroups/{group:id}/hosts/{host:idorname}](#)
- [POST /clusters/{cluster:id}/affinitygroups/{group:id}/vmlabels](#)
- [GET /clusters/{cluster:id}/affinitygroups/{group:id}/vmlabels](#)
- [DELETE /clusters/{cluster:id}/affinitygroups/{group:id}/vmlabels/{label: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}/enabledfeatures](#)
- [POST /clusters/{cluster:id}/enabledfeatures](#)
- [GET /clusters/{cluster:id}/enabledfeatures/{feature:id}](#)
- [DELETE /clusters/{cluster:id}/enabledfeatures/{feature:id}](#)
- [GET /clusters/{cluster:id}/externalnetworkproviders](#)
- [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}/refreshglusterhealstatus](#)
- [POST /clusters/{cluster:id}/resetemulatedmachine](#)
- [POST /clusters/{cluster:id}/syncallnetworks](#)
- [POST /clusters/{cluster:id}/upgrade](#)
- [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}/cleanfinishedtasks
- 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}/hostlabels
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/hostlabels
- DELETE
/datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/hostlabels/{label:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/hosts
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/hosts
- DELETE
/datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/hosts/{host:idorname}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/vmlabels
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/vmlabels
- DELETE
/datacenters/{datacenter:id}/clusters/{cluster:id}/affinitygroups/{group:id}/vmlabels/{label: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}/enabledfeatures
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/enabledfeatures
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/enabledfeatures/{feature:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/enabledfeatures/{feature:id}
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/externalnetworkproviders
- 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/{brick:id}
- GET
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}
- GET
/datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}
- 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}/refreshglusterhealstatus
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/resetemulatedmachine
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/syncallnetworks
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/upgrade
- 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}/quotastorageimits
- GET /datacenters/{datacenter:id}/quotas/{quota:id}/quotastorageimits
- GET /datacenters/{datacenter:id}/quotas/{quota:id}/quotastorageimits/{limit:id}
- DELETE /datacenters/{datacenter:id}/quotas/{quota:id}/quotastorageimits/{limit:id}
- POST /datacenters/{datacenter:id}/setmaster
- POST /datacenters/{datacenter:id}/storagedomains
- GET /datacenters/{datacenter:id}/storagedomains
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}
- DELETE /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/activate
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/deactivate
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks
- PUT /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}

- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}
- DELETE /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/copy
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/export
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/move
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permission:id}
- DELETE /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permission:id}
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/register
- POST /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/sparsify
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/statistics
- GET /datacenters/{datacenter:id}/storagedomains/{storagedomain:id}/disks/{disk:id}/statistics/{statistics: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
- PUT /disks/{disk:id}

- [GET /disks/{disk:id}](#)
- [DELETE /disks/{disk:id}](#)
- [POST /disks/{disk:id}/convert](#)
- [POST /disks/{disk:id}/copy](#)
- [GET /disks/{disk:id}/disksnapshots](#)
- [GET /disks/{disk:id}/disksnapshots/{snapshot:id}](#)
- [DELETE /disks/{disk:id}/disksnapshots/{snapshot:id}](#)
- [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}/reduce](#)
- [POST /disks/{disk:id}/refreshlun](#)
- [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 /externaltemplateimports](#)
- [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}/copyhostnetworks
- GET /hosts/{host:id}/cpuunits
- POST /hosts/{host:id}/deactivate
- GET /hosts/{host:id}/devices
- GET /hosts/{host:id}/devices/{device:id}
- POST /hosts/{host:id}/discoveriscsi
- POST /hosts/{host:id}/enrollcertificate

- GET /hosts/{host:id}/externalnetworkproviderconfigurations
- GET /hosts/{host:id}/externalnetworkproviderconfigurations/{configuration:id}
- 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}
- GET /hosts/{host:id}/nics/{nic:id}/linklayerdiscoveryprotocolelements
- 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}/syncallnetworks](#)
- [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}/cancel](#)
- [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}](#)
- [POST /macpools/{macpool:id}/permissions](#)
- [GET /macpools/{macpool:id}/permissions](#)
- [GET /macpools/{macpool:id}/permissions/{permission:id}](#)
- [DELETE /macpools/{macpool:id}/permissions/{permission: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}
- GET /options/{option:id}
- POST /permissions
- GET /permissions
- GET /permissions/{permission:id}
- DELETE /permissions/{permission:id}
- POST /roles
- GET /roles
- GET /roles/{role:id}
- DELETE /roles/{role:id}

- [PUT /roles/{role:id}](#)
- [POST /roles/{role:id}/permits](#)
- [GET /roles/{role:id}/permits](#)
- [GET /roles/{role:id}/permits/{permit:id}](#)
- [DELETE /roles/{role:id}/permits/{permit:id}](#)
- [POST /schedulingpolicies](#)
- [GET /schedulingpolicies](#)
- [GET /schedulingpolicies/{policy:id}](#)
- [PUT /schedulingpolicies/{policy:id}](#)
- [DELETE /schedulingpolicies/{policy:id}](#)
- [POST /schedulingpolicies/{policy:id}/balances](#)
- [GET /schedulingpolicies/{policy:id}/balances](#)
- [GET /schedulingpolicies/{policy:id}/balances/{balance:id}](#)
- [DELETE /schedulingpolicies/{policy:id}/balances/{balance:id}](#)
- [POST /schedulingpolicies/{policy:id}/filters](#)
- [GET /schedulingpolicies/{policy:id}/filters](#)
- [GET /schedulingpolicies/{policy:id}/filters/{filter:id}](#)
- [DELETE /schedulingpolicies/{policy:id}/filters/{filter:id}](#)
- [POST /schedulingpolicies/{policy:id}/weights](#)
- [GET /schedulingpolicies/{policy:id}/weights](#)
- [GET /schedulingpolicies/{policy:id}/weights/{weight:id}](#)
- [DELETE /schedulingpolicies/{policy:id}/weights/{weight:id}](#)
- [GET /schedulingpolicyunits](#)
- [GET /schedulingpolicyunits/{unit:id}](#)
- [DELETE /schedulingpolicyunits/{unit:id}](#)
- [POST /storageconnections](#)
- [GET /storageconnections](#)
- [GET /storageconnections/{storageconnection:id}](#)
- [PUT /storageconnections/{storageconnection:id}](#)

- DELETE /storageconnections/{storageconnection:id}
- POST /storagedomains
- GET /storagedomains
- GET /storagedomains/{storagedomain:id}
- PUT /storagedomains/{storagedomain:id}
- DELETE /storagedomains/{storagedomain:id}
- POST /storagedomains/{storagedomain:id}/diskprofiles
- GET /storagedomains/{storagedomain:id}/diskprofiles
- GET /storagedomains/{storagedomain:id}/diskprofiles/{profile:id}
- DELETE /storagedomains/{storagedomain:id}/diskprofiles/{profile:id}
- POST /storagedomains/{storagedomain:id}/disks
- GET /storagedomains/{storagedomain:id}/disks
- PUT /storagedomains/{storagedomain:id}/disks/{disk:id}
- 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}/reduce
- 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}/mediateddevices
- GET /templates/{template:id}/mediateddevices
- GET /templates/{template:id}/mediateddevices/{device:id}
- PUT /templates/{template:id}/mediateddevices/{device:id}
- DELETE /templates/{template:id}/mediateddevices/{device: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}](#)
- [PUT /users/{user:id}](#)
- [DELETE /users/{user:id}](#)
- [POST /users/{user:id}/eventsubscriptions](#)
- [GET /users/{user:id}/eventsubscriptions](#)
- [GET /users/{user:id}/eventsubscriptions/{eventsubscription:id}](#)
- [DELETE /users/{user:id}/eventsubscriptions/{eventsubscription:id}](#)
- [GET /users/{user:id}/groups](#)
- [POST /users/{user:id}/options](#)

- GET /users/{user:id}/options
- GET /users/{user:id}/options/{option:id}
- DELETE /users/{user:id}/options/{option: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}/autopincpuandnumanodes](#)
- [POST /vms/{vm:id}/backups](#)
- [GET /vms/{vm:id}/backups](#)
- [GET /vms/{vm:id}/backups/{backup:id}](#)
- [GET /vms/{vm:id}/backups/{backup:id}/disks](#)
- [GET /vms/{vm:id}/backups/{backup:id}/disks/{disk:id}](#)
- [POST /vms/{vm:id}/backups/{backup:id}/finalize](#)
- [POST /vms/{vm:id}/cancelmigration](#)
- [POST /vms/{vm:id}/cdroms](#)
- [GET /vms/{vm:id}/cdroms](#)
- [GET /vms/{vm:id}/cdroms/{cdrom:id}](#)
- [PUT /vms/{vm:id}/cdroms/{cdrom:id}](#)
- [GET /vms/{vm:id}/checkpoints](#)

- GET /vms/{vm:id}/checkpoints/{checkpoint:id}
- DELETE /vms/{vm:id}/checkpoints/{checkpoint:id}
- GET /vms/{vm:id}/checkpoints/{checkpoint:id}/disks
- GET /vms/{vm:id}/checkpoints/{checkpoint:id}/disks/{disk: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}/mediateddevices](#)
- [GET /vms/{vm:id}/mediateddevices](#)
- [GET /vms/{vm:id}/mediateddevices/{device:id}](#)
- [PUT /vms/{vm:id}/mediateddevices/{device:id}](#)
- [DELETE /vms/{vm:id}/mediateddevices/{device:id}](#)
- [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}/networkfilterparameters](#)
- [POST /vms/{vm:id}/nics/{nic:id}/networkfilterparameters](#)
- [GET /vms/{vm:id}/nics/{nic:id}/networkfilterparameters/{parameter:id}](#)
- [PUT /vms/{vm:id}/nics/{nic:id}/networkfilterparameters/{parameter:id}](#)
- [DELETE /vms/{vm:id}/nics/{nic:id}/networkfilterparameters/{parameter:id}](#)
- [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}
- POST /vms/{vm:id}/reset
- POST /vms/{vm:id}/screenshot
- 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ーグループを管理します。

表6.1 メソッドの概要

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

6.1.1. get GET

アフィニティー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

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
group	AffinityGroup	Out	アフィニティーグループ。

6.1.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.1.2. remove DELETE

アフィニティー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

名前	タイプ	方向	概要
async	Boolean	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

名前	タイプ	方向	概要
async	Boolean	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. AFFINITYGROUPHOST

このサービスは、1台のホストを管理し、アフィニティーグループの割り当てを行います。

表6.5 メソッドの概要

名前	概要
remove	アフィニティーグループからホストを削除します。

6.2.1. remove DELETE

アフィニティーグループからホストを削除します。

表6.6 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. AFFINITYGROUPHOSTLABEL

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単一のホストラベルを管理します。

表6.7 メソッドの概要

名前	概要
remove	アフィニティーグループからこのラベルを削除します。

6.3.1. remove DELETE

アフィニティグループからこのラベルを削除します。

表6.8 パラメーターの概要

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

6.4. AFFINITYGROUPHOSTLABELS

このサービスは、アフィニティグループに割り当てられたすべてのホストラベルのコレクションを管理します。

表6.9 メソッドの概要

名前	概要
add	アフィニティグループにホストラベルを追加します。
list	このアフィニティグループに割り当てられたすべてのホストラベルを一覧表示します。

6.4.1. add POST

アフィニティグループにホストラベルを追加します。

たとえば、ラベル **789** をクラスター **123** のアフィニティグループ **456** に追加するには、以下のような要求を送信します。

```
POST /ovirt-engine/api/clusters/123/affinitygroups/456/hostlabels
```

以下のボディを使用します。

```
<affinity_label id="789"/>
```

表6.10 パラメーターの概要

名前	タイプ	方向	概要
ラベル	AffinityLabel	In/Out	アフィニティグループに追加する AffinityLabel オブジェクト。

6.4.2. 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ホストラベルを一覧表示します。

返されるラベルの順序は保証されません。

表6.11 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
labels	AffinityLabel []	Out	アフィニティーグループに割り当てられたホストラベル。
max	Integer	In	返すホストラベルの最大数を設定します。

6.4.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.4.2.2. max

返すホスト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。

6.5. AFFINITYGROUPHOSTS

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ホストのコレクションを管理します。

表6.12 メソッドの概要

名前	概要
add	アフィニティーグループにホストを追加します。
list	このアフィニティーグループに割り当てられたホストを一覧表示します。

6.5.1. add POST

アフィニティーグループにホストを追加します。

たとえば、ホスト **789** をクラスター **123** のアフィニティーグループ **456** に追加するには、以下のような要求を送信します。

```
POST /ovirt-engine/api/clusters/123/affinitygroups/456/hosts
```

以下のボディを使用します。

```
<host id="789"/>
```

表6.13 パラメーターの概要

名前	タイプ	方向	概要
ホスト	ホスト	In/Out	アフィニティーグループに追加するホスト。

6.5.2. list GET

このアフィニティーグループに割り当てられたホストを一覧表示します。

返されるホストの順序は保証されません。

表6.14 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
hosts	Host[]	Out	このアフィニティーグループに割り当てられたホストの一覧。
max	Integer	In	返すホストの最大数を設定します。

6.5.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.5.2.2. 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。

6.6. 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6.15 メソッドの概要

名前	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。

6.6.1. 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6.16 パラメーターの概要

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

6.7. AFFINITYGROUPVMLABEL

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単一の仮想マシンラベルを管理します。

表6.17 メソッドの概要

名前	概要
remove	アフィニティーグループからこのラベルを削除します。

6.7.1. remove DELETE

アフィニティーグループからこのラベルを削除します。

表6.18 パラメーターの概要

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

6.8. AFFINITYGROUPVMLABELS

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ラベルのコレクションを管理します。

表6.19 メソッドの概要

名前	概要
add	仮想マシンラベル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ラベルを一覧表示します。

6.8.1. add POST

仮想マシンラベルをアフィニティーグループに追加します。

たとえば、ラベル **789** をクラスター **123** のアフィニティーグループ **456** に追加するには、以下のような要求を送信します。

```
POST /ovirt-engine/api/clusters/123/affinitygroups/456/vmlabels
```

以下のボディを使用します。

```
<affinity_label id="789"/>
```

表6.20 パラメーターの概要

名前	タイプ	方向	概要
ラベル	AffinityLabel	In/Out	アフィニティーグループに追加する AffinityLabel オブジェクト。

6.8.2. list GET

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ラベルを一覧表示します。

返されるラベルの順序は保証されません。

表6.21 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
labels	AffinityLabel []	Out	アフィニティーグループに割り当てられる仮想マシンのラベル。
max	Integer	In	返す仮想マシンラベルの最大数を設定します。

6.8.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.8.2.2. max

返す仮想マシン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。

6.9. 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6.22 メソッドの概要

名前	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。

6.9.1. 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、仮想マシンの **789** をクラスター **123** のアフィニティグループ **456** に追加するには、以下のような要求を送信します。

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

以下のボディを使用します。

```
<vm id="789"/>
```

表6.23 パラメーターの概要

名前	タイプ	方向	概要
vm	Vm	In/Out	

6.9.2. 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。

返される仮想マシンの順序は保証されません。

表6.24 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す仮想マシンの最大数を設定します。
vms	Vm[]	Out	

6.9.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.9.2.2. 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す。

6.10. 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6.25 メソッドの概要

名前	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。

6.10.1. 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>
  <hosts_rule>
    <enforcing>true</enforcing>
    <positive>true</positive>
  </hosts_rule>
  <vms_rule>
    <enabled>false</enabled>
  </vms_rule>
</affinity_group>
```

表6.26 パラメーターの概要

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

6.10.2. list GET

既存のアフィニティグループを一覧表示します。

アフィニティグループの結果の順序は保証されません。

表6.27 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
groups	AffinityGroup []	Out	既存のアフィニティグループの一覧。

名前	タイプ	方向	概要
max	Integer	In	返すアフィニティグループの最大数を設定します。

6.10.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.10.2.2. max

返すアフィニティ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グループが返されます。

6.11. AFFINITYLABEL

単一のアフィニティラベルの詳細。

表6.28 メソッドの概要

名前	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。
update	ラベルを更新します。

6.11.1. get GET

ラベルの詳細を取得します。

表6.29 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
ラベル	AffinityLabel	Out	

6.11.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.11.2. 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。

6.11.3. 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6.30 パラメーターの概要

名前	タイプ	方向	概要
ラベル	AffinityLabel	In/Out	

6.12. 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6.31 メソッドの概要

名前	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。

6.12.1. 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6.32 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
ホスト	ホスト	Out	

6.12.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.12.2. remove DELETE

ホストからラベルを削除します。

6.13. 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6.33 メソッドの概要

名前	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。

6.13.1. add POST

Add a label to a host.

表6.34 パラメーターの概要

名前	タイプ	方向	概要
ホスト	ホスト	In/Out	

6.13.2. list GET

ラベルの付いたホストをすべて一覧表示します。

返されるホストの順序は保証されません。

表6.35 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
hosts	Host[]	Out	

6.13.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.14. 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6.36 メソッドの概要

名前	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。

6.14.1. 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6.37 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
vm	Vm	Out	

6.14.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.14.2. remove DELETE

仮想マシンからラベルを削除します。

6.15. 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6.38 メソッドの概要

名前	概要
add	仮想マシンにラベルを追加します。
list	ラベルの付いた仮想マシンの一覧を表示します。

6.15.1. add POST

仮想マシンにラベルを追加します。

表6.39 パラメーターの概要

名前	タイプ	方向	概要
vm	Vm	In/Out	

6.15.2. list GET

ラベルの付いた仮想マシンの一覧を表示します。

返される仮想マシンの順序は保証されません。

表6.40 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
vms	Vm[]	Out	

6.15.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.16. 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6.41 メソッドの概要

名前	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。

6.16.1. 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6.42 パラメーターの概要

名前	タイプ	方向	概要
ラベル	AffinityLabel	In/Out	

6.16.2. 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。

返されるラベルの順序は保証されません。

表6.43 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
labels	AffinityLabel []	Out	
max	Integer	In	返すラベルの最大数を設定します。

6.16.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.16.2.2. 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。

6.17. エリア

このアノテーションは、どの oVirt エリアがアノテーション付きの概念に関連しているかを指定することが目的です。現在、以下のエリアが使用されており、oVirt チームと密接に関連していますが、必ずしも同じではありません。

- インフラストラクチャー
- Network
- SLA
- Storage
- 仮想化

概念は、複数の領域に関連付けるか、領域なしに割り当てることができます。

このアノテーションの値は、レポートのみを目的としており、生成されたすべてのコードやモデルの有効性には影響しません。

6.18. 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6.44 メソッドの概要

名前	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。

6.18.1. get GET

割り当てられたラベルの詳細を取得します。

表6.45 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
ラベル	AffinityLabel	Out	

6.18.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.18.2. 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。

6.19. 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6.46 メソッドの概要

名前	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。

6.19.1. add POST

ラベルをエンティティーに割り当てます。

表6.47 パラメーターの概要

名前	タイプ	方向	概要
ラベル	AffinityLabel	In/Out	

6.19.2. list GET

エンティティに割り当てられたすべてのラベルを一覧表示します。

返されるエンティティの順序は保証されません。

表6.48 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
ラベル	AffinityLabel []	Out	

6.19.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.20. ASSIGNEDCPUPROFILE

表6.49 メソッドの概要

名前	概要
get	
remove	

6.20.1. get GET

表6.50 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
profile	CpuProfile	Out	

6.20.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.20.2. remove DELETE

表6.51 パラメーターの概要

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

6.21. ASSIGNEDCPUFILES

表6.52 メソッドの概要

名前	概要
add	クラスタの新しい cpu プロファイルを追加します。
list	クラスタに割り当てられた CPU プロファイルを一覧表示します。

6.21.1. add POST

クラスタの新しい cpu プロファイルを追加します。

表6.53 パラメーターの概要

名前	タイプ	方向	概要
profile	CpuProfile	In/Out	

6.21.2. list GET

クラスタに割り当てられた CPU プロファイルを一覧表示します。

返される CPU プロファイルの順序は保証されません。

表6.54 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプロファイルの最大数を設定します。
profiles	CpuProfile[]	Out	

6.21.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.21.2.2. 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。

6.22. ASSIGNEDDISKPROFILE

表6.55 メソッドの概要

名前	概要
get	
remove	

6.22.1. get GET

表6.56 パラメーターの概要

名前	タイプ	方向	概要
disk_profile	DiskProfile	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.22.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.22.2. remove DELETE

表6.57 パラメーターの概要

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

6.23. ASSIGNEDDISKPROFILES

表6.58 メソッドの概要

名前	概要
add	ストレージドメインの新規ディスクプロファイルを追加します。
list	ストレージドメインに割り当てられたディスクプロファイルの一覧を返します。

6.23.1. add POST

ストレージドメインの新規ディスクプロファイルを追加します。

表6.59 パラメーターの概要

名前	タイプ	方向	概要
profile	DiskProfile	In/Out	

6.23.2. list GET

ストレージドメインに割り当てられたディスクプロファイルの一覧を返します。

返されるディスクプロファイルの順序は保証されません。

表6.60 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプロファイルの最大数を設定します。
profiles	DiskProfile[]	Out	

6.23.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.23.2.2. 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。

6.24. 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6.61 メソッドの概要

名前	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。

6.24.1. 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6.62 パラメーターの概要

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

6.24.2. 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>
```

返されるパーミッションの順序は保証されません。

表6.63 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
permissions	Permission[]	Out	パーミッションのリスト

6.24.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.25. 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6.64 メソッドの概要

名前	概要
list	パーミッションに割り当てられたロールを返します。

6.25.1. list GET

パーミッションに割り当てられたロールを返します。

返されるロールの順序は保証されません。

表6.65 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すロールの最大数を設定します。
roles	Role[]	Out	

6.25.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.25.1.2. 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。

6.26. ASSIGNEDTAG

システムの特定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6.66 メソッドの概要

名前	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	システムの特定のエンティティからタグの割り当てを解除します。

6.26.1. 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6.67 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
tag	タグ	Out	割り当てられたタグ。

6.26.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.26.2. remove DELETE

システムの特定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6.68 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. 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6.69 メソッドの概要

名前	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。

6.27.1. 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6.70 パラメーターの概要

名前	タイプ	方向	概要
tag	タグ	In/Out	割り当てられたタグ。

6.27.2. 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、以下のように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>
```

返されるタグの順序は保証されません。

表6.71 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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。

6.27.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.27.2.2. 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。

6.28. ASSIGNEDVNICPROFILE

表6.72 メソッドの概要

名前	概要
get	
remove	

6.28.1. get GET

表6.73 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
profile	VnicProfile	Out	

6.28.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.28.2. remove DELETE

表6.74 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. ASSIGNEDVNICPROFILES

表6.75 メソッドの概要

名前	概要
add	ネットワーク用の新しい仮想ネットワークインターフェイスカードプロファイルを追加します。
list	ネットワークに割り当てられる VNIC プロファイルの一覧を返します。

6.29.1. add POST

ネットワーク用の新しい仮想ネットワークインターフェイスカードプロファイルを追加します。

表6.76 パラメーターの概要

名前	タイプ	方向	概要
profile	VnicProfile	In/Out	

6.29.2. list GET

ネットワークに割り当てられる VNIC プロファイルの一覧を返します。

返される VNIC プロファイルの順序は保証されません。

表6.77 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプロファイルの最大数を設定します。
profiles	VnicProfile[]	Out	

6.29.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.29.2.2. 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。

6.30. ATTACHEDSTORAGEDOMAIN

表6.78 メソッドの概要

名前	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	

6.30.1. 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6.79 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.30.2. 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。たとえば、ストレージドメイン **456** を非アクティブ化するには、次のリクエストを送信します。

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

リクエスト本文は以下のようになります。

<action/>

force パラメーターが **true** の場合、ストレージドメインの非アクティブ化が失敗する前に行われた OVF 更新が失敗した場合でも、操作は成功します。**force** パラメーターが **false** で OVF の更新に失敗すると、ストレージドメインの非アクティブ化も失敗します。

表6.80 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
force	Boolean	In	ストレージドメインの OVF 更新が失敗した場合でも、操作が成功し、ストレージドメインを非アクティブ化状態に移行する必要があるかどうかを示します。

6.30.2.1. force

ストレージドメインの OVF 更新が失敗した場合でも、操作が成功し、ストレージドメインを非アクティブ化状態に移行する必要があるかどうかを示します。たとえば、**force** フラグを使用してストレージドメイン **456** を非アクティブ化するには、次のリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<action>
  <force>true</force>
</action>
```

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。

6.30.3. get GET

表6.81 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
storage_domain	StorageDomain	Out	

6.30.3.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

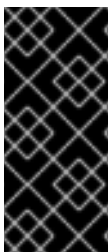
6.30.4. remove DELETE

表6.82 パラメーターの概要

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

6.31. ATTACHEDSTORAGEDOMAINDISK

データセンターに接続されたストレージドメインで使用可能な単一のディスクを管理します。



重要

エンジンのバージョン 4.2 以降、このサービスは、ストレージドメインで使用可能なディスクを一覧表示し、未登録のディスクを登録することのみを目的としています。ディスクのコピー、ディスクの移動など、他のすべての操作は非推奨になり、将来削除される予定です。これらの操作を実行するには、[システムのすべてのディスクを管理するサービス](#)、または [特定のディスクを管理するサービス](#) を使用します。

表6.83 メソッドの概要

名前	概要
copy	指定したストレージドメインにディスクをコピーします。

名前	概要
export	ディスクをエクスポートストレージドメインにエクスポートします。
get	ディスクの説明を取得します。
move	ディスクを別のストレージドメインに移動します。
register	登録解除されたディスクを登録します。
remove	ディスクを削除します。
sparsify	ディスクをスパース化します。
update	ディスクを更新します。

6.31.1. copy POST

指定したストレージドメインにディスクをコピーします。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクをコピーするには、そのディスクを管理するサービスの [コピー](#) 操作を使用します。

表6.84 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In	作成されるディスクの説明。
storage_domain	StorageDomain	In	新しいディスクが作成されるストレージドメイン。

6.31.2. export POST

ディスクをエクスポートストレージドメインにエクスポートします。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクをエクスポートするには、そのディスクを管理するサービスの [エクスポート](#) 操作を使用します。

表6.85 パラメーターの概要

名前	タイプ	方向	概要
storage_domain	StorageDomain	In	ディスクがエクスポートされるエクスポートストレージドメイン。

6.31.3. get GET

ディスクの説明を取得します。

表6.86 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	Out	ディスクの説明。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.31.3.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.31.4. move POST

ディスクを別のストレージドメインに移動します。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクを移動するには、そのディスクを管理するサービスの [移動](#) 操作を使用します。

表6.87 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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	ディスクが移動されるストレージドメイン。

6.31.5. register POST

登録解除されたディスクを登録します。

6.31.6. remove DELETE

ディスクを削除します。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクを削除するには、そのディスクを管理するサービスの [remove](#) 操作を使用します。

6.31.7. sparsify POST

ディスクをスパース化します。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクを削除するには、そのディスクを管理するサービスの [remove](#) 操作を使用します。

6.31.8. update PUT

ディスクを更新します。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクを更新するには、そのディスクを管理するサービスの [更新](#) 操作を使用します。

表6.88 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In/Out	ディスクに適用する更新。

6.32. ATTACHEDSTORAGEDOMAINDISKS

データセンターに接続されているストレージドメイン内で利用可能なディスクのコレクションを管理します。

表6.89 メソッドの概要

名前	概要
add	ディスクを追加または登録します。
list	ストレージドメインで利用可能なディスクの一覧を取得します。

6.32.1. add POST

ディスクを追加または登録します。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。新しいディスクを追加するには、システムのディスクを管理するサービスの **add** 操作を使用します。未登録のディスクを登録するには、そのディスクを管理するサービスの **登録** 操作を使用します。

表6.90 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In/Out	追加または登録するディスク。
unregistered	Boolean	In	新しいディスクを追加するか、または既存の登録されていないディスクを登録する必要があるかどうかを示します。

6.32.1.1. unregistered

新しいディスクを追加するか、または既存の登録されていないディスクを登録する必要があるかどうかを示します。値が **true** の場合、登録するディスクの ID を指定する必要があります。たとえば、id **456** でディスクを登録するには、以下のように要求を送信します。

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

リクエスト本文は以下のようになります。

```
<disk id="456"/>
```

値が **false** の場合、ストレージドメインに新しいディスクが作成されます。この場合、**provisioned_size** 属性、**format** 属性、および **name** 属性が必須となります。たとえば、1 GiB の書き込みディスクに新しいコピーを作成するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<disk>
  <name>mydisk</name>
  <format>cow</format>
  <provisioned_size>1073741824</provisioned_size>
</disk>
```

デフォルト値は **false** です。

6.32.2. list GET

ストレージドメインで利用可能なディスクの一覧を取得します。

表6.91 パラメーターの概要

名前	タイプ	方向	概要
disks	Disk[]	Out	取得されたディスクの一覧。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すディスクの最大数を設定します。

6.32.2.1. disks

取得されたディスクの一覧。

返されるディスクの順序は保証されません。

6.32.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.32.2.3. 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。

6.33. ATTACHEDSTORAGEDOMAINS

データセンターに接続されているストレージドメインを管理します。

表6.92 メソッドの概要

名前	概要
add	既存のストレージドメインをデータセンターに接続します。
list	データセンターにアタッチされているストレージドメインの一覧を返します。

6.33.1. add POST

既存のストレージドメインをデータセンターに接続します。

表6.93 パラメーターの概要

名前	タイプ	方向	概要
storage_domain	StorageDomain	In/Out	データセンターにアタッチするストレージドメイン。

6.33.2. list GET

データセンターにアタッチされているストレージドメインの一覧を返します。

返されるストレージドメインの順序は保証されません。

表6.94 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すストレージドメインの最大数を設定します。
storage_do mains	StorageDomain[]	Out	データセンターに接続されているストレージドメインの一覧。

6.33.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.33.2.2. 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。

6.34. バランス

表6.95 メソッドの概要

名前	概要
get	
remove	

6.34.1. get GET

表6.96 パラメーターの概要

名前	タイプ	方向	概要
balance	バランス	Out	
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.34.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.34.2. remove DELETE

表6.97 パラメーターの概要

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

6.35. バランス

表6.98 メソッドの概要

名前	概要
add	指定のユーザー定義のスケジューリングポリシーにバランスモジュールを追加します。
list	スケジューリングポリシーで使用されるバランスモジュールの一覧を返します。

6.35.1. add POST

指定のユーザー定義のスケジューリングポリシーにバランスモジュールを追加します。

表6.99 パラメーターの概要

名前	タイプ	方向	概要
balance	バランス	In/Out	

6.35.2. list GET

スケジューリングポリシーで使用されるバランスモジュールの一覧を返します。

返される分散モジュールの順序は保証されません。

表6.100 パラメーターの概要

名前	タイプ	方向	概要
balances	Balance[]	Out	
filter	Boolean	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。

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すバランスの最大数を設定します。

6.35.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.35.2.2. 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。

6.36. ブックマーク

ブックマークを管理するサービス

表6.101 メソッドの概要

名前	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。

6.36.1. 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6.102 パラメーターの概要

名前	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。

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.36.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.36.2. 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6.103 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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

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6.104 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.37. ブックマーク

ブックマークを管理するサービス

表6.105 メソッドの概要

名前	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。

6.37.1. 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6.106 パラメーターの概要

名前	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。

6.37.2. 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>
```

返されたブックマークの順序は保証されません。

表6.107 パラメーターの概要

名前	タイプ	方向	概要
bookmarks	Bookmark[]	Out	利用可能なブックマークの一覧。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すブックマークの最大数を設定します。

6.37.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.37.2.2. 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。

6.38. クラスタ

特定のクラスタを管理するサービス。

表6.108 メソッドの概要

名前	概要
get	クラスタに関する情報を取得します。
refreshglusterhealstatus	クラスタ内のすべてのボリュームの Gluster 修復情報を更新します。
remove	システムからクラスタを削除します。
resetemulatedmachine	
syncallnetworks	クラスタのすべてのネットワークを同期します。
update	クラスタに関する情報を更新します。
upgrade	アクションの値に基づいて、クラスタのアップグレードプロセスを開始、更新、または終了します。

6.38.1. get GET

クラスタ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 Nehalem 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>
  <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>
```

```

<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6.109 パラメーターの概要

名前	タイプ	方向	概要
cluster	クラスター	Out	
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.38.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.38.2. refreshglusterhealstatus POST

クラスター内のすべてのボリュームの Gluster 修復情報を更新します。

たとえば、Cluster **123** では、以下のような要求を送信します。

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

6.38.3. 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6.110 パラメーターの概要

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

6.38.4. resetemulatedmachine POST

表6.111 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.38.5. syncallnetworks POST

クラスターのすべてのネットワークを同期します。

POST /ovirt-engine/api/clusters/123/syncallnetworks

リクエスト本文は以下のようになります。

```
<action/>
```

表6.112 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.38.6. 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 を更新するには、次のようにします。

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6.113 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	更新を非同期的に実行するかどうかを指定します。

名前	タイプ	方向	概要
cluster	クラスター	In/Out	

6.38.7. upgrade POST

アクションの値に基づいて、クラスターのアップグレードプロセスを開始、更新、または終了します。このアクションは、**start**、**stop**、または **update_progress** の値をとるアクション値に基づいて、クラスターにアップグレードのマークを付けたり、進行状況を更新したり、クラスターのアップグレード実行フラグをクリアしたりします。

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

アップグレードの対象となるクラスターをマークするための次のようなリクエスト本文を使用します。

```
<action>
  <upgrade_action>
    start
  </upgrade_action>
</action>
```

アップグレードを開始した後、次のようなリクエスト本文を使用して、進行状況を 15% に更新します。

```
<action>
  <upgrade_action>
    update_progress
  </upgrade_action>
  <upgrade_percent_complete>
    15
  </upgrade_percent_complete>
</action>
```

表6.114 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
correlation_id	文字列	In	アップグレード関連識別子を明示的に設定します。
upgrade_action	ClusterUpgradeAction	In	実行するアクション。
upgrade_percent_complete	Integer	In	アップグレードの進行状況を、プロセス全体の完了率として更新します。

6.38.7.1. correlation_id

アップグレード関連識別子を明示的に設定します。クラスタのアップグレードの詳細を示すイベントをアップグレード自体に関連付けるために使用します。指定のない場合は、**Correlation-Id** http ヘッダーの関連 ID が使用されます。

6.39. CLUSTERENABLEDFEATURE

クラスタの有効機能を表します。

表6.115 メソッドの概要

名前	概要
get	有効なクラスタ機能に関する情報を提供します。
remove	クラスタ機能を無効にします。

6.39.1. get GET

有効なクラスタ機能に関する情報を提供します。

たとえば、クラスタ **123** で有効な機能 **456** の詳細を確認するには、以下のような要求を送信します。

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

これにより、名前が含まれる [ClusterFeature](#) オブジェクトが返されます。

```
<cluster_feature id="456">
  <name>libgfapi_supported</name>
</cluster_feature>
```

表6.116 パラメーターの概要

名前	タイプ	方向	概要
feature	ClusterFeature	Out	有効なクラスタ機能を取得している。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.39.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.39.2. remove DELETE

クラスタ機能を無効にします。

たとえば、クラスタ **123** の機能 **456** を無効にするには、以下のように要求を送信します。

-

DELETE /ovirt-engine/api/clusters/123/enabledfeatures/456

6.40. CLUSTERENABLEDFEATURES

このクラスターに対して有効にされる追加機能に関する情報を提供します。有効な機能はクラスターレベルの利用可能な機能です。

表6.117 メソッドの概要

名前	概要
add	クラスターの追加機能を有効にします。
list	クラスターに有効な追加機能を一覧表示します。

6.40.1. add POST

クラスターの追加機能を有効にします。

たとえば、クラスター **123** で機能 **456** を有効にするには、以下のような要求を送信します。

POST /ovirt-engine/api/clusters/123/enabledfeatures

リクエスト本文は以下のようになります。

```
<cluster_feature id="456"/>
```

表6.118 パラメーターの概要

名前	タイプ	方向	概要
feature	ClusterFeature	In/Out	

6.40.2. list GET

クラスターに有効な追加機能を一覧表示します。

たとえば、クラスター **123** の機能を有効にするには、以下のような要求を送信します。

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

これにより、機能のリストが返されます。

```
<enabled_features>
  <cluster_feature id="123">
    <name>test_feature</name>
  </cluster_feature>
  ...
</enabled_features>
```

表6.119 パラメーターの概要

名前	タイプ	方向	概要
features	ClusterFeature[]	Out	取得される機能。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.40.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.41. CLUSTEREXTERNALPROVIDERS

このサービスは、外部プロバイダーを一覧表示します。

表6.120 メソッドの概要

名前	概要
list	外部プロバイダー一覧を返します。

6.41.1. list GET

外部プロバイダー一覧を返します。

返されたプロバイダー一覧の順序は保証されません。

表6.121 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
providers	ExternalProvider[]	Out	

6.41.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.42. CLUSTERFEATURE

クラスターレベルで有効になっている機能を表します。

表6.122 メソッドの概要

名前	概要
get	クラスターレベルでサポートされるクラスター機能に関する情報を提供します。

6.42.1. get GET

クラスターレベルでサポートされるクラスター機能に関する情報を提供します。

たとえば、クラスターレベル 4.1 のクラスター機能 **456** の詳細は、以下のような要求を送信します。

```
GET /ovirt-engine/api/clusterlevels/4.1/clusterfeatures/456
```

これにより、名前が含まれる `ClusterFeature` オブジェクトが返されます。

```
<cluster_feature id="456">
  <name>libgfapi_supported</name>
</cluster_feature>
```

表6.123 パラメーターの概要

名前	タイプ	方向	概要
feature	<code>ClusterFeature</code>	Out	取得されるクラスター機能。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.42.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.43. CLUSTERFEATURES

クラスターレベルでサポートされるクラスター機能に関する情報を提供します。

表6.124 メソッドの概要

名前	概要
list	クラスターレベルでサポートされるクラスター機能を一覧表示します。

6.43.1. list GET

クラスターレベルでサポートされるクラスター機能を一覧表示します。

```
GET /ovirt-engine/api/clusterlevels/4.1/clusterfeatures
```

これにより、クラスターレベルでサポートされるクラスター機能のリストが返されます。

```

<cluster_features>
  <cluster_feature id="123">
    <name>test_feature</name>
  </cluster_feature>
  ...
</cluster_features>

```

表6.125 パラメーターの概要

名前	タイプ	方向	概要
features	ClusterFeature[]	Out	取得される機能。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.43.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.44. 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6.126 メソッドの概要

名前	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。

6.44.1. 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 Nehalem 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6.127 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
level	ClusterLevel	Out	取得したクラスターレベル。

6.44.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.45. 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6.128 メソッドの概要

名前	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。

6.45.1. 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>

```


返されるクラスターレベルの順序は保証されません。

表6.129 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
levels	ClusterLevel []	Out	取得されるクラスターレベル。

6.45.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.46. 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6.130 メソッドの概要

名前	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。

6.46.1. get GET

クラスターネットワークの詳細を取得します。

表6.131 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
network	Network	Out	クラスターネットワーク

6.46.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.46.2. 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。

6.46.3. update PUT

クラスターのネットワークを更新します。

表6.132 パラメーターの概要

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

6.47. CLUSTER NETWORKS

クラスターネットワークを管理するサービス

表6.133 メソッドの概要

名前	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。

6.47.1. 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6.134 パラメーターの概要

名前	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。

6.47.2. 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。

返されるクラスターの順序は保証されません。

表6.135 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すネットワークの最大数を設定します。
networks	Network[]	Out	クラスターに割り当てられているネットワークの一覧。

6.47.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.47.2.2. 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。

6.48. CLUSTERS

クラスターを管理するサービス

表6.136 メソッドの概要

名前	概要
add	新規クラスターを作成します。
list	システムのクラスター一覧を返します。

6.48.1. add POST

新規クラスターを作成します。

これには、**name**、**cpu.type**、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 Nehalem Family</type>
  </cpu>
  <data_center id="123"/>
</cluster>
```

クラスターに追加されるすべてのホストにデプロイされる外部ネットワークプロバイダーでクラスターを作成するには、以下のような要求を送信します。

POST /ovirt-engine/api/clusters

必要なプロバイダーへの参照が含まれるリクエスト本文:

```
<cluster>
  <name>mycluster</name>
  <cpu>
    <type>Intel Nehalem Family</type>
  </cpu>
  <data_center id="123"/>
  <external_network_providers>
    <external_provider name="ovirt-provider-ovn"/>
  </external_network_providers>
</cluster>
```

表6.137 パラメーターの概要

名前	タイプ	方向	概要
cluster	クラスター	In/Out	

6.48.2. list GET

システムのクラスター一覧を返します。

sortby 句が **search** パラメーターに含まれている場合にのみ、返されるクラスターの順序は保証されません。

表6.138 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	In	検索時に大文字と小文字の区別が考慮されるかどうかを示します。
clusters	Cluster[]	Out	
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すことのできるクラスターの最大数。
search	文字列	In	返されたクラスターを制限するために使用されるクエリー文字列。

6.48.2.1. case_sensitive

検索時に大文字と小文字の区別が考慮され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。無視するケースを検索するには、**false** に設定します。

6.48.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.48.2.3. 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、すべてのクラスタが返されます。

6.49. コピー可能

表6.139 メソッドの概要

名前	概要
copy	

6.49.1. copy POST

表6.140 パラメーターの概要

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

6.50. CPUPROFILE

表6.141 メソッドの概要

名前	概要
get	
remove	
update	システムで指定した cpu プロファイルを更新します。

6.50.1. get GET

表6.142 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
profile	CpuProfile	Out	

6.50.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.50.2. remove DELETE

表6.143 パラメーターの概要

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

6.50.3. update PUT

システムで指定した cpu プロファイルを更新します。

表6.144 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.51. CPUPROFILES

表6.145 メソッドの概要

名前	概要
add	システムに新しい cpu プロファイルを追加します。
list	システムの CPU プロファイル一覧を返します。

6.51.1. add POST

システムに新しい cpu プロファイルを追加します。

表6.146 パラメーターの概要

名前	タイプ	方向	概要
profile	CpuProfile	In/Out	

6.51.2. list GET

システムの CPU プロファイル一覧を返します。

返される CPU プロファイルの一覧の順序はランダムです。

表6.147 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプロファイルの最大数を設定します。
profile	CpuProfile[]	Out	

6.51.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.51.2.2. 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。

6.52. DATACENTER

データセンターを管理するサービス

表6.148 メソッドの概要

名前	概要
cleanfinishedtasks	現在、SPM に未クリアのタスクがある場合、Storage Pool Manager (SPM) は別のホストへの切り替えに失敗します。
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。
setmaster	データセンター内のストレージドメインをマスターとして手動で設定する際に使用します。
update	データセンターを更新します。

名前

概要

6.52.1. cleanfinishedtasks POST

現在、SPM に未クリアのタスクがある場合、Storage Pool Manager (SPM) は別のホストへの切り替えに失敗します。終了したすべてのタスクを消去すると SPM スイッチが有効になります。

たとえば、ID **123** のデータセンターで終了したタスクをすべてクリーニングするには、以下のように要求を送信します。

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

リクエスト本文は以下のようになります。

```
<action/>
```

表6.149 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.52.2. 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6.150 パラメーターの概要

名前	タイプ	方向	概要
data_center	DataCenter	Out	
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.52.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.52.3. 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6.151 パラメーターの概要

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

6.52.3.1. 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。

6.52.4. setmaster POST

データセンター内のストレージドメインをマスターとして手動で設定する際に使用します。たとえば、ID '456' のストレージドメインを ID '123' のデータセンターのマスターとして設定するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

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

新規マスターストレージドメインは名前でも指定することもできます。

表6.152 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
storage_domain	StorageDomain	In	データセンターの新規マスターストレージドメイン。

6.52.5. 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6.153 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.53. 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6.154 メソッドの概要

名前	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。

6.53.1. 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6.155 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
network	Network	Out	データセンターネットワーク。

6.53.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.53.2. remove DELETE

ネットワークを削除します。

6.53.3. update PUT

データセンター内のネットワークを更新します。

表6.156 パラメーターの概要

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

6.54. DATACENTERNETWORKS

データセンターネットワークを管理するサービス

表6.157 メソッドの概要

名前	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。

6.54.1. 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6.158 パラメーターの概要

名前	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。

6.54.2. 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。

返されるネットワーク一覧の順序は保証されません。

表6.159 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すネットワークの最大数を設定します。
networks	Network[]	Out	データセンターにあるネットワークの一覧。

6.54.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.54.2.2. 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。

6.55. DATACENTERS

データセンターを管理するサービス

表6.160 メソッドの概要

名前	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。

6.55.1. 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6.161 パラメーターの概要

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

6.55.2. 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。

返されるデータセンターのリストの順序は、**sortby** 句が **search** パラメーターに含まれている場合のみ保証されます。

表6.162 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すデータセンターの最大数を設定します。
search	文字列	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。

6.55.2.1. 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。

6.55.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.55.2.3. 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。

6.56. ディスク

単一ディスクを管理します。

表6.163 メソッドの概要

名前	概要
convert	ディスクフォーマットおよび/または事前割り当てモードを変換します。
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	ディスクをエクスポートストレージドメインにエクスポートします。
get	ディスクの説明を取得します。
move	ディスクを別のストレージドメインに移動します。
reduce	ディスクイメージのサイズを縮小します。
refreshlun	ストレージの最新情報で直接 LUN ディスクを更新します。
remove	ディスクを削除します。
sparsify	ディスクをスパース化します。
update	指定されたディスクのパラメーターを更新します。

6.56.1. convert POST

ディスクフォーマットや事前割り当てモードを変換します。

たとえば、ディスク形式を `preallocated-cow` から `sparse-raw` イメージに変換するには、以下のようなリクエストを送信します。

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

リクエスト本文は、以下のようになります。

```
<action>
  <disk>
    <sparse>true</sparse>
    <format>raw</format>
  </disk>
</action>
```

注: ディスクをスパース化するには、ディスクがブロックストレージドメイン上にある場合、変換が2回必要になることがあります。例: ディスクがRAWの場合、QCOWに変換するとディスクが大きくなります。サイズを小さくするために、ディスクを再度QCOWに変換し、同じ割り当てポリシーを維持することができます。

表6.164 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In	ディスクの説明。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.56.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.56.2. 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をコピー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>
```

ディスクプロファイルまたは現在ディスクが使用しているクォータが新規ストレージドメインに定義されていない場合は、明示的に指定できます。これらが指定されていない場合、最初に利用可能なディスクプロファイルとデフォルトのクォータが使用されます。

たとえば、ディスクプロファイル **987** とクォータ **753** を指定するには、以下のようなリクエスト本文を送信します。

```
<action>
  <storage_domain id="456"/>
  <disk_profile id="987"/>
  <quota id="753"/>
</action>
```

表6.165 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

名前	タイプ	方向	概要
disk_profile	DiskProfile	In	新規ストレージドメインのディスクのディスクプロファイル。
filter	Boolean	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。
quota	クォータ	In	新規ストレージドメインのディスクのクォータ。
storage_domain	StorageDomain	In	新しいディスクが作成されるストレージドメイン。

6.56.2.1. disk_profile

新規ストレージドメインのディスクのディスクプロファイル。

ディスクプロファイルはストレージドメインに定義されているため、古いディスクプロファイルは新しいストレージドメインに存在しません。このパラメーターを使用しない場合には、ユーザーにパーミッションがある新規ストレージドメインの最初のディスクプロファイルがディスクに割り当てられます。

6.56.2.2. quota

新規ストレージドメインのディスクのクォータ。

このオプションのパラメーターを使用して、ディスクの新しいクォータを指定できます。これは、現在のクォータが新しいストレージドメインに定義されていない可能性があるためです。このパラメーターが使用されず、古いクォータが新しいストレージドメインに定義されていない場合、デフォルトの(無制限の)クォータがディスクに割り当てられます。

6.56.2.3. storage_domain

新しいディスクが作成されるストレージドメイン。これは、**id** または **name** 属性を使用して指定できます。たとえば、**mydata** というストレージドメインにディスクをコピーするには、以下のように要求を送信します。

```
POST /ovirt-engine/api/storagedomains/123/disks/789
```

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>
</action>
```

6.56.3. export POST

ディスクをエクスポートストレージドメインにエクスポートします。

表6.166 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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	ディスクがエクスポートされるエクスポートストレージドメイン。

6.56.4. get GET

ディスクの説明を取得します。

表6.167 パラメーターの概要

名前	タイプ	方向	概要
all_content	Boolean	In	ディスクのすべての属性を応答に含めるかどうかを指定します。
disk	ディスク	Out	ディスクの説明。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.56.4.1. all_content

ディスクのすべての属性を応答に含めるかどうかを指定します。

デフォルトでは、以下のディスク属性が除外されます。

- **vms**

たとえば、ディスク '123' の完全な表現を取得するには、以下のコマンドを実行します。

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

6.56.4.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.56.5. 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>
```

ディスクプロファイルまたはディスクで現在使用されているクォータが新しいストレージドメインに対して定義されていない場合は、それらを明示的に指定できます。そうでない場合は、最初に使用可能なディスクプロファイルとデフォルトのクォータが使用されます。

たとえば、ディスクプロファイル **987** とクォータ **753** を明示的に使用するには、次のようなリクエスト本文を送信します。

```
<action>
  <storage_domain id="456"/>
  <disk_profile id="987"/>
  <quota id="753"/>
</action>
```

表6.168 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
disk_profile	DiskProfile	In	新規ストレージドメインのディスクのディスクプロファイル。
filter	Boolean	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。
quota	クォータ	In	新規ストレージドメインのディスクのクォータ。
storage_domain	StorageDomain	In	ディスクが移動されるストレージドメイン。

6.56.5.1. disk_profile

新規ストレージドメインのディスクのディスクプロファイル。

ディスクプロファイルはストレージドメインに定義されているため、古いディスクプロファイルは新しいストレージドメインに存在しません。このパラメーターを使用しない場合には、ユーザーにパーミッションがある新規ストレージドメインの最初のディスクプロファイルがディスクに割り当てられます。

6.56.5.2. quota

新規ストレージドメインのディスクのクォータ。

このオプションのパラメーターを使用して、ディスクの新しいクォータを指定できます。これは、現在

のクォータが新しいストレージドメインに定義されていない可能性があるためです。このパラメーターが使用されず、古いクォータが新しいストレージドメインに定義されていない場合、デフォルトの(無制限の)クォータがディスクに割り当てられます。

6.56.6. reduce POST

ディスクイメージのサイズを縮小します。

論理ボリュームで **縮小** を呼び出します(つまり、ブロックストレージドメインにのみ適用されます)。これは、フローティングディスクおよび実行されていない仮想マシンに接続されているディスクに適用されます。最適なサイズは自動的に算出されるため、サイズを指定する必要はありません。

表6.169 パラメーターの概要

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

6.56.7. refreshlun POST

ストレージの最新情報で直接 LUN ディスクを更新します。

直接 LUN ディスクの更新は、以下の場合に役に立ちます。

- LUN はホストパラメーターなしで API を使用して追加されたため、ストレージからの情報は含まれていません ([DisksService::add](#) を参照)。
- LUN に関する新しい情報がストレージで利用可能であり、それを使用して LUN を更新する必要があります。

ホスト **456** を使用して直接 LUN ディスク **123** を更新するには、以下の要求を送信します。

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

リクエスト本文は、以下のようになります。

```
<action>
  <host id='456' />
</action>
```

表6.170 パラメーターの概要

名前	タイプ	方向	概要
ホスト	ホスト	In	ダイレクト LUN ディスクの更新に使用されるホスト。

6.56.8. remove DELETE

ディスクを削除します。

表6.171 パラメーターの概要

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

6.56.9. 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。

6.56.10. update PUT

指定されたディスクのパラメーターを更新します。

この操作により、次のフローティングディスクのプロパティーを更新できます。

- イメージディスクの場合:
provisioned_size、**alias**、**description**、**wipe_after_delete**、**shareable**、**backup** および **disk_profile**。
- LUN ディスクの場合: **alias**、**description** および **shareable**。
- Cinder 統合は、マネージドブロックストレージに置き換えられました。
- Cinder および 管理対象ブロックのディスクの場合: **provisioned_size**、**alias** および **description**。
- VM 接続ディスクの場合は、**qcow_version** も更新できます。

たとえば、ディスクの更新は、次の要求を使用して実行できます。

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

リクエスト本文は以下のようになります。

```
<disk>
  <qcow_version>qcow2_v3</qcow_version>
  <alias>new-alias</alias>
  <description>new-desc</description>
</disk>
```

バックエンド操作は非同期であるため、ユーザーに返されるディスク要素は、変更されたプロパティーと同期されない場合があります。

表6.172 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In/Out	ディスクに適用する更新。

6.57. 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6.173 メソッドの概要

名前	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。

6.57.1. 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6.174 パラメーターの概要

名前	タイプ	方向	概要
attachment	DiskAttachm ent	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.57.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.57.2. 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6.175 パラメーターの概要

名前	タイプ	方向	概要
detach_only	Boolean	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。

6.57.2.1. 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。

6.57.3. 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6.176 パラメーターの概要

名前	タイプ	方向	概要
disk_attachment	DiskAttachment	In/Out	

6.58. 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6.177 メソッドの概要

名前	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。

6.58.1. 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>
```

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6.178 パラメーターの概要

名前	タイプ	方向	概要
attachment	DiskAttachment	In/Out	仮想マシンに追加するディスクアタッチメント。

6.58.2. 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。

返されたディスク割り当ての一覧の順序は保証されません。

表6.179 パラメーターの概要

名前	タイプ	方向	概要
attachments	DiskAttachment[]	Out	仮想マシンに割り当てられるディスク割り当ての一覧。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.58.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.59. DISKPROFILE

表6.180 メソッドの概要

名前	概要
get	
remove	
update	システムで指定したディスクプロファイルを更新します。

6.59.1. get GET

表6.181 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
profile	DiskProfile	Out	

6.59.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.59.2. remove DELETE

表6.182 パラメーターの概要

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

6.59.3. update PUT

システムで指定したディスクプロファイルを更新します。

表6.183 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.60. DISKPROFILES

表6.184 メソッドの概要

名前	概要
add	システムに新しいディスクプロファイルを追加します。
list	システムのディスクプロファイル一覧を返します。

6.60.1. add POST

システムに新しいディスクプロファイルを追加します。

表6.185 パラメーターの概要

名前	タイプ	方向	概要
profile	DiskProfile	In/Out	

6.60.2. list GET

システムのディスクプロファイル一覧を返します。

返されるディスクプロファイルのリストの順序は保証されません。

表6.186 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプロファイルの最大数を設定します。
profile	DiskProfile[]	Out	

6.60.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.60.2.2. 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。

6.61. DISKSNAPSHOT

表6.187 メソッドの概要

名前	概要
get	
remove	

6.61.1. get GET

表6.188 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
snapshot	DiskSnapshot	Out	

6.61.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.61.2. remove DELETE

表6.189 パラメーターの概要

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

6.62. DISKSNAPSHOTS

ストレージドメインで利用可能なディスクスナップショットのコレクションを管理します。

表6.190 メソッドの概要

名前	概要
list	ストレージドメインのディスクスナップショットの一覧を返します。

6.62.1. list GET

ストレージドメインのディスクスナップショットの一覧を返します。

返されるディスクスナップショットのリストの順序は保証されません。

表6.191 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
include_active	Boolean	In	true の場合は、アクティブなスナップショットも返します。
include_template	Boolean	In	true の場合は、テンプレートスナップショットも返します。
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	

6.62.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.62.1.2. include_active

true の場合は、アクティブなスナップショットも返します。指定されていない場合は、アクティブなスナップショットが返されません。

6.62.1.3. include_template

true の場合は、テンプレートスナップショットも返します。指定されていない場合には、テンプレートスナップショットが返されません。

6.62.1.4. 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。

6.63. 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6.192 メソッドの概要

名前	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。

6.63.1. 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、マネージドブロックディスクの 3 種類があります。Cinder 統合は、マネージドブロックストレージに置き換えられました。

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。

ブロックストレージドメイン (つまり、**ストレージタイプ** が iSCSI または FCP のストレージドメイン) は、raw **format** と **sparse=true** の組み合わせをサポートしていないため、**sparse=false** を明示的に指定する必要があることに注意してください。

ID が **123** で、増分バックアップが有効になっているストレージドメインに、**provisioned_size**、**format**、**name** を指定して新しいフローティングイメージ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>
  <storage_domains>
    <storage_domain id="123"/>
  </storage_domains>
  <name>mydisk</name>
  <provisioned_size>1048576</provisioned_size>
```

```
<format>cow</format>
<backup>incremental</backup>
</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。

Cinder 統合は、マネージドブロックストレージに置き換えられました。

ディスクスナップショットをアップロードするためのフローティングディスクの追加:

エンジンのバージョン 4.2 以降、スナップショットでディスクをアップロードできます。この要求は、イメージチェーンのベースイメージを作成するために使用する必要があります (連続するディスクスナップショット (イメージ) は、スナップショットを作成するときに **disk-attachments** 要素を使用して作成する必要があります)。

ディスクは、アップロードされたイメージと同じディスク識別子とイメージ識別子で作成する必要があります。つまり、識別子はバックアッププロセスの一部として保存する必要があります。イメージ識別子は、**qemu-img info** コマンドを使用して取得することもできます。たとえば、ディスクイメージが

b7a4c6c5-443b-47c5-967f-6abc79675e8b/myimage.img という名前のファイルに保存されている場合:

```
$ qemu-img info b7a4c6c5-443b-47c5-967f-6abc79675e8b/myimage.img
image: b548366b-fb51-4b41-97be-733c887fe305
file format: qcow2
virtual size: 1.0G (1073741824 bytes)
disk size: 196K
cluster_size: 65536
backing file: ad58716a-1fe9-481f-815e-664de1df04eb
backing file format: raw
```

上記の **qemu-img info** コマンドで取得したディスク ID とイメージ ID で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 id="b7a4c6c5-443b-47c5-967f-6abc79675e8b">
  <image_id>b548366b-fb51-4b41-97be-733c887fe305</image_id>
  <storage_domains>
    <storage_domain id="123"/>
  </storage_domains>
  <name>mydisk</name>
  <provisioned_size>1048576</provisioned_size>
  <format>cow</format>
</disk>
```

表6.193 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In/Out	ディスク。

6.63.2. 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>
  ...
</disks>
```



```

<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>

```

返されるディスクのリストの順序は、**sortby** 句が **search** パラメーターに含まれている場合にのみ保証されます。

表6.194 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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	取得されたディスクの一覧。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すディスクの最大数を設定します。
search	文字列	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。

6.63.2.1. 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。

6.63.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.63.2.3. 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。

6.64. 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6.195 メソッドの概要

名前	概要
get	認証ドメイン情報を取得します。

6.64.1. 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6.196 パラメーターの概要

名前	タイプ	方向	概要
domain	Domain	Out	認証ドメイン。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.64.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.65. DOMAINGROUP

表6.197 メソッドの概要

名前	概要
get	

6.65.1. get GET

表6.198 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
get	グループ	Out	

6.65.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.66. DOMAINGROUPS

表6.199 メソッドの概要

名前	概要
list	グループ一覧を返します。

6.66.1. list GET

グループ一覧を返します。

返されたグループリストの順序は保証されません。

表6.200 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
groups	Group[]	Out	
max	Integer	In	返すグループの最大数を設定します。
search	文字列	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。

6.66.1.1. 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。

6.66.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.66.1.3. 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。

6.67. 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6.201 メソッドの概要

名前	概要
get	ドメインユーザー情報を取得します。

6.67.1. 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6.202 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
user	User	Out	ドメインユーザー。

6.67.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.68. DOMAINUSERGROUPS

AAA 内線番号のユーザーのグループメンバーシップを表示するサービス。

表6.203 メソッドの概要

名前	概要
list	ユーザーがメンバーになっているグループの一覧を返します。

6.68.1. list GET

ユーザーがメンバーになっているグループの一覧を返します。

表6.204 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
groups	Group[]	Out	ユーザーがメンバーになっているグループのリスト。

6.68.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.69. 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6.205 メソッドの概要

名前	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。

6.69.1. 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>

```

返されるユーザーのリストの順序は保証されません。

表6.206 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すユーザーの最大数を設定します。
search	文字列	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。

6.69.1.1. 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。

6.69.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.69.1.3. 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。

6.70. 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6.207 メソッドの概要

名前	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。

6.70.1. 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>
```

返されるドメイン一覧の順序は保証されません。

表6.208 パラメーターの概要

名前	タイプ	方向	概要
domains	Domain[]	Out	ドメインの一覧。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すドメインの最大数を設定します。

6.70.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.70.1.2. 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。

6.71. 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6.209 メソッドの概要

名前	概要
list	Katello エラータの表現を取得します。

6.71.1. 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>
```

返されたエラーター一覧の順序は保証されません。

表6.210 パラメーターの概要

名前	タイプ	方向	概要
errata	KatelloErratum[]	Out	Katello エラータの表現。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すエラータの最大数を設定します。

6.71.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.71.1.2. 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。

6.72. イベント

システムのイベントを管理するサービス。

表6.211 メソッドの概要

名前	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。

6.72.1. 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6.212 パラメーターの概要

名前	タイプ	方向	概要
event	イベント	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.72.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.72.2. 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6.213 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. EVENTSUBSCRIPTION

システムの特定のイベントサブスクリプションを管理するサービス。

表6.214 メソッドの概要

名前	概要
get	event-subscription に関する情報を取得します。
remove	event-subscription をシステムから削除します。

6.73.1. get GET

event-subscription に関する情報を取得します。

たとえば、ユーザー '123' のサブスクリプションに関する情報をイベント 'vm_console_detected' を取得するには、次のようにします。

```
GET /ovirt-engine/api/users/123/vm_console_detected
```

```
<event-subscription href="/ovirt-engine/api/users/123/event-subscriptions/vm_console_detected">
  <event>vm_console_detected</event>
  <notification_method>smtp</notification_method>
  <user href="/ovirt-engine/api/users/123" id="123"/>
  <address>a@b.com</address>
</event-subscription>
```

表6.215 パラメーターの概要

名前	タイプ	方向	概要
event_subscriptions	EventSubscription	Out	event-subscription

6.73.2. remove DELETE

event-subscription をシステムから削除します。

たとえば、ユーザー 123 のサブスクリプションを **vm_console_detected** イベントから削除するには、以下を実行します。

```
DELETE /ovirt-engine/api/users/123/vm_console_detected
```

表6.216 パラメーターの概要

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

6.74. EVENTSUBSCRIPTIONS

ユーザーのイベントサブスクリプションのコレクションを管理するサービスを表します。

表6.217 メソッドの概要

名前	概要
add	システムに新しい event-subscription を追加します。
list	提供されたユーザーの event-subscriptions を一覧表示します。

6.74.1. add POST

システムに新しい event-subscription を追加します。

event-subscription は、ユーザーのコンテキストに常に追加されます。たとえば、ユーザー **123** の **host_high_cpu_use** に新しい event-subscription を追加し、通知を電子メールアドレス **a@b.com** に送信するには、以下のようなリクエストを送信します。

```
POST /ovirt-engine/api/users/123/eventsubscriptions
```

リクエスト本文は以下のようになります。

```
<event_subscription>
  <event>host_high_cpu_use</event>
  <address>a@b.com</address>
</event_subscription>
```

イベント名は新しい event-subscription エンティティの ID になります (GET ... /api/users/123/eventsubscriptions/host_high_cpu_use)。

要求の本文にはユーザー ID が指定されないことに注意してください。これは、ユーザー ID(この場合は 123) がコンテキストから API にすでに認識されているためです。また、event-subscription エンティティには notification-method フィールドが含まれていますが、リクエスト本文にも提供されていないことにも注意してください。これは、SNMP 通知が API レイヤーでまだサポートされていないため、現在は常に SMTP に設定されているためです。

表6.218 パラメーターの概要

名前	タイプ	方向	概要
event_subscription	EventSubscription	In/Out	追加された event-subscription。

6.74.2. list GET

提供されたユーザーの event-subscriptions を一覧表示します。

たとえば、ユーザー **123** の event-subscriptions を一覧表示するには、以下を実行します。

```
GET /ovirt-engine/api/users/123/event-subscriptions
```

```
<event-subscriptions>
<event-subscription href="/ovirt-engine/api/users/123/event-subscriptions/host_install_failed">
  <event>host_install_failed</event>
  <notification_method>smtp</notification_method>
  <user href="/ovirt-engine/api/users/123" id="123"/>
  <address>a@b.com</address>
</event-subscription>
<event-subscription href="/ovirt-engine/api/users/123/event-subscriptions/vm_paused">
  <event>vm_paused</event>
  <notification_method>smtp</notification_method>
  <user href="/ovirt-engine/api/users/123" id="123"/>
  <address>a@b.com</address>
</event-subscription>
</event-subscriptions>
```

表6.219 パラメーターの概要

名前	タイプ	方向	概要
event_subscriptions	EventSubscription[]	Out	指定されたユーザーの event-subscriptions の一覧
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すイベントサブスクリプションの最大数を設定します。

6.74.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.74.2.2. max

返すイベントサブスクリプションの最大数を設定します。指定されていない場合は、event-subscriptions がすべて返されます。

6.75. イベント

システムのイベントを管理するサービス。

表6.220 メソッドの概要

名前	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	

6.75.1. add POST

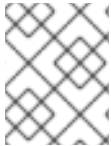
外部イベントを内部監査ログに追加します。

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6.221 パラメーターの概要

名前	タイプ	方向	概要
event	イベント	In/Out	

6.75.2. 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。

返されるイベントのリストの順序は常に付与されます。**sortby** 句が **search** パラメーターに含まれてい

る場合、イベントはその句に従って順序付けられます。**orderby** 句が含まれていない場合、イベントは **id** 属性の値でソートされ、値は最も高いものから順に並べ替えられます。これは、**max** パラメーターと組み合わせて、最新のイベントの取得を簡素化します。

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

表6.222 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
from	Integer	In	イベントを返すまでのイベントインデックスを示します。
max	Integer	In	返すイベントの最大数を設定します。
search	文字列	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。

6.75.2.1. 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。

6.75.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.75.2.3. from

イベントを返すまでのイベントインデックスを示します。イベントのインデックスは厳密に増加されるため、このパラメーターを使用すると、より大きなインデックスを持つ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** を超えるインデックス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。

6.75.2.4. 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。

6.75.2.5. 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
```

6.75.3. undelete POST

表6.223 パラメーターの概要

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

6.76. EXTERNALCOMPUTERESOURCE

単一の外部コンピュートリソースを管理します。

コンピュートリソースは、ホストの外部プロバイダーに関する用語です。外部プロバイダーは、プロビジョニングされたホストの登録先も把握する必要があります。エンジンのログイン詳細は、外部プロバイダー側でコンピュートリソースとして保存されます。

表6.224 メソッドの概要

名前	概要
get	外部コンピュートリソースの詳細を取得します。

6.76.1. get GET

外部コンピュートリソースの詳細を取得します。

たとえば、プロバイダー **123** のコンピューティングリソース **234** 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/externalhostproviders/123/computeresources/234
```

以下のような応答が返されます。

```
<external_compute_resource href="/ovirt-
engine/api/externalhostproviders/123/computeresources/234" id="234">
  <name>hostname</name>
  <provider>oVirt</provider>
  <url>https://hostname/api</url>
  <user>admin@internal</user>
  <external_host_provider href="/ovirt-engine/api/externalhostproviders/123" id="123"/>
</external_compute_resource>
```

表6.225 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
resource	ExternalComputeResource	Out	外部コンピュートリソース情報

6.76.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.77. EXTERNALCOMPUTERESOURCES

外部コンピュートリソースのコレクションを管理します。

コンピュートリソースは、ホストの外部プロバイダーに関する用語です。外部プロバイダーは、プロビジョニングされたホストの登録先も把握する必要があります。エンジンのログイン詳細は、外部プロバイダー側でコンピュートリソースとして保存されます。

表6.226 メソッドの概要

名前	概要
list	外部コンピュートリソースの一覧を取得します。

6.77.1. list GET

外部コンピュートリソースの一覧を取得します。

たとえば、外部ホストプロバイダー **123** のコンピュートリソースを取得するには、以下のような要求を送信します。

```
GET /ovirt-engine/api/externalhostproviders/123/computeresources
```

以下のような応答が返されます。

```
<external_compute_resources>
  <external_compute_resource href="/ovirt-
engine/api/externalhostproviders/123/computeresources/234" id="234">
    <name>hostname</name>
    <provider>oVirt</provider>
    <url>https://address/api</url>
    <user>admin@internal</user>
    <external_host_provider href="/ovirt-engine/api/externalhostproviders/123" id="123"/>
  </external_compute_resource>
  ...
</external_compute_resources>
```

返されたコンピュートリソースのリストの順序は保証されません。

表6.227 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すリソースの最大数を設定します。

名前	タイプ	方向	概要
resources	<code>ExternalComputeResource[]</code>	Out	外部コンピューターリソースの一覧。

6.77.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.77.1.2. 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。

6.78. EXTERNALDISCOVEREDHOST

このサービスは、単一の検出されたホストを管理します。

表6.228 メソッドの概要

名前	概要
get	検出されたホスト情報を取得します。

6.78.1. get GET

検出されたホスト情報を取得します。

Foreman などの外部プロバイダー管理システムで管理するホストに関する情報を取得します。この情報には、ホスト名、アドレス、サブネット、ベースイメージなどが含まれます。

たとえば、プロバイダー **123** からホスト **234** の詳細を取得するには、以下のような要求を送信します。

```
GET /ovirt-engine/api/externalhostproviders/123/discoveredhosts/234
```

結果は以下のようになります。

```
<external_discovered_host href="/ovirt-engine/api/externalhostproviders/123/discoveredhosts/234" id="234">
  <name>mac001a4ad04040</name>
  <ip>10.34.67.43</ip>
  <last_report>2017-04-24 11:05:41 UTC</last_report>
  <mac>00:1a:4a:d0:40:40</mac>
  <subnet_name>sat0</subnet_name>
  <external_host_provider href="/ovirt-engine/api/externalhostproviders/123" id="123"/>
</external_discovered_host>
```

表6.229 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
ホスト	ExternalDisc overedHost	Out	ホストのハードウェアおよび設定情報。

6.78.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.79. EXTERNALDISCOVEREDHOSTS

このサービスは、外部検出ホストを管理します。

表6.230 メソッドの概要

名前	概要
list	検出されたホストの情報の一覧を取得します。

6.79.1. list GET

検出されたホストの情報の一覧を取得します。

検出されたホストは、Foreman などのサードパーティーのプロバイダーから取得されます。

プロバイダー **123** の検出されたホストを一覧表示するには、以下を送信します。

```
GET /ovirt-engine/api/externalhostproviders/123/discoveredhost
```

```
<external_discovered_hosts>
  <external_discovered_host href="/ovirt-engine/api/externalhostproviders/123/discoveredhosts/456"
  id="456">
    <name>mac001a4ad04031</name>
    <ip>10.34.67.42</ip>
    <last_report>2017-04-24 11:05:41 UTC</last_report>
    <mac>00:1a:4a:d0:40:31</mac>
    <subnet_name>sat0</subnet_name>
    <external_host_provider href="/ovirt-engine/api/externalhostproviders/123" id="123"/>
  </external_discovered_host>
  <external_discovered_host href="/ovirt-engine/api/externalhostproviders/123/discoveredhosts/789"
  id="789">
    <name>mac001a4ad04040</name>
    <ip>10.34.67.43</ip>
    <last_report>2017-04-24 11:05:41 UTC</last_report>
    <mac>00:1a:4a:d0:40:40</mac>
    <subnet_name>sat0</subnet_name>
```

```

<external_host_provider href="/ovirt-engine/api/externalhostproviders/123" id="123"/>
</external_discovered_host>
...
</external_discovered_hosts>

```

返されるホスト一覧の順序は保証されません。

表6.231 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
hosts	ExternalDisc overedHost[]	Out	検出されたホストの一覧
max	Integer	In	返すホストの最大数を設定します。

6.79.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.79.1.2. 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。

6.80. EXTERNALHOST

表6.232 メソッドの概要

名前	概要
get	

6.80.1. get GET

表6.233 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
ホスト	ExternalHost	Out	

6.80.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.81. EXTERNALHOSTGROUP

このサービスは、1つのホストグループ情報を管理します。

ホストグループは、ホストプロバイダーの用語です。ホストグループには、新規検出されたホストに適用されるプロビジョニングの詳細が含まれます。サブネット、オペレーティングシステム、ドメインなどの情報

表6.234 メソッドの概要

名前	概要
get	ホストグループ情報を取得します。

6.81.1. get GET

ホストグループ情報を取得します。

たとえば、プロバイダー **123** の hostgroup **234** 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/externalhostproviders/123/hostgroups/234
```

以下のような応答が返されます。

```
<external_host_group href="/ovirt-engine/api/externalhostproviders/123/hostgroups/234" id="234">
  <name>rhel7</name>
  <architecture_name>x86_64</architecture_name>
  <domain_name>s.com</domain_name>
  <operating_system_name>RedHat 7.3</operating_system_name>
  <subnet_name>sat0</subnet_name>
  <external_host_provider href="/ovirt-engine/api/externalhostproviders/123" id="123"/>
</external_host_group>
```

表6.235 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
group	ExternalHost Group	Out	ホストグループ情報。

6.81.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.82. EXTERNALHOSTGROUPS

このサービスは、ホストグループを管理します。

表6.236 メソッドの概要

名前	概要
list	外部ホストプロバイダーからホストグループの一覧を取得します。

6.82.1. list GET

外部ホストプロバイダーからホストグループの一覧を取得します。

ホストグループはホストプロバイダーの用語です。ホストグループにはプロビジョニングの詳細が含まれます。この API は、外部プロバイダーによって公開されるすべてのホストグループを返します。

たとえば、プロバイダー **123** のすべてのホストグループ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/externalhostproviders/123/hostgroups
```

応答は以下のようになります。

```
<external_host_groups>
  <external_host_group href="/ovirt-engine/api/externalhostproviders/123/hostgroups/234" id="234">
    <name>rhel7</name>
    <architecture_name>x86_64</architecture_name>
    <domain_name>example.com</domain_name>
    <operating_system_name>RedHat 7.3</operating_system_name>
    <subnet_name>sat0</subnet_name>
    <external_host_provider href="/ovirt-engine/api/externalhostproviders/123" id="123"/>
  </external_host_group>
  ...
</external_host_groups>
```

返されたホストグループの一覧の順序は保証されません。

表6.237 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
groups	ExternalHost Group[]	Out	外部ホストプロバイダーで利用可能なホストグループの一覧
max	Integer	In	返すグループの最大数を設定します。

6.82.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.82.1.2. 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。

6.83. EXTERNALHOSTPROVIDER

Foreman や Satellite などの外部ホストのプロバイダーを表します。

詳細は、[フォアマンのドキュメント](#) を参照してください。詳細は、[Satellite のドキュメント](#) を参照してください。

表6.238 メソッドの概要

名前	概要
get	外部ホストプロバイダー情報の取得 ホストプロバイダー、Foreman または Satellite は ovirt の外部プロバイダーとして設定できます。
importcertificates	外部ホストプロバイダーの SSL 証明書をインポートします。
remove	
testconnectivity	外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。
update	システム内の指定された外部ホストプロバイダーを更新します。

6.83.1. get GET

外部ホストプロバイダー情報の取得

ホストプロバイダー、Foreman または Satellite は ovirt の外部プロバイダーとして設定できます。ovirt に割り当てられた特定のホストプロバイダーに関する詳細は、この API を使用します。

たとえば、ホストプロバイダー **123** 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/externalhostproviders/123
```

応答は以下のようになります。

```
<external_host_provider href="/ovirt-engine/api/externalhostproviders/123" id="123">
  <name>mysatellite</name>
  <requires_authentication>true</requires_authentication>
  <url>https://mysatellite.example.com</url>
  <username>admin</username>
</external_host_provider>
```


表6.239 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
provider	ExternalHost Provider	Out	

6.83.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.83.2. importcertificates POST

外部ホストプロバイダーのSSL 証明書をインポートします。

表6.240 パラメーターの概要

名前	タイプ	方向	概要
certificates	Certificate[]	In	

6.83.3. remove DELETE

表6.241 パラメーターの概要

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

6.83.4. testconnectivity POST

外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。

POST /ovirt-engine/api/externalhostproviders/123/testconnectivity

表6.242 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.83.5. update PUT

システム内の指定された外部ホストプロバイダーを更新します。

表6.243 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.84. EXTERNALHOSTPROVIDERS

表6.244 メソッドの概要

名前	概要
add	新しい外部ホストプロバイダーをシステムに追加します。
list	外部ホストプロバイダーの一覧を返します。

6.84.1. add POST

新しい外部ホストプロバイダーをシステムに追加します。

表6.245 パラメーターの概要

名前	タイプ	方向	概要
provider	ExternalHost Provider	In/Out	

6.84.2. list GET

外部ホストプロバイダーの一覧を返します。

返されるホストプロバイダーのリストの順序は保証されません。

表6.246 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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	

名前	タイプ	方向	概要
search	文字列	In	返される外部ホストプロバイダーを制限するために使用されるクエリー文字列。

6.84.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.84.2.2. 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。

6.85. EXTERNALHOSTS

表6.247 メソッドの概要

名前	概要
list	外部ホストの一覧を返します。

6.85.1. list GET

外部ホストの一覧を返します。

返されるホスト一覧の順序は保証されません。

表6.248 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
hosts	ExternalHost []	Out	
max	Integer	In	返すホストの最大数を設定します。

6.85.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.85.1.2. 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。

6.86. EXTERNALNETWORKPROVIDERCONFIGURATION

ホスト上のシステムによって外部ネットワークプロバイダーがどのようにプロビジョニングされるかについて説明します。

表6.249 メソッドの概要

名前	概要
get	ホスト上の外部ネットワークプロバイダーに関する情報を返します。

6.86.1. get GET

ホスト上の外部ネットワークプロバイダーに関する情報を返します。

表6.250 パラメーターの概要

名前	タイプ	方向	概要
configuration	ExternalNetworkProviderConfiguration[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.86.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.87. EXTERNALNETWORKPROVIDERCONFIGURATIONS

ホスト上のシステムによってプロビジョニングされたすべての外部ネットワークプロバイダーを一覧表示するサービス。

表6.251 メソッドの概要

名前	概要
list	ホスト上のすべての外部ネットワークプロバイダーの一覧を返します。

6.87.1. list GET

ホスト上のすべての外部ネットワークプロバイダーの一覧を返します。

返されたネットワークのリストの順序は保証されません。

表6.252 パラメーターの概要

名前	タイプ	方向	概要
configurations	ExternalNetworkProviderConfiguration[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.87.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.88. EXTERNALPROVIDER

外部プロバイダーを管理する機能を提供します。

表6.253 メソッドの概要

名前	概要
importcertificates	外部ホストプロバイダーの SSL 証明書をインポートします。
testconnectivity	外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。

6.88.1. importcertificates POST

外部ホストプロバイダーの SSL 証明書をインポートします。

表6.254 パラメーターの概要

名前	タイプ	方向	概要
certificates	Certificate[]	In	

6.88.2. testconnectivity POST

外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。

POST /ovirt-engine/api/externalhostproviders/123/testconnectivity

表6.255 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.89. EXTERNALPROVIDERCERTIFICATE

外部プロバイダーの特定の証明書を表示するサービス。

表6.256 メソッドの概要

名前	概要
get	特定の証明書を取得します。

6.89.1. get GET

特定の証明書を取得します。

```
GET /ovirt-engine/api/externalhostproviders/123/certificate/0
```

応答の例を以下に示します。

```
<certificate id="0">
  <organization>provider.example.com</organization>
  <subject>CN=provider.example.com</subject>
  <content>...</content>
</certificate>
```

表6.257 パラメーターの概要

名前	タイプ	方向	概要
証明書 (certificate)	証明書	Out	証明書の詳細。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.89.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.90. EXTERNALPROVIDERCERTIFICATES

外部プロバイダーの証明書を表示するサービス。

表6.258 メソッドの概要

名前	概要
list	外部プロバイダーによって提示された証明書のチェーンを返します。

6.90.1. list GET

外部プロバイダーによって提示された証明書のチェーンを返します。

```
GET /ovirt-engine/api/externalhostproviders/123/certificates
```

応答の例を以下に示します。

```
<certificates>
  <certificate id="789">...</certificate>
  ...
</certificates>
```

返される証明書の順序は常に署名の順序であることが保証されます。最初はサーバー自体の証明書であり、2番目は最初に署名する CA の証明書です。

表6.259 パラメーターの概要

名前	タイプ	方向	概要
certificates	Certificate[]	Out	証明書の詳細を含むリスト。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す証明書の最大数を設定します。

6.90.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.90.1.2. 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。

6.91. EXTERNALTEMPLATEIMPORTS

外部テンプレートをインポートする機能を提供します。現在、OVAのみをサポートしています。

表6.260 メソッドの概要

名前	概要
add	この操作は、外部ハイパーバイザーからテンプレートをインポートするために使用されます。

6.91.1. add POST

この操作は、外部ハイパーバイザーからテンプレートをインポートするために使用されます。

たとえば、テンプレート OVA 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 /externaltemplateimports

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

```
<external_template_import>
  <template>
    <name>my_template</name>
  </template>
  <cluster id="2b18aca2-4469-11eb-9449-482ae35a5f83" />
  <storage_domain id="8bb5ade5-e988-4000-8b93-dbf6717fe50" />
  <url>ova:///mnt/ova/ova_template.ova</url>
  <host id="8bb5ade5-e988-4000-8b93-dbf6717fe50" />
</external_template_import>
```

表6.261 パラメーターの概要

名前	タイプ	方向	概要
import	ExternalTemplateImport	In/Out	

6.92. 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6.262 メソッドの概要

名前	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。

6.92.1. 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://vmware_user@vcenter-host/DataCenter/Cluster/esxi-host?no_verify=1</url>
  <drivers_iso id="virtio-win-1.6.7.iso" />
</external_vm_import>

```

表6.263 パラメーターの概要

名前	タイプ	方向	概要
import	ExternalVml mport	In/Out	

6.93. FENCEAGENT

特定のホストのフェンスエージェントを管理するサービス。

表6.264 メソッドの概要

名前	概要
get	このフェンスエージェントの詳細を取得します。
remove	特定ホストのフェンスエージェントを削除します。
update	fencing-agent を更新します。

6.93.1. get GET

このフェンスエージェントの詳細を取得します。

```
GET /ovirt-engine/api/hosts/123/fenceagents/0
```

応答の例を以下に示します。

```

<agent id="0">
  <type>apc</type>
  <order>1</order>
  <ip>192.168.1.101</ip>
  <user>user</user>
  <password>xxx</password>

```

```
<port>9</port>
<options>name1=value1, name2=value2</options>
</agent>
```

表6.265 パラメーターの概要

名前	タイプ	方向	概要
agent (エージェント)	エージェント	Out	フェンスエージェントの詳細。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.93.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.93.2. remove DELETE

特定ホストのフェンスエージェントを削除します。

```
DELETE /ovirt-engine/api/hosts/123/fenceagents/0
```

表6.266 パラメーターの概要

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

6.93.3. update PUT

fencing-agent を更新します。

表6.267 パラメーターの概要

名前	タイプ	方向	概要
agent (エージェント)	エージェント	In/Out	フェンスエージェントの詳細。
async	Boolean	In	更新を非同期的に実行するかどうかを指定します。

6.94. FENCEAGENTS

特定のホストのフェンスエージェントを管理するサービス。

表6.268 メソッドの概要

名前	概要
add	ホストに新しい fencing-agent を追加します。
list	ホストに設定されたフェンシングエージェントの一覧を返します。

6.94.1. add POST

ホストに新しい fencing-agent を追加します。

POST /ovirt-engine/api/hosts/123/fenceagents

You should consult the /usr/sbin/fence_<agent_name> manual page for the legal parameters to [name1=value1, name2=value2,...] in the options field. If any parameter in options appears by name that means that it is mandatory. For example in <options>slot=7[,name1=value1, name2=value2,...]</options> slot is mandatory.

apc、bladecenter、wti フェンシングエージェントのサンプルリクエスト:

```
<agent>
  <type>apc</type>
  <order>1</order>
  <ip>192.168.1.101</ip>
  <user>user</user>
  <password>xxx</password>
  <port>9</port>
  <options>slot=7[,name1=value1, name2=value2,...]</options>
</agent>
```

apc_snmp、hpblade、ilo、ilo2、ilo_ssh、redfish、rsa フェンシングエージェントのサンプルリクエスト:

```
<agent>
  <type>apc_snmp</type>
  <order>1</order>
  <ip>192.168.1.101</ip>
  <user>user</user>
  <password>xxx</password>
  <port>9</port>
  <options>[name1=value1, name2=value2,...]</options>
</agent>
```

cisco_ucs、drac5、eps フェンシングエージェントのサンプルリクエスト:

```
<agent>
  <type>cisco_ucs</type>
  <order>1</order>
  <ip>192.168.1.101</ip>
  <user>user</user>
```

```
<password>xxx</password>
<options>slot=7[,name1=value1, name2=value2,...]</options>
</agent>
```

drac7、ilo3、ilo4、ipmilan、rsb フェンシングエージェントのサンプルリクエスト:

```
<agent>
  <type>drac7</type>
  <order>1</order>
  <ip>192.168.1.101</ip>
  <user>user</user>
  <password>xxx</password>
  <options>[name1=value1, name2=value2,...]</options>
</agent>
```

表6.269 パラメーターの概要

名前	タイプ	方向	概要
agent (エージェント)	エージェント	In/Out	

6.94.2. list GET

ホストに設定されたフェンシングエージェントの一覧を返します。

GET /ovirt-engine/api/hosts/123/fenceagents

応答の例を以下に示します。

```
<agents>
  <agent id="0">
    <type>apc</type>
    <order>1</order>
    <ip>192.168.1.101</ip>
    <user>user</user>
    <password>xxx</password>
    <port>9</port>
    <options>name1=value1, name2=value2</options>
  </agent>
</agents>
```

返されたフェンシングエージェントのリストの順序は保証されません。

表6.270 パラメーターの概要

名前	タイプ	方向	概要
agents	Agent[]	Out	フェンスエージェントの詳細のリスト。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
max	Integer	In	返すエージェントの最大数を設定します。

6.94.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.94.2.2. 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。

6.95. ファイル

表6.271 メソッドの概要

名前	概要
get	

6.95.1. get GET

表6.272 パラメーターの概要

名前	タイプ	方向	概要
file	ファイル	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.95.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.96. 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 イメージと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に対象としています。

CD-ROM 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6.273 メソッドの概要

名前	概要
list	ストレージドメインで利用可能な ISO イメージおよび仮想フロッピーのディスク一覧を返します。

6.96.1. list GET

ストレージドメインで利用可能な ISO イメージおよび仮想フロッピーのディスク一覧を返します。返されるリストの順序は保証されません。

refresh パラメーターが **false** の場合、返されるリストはストレージドメインへの最近の変更を反映していない可能性があります。たとえば、最近追加された新しい ISO ファイルが含まれていない場合があります。これは、サーバーがファイルの一覧をキャッシュしてパフォーマンスを向上するためです。最新の結果を取得するには、**refresh** パラメーターを **true** に設定します。

refresh パラメーターのデフォルト値は **true** ですが、**ForceRefreshDomainFilesByDefault** 値を使用して変更できます。

```
# engine-config -s ForceRefreshDomainFilesByDefault=false
```



重要

refresh パラメーターの値を **true** に設定すると、サーバーのパフォーマンスに影響します。必要な場合にのみ使用してください。

表6.274 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	In	search パラメーターを使用して実行した検索を討すべきかどうかを示します。
file	File[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すファイルの最大数を設定します。
refresh	Boolean	In	特定の間隔で更新されるキャッシュされた結果を表示するのではなく、ファイルのリストをストレージドメインから更新する必要があるかどうかを示します。
search	文字列	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ー文字列。

6.96.1.1. case_sensitive

search パラメーターを使用して実行した検索を討すべきかどうかを示します。デフォルト値は **true** です。

6.96.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.96.1.3. max

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

6.97. FILTER

表6.275 メソッドの概要

名前	概要
get	
remove	

6.97.1. get GET

表6.276 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
result	Filter	Out	

6.97.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.97.2. remove DELETE

表6.277 パラメーターの概要

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

6.98. フィルター

スケジューリングポリシーで使用されるフィルターを管理します。

表6.278 メソッドの概要

名前	概要
add	指定したユーザー定義のスケジューリングポリシーにフィルターを追加します。
list	スケジューリングポリシーで使用されるフィルターの一覧を返します。

6.98.1. add POST

指定したユーザー定義のスケジューリングポリシーにフィルターを追加します。

表6.279 パラメーターの概要

名前	タイプ	方向	概要
filter	Filter	In/Out	

6.98.2. list GET

スケジューリングポリシーで使用されるフィルターの一覧を返します。

返されるフィルターのリストの順序は保証されません。

表6.280 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すフィルターの最大数を設定します。

6.98.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.98.2.2. 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。

6.99. フォロー

6.100. GLUSTERBRICK

このサービスは単一の gluster ブリックを管理します。

表6.281 メソッドの概要

名前	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。

6.100.1. 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>
    </memory_pool>
  </memory_pools>
</brick>
```

```

<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_o
ptions>
<pid>25589</pid>
<port>49155</port>
</brick>

```

表6.282 パラメーターの概要

名前	タイプ	方向	概要
brick	GlusterBrick	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.100.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.100.2. 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。

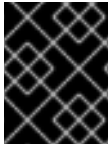
```
DELETE /ovirt-engine/api/clusters/567/glustervolumes/123/glusterbricks/234
```

表6.283 パラメーターの概要

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

6.100.3. 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6.284 パラメーターの概要

名前	タイプ	方向	概要
<code>async</code>	Boolean	In	代替を非同期的に実行するかどうかを指定します。
<code>force</code>	Boolean	In	

6.101. 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6.285 メソッドの概要

名前	概要
<code>activate</code>	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。
<code>add</code>	gluster ボリュームに、ブリックの一覧を追加します。
<code>list</code>	gluster ボリュームのブリックを一覧表示します。
<code>migrate</code>	ブリックを削除する前にデータの移行を開始します。
<code>remove</code>	gluster ボリュームからブリックを削除します。
<code>stopmigrate</code>	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。

6.101.1. 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6.286 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
ブリック	GlusterBrick[]	In	再アクティベートする必要があるブリックの一覧。

6.101.2. 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6.287 パラメーターの概要

名前	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。

6.101.3. 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>
```

返される一覧の順序は、gluster ボリュームの作成時に提供されるブリックの順序に基づいています。

表6.288 パラメーターの概要

名前	タイプ	方向	概要
bricks	GlusterBrick[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すブリックの最大数を設定します。

6.101.3.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.101.3.2. 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。

6.101.4. 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6.289 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.101.5. 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6.290 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.101.6. 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6.291 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.101.6.1. 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。

6.102. GLUSTERHOOK

表6.292 メソッドの概要

名前	概要
disable	クラスタの全サーバーで 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	
remove	クラスタ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。

6.102.1. disable POST

クラスタの全サーバーで 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6.293 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.102.2. enable POST

クラスタの全サーバーで 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6.294 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.102.3. get GET

表6.295 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
hook	GlusterHook	Out	

6.102.3.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.102.4. 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6.296 パラメーターの概要

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

6.102.5. 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6.297 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
ホスト	ホスト	In	
resolution_type	文字列	In	

6.103. GLUSTERHOOKS

表6.298 メソッドの概要

名前	概要
list	フック一覧を返します。

6.103.1. list GET

フック一覧を返します。

返されるフック一覧の順序は保証されません。

表6.299 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
hooks	GlusterHook []	Out	
max	Integer	In	返すフックの最大数を設定します。

6.103.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.103.1.2. 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。

6.104. GLUSTERVOLUME

このサービスは単一の gluster ボリュームを管理します。

表6.300 メソッドの概要

名前	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。

名前	概要
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。

6.104.1. 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6.301 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
volume	GlusterVolume	Out	gluster ボリュームを表します。

6.104.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.104.2. 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6.302 パラメーターの概要

名前	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。

6.104.3. 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6.303 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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	Boolean	In	リバランスを強制的に起動するかどうかを示します。

6.104.3.1. 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。

6.104.3.2. 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。

6.104.4. 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6.304 パラメーターの概要

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

6.104.5. 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6.305 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.104.6. 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6.306 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	In	
option	オプション	In	リセットするオプション。

6.104.7. 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6.307 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.104.8. 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6.308 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	In	ボリュームが強制的に起動するかどうかを示します。

6.104.8.1. 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。

6.104.9. 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6.309 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.104.10. 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6.310 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	In	

6.104.11. 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6.311 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.104.12. 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6.312 パラメーターの概要

名前	タイプ	方向	概要
----	-----	----	----

名前	タイプ	方向	概要
async	Boolean	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。

6.105. 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6.313 メソッドの概要

名前	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。

6.105.1. 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6.314 パラメーターの概要

名前	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。

6.105.2. 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
```

返されるボリューム一覧の順序は保証されません。

表6.315 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すボリュームの最大数を設定します。
search	文字列	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	

6.105.2.1. 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。

6.105.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.105.2.3. 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。

6.106. グループ

ユーザーのグループを管理します。このサービスを使用して、グループの詳細を取得したり、グループを削除します。新規グループを追加するには、グループのコレクションを管理する [service](#) を使用します。

表6.316 メソッドの概要

名前	概要
get	システムグループ情報を取得します。
remove	システムグループを削除します。

6.106.1. get GET

システムグループ情報を取得します。

使用方法

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

グループ情報を返します。

```
<group href="/ovirt-engine/api/groups/123" id="123">
  <name>mygroup</name>
  <link href="/ovirt-engine/api/groups/123/roles" rel="roles"/>
  <link href="/ovirt-engine/api/groups/123/permissions"/>
  <link href="/ovirt-engine/api/groups/123/tags" rel="tags"/>

  <domain_entry_id>476652557A382F67696B6D2B32762B37796E46476D513D3D</domain_entry_id
  >
  <namespace>DC=example,DC=com</namespace>
  <domain href="/ovirt-engine/api/domains/ABCDEF" id="ABCDEF">
    <name>myextension-authz</name>
  </domain>
</group>
```

表6.317 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
get	グループ	Out	システムグループ。

6.106.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.106.2. remove DELETE

システムグループを削除します。

使用方法

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

表6.318 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. グループ

ユーザーのグループのコレクションを管理します。

表6.319 メソッドの概要

名前	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	システム内のすべてのグループを一覧表示します。

6.107.1. 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6.320 パラメーターの概要

名前	タイプ	方向	概要
group	グループ	In/Out	追加するグループ。

6.107.2. list GET

システム内のすべてのグループを一覧表示します。

使用方法

```
GET /ovirt-engine/api/groups
```

グループの一覧を返します。

```
<groups>
  <group href="/ovirt-engine/api/groups/123" id="123">
    <name>mygroup</name>
    <link href="/ovirt-engine/api/groups/123/roles" rel="roles"/>
    <link href="/ovirt-engine/api/groups/123/permissions" rel="permissions"/>
    <link href="/ovirt-engine/api/groups/123/tags" rel="tags"/>

  <domain_entry_id>476652557A382F67696B6D2B32762B37796E46476D513D3D</domain_entry_id
  >
    <namespace>DC=example,DC=com</namespace>
    <domain href="/ovirt-engine/api/domains/ABCDEF" id="ABCDEF">
      <name>myextension-authz</name>
    </domain>
  </group>
  ...
</groups>
```

返されたグループリストの順序は保証されません。

表6.321 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
groups	Group[]	Out	グループの一覧。
max	Integer	In	返すグループの最大数を設定します。
search	文字列	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。

6.107.2.1. 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。

6.107.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.107.2.3. 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。

6.108. ホスト

ホストを管理するサービス

表6.322 メソッドの概要

名前	概要
activate	仮想マシンを実行するなど、使用するホストをアクティベートします。
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。
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。
copyhostnetworks	指定したホストのネットワーク設定を現在のホストにコピーします。
deactivate	ホストを無効にして、メンテナンスタスクを実行します。
discoveriscsi	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。
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 Storage Pool Manager (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	最新バージョンの 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	このメソッドは、Engine バージョン 4 以降非推奨となりました。
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。

名前	概要
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。
syncallnetworks	ホスト上の全ネットワークを同期するには、以下のような要求を送信します。 [source] ---- POST /ovirt-engine/api/hosts/123/syncallnetworks ---- リクエスト本文は以下のようになります。 [source,xml] ---- <action/> ----
unregisteredstoragedomainsdiscover	設定にインポートする候補であるブロックストレージドメインを検出します。
update	ホストのプロパティを更新します。
upgrade	ホストでVDSM および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。

6.108.1. activate POST

仮想マシンを実行するなど、使用するホストをアクティベートします。

表6.323 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.108.2. 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6.324 パラメーターの概要

名前	タイプ	方向	概要
activate	Boolean	In	true に設定すると、このホストは承認の完了後にアクティベートされます。
async	Boolean	In	承認を非同期的に実行するかどうかを指定します。

名前	タイプ	方向	概要
cluster	クラスター	In	ホストが承認後に追加されるクラスター。
ホスト	ホスト	In	承認するホスト。
reboot	Boolean	In	インストールに成功した後にホストを再起動するかどうかを示します。

6.108.2.1. activate

true に設定すると、このホストは承認の完了後にアクティベートされます。'false' に設定すると、ホストは承認後も 'maintenance' ステータスのままになります。このパラメーターがない場合は 'true' と解釈されます。これは、望ましいデフォルトの動作が承認後にホストをアクティブ化するからです。

6.108.2.2. reboot

インストールに成功した後にホストを再起動するかどうかを示します。デフォルト値は **true** です。

6.108.3. 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/>
```



重要

Red Hat Virtualization Manager 4.3 以降、[setupnetworks](#) リクエストで **commit_on_success** を指定することもできます。この場合、セットアップが完了し、{hypervisor-name} と Red Hat Virtualization Manager 間の接続が再確立され、個別の [commitnetconfig](#) リクエストを待つことなく、新しい設定が {hypervisor-name} に自動的に保存されます。

表6.325 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.108.4. copyhostnetworks POST

指定したホストのネットワーク設定を現在のホストにコピーします。



重要

ソースホストに存在しないネットワークアタッチメントは、コピー操作によってターゲットホストから消去されます。

別のホストからネットワークをコピーするには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

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

表6.326 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
source_host	ホスト	In	ネットワークのコピー元となるホスト。

6.108.5. deactivate POST

ホストを無効にして、メンテナンスタスクを実行します。

表6.327 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	文字列	In	

名前	タイプ	方向	概要
stop_gluster_service	Boolean	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。

6.108.5.1. 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操作中に使用できます。この変数のデフォルト値は **false** です。

6.108.6. discoveriscsi 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。検出されたデータを含む IscsiDetails オブジェクトの一覧を返します。

たとえば、**myiscsi.example.com** で利用可能な iSCSI ターゲットをホスト **123** から検出するには、次のようなリクエストを送信します。

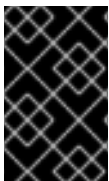
```
POST /ovirt-engine/api/hosts/123/discoveriscsi
```

リクエスト本文は以下のようになります。

```
<action>
  <iscsi>
    <address>myiscsi.example.com</address>
  </iscsi>
</action>
```

結果は以下のようになります。

```
<discovered_targets>
  <iscsi_details>
    <address>10.35.1.72</address>
    <port>3260</port>
    <portal>10.35.1.72:3260,1</portal>
    <target>iqn.2015-08.com.tgt:444</target>
  </iscsi_details>
</discovered_targets>
```



重要

このメソッドを使用して、iSCSI ターゲットを検出する場合には、FQDN または IP アドレスを使用できますが、REST API メソッドを使用してログインするには、検出された iSCSI ターゲットの詳細を使用する必要があります。

表6.328 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
discovered_targets	iscsiDetails[]	Out	すべての接続情報を含む検出されたターゲット。
iscsi	iscsiDetails[]	In	ターゲット iSCSI デバイス。

6.108.7. enrollcertificate POST

ホストの証明書を登録します。有効期限が近づいている、またはすでに有効期限が切れているという警告が表示された場合に便利です。

表6.329 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.108.8. 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6.330 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	文字列	In	
maintenance_after_restart	Boolean	In	再起動後にホストをメンテナンスする必要があるかどうかを示します。
power_management	PowerManagement	Out	

6.108.9. forceselectspm POST

ホストを Storage Pool Manager (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6.331パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.108.10. get GET

ホストの詳細を取得します。

GET /ovirt-engine/api/hosts/123

表6.332パラメーターの概要

名前	タイプ	方向	概要
all_content	Boolean	In	ホストのすべての属性を応答に含めるかどうかを指定します。
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
ホスト	ホスト	Out	クエリーされたホスト。

6.108.10.1. all_content

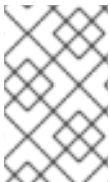
ホストのすべての属性を応答に含めるかどうかを指定します。

デフォルトでは、以下の属性が除外されます。

- **hosted_engine**

たとえば、ホスト '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/hosts/123?all_content=true
```



注記

これらの属性は、取得後にパフォーマンスに影響を及ぼすため、デフォルトでは含まれていません。これら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、特に必要な場合にのみ注意して使用してください。

6.108.10.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.108.11. install POST

最新バージョンの VDSM と関連ソフトウェアをホストにインストールします。

このアクションは、ホストをエンジンに追加するときに行われるホスト上のすべての設定ステップ (kdump 設定、ホスト型エンジンのデプロイ、カーネルオプションの変更など) も実行します。

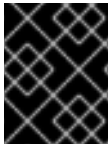
ホストタイプは、アクションの追加パラメーターを定義します。

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で **curl** と 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"
  "deploy_hosted_engine" : "true"
}
'\
"https://engine.example.com/ovirt-engine/api/hosts/123"
```



重要

エンジンのバージョン 4.1.2 以降、ホストが再インストールされると、デフォルトでホストファイアウォールの定義をオーバーライドするようになりました。

表6.333 パラメーターの概要

名前	タイプ	方向	概要
activate	Boolean	In	'true' に設定すると、このホストはインストールの完了後にアクティブになります。
async	Boolean	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。
deploy_hosted_engine	Boolean	In	true に設定すると、このホストは自己ホスト型エンジンコンポーネントもデプロイします。
ホスト	ホスト	In	override_iptables プロパティは、ファイアウォール設定をデフォルトの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	文字列	In	{hypervisor-name} をインストールする場合、ISO イメージファイルが必要です。
reboot	Boolean	In	インストールに成功した後にホストを再起動するかどうかを示します。
root_password	文字列	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。

名前	タイプ	方向	概要
ssh	Ssh	In	ホストへの接続に使用される SSH の詳細。
undeploy_hosted_engine	Boolean	In	true に設定すると、このホストは自己ホスト型エンジンコンポーネントをアンデプロイし、このホストは高可用性クラスターの一部として機能しなくなります。

6.108.11.1. activate

'true' に設定すると、このホストはインストールの完了後にアクティブになります。'false' に設定すると、ホストはインストール後も 'maintenance' ステータスのままになります。このパラメーターがない場合は 'true' と解釈されます。これは、望ましいデフォルトの動作がインストール後にホストをアクティブ化するからです。

6.108.11.2. deploy_hosted_engine

true に設定すると、このホストは自己ホスト型エンジンコンポーネントもデプロイします。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、セルフホスト型エンジン領域での操作は一切行われません。

6.108.11.3. reboot

インストールに成功した後にホストを再起動するかどうかを示します。デフォルト値は **true** です。

6.108.11.4. undeploy_hosted_engine

true に設定すると、このホストは自己ホスト型エンジンコンポーネントをアンデプロイし、このホストは高可用性クラスターの一部として機能しなくなります。欠落している値は **true**、つまりアン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、セルフホスト型エンジン領域での操作は一切行われません。

6.108.12. iscsidiscover POST

このメソッドは、Engine バージョン 4.4.6 以降非推奨となりました。代わりに DiscoverIscsi を使用する必要があります。

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。検出されたデータを含む文字列の配列を返します。

たとえば、**myiscsi.example.com** で利用可能な iSCSI ターゲットをホスト **123** から検出するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<action>
  <iscsi>
    <address>myiscsi.example.com</address>
```

```
</iscsi>
</action>
```

表6.334 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
iscsi_targets	String[]	Out	iSCSI ターゲット。

6.108.12.1. iscsi_targets

iSCSI ターゲット。*

6.108.13. 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。



重要

このメソッドを使用してログインする場合は、discoveriscsi メソッドで検出された iSCSI ターゲットの詳細を使用する必要があります。

表6.335 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.108.14. refresh POST

ホストデバイスと機能を更新します。

表6.336 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.108.15. 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6.337 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	削除を非同期的に実行するかどうかを指定します。
force	Boolean	In	ホストが応答しない場合、またはホストが Gluster Storage クラスターの一部であり、ボリュームブリックがある場合でも、ホストを削除する必要があることを示します。

6.108.16. 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** と **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>
      <ip_address_assignment>
        <assignment_method>static</assignment_method>
        <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"

```



注記

これは、API のバージョン 4 で有効です。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"/>
```

次のコードで Python SDK を使用しても、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。



重要

Red Hat Virtualization Manager 4.3 以降、[setupnetworks](#) リクエストで **commit_on_success** を指定することもできます。この場合、セットアップが完了し、`{hypervisor-name}` と Red Hat Virtualization Manager 間の接続が再確立され、個別の [commitnetconfig](#) リクエストを待つことなく、新しい設定が `{hypervisor-name}` に自動的に保存されます。

表6.338 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	In	
commit_on_success	Boolean	In	セットアップが完了し、 <code>{hypervisor-name}</code> と Red Hat Virtualization Manager 間の接続が再確立され、個別の commitnetconfig リクエストを待つことなく、設定を <code>{hypervisor-name}</code> に自動的に保存するかどうかを指定します。

名前	タイプ	方向	概要
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	同期されるネットワーク接続の一覧。

6.108.16.1. commit_on_success

セットアップが完了し、{hypervisor-name} と Red Hat Virtualization Manager 間の接続が再確立され、個別の `commitnetconfig` リクエストを待つことなく、設定を {hypervisor-name} に自動的に保存するかどうかを指定します。デフォルト値は **false** です。これは、設定が自動的に保存されないことを意味します。

6.108.17. syncallnetworks POST

ホスト上の全ネットワークを同期するには、以下のような要求を送信します。

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

リクエスト本文は以下のようになります。

```
<action/>
```

表6.339 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.108.18. unregisteredstoragedomainsdiscover POST

設定にインポートする候補であるブロックストレージドメインを検出します。FCP の場合は、引数は必要ありません。

表6.340 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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_domains	StorageDomainain[]	Out	

6.108.19. 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6.341 パラメーターの概要

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

6.108.20. upgrade POST

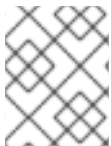
ホストで VDSM および選択したソフトウェアをアップグレードします。

表6.342 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
image	文字列	In	Vintage Node はサポートされなくなり、非推奨になったため、このプロパティは関連しなくなりました。
reboot	Boolean	In	アップグレード後にホストを再起動する必要があるかどうかを示します。
timeout	Integer	In	アップグレードのタイムアウト。

6.108.20.1. reboot

アップグレード後にホストを再起動する必要があるかどうかを示します。デフォルトでは、ホストは再起動されます。



注記

{hypervisor-name} の場合、このパラメーターは無視されます。{hypervisor-name} は、アップグレード後に常に再起動されます。

6.108.20.2. timeout

アップグレードのタイムアウト。

アップグレードが完了するまでの最大待機時間 (分単位)。デフォルト値は **ANSIBLE_PLAYBOOK_EXEC_DEFAULT_TIMEOUT** 設定オプションで指定します。

6.108.21. 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、管理ポータル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。

6.109. HOSTCPUUNITS

表6.343 メソッドの概要

名前	概要
list	トポロジー (ソケット、コア) に関する詳細情報と、現在の CPU ピン留めに関する情報を含むすべてのホストの CPU の一覧を返します。

6.109.1. list GET

トポロジー (ソケット、コア) に関する詳細情報と、現在の CPU ピン留めに関する情報を含むすべてのホストの CPU の一覧を返します。

表6.344 パラメーターの概要

名前	タイプ	方向	概要
cpu_units	HostCpuUnit []	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.109.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.110. 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6.345 メソッドの概要

名前	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。

6.110.1. 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6.346 パラメーターの概要

名前	タイプ	方向	概要
device	HostDevice	Out	

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.110.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.111. HOSTDEVICES

ホストデバイスにアクセスするためのサービス。

表6.347 メソッドの概要

名前	概要
list	ホストのデバイスを一覧表示します。

6.111.1. list GET

ホストのデバイスを一覧表示します。

返されるデバイス一覧の順序は保証されません。

表6.348 パラメーターの概要

名前	タイプ	方向	概要
devices	HostDevice[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すデバイスの最大数を設定します。

6.111.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.111.1.2. 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。

6.112. HOSTHOOK

表6.349 メソッドの概要

名前	概要
get	

6.112.1. get GET

表6.350 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
hook	Hook	Out	

6.112.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.113. HOSTHOOKS

表6.351 メソッドの概要

名前	概要
list	ホスト用に設定されたフックの一覧を返します。

6.113.1. list GET

ホスト用に設定されたフックの一覧を返します。

返されるフックのリストの順序はランダムです。

表6.352 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
hooks	Hook[]	Out	
max	Integer	In	返すフックの最大数を設定します。

6.113.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.113.1.2. 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。

6.114. 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6.353 メソッドの概要

名前	概要
<code>get</code>	
<code>updatevirtualfunctionsconfiguration</code>	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。

6.114.1. get GET

表6.354 パラメーターの概要

名前	タイプ	方向	概要
<code>all_content</code>	Boolean	In	ホストネットワークインターフェイス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。
<code>follow</code>	文字列	In	たどる 必要のある内部リンクを指定します。
<code>nic</code>	HostNic	Out	

6.114.1.1. all_content

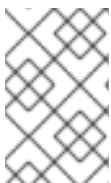
ホストネットワークインターフェイスのすべての属性を応答に含める必要があるかどうかを示します。

デフォルトでは、以下の属性が除外されます。

- `virtual_functions_configuration`

たとえば、ホスト '123' の完全な表現ネットワークインターフェイス '456' を取得するには、以下を実行します。

```
GET /ovirt-engine/api/hosts/123/nics/456?all_content=true
```



注記

これらの属性は、取得後にパフォーマンスに影響を及ぼすため、デフォルトでは含まれていません。これら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、特に必要な場合にのみ注意して使用してください。

6.114.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.114.2. 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。入力は、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6.355 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.115. 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6.356 メソッドの概要

名前	概要
list	ホストのネットワークインターフェイスの一覧を返します。

6.115.1. list GET

ホストのネットワークインターフェイスの一覧を返します。

返されるネットワークインターフェイスのリストの順序は保証されません。

表6.357 パラメーターの概要

名前	タイプ	方向	概要
all_content	Boolean	In	ホストネットワークインターフェイス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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	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	

6.115.1.1. all_content

ホストネットワークインターフェイス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。

- **virtual_functions_configuration**

たとえば、ホスト '123' のネットワークインターフェイス '456' 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/hosts/123/nics?all_content=true
```



注記

これらの属性は、取得後にパフォーマンスに影響を及ぼすため、デフォルトでは含まれていません。これら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、特に必要な場合にのみ注意して使用してください。

6.115.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.115.1.3. 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。

6.116. HOSTNUMANODE

表6.358 メソッドの概要

名前	概要
get	

6.116.1. get GET

表6.359 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
node	NumaNode	Out	

6.116.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.117. HOSTNUMANODES

表6.360 メソッドの概要

名前	概要
list	ホストの NUMA ノードの一覧を返します。

6.117.1. list GET

ホストの NUMA ノードの一覧を返します。

返される NUMA ノードのリストの順序は保証されません。

表6.361 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すノードの最大数を設定します。
nodes	NumaNode[]	Out	

6.117.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.117.1.2. 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。

6.118. HOSTSTORAGE

ホストストレージを管理するサービス。

表6.362 メソッドの概要

名前	概要
list	ストレージのリストを取得します。

6.118.1. 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>
```

返されるストレージのリストの順番は保証されていません。

表6.363 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
report_status	Boolean	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。

6.118.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.118.1.2. 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>

```

6.119. ホスト

ホストを管理するサービス。

表6.364 メソッドの概要

名前	概要
add	新しいホストを作成します。
list	利用可能なすべてのホストのリストを取得します。

6.119.1. 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。**name**、**address**、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、次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

エンジンのバージョン 4.1.2 以降、ホストが新しく追加されると、デフォルトでホストのファイアウォール定義がオーバーライドされるようになりました。

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

クラスターに、自動デプロイメントがサポートされているデフォルトの外部ネットワークプロバイダーがある場合、ホストの追加時に外部ネットワークプロバイダーがデプロイされます。自動デプロイメントでは、OVN の外部ネットワークプロバイダーのみがサポートされます。クラスターで定義されているもの以外の外部ネットワークプロバイダーをデプロイするには、ホストを追加するときに以下のリクエストを送信して、外部ネットワークプロバイダーを上書きします。

POST /ovirt-engine/api/hosts

external_network_provider_configuration に必要なプロバイダーへの参照が含まれるリクエスト本文:

```
<host>
  <name>myhost</name>
  <address>myhost.example.com</address>
  <root_password>123456</root_password>
  <external_network_provider_configurations>
    <external_network_provider_configuration>
      <external_network_provider name="ovirt-provider-ovn"/>
    </external_network_provider_configuration>
  </external_network_provider_configurations>
</host>
```

表6.365 パラメーターの概要

名前	タイプ	方向	概要
----	-----	----	----

名前	タイプ	方向	概要
activate	Boolean	In	true に設定すると、このホストはインストールの完了後にアクティブになります。
deploy_hosted_engine	Boolean	In	true に設定すると、このホストはホスト型エンジンコンポーネントをデプロイします。
ホスト	ホスト	In/Out	新しいホストが作成され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。
reboot	Boolean	In	インストールに成功した後にホストを再起動するかどうかを示します。
undeploy_hosted_engine	Boolean	In	true に設定すると、このホストはホスト型エンジンコンポーネントをアンデプロイし、高可用性クラスターの一部として機能しません。

6.119.1.1. activate

true に設定すると、このホストはインストールの完了後にアクティブになります。**false** に設定すると、ホストはインストール後も **maintenance** ステータスのままになります。このパラメーターがない場合は **true** と解釈されます。これは、望ましいデフォルトの動作がインストール後にホストをアクティブ化するからです。

6.119.1.2. deploy_hosted_engine

true に設定すると、このホストはホスト型エンジンコンポーネントをデプロイします。欠落している値は **true** として扱われます。つまり、ホスト型エンジンコンポーネントをデプロイします。このパラメーターを省略すると **false** になり、ホストはホスト型エンジン領域で操作を実行しません。

6.119.1.3. reboot

インストールに成功した後にホストを再起動するかどうかを示します。デフォルト値は **true** です。

6.119.1.4. undeploy_hosted_engine

true に設定すると、このホストはホスト型エンジンコンポーネントをアンデプロイし、高可用性クラスターの一部として機能しません。欠落した値は **true**、すなわちアン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** になり、ホストはホスト型エンジン領域で操作を実行しません。

6.119.2. 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
```

応答本文は次のようになります。

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

返されるホストのリストの順序は、**sortby** 句が **search** パラメーターに含まれている場合にのみ保証されます。

表6.366 パラメーターの概要

名前	タイプ	方向	概要
all_content	Boolean	In	ホスト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	Boolean	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。
check_vms_in_affinity_closure	Boolean	In	このパラメーターを migration_target_of とともに使用して、リストされた仮想マシン、およびリストされた仮想マシンとの親和性を強化している他のすべての仮想マシンの有効な移行ターゲットを取得できます。
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
hosts	Host[]	Out	
max	Integer	In	返すホストの最大数を設定します。
migration_target_of	文字列	In	仮想マシン ID のコンマ区切りリストを受け入れ、これらの仮想マシンを移行できるホストを返します。
search	文字列	In	返されたホストを制限するために使用されるクエリー文字列。

6.119.2.1. all_content

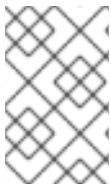
ホストのすべての属性を応答に含める必要があるかどうかを示します。

デフォルトでは、以下のホスト属性が除外されます。

- **hosted_engine**

たとえば、ホスト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/hosts?all_content=true
```



注記

これらの属性は、取得後にパフォーマンスに影響を及ぼすため、デフォルトでは含まれていません。これら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、特に必要な場合にのみ注意して使用してください。

6.119.2.2. 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。

6.119.2.3. check_vms_in_affinity_closure

このパラメーターを **migration_target_of** とともに使用して、リストされた仮想マシン、およびリストされた仮想マシンとの親和性を強化している他のすべての仮想マシンの有効な移行ターゲットを取得できます。

これは、仮想マシンがポジティブアフィニティグループ内の他のマシンと一緒に移行される場合に役立ちます。

デフォルト値は **false** です。

```
GET /ovirt-engine/api/hosts?migration_target_of=123,456&check_vms_in_affinity_closure=true
```

6.119.2.4. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.119.2.5. 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。

6.119.2.6. migration_target_of

仮想マシン ID のコンマ区切りリストを受け入れ、これらの仮想マシンを移行できるホストを返します。

たとえば、ID 123 の仮想マシンと ID 456 の仮想マシンの移行先ホストの一覧を取得するには、以下のリクエストを送信します。

```
GET /ovirt-engine/api/hosts?migration_target_of=123,456
```

6.120. ICON

アイコンを管理するサービス (読み取り専用)。

表6.367 メソッドの概要

名前	概要
get	アイコンを取得します。

6.120.1. 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6.368 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
icon	Icon	Out	取得したアイコン。

6.120.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.121. ICONS

アイコンを管理するサービス。

表6.369 メソッドの概要

名前	概要
list	アイコンのリストを取得します。

6.121.1. 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>

```

返されるアイコン一覧の順序は保証されません。

表6.370 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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。

6.121.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.121.1.2. 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。

6.122. イメージ

表6.371 メソッドの概要

名前	概要
get	
import	イメージをインポートします。

6.122.1. get GET

表6.372 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
image	イメージ	Out	

6.122.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.122.2. 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す。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6.373 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	クラスター	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	Boolean	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。

名前	タイプ	方向	概要
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。

6.123. 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、[イメージ転送](#) サービスの [追加](#) を使用し、データを転送するイメージを指定して転送を作成する必要があります。

その後、転送はこのサービスによって管理されます。

oVirt の Python の SDK を使用する:

ID が **123** の **disk** をアップロードします (データセンター内のランダムなホスト上に)。

```
transfers_service = system_service.image_transfers_service()
transfer = transfers_service.add(
    types.ImageTransfer(
        disk=types.Disk(
            id='123'
        )
    )
)
```

ID が **456** の **host** に ID が **123** の **disk** をアップロードします。

```
transfers_service = system_service.image_transfers_service()
transfer = transfers_service.add(
    types.ImageTransfer(
        disk=types.Disk(
            id='123'
        ),
        host=types.Host(
            id='456'
        )
    )
)
```

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(
        disk=types.Disk(
            id='123'
        ),

```



```

        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。

以下はその例です。

```

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。これは、選択したホストで `ovirt-imageio` が実行されていない場合に発生する可能性があります。

表6.374 メソッドの概要

名前	概要
cancel	イメージ転送セッションをキャンセルします。
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。

6.123.1. cancel POST

イメージ転送セッションをキャンセルします。これにより、転送操作が終了し、部分的なイメージが削除されます。

6.123.2. extend POST

イメージ転送セッションを延長します。

6.123.3. 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。

6.123.4. get GET

イメージ転送エンティティーを取得します。

表6.375 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
image_transfer	ImageTransfer	Out	

6.123.4.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.123.5. pause POST

イメージ転送セッションを一時停止します。

6.123.6. 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。以下はその例です。

```
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()
```

6.124. IMAGETRANSFERS

これは、Red Hat Virtualization の Image I/O API を実行するためのイメージ転送を管理するサービスです。詳細は、[イメージ転送](#)を参照してください。

表6.376 メソッドの概要

名前	概要
add	新しいイメージ転送を追加します。

名前	概要
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。

6.124.1. 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、ディスク、またはディスクのスナップショットを指定する必要があります。



重要

image 属性は、エンジンのバージョン 4.2 以降非推奨になりました。代わりに、**disk** または **snapshot** の属性を使用してください。

disk をダウンロードまたはアップロードするための新しいイメージ転送の作成:

ID **123** のディスクをダウンロードまたはアップロードするためのイメージ転送を作成するには、次のリクエストを送信します。

```
POST /ovirt-engine/api/imagetransfers
```

リクエスト本文は以下のようになります。

```
<image_transfer>
  <disk id="123"/>
  <direction>upload|download</direction>
</image_transfer>
```

disk_snapshot をダウンロードまたはアップロードするための新しいイメージ転送の作成:

ID **456** の **disk_snapshot** をダウンロードまたはアップロードするためのイメージ転送を作成するには、次のリクエストを送信します。

```
POST /ovirt-engine/api/imagetransfers
```

リクエスト本文は以下のようになります。

```
<image_transfer>
  <snapshot id="456"/>
  <direction>download|upload</direction>
</image_transfer>
```

表6.377 パラメーターの概要

名前	タイプ	方向	概要
image_transfer	ImageTransfer	In/Out	追加するイメージ転送。

6.124.2. 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。

返されるイメージ転送一覧の順序は保証されません。

表6.378 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
image_transfer	ImageTransfer[]	Out	現在実行されているイメージ転送のリスト。

6.124.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.125. IMAGES

ストレージドメインまたは OpenStack イメージプロバイダーで利用可能なイメージのセットを管理します。

表6.379 メソッドの概要

名前	概要
list	ストレージドメインまたはプロバイダーで使用可能なイメージの一覧を返します。

6.125.1. list GET

ストレージドメインまたはプロバイダーで使用可能なイメージの一覧を返します。

返されるイメージ一覧の順序は保証されません。

表6.380 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
images	Image[]	Out	
max	Integer	In	返すイメージの最大数を設定します。

6.125.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.125.1.2. 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。

6.126. INSTANCETYPE

表6.381 メソッドの概要

名前	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。

6.126.1. 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6.382 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
instance_type	InstanceType	Out	

6.126.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.126.2. 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6.383 パラメーターの概要

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

6.126.3. 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/instancetype/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6.384 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.127. INSTANCETYPEGRAPHICSCONSOLE

表6.385 メソッドの概要

名前	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。

6.127.1. 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6.386 パラメーターの概要

名前	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.127.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.127.2. 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6.387 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. INSTANCETYPEGRAPHICSCONSOLES

表6.388 メソッドの概要

名前	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。

6.128.1. 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6.389 パラメーターの概要

名前	タイプ	方向	概要
console	GraphicsConsole	In/Out	

6.128.2. 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。

返されるグラフィックコンソール一覧の順序は保証されません。

表6.390 パラメーターの概要

名前	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すコンソールの最大数を設定します。

6.128.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.128.2.2. 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。

6.129. INSTANCETYPENIC

表6.391 メソッドの概要

名前	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。

6.129.1. 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6.392 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
nic	Nic	Out	

6.129.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.129.2. 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6.393 パラメーターの概要

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

6.129.3. 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6.394 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.130. INSTANCETYPENICS

表6.395 メソッドの概要

名前	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。
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。

6.130.1. 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6.396 パラメーターの概要

名前	タイプ	方向	概要
nic	Nic	In/Out	

6.130.2. 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。

返されるネットワークインターフェイスのリストの順序は保証されません。

表6.397 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す NIC の最大数を設定します。
nics	Nic[]	Out	
search	文字列	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。

6.130.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.130.2.2. 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。

6.131. INSTANCETYPEWATCHDOG

表6.398 メソッドの概要

名前	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。

6.131.1. 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6.399 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
watchdog	Watchdog	Out	

6.131.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.131.2. 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6.400 パラメーターの概要

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

6.131.3. 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6.401 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.132. INSTANCETYPEWATCHDOGS

表6.402 メソッドの概要

名前	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。

6.132.1. 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6.403 パラメーターの概要

名前	タイプ	方向	概要
watchdog	Watchdog	In/Out	

6.132.2. 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。

返されるウォッチドッグ一覧の順序は保証されません。

表6.404 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すウォッチドッグの最大数を設定します。
search	文字列	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	

6.132.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.132.2.2. 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。

6.133. INSTANCETYPES

表6.405 メソッドの概要

名前	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。

6.133.1. 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/instancetypees
```

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6.406 パラメーターの概要

名前	タイプ	方向	概要
instance_type	InstanceType	In/Out	

6.133.2. 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。

返されるインスタンスタイプ一覧の順序は保証されません。

表6.407 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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	文字列	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。

6.133.2.1. 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。

6.133.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.133.2.3. 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。

6.134. ISCSIBOND

表6.408 メソッドの概要

名前	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。

6.134.1. get GET

表6.409 パラメーターの概要

名前	タイプ	方向	概要
bond	IscsiBond	Out	iSCSI ボンディング。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.134.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.134.2. 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6.410 パラメーターの概要

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

6.134.3. 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6.411 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	更新する iSCSI ボンディング。

6.135. ISCSIBONDS

表6.412 メソッドの概要

名前	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	データセンターで設定されている iSCSI ボンディングの一覧を返します。

6.135.1. 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6.413 パラメーターの概要

名前	タイプ	方向	概要
bond	IscsiBond	In/Out	

6.135.2. list GET

データセンターで設定されている iSCSI ボンディングの一覧を返します。

返される iSCSI ボンディング一覧の順序は保証されません。

表6.414 パラメーターの概要

名前	タイプ	方向	概要
bonds	IscsiBond[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すボンディングの最大数を設定します。

6.135.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.135.2.2. 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。

6.136. JOB

ジョブを管理するサービス

表6.415 メソッドの概要

名前	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。

6.136.1. 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6.416 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.136.2. 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6.417 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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	Boolean	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。

6.136.2.1. 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。

6.136.3. 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6.418 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
job	Job	Out	ジョブの表現を取得します。

6.136.3.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.137. ジョブ

ジョブを管理するサービス。

表6.419 メソッドの概要

名前	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。

6.137.1. 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6.420 パラメーターの概要

名前	タイプ	方向	概要
job	Job	In/Out	追加されるジョブ。

6.137.2. 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>

```

返されるジョブ一覧の順序は保証されません。

表6.421 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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。
search	文字列	In	返されたホストを制限するために使用されるクエリー文字列。

6.137.2.1. 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。

6.137.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.137.2.3. 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。

6.138. 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6.422 メソッドの概要

名前	概要
list	Katello エラータの表現を取得します。

6.138.1. 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>
```

返されたエラーター一覧の順序は保証されません。

表6.423 パラメーターの概要

名前	タイプ	方向	概要
errata	KatelloErratum[]	Out	Katello エラータの表現。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すエラータの最大数を設定します。

6.138.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.138.1.2. 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。

6.139. KATELLOERRATUM

Katello エラータを管理するサービスです。

表6.424 メソッドの概要

名前	概要
get	Katello エラータを取得します。

6.139.1. 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6.425 パラメーターの概要

名前	タイプ	方向	概要
erratum	KatelloErratum	Out	Katello エラータの表現を取得します。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.139.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.140. LINKLAYERDISCOVERYPROTOCOL

Link Layer Discovery Protocol (LLDP) によって受信された情報要素を取得するサービス。

表6.426 メソッドの概要

名前	概要
list	LLDP によって受信された情報要素を取得します。

6.140.1. list GET

LLDP によって受信された情報要素を取得します。

表6.427 パラメーターの概要

名前	タイプ	方向	概要
element	LinkLayerDiscoveryProtocolElement[]	Out	LLDP が受信した情報要素のリストを取得します。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.140.1.1. element

LLDP が受信した情報要素のリストを取得します。

たとえば、ホスト **123** の NIC **321** で受信した情報要素を取得するには、以下のようなリクエストを送信します。

```
GET ovirt-engine/api/hosts/123/nics/321/linklayerdiscoveryprotocolelements
```

以下のような応答が返されます。

```
<link_layer_discovery_protocol_elements>
...
<link_layer_discovery_protocol_element>
  <name>Port Description</name>
  <properties>
    <property>
      <name>port description</name>
      <value>Summit300-48-Port 1001</value>
    </property>
  </properties>
  <type>4</type>
</link_layer_discovery_protocol_element>
...
<link_layer_discovery_protocol_elements>
```

6.140.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.141. MACPOOL

表6.428 メソッドの概要

名前	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。

6.141.1. get GET

表6.429 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
pool	MacPool	Out	

6.141.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.141.2. 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6.430 パラメーターの概要

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

6.141.3. 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6.431 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.142. MACPOOLS

表6.432 メソッドの概要

名前	概要
add	新しい MAC アドレスプールを作成します。
list	システムの MAC アドレスプールの一覧を返します。

6.142.1. 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、次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6.433 パラメーターの概要

名前	タイプ	方向	概要
pool	MacPool	In/Out	

6.142.2. list GET

システムの MAC アドレスプールの一覧を返します。

返される MAC アドレスプールのリストは保証されません。

表6.434 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプールの最大数を設定します。
pools	MacPool[]	Out	

6.142.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.142.2.2. 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。

6.143. MEASURABLE

6.144. MOVEABLE

表6.435 メソッドの概要

名前	概要
move	

6.144.1. move POST

表6.436 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.145. NETWORK

ネットワークを管理するサービス

表6.437 メソッドの概要

名前	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。

6.145.1. get GET

論理ネットワークを取得します。

以下はその例です。

```
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6.438 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
network	Network	Out	

6.145.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.145.2. 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
```



注記

外部論理ネットワークを削除するには、[OpenStack Networking API](#) を使用して、そのプロバイダーから直接ネットワークを削除する必要があります。プロバイダーで **auto_sync** が有効になっている場合、Red Hat Virtualization 内の外部ネットワークを表すエンティティは自動的に削除されます。それ以外の場合は、このメソッドを使用してエンティティを削除する必要があります。

表6.439 パラメーターの概要

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

6.145.3. 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>
```

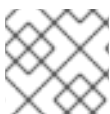
ネットワークの最大伝送単位は、**mtu** 属性の整数値を指定する PUT 要求を使用して設定されます。

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>
```



注記

外部ネットワークの更新はプロバイダーに伝播されません。

表6.440 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.146. NETWORKATTACHMENT

表6.441 メソッドの概要

名前	概要
get	
remove	
update	ホスト上の指定されたネットワークアタッチメントを更新します。

6.146.1. get GET

表6.442 パラメーターの概要

名前	タイプ	方向	概要
attachment	NetworkAttachment	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.146.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.146.2. remove DELETE

表6.443 パラメーターの概要

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

6.146.3. update PUT

ホスト上の指定されたネットワークアタッチメントを更新します。

表6.444 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.147. NETWORKATTACHMENTS

ホストまたはホスト NIC のネットワークアタッチメントのセットを管理します。

表6.445 メソッドの概要

名前	概要
add	ネットワークインターフェイスに新しいネットワークアタッチメントを追加します。
list	ホストまたはホスト NIC のネットワークアタッチメントの一覧を返します。

6.147.1. add POST

ネットワークインターフェイスに新しいネットワークアタッチメントを追加します。

表6.446 パラメーターの概要

名前	タイプ	方向	概要
attachment	NetworkAttachment	In/Out	

6.147.2. list GET

ホストまたはホスト NIC のネットワークアタッチメントの一覧を返します。

返されるネットワークアタッチメント一覧の順序は保証されません。

表6.447 パラメーターの概要

名前	タイプ	方向	概要
attachments	NetworkAttachment[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すアタッチメントの最大数を設定します。

6.147.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.147.2.2. 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。

6.148. 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6.448 メソッドの概要

名前	概要
<code>get</code>	ネットワークフィルターの表現を取得します。

6.148.1. get GET

ネットワークフィルターの表現を取得します。

表6.449 パラメーターの概要

名前	タイプ	方向	概要
<code>follow</code>	文字列	In	たどる 必要のある内部リンクを指定します。
<code>network_filter</code>	NetworkFilter	Out	

6.148.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.149. 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6.450 メソッドの概要

名前	概要
list	ネットワークフィルターの表現を取得します。

6.149.1. list GET

ネットワークフィルターの表現を取得します。

返されるネットワークフィルター一覧の順序は保証されません。

表6.451 パラメーターの概要

名前	タイプ	方向	概要
filters	NetworkFilter[]	Out	

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.149.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.150. NETWORKLABEL

表6.452 メソッドの概要

名前	概要
get	
remove	論理ネットワークからラベルを削除します。

6.150.1. get GET

表6.453 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
label	NetworkLabel	Out	

6.150.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.150.2. 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/networklabels/exemplary
```

表6.454 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. NETWORKLABELS

ネットワークまたはホスト NIC にアタッチされたラベルの `ser` を管理します。

表6.455 メソッドの概要

名前	概要
add	論理ネットワークにラベルをアタッチします。
list	ネットワークまたはホスト NIC にアタッチされたラベルの一覧を返します。

6.151.1. 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/networklabels
```

リクエスト本文は以下のようになります。

```
<network_label id="mylabel"/>
```

表6.456 パラメーターの概要

名前	タイプ	方向	概要
label	NetworkLabel	In/Out	

6.151.2. list GET

ネットワークまたはホスト NIC にアタッチされたラベルの一覧を返します。

返されるラベル一覧の順序は保証されません。

表6.457 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
labels	NetworkLabel[]	Out	
max	Integer	In	返すラベルの最大数を設定します。

6.151.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.151.2.2. 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。

6.152. NETWORKS

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6.458 メソッドの概要

名前	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。

6.152.1. 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>
```

外部 **OpenStack** ネットワークプロバイダー **456** の上に **exnetwork** という名前のネットワークを作成するには、以下のようなリクエストを送信します。

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

```
<network>
  <name>exnetwork</name>
  <external_provider id="456"/>
  <data_center id="123"/>
</network>
```

表6.459 パラメーターの概要

名前	タイプ	方向	概要
network	Network	In/Out	

6.152.2. list GET

論理ネットワークを一覧表示します。

以下はその例です。

```
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>

```

返されるネットワークのリストの順序は、**sortby** 句が **search** パラメーターに含まれている場合にのみ保証されます。

表6.460 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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	文字列	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。

6.152.2.1. 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。

6.152.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.152.2.3. 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。

6.153. NICNETWORKFILTERPARAMETER

このサービスは、ネットワークフィルターのパラメーターを管理します。

表6.461 メソッドの概要

名前	概要
get	ネットワークフィルターパラメーターの表現を取得します。

名前	概要
remove	フィルターパラメーターを削除します。
update	ネットワークフィルターパラメーターを更新します。

6.153.1. get GET

ネットワークフィルターパラメーターの表現を取得します。

表6.462 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
parameter	NetworkFilterParameter[]	Out	ネットワークフィルターパラメーターの表現。

6.153.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.153.2. remove DELETE

フィルターパラメーターを削除します。

たとえば、仮想マシン **789** の NIC **456** で ID **123** を持つフィルターパラメーターを削除するには、以下のような要求を送信します。

```
DELETE /ovirt-engine/api/vms/789/nics/456/networkfilterparameters/123
```

6.153.3. update PUT

ネットワークフィルターパラメーターを更新します。

たとえば、仮想マシン **789** の NIC **456** で ID **123** を持つネットワークフィルターパラメーターを更新するには、以下のような要求を送信します。

```
PUT /ovirt-engine/api/vms/789/nics/456/networkfilterparameters/123
```

リクエスト本文は以下のようになります。

```
<network_filter_parameter>
  <name>updatedName</name>
  <value>updatedValue</value>
</network_filter_parameter>
```

表6.463 パラメーターの概要

名前	タイプ	方向	概要
parameter	NetworkFilterParameter[]	In/Out	更新中のネットワークフィルターパラメーター。

6.154. NICNETWORKFILTERPARAMETERS

このサービスは、ネットワークフィルターのパラメーターのコレクションを管理します。

表6.464 メソッドの概要

名前	概要
add	ネットワークフィルターパラメーターを追加します。
list	ネットワークフィルターパラメーターの表現を取得します。

6.154.1. add POST

ネットワークフィルターパラメーターを追加します。

たとえば、仮想マシン **789** の NIC **456** にネットワークフィルターのパラメーターを追加するには、以下のような要求を送信します。

```
POST /ovirt-engine/api/vms/789/nics/456/networkfilterparameters
```

リクエスト本文は以下のようになります。

```
<network_filter_parameter>
  <name>IP</name>
  <value>10.0.1.2</value>
</network_filter_parameter>
```

表6.465 パラメーターの概要

名前	タイプ	方向	概要
parameter	NetworkFilterParameter[]	In/Out	追加中のネットワークフィルターパラメーター。

6.154.2. list GET

ネットワークフィルターパラメーターの表現を取得します。

返されるネットワークフィルター一覧の順序は保証されません。

表6.466 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
parameters	NetworkFilterParameter[]	Out	ネットワークフィルターパラメーターのリスト。

6.154.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.155. OPENSTACKIMAGE

表6.467 メソッドの概要

名前	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。

6.155.1. get GET

表6.468 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
image	Openstackimage	Out	

6.155.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.155.2. 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ようになります。

```

<action>
  <storage_domain>
    <name>images0</name>
  </storage_domain>
  <cluster>
    <name>images0</name>
  </cluster>
</action>

```

表6.469 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	クラスター	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	Boolean	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	

6.156. OPENSTACKIMAGEPROVIDER

表6.470 メソッドの概要

名前	概要
get	
importcertificates	外部ホストプロバイダーの SSL 証明書をインポートします。
remove	
testconnectivity	外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。
update	システム内の指定された OpenStack イメージプロバイダーを更新します。

6.156.1. get GET

表6.471 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
provider	OpenStack Image Provider	Out	

6.156.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.156.2. importcertificates POST

外部ホストプロバイダーの SSL 証明書をインポートします。

表6.472 パラメーターの概要

名前	タイプ	方向	概要
certificates	Certificate[]	In	

6.156.3. remove DELETE

表6.473 パラメーターの概要

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

6.156.4. testconnectivity POST

外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。

POST /ovirt-engine/api/externalhostproviders/123/testconnectivity

表6.474 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.156.5. update PUT

システム内の指定された OpenStack イメージプロバイダーを更新します。

表6.475 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.157. OPENSTACKIMAGEPROVIDERS

表6.476 メソッドの概要

名前	概要
add	新しい OpenStack イメージプロバイダーをシステムに追加します。
list	プロバイダーの一覧を返します。

6.157.1. add POST

新しい OpenStack イメージプロバイダーをシステムに追加します。

表6.477 パラメーターの概要

名前	タイプ	方向	概要
provider	OpenStackImageProvider	In/Out	

6.157.2. list GET

プロバイダーの一覧を返します。

返されるプロバイダー一覧の順序は保証されません。

表6.478 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプロバイダーの最大数を設定します。

名前	タイプ	方向	概要
providers	OpenStackImageProvider[]	Out	
search	文字列	In	返される OpenStack イメージプロバイダーを制限するために使用されるクエリー文字列。

6.157.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.157.2.2. 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。

6.158. OPENSTACKIMAGES

表6.479 メソッドの概要

名前	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。

6.158.1. 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。

返されるイメージ一覧の順序は保証されません。

表6.480 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
images	OpenStackImage[]	Out	
max	Integer	In	返すイメージの最大数を設定します。

6.158.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.158.1.2. 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。

6.159. OPENSTACKNETWORK

表6.481 メソッドの概要

名前	概要
get	
import	この操作は、外部ネットワークを Red Hat Virtualization にインポートします。

6.159.1. get GET

表6.482 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
network	OpenStackNetwork	Out	

6.159.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.159.2. import POST

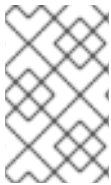
この操作は、外部ネットワークを Red Hat Virtualization にインポートします。指定したデータセンターにネットワークが追加されます。

表6.483 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.159.2.1. 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。



注記

プロバイダーで **auto_sync** が有効になっている場合、ネットワークが自動的にインポートされる可能性があります。これを防ぐには、**auto_sync** を `false` に設定して自動インポートを無効にし、ネットワークのインポート後に再度有効にすることができます。

6.160. 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6.484 メソッドの概要

名前	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。
importcertificates	外部ホストプロバイダーの SSL 証明書をインポートします。
remove	プロバイダーを削除します。
testconnectivity	外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。
update	プロバイダーを更新します。

6.160.1. 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6.485 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
provider	OpenStackNetworkProvider	Out	

6.160.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.160.2. importcertificates POST

外部ホストプロバイダーの SSL 証明書をインポートします。

表6.486 パラメーターの概要

名前	タイプ	方向	概要
certificates	Certificate[]	In	

6.160.3. 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6.487 パラメーターの概要

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

6.160.4. testconnectivity POST

外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。

```
POST /ovirt-engine/api/externalhostproviders/123/testconnectivity
```

表6.488 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.160.5. 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6.489 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.161. 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6.490 メソッドの概要

名前	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	プロバイダーの一覧を返します。

6.161.1. 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6.491 パラメーターの概要

名前	タイプ	方向	概要
provider	OpenStackNetworkProvider	In/Out	

6.161.2. list GET

プロバイダーの一覧を返します。

返されるプロバイダー一覧の順序は保証されません。

表6.492 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプロバイダーの最大数を設定します。
providers	OpenStackNetworkProvider[]	Out	
search	文字列	In	返される OpenStack ネットワークプロバイダーを制限するために使用されるクエリー文字列。

6.161.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.161.2.2. 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。

6.162. OPENSTACKNETWORKS

表6.493 メソッドの概要

名前	概要
list	ネットワークの一覧を返します。

6.162.1. list GET

ネットワークの一覧を返します。

返されるネットワーク一覧の順序は保証されません。

表6.494 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すネットワークの最大数を設定します。
networks	OpenStackNetwork[]	Out	

6.162.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.162.1.2. 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。

6.163. OPENSTACKSUBNET

表6.495 メソッドの概要

名前	概要
get	
remove	

6.163.1. get GET

表6.496 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
subnet	OpenStackSubnet	Out	

6.163.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.163.2. remove DELETE

表6.497 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. OPENSTACKSUBNETS

表6.498 メソッドの概要

名前	概要
add	
list	サブネットワークの一覧を返します。

6.164.1. add POST

表6.499 パラメーターの概要

名前	タイプ	方向	概要
subnet	OpenStackS ubnet	In/Out	

6.164.2. list GET

サブネットワークの一覧を返します。

返されるサブネットワーク一覧の順序は保証されません。

表6.500 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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	

6.164.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.164.2.2. 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。

6.165. OPENSTACKVOLUMEAUTHENTICATIONKEY

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表6.501 メソッドの概要

名前	概要
get	
remove	
update	指定された認証キーを更新します。

6.165.1. get GET

表6.502 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
鍵 (key)	OpenstackVolumeAuthenticationKey	Out	

6.165.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.165.2. remove DELETE

表6.503 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. update PUT

指定された認証キーを更新します。

表6.504 パラメーターの概要

名前	タイプ	方向	概要
鍵 (key)	OpenstackVolumeAuthenticationKey	In/Out	

6.166. OPENSTACKVOLUMEAUTHENTICATIONKEYS

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表6.505 メソッドの概要

名前	概要
add	OpenStack ボリュームプロバイダーに新しい認証キーを追加します。
list	認証キーの一覧を返します。

6.166.1. add POST

OpenStack ボリュームプロバイダーに新しい認証キーを追加します。

表6.506 パラメーターの概要

名前	タイプ	方向	概要
鍵 (key)	OpenstackVolumeAuthenticationKey	In/Out	

6.166.2. list GET

認証キーの一覧を返します。

返される認証キー一覧の順序は保証されません。

表6.507 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
keys	OpenstackVolumeAuthenticationKey[]	Out	
max	Integer	In	返すことのできるキーの最大数。

6.166.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.166.2.2. 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。

6.167. OPENSTACKVOLUMEPROVIDER

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表6.508 メソッドの概要

名前	概要
get	
importcertificates	外部ホストプロバイダーの SSL 証明書をインポートします。
remove	
testconnectivity	外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。
update	システム内の指定された OpenStack ボリュームプロバイダーを更新します。

6.167.1. get GET

表6.509 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
provider	OpenStackVolumeProvider	Out	

6.167.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.167.2. importcertificates POST

外部ホストプロバイダーの SSL 証明書をインポートします。

表6.510 パラメーターの概要

名前	タイプ	方向	概要
certificates	Certificate[]	In	

6.167.3. remove DELETE

表6.511 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	削除を非同期的に実行するかどうかを指定します。
force	Boolean	In	操作中に何らかの障害が発生した場合でも、操作が成功し、プロバイダーがデータベースから削除されるかどうかを示します。

6.167.3.1. force

操作中に何らかの障害が発生した場合でも、操作が成功し、プロバイダー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。

6.167.4. testconnectivity POST

外部プロバイダーの接続をテストするには、123 がプロバイダーの ID である次の要求を実行する必要があります。

```
POST /ovirt-engine/api/externalhostproviders/123/testconnectivity
```

表6.512 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.167.5. update PUT

システム内の指定された OpenStack ボリュームプロバイダーを更新します。

表6.513 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.168. OPENSTACKVOLUMEPROVIDERS

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表6.514 メソッドの概要

名前	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。

6.168.1. add POST

新しいボリュームプロバイダーを追加します。

以下はその例です。

```
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6.515 パラメーターの概要

名前	タイプ	方向	概要
provider	OpenStackV olumeProvid er	In/Out	

6.168.2. list GET

ボリュームプロバイダーのリストを取得します。

返されるボリュームプロバイダー一覧の順序は保証されません。

表6.516 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプロバイダーの最大数を設定します。

名前	タイプ	方向	概要
providers	OpenStackVolumeProvider[]	Out	
search	文字列	In	返されたボリュームプロバイダーを制限するために使用されるクエリー文字列です。

6.168.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.168.2.2. 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。

6.169. OPENSTACKVOLUMETYPE

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表6.517 メソッドの概要

名前	概要
get	

6.169.1. get GET

表6.518 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
type	OpenStackVolumeType	Out	

6.169.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.170. OPENSTACKVOLUMETYPES

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表6.519 メソッドの概要

名前	概要
list	ボリュームタイプの一覧を返します。

6.170.1. list GET

ボリュームタイプの一覧を返します。

返されるボリュームタイプ一覧の順序は保証されません。

表6.520 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すボリュームタイプの最大数を設定します。
types	OpenStackVolumeType[]	Out	

6.170.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.170.1.2. 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。

6.171. OPERATINGSYSTEM

表6.521 メソッドの概要

名前	概要
get	

6.171.1. get GET

表6.522 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
operating_system	OperatingSystemInfo	Out	

6.171.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.172. OPERATINGSYSTEMS

システムで使用可能なオペレーティングシステムのタイプのセットを管理します。

表6.523 メソッドの概要

名前	概要
list	システムで利用可能なオペレーティングシステムのタイプの一覧を返します。

6.172.1. list GET

システムで利用可能なオペレーティングシステムのタイプの一覧を返します。

返されるオペレーティングシステム一覧の順序は保証されません。

表6.524 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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	

6.172.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.172.1.2. 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。

6.173. パーミッション

表6.525 メソッドの概要

名前	概要
get	
remove	

6.173.1. get GET

表6.526 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
permission	パーミッション	Out	

6.173.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.173.2. remove DELETE

表6.527 パラメーターの概要

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

6.174. PERMIT

ロールの特定の permit を管理するサービス。

表6.528 メソッドの概要

名前	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。

6.174.1. 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6.529 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
permit	Permit	Out	ロールの permit。

6.174.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.174.2. 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6.530 パラメーターの概要

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

6.175. PERMITS

特定のロールの permits サブコレクションを表します。

表6.531 メソッドの概要

名前	概要
add	ロールに permit を追加します。

名前	概要
list	ロールの permits を一覧表示します。

6.175.1. add POST

ロールに permit を追加します。permit 名は、[cluster_levels](#) 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6.532 パラメーターの概要

名前	タイプ	方向	概要
permit	Permit	In/Out	追加する permit。

6.175.2. 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>
```

返される permits 一覧の順序は保証されません。

表6.533 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す permits の最大数を設定します。
permits	Permit[]	Out	permits のリスト。

6.175.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.175.2.2. 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。

6.176. QOS

表6.534 メソッドの概要

名前	概要
get	データセンターで指定された QoS を取得します。
remove	データセンターから指定された QoS を削除します。
update	データセンターで指定された QoS を更新します。

6.176.1. get GET

データセンターで指定された QoS を取得します。

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

以下のような応答が得られます。

```
<qos href="/ovirt-engine/api/datacenters/123/qoss/123" id="123">
  <name>123</name>
  <description>123</description>
  <max_iops>1</max_iops>
  <max_throughput>1</max_throughput>
  <type>storage</type>
  <data_center href="/ovirt-engine/api/datacenters/123" id="123"/>
</qos>
```

表6.535 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
qos	Qos	Out	クエリーされた QoS オブジェクト。

6.176.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.176.2. remove DELETE

データセンターから指定された QoS を削除します。

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

表6.536 パラメーターの概要

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

6.176.3. update PUT

データセンターで指定された QoS を更新します。

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

curl の例:

```
curl -u admin@internal:123456 -X PUT -H "content-type: application/xml" -d \  
"<qos><name>321</name><description>321</description><max_iops>10</max_iops></qos>" \  
https://engine/ovirt-engine/api/datacenters/123/qoss/123
```

次のような応答が返されます。

```
<qos href="/ovirt-engine/api/datacenters/123/qoss/123" id="123">  
  <name>321</name>  
  <description>321</description>  
  <max_iops>10</max_iops>  
  <max_throughput>1</max_throughput>  
  <type>storage</type>  
  <data_center href="/ovirt-engine/api/datacenters/123" id="123"/>  
</qos>
```

表6.537 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	更新された QoS オブジェクト。

6.177. QOSS

データセンターで利用可能な一連の **サービス品質** 設定を管理します。

表6.538 メソッドの概要

名前	概要
add	dataCenter に新しい QoS を追加します。
list	データセンターで利用可能な サービス品質 設定の一覧を返します。

6.177.1. add POST

dataCenter に新しい QoS を追加します。

POST /ovirt-engine/api/datacenters/123/qoss

応答は次のようになります。

```
<qos href="/ovirt-engine/api/datacenters/123/qoss/123" id="123">
  <name>123</name>
  <description>123</description>
  <max_iops>10</max_iops>
  <type>storage</type>
  <data_center href="/ovirt-engine/api/datacenters/123" id="123"/>
</qos>
```

表6.539 パラメーターの概要

名前	タイプ	方向	概要
qos	Qos	In/Out	QoS オブジェクトが追加されました。

6.177.2. list GET

データセンターで利用可能な **サービス品質** 設定の一覧を返します。

GET /ovirt-engine/api/datacenter/123/qoss

次のような応答が返されます。

```
<qoss>
  <qos href="/ovirt-engine/api/datacenters/123/qoss/1" id="1">...</qos>
  <qos href="/ovirt-engine/api/datacenters/123/qoss/2" id="2">...</qos>
  <qos href="/ovirt-engine/api/datacenters/123/qoss/3" id="3">...</qos>
</qoss>
```

返されるサービス品質設定の一覧は保証されません。

表6.540 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す QoS 記述子の最大数を設定します。
qoss	Qos[]	Out	クエリーされた QoS オブジェクトの一覧。

6.177.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.177.2.2. 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。

6.178. クォータ

表6.541 メソッドの概要

名前	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。

6.178.1. 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6.542 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
quota	クォータ	Out	

6.178.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.178.2. 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6.543 パラメーターの概要

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

6.178.3. 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6.544 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.179. QUOTA_CLUSTERLIMIT

表6.545 メソッドの概要

名前	概要
get	
remove	

6.179.1. get GET

表6.546 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
limit	QuotaClusterLimit	Out	

6.179.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.179.2. remove DELETE

表6.547 パラメーターの概要

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

6.180. QUOTA_CLUSTERLIMITS

クラスター用に設定されたクォータ制限のセットを管理します。

表6.548 メソッドの概要

名前	概要
add	指定された Quota にクラスター制限を追加します。
list	クラスターに設定されたクォータ制限のセットを返します。

6.180.1. add POST

指定された Quota にクラスター制限を追加します。

表6.549 パラメーターの概要

名前	タイプ	方向	概要
limit	QuotaClusterLimit	In/Out	

6.180.2. list GET

クラスターに設定されたクォータ制限のセットを返します。

返されるクォータ制限の一覧は保証されません。

表6.550 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
limits	QuotaClusterLimit[]	Out	
max	Integer	In	返す制限の最大数を設定します。

6.180.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.180.2.2. 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。

6.181. QUOTASTORAGELIMIT

表6.551 メソッドの概要

名前	概要
get	
remove	

6.181.1. get GET

表6.552 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
limit	QuotaStorageLimit	Out	

6.181.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.181.2. remove DELETE

表6.553 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	更新を非同期的に実行するかどうかを指定します。

6.182. QUOTASTORAGELIMITS

クォータ用に設定された一連のストレージ制限を管理します。

表6.554 メソッドの概要

名前	概要
add	指定されたクォータにストレージ制限を追加します。
list	クォータ用に設定されたストレージ制限の一覧を返します。

6.182.1. add POST

指定されたクォータにストレージ制限を追加します。

データセンター内のすべてのストレージドメインに対して 100 GiB のストレージ制限を作成するには、以下のようなリクエストを送信します。

```
POST /ovirt-engine/api/datacenters/123/quotas/456/quotastoragelimits
```

リクエスト本文は以下のようになります。

```
<quota_storage_limit>
  <limit>100</limit>
</quota_storage_limit>
```

ID **000** のストレージドメインに 50GiB のストレージ制限を作成するには、以下のようなリクエストを送信します。

```
POST /ovirt-engine/api/datacenters/123/quotas/456/quotastoragelimits
```

リクエスト本文は以下のようになります。

```
<quota_storage_limit>
  <limit>50</limit>
  <storage_domain id="000"/>
</quota_storage_limit>
```

表6.555 パラメーターの概要

名前	タイプ	方向	概要
limit	QuotaStorageLimit	In/Out	

6.182.2. list GET

クォータ用に設定されたストレージ制限の一覧を返します。

返されるストレージ制限一覧の順序は保証されません。

表6.556 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
limits	QuotaStorageLimit[]	Out	
max	Integer	In	返す制限の最大数を設定します。

6.182.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.182.2.2. 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。

6.183. QUOTAS

データセンター用に設定された一連のクォータを管理します。

表6.557 メソッドの概要

名前	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。

6.183.1. 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6.558 パラメーターの概要

名前	タイプ	方向	概要
quota	クォータ	In/Out	

6.183.2. list GET

データセンターのクォータを一覧表示します。

返されるクォータ一覧の順序は保証されません。

表6.559 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すクォータ記述子の最大数を設定します。

名前	タイプ	方向	概要
quotas	Quota[]	Out	

6.183.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.183.2.2. 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。

6.184. ロール

表6.560 メソッドの概要

名前	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。

6.184.1. 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6.561 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
role	ロール	Out	ロールを取得しました。

6.184.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.184.2. 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6.562 パラメーターの概要

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

6.184.3. 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6.563 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.185. 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6.564 メソッドの概要

名前	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。

6.185.1. 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6.565 パラメーターの概要

名前	タイプ	方向	概要
role	ロール	In/Out	追加されるロール。

6.185.2. 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>
```

```

</role>
...
</roles>

```

返されるロール一覧の順序は保証されません。

表6.566 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すロールの最大数を設定します。
roles	Role[]	Out	ロールの一覧を取得しました。

6.185.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.185.2.2. 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。

6.186. SCHEDULINGPOLICIES

システムで利用可能な一連のスケジューリングポリシーを管理します。

表6.567 メソッドの概要

名前	概要
add	新しいスケジューリングポリシーをシステムに追加します。
list	システムで利用可能なスケジューリングポリシーの一覧を返します。

6.186.1. add POST

新しいスケジューリングポリシーをシステムに追加します。

表6.568 パラメーターの概要

名前	タイプ	方向	概要
policy	SchedulingPolicy	In/Out	

6.186.2. list GET

システムで利用可能なスケジューリングポリシーの一覧を返します。

返されるスケジューリングポリシー一覧の順序は保証されません。

表6.569 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すポリシーの最大数を設定します。
policies	SchedulingPolicy	Out	

6.186.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.186.2.2. 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。

6.187. SCHEDULINGPOLICY

表6.570 メソッドの概要

名前	概要
get	
remove	
update	システム内の指定されたユーザー定義のスケジューリングポリシーを更新します。

6.187.1. get GET

表6.571 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
policy	SchedulingPolicy	Out	

6.187.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.187.2. remove DELETE

表6.572 パラメーターの概要

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

6.187.3. update PUT

システム内の指定されたユーザー定義のスケジューリングポリシーを更新します。

表6.573 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.188. SCHEDULINGPOLICYUNIT

表6.574 メソッドの概要

名前	概要
get	
remove	

6.188.1. get GET

表6.575 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
unit	SchedulingPolicyUnit	Out	

6.188.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.188.2. remove DELETE

表6.576 パラメーターの概要

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

6.189. SCHEDULINGPOLICYUNITS

システムで利用可能なスケジューリングポリシーユニットのセットを管理します。

表6.577 メソッドの概要

名前	概要
list	システムで利用可能なスケジューリングポリシーユニットの一覧を返します。

6.189.1. list GET

システムで利用可能なスケジューリングポリシーユニットの一覧を返します。

返されるスケジューリングポリシーユニット一覧の順序は保証されません。

表6.578 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
max	Integer	In	返すポリシーユニットの最大数を設定します。
units	SchedulingPolicyUnit[]	Out	

6.189.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.189.1.2. 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。

6.190. スナップショット

表6.579 メソッドの概要

名前	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。

6.190.1. get GET

表6.580 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
snapshot	スナップショット	Out	

6.190.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.190.2. remove DELETE

表6.581 パラメーターの概要

名前	タイプ	方向	概要
all_content	Boolean	In	仮想マシンのスナップショット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。
async	Boolean	In	削除を非同期的に実行するかどうかを指定します。

6.190.2.1. all_content

仮想マシンのスナップショット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、属性 **initialization.configuration.data** は除外されています。

たとえば、ID **123** の仮想マシンの ID **456** のスナップショットの完全な表現を取得するには、以下のようなリクエストを送信します。

```
GET /ovirt-engine/api/vms/123/snapshots/456?all_content=true
```

6.190.3. 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/>
```



注記

仮想マシンを実行する前に、コミット操作が完了し、仮想マシンがダウンしていることを確認してください。

表6.582 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	スナップショットの復元に含まれるディスクを指定します。
restore_memory	Boolean	In	

6.190.3.1. disks

スナップショットの復元に含まれるディスクを指定します。

ディスクパラメーターごとに、その **image_id** も指定する必要があります。

たとえば、識別子が **111** で **image_id** が **222** のディスクを含む、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
```

リクエスト本文:

```
<action>
  <disks>
    <disk id="111">
      <image_id>222</image_id>
    </disk>
  </disks>
</action>
```

6.191. SNAPSHOTCDROM

表6.583 メソッドの概要

名前	概要
get	

6.191.1. get GET

表6.584 パラメーターの概要

名前	タイプ	方向	概要
cdrom	Cdrom	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.191.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.192. SNAPSHOTCDROMS

仮想マシンのスナップショットの CD-ROM デバイスのセットを管理します。

表6.585 メソッドの概要

名前	概要
list	スナップショットの CD-ROM デバイスの一覧を返します。

6.192.1. list GET

スナップショットの CD-ROM デバイスの一覧を返します。

返される CD-ROM デバイス一覧の順序は保証されません。

表6.586 パラメーターの概要

名前	タイプ	方向	概要
cdroms	Cdrom[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す CDROM の最大数を設定します。

6.192.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.192.1.2. 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。

6.193. SNAPSHOTDISK

表6.587 メソッドの概要

名前	概要
get	

6.193.1. get GET

表6.588 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.193.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.194. SNAPSHOTDISKS

スナップショットのディスクセットを管理します。

表6.589 メソッドの概要

名前	概要
list	スナップショットのディスクの一覧を返します。

6.194.1. list GET

スナップショットのディスクの一覧を返します。

返されるフック一覧の順序は保証されません。

表6.590 パラメーターの概要

名前	タイプ	方向	概要
disks	Disk[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すディスクの最大数を設定します。

6.194.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.194.1.2. 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。

6.195. SNAPSHOTNIC

表6.591 メソッドの概要

名前	概要
get	

6.195.1. get GET

表6.592 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
nic	Nic	Out	

6.195.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.196. SNAPSHOTNICS

スナップショットのNICのセットを管理します。

表6.593 メソッドの概要

名前	概要
list	スナップショットのNICの一覧を返します。

6.196.1. list GET

スナップショットのNICの一覧を返します。

返されるNIC一覧の順序は保証されません。

表6.594 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すNICの最大数を設定します。
nics	Nic[]	Out	

6.196.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.196.1.2. 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。

6.197. スナップショット

ストレージドメインまたは仮想マシンの一連のスナップショットを管理します。

表6.595 メソッドの概要

名前	概要
add	仮想マシンのスナップショットを作成します。
list	ストレージドメインまたは仮想マシンのスナップショットの一覧を返します。

6.197.1. 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>
```

スナップショットにディスクのサブセットのみを含めるには、リクエスト本文に **disk_attachments** 要素を追加します。**disk_attachments** 要素で指定されていないディスクは、スナップショットの一部にならないことに注意してください。空の **disk_attachments** 要素が渡された場合は、スナップショットには仮想マシン設定のみが含まれます。**disk_attachments** 要素が渡されない場合は、すべてのディスクがスナップショットに含まれます。

ディスクごとに、新しいアクティブなイメージ ID を設定するために **image_id** 要素を指定することができます。これは、バックアップから一連のイメージを復元するために使用されます。つまり、スナップショットを使用してディスクを復元する場合、関連する **image_id** を各スナップショットに指定する必要があります (ディスクスナップショットの識別子がバックアップと同一になるようにするため)。

```
<snapshot>
  <description>My snapshot</description>
  <disk_attachments>
    <disk_attachment>
      <disk id="123">
        <image_id>456</image_id>
      </disk>
    </disk_attachment>
  </disk_attachments>
</snapshot>
```


重要

スナップショットが作成される時、`persist_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6.596 パラメーターの概要

名前	タイプ	方向	概要
snapshot	スナップ ショット	In/Out	

6.197.2. list GET

ストレージドメインまたは仮想マシンのスナップショットの一覧を返します。

返されるスナップショット一覧の順序は保証されません。

表6.597 パラメーターの概要

名前	タイプ	方向	概要
all_content	Boolean	In	仮想マシンのスナップショット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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すスナップショットの最大数を設定します。
snapshots	Snapshot[]	Out	

6.197.2.1. all_content

仮想マシンのスナップショット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、属性 **initialization.configuration.data** は除外されています。

たとえば、ID **123** スナップショットを持つ仮想マシンの完全な表現を取得するには、以下のようなリクエストを送信します。

```
GET /ovirt-engine/api/vms/123/snapshots?all_content=true
```

6.197.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.197.2.3. 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。

6.198. SSHPUBLICKEY

表6.598 メソッドの概要

名前	概要
get	
remove	
update	キーを新しいリソースに置き換えます。

6.198.1. get GET

表6.599 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
鍵 (key)	SshPublicKey	Out	

6.198.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.198.2. remove DELETE

表6.600 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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

キーを新しいリソースに置き換えます。



重要

エンジンのバージョン 4.4.8 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。代わりに、DELETE に続いて [追加操作](#) を使用してください。

表6.601 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	更新を非同期的に実行するかどうかを指定します。
鍵 (key)	SshPublicKey	In/Out	

6.199. SSHPUBLICKEYS

表6.602 メソッドの概要

名前	概要
add	
list	ユーザーの SSH 公開鍵の一覧を返します。

6.199.1. add POST

表6.603 パラメーターの概要

名前	タイプ	方向	概要
鍵 (key)	SshPublicKey	In/Out	

6.199.2. list GET

ユーザーの SSH 公開鍵の一覧を返します。

たとえば、識別子 **123** を持つユーザーの SSH キーの一覧を取得するには、以下のようなリクエストを送信します。

```
GET /ovirt-engine/api/users/123/sshpublickeys
```

結果は以下の XML ドキュメントになります。

```
<ssh_public_keys>
  <ssh_public_key href="/ovirt-engine/api/users/123/sshpublickeys/456" id="456">
    <content>ssh-rsa ...</content>
  </ssh_public_key>
</ssh_public_keys>
```

```
<user href="/ovirt-engine/api/users/123" id="123"/>
</ssh_public_key>
</ssh_public_keys>
```

または、以下の JSON オブジェクトになります。

```
{
  "ssh_public_key": [
    {
      "content": "ssh-rsa ...",
      "user": {
        "href": "/ovirt-engine/api/users/123",
        "id": "123"
      },
      "href": "/ovirt-engine/api/users/123/sshpublickeys/456",
      "id": "456"
    }
  ]
}
```

返されるキー一覧の順序は保証されません。

表6.604 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
keys	SshPublicKey[]	Out	
max	Integer	In	返すことのできるキーの最大数。

6.199.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.199.2.2. 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。

6.200. 統計

表6.605 メソッドの概要

名前	概要
get	

6.200.1. get GET

表6.606 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
Statistic[]	統計	Out	

6.200.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.201. 統計

表6.607 メソッドの概要

名前	概要
list	統計のリストを取得します。

6.201.1. 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>
```

返される統計一覧の順序は保証されません。

表6.608 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す統計の最大数を設定します。
statistics	Statistic[]	Out	

6.201.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.201.1.2. 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。

6.202. STEP

ステップを管理するサービス。

表6.609 メソッドの概要

名前	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。

6.202.1. 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6.610 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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	Boolean	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。

6.202.1.1. 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。

6.202.2. 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6.611 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
step	Step	Out	ステップの表現を取得します。

6.202.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.203. STEPS

ステップを管理するサービスです。

表6.612 メソッドの概要

名前	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。

6.203.1. 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6.613 パラメーターの概要

名前	タイプ	方向	概要
step	Step	In/Out	追加されるステップ。

6.203.2. 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>

```

返されるステップ一覧の順序は保証されません。

表6.614 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	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。

6.203.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.203.2.2. 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。

6.204. STORAGE

表6.615 メソッドの概要

名前	概要
get	

6.204.1. get GET

表6.616 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
report_status	Boolean	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	

6.204.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.204.1.2. 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>
```

6.205. STORAGE DOMAIN

表6.617 メソッドの概要

名前	概要
get	ストレージドメインの説明を取得します。
isattached	ストレージサーバーの一部である <code>is_attached</code> ブール値フィールドを使用して、ストレージドメインがすでにデータセンターにアタッチされているかどうかをクエリーするために使用されます。
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。

名前	概要
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。

6.205.1. get GET

ストレージドメインの説明を取得します。

表6.618 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
storage_domain	StorageDomain	Out	ストレージドメインの説明。

6.205.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.205.2. isattached POST

ストレージサーバーの一部である `is_attached` ブール値フィールドを使用して、ストレージドメインがすでにデータセンターにアタッチされているかどうかをクエリーするために使用されます。重要: この API を実行すると、ホストがストレージドメインから切断されます。

表6.619 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
ホスト	ホスト	In	データセンターのホストを示します。
is_attached	Boolean	Out	ストレージドメインがデータセンターにアタッチされているかどうかを示します。

6.205.3. 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>
```

Note that this operation is only applicable to block storage domains (i.e., storage domains with the xref:types-storage_type[storage type] of iSCSI or FCP).

表6.620 パラメーターの概要

名前	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。

6.205.4. 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6.621 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.205.5. 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。その後、ストレージドメインは、同じセットアップまたは別のセットアップに、すべてのデータとともに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、いずれにせよ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、LUN またはディレクトリーからメタデータが削除されるため、同じセットアップまたは異なるセットアップにインポートできません。

表6.622 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	削除を非同期的に実行するかどうかを指定します。
destroy	Boolean	In	ストレージにアクセスできない場合でも、操作が成功して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	Boolean	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> <pre>[source] ---- DELETE /ovirt-engine/api/storageDomains/123?format=true ----</pre> <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>
ホスト	文字列	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。

6.205.5.1. destroy

ストレージにアクセスできない場合でも、操作が成功して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。**destroy** の値が **true** の場合、**host** パラメーターは無視されます。

6.205.5.2. ホスト

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す。ただし、**destroy** パラメーターが含まれていて、その値が **true** である場合を除きます。この場合、**host** パラメーターは無視されます。

値には、ホストの名前または識別子が含まれている必要があります。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
```

6.205.6. update PUT

ストレージドメインを更新します。

[StorageDomain](#) の属性のすべてが、作成後に更新可能なわけではありません。更新できるのは、**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、既に存在するディスクの wipe after delete プロパティは変更されないことに注意してください)。

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、以下のようになります。

```
<storage_domain>
  <name>data2</name>
  <wipe_after_delete>true</wipe_after_delete>
</storage_domain>
```

表6.623 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	StorageDomain	In/Out	更新されたストレージドメイン。

6.205.7. 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存在する仮想マシンとディスクのメタデータを格納した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6.624 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	OVF_STORE 更新を非同期的に実行するかどうかを示します。

6.206. STORAGEDOMAINCONTENTDISK

表6.625 メソッドの概要

名前	概要
get	

6.206.1. get GET

表6.626 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	Out	
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.206.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.207. STORAGEDOMAINCONTENTDISKS

ストレージドメインで利用可能なディスクのセットを管理します。

表6.627 メソッドの概要

名前	概要
list	ストレージドメインで利用可能なディスクの一覧を返します。

6.207.1. list GET

ストレージドメインで利用可能なディスクの一覧を返します。

返されるディスクのリストの順序は、**sortby** 句が **search** パラメーターに含まれている場合にのみ保証されます。

表6.628 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すディスクの最大数を設定します。
search	文字列	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。

6.207.1.1. 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。

6.207.1.2. follow

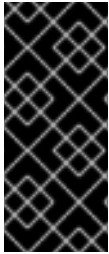
たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.207.1.3. 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。

6.208. STORAGEDOMAINDISK

ストレージドメインで利用可能な単一のディスクを管理します。



重要

エンジンのバージョン 4.2 以降、このサービスは、ストレージドメインで使用可能なディスクを一覧表示し、未登録のディスクを登録することのみを目的としています。ディスクのコピー、ディスクの移動など、他のすべての操作は非推奨になり、将来削除される予定です。これらの操作を実行するには、[システムのすべてのディスクを管理するサービス](#)、または [特定のディスクを管理するサービス](#) を使用します。

表6.629 メソッドの概要

名前	概要
copy	指定したストレージドメインにディスクをコピーします。
export	ディスクをエクスポートストレージドメインにエクスポートします。
get	ディスクの説明を取得します。
move	ディスクを別のストレージドメインに移動します。
reduce	ディスクイメージのサイズを縮小します。
remove	ディスクを削除します。
sparsify	ディスクをスパース化します。
update	ディスクを更新します。

6.208.1. copy POST

指定したストレージドメインにディスクをコピーします。



重要

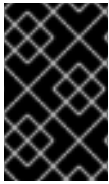
エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクをコピーするには、そのディスクを管理するサービスの [コピー](#) 操作を使用します。

表6.630 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In	作成されるディスクの説明。
storage_domain	StorageDomain	In	新しいディスクが作成されるストレージドメイン。

6.208.2. export POST

ディスクをエクスポートストレージドメインにエクスポートします。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクをエクスポートするには、そのディスクを管理するサービスの [エクスポート](#) 操作を使用します。

表6.631 パラメーターの概要

名前	タイプ	方向	概要
storage_domain	StorageDomain	In	ディスクがエクスポートされるエクスポートストレージドメイン。

6.208.3. get GET

ディスクの説明を取得します。

表6.632 パラメーターの概要

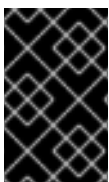
名前	タイプ	方向	概要
disk	ディスク	Out	ディスクの説明。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.208.3.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.208.4. move POST

ディスクを別のストレージドメインに移動します。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクを移動するには、そのディスクを管理するサービスの [移動](#) 操作を使用します。

表6.633 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

名前	タイプ	方向	概要
filter	Boolean	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	ディスクが移動されるストレージドメイン。

6.208.5. reduce POST

ディスクイメージのサイズを縮小します。

論理ボリュームで **縮小** を呼び出します (つまり、ブロックストレージドメインにのみ適用されます)。これは、フローティングディスクおよび実行されていない仮想マシンに接続されているディスクに適用されます。最適なサイズは自動的に算出されるため、サイズを指定する必要はありません。

表6.634 パラメーターの概要

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

6.208.6. remove DELETE

ディスクを削除します。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクを削除するには、そのディスクを管理するサービスの **remove** 操作を使用します。

6.208.7. sparsify POST

ディスクをスパース化します。

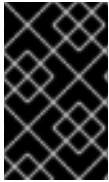


重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクを削除するには、そのディスクを管理するサービスの **remove** 操作を使用します。

6.208.8. update PUT

ディスクを更新します。



重要

エンジンのバージョン 4.2 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。ディスクを更新するには、そのディスクを管理するサービスの [更新](#) 操作を使用します。

表6.635 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In/Out	ディスクに適用する更新。

6.209. STORAGEDOMAINDISKS

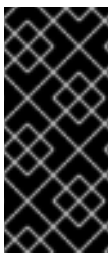
特定のストレージドメイン内で利用可能なディスクコレクションを管理します。

表6.636 メソッドの概要

名前	概要
add	ディスクを追加または登録します。
list	ストレージドメインで利用可能なディスクの一覧を取得する。

6.209.1. add POST

ディスクを追加または登録します。



重要

Red Hat Virtualization Manager のバージョン 4.2 以降、この操作は非推奨となり、後方互換性のためにのみ保持されています。これは今後削除されます。新しいディスクを追加するには、システムのディスクを管理するサービスの [add](#) 操作を使用します。未登録のディスクを登録するには、そのディスクを管理するサービスの [登録](#) 操作を使用します。

表6.637 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In/Out	追加または登録するディスク。
unregistered	Boolean	In	新しいディスクを追加するか、または既存の登録されていないディスクを登録する必要があるかどうかを示します。

6.209.1.1. unregistered

新しいディスクを追加するか、または既存の登録されていないディスクを登録する必要があるかどうか

新しいディスクを追加するか、または既存の登録されていないディスクを登録する必要があるかどうかを示します。値が **true** の場合、登録するディスクの ID を指定する必要があります。たとえば、ID **456** のディスクを登録する場合は、以下のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<disk id="456"/>
```

値が **false** の場合、ストレージドメインに新しいディスクが作成されます。この場合、**provisioned_size** 属性、**format** 属性、および **name** 属性が必須となります。たとえば、1 GiB の書き込みディスクに新しいコピーを作成するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<disk>
  <name>mydisk</name>
  <format>cow</format>
  <provisioned_size>1073741824</provisioned_size>
</disk>
```

デフォルト値は **false** です。

このパラメーターは、Red Hat Virtualization Manager のバージョン 4.2 以降で非推奨となっています。

6.209.2. list GET

ストレージドメインで利用可能なディスクの一覧を取得する。

返されるディスク一覧の順序は保証されません。

表6.638 パラメーターの概要

名前	タイプ	方向	概要
disks	Disk[]	Out	取得されたディスクの一覧。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すディスクの最大数を設定します。
unregistered	Boolean	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。

6.209.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.209.2.2. max

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

6.209.2.3. 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。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。

6.210. STORAGEDOMAINSERVERCONNECTION

表6.639 メソッドの概要

名前	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。

6.210.1. get GET

表6.640 パラメーターの概要

名前	タイプ	方向	概要
connection	StorageConnection	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.210.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.210.2. remove DELETE

ストレージ接続をストレージから切り離します。

表6.641 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.211. STORAGEDOMAINSERVERCONNECTIONS

ストレージドメインに存在するストレージサーバーへの一連の接続を管理します。

表6.642 メソッドの概要

名前	概要
add	
list	ストレージドメインに存在するストレージサーバーへの接続の一覧を返します。

6.211.1. add POST

表6.643 パラメーターの概要

名前	タイプ	方向	概要
connection	StorageConnection	In/Out	

6.211.2. list GET

ストレージドメインに存在するストレージサーバーへの接続の一覧を返します。

返される接続リストの順序は保証されません。

表6.644 パラメーターの概要

名前	タイプ	方向	概要
connections	StorageConnection[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す接続の最大数を設定します。

6.211.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.211.2.2. 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。

6.212. STORAGEDOMAINTEMPLATE

表6.645 メソッドの概要

名前	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	テンプレートの登録とは、テンプレートとディスクの設定をコピープロセスなしでデータベースに挿入することにより、データドメインからテンプレートをインポートすることを意味します。
remove	

6.212.1. get GET

表6.646 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
template	Template	Out	

6.212.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.212.2. 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>

```

クラスター ID または名前を指定せずにエンティティを登録すると、エンティティの OVF からのクラスター名が使用されます (登録要求にクラスターマッピングも含まれていない場合)。

表6.647 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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。
cluster	クラスター	In	
exclusive	Boolean	In	
storage_domain	StorageDomain	In	
template	Template	In	
vm	Vm	In	

6.212.2.1. 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ときに、**clone** パラメーターを **false** に設定してテンプレートをインポートできます。

6.212.3. register POST

テンプレートの登録とは、テンプレートとディスクの設定をコピープロセスなしでデータベースに挿入することにより、データドメインからテンプレートをインポートすることを意味します。

表6.648 パラメーターの概要

名前	タイプ	方向	概要
allow_partial_import	Boolean	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	Boolean	In	登録を非同期的に実行するかどうかを指定します。
clone	Boolean	In	
cluster	クラスター	In	
exclusive	Boolean	In	
registration_configuration	Registration Configuration	In	このパラメーターは、テンプレートの登録方法を記述します。
template	Template	In	
vnic_profile_mappings	VnicProfileMapping[]	In	インポート/登録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を記述する非推奨となった属性。

6.212.3.1. 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くても、システム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。

6.212.3.2. registration_configuration

このパラメーターは、テンプレートの登録方法を記述します。

このパラメーターは任意です。パラメーターが指定されていない場合、テンプレートは作成された元の環境と同じ設定で登録されます。

6.212.3.3. vnic_profile_mappings

インポート/登録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を記述する非推奨となった属性。

**警告**

この属性は、エンジンのバージョン 4.2.1 以降は非推奨となっており、後方互換性のためにのみ保持されていることに注意してください。これは今後削除されます。**vnuc_profile_mappings** を指定するには、[RegistrationConfiguration](#) タイプ内の **vnuc_profile_mappings** 属性を使用します。

6.212.4. remove DELETE

表6.649 パラメーターの概要

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

6.213. STORAGEDOMAINTEMPLATES

ストレージドメインで利用可能な一連のテンプレートを管理します。

表6.650 メソッドの概要

名前	概要
list	ストレージドメインで利用可能なテンプレートの一覧を返します。

6.213.1. list GET

ストレージドメインで利用可能なテンプレートの一覧を返します。

返されるテンプレート一覧の順序は保証されません。

表6.651 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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	Boolean	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。

6.213.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.213.1.2. 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。

6.213.1.3. 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。

6.214. STORAGEDOMAINVM

表6.652 メソッドの概要

名前	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。

6.214.1. get GET

表6.653 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
vm	Vm	Out	

6.214.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.214.2. 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>
```

クラスター ID または名前を指定せずにエンティティを登録すると、エンティティの OVF からのクラスター名が使用されます (登録要求にクラスターマッピングも含まれていない場合)。

表6.654 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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	クラスター	In	
collapse_snapshots	Boolean	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。
exclusive	Boolean	In	
storage_domain	StorageDomain	In	
vm	Vm	In	

6.214.2.1. 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。

6.214.2.2. 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。

6.214.3. register POST

表6.655 パラメーターの概要

名前	タイプ	方向	概要
allow_partial_import	Boolean	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	Boolean	In	登録を非同期的に実行するかどうかを指定します。
clone	Boolean	In	

名前	タイプ	方向	概要
cluster	クラスター	In	
reassign_bad_macs	Boolean	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。
registration_configuration	Registration Configuration	In	このパラメーターは、仮想マシンを登録する方法を記述します。
vm	Vm	In	
vnic_profile_mappings	VnicProfileMapping[]	In	インポート/登録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を記述する非推奨となった属性。

6.214.3.1. 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。

6.214.3.2. 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。

6.214.3.3. registration_configuration

このパラメーターは、仮想マシンを登録する方法を記述します。

このパラメーターは任意です。パラメーターが指定されていない場合、仮想マシンは、作成された元の環境と同じ設定で登録されます。

6.214.3.4. vnic_profile_mappings

インポート/登録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を記述する非推奨となった属性。



警告

この属性は、エンジンのバージョン 4.2.1 以降は非推奨となっており、後方互換性のためにのみ保持されていることに注意してください。これは今後削除されます。**vnic_profile_mappings** を指定するには、[RegistrationConfiguration](#) タイプ内の **vnic_profile_mappings** 属性を使用します。

6.214.4. 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6.656 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. 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6.657 メソッドの概要

名前	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。

6.215.1. 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6.658 パラメーターの概要

名前	タイプ	方向	概要
attachment	DiskAttachment	Out	ディスクアタッチメント。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.215.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.216. 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6.659 メソッドの概要

名前	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。

6.216.1. 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。

返されるディスクアタッチメント一覧の順序は保証されません。

表6.660 パラメーターの概要

名前	タイプ	方向	概要
attachments	DiskAttachment[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.216.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.217. 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6.661 メソッドの概要

名前	概要
list	エクスポートストレージドメインの仮想マシンの一覧を返します。

6.217.1. list GET

エクスポートストレージドメインの仮想マシンの一覧を返します。

返される仮想マシン一覧の順序は保証されません。

表6.662 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す仮想マシンの最大数を設定します。
unregistered	Boolean	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。
vm	Vm[]	Out	

6.217.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.217.1.2. 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。

6.217.1.3. 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。

6.218. STORAGEDOMAINS

システム内の一連のストレージドメインを管理します。

表6.663 メソッドの概要

名前	概要
add	新しいストレージドメインを追加します。
list	システム内のストレージドメインの一覧を返します。

6.218.1. add POST

新しいストレージドメインを追加します。

新しい `StorageDomain` の作成には、**name**、**type**、**host**、および **storage** の属性が必要です。**id** または **name** 属性で **host** 属性を特定します。Red Hat Virtualization 3.6 以降では、ストレージドメインでデフォルトによる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できますが、その場合はすでに存在している `wipe after delete` プロパティは変更されません。

name、**type**、**storage.type**、**storage.address**、**storage.path** を指定し、ID が **123** のホストを使用して新規ストレージドメインを追加するには、次のような要求を送信します。

```
POST /ovirt-engine/api/storageDomains
```

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>
  <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6.664 パラメーターの概要

名前	タイプ	方向	概要
storage_domain	StorageDomain	In/Out	追加するストレージドメイン

6.218.2. list GET

システム内のストレージドメインの一覧を返します。

返されるストレージドメインのリストの順序は、**sortby** 句が **search** パラメーターに含まれている場合にのみ保証されます。

表6.665 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	In	検索時に大文字と小文字の区別が考慮されるかどうかを示します。
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	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。
search	文字列	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	システム内のストレージドメインのリスト。

6.218.2.1. case_sensitive

検索時に大文字と小文字の区別が考慮され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。大文字小文字を無視して検索したい場合は、**false** を設定します。

6.218.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.218.2.3. 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。

6.219. STORAGESEVERCONNECTION

表6.666 メソッドの概要

名前	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。

6.219.1. get GET

表6.667 パラメーターの概要

名前	タイプ	方向	概要
connection	StorageConnection	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.219.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.219.2. 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6.668 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	削除を非同期的に実行するかどうかを指定します。
ホスト	文字列	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。

6.219.2.1. ホスト

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
```

6.219.3. update PUT

ストレージ接続を更新します。

たとえば、NFS 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>
</storage_connection>
```

iSCSI ストレージサーバーの接続を変更するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<storage_connection>
  <port>3260</port>
  <target>iqn.2017-01.com.myhost:444</target>
</storage_connection>
```

表6.669 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	更新を非同期的に実行するかどうかを指定します。
connection	StorageConnection	In/Out	
force	Boolean	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。

6.219.3.1. 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。

6.220. STORAGESERVERCONNECTIONEXTENSION

表6.670 メソッドの概要

名前	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。

6.220.1. get GET

表6.671 パラメーターの概要

名前	タイプ	方向	概要
extension	StorageConnectionExtension	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.220.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.220.2. remove DELETE

表6.672 パラメーターの概要

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

6.220.3. 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6.673 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.221. STORAGESEVERCONNECTIONEXTENSIONS

表6.674 メソッドの概要

名前	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	ストレージ接続エクステンションの一覧を返します。

6.221.1. 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6.675 パラメーターの概要

名前	タイプ	方向	概要
extension	StorageConnectionExtension	In/Out	

6.221.2. list GET

ストレージ接続エクステンションの一覧を返します。

返されるストレージ接続のリストの順序は保証されません。

表6.676 パラメーターの概要

名前	タイプ	方向	概要
extensions	StorageConnectionExtension[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すエクステンションの最大数を設定します。

6.221.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.221.2.2. 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。

6.222. STORAGESERVERCONNECTIONS

表6.677 メソッドの概要

名前	概要
add	新しいストレージ接続を作成します。
list	ストレージ接続の一覧を返します。

6.222.1. 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6.678 パラメーターの概要

名前	タイプ	方向	概要
connection	StorageCon nection	In/Out	

6.222.2. list GET

ストレージ接続の一覧を返します。

返される接続リストの順序は保証されません。

表6.679 パラメーターの概要

名前	タイプ	方向	概要
connections	StorageConnection[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す接続の最大数を設定します。

6.222.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.222.2.2. 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。

6.223. システム

表6.680 メソッドの概要

名前	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	

6.223.1. 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6.681 パラメーターの概要

名前	タイプ	方向	概要
api	Api	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.223.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.223.2. reloadconfigurations POST

表6.682 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.224. SYSTEMOPTION

システムの特定の設定オプションの値を提供するサービス。

表6.683 メソッドの概要

名前	概要
get	特定の設定オプションの値を取得します。

6.224.1. get GET

特定の設定オプションの値を取得します。

たとえば、設定オプション **MigrationPolicies** の値を取得するには、以下のようなリクエストを送信します。

```
GET /ovirt-engine/api/options/MigrationPolicies
```

そのリクエストに対する応答は以下のようになります。

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<system_option href="/ovirt-engine/api/options/MigrationPolicies" id="MigrationPolicies">
  <name>MigrationPolicies</name>
  <values>
```

```

<system_option_value>
  <value>[{"id":{"uuid":"80554327-0569-496b-bdeb-fcbbf52b827b"},...}]</value>
  <version>4.2</version>
</system_option_value>
<system_option_value>
  <value>[{"id":{"uuid":"80554327-0569-496b-bdeb-fcbbf52b827b"},...}]</value>
  <version>4.3</version>
</system_option_value>
<system_option_value>
  <value>[{"id":{"uuid":"80554327-0569-496b-bdeb-fcbbf52b827b"},...}]</value>
  <version>4.4</version>
</system_option_value>
<system_option_value>
  <value>[{"id":{"uuid":"80554327-0569-496b-bdeb-fcbbf52b827b"},...}]</value>
  <version>4.5</version>
</system_option_value>
<system_option_value>
  <value>[{"id":{"uuid":"80554327-0569-496b-bdeb-fcbbf52b827b"},...}]</value>
  <version>4.6</version>
</system_option_value>
<system_option_value>
  <value>[{"id":{"uuid":"80554327-0569-496b-bdeb-fcbbf52b827b"},...}]</value>
  <version>4.7</version>
</system_option_value>
</values>
</system_option>

```



注記

設定オプションをクエリーするには、適切な権限が必要です。一部のオプションは、管理者権限を持つユーザーのみがクエリーできます。



重要

後方互換性はなく、オプションの名前または値についての保証もありません。オプションは削除でき、その意味はいつでも変更することができます。

エンジンと同時にリリースされるアプリケーション以外では、このサービスを使用しないことを強くお勧めします。他のアプリケーションでの使用はサポートされていません。したがって、アクセス可能な設定オプションを記載したドキュメントはありません。

表6.684 パラメーターの概要

名前	タイプ	方向	概要
option	SystemOption	Out	システムの返された設定オプション。
version	文字列	In	設定オプションの特定のバージョンのみを返す必要があることを指定するオプションのバージョンパラメーター。

6.224.1.1. version

設定オプションの特定のバージョンのみを返す必要があることを指定するオプションのバージョンパラメーター。このパラメーターが使用されない場合は、すべてのバージョンが返されます。

たとえば、バージョン **4.2** のみの **MigrationPolicies** オプションの値を取得するには、以下のようなリクエストを送信します。

```
GET /ovirt-engine/api/options/MigrationPolicies?version=4.2
```

その要求に対する応答は以下のようになります。

```
<system_option href="/ovirt-engine/api/options/MigrationPolicies" id="MigrationPolicies">
  <name>MigrationPolicies</name>
  <values>
    <system_option_value>
      <value>[{"id":{"uuid":"80554327-0569-496b-bdeb-fcbbf52b827b"},...}]</value>
      <version>4.2</version>
    </system_option_value>
  </values>
</system_option>
```

6.225. SYSTEMOPTIONS

システムの設定オプションの値を提供するサービス。

6.226. 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6.685 メソッドの概要

名前	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。

6.226.1. 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6.686 パラメーターの概要

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

6.226.2. 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>

```

返されるパーミッションの順序は保証されません。

表6.687 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
permissions	Permission[]	Out	パーミッションのリスト

6.226.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.227. タグ

システム内の特定のタグを管理するサービス。

表6.688 メソッドの概要

名前	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。

6.227.1. 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6.689 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
tag	タグ	Out	タグ。

6.227.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.227.2. 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6.690 パラメーターの概要

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

6.227.3. 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6.691 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.228. 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6.692 メソッドの概要

名前	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。

6.228.1. 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6.693 パラメーターの概要

名前	タイプ	方向	概要
tag	タグ	In/Out	追加されたタグ。

6.228.2. 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)

```

返されるタグ一覧の順序は保証されません。

表6.694 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すタグの最大数を設定します。

名前	タイプ	方向	概要
tags	Tag[]	Out	システム内のすべてのタグの一覧。

6.228.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.228.2.2. 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。

6.229. 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6.695 メソッドの概要

名前	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。
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。

6.229.1. 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送信します。

```
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>
```

エンジンのバージョン 4.2 以降、テンプレートを仮想アプライアンス (OVA) としてエクスポートすることも可能となっています。たとえば、テンプレート **123** を、ホスト **myhost** のディレクトリー **/home/ovirt/** に配置される **myvm.ova** という名前の OVA ファイルとしてエクスポートするには、次のようにします。

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>
  <host>
    <name>myhost</name>
  </host>
  <directory>/home/ovirt</directory>
  <filename>myvm.ova</filename>
</action>
```

表6.696 パラメーターの概要

名前	タイプ	方向	概要
exclusive	Boolean	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。

6.229.1.1. 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。

6.229.2. 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6.697 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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。

6.229.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.229.3. 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6.698 パラメーターの概要

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

6.229.4. update PUT

テンプレートを更新します。

name、**description**、**type**、**memory**、**cpu**、**topology**、**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、1GiB のメモリを持つようにテンプレートを更新するには、次のようなリクエストを送信します。

```
PUT /ovirt-engine/api/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6.699 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.230. TEMPLATECDROM

テンプレートで CD-ROM デバイスを管理するサービス。

表6.700 メソッドの概要

名前	概要
<code>get</code>	この CDROM デバイスに関する情報を返します。

6.230.1. 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6.701 パラメーターの概要

名前	タイプ	方向	概要
<code>cdrom</code>	<code>Cdrom</code>	Out	CDROM デバイスに関する情報。
<code>follow</code>	文字列	In	たどる 必要のある内部リンクを指定します。

6.230.1.1. 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>
```

6.230.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.231. TEMPLATECDROMS

テンプレートの CD-ROM デバイスを一覧表示します。

表6.702 メソッドの概要

名前	概要
list	テンプレートの CD-ROM デバイスの一覧を返します。

6.231.1. list GET

テンプレートの CD-ROM デバイスの一覧を返します。

返される CD-ROM デバイス一覧の順序は保証されません。

表6.703 パラメーターの概要

名前	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す CD-ROM の最大数を設定します。

6.231.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.231.1.2. 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。

6.232. TEMPLATEDISK

表6.704 メソッドの概要

名前	概要
copy	テンプレートにアタッチされた指定のディスクを特定のストレージドメインにコピーします。
export	
get	
remove	

6.232.1. copy POST

テンプレートにアタッチされた指定のディスクを特定のストレージドメインにコピーします。

表6.705 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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	

6.232.2. export POST

表6.706 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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	

6.232.3. get GET

表6.707 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.232.3.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.232.4. remove DELETE

表6.708 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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. 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6.709 メソッドの概要

名前	概要
get	アタッチメントの詳細を返します。
remove	テンプレートからディスクを削除します。

6.233.1. get GET

アタッチメントの詳細を返します。

表6.710 パラメーターの概要

名前	タイプ	方向	概要
attachment	DiskAttachment	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.233.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.233.2. 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6.711 パラメーターの概要

名前	タイプ	方向	概要
force	Boolean	In	
storage_do main	文字列	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。

6.234. 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6.712 メソッドの概要

名前	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。

6.234.1. 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。

返されるアタッチメント一覧の順序は保証されません。

表6.713 パラメーターの概要

名前	タイプ	方向	概要
attachment s	DiskAttachm ent[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.234.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.235. TEMPLATEDISKS

表6.714 メソッドの概要

名前	概要
list	テンプレートのディスクの一覧を返します。

6.235.1. list GET

テンプレートのディスクの一覧を返します。

返されるフック一覧の順序は保証されません。

表6.715 パラメーターの概要

名前	タイプ	方向	概要
disks	Disk[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すディスクの最大数を設定します。

6.235.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.235.1.2. 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。

6.236. TEMPLATEGRAPHICSCONSOLE

表6.716 メソッドの概要

名前	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。

6.236.1. 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6.717 パラメーターの概要

名前	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.236.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.236.2. 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6.718 パラメーターの概要

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

6.237. TEMPLATEGRAPHICSCONSOLES

表6.719 メソッドの概要

名前	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。

6.237.1. 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6.720 パラメーターの概要

名前	タイプ	方向	概要
console	GraphicsConsole	In/Out	

6.237.2. 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。

返されるグラフィックコンソール一覧の順序は保証されません。

表6.721 パラメーターの概要

名前	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	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。

6.237.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.237.2.2. 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。

6.238. TEMPLATEMEDIATEDDEVICE

表6.722 メソッドの概要

名前	概要
get	テンプレートの仲介デバイスの設定を取得します。
remove	仲介デバイスをテンプレートから削除します。
update	仲介デバイスに関する情報を更新します。

6.238.1. get GET

テンプレートの仲介デバイスの設定を取得します。

表6.723 パラメーターの概要

名前	タイプ	方向	概要
device	VmMediated Device	Out	テンプレートの仲介デバイスに関する情報。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.238.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.238.2. remove DELETE

仲介デバイスをテンプレートから削除します。

表6.724 パラメーターの概要

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

6.238.3. update PUT

仲介デバイスに関する情報を更新します。

specParams 要素を使用して情報を更新できます。

たとえば、仲介デバイスを更新するには、次のような要求を送信します。

```
PUT /ovirt-engine/api/templates/123/mediateddevices/00000000-0000-0000-0000-000000000000
<vm_mediated_device>
  <spec_params>
    <property>
      <name>mdevType</name>
      <value>nvidia-11</value>
    </property>
  </spec_params>
</vm_mediated_device>
```

応答本文を使用:

```
<vm_mediated_device href="/ovirt-engine/api/templates/123/mediateddevices/00000000-0000-0000-0000-000000000000" id="00000000-0000-0000-0000-000000000000">
  <template href="/ovirt-engine/api/templates/123" id="123"/>
  <spec_params>
    <property>
      <name>mdevType</name>
      <value>nvidia-11</value>
    </property>
  </spec_params>
</vm_mediated_device>
```

表6.725 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	更新を非同期的に実行するかどうかを指定します。
devices	VmMediated Device	In/Out	仲介デバイスに関する情報。

6.238.3.1. devices

仲介デバイスに関する情報。

リクエストデータには **specParams** プロパティが含まれている必要があります。応答データには、更新された仲介デバイスに関する完全な情報が含まれています。

6.239. TEMPLATEMEDIATEDDEVICES

テンプレートの仲介デバイスを管理するサービス。

表6.726 メソッドの概要

名前	概要
add	テンプレートに新しい仲介デバイスを追加します。
list	テンプレートの設定済み仲介デバイスをすべてリストします。

6.239.1. add POST

テンプレートに新しい仲介デバイスを追加します。

表6.727 パラメーターの概要

名前	タイプ	方向	概要
device	VmMediated Device	In/Out	

6.239.2. list GET

テンプレートの設定済み仲介デバイスをすべてリストします。

返される仲介デバイス一覧の順序は保証されません。

表6.728 パラメーターの概要

名前	タイプ	方向	概要
devices	VmMediated Device[]	Out	テンプレートの仲介デバイスのリスト。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す仲介デバイスの最大数を設定します。

6.239.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.239.2.2. max

返す仲介デバイスの最大数を設定します。指定されていない場合は、すべての仲介デバイスが返されます。

6.240. TEMPLATENIC

表6.729 メソッドの概要

名前	概要
get	
remove	
update	テンプレートに割り当てられている指定のネットワークインターフェイスカードを更新します。

6.240.1. get GET

表6.730 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
nic	Nic	Out	

6.240.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.240.2. remove DELETE

表6.731 パラメーターの概要

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

6.240.3. update PUT

テンプレートに割り当てられている指定のネットワークインターフェイスカードを更新します。

表6.732 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.241. TEMPLATENICS

表6.733 メソッドの概要

名前	概要
add	テンプレートに新しいネットワークインターフェイスカードを追加します。
list	テンプレートの NIC の一覧を返します。

6.241.1. add POST

テンプレートに新しいネットワークインターフェイスカードを追加します。

表6.734 パラメーターの概要

名前	タイプ	方向	概要
nic	Nic	In/Out	

6.241.2. list GET

テンプレートの NIC の一覧を返します。

返される NIC 一覧の順序は保証されません。

表6.735 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す NIC の最大数を設定します。
nics	Nic[]	Out	

6.241.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.241.2.2. 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。

6.242. TEMPLATEWATCHDOG

表6.736 メソッドの概要

名前	概要
get	
remove	
update	指定された ID で識別されるテンプレートのウォッチドッグを更新します。

6.242.1. get GET

表6.737 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
watchdog	Watchdog	Out	

6.242.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.242.2. remove DELETE

表6.738 パラメーターの概要

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

6.242.3. update PUT

指定された ID で識別されるテンプレートのウォッチドッグを更新します。

表6.739 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.243. TEMPLATEWATCHDOGS

表6.740 メソッドの概要

名前	概要
add	指定された ID で識別されるテンプレートにウォッチドッグを追加します。
list	ウォッチドッグの一覧を返します。

6.243.1. add POST

指定された ID で識別されるテンプレートにウォッチドッグを追加します。

表6.741 パラメーターの概要

名前	タイプ	方向	概要
watchdog	Watchdog	In/Out	

6.243.2. list GET

ウォッチドッグの一覧を返します。

返されるウォッチドッグ一覧の順序は保証されません。

表6.742 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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	

6.243.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.243.2.2. 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。

6.244. 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6.743 メソッドの概要

名前	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。

6.244.1. 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。仮想マシンを識別するには、**vm.id** または **vm.name** 属性を使用します。たとえば、識別子が **123** の仮想マシンからテンプレートを作成するには、以下のよう
なリクエストを送信します。

```
POST /ovirt-engine/api/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>
```

バージョン 4.3 以降、スナップショットから仮想マシンテンプレートを作成するには、以下のよう
なリクエスト本文を送信します。

```
<template>
  <name>mytemplate</name>
  <vm id="123">
    <snapshots>
      <snapshot id="456"/>
    </snapshots>
  </vm>
</template>
```

テンプレートのディスクをカスタマイズして、元の仮想マシンのディスクとは異なる特性を作成でき
ます。これを行うには、**vm.disk_attachments** 属性を使用し、元の仮想マシンのディスクの識別子と
変更する特性を指定します。たとえば、元の仮想マシンに識別子 **456** のディスクがあり、そのディスクの
名前を **mydisk** に、フォーマットを **Copy On Write** に変更して **sparse** にする場合は、次のよう
なリクエスト本文を送信します。

```
<template>
  <name>mytemplate</name>
  <vm id="123">
    <disk_attachments>
      <disk_attachment>
        <disk id="456">
          <name>mydisk</name>
          <format>cow</format>
          <sparse>>true</sparse>
        </disk>
      </disk_attachment>
```

```

</disk_attachments>
</vm>
</template>

```

テンプレートは、既存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の **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_name>mytemplate_001</version_name>
  </version>
</template>

```

テンプレートの宛先ストレージドメインは、次の2つのいずれかの方法でカスタマイズできます。

1. リクエストレベルでグローバルにカスタマイズします。リクエストには、ストレージドメインで作成するディスクアタッチメントをリストする必要があります。ディスクアタッチメントがリストされていない場合、グローバルストレージドメインパラメーターは無視されます。

```

<template>
  <name>mytemplate</name>
  <storage_domain id="123"/>
  <vm id="456">
    <disk_attachments>
      <disk_attachment>
        <disk id="789">
          <format>cow</format>
          <sparse>true</sparse>
        </disk>
      </disk_attachment>
    </disk_attachments>
  </vm>
</template>

```

2. 各ディスクアタッチメントごとにカスタマイズします。各ディスクアタッチメントに必要なストレージドメインを指定します。グローバルストレージ定義を指定すると、ディスクアタッチメントごとのストレージドメインの指定が上書きされます。

```

<template>
  <name>mytemplate</name>
  <vm id="123">
    <disk_attachments>
      <disk_attachment>
        <disk id="456">
          <format>cow</format>
          <sparse>true</sparse>
          <storage_domains>
            <storage_domain id="789"/>
          </storage_domains>
        </disk_attachment>
      </disk_attachments>
    </vm>
  </template>

```



```

    </disk>
  </disk_attachment>
</disk_attachments>
</vm>
</template>

```

表6.744 パラメーターの概要

名前	タイプ	方向	概要
clone_permissions	Boolean	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	Boolean	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。

6.244.1.1. 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>

```

6.244.1.2. 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。

シーリングは、SSH キー、UDEV ルール、MAC アドレス、システム ID、ホスト名など、すべてのホスト固有の設定をファイルシステムから消去するため、テンプレートを使用して、手動の介入なしで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 オペレーティングシステムでのみサポートされています。

6.244.2. list GET

仮想マシンテンプレートの一覧を返します。

以下はその例です。

```
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。

返されるテンプレート一覧の順序は保証されません。

表6.745 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すテンプレートの最大数を設定します。
search	文字列	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。

6.244.2.1. 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。

6.244.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.244.2.3. 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。

6.245. UNMANAGEDNETWORK

表6.746 メソッドの概要

名前	概要
get	
remove	

6.245.1. get GET

表6.747 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
network	Unmanaged Network	Out	

6.245.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.245.2. remove DELETE

表6.748 パラメーターの概要

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

6.246. UNMANAGEDNETWORKS

表6.749 メソッドの概要

名前	概要
list	ホストの管理されていないネットワークの一覧を返します。

6.246.1. list GET

ホストの管理されていないネットワークの一覧を返します。

返されるネットワーク一覧の順序は保証されません。

表6.750 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すネットワークの最大数を設定します。
networks	Unmanaged Network[]	Out	

6.246.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.246.1.2. 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。

6.247. 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](#) を使用してください。

表6.751 メソッドの概要

名前	概要
get	システムユーザー情報を取得します。
remove	システムユーザーを削除します。
update	ユーザーに関する情報を更新します。

6.247.1. 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6.752 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
user	User	Out	システムユーザー。

6.247.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.247.2. 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6.753 パラメーターの概要

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

6.247.3. update PUT

ユーザーに関する情報を更新します。

user_options フィールドのみを更新できます。

たとえば、ユーザーのオプションを更新する場合は、以下のようになります。

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

リクエスト本文は以下のようになります。

```

<user>
  <user_options>
    <property>
      <name>test</name>
      <value>["any","JSON"]</value>
    </property>
  </user_options>
</user>

```



重要

エンジンのバージョン 4.4.5 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。代わりに [options](#) エンドポイントを使用してください。

表6.754 パラメーターの概要

名前	タイプ	方向	概要
user	User	In/Out	

6.248. USEROPTION

表6.755 メソッドの概要

名前	概要
get	JSON 型のユーザープロファイルプロパティを返します。
remove	JSON 型の既存のプロパティを削除します。

6.248.1. get GET

JSON 型のユーザープロファイルプロパティを返します。

リクエストの例 (識別子 **123** のユーザーと識別子 **456** のオプションの場合):

```
GET /ovirt-engine/api/users/123/options/456
```

結果は以下の XML ドキュメントになります。

```

<user_option href="/ovirt-engine/api/users/123/options/456" id="456">
  <name>SomeName</name>
  <content>["any", "JSON"]</content>
  <user href="/ovirt-engine/api/users/123" id="123"/>
</user_option>

```

表6.756 パラメーターの概要

名前	タイプ	方向	概要
option	UserOption	Out	

6.248.2. remove DELETE

JSON 型の既存のプロパティを削除します。

リクエストの例 (識別子 **123** のユーザーと識別子 **456** のオプションの場合):

```
DELETE /ovirt-engine/api/users/123/options/456
```

6.249. USEROPTIONS

表6.757 メソッドの概要

名前	概要
add	JSON 型の新しいユーザープロファイルプロパティを追加します。
list	タイプ JSON のユーザープロファイルプロパティの一覧を返します。

6.249.1. add POST

JSON 型の新しいユーザープロファイルプロパティを追加します。

リクエストの例 (識別子 **123** のユーザーの場合):

```
POST /ovirt-engine/api/users/123/options
```

Payload:

```
<user_option>
  <name>SomeName</name>
  <content>["any", "JSON"]</content>
</user_option>
```

表6.758 パラメーターの概要

名前	タイプ	方向	概要
option	UserOption	In/Out	

6.249.2. list GET

タイプ JSON のユーザープロファイルプロパティの一覧を返します。

リクエストの例 (識別子 **123** のユーザーの場合):

GET /ovirt-engine/api/users/123/options

結果は以下の XML ドキュメントになります。

```
<user_options>
  <user_option href="/ovirt-engine/api/users/123/options/456" id="456">
    <name>SomeName</name>
    <content>["any", "JSON"]</content>
    <user href="/ovirt-engine/api/users/123" id="123"/>
  </user_option>
</user_options>
```

表6.759 パラメーターの概要

名前	タイプ	方向	概要
オプション	UserOption[]	Out	

6.250. USERS

システム内のユーザーを管理するサービス。

表6.760 メソッドの概要

名前	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。

6.250.1. 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6.761 パラメーターの概要

名前	タイプ	方向	概要
user	User	In/Out	

6.250.2. 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>
```

返されるユーザーのリストの順序は保証されません。

表6.762 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すユーザーの最大数を設定します。
search	文字列	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	ユーザーのリスト。

6.250.2.1. 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。

6.250.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.250.2.3. 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。

6.251. VIRTUALFUNCTIONALLOWEDNETWORK

表6.763 メソッドの概要

名前	概要
get	
remove	

6.251.1. get GET

表6.764 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
network	Network	Out	

6.251.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.251.2. remove DELETE

表6.765 パラメーターの概要

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

6.252. VIRTUALFUNCTIONALLOWEDNETWORKS

表6.766 メソッドの概要

名前	概要
add	
list	ネットワークの一覧を返します。

6.252.1. add POST

表6.767 パラメーターの概要

名前	タイプ	方向	概要
network	Network	In/Out	

6.252.2. list GET

ネットワークの一覧を返します。

返されるネットワーク一覧の順序は保証されません。

表6.768 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	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	

6.252.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.252.2.2. 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。

6.253. VM

表6.769 メソッドの概要

名前	概要
autopincpuand numanodes	VM に自動 CPU および NUMA 設定を適用します。
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	
commitsnapshott	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。

名前	概要
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。
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	
reset	仮想マシンにリセットリクエストを送信します。
screenshot	VMの現在の状態のスクリーンショットをキャプチャーします。
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	指定された仮想マシン ID のシステム内の仮想マシンを更新します。

6.253.1. autopincpuandnumanodes POST

VM に自動 CPU および NUMA 設定を適用します。



重要

エンジンのバージョン 4.5 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。代わりに、PUT に続いて [更新操作](#) を使用してください。

リクエストの例:

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

リクエスト本文は以下のようになります。

```
<action>
  <optimize_cpu_settings>true</optimize_cpu_settings>
</action>
```

表6.770 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
optimize_cpu_settings	Boolean	In	自動 CPU および NUMA 設定を適用する方法を指定します。

6.253.1.1. optimize_cpu_settings

自動 CPU および NUMA 設定を適用する方法を指定します。true に設定すると、VM 固定ホストハードウェアに適合するように CPU トポロジーが調整されます。それ以外の場合は、VM CPU トポロジーが使用されます。

6.253.2. 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6.771 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.3. clone POST

表6.772 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	クローンを非同期で実行するかどうかを示します。

名前	タイプ	方向	概要
discard_snapshots	Boolean	In	スナップショットを折りたたんで仮想マシンのクローンを作成する必要がある場合は、 discard_snapshots パラメーターを使用します。
storage_domain	StorageDomain	In	仮想マシンディスクのコピー先のストレージドメイン。
vm	Vm	In	

6.253.3.1. discard_snapshots

スナップショットを折りたたんで仮想マシンのクローンを作成する必要がある場合は、**discard_snapshots** パラメーターを使用します。デフォルトは true です。

6.253.4. 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6.773 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.5. 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6.774 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.6. export POST

仮想マシンをエクスポートします。

仮想マシンは、エクスポートドメインにエクスポートできます。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>
```

エンジンのバージョン 4.2 以降、仮想マシンを仮想アプライアンス (OVA) としてエクスポートすることもできます。たとえば、仮想マシン **123** を、ホスト **myhost** のディレクトリー **/home/ovirt/** に配置される **myvm.ova** という名前の OVA ファイルとしてエクスポートするには、次のようにします。

```
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>
  <host>
    <name>myhost</name>
  </host>
  <directory>/home/ovirt</directory>
  <filename>myvm.ova</filename>
</action>
```



注記

エクスポート操作が完了したことを確認してから、エクスポートドメインで何らかの操作を試みます。

表6.775 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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。

名前	タイプ	方向	概要
exclusive	Boolean	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す (オーバーライド)。
storage_domain	StorageDomain	In	仮想マシンをエクスポートする (エクスポート) ストレージドメイン。

6.253.7. 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。

例:

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

```
<action/>
```

表6.776 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.8. get GET

仮想マシンの説明を取得します。

表6.777 パラメーターの概要

名前	タイプ	方向	概要
all_content	Boolean	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	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
next_run	Boolean	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。
ovf_as_ova	Boolean	In	結果が、その VM の OVA ファイルに表示されるように OVF を公開する必要があるかどうかを示します。
vm	Vm	Out	仮想マシンの説明。

6.253.8.1. 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。

6.253.8.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.253.8.3. 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
```

6.253.8.4. ovf_as_ova

結果が、その VM の OVA ファイルに表示されるように OVF を公開する必要があるかどうかを示します。仮想マシンを記述する OVF ドキュメント。このパラメーターは、`all_content=True` が設定されている場合にのみ機能します。OVF は **initialization.configuration.data** に提示されます。

以下はその例です。

```
GET /vms/{vm:id}?all_content=true&ovf_as_ova=true
```

6.253.9. 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。

以下はその例です。

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

リクエスト本文:

```
<action/>
```

表6.778 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.10. 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。

例:

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

```
<action>
  <maintenance_enabled>true</maintenance_enabled/>
</action>
```

表6.779 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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。

6.253.11. migrate POST

仮想マシンを別の物理ホストに移行します。

例:

```
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6.780 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	移行を非同期的に実行するかどうかを指定します。
cluster	クラスター	In	仮想マシンの移行先のクラスターを指定します。
force	Boolean	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。
ホスト	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。
migrate_vms_in_affinity_closure	Boolean	In	同じホスト上で実行されている、この仮想マシンとのポジティブな適用アフィニティグループ内の他のすべての仮想マシンも移行します。

6.253.11.1. 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。



警告

別のクラスターへのライブマイグレーションはサポートされていません。移行を試みる前に、ターゲットクラスターのハードウェアアーキテクチャーとネットワークアーキテクチャーを十分に検討してください。

6.253.11.2. 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。

6.253.11.3. ホスト

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。

6.253.11.4. migrate_vms_in_affinity_closure

同じホスト上で実行されている、この仮想マシンとのポジティブな適用アフィニティグループ内の他のすべての仮想マシンも移行します。

デフォルト値は **false** です。

6.253.12. 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6.781 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	スナップショットのプレビューに含まれるディスクを指定します。

名前	タイプ	方向	概要
lease	StorageDomainLease	In	スナップショットのプレビューで使用するリースストレージドメイン ID を指定します。
restore_memory	Boolean	In	
snapshot	スナップショット	In	
vm	Vm	In	

6.253.12.1. disks

スナップショットのプレビューに含まれるディスクを指定します。

ディスクパラメーターごとに、その **image_id** も指定する必要があります。

たとえば、識別子が **111** で、その **image_id** が **222** のディスクを含む識別子 **456** のスナップショットをプレビューするには、次のようなリクエストを送信します。

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

リクエスト本文:

```
<action>
  <disks>
    <disk id="111">
      <image_id>222</image_id>
    </disk>
  </disks>
  <snapshot id="456"/>
</action>
```

6.253.12.2. lease

スナップショットのプレビューで使用するリースストレージドメイン ID を指定します。リースパラメーターが渡されない場合、プレビューされたスナップショットリースストレージドメインが使用されます。ストレージドメインパラメーターが空のリースパラメーターが渡された場合、スナップショットプレビューにリースは使用されません。リースパラメーターがストレージドメインパラメーターと共に渡された場合、仮想マシンスナップショットの1つに属するリースドメイン ID の1つだけをストレージドメイン ID にすることができます。これはオプションのパラメーターで、デフォルトは **null** に設定されています。

6.253.13. reboot POST

仮想マシンに再起動リクエストを送信します。

以下はその例です。

```
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/>
```

バックアップが実行されている場合でも仮想マシンを再起動するには、アクションに force 要素を含める必要があります。

たとえば、仮想マシン **123** を強制的に再起動するには、以下を実行します。

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

```
<action>
  <force>true</force>
</action>
```

表6.782 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
force	Boolean	In	仮想マシンのバックアップが実行されている場合でも、仮想マシンを強制的に再起動する必要があるかどうかを示します。

6.253.14. 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6.783 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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	Boolean	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。

6.253.14.1. 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。

6.253.15. reordermacaddresses POST

表6.784 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.16. reset POST

仮想マシンにリセットリクエストを送信します。

以下はその例です。

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

reset 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6.785 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.17. screenshot POST

VM の現在の状態のスクリーンショットをキャプチャーします。

以下はその例です。

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

screenshot 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/>
```

6.253.18. 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。

以下はその例です。

```
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/>
```

バックアップが実行されている場合でも仮想マシンをシャットダウンするには、アクションに **force** 要素を含める必要があります。

たとえば、仮想マシン **123** を強制的にシャットダウンするには、以下を実行します。

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

```
<action>
  <force>true</force>
</action>
```

表6.786 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
force	Boolean	In	バックアップが実行されている場合でも、仮想マシンを強制的にシャットダウンする必要があるかどうかを示します。
reason	文字列	In	仮想マシンが停止した理由。

6.253.18.1. 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。

6.253.19. 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6.787 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
authorized_key	AuthorizedKey	In	
filter	Boolean	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	Boolean	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	Boolean	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_ignition	Boolean	In	true に設定すると、初期化タイプは Ignition に設定されます。
use_initialization	Boolean	In	true に設定すると、初期化タイプは VM の OS によって設定されます。
use_sysprep	Boolean	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。
volatile	Boolean	In	ゲストが開始した再起動の場合でも、この実行設定が破棄されることを示します。

6.253.19.1. 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。

6.253.19.2. 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。詳細には、[cloud-init のドキュメント](#) を参照してください。

6.253.19.3. use_ignition

true に設定すると、初期化タイプは **Ignition** に設定されます。デフォルト値は **false** です。詳細は、[Ignition のドキュメント](#) を参照してください。

6.253.19.4. use_initialization

true に設定すると、初期化タイプは VM の OS によって設定されます。Windows は **Sysprep** に、Linux は **cloud-init** に、RedHat CoreOS は **Ignition** に設定されます。いずれかの初期化タイプが明示的に設定されている場合 (`useCloudInit`、`useSysprep`、または `useIgnition`)、それらが優先され、このフラグは無視されます。デフォルト値は **false** です。

6.253.19.5. 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。詳細については、[Sysprep](#) を参照してください。

6.253.19.6. vm

この特定の実行のための仮想マシンの定義。

以下はその例です。

```
<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。

6.253.19.7. volatile

ゲストが開始した再起動の場合でも、この実行設定が破棄されることを示します。デフォルト値は **false** です。

6.253.20. 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。

以下はその例です。

```
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/>
```

バックアップが実行されている場合でも仮想マシンを停止するには、アクションに `force` 要素を含める必要があります。

たとえば、仮想マシン **123** を強制的に停止するには、以下を実行します。

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

```
<action>
  <force>true</force>
</action>
```

表6.788 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
force	Boolean	In	バックアップが実行されている場合でも、仮想マシンを強制的に停止する必要があるかどうかを示します。
reason	文字列	In	仮想マシンが停止した理由。

6.253.20.1. 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。

6.253.21. 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。

以下はその例です。

```
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6.789 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.22. 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。

例:

```
POST /api/vms/123/thawfilesystems
```

```
<action/>
```

表6.790 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.23. 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。

以下はその例です。

```
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6.791 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.253.24. 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6.792 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.253.25. update PUT

指定された仮想マシン ID のシステム内の仮想マシンを更新します。

表6.793 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	更新を非同期的に実行するかどうかを指定します。
next_run	Boolean	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	

6.253.25.1. 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。

6.254. 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6.794 メソッドの概要

名前	概要
get	アプリケーションに関する情報を返します。

6.254.1. get GET

アプリケーションに関する情報を返します。

表6.795 パラメーターの概要

名前	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	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.254.1.1. 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>
```

6.254.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.255. 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6.796 メソッドの概要

名前	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。

6.255.1. 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。

返されるアプリケーション一覧の順序は保証されません。

表6.797 パラメーターの概要

名前	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	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すアプリケーションの最大数を設定します。

6.255.1.1. 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>
```

6.255.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.255.1.3. 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。

6.256. VMBACKUP

仮想マシンのバックアップを管理するサービス。

表6.798 メソッドの概要

名前	概要
finalize	仮想マシンのバックアップエンティティの最終処理を行います。
get	仮想マシンのバックアップに関する情報を返します。

6.256.1. finalize POST

仮想マシンのバックアップエンティティの最終処理を行います。

バックアップを終了し、リソースのロックを解除し、クリーンアップを実行します。ID が 123 の仮想マシンと ID が 456 のバックアップをファイナライズするには、次のように要求を送信します。

```
POST /ovirt-engine/api/vms/123/backups/456/finalize
```

リクエスト本文の場合は、以下のようになります。

```
<action />
```

6.256.2. get GET

仮想マシンのバックアップに関する情報を返します。

表6.799 パラメーターの概要

名前	タイプ	方向	概要
backup	バックアップ	Out	仮想マシンのバックアップエンティティに関する情報。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.256.2.1. backup

仮想マシンのバックアップエンティティに関する情報。

```
<backups>
  <backup id="backup-uuid">
    <from_checkpoint_id>previous-checkpoint-uuid</from_checkpoint_id>
```

```

<link href="/ovirt-engine/api/vms/vm-uuid/backups/backup-uuid/disks" rel="disks"/>
<status>initializing</status>
<creation_date>
</backup>
</backups>

```

6.256.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.257. VMBACKUPDISK

表6.800 メソッドの概要

名前	概要
get	ディスクの説明を取得します。

6.257.1. get GET

ディスクの説明を取得します。

表6.801 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	Out	ディスクの説明。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.257.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.258. VMBACKUPDISKS

表6.802 メソッドの概要

名前	概要
list	バックアップ内のディスクの一覧を返します。

6.258.1. list GET

バックアップ内のディスクの一覧を返します。

表6.803 パラメーターの概要

名前	タイプ	方向	概要
disks	Disk[]	Out	取得されたディスクの一覧。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すディスクの最大数を設定します。

6.258.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.258.1.2. max

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

6.259. VMBACKUPS

仮想マシンのバックアップを一覧表示します。

表6.804 メソッドの概要

名前	概要
add	新しいバックアップエンティティを仮想マシンに追加します。
list	仮想マシンのバックアップのリスト。

6.259.1. add POST

新しいバックアップエンティティを仮想マシンに追加します。

たとえば、チェックポイント ID **previous-checkpoint-uuid** から仮想マシンの新しい増分バックアップを開始するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```
<backup>
  <from_checkpoint_id>previous-checkpoint-uuid</from_checkpoint_id>
  <disks>
    <disk id="disk-uuid" />
    ...
  </disks>
</backup>
```

応答本文:

```

<backup id="backup-uuid">
  <from_checkpoint_id>previous-checkpoint-uuid</from_checkpoint_id>
  <to_checkpoint_id>new-checkpoint-uuid</to_checkpoint_id>
  <disks>
    <disk id="disk-uuid" />
    ...
  </disks>
  <status>initializing</status>
  <creation_date>
</backup>

```

作成されたバックアップの ID を提供するには、次のようなリクエストを送信します。

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

リクエスト本文は以下のようになります。

```

<backup id="backup-uuid">
  <from_checkpoint_id>previous-checkpoint-uuid</from_checkpoint_id>
  <disks>
    <disk id="disk-uuid" />
    ...
  </disks>
</backup>

```

表6.805 パラメーターの概要

名前	タイプ	方向	概要
backup	バックアップ	In/Out	仮想マシンのバックアップエンティティーに関する情報。
require_consistency	Boolean	In	VM がフリーズに失敗した場合にバックアップが失敗するかどうかを示します。
use_active	Boolean	In	バックアップの実行にアクティブボリュームを使用するかどうかを示します。

6.259.1.1. require_consistency

VM がフリーズに失敗した場合にバックアップが失敗するかどうかを示します。

requireConsistency=True の場合に、VM のフリーズに失敗した場合は VM のバックアップは失敗します。

REST API 呼び出しは次のようになります。

```
POST /ovirt-engine/api/vms/123/backups?require_consistency=true
```

requireConsistency フラグのデフォルト値は **false** です。

6.259.1.2. use_active

バックアップの実行にアクティブボリュームを使用するかどうかを示します。

useActive=False の場合、バックアップ操作のスナップショットが作成されます。

REST API 呼び出しは次のようになります。

```
POST /ovirt-engine/api/vms/123/backups?use_active=false
```

useActive フラグのデフォルト値は **false** です。

6.259.2. list GET

仮想マシンのバックアップのリスト。

表6.806 パラメーターの概要

名前	タイプ	方向	概要
backups	Backup[]	Out	仮想マシンのバックアップエンティティに関する情報。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す仮想マシンバックアップの最大数を設定します。

6.259.2.1. backups

仮想マシンのバックアップエンティティに関する情報。

```
<backups>
  <backup id="backup-uuid">
    <from_checkpoint_id>previous-checkpoint-uuid</from_checkpoint_id>
    <disks>
      <disk id="disk-uuid" />
      ...
    </disks>
    <status>initializing</status>
    <creation_date>
  </backup>
</backups>
```

6.259.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.259.2.3. max

返す仮想マシンバックアップの最大数を設定します。指定されていない場合は、すべての仮想マシンのバックアップが返されます。

6.260. 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6.807 メソッドの概要

名前	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。

6.260.1. 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6.808 パラメーターの概要

名前	タイプ	方向	概要
cdrom	Cdrom	Out	CDROM デバイスに関する情報。
current	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.260.1.1. 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。

6.260.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.260.2. 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のようになります。

```
<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6.809 パラメーターの概要

名前	タイプ	方向	概要
cdrom	Cdrom	In/Out	CDROM デバイスに関する情報。
current	Boolean	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。

6.260.2.1. 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。

6.261. 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6.810 メソッドの概要

名前	概要
add	指定された ID で識別される仮想マシンに cdrom を追加します。
list	仮想マシンの CDROM デバイスの一覧を返します。

6.261.1. add POST

指定された ID で識別される仮想マシンに cdrom を追加します。

表6.811 パラメーターの概要

名前	タイプ	方向	概要
cdrom	Cdrom	In/Out	

6.261.2. list GET

仮想マシンの CDROM デバイスの一覧を返します。

返される CD-ROM デバイス一覧の順序は保証されません。

表6.812 パラメーターの概要

名前	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す CDROM の最大数を設定します。

6.261.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.261.2.2. 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。

6.262. VMCHECKPOINT

仮想マシンのチェックポイントを管理するサービス。

表6.813 メソッドの概要

名前	概要
get	仮想マシンのチェックポイントに関する情報を返します。
remove	仮想マシンのチェックポイントエンティティを削除します。

6.262.1. get GET

仮想マシンのチェックポイントに関する情報を返します。

表6.814 パラメーターの概要

名前	タイプ	方向	概要
checkpoint	Checkpoint	Out	仮想マシンのチェックポイントエンティティに関する情報。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.262.1.1. checkpoint

仮想マシンのチェックポイントエンティティに関する情報。

```
<checkpoint id="checkpoint-uuid">
  <link href="/ovirt-engine/api/vms/vm-uuid/checkpoints/checkpoint-uuid/disks" rel="disks"/>
```

```
<parent_id>parent-checkpoint-uuid</parent_id>
<creation_date>xxx</creation_date>
<vm href="/ovirt-engine/api/vms/vm-uuid" id="vm-uuid"/>
</checkpoint>
```

6.262.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.262.2. remove DELETE

仮想マシンのチェックポイントエンティティを削除します。

libvirt とデータベースからチェックポイントを削除します。

表6.815 パラメーターの概要

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

6.263. VMCHECKPOINTDISK

表6.816 メソッドの概要

名前	概要
get	ディスクの説明を取得します。

6.263.1. get GET

ディスクの説明を取得します。

表6.817 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	Out	ディスクの説明。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.263.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.264. VMCHECKPOINTDISKS

表6.818 メソッドの概要

名前	概要
list	チェックポイント内のディスクの一覧を返します。

6.264.1. list GET

チェックポイント内のディスクの一覧を返します。

表6.819 パラメーターの概要

名前	タイプ	方向	概要
disks	Disk[]	Out	取得されたディスクの一覧。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すディスクの最大数を設定します。

6.264.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.264.1.2. max

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

6.265. VMCHECKPOINTS

仮想マシンのチェックポイントを一覧表示します。

表6.820 メソッドの概要

名前	概要
list	仮想マシンチェックポイントのリスト。

6.265.1. list GET

仮想マシンチェックポイントのリスト。

ID が 123 の仮想マシンのチェックポイントのリストを取得するには、次のようにリクエストを送信します。

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

表6.821 パラメーターの概要

名前	タイプ	方向	概要
checkpoints	Checkpoint[]	Out	仮想マシンのチェックポイントエンティティに関する情報。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す仮想マシンチェックポイントの最大数を設定します。

6.265.1.1. checkpoints

仮想マシンのチェックポイントエンティティに関する情報。

```
<checkpoints>
  <checkpoint id="checkpoint-uuid">
    <link href="/ovirt-engine/api/vms/vm-uuid/checkpoints/checkpoint-uuid/disks" rel="disks"/>
    <parent_id>parent-checkpoint-uuid</parent_id>
    <creation_date>xxx</creation_date>
    <vm href="/ovirt-engine/api/vm-uuid" id="vm-uuid"/>
  </checkpoint>
</checkpoints>
```

6.265.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.265.1.3. max

返す仮想マシンチェックポイントの最大数を設定します。指定されていない場合は、すべての仮想マシンのチェックポイントが返されます。

6.266. VMDISK

表6.822 メソッドの概要

名前	概要
activate	
deactivate	
export	
get	
move	

名前	概要
reduce	ディスクイメージのサイズを縮小します。
remove	仮想マシンからディスクを切り離します。
update	

6.266.1. activate POST

表6.823 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.266.2. deactivate POST

表6.824 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.266.3. export POST

表6.825 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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。

6.266.4. get GET

表6.826 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	Out	

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.266.4.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.266.5. move POST

表6.827 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	Boolean	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。

6.266.6. reduce POST

ディスクイメージのサイズを縮小します。

論理ボリュームで **縮小** を呼び出します (つまり、ブロックストレージドメインにのみ適用されます)。これは、フローティングディスクおよび実行されていない仮想マシンに接続されているディスクに適用されます。最適なサイズは自動的に算出されるため、サイズを指定する必要はありません。

表6.828 パラメーターの概要

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

6.266.7. 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6.829 パラメーターの概要

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

6.266.8. update PUT

表6.830 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.267. VMDISKS

表6.831 メソッドの概要

名前	概要
add	
list	仮想マシンのディスクの一覧を返します。

6.267.1. add POST

表6.832 パラメーターの概要

名前	タイプ	方向	概要
disk	ディスク	In/Out	

6.267.2. list GET

仮想マシンのディスクの一覧を返します。

返されるフック一覧の順序は保証されません。

表6.833 パラメーターの概要

名前	タイプ	方向	概要
disks	Disk[]	Out	
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
max	Integer	In	返すディスクの最大数を設定します。

6.267.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.267.2.2. 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。

6.268. VMGRAPHICSCONSOLE

表6.834 メソッドの概要

名前	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。

6.268.1. 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。



重要

デフォルトでは、**current** パラメーターが指定されていない場合、返されるデータは仮想マシンの次の実行に対応します。システムの現在の実装では、これは、システムが次の実行に使用されるアドレスとポートを認識していないため、**address** と **port** の属性が入力されないことを意味します。ほとんどの場合、これらの属性が必要であるため、**current** パラメーターを値 **true** で常に明示的に含めることを強くお勧めします。

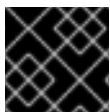
表6.835 パラメーターの概要

名前	タイプ	方向	概要
----	-----	----	----

名前	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	Boolean	In	返されるデータが仮想マシンの次の実行に対応するか、現在の実行に対応するかを指定します。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.268.1.1. current

返されるデータが仮想マシンの次の実行に対応するか、現在の実行に対応するかを指定します。



重要

値が **true** でない限り、**address** と **port** の属性は入力されません。

たとえば、**address** と **port** の属性など、仮想マシンの現在の実行に関するデータを取得するには、次のようなリクエストを送信します。

```
GET /ovit-engine/api/vms/123/graphicsconsoles/456?current=true
```

デフォルト値は **false** です。

6.268.1.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.268.2. proxyticket POST

表6.836 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.268.3. 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/remotewriterconnectionfile

remotewriterconnectionfile 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6.837 パラメーターの概要

名前	タイプ	方向	概要
remote_viewer_connection_file	文字列	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。

6.268.3.1. 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。

6.268.4. 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6.838 パラメーターの概要

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

6.268.5. 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6.839 パラメーターの概要

名前	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。

6.269. VMGRAPHICSCONSOLES

表6.840 メソッドの概要

名前	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。

6.269.1. 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6.841 パラメーターの概要

名前	タイプ	方向	概要
console	GraphicsConsole	In/Out	

6.269.2. 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。



重要

デフォルトでは、**current** パラメーターが指定されていない場合、返されるデータは仮想マシンの次の実行に対応します。システムの現在の実装では、これは、システムが次の実行に使用されるアドレスとポートを認識していないため、**address** と **port** の属性が入力されないことを意味します。ほとんどの場合、これらの属性が必要であるため、**current** パラメーターを値 **true** で常に明示的に含めることを強くお勧めします。

返されるグラフィックコンソール一覧の順序は保証されません。

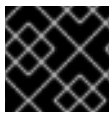
表6.842 パラメーターの概要

名前	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。

名前	タイプ	方向	概要
current	Boolean	In	返されるデータが仮想マシンの次の実行に対応するか、現在の実行に対応するかを指定します。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すコンソールの最大数を設定します。

6.269.2.1. current

返されるデータが仮想マシンの次の実行に対応するか、現在の実行に対応するかを指定します。



重要

値が **true** でない限り、**address** と **port** の属性は入力されません。

たとえば、**address** と **port** の属性など、仮想マシンの現在の実行に関するデータを取得するには、次のようなリクエストを送信します。

```
GET /ovirt-engine/api/vms/123/graphicsconsoles?current=true
```

デフォルト値は **false** です。

6.269.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.269.2.3. 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。

6.270. 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6.843 メソッドの概要

名前	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。

6.270.1. 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。

例:

```
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6.844 パラメーターの概要

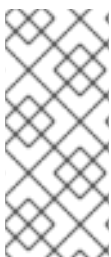
名前	タイプ	方向	概要
device	HostDevice	Out	特定の仮想マシンにアタッチされたホストデバイスに関する情報を取得しました。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.270.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.270.2. 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6.845 パラメーターの概要

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

6.271. 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6.846 メソッドの概要

名前	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。

6.271.1. 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。

例:

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

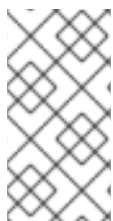
HostDevice タイプのリクエスト本文の例

```
<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6.847 パラメーターの概要

名前	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。

6.271.2. 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。

返されるデバイス一覧の順序は保証されません。

表6.848 パラメーターの概要

名前	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すデバイスの最大数を設定します。

6.271.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.271.2.2. 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。

6.272. VM-MEDIATED-DEVICE

表6.849 メソッドの概要

名前	概要
get	仮想マシンの仲介デバイス設定を取得します。
remove	仲介デバイスを仮想マシンから削除します。
update	仲介デバイスに関する情報を更新します。

6.272.1. get GET

仮想マシンの仲介デバイス設定を取得します。

表6.850 パラメーターの概要

名前	タイプ	方向	概要
device	VmMediated Device	Out	仮想マシンの仲介デバイスに関する情報。
follow	文字列	In	たどる 必要のある内部リンクを指定します。

6.272.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.272.2. remove DELETE

仲介デバイスを仮想マシンから削除します。

表6.851パラメーターの概要

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

6.272.3. update PUT

仲介デバイスに関する情報を更新します。

specParams 要素を使用して情報を更新できます。

たとえば、仲介デバイスを更新するには、次のような要求を送信します。

```
PUT /ovirt-engine/api/vms/123/mediateddevices/00000000-0000-0000-0000-000000000000
<vm_mediated_device>
  <spec_params>
    <property>
      <name>mdevType</name>
      <value>nvidia-11</value>
    </property>
  </spec_params>
</vm_mediated_device>
```

応答本文を使用:

```
<vm_mediated_device href="/ovirt-engine/api/vms/123/mediateddevices/00000000-0000-0000-0000-000000000000" id="00000000-0000-0000-0000-000000000000">
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
  <spec_params>
    <property>
      <name>mdevType</name>
      <value>nvidia-11</value>
```

```

</property>
</spec_params>
</vm_mediated_device>

```

表6.852 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	In	更新を非同期的に実行するかどうかを指定します。
device	VmMediated Device	In/Out	仲介デバイスに関する情報。

6.272.3.1. device

仲介デバイスに関する情報。

リクエストデータには **specParams** プロパティが含まれている必要があります。応答データには、更新された仲介デバイスに関する完全な情報が含まれています。

6.273. VM-MEDIATED DEVICES

仮想マシンの仲介デバイスを管理するサービス。

表6.853 メソッドの概要

名前	概要
add	新しい仲介デバイスを仮想マシンに追加します。
list	仮想マシンの設定済み仲介デバイスをすべて一覧表示します。

6.273.1. add POST

新しい仲介デバイスを仮想マシンに追加します。

表6.854 パラメーターの概要

名前	タイプ	方向	概要
device	VmMediated Device	In/Out	

6.273.2. list GET

仮想マシンの設定済み仲介デバイスをすべて一覧表示します。

返される仲介デバイス一覧の順序は保証されません。

表6.855 パラメーターの概要

名前	タイプ	方向	概要
devices	VmMediated Device[]	Out	仮想マシンの仲介デバイスのリスト。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す仲介デバイスの最大数を設定します。

6.273.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.273.2.2. max

返す仲介デバイスの最大数を設定します。指定されていない場合は、すべての仲介デバイスが返されます。

6.274. VMNIC

表6.856 メソッドの概要

名前	概要
activate	
deactivate	
get	
remove	NIC を削除します。
update	NIC を更新します。

6.274.1. activate POST

表6.857 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.274.2. deactivate POST

表6.858 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.274.3. get GET

表6.859 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
nic	Nic	Out	

6.274.3.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.274.4. 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6.860 パラメーターの概要

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

6.274.5. 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6.861 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.275. VMNICS

表6.862 メソッドの概要

名前	概要
add	仮想マシンに NIC を追加します。
list	仮想マシンの NIC の一覧を返します。

6.275.1. add POST

仮想マシンに NIC を追加します。

次の例では、**virtio** と NIC プロファイル **456** 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 **123** に追加します。

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

```
<nic>
  <name>mynic</name>
  <interface>virtio</interface>
  <vnic_profile id="456"/>
</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>
  <interface>virtio</interface>
  <vnic_profile id="456"/>
</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6.863 パラメーターの概要

名前	タイプ	方向	概要
nic	Nic	In/Out	

6.275.2. list GET

仮想マシンの NIC の一覧を返します。

返される NIC 一覧の順序は保証されません。

表6.864 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す NIC の最大数を設定します。
nics	Nic[]	Out	

6.275.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.275.2.2. 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。

6.276. VMNUMANODE

表6.865 メソッドの概要

名前	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。

6.276.1. get GET

表6.866 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
node	VirtualNuma Node[]	Out	

6.276.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.276.2. 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
```



注記

まず最高位のインデックスから numa ノードを削除する必要があります。

表6.867 パラメーターの概要

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

6.276.3. 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6.868 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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	

6.277. VMNUMANODES

表6.869 メソッドの概要

名前	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。

6.277.1. 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>
  <numa_tune_mode>strict</numa_tune_mode>
</vm_numa_node>
```

表6.870 パラメーターの概要

名前	タイプ	方向	概要
node	VirtualNuma Node[]	In/Out	

6.277.2. list GET

仮想マシンの仮想 NUMA ノードを一覧表示します。

返される NUMA ノードのリストの順序は保証されません。

表6.871 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	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	VirtualNumaNode[]	Out	

6.277.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.277.2.2. 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。

6.278. VMPOOL

仮想マシンプールを管理するためのサービス。

表6.872 メソッドの概要

名前	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。

6.278.1. 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6.873 パラメーターの概要

名前	タイプ	方向	概要
----	-----	----	----

名前	タイプ	方向	概要
async	Boolean	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。

6.278.2. 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6.874 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
pool	VmPool	Out	取得した仮想マシンプール。

6.278.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.278.3. remove DELETE

仮想マシンプールを削除します。

DELETE /ovirt-engine/api/vmpools/123

表6.875 パラメーターの概要

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

6.278.4. 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。

```
<vmPool>
  <name>VM_Pool_B</name>
  <description>Virtual Machine Pool B</description>
  <size>3</size>
  <prestarted_vms>1</size>
  <max_user_vms>2</size>
</vmPool>
```

表6.876 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。
seal	Boolean	In	プール用に作成された仮想マシンを作成後にシールする必要があるかどうかを指定します。

6.278.4.1. seal

プール用に作成された仮想マシンを作成後にシールする必要があ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、プール用に作成された仮想マシンは作成後にシールされます。値が 'false' の場合、仮想マシンはシールされません。パラメーターが提供されていない場合、仮想マシンは、シールされたテンプレートから作成され、そしてゲスト OS が Windows に設定されていない場合にのみ、シールされます。このパラメーターは、プールが更新されたときに作成された仮想マシンにのみ影響します。

たとえば、仮想マシンプールを更新し、作成された追加の仮想マシンをシールするには、次のようなりクエストを送信します。

PUT /ovirt-engine/api/vmpools/123?seal=true

以下のボディを使用します。

```
<vmpool>
  <name>VM_Pool_B</name>
  <description>Virtual Machine Pool B</description>
  <size>7</size>
</vmpool>
```

6.279. 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6.877 メソッドの概要

名前	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。

6.279.1. 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6.878 パラメーターの概要

名前	タイプ	方向	概要
pool	VmPool	In/Out	追加するプール。
seal	Boolean	In	プール用に作成された仮想マシンを作成後にシールする必要があるかどうかを指定します。

6.279.1.1. seal

プール用に作成された仮想マシンを作成後にシールする必要があ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、プール用に作成された仮想マシンは作成後にシールされます。値が 'false' の場合、仮想マシンはシールされません。パラメーターが提供されていない場合、仮想マシンは、シールされたテンプレートから作成され、そしてゲスト OS が Windows に設定されていない場合にのみ、シールされます。このパラメーターは、プールの作成時に作成された仮想マシンにのみ影響します。

たとえば、5つの仮想マシンを含む仮想マシンプールを作成してそれらをシールするには、以下のよう
なリクエストを送信します。

```
POST /ovirt-engine/api/vmpools?seal=true
```

以下のボディを使用します。

```
<vm_pool>
  <name>mypool</name>
  <cluster id="123"/>
  <template id="456"/>
  <size>5</size>
</vm_pool>
```

6.279.2. 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>
```

返されるプールのリストの順序は、**sortby** 句が **search** パラメーターに含まれている場合にのみ保証されます。

表6.879 パラメーターの概要

名前	タイプ	方向	概要
case_sensitive	Boolean	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	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すプールの最大数を設定します。

名前	タイプ	方向	概要
pools	VmPool	Out	取得したプール。
search	文字列	In	返されたプールを制限するために使用されるクエリ文字列。

6.279.2.1. 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。

6.279.2.2. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.279.2.3. 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。

6.280. VMREPORTEDDEVICE

表6.880 メソッドの概要

名前	概要
get	

6.280.1. get GET

表6.881 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
reported_device	ReportedDevice	Out	

6.280.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.281. VMREPORTEDDEVICES

表6.882 メソッドの概要

名前	概要
list	仮想マシンの報告されたデバイスの一覧を返します。

6.281.1. list GET

仮想マシンの報告されたデバイスの一覧を返します。

返されるデバイス一覧の順序は保証されません。

表6.883 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すデバイスの最大数を設定します。
reported_device	ReportedDevice[]	Out	

6.281.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.281.1.2. 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。

6.282. VMSESSION

表6.884 メソッドの概要

名前	概要
get	

6.282.1. get GET

表6.885 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。

名前	タイプ	方向	概要
session	Session	Out	

6.282.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.283. 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6.886 メソッドの概要

名前	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。

6.283.1. 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>
```

返されるセッション一覧の順序は保証されません。

表6.887 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すセッションの最大数を設定します。

名前	タイプ	方向	概要
sessions	Session[]	Out	

6.283.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.283.1.2. 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。

6.284. VMWATCHDOG

仮想マシンのウォッチドッグを管理するサービス。

表6.888 メソッドの概要

名前	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。

6.284.1. get GET

ウォッチドッグに関する情報を返します。

表6.889 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
watchdog	Watchdog	Out	ウォッチドッグに関する情報です。

6.284.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.284.1.2. 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>
```

6.284.2. 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6.890 パラメーターの概要

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

6.284.3. 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6.891 パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.284.3.1. 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。

6.285. VMWATCHDOGS

仮想マシンのウォッチドッグを一覧表示します。

表6.892 メソッドの概要

名前	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。

6.285.1. 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6.893 パラメーターの概要

名前	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。

6.285.1.1. 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。

6.285.2. list GET

仮想マシンのウォッチドッグのリスト。

返されるウォッチドッグ一覧の順序は保証されません。

表6.894 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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。

6.285.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#) を参照してください。

6.285.2.2. 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。

6.285.2.3. 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>
  </watchdog>
</watchdogs>
```

```
<model>i6300esb</model>
</watchdog>
</watchdogs>
```

6.286. VMS

表6.895 メソッドの概要

名前	概要
add	新しい仮想マシンを作成します。
list	システムの仮想マシンの一覧を返します。

6.286.1. 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6.896 パラメーターの概要

名前	タイプ	方向	概要
auto_pinning_policy	AutoPinning Policy	In	自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。
clone	Boolean	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	Boolean	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。

名前	タイプ	方向	概要
filter	Boolean	In	管理者ユーザーにのみ関連しています。
seal	Boolean	In	作成後に仮想マシンをシールするかどうかを指定します。
vm	Vm	In/Out	

6.286.1.1. auto_pinning_policy

自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。



重要

エンジンのバージョン 4.5 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。代わりに、POST に続いて [追加操作](#) を使用してください。

リクエストの例:

```
POST /ovirt-engine/api/vms?auto_pinning_policy=existing/adjust
```

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>
  <placement_policy>
    <hosts>
      <host>
        <name>myhost</name>
      </host>
    </hosts>
  </placement_policy>
</vm>
```

6.286.1.2. 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。

6.286.1.3. 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>
```

6.286.1.4. filter

管理者ユーザーにのみ関連しています。作成された仮想マシンでこのユーザーに UserVmManager ロールを割り当てるかどうかを示します。これにより、ユーザーは後で管理者権限を放棄して (filter=true を指定することにより)、非管理者ユーザーであるかのように仮想マシンにアクセスできるようになります。



注記

既存の仮想マシンに対する admin-as-user (filter=true を指定) POST リクエストは、以前に管理者がユーザーとして (filter=true を使用して) 作成した仮想マシン以外では失敗します。

6.286.1.5. seal

作成後に仮想マシンをシール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、仮想マシンは作成後にシールされます。値が 'false' の場合、仮想マシンはシールされません。パラメーターが提供されていない場合、仮想マシンは、シールされたテンプレートから作成され、そしてゲスト OS が Windows に設定されていない場合にのみ、シールされます。

たとえば、**mytemplate** テンプレートから仮想マシンを作成してシールするには、以下のようなリクエストを送信します。

```
POST /ovirt-engine/api/vms?seal=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>
```

6.286.2. list GET

システムの仮想マシンの一覧を返します。

返される仮想マシンのリストの順序は、**orderby** 句が **search** パラメーターに含まれている場合にのみ保証されます。

表6.897 パラメーターの概要

名前	タイプ	方向	概要
all_content	Boolean	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	Boolean	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	Boolean	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。

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返すことのできる結果の最大数。
ovf_as_ova	Boolean	In	結果が、その VM の OVA ファイルに表示されるように OVF を公開する必要があるかどうかを示します。
search	文字列	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	

6.286.2.1. 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。

6.286.2.2. 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。

6.286.2.3. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.286.2.4. ovf_as_ova

結果が、その VM の OVA ファイルに表示されるように OVF を公開する必要があるかどうかを示します。仮想マシンを記述する OVF ドキュメント。このパラメーターは、all_content=True が設定されている場合にのみ機能します。OVF は **initialization.configuration.data** に提示されます。

以下はその例です。

```
GET /vms?all_content=true&ovf_as_ova=true
```

6.287. VNICPROFILE

このサービスは、vNIC プロファイルを管理します。

表6.898 メソッドの概要

名前	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。

6.287.1. get GET

vNIC プロファイルの詳細を取得します。

表6.899 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
profile	VnicProfile	Out	

6.287.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.287.2. remove DELETE

vNIC プロファイルを削除します。

表6.900 パラメーターの概要

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

名前	タイプ	方向	概要
----	-----	----	----

6.287.3. update PUT

vNIC プロファイルの詳細を更新します。

表6.901パラメーターの概要

名前	タイプ	方向	概要
async	Boolean	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。

6.288. 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6.902 メソッドの概要

名前	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。

6.288.1. 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。



注記

外部ネットワーク用に自動作成された vNIC プロファイルには、ネットワークフィルターがありません。

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。以下はその例です。

```
<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。以下はその例です。

```
<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。以下はその例です。

```
<vnic_profile>
  <name>user_choice_network_filter</name>
  <network_filter id="0000001b-001b-001b-001b-0000000001d5"/>
</vnic_profile>
```

表6.903 パラメーターの概要

名前	タイプ	方向	概要
profile	VnicProfile	In/Out	追加される vNIC プロファイル。

6.288.2. list GET

すべての vNIC プロファイルを一覧表示します。

返される vNIC プロファイル一覧の順序は保証されません。

表6.904 パラメーターの概要

名前	タイプ	方向	概要
follow	文字列	In	たどる 必要のある内部リンクを指定します。
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。

6.288.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.288.2.2. 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。

6.289. 重み

表6.905 メソッドの概要

名前	概要
get	
remove	

6.289.1. get GET

表6.906 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
weight	重み	Out	

6.289.1.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.289.2. remove DELETE

表6.907 パラメーターの概要

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

6.290. 重み

表6.908 メソッドの概要

名前	概要
add	指定したユーザー定義のスケジューリングポリシーに重みを追加します。
list	重みの一覧を返します。

6.290.1. add POST

指定したユーザー定義のスケジューリングポリシーに重みを追加します。

表6.909 パラメーターの概要

名前	タイプ	方向	概要
weight	重み	In/Out	

6.290.2. list GET

重みの一覧を返します。

返される重みのリストの順序は保証されません。

表6.910 パラメーターの概要

名前	タイプ	方向	概要
filter	Boolean	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。
follow	文字列	In	たどる 必要のある内部リンクを指定します。
max	Integer	In	返す重みの最大数を設定します。
weights	Weight[]	Out	

6.290.2.1. follow

たどる 必要のある内部リンクを指定します。これらのリンクで参照されるオブジェクトは、現在の要求の一部としてフェッチされます。詳細は、[こちら](#)を参照してください。

6.290.2.2. 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

Gluster ボリュームでサポートされているアクセスプロトコルを表します。**gluster** と **nfs** はデフォルトで有効になっています。

表7.1 値の概要

名前	概要
cifs	CIFS アクセスプロトコル。
gluster	Gluster アクセスプロトコル。
nfs	NFS アクセスプロトコル。

7.2. ACTION 構造体

表7.2 属性の概要

名前	タイプ	概要
activate	Boolean	
allow_partial_import	Boolean	
async	Boolean	
attachment	DiskAttachment	
authorized_key	AuthorizedKey	
auto_pinning_policy	AutoPinningPolicy	
bricks	GlusterBrick[]	
certificates	Certificate[]	
check_connectivity	Boolean	
clone	Boolean	

名前	タイプ	概要
clone_permissions	Boolean	
cluster	クラスター	
collapse_snapshots	Boolean	
comment	文字列	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。
commit_on_success	Boolean	
connection	StorageConnection	
connectivity_timeout	Integer	
correlation_id	文字列	
data_center	DataCenter	
deploy_hosted_engine	Boolean	
description	文字列	プレーンテキストでの人間が判読できる説明。
details	GlusterVolumeProfileDetails	
directory	文字列	
discard_snapshots	Boolean	
discovered_targets	IscsiDetails[]	
disk	ディスク	
disk_profile	DiskProfile	
disks	Disk[]	

名前	タイプ	概要
exclusive	Boolean	
fault	異常	
fence_type	文字列	
filename	文字列	
filter	Boolean	
fix_layout	Boolean	
follow	文字列	
force	Boolean	
grace_period	GracePeriod	
ホスト	ホスト	
id	文字列	一意の ID
image	文字列	
image_transfer	ImageTransfer	
import_as_template	Boolean	
is_attached	Boolean	
iscsi	IscsiDetails[]	
iscsi_targets	String[]	
job	Job	
lease	StorageDomainLe ase	
logical_units	LogicalUnit[]	
maintenance_after_restart	Boolean	

名前	タイプ	概要
maintenance_enabled	Boolean	
migrate_vms_in_affinity_closure	Boolean	
modified_bonds	HostNic[]	
modified_labels	NetworkLabel[]	
modified_network_attachments	NetworkAttachment[]	
name	文字列	人間が判読できるプレーンテキストでの名前。
optimize_cpu_settings	Boolean	
option	オプション	
pause	Boolean	
permission	パーミッション	
power_management	PowerManagement	
proxy_ticket	ProxyTicket	
quota	クォータ	
reason	文字列	
reassign_bad_macs	Boolean	
reboot	Boolean	
registration_configuration	RegistrationConfiguration	
remote_viewer_connection_file	文字列	

名前	タイプ	概要
removed_bonds	HostNic[]	
removed_labels	NetworkLabel[]	
removed_network_attachments	NetworkAttachment[]	
resolution_type	文字列	
restore_memory	Boolean	
root_password	文字列	
seal	Boolean	
snapshot	スナップショット	
source_host	ホスト	
ssh	Ssh	
status	文字列	
stop_gluster_service	Boolean	
storage_domain	StorageDomain	
storage_domains	StorageDomain[]	
succeeded	Boolean	
synchronized_network_attachments	NetworkAttachment[]	
template	Template	
ticket	Ticket	
timeout	Integer	
undeploy_hosted_engine	Boolean	

名前	タイプ	概要
upgrade_action	ClusterUpgradeAction	
upgrade_percent_complete	Integer	
use_cloud_init	Boolean	
use_ignition	Boolean	
use_initialization	Boolean	
use_sysprep	Boolean	
virtual_functions_configuration	HostNicVirtualFunctionsConfiguration	
vm	Vm	
vnic_profile_mappings	VnicProfileMapping[]	
volatile	Boolean	

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

名前	タイプ	概要
broken	Boolean	アフィニティーグループが壊れているかどうかを指定します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
enforcing	Boolean	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。

名前	タイプ	概要
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	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
positive	Boolean	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。
priority	10 進数	アフィニティーグループの優先度。
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. broken

アフィニティーグループが壊れているかどうかを指定します。ルールのいずれかが満たされない場合、アフィニティーグループは壊れていると見なされます。壊れたフィールドは、エンジンのコンピュートフィールドです。そのため、このフィールドは GET リクエストでのみ使用できます。

7.3.2. 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。

7.3.3. 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

名前	タイプ	概要
cluster	クラスター	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。
host_labels	AffinityLabel[]	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ラベルのリスト。
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。
vm_labels	AffinityLabel[]	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ラベルのリスト。
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

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
has_implicit_affinity_group	Boolean	このプロパティは、ラベルの従来の動作を有効にします。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
read_only	Boolean	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. has_implicit_affinity_group

このプロパティは、ラベルの従来の動作を有効にします。**true** の場合、ラベルは仮想マシンからホストへの positive enforcing アフィニティグループとして動作します。

このパラメーターは、互換性のあるバージョン 4.3 以下のクラスターでのみ使用されます。

7.4.2. 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

名前	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

名前	タイプ	概要
enabled	Boolean	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	Boolean	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	Boolean	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

アフィニティーグループが、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。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

アフィニティーグループが、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。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

名前	タイプ	概要
address	文字列	フェンスエージェントのアドレス。
comment	文字列	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	Boolean	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	文字列	プレーンテキストでの人間が判読できる説明。
encrypt_options	Boolean	オプションを暗号化するかどうかを指定します。
id	文字列	一意の ID
name	文字列	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	文字列	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	文字列	フェンスエージェントのタイプ。
username	文字列	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

名前	タイプ	概要
ホスト	ホスト	ホストサービスへの参照。

7.6.1. ホスト

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

非推奨エージェントの設定オプション。

Red Hat Virtualization 4.4.0 以降、OpenStack Neutron エージェントのデプロイメントが削除されたため、無視されます。OpenStack ホストのデプロイメントは、Red Hat OpenStack Platform Director または TripleO によって実行できます。

表7.10 属性の概要

名前	タイプ	概要
address	文字列	
broker_type	MessageBrokerType	
network_mappings	文字列	Red Hat Virtualization 4 以降、Open vSwitch インターフェイスのマッピングは VDSM によって管理されるため、使用はお勧めしません。
password	文字列	
port	Integer	
username	文字列	

7.7.1. network_mappings

Red Hat Virtualization 4.2.0 以降、Open vSwitch インターフェイスのマッピングは VDSM によって管理されるため、使用はお勧めしません。

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

名前	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	Date	この情報が生成された日時。

表7.12 リンクの概要

名前	タイプ	概要
<code>authenticated_user</code>	User	認証されたユーザーへの参照。
<code>effective_user</code>	User	実効ユーザーへの参照。

7.8.1. `authenticated_user`

認証されたユーザーへの参照。

認証されたユーザーは、現在の要求を受け入れるためにクレデンシャルが検証されたユーザーです。システムの現在のバージョンでは、認証されたユーザーと実効ユーザーは常に同じです。今後、ユーザー偽装に対するサポートが導入されると、同じではなくなる可能性があります。

7.8.2. `effective_user`

実効ユーザーへの参照。

実効ユーザーは、現在のリクエスト中にそのパーミッションが適用されるユーザーです。システムの現在のバージョンでは、認証されたユーザーと実効ユーザーは常に同じです。今後、ユーザー偽装に対するサポートが導入されると、同じではなくなる可能性があります。

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7.13 属性の概要

名前	タイプ	概要
<code>hosts</code>	ApiSummaryItem	ホストの概要。
<code>storage_domains</code>	ApiSummaryItem	ストレージドメインの概要。
<code>users</code>	ApiSummaryItem	ユーザーの概要。
<code>vms</code>	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7.14 属性の概要

名前	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

仮想マシンにインストールされたアプリケーションを表します。仮想マシンのオペレーティングシステムにアプリケーションをデプロイすると、ゲストエージェントによってアプリケーションが報告されます。

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/vms/123/applications/456
```

結果は以下のようになります。

```
<application href="/ovirt-engine/api/vms/123/applications/456" id="456">
  <name>application-test-1.0.0-el7</name>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
</application>
```

表7.15 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.16 リンクの概要

名前	タイプ	概要
vm	Vm	アプリケーションがインストールされている仮想マシンへの参照。

7.12. アーキテクチャー ENUM

表7.17 値の概要

名前	概要
aarch64	AARCH64 CPU アーキテクチャー。
ppc64	
s390x	IBM S390X CPU アーキテクチャー。
undefined	
x86_64	

7.12.1. s390x

IBM S390X CPU アーキテクチャー。

S390X アーキテクチャーで実行されている仮想マシンおよびクラスターに対して指定する必要があります。

S390 は、一般的なマシンアーキテクチャーまたはその 31 ビットバリエーションのいずれかを記述するために、あいまいな方法で使用されることが多いことに注意してください。S390X は、X86_64 や PPC64 などの他のアーキテクチャーと一致する 64 ビットアーキテクチャー専用で使用されます。

7.13. AUTHORIZEDKEY 構造体

表7.18 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
鍵 (key)	文字列	
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.19 リンクの概要

名前	タイプ	概要
user	User	

7.14. AUTONUMASTATUS ENUM

表7.20 値の概要

名前	概要
disable	
enable	
unknown	

7.15. AUTOPINNINGPOLICY ENUM

CPU および NUMA ピニングポリシーを表すタイプ。



重要

エンジンのバージョン 4.5 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後削除されます。代わりに、CpuPinningPolicy を使用してください。

表7.21 値の概要

名前	概要
adjust	CPU と NUMA ピニングは、専用ホストによって設定されます。
disabled	CPU と NUMA ピニングは計算されません。
existing	CPU と NUMA ピニングは、仮想マシンの現在の状態によって設定されます。

7.15.1. adjust

CPU と NUMA ピニングは、専用ホストによって設定されます。

現時点では、CPU と NUMA ピニングが専用のホスト CPU トポロジーを使用することを意味します。仮想マシンの設定は、可能な限り最高のパフォーマンスが得られるように、ホストに合わせて自動的に設定されます。

7.15.2. disabled

CPU と NUMA ピニングは計算されません。

現時点では、CPU と NUMA ピニングが現在の仮想マシン設定に対して計算されないことを意味します。デフォルトでは、仮想マシントポロジーは1ソケット、1コア、および1スレッドに設定されています。

7.15.3. existing

CPU と NUMA ピニングは、仮想マシンの現在の状態によって設定されます。

現時点では、CPU と NUMA ピニングが提供された仮想マシンの CPU トポロジーを使用することを意味します。CPU トポロジーが指定されていない場合は、エンジンのデフォルト (1 ソケット、1 コア、および 1 スレッドで設定された仮想マシントポロジー) が使用されます。

7.16. BACKUP 構造体

表7.22 属性の概要

名前	タイプ	概要
comment	文字列	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	Date	バックアップ作成日。
description	文字列	プレーンテキストでの人間が判読できる説明。
from_checkpoint_id	文字列	増分バックアップを開始するチェックポイント ID。
id	文字列	一意の ID
modification_date	Date	バックアップの変更日。
name	文字列	人間が判読できるプレーンテキストでの名前。
phase	BackupPhase	バックアップ操作のフェーズ。
to_checkpoint_id	文字列	このバックアップ操作によって作成されたチェックポイント ID。

7.16.1. to_checkpoint_id

このバックアップ操作によって作成されたチェックポイント ID。この ID は、次の増分バックアップで `fromCheckpointId` として使用できます。

表7.23 リンクの概要

名前	タイプ	概要
disks	Disk[]	仮想マシンのバックアップに含まれるディスクのリスト。
ホスト	ホスト	バックアップの開始に使用されたホスト。
snapshot	スナップショット	バックアップがスナップショットを使用している場合は、作成されたスナップショットへの参照。
vm	Vm	バックアップに関連付けられた仮想マシンへの参照。

7.17. BACKUPPHASE ENUM

表7.24 値の概要

名前	概要
failed	バックアップに失敗したことを示す最終フェーズ。
finalizing	このフェーズでは、バックアップを完了して関連するディスクのロックを解除するために、バックアップで <code>stop_backup</code> 操作が呼び出されます。
initializing	バックアップの初期段階。
ready	このフェーズは、関連するディスクのバックアップ URL の使用準備が整い、イメージ転送を使用してダウンロードされていることを意味します。
starting	フェーズは、 <code>vdsm/libvirt</code> で <code>start_backup</code> 操作を呼び出す前に設定されます (つまり、フローを完了するには <code>stop_backup</code> を呼び出す必要があります)。
succeeded	バックアップが正常に終了したことを示す最終フェーズ。

7.17.1. initializing

バックアップの初期段階。エンティティ作成時に設定されます。

7.18. BALANCE 構造体

表7.25 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.26 リンクの概要

名前	タイプ	概要
scheduling_policy	SchedulingPolicy	
scheduling_policy_unit	SchedulingPolicyUnit	

7.19. BIOS 構造体

表7.27 属性の概要

名前	タイプ	概要
Boot Menu	BootMenu	
type	BiosType	チップセットと BIOS タイプの組み合わせ。

7.20. BIOSTYPE ENUM

チップセットと BIOS タイプの組み合わせを表すタイプ。

表7.28 値の概要

名前	概要
cluster_default	クラスター全体のデフォルトを使用します。
i440fx_sea_bios	SeaBIOS を搭載した i440fx チップセット。
q35_ovmf	OVMF (UEFI) BIOS を搭載した q35 チップセット。
q35_sea_bios	SeaBIOS を搭載した q35 チップセット。
q35_secure_boot	SecureBoot が有効な OVMF (UEFI) BIOS を搭載した q35 チップセット。

7.20.1. cluster_default

クラスター全体のデフォルトを使用します。

この値はクラスターには使用できません。

7.20.2. i440fx_sea_bios

SeaBIOS を搭載した i440fx チップセット。

x86 以外のアーキテクチャーでは、これが許可されている唯一のデフォルト以外の値です。

7.21. BLOCKSTATISTIC 構造体

表7.29 属性の概要

名前	タイプ	概要
statistics	Statistic[]	

7.22. 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7.30 属性の概要

名前	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。

7.22.1. 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。

7.22.2. 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。

7.22.3. 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7.31 リンクの概要

名前	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。

7.22.4. 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。

以下はその例です。

```
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>
```

7.23. BOOKMARK 構造体

システム内のブックマークを表します。

表7.32 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
value	文字列	エンジン内の検索を表すブックマーク値。

7.24. BOOT 構造体

仮想マシンのブートシーケンスの設定。

表7.33 属性の概要

名前	タイプ	概要
devices	BootDevice[]	起動デバイスの順序付きリスト。

7.24.1. devices

起動デバイスの順序付きリスト。仮想マシンは、指定された起動デバイスから指定された順序で起動しようとしています。

7.25. BOOTDEVICE ENUM

仮想マシンが起動できるデバイスの種類を表します。

表7.34 値の概要

名前	概要
cdrom	CD-ROM から起動します。
hd	ハードドライブから起動します。
network	PXE を使用して、ネットワークから起動します。

7.25.1. cdrom

CD-ROM から起動します。CD-ROM は、仮想マシンが属する ata センターにアタッチされた ISO ドメインで使用可能な ISO ファイルのリストから選択できます。

7.25.2. network

PXE を使用して、ネットワークから起動します。仮想マシンが接続されているネットワーク上で PXE が設定されている必要があります。

7.26. BOOTMENU STRUCT

仮想マシンとテンプレートの起動メニュー設定を表します。

表7.35 属性の概要

名前	タイプ	概要
enabled	Boolean	この仮想マシン (またはテンプレート) に対して起動メニューが有効になっているかどうかを示します。

7.27. 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7.36 値の概要

名前	概要
autoconf	ステートレスアドレスの自動設定。
dhcp	Dynamic host configuration protocol (DHCP)
none	アドレス設定なし。
poly_dhcp_autoconf	ステートレスアドレス自動設定 (SLAAC) と併用する DHCP。
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。

名前	概要
----	----

7.27.1. 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 Wipedia 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。

7.27.2. dhcp

Dynamic host configuration protocol (DHCP)

詳細は、[こちらの Wipedia の記事](#) を参照してください。

7.27.3. poly_dhcp_autoconf

ステートレスアドレス自動設定 (SLAAC) と併用する DHCP。

SLAAC メカニズムは [RFC 4862](#) で定義されています。詳細は、[Stateless address auto-configuration](#) の記事および [DHCP](#) アーティクル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。

7.28. BRICKPROFILEDETAIL 構造体

表7.37 属性の概要

名前	タイプ	概要
<code>profile_details</code>	ProfileDetail[]	

表7.38 リンクの概要

名前	タイプ	概要
<code>brick</code>	GlusterBrick	

7.29. CDROM 構造体

表7.39 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
file	ファイル	
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.40 リンクの概要

名前	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。

7.29.1. 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。

7.30. CERTIFICATE 構造体

表7.41 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

名前	タイプ	概要
organization	文字列	
サブジェクト (subject)	文字列	

7.31. CHECKPOINT 構造体

表7.42 属性の概要

名前	タイプ	概要
comment	文字列	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	Date	チェックポイントの作成日。
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
parent_id	文字列	親チェックポイント ID。
state	CheckpointState	チェックポイントの状態。

表7.43 リンクの概要

名前	タイプ	概要
disks	Disk[]	バックアップチェックポイントに含まれるディスクのリスト。
vm	Vm	チェックポイントに関連付けられた仮想マシンへの参照。

7.32. CHECKPOINTSTATE ENUM

表7.44 値の概要

名前	概要
created	チェックポイントの初期状態。
invalid	INVALID 状態は、チェックポイントを増分バックアップに使用できなくなり、削除する必要がある場合に設定されます (たとえば、古い仮想マシンスナップショットにコミットした後)。

7.32.1. created

チェックポイントの初期状態。エンティティ作成時に設定されます。

7.33. CLOUDINIT STRUCT

cloud-init 設定を指定する非推奨のタイプ。

このタイプは非推奨となり、[initialization](#) タイプ内の代替属性に置き換えられました。詳細は、[cloud_init](#) 属性のドキュメントを参照してください。

表7.45 属性の概要

名前	タイプ	概要
authorized_keys	AuthorizedKey[]	
files	File[]	
ホスト	ホスト	
network_configuration	NetworkConfiguration	
regenerate_ssh_keys	Boolean	
timezone	文字列	
users	User[]	

7.34. CLOUDINITNETWORKPROTOCOL ENUM

cloud-init プロトコルの値を定義します。このプロトコルは、cloud-init によって処理されるために仮想マシンに渡される前に、cloud-init ネットワークパラメーターがどのようにフォーマットされるかを決定します。

サポートされるプロトコルは、cloud-init のバージョンに依存します。詳細は、[ネットワーク設定ソース](#)を参照してください。

表7.46 値の概要

名前	概要
eni	レガシープロトコル。
openstack_metadata	IPv6などをサポートする ENI プロトコルの後継。

7.34.1. eni

レガシープロトコル。IPv6 には対応していません。詳細は、[ネットワーク設定 ENI\(Legacy\)](#) を参照してください。

7.34.2. openstack_metadata

IPv6 などをサポートする ENI プロトコルの後継。これはデフォルト値です。詳細は、[API: Proxy neutron configuration to guest instance](#) を参照してください。

7.35. CLUSTER 構造体

クラスターのタイプの表現。

クラスターの JSON の表現。

```
{
  "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",
    "firewall_type": "iptables",
    "ha_reservation": "false",
    "ksm": {
      "enabled": "true",
      "merge_across_nodes": "true"
    },
    "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"
  }
},
"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",
```


名前	タイプ	概要
fips_mode	FipsMode	クラスターの FIPS モード。
firewall_type	FirewallType	このクラスター内のホストで使用されるファイアウォールのタイプ。
gluster_service	Boolean	
gluster_tuned_profile	文字列	調整済みプロファイルの名前。
ha_reservation	Boolean	
id	文字列	一意の ID
ksm	Ksm	
log_max_memory_used_threshold	Integer	監査ログイベントを記録するためのメモリー消費のしきい値。
log_max_memory_used_threshold_type	LogMaxMemoryUsedThresholdType	監査ログイベントをログに記録するためのメモリー消費しきい値のタイプ。
maintenance_reason_required	Boolean	このプロパティーは関連性がなくなり、非推奨になりました。
memory_policy	MemoryPolicy	
migration	MigrationOptions	実行中の仮想マシンを別のホストに移行するためのクラスター全体の設定への参照。
name	文字列	人間が判読できるプレーンテキストでの名前。
optional_reason	Boolean	このプロパティーは関連性がなくなり、非推奨になりました。
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。

名前	タイプ	概要
<code>threads_as_cores</code>	Boolean	
<code>trusted_service</code>	Boolean	
<code>tunnel_migration</code>	Boolean	
<code>upgrade_correlation_id</code>	文字列	アップグレード関連識別子。
<code>upgrade_in_progress</code>	Boolean	クラスタのアップグレードが開始されたかどうかを示します。
<code>upgrade_percent_complete</code>	Integer	アップグレードが進行中の場合、報告されたアップグレードの完了率。
<code>version</code>	バージョン	クラスタの互換バージョン。
<code>virt_service</code>	Boolean	
<code>vnc_encryption</code>	Boolean	VNC 暗号化を有効にします。

7.35.1. bios_type

チップセットと BIOS タイプの組み合わせ。

この値は、`biosType` が `CLUSTER_DEFAULT` に設定されているクラスタ内のすべての仮想マシンのデフォルトとして使用されます。

7.35.2. custom_scheduling_policy_properties

クラスタ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にのみ適用されます。

たとえば、クラスタ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。

7.35.3. 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。

以下はその例です。

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

リクエスト本文は以下のようになります。

```

<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>

```

7.35.4. fips_mode

クラスターの FIPS モード。

FIPS モードは、ホストに対するクラスターのポリシーを表します。クラスターに追加されたホストは、クラスターの FIPS モードを満たしているかどうかチェックされ、満たしていない場合は操作不能になります。値が明示的に指定されない限り、新しいクラスターはデフォルトで **UNDEFINED** に初期化されます。この値は、クラスターに追加された最初のホストの FIPS モードに自動的に変更されます。

7.35.5. gluster_tuned_profile

調整済みプロファイルの名前。

クラスター内のすべてのホストに設定するように **調整** されたプロファイル。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。

7.35.6. log_max_memory_used_threshold

監査ロギイベントを記録するためのメモリー消費のしきい値。

パーセンテージの場合、使用メモリーが指定された値を超えると、監査ロギイベントがログに記録されます。絶対値の場合、空きメモリーが MB で指定された値を下回ると、監査ロギイベントがログに記録されます。

7.35.7. log_max_memory_used_threshold_type

監査ロギイベントをログに記録するためのメモリー消費しきい値のタイプ。

percentage と absolute_value_in_mb のいずれかを選択できます。

7.35.8. maintenance_reason_required

このプロパティーは関連性がなくなり、非推奨になりました。デフォルト値は true です。

7.35.9. migration

実行中の仮想マシンを別のホストに移行するためのクラスター全体の設定への参照。



注記

このメソッドが返す ID で移行ポリシーを照会する API はまだ実装されていません。`/ovirt-engine/api/options/MigrationPolicies` を使用して、すべての移行ポリシーとその ID のリストを取得します。

7.35.10. optional_reason

このプロパティーは関連性がなくなり、非推奨になりました。デフォルト値は true です。

7.35.11. 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。

7.35.12. upgrade_correlation_id

アップグレード相関識別子。クラスターのアップグレードの詳細を示すイベントをアップグレード自体に関連付けるために使用します。

7.35.13. 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。

以下はその例です。

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

次のように応答します。

```
<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。

7.35.14. vnc_encryption

VNC 暗号化を有効にします。このプロパティのデフォルト値は false です。

表7.48 リンクの概要

名前	タイプ	概要
affinity_groups	AffinityGroup[]	
cpu_profiles	CpuProfile[]	
data_center	DataCenter	

名前	タイプ	概要
enabled_features	ClusterFeature[]	クラスターで有効になっているカスタム機能。
external_network_providers	ExternalProvider[]	クラスターで利用可能な外部ネットワークプロバイダーへの参照。
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。

7.35.15. external_network_providers

クラスターで利用可能な外部ネットワークプロバイダーへの参照。

外部ネットワークプロバイダーの自動デプロイメントがサポートされる場合には、参照されるネットワークプロバイダーのネットワークはクラスター内のすべてのノードで利用可能になります。クラスターの外部ネットワークプロバイダーは、[クラスターの追加](#) 時にのみ設定できます。この値は [ホストの追加](#) 時に個々のホストに対して上書きされる場合があります。

7.35.16. 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。

7.36. CLUSTERFEATURE 構造体

タイプは、クラスターレベルで使用できる追加機能を表します。

表7.49 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.50 リンクの概要

名前	タイプ	概要
cluster_level	ClusterLevel	クラスターレベルへの参照。

7.37. 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7.51 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	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。

表7.52 リンクの概要

名前	タイプ	概要
cluster_features	ClusterFeature[]	このクラスターレベルでサポートされる追加機能。

7.38. CLUSTERUPGRADEACTION ENUM

クラスターアップグレードアクションのアクションタイプ。

表7.53 値の概要

名前	概要
finish	クラスターの <code>upgrade_running</code> フラグを <code>false</code> にマークして、クラスターのアップグレードプロセスを完了するために渡されるアップグレードアクション。
start	クラスターの <code>upgrade_running</code> フラグを <code>true</code> にマークして、クラスターのアップグレードプロセスを開始するために渡されるアップグレードアクション。
update_progress	クラスターのアップグレードの進行状況を更新するために渡されるアップグレードアクション。

7.38.1. finish

クラスターの `upgrade_running` フラグを `false` にマークして、クラスターのアップグレードプロセスを完了するために渡されるアップグレードアクション。これは、クラスターのアップグレードプロセスの最後に使用する必要があります。

7.38.2. start

クラスターの `upgrade_running` フラグを `true` にマークして、クラスターのアップグレードプロセスを開始するために渡されるアップグレードアクション。これは、クラスターのアップグレードプロセスの開始時に使用する必要があります。

7.38.3. update_progress

クラスターのアップグレードの進行状況を更新するために渡されるアップグレードアクション。これは、アップグレードの進行に合わせて使用する必要があります。

7.39. CONFIGURATION 構造体

表7.54 属性の概要

名前	タイプ	概要
data	文字列	仮想マシンを記述するドキュメント。
type	<code>ConfigurationType</code>	

7.39.1. 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>

```

7.40. CONFIGURATIONTYPE ENUM

設定フォーマットのタイプ。

表7.55 値の概要

名前	概要
ova	標準 OVF タイプの ConfigurationType。
ovf	oVirt 互換 OVF タイプの ConfigurationType。

7.40.1. ova

標準 OVF タイプの ConfigurationType。

提供される仮想マシン設定は、Open Virtualization Format (OVF) 標準に準拠しています。この値は、oVirt または他のベンダーによって生成された Open Virtual Appliance (OVA) から抽出された OVF 設定に使用する必要があります。[OVF 仕様](#) を参照してください。

7.40.2. ovf

oVirt 互換 OVF タイプの ConfigurationType。

提供される仮想マシン設定は、Open Virtualization Format (OVF) の oVirt 互換形式に準拠しています。OVF の oVirt 互換形式は、他のベンダーが使用する OVF 標準とは異なる場合があることに注意してください。この値は、ストレージドメインから取得される OVF 設定に使用する必要があります。

7.41. CONSOLE 構造体

シリアルコンソールデバイスの表現。

表7.56 属性の概要

名前	タイプ	概要
enabled	Boolean	シリアルコンソールデバイスを有効/無効にします。

7.42. CORE 構造体

表7.57 属性の概要

名前	タイプ	概要
index	Integer	
socket	Integer	

7.43. CPU 構造体

表7.58 属性の概要

名前	タイプ	概要
architecture	アーキテクチャー	
cores	Core[]	
cpu_tune	CpuTune	
level	Integer	
mode	CpuMode	
name	文字列	
speed	10 進数	
topology	CpuTopology	
type	文字列	

7.44. CPUMODE ENUM

表7.59 値の概要

名前	概要
custom	
host_model	
host_passthrough	

7.45. CPUPINNINGPOLICY ENUM

CPU および NUMA ピニングポリシーを表すタイプ。

表7.60 値の概要

名前	概要
dedicated	CPU ピニングは、仮想マシン起動時にエンジンによって自動的に計算され、仮想マシン停止時に削除されます。
isolate_threads	CPU ピニングは、仮想マシン起動時にエンジンによって自動的に計算され、仮想マシン停止時に削除されます。
manual	CPU ピニングは手動で設定されます。
none	CPU ピニングは設定されません。
resize_and_pin_numa	CPU と NUMA ピニングは、専用ホストによって設定されます。

7.45.1. dedicated

CPU ピニングは、仮想マシン起動時にエンジンによって自動的に計算され、仮想マシン停止時に削除されます。

ピンニングは排他的です。つまり、他の仮想マシンはピンニングされた物理 CPU を使用できません。

7.45.2. isolate_threads

CPU ピニングは、仮想マシン起動時にエンジンによって自動的に計算され、仮想マシン停止時に削除されます。

ピンニングは排他的で、各仮想スレッドは排他的な物理コアを取得します。これは、ピンニングされた物理 CPU を他の仮想マシンが使用できないことを意味します。

7.45.3. manual

CPU ピニングは手動で設定されます。

これは、現時点で CPU ピニングが現在の仮想マシン設定に対して手動で設定されることを意味します。仮想マシンは、少なくとも1つのホストにピンングされる必要があります。ピンングは、CpuTune を使用して、CPU 設定内で提供されます。

7.45.4. none

CPU ピニングは設定されません。

これは、現時点で CPU ピニングが現在の仮想マシン設定に対して設定されないことを意味します。デフォルトでは、仮想マシントポロジーは1ソケット、1コア、および1スレッドに設定されます。

7.45.5. resize_and_pin_numa

CPU と NUMA ピニングは、専用ホストによって設定されます。

CPU と NUMA ピニングでは、専用のホスト CPU トポロジーが使用されます。仮想マシンの設定は、可能な限り最高のパフォーマンスが得られるように、ホストに合わせて自動的に設定されます。

7.46. CPUPROFILE 構造体

表7.61 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.62 リンクの概要

名前	タイプ	概要
cluster	クラスター	
permissions	Permission[]	
qos	Qos	

7.47. CPUTOPOLOGY 構造体

表7.63 属性の概要

名前	タイプ	概要
cores	Integer	

名前	タイプ	概要
sockets	Integer	
threads	Integer	

7.48. CPUTUNE 構造体

表7.64 属性の概要

名前	タイプ	概要
vcpu_pins	VcpuPin[]	

7.49. CPUTYPE 構造体

サポートされている CPU タイプを記述します。

表7.65 属性の概要

名前	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	文字列	CPU タイプの名前 (例: Intel Nehalem Family)。

7.50. CREATIONSTATUS ENUM

表7.66 値の概要

名前	概要
完了	
failed	
in_progress	
pending	

7.51. CUSTOMPROPERTY 構造体

カスタムプロパティの表現。

表7.67 属性の概要

名前	タイプ	概要
name	文字列	プロパティ名
regexp	文字列	カスタムプロパティが取得できる使用可能な値を定義する正規表現。
value	文字列	プロパティの値。

7.52. DATACENTER 構造体

表7.68 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
local	Boolean	
name	文字列	人間が判読できるプレーンテキストでの名前。
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。

7.52.1. 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。

以下はその例です。

```
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7.69 リンクの概要

名前	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。

7.53. DATACENTERSTATUS ENUM

表7.70 値の概要

名前	概要
contend	
maintenance	
not_operational	
problematic	
uninitialized	
up	

7.54. 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7.71 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.72 リンクの概要

名前	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。

7.54.1. 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。

7.55. DISK 構造体

仮想ディスクデバイスを表します。

表7.73 属性の概要

名前	タイプ	概要
active	Boolean	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	文字列	
backup	DiskBackup	ディスクでサポートされているバックアップ動作。
backup_mode	DiskBackupMode	ディスクバックアップのタイプ (完全/増分)。ディスクバックアップが進行中の場合にのみ表示されます。
bootable	Boolean	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	文字列	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	DiskContentType	ディスク上に存在する実際のコンテンツを示します。
description	文字列	プレーンテキストでの人間が判読できる説明。
external_disk	文字列	外部ディスクを使用します。
format	DiskFormat	基礎となるストレージフォーマット。
id	文字列	一意の ID
image_id	文字列	
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ーのタイプ。
logical_name	文字列	
lun_storage	HostStorage[]	
name	文字列	人間が判読できるプレーンテキストでの名前。

名前	タイプ	概要
propagate_errors	Boolean	ディスクエラーが原因で仮想マシンを一時停止する必要があるかどうか、または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	Boolean	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	SCSI パススルーが有効かどうか、およびそのポリシーを示します。
shareable	Boolean	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	Boolean	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	
total_size	Integer	すべてのスナップショットを含むディスクの合計サイズ (バイト単位)。
uses_scsi_reservation	Boolean	
wipe_after_delete	Boolean	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。

7.55.1. 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ときに、サーバーがこの属性を含まない要求を受け入れる場合、未定義となります。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、それ以外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。

7.55.2. 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、ディスクが実際に使用するバイト数です。**cow** フォーマット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。

7.55.3. 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。

7.55.4. external_disk

外部ディスクを使用します。

外部ディスクは、次のように、ローカルファイルまたはブロックデバイスへのパス、または QEMU でサポートされている URL にできます。

- `nbd:<host>:<port>[:exportname=<export>]`
- `nbd:unix:</path>[:exportname=<export>]`
- `http://[<username>[:<password>]@]<host>/<path>`
- `https://[<username>[:<password>]@]<host>/<path>`
- `ftp://[<username>[:<password>]@]<host>/<path>`
- `ftps://[<username>[:<password>]@]<host>/<path>`

サポートされるプロトコルおよび詳細情報は、QEMU のマニュアルを参照してください。

7.55.5. 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。

7.55.6. 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。

7.55.7. 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。

7.55.8. 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 update 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 ボリュームをサポートします。V4 ストレージドメインは QCOW2v3 もサポートします。異なる QCOW バージョンの機能の詳細は、[QCOW3](#) を参照してください。

7.55.9. read_only

ディスクが読み取り専用モードかどうかを示します。

バージョン 4.0 以降では、この属性は API に表示されず、[DiskAttachment](#) に移動されました。

Red Hat Virtualization Manager のバージョン 4.1.2 以降、この属性は非推奨となり、将来的に削除される予定です。読み取り専用モードでディスクをアタッチするには、[DiskAttachment](#) タイプの **read_only** 属性を使用します。以下はその例です。

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

```
<disk_attachment>
  <read_only>true</read_only>
  ...
</disk_attachment>
```

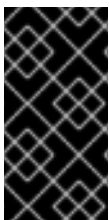
7.55.10. sgio

SCSI パススルーが有効かどうか、およびそのポリシーを示します。

filtered/unfiltered の値を設定すると、非特権/特権 SCSI I/O を備えた LUN ディスクの SCSI パススルーが有効になります。SCSI パススルーを無効にするには、値を **disabled** に設定する必要があります

7.55.11. 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、たとえば [GlusterFS](#) や [GFS](#) などの共有ファイルシステムを使用して、データの破損を回避するためにディスクへのアクセスを調整するのは、接続された仮想マシンのゲストオペレーティングシステムのロールです。

7.55.12. total_size

すべてのスナップショットを含むディスクの合計サイズ (バイト単位)。

合計サイズは、ディスクによって実際に使用されたバイト数にスナップショットのサイズを加えたものです。ダイレクト LUN および Cinder ディスクには設定されません。スナップショットのないディスクの場合、合計サイズは実際のサイズと同じです。

7.55.13. 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ければ削除できません。
- 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7.74 リンクの概要

名前	タイプ	概要
disk_profile	DiskProfile	
disk_snapshots	DiskSnapshot[]	
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。

7.55.14. statistics

ディスクによって公開される統計。以下はその例です。

```

<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>

```

ディスク取得時において、これらの統計は直接含まれず、リンクのみが含まれます。統計を取得するには、含まれているリンクに従ってください。

GET /ovirt-engine/api/disks/123/statistics

7.55.15. 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のみ必要です。

7.55.16. 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。

7.56. 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7.75 属性の概要

名前	タイプ	概要
active	Boolean	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	Boolean	ディスクが起動可能かどうかを定義します。
comment	文字列	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。

名前	タイプ	概要
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	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のタイプ。
logical_name	文字列	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	文字列	人間が判読できるプレーンテキストでの名前。
pass_discard	Boolean	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	Boolean	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	Boolean	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。

7.56.1. 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。

7.56.2. 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。

7.56.3. 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>
```

7.56.4. 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 予約を設定できません。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7.76 リンクの概要

名前	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。

7.57. DISKBACKUP ENUM

ディスクで有効になっているバックアップメカニズムの列挙を表します。

表7.77 値の概要

名前	概要
incremental	増分バックアップのサポート。
none	バックアップのサポートなし。

7.58. DISKBACKUPMODE ENUM

バックアップモードの列挙を表します

表7.78 値の概要

名前	概要
full	このディスクはフルバックアップをサポートしています。

名前	概要
incremental	このディスクは増分バックアップをサポートしています。

7.58.1. full

このディスクはフルバックアップをサポートしています。ゼロエクステントをクエリーして、すべてのディスクデータをダウンロードできます。

7.58.2. incremental

このディスクは増分バックアップをサポートしています。ダーティーエクステントをクエリーして、変更されたブロックをダウンロードできます。

7.59. DISKCONTENTTYPE ENUM

ディスク上にある実際のコンテンツ。

表7.79 値の概要

名前	概要
backup_scratch	ディスクには、保護された VM バックアップデータが含まれています。
data	ディスクにはデータが含まれています。
hosted_engine	ディスクには、ホスト型エンジン VM ディスクが含まれています。
hosted_engine_configuration	ディスクには、Hosted Engine 設定ディスクが含まれています。
hosted_engine_metadata	ディスクには、Hosted Engine メタデータディスクが含まれています。
hosted_engine_sanlock	ディスクには、Hosted Engine Sanlock ディスクが含まれています。
iso	ディスクには、CDROM デバイスで使用される ISO イメージが含まれています。
memory_dump_volume	ディスクには、ライブスナップショットからのメモリーダンプが含まれています。
memory_metadata_volume	ディスクには、ライブスナップショットからのメモリーメタデータが含まれていません。
ovf_store	ディスクは OVF ストアです。

7.60. DISKFORMAT ENUM

ディスクの基礎となるストレージフォーマット。

表7.80 値の概要

名前	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。

7.61. DISKINTERFACE ENUM

コントローラーとのディスク通信の基盤となるストレージインターフェイス。

表7.81 値の概要

名前	概要
ide	レガシーコントローラーデバイス。
sata	SATA コントローラーデバイス。
spapr_vscsi	SCSI プロトコルを使用して、IBM pSeries ファミリーのマシンでサポートされる準仮想化デバイス。
virtio	ゲストのデバイスドライバーだけが仮想環境で実行されていることを認識している仮想化インターフェイス。
virtio-scsi	準仮想化 SCSI コントローラーデバイス。

7.61.1. ide

レガシーコントローラーデバイス。ほぼすべてのゲストオペレーティングシステムで動作するため、互換性に優れています。パフォーマンスは、他の選択肢よりも低くなります。

7.61.2. virtio

ゲストのデバイスドライバーだけが仮想環境で実行されていることを認識している仮想化インターフェイス。ゲストが高性能のディスク操作を取得できるようにします。

7.61.3. virtio_scsi

準仮想化 SCSI コントローラーデバイス。SCSI プロトコルを使用して、ダイレクト物理ストレージデバイスアドレスを経由するゲストとの高速インターフェイス。

7.62. DISKPROFILE STRUCT

表7.82 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.83 リンクの概要

名前	タイプ	概要
permissions	Permission[]	
qos	Qos	
storage_domain	StorageDomain	

7.63. DISKSNAPSHOT 構造体

表7.84 属性の概要

名前	タイプ	概要
active	Boolean	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	文字列	
backup	DiskBackup	ディスクでサポートされているバックアップ動作。
backup_mode	DiskBackupMode	ディスクバックアップのタイプ (完全/増分)。ディスクバックアップが進行中の場合にのみ表示されます。
bootable	Boolean	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	文字列	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	DiskContentType	ディスク上に存在する実際のコンテンツを示します。

名前	タイプ	概要
description	文字列	プレーンテキストでの人間が判読できる説明。
external_disk	文字列	外部ディスクを使用します。
format	DiskFormat	基礎となるストレージフォーマット。
id	文字列	一意の ID
image_id	文字列	
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	文字列	
lun_storage	HostStorage[]	
name	文字列	人間が判読できるプレーンテキストでの名前。
propagate_errors	Boolean	ディスクエラーが原因で仮想マシンを一時停止する必要があるかどうか、または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	Boolean	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	SCSI パススルーが有効かどうか、およびそのポリシーを示します。
shareable	Boolean	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	Boolean	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	

名前	タイプ	概要
total_size	Integer	すべてのスナップショットを含むディスクの合計サイズ (バイト単位)。
uses_scsi_reservation	Boolean	
wipe_after_delete	Boolean	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。

7.63.1. 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ときに、サーバーがこの属性を含まない要求を受け入れる場合、未定義となります。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、それ以外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。

7.63.2. 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、ディスクが実際に使用するバイト数です。cow フォーマット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。

7.63.3. 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。

7.63.4. external_disk

外部ディスクを使用します。

外部ディスクは、次のように、ローカルファイルまたはブロックデバイスへのパス、または QEMU でサポートされている URL にできます。

- `nbd:<host>:<port>[:exportname=<export>]`

- `nbd:unix:</path>[:exportname=<export>]`
- `http://[<username>[:<password>]@]<host>/<path>`
- `https://[<username>[:<password>]@]<host>/<path>`
- `ftp://[<username>[:<password>]@]<host>/<path>`
- `ftps://[<username>[:<password>]@]<host>/<path>`

サポートされるプロトコルおよび詳細情報は、QEMU のマニュアルを参照してください。

7.63.5. 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。

7.63.6. 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。

7.63.7. 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。

7.63.8. qcow_version

QCOW ポリユームの基礎となる QCOW バージョン。QCOW バージョンは、ポリユ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 update API を使用して更新でき、QCOW ポリユ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 ポリユームをサポートします。V4 ストレージドメインは QCOW2v3 もサポートします。異なる QCOW バージョンの機能の詳細は、[QCOW3](#) を参照してください。

7.63.9. read_only

ディスクが読み取り専用モードかどうかを示します。

バージョン 4.0 以降では、この属性は API に表示されず、[DiskAttachment](#) に移動されました。

Red Hat Virtualization Manager のバージョン 4.1.2 以降、この属性は非推奨となり、将来的に削除される予定です。読み取り専用モードでディスクをアタッチするには、[DiskAttachment](#) タイプの `read_only` 属性を使用します。以下はその例です。

■

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

```
<disk_attachment>
  <read_only>true</read_only>
  ...
</disk_attachment>
```

7.63.10. sgio

SCSI パススルーが有効かどうか、およびそのポリシーを示します。

filtered/unfiltered の値を設定すると、非特権/特権 SCSI I/O を備えた LUN ディスクの SCSI パススルーが有効になります。SCSI パススルーを無効にするには、値を **disabled** に設定する必要があります

7.63.11. 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、たとえば [GlusterFS](#) や [GFS](#) などの共有ファイルシステムを使用して、データの破損を回避するためにディスクへのアクセスを調整するのは、接続された仮想マシンのゲストオペレーティングシステムのロールです。

7.63.12. total_size

すべてのスナップショットを含むディスクの合計サイズ (バイト単位)。

合計サイズは、ディスクによって実際に使用されたバイト数にスナップショットのサイズを加えたものです。ダイレクト LUN および Cinder ディスクには設定されません。スナップショットのないディスクの場合、合計サイズは実際のサイズと同じです。

7.63.13. 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ければ削除できません。
- 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7.85 リンクの概要

名前	タイプ	概要
disk	ディスク	
disk_profile	DiskProfile	
disk_snapshots	DiskSnapshot[]	

名前	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。
openstack_volume_type	OpenStackVolumeType	
parent	DiskSnapshot	親ディスクのスナップショット。
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。

7.63.14. statistics

ディスクによって公開される統計。以下はその例です。

```
<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>
```


ディスク取得時において、これらの統計は直接含まれず、リンクのみが含まれます。統計を取得するには、含まれているリンクに従ってください。

```
GET /ovirt-engine/api/disks/123/statistics
```

7.63.15. 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のみ必要です。

7.63.16. 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。

7.64. DISKSTATUSENUM

ディスクの現在のステータス表現。

表7.86 値の概要

名前	概要
illegal	仮想マシンはディスクにアクセスできず、ユーザーは問題を解決するためのアクションを実行する必要があります。
locked	ディスクはシステムによって使用されているため、この時点では仮想マシンからアクセスできません。
ok	ディスクのステータスは正常で、仮想マシンからアクセスできます。

7.64.1. locked

ディスクはシステムによって使用されているため、この時点では仮想マシンからアクセスできません。これは通常、ディスクが解放されるまでの一時的なステータスです。

7.65. DISKSTORAGETYPE ENUM

表7.87 値の概要

名前	概要
cinder	
image	

名前	概要
lun	
managed_block_storage	cinderlib ドライバーを使用して作成されたストレージドメインに使用されるストレージタイプ。

7.66. DISKTYPE ENUM

表7.88 値の概要

名前	概要
data	
system	

7.67. DISPLAY 構造体

グラフィックコンソール設定を表します。

表7.89 属性の概要

名前	タイプ	概要
address	文字列	グラフィックコンソールクライアントを接続するゲストの IP アドレス。
allow_override	Boolean	ホストごとに表示アドレスをオーバーライドするかどうかを示します。
証明書 (certificate)	証明書	TLS 接続の場合の TLS 証明書。
copy_paste_enabled	Boolean	ユーザーがコンテンツを外部ホストからグラフィックコンソールにコピーアンドペーストできるかどうかを示します。
disconnect_action	文字列	グラフィックコンソールが切断されたときに実行されるアクションを返します。
disconnect_action_delay	Integer	グラフィックコンソールの切断アクションが実行されるまでの遅延 (分単位)。
file_transfer_enabled	Boolean	ユーザーが外部ホストからグラフィックコンソールにファイルをドラッグアンドドロップできるかどうかを示します。

名前	タイプ	概要
keyboard_layout	文字列	このグラフィックコンソールで使用するキーボードレイアウト。
monitors	Integer	このグラフィックコンソール用に開いているモニターの数。
port	Integer	グラフィックコンソールクライアントを接続するゲストのポートアドレス。
proxy	文字列	グラフィックコンソールクライアントがゲストに接続するために使用するプロキシ IP。
secure_port	Integer	TLS を使用する場合に、グラフィックコンソールクライアントを接続するための、ゲスト上の保護されたポートアドレス。
single_qxl_pci	Boolean	エンジンは、オペレーティングシステムに従って自動的に設定するようになりました。
Smartcard Enabled	Boolean	スマートカード認証を使用するかどうかを示します。
type	DisplayType	グラフィックコンソールプロトコルのタイプ。

7.67.1. allow_override

ホストごとに表示アドレスをオーバーライドするかどうかを示します。**Host.display** 属性にのみ関連します。設定されている場合、仮想マシンのグラフィカルコンソールアドレスは、ホストで指定された表示アドレスによって上書きされます。設定されていない場合、仮想マシンのグラフィカルコンソールアドレスは上書きされません。

7.67.2. 証明書 (certificate)

TLS 接続の場合の TLS 証明書。TLS が有効になっていない場合、報告されません。

7.67.3. copy_paste_enabled

ユーザーがコンテンツを外部ホストからグラフィックコンソールにコピーアンドペーストできるかどうかを示します。このオプションは、SPICE コンソールタイプでのみ使用できます。

7.67.4. disconnect_action

グラフィックコンソールが切断されたときに実行されるアクションを返します。オプションは次のとおりです。

none

アクションは実行されません。

lock_screen

現在アクティブなユーザーセッションをロックします。

logout

現在アクティブなユーザーセッションをログアウトします。

reboot

仮想マシンの正常な再起動を開始します。

shutdown

仮想マシンの正常なシャットダウンを開始します。

このオプションは、SPICE コンソールタイプでのみ使用できます。

7.67.5. disconnect_action_delay

グラフィックコンソールの切断アクションが実行されるまでの遅延 (分単位)。このオプションは、シャットダウン切断アクションでのみ使用できます。

7.67.6. file_transfer_enabled

ユーザーが外部ホストからグラフィックコンソールにファイルをドラッグアンドドロップできるかどうかを示します。このオプションは、SPICE コンソールタイプでのみ使用できます。

7.67.7. keyboard_layout

このグラフィックコンソールで使用するキーボードレイアウト。このオプションは、VNC コンソールタイプでのみ使用できます。キーボードが有効になっていない場合は報告されません。

7.67.8. monitors

このグラフィックコンソール用に開いているモニターの数。このオプションは、SPICE コンソールタイプでのみ使用できます。使用できる値は、1、2または4です。

7.67.9. proxy

グラフィックコンソールクライアントがゲストに接続するために使用するプロキシ IP。クライアントがゲストのネットワークの外にある場合に便利です。このオプションは、SPICE コンソールタイプでのみ使用できます。このプロキシは、グローバル設定、クラスターレベル、仮想マシンプールレベルで設定するか、仮想マシンごとに無効にすることができます。プロキシが上記の場所のいずれかに設定されていて、仮想マシンに対して無効になっていない場合、このメソッドによって返されます。プロキシが設定されていない場合、何も報告されません。

7.67.10. secure_port

TLS を使用する場合に、グラフィックコンソールクライアントを接続するための、ゲスト上の保護されたポートアドレス。TLS が有効になっていない場合、報告されません。

7.67.11. single_qxl_pci

エンジンは、オペレーティングシステムに従って自動的に設定するようになりました。そのため、4.4.5 から非推奨になりました。各モニターに1つの PCI スロットを使用するか、複数のモニターのすべてに単一の PCI チャネルを使用するかを示します。このオプションは、SPICE コンソールタイプでのみ使用でき、ゲスト Linux ベースの OS を接続する場合にのみ使用できます。

7.67.12. smartcard_enabled

スマートカード認証を使用するかどうかを示します。このオプションは、SPICE コンソールタイプでのみ使用できます。

7.68. DISPLAYTYPE ENUM

仮想マシンのグラフィックコンソールへの接続に使用されるプロトコルの列挙を表します。

表7.90 値の概要

名前	概要
spice	タイプ SPICE の表示。
vnc	タイプ VNC の表示。

7.68.1. spice

タイプ SPICE の表示。詳細は、[SPICE のドキュメント](#) を参照してください。

7.68.2. vnc

タイプ VNC の表示。VNC は Virtual Network Computing の略で、RFB (Remote Frame Buffer) プロトコルを使用して別のマシンをリモートで制御するグラフィカルデスクトップ共有システムです。

7.69. DNS 構造体

DNS リゾルバー設定を表します。

表7.91 属性の概要

名前	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。

7.70. DNSRESOLVERCONFIGURATION 構造体

DNS リゾルバー設定を表します。

表7.92 属性の概要

名前	タイプ	概要
name_servers	String[]	ネームサーバーのアドレスの配列。

7.70.1. 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。

7.71. 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7.93 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
user	User	

表7.94 リンクの概要

名前	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。

7.71.1. 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。

7.72. DYNAMICCPU STRUCT

仮想マシンの動的 CPU の設定。

表7.95 属性の概要

名前	タイプ	概要
cpu_tune	CpuTune	
topology	CpuTopology	

7.73. ENTITYEXTERNALSTATUS ENUM

外部エンティティのステータスを表すタイプ。

表7.96 値の概要

名前	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。
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。

7.73.1. 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。

7.73.2. 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。

7.74. ENTITYPROFILEDETAIL STRUCT

表7.97 属性の概要

名前	タイプ	概要
profile_details	ProfileDetail[]	

7.75. ERRORHANDLING 構造体

表7.98 属性の概要

名前	タイプ	概要
on_error	MigrateOnError	

7.76. EVENT 構造体

イベントを表すタイプ。

表7.99 属性の概要

名前	タイプ	概要
code	Integer	イベントコード。
comment	文字列	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	文字列	イベント相関識別子。
custom_data	文字列	カスタムイベントデータを表すフリーテキスト。
custom_id	Integer	カスタムイベント識別子。
description	文字列	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	文字列	一意の ID
index	Integer	このイベントの数値インデックス。
log_on_host	Boolean	イベントを <code>hypervisor</code> にも書き込むかどうかを指定します。
name	文字列	人間が判読できるプレーンテキストでの名前。
origin	文字列	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	Date	イベント時間。

7.76.1. 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。

7.76.2. 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。

7.76.3. index

このイベントの数値インデックス。イベントのインデックスは常に増加しているため、インデックスが高いイベントはインデックスが低いイベントよりも古いことが保証されます。



重要

エンジンの現在の実装では、**id** 属性はこの **index** 属性と同じ値を持ちます。これは、API ユーザーが依存すべきではない実装の詳細です。今後は、**id** 属性は、数字以外の文字を含み、暗黙的な順序を持たない任意の文字列に変更される可能性があります。一方、この **index** 属性は、今後も整数のままで順序付けされることが保証されています。

7.76.4. log_on_host

イベントを `${hypervisor.name}` ログにも書き込むかどうかを指定します。ホストが指定されていない場合、イベントの説明はすべてのホストに書き込まれます。デフォルトは `false` です。

表7.100 リンクの概要

名前	タイプ	概要
cluster	クラスター	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。
ホスト	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。

7.76.5. 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。

7.76.6. 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。

7.76.7. ホスト

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。

7.76.8. 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。

7.76.9. 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。

7.76.10. 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。

7.76.11. 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。

7.77. EVENTSUBSCRIPTION 構造体

表7.101 属性の概要

名前	タイプ	概要
address	文字列	通知の送信先の電子メールアドレス。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
event	NotifiableEvent	subscribed-for イベント。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
notification_method	NotificationMethod	通知方法: SMTP または SNMP。
user	User	サブスクライブしているユーザー。

7.77.1. address

通知の送信先の電子メールアドレス。

指定しない場合、通知はユーザーの電子メールに送信されます。現時点では、ユーザーごとに1つのアドレスのみがサポートされます。既存のサブスクリプションとは異なる電子メールアドレスを持つサブスクリプションが追加された場合、提供されたアドレスがこのユーザーのイベントサブスクリプションの既存アドレスと競合するという説明と共に 409 (CONFLICT) ステータスが返されます。

今後このフィールドは非推奨になる止される可能性があり、通知は常にユーザーの電子メールアドレスに送信されます。

7.77.2. event

subscribed-for イベント。

(ユーザーと組み合わせ、イベントサブスクリプションを一意に識別します)。

7.77.3. notification_method

通知方法: SMTP または SNMP。

現在 API でサポートされているのは SMTP のみです。SNMP のサポートは、今後追加される予定です。

7.77.4. user

サブスクライブしているユーザー。

イベント名と組み合わせて、イベントサブスクリプションを一意に識別します。

7.78. EXTERNALCOMPUTERESOURCE STRUCT

表7.102 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
provider	文字列	
url	文字列	
user	文字列	

表7.103 リンクの概要

名前	タイプ	概要
external_host_provider	ExternalHostProvider	

7.79. EXTERNALDISCOVEREDHOST STRUCT

表7.104 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。

名前	タイプ	概要
id	文字列	一意の ID
ip	文字列	
last_report	文字列	
mac	文字列	
name	文字列	人間が判読できるプレーンテキストでの名前。
subnet_name	文字列	

表7.105 リンクの概要

名前	タイプ	概要
external_host_provider	ExternalHostProvider	

7.80. EXTERNALHOST 構造体

ホストプロバイダー (Foreman/Satellite など) によってプロビジョニングされたホストを表します。

詳細については、[Foreman のドキュメント](#) を参照してください。Red Hat Satellite の詳細については、[Satellite のドキュメント](#) を参照してください。

表7.106 属性の概要

名前	タイプ	概要
address	文字列	ホストのアドレスか、FQDN (完全修飾ドメイン名) の IP アドレス。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.107 リンクの概要

名前	タイプ	概要
external_host_provider	ExternalHostProvider	ホストの管理に使用する外部ホストプロバイダーへの参照。

7.81. EXTERNALHOSTGROUP STRUCT

表7.108 属性の概要

名前	タイプ	概要
--architecture <name>	文字列	
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
domain_name	文字列	
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
operating_system_name	文字列	
subnet_name	文字列	

表7.109 リンクの概要

名前	タイプ	概要
external_host_provider	ExternalHostProvider	

7.82. EXTERNALHOSTPROVIDER STRUCT

Foreman や Satellite などの外部ホストのプロバイダーを表します。

詳細については、[Foreman のドキュメント](#) を参照してください。Red Hat Satellite の詳細については、[Satellite のドキュメント](#) を参照してください。

表7.110 属性の概要

名前	タイプ	概要
authentication_url	文字列	外部プロバイダー認証 URL アドレスを定義します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
password	文字列	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	Boolean	プロバイダー認証が必要かどうかを定義します。
url	文字列	外部プロバイダーの URL アドレスを定義します。
username	文字列	認証プロセス中に使用されるユーザー名を定義します。

7.82.1. 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7.111 リンクの概要

名前	タイプ	概要
certificates	Certificate[]	このプロバイダーに対してエンジンがサポートする証明書への参照。
compute_resources	ExternalComputeResource[]	ホストプロバイダーで表されるコンピュータリソースへの参照。
discovered_hosts	ExternalDiscoveredHost[]	ホストプロバイダーで検出されたホストへの参照。
host_groups	ExternalHostGroup[]	ホストプロバイダーのホストグループへの参照。
hosts	Host[]	ホストプロバイダーによってプロビジョニングされるホストへの参照。

7.82.2. compute_resources

ホストプロバイダーで表されるコンピュータリソースへの参照。各ホストプロバイダーには、必要に応じてコンピュータリソースとして定義されたエンジンがあり、エンジン内に仮想マシンを作成できます。このコンピュータリソースの詳細は、ハイパーバイザーをデプロイするために、ベアメタルプロビジョニングのユースケースで使用されます。

7.82.3. discovered_hosts

ホストプロバイダーで検出されたホストへの参照。検出されたホストは、まだプロビジョニングされていないホストです。

7.82.4. host_groups

ホストプロバイダーのホストグループへの参照。ホストグループには、ホストプロバイダーがこのグループのメンバーであるすべてのホストに適用するさまざまなプロパティが含まれています。たとえばインストールされたソフトウェア、システム定義、パスワードなどです。

7.83. EXTERNALNETWORKPROVIDERCONFIGURATION 構造体

ホスト上で外部ネットワークプロバイダーをプロビジョニングする方法について説明します。

表7.112 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.113 リンクの概要

名前	タイプ	概要
external_network_provider	ExternalProvider	外部ネットワークプロバイダーへのリンク。
ホスト	ホスト	ホストへのリンク。

7.84. EXTERNALPROVIDER 構造体

外部プロバイダーを表します。

表7.114 属性の概要

名前	タイプ	概要
authentication_url	文字列	外部プロバイダー認証 URL アドレスを定義します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
password	文字列	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	Boolean	プロバイダー認証が必要かどうかを定義します。
url	文字列	外部プロバイダーの URL アドレスを定義します。
username	文字列	認証プロセス中に使用されるユーザー名を定義します。

7.84.1. 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。

7.85. EXTERNALSTATUSENUM

外部からの状態を表します。現在、このステータスは **ホスト** と **ストレージドメイン** で使用され、外部システムが認識しているオブジェクトのステータスを更新できるようにします。

表7.115 値の概要

名前	概要
error	エラーのステータス。
failure	失敗のステータス。
info	情報のステータス。
ok	OK のステータス。
warning	警告のステータス。

7.85.1. error

エラーのステータス。関連するオブジェクトに何らかのエラーがあります。

7.85.2. failure

失敗のステータス。関連するオブジェクトが失敗しています。

7.85.3. info

情報のステータス。関連するオブジェクトは OK ステータスですが、管理者に関連する可能性のある情報があります。

7.85.4. ok

OK のステータス。関連オブジェクトは正常に動作しています。

7.85.5. warning

警告のステータス。関連するオブジェクトは正常に機能していますが、管理者に関連する可能性のある警告がいくつかあります。

7.86. EXTERNALSYSTEMTYPE ENUM

step に関連する外部システムの種類を表す。

表7.116 値の概要

名前	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。

7.87. EXTERNALTEMPLATEIMPORT 構造体

外部システムからのテンプレートのインポート操作のパラメーターについて説明します。現在、OVA のみをサポートしています。

表7.117 属性の概要

名前	タイプ	概要
clone	Boolean	オプション:
url	文字列	エンジンに渡される URL。

7.87.1. clone

オプション:インポートされたテンプレート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、テンプレート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、同じテンプレートを複数回インポートすることはできません。同じテンプレート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。

7.87.2. url

エンジンに渡される URL。

例:

```
ova:///mnt/ova/ova_file.ova
```

表7.118 リンクの概要

名前	タイプ	概要
cluster	クラスター	結果のテンプレート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:
host	ホスト	OVA ファイルが存在するホストを指定します。
quota	クォータ	オプション:
storage_domain	StorageDomain	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。
template	Template	新しく作成されたテンプレートの名前を指定するために使用されるテンプレートエンティティ。

7.87.3. cpu_profile

オプション:結果のテンプレートの CPU プロファイルを指定します。

7.87.4. quota

オプション:結果のテンプレートに適用されるクォータを指定します。

7.87.5. template

新しく作成されたテンプレートの名前を指定するために使用されるテンプレートエンティティ。

名前が指定されていない場合は、ソーステンプレート名が使用されます。

7.88. 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7.119 属性の概要

名前	タイプ	概要
name	文字列	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	文字列	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	Boolean	オプション:
url	文字列	変換のために virt-v2v ツールに渡される URL。
username	文字列	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。

7.88.1. sparse

オプション: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。

指定されていない場合:- oVirt によって生成された OVA をインポートする場合、OVF 内のディスクの設定に従って決定されます。- それ以外の場合は、true に設定されます。

7.88.2. url

変換のために **virt-v2v** ツールに渡される URL。

例:

```
vpx://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7.120 リンクの概要

名前	タイプ	概要
cluster	クラスター	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	ファイル	オプション:
ホスト	ホスト	オプション:
quota	クォータ	オプション:

名前	タイプ	概要
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。

7.88.3. 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。

7.88.4. 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。

7.88.5. ホスト

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。

7.88.6. 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。

7.88.7. 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。

7.89. 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7.121 値の概要

名前	概要
kvm	
vmware	
xen	

7.90. FAULT 構造体

表7.122 属性の概要

名前	タイプ	概要
<code>detail</code>	文字列	
<code>reason</code>	文字列	

7.91. FENCETYPE ENUM

フェンス操作のタイプを表すタイプ。

表7.123 値の概要

名前	概要
<code>manual</code>	電源管理を介した手動のホストフェンシング。
<code>restart</code>	電源管理を介してホストを再起動します。
<code>start</code>	電源管理を介してホストを起動します。
<code>status</code>	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。
<code>stop</code>	電源管理を介してホストを停止します。

7.92. 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7.124 属性の概要

名前	タイプ	概要
<code>enabled</code>	Boolean	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。
<code>skip_if_connectivity_broken</code>	SkipIfConnectivityBroken	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。
<code>skip_if_gluster_bricks_up</code>	Boolean	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。
<code>skip_if_gluster_quorum_not_met</code>	Boolean	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。
<code>skip_if_sd_active</code>	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。

7.92.1. 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。

7.92.2. 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。

7.92.3. 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。

7.92.4. 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。

7.93. FILE 構造体

表7.125 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
type	文字列	

表7.126 リンクの概要

名前	タイプ	概要
storage_domain	StorageDomain	

7.94. FILTER 構造体

表7.127 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
position	Integer	

表7.128 リンクの概要

名前	タイプ	概要
scheduling_policy_unit	SchedulingPolicyUnit	

7.95. FIPSMODE ENUM

クラスターへの FIPS モードの表現。

表7.129 値の概要

名前	概要
disabled	FIPS モードが無効になっている。
enabled	FIPS モードが有効です。
undefined	FIPS モードはまだ評価されていません。

7.95.1. disabled

FIPS モードが無効になっている。

これは、FIPS モードが無効になっており、その中のホストで FIPS モードが無効になっている必要があることを意味します。

7.95.2. enabled

FIPS モードが有効です。

これは、FIPS モードが有効になっており、その中のホストで FIPS モードが有効になっている必要があることを意味します。

7.95.3. undefined

FIPS モードはまだ評価されていません。

現時点で、これは FIPS モードが未定であることを意味します。ホストが追加されると、この値はホストの設定に従って切り替わります。

7.96. FIREWALLTYPE ENUM

システムでサポートされているすべてのファイアウォールタイプについて説明します。

表7.130 値の概要

名前	概要
firewalld	FirewallD ファイアウォールタイプ。
iptables	IPTables ファイアウォールタイプ。

7.96.1. firewalld

FirewallD ファイアウォールタイプ。

クラスタのファイアウォールタイプが **firewalld** に設定されている場合、クラスタ内のすべてのホストのファイアウォールは、**firewalld** を使用して設定されます。[FirewallD](#) は、バージョン 4.2 で IPTables を置き換えました。コマンドラインプログラムと動的設定を使用して設定を簡素化します。

7.96.2. iptables

IPTables ファイアウォールタイプ。

iptables は非推奨になりました。

7.97. FLOPPY 構造体

フロッピーファイルの基盤となる表現。

表7.131 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
file	ファイル	フロッピーデバイスのコンテンツとそのタイプを表すファイルオブジェクト。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.132 リンクの概要

名前	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。

7.97.1. 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。

7.98. FOPSTATISTIC STRUCT

表7.133 属性の概要

名前	タイプ	概要
name	文字列	
statistics	Statistic[]	

7.99. GLUSTERBRICK 構造体

表7.134 属性の概要

名前	タイプ	概要
brick_dir	文字列	
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
device	文字列	
fs_name	文字列	
gluster_clients	GlusterClient[]	
id	文字列	一意の ID

名前	タイプ	概要
memory_pools	GlusterMemoryPool[]	
mnt_options	文字列	
name	文字列	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	文字列	
status	GlusterBrickStatus	

表7.135 リンクの概要

名前	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。

7.99.1. 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。

7.100. GLUSTERBRICKADVANCEDDETAILS STRUCT

表7.136 属性の概要

名前	タイプ	概要
comment	文字列	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。

名前	タイプ	概要
description	文字列	プレーンテキストでの人間が判読できる説明。
device	文字列	
fs_name	文字列	
gluster_clients	GlusterClient[]	
id	文字列	一意の ID
memory_pools	GlusterMemoryPool[]	
mnt_options	文字列	
name	文字列	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7.137 リンクの概要

名前	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。

7.100.1. 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。

7.101. GLUSTERBRICKMEMORYINFO STRUCT

表7.138 属性の概要

名前	タイプ	概要
memory_pools	GlusterMemoryPool[]	

7.102. GLUSTERBRICKSTATUS ENUM

表7.139 値の概要

名前	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。

7.103. GLUSTERCLIENT 構造体

表7.140 属性の概要

名前	タイプ	概要
bytes_read	Integer	
bytes_written	Integer	
client_port	Integer	
host_name	文字列	

7.104. GLUSTERHOOK STRUCT

表7.141 属性の概要

名前	タイプ	概要
checksum	文字列	
comment	文字列	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	文字列	

名前	タイプ	概要
content	文字列	
content_type	HookContentType	
description	文字列	プレーンテキストでの人間が判読できる説明。
gluster_command	文字列	
id	文字列	一意の ID
name	文字列	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7.142 リンクの概要

名前	タイプ	概要
cluster	クラスター	
server_hooks	GlusterServerHook[]	

7.105. GLUSTERHOOKSTATUS ENUM

表7.143 値の概要

名前	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。

7.106. GLUSTERMEMORYPOOL STRUCT

表7.144 属性の概要

名前	タイプ	概要
alloc_count	Integer	
cold_count	Integer	
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
hot_count	Integer	
id	文字列	一意の ID
max_alloc	Integer	
max_stdalloc	Integer	
name	文字列	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	文字列	

7.107. GLUSTERSERVERHOOK STRUCT

表7.145 属性の概要

名前	タイプ	概要
checksum	文字列	
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
status	GlusterHookStatus	

表7.146 リンクの概要

名前	タイプ	概要
ホスト	ホスト	

7.108. GLUSTERSTATE ENUM

表7.147 値の概要

名前	概要
down	
unknown	
up	

7.109. GLUSTERVOLUME 構造体

表7.148 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
disperse_count	Integer	
id	文字列	一意の ID
name	文字列	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	
status	GlusterVolumeStatus	
stripe_count	Integer	
transport_types	TransportType[]	

名前	タイプ	概要
volume_type	GlusterVolumeType	

表7.149 リンクの概要

名前	タイプ	概要
bricks	GlusterBrick[]	
cluster	クラスター	
statistics	Statistic[]	

7.110. GLUSTERVOLUMEPROFILEDETAILS STRUCT

表7.150 属性の概要

名前	タイプ	概要
brick_profile_details	BrickProfileDetail[]	
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
nfs_profile_details	NfsProfileDetail[]	

7.111. GLUSTERVOLUMESTATUS ENUM

表7.151 値の概要

名前	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。

名前	概要
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。

7.112. GLUSTERVOLUMETYPE ENUM

Gluster ボリュームのタイプを表すタイプ。

表7.152 値の概要

名前	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。

7.112.1. 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。

7.112.2. 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。

7.112.3. 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。

7.112.4. 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。

7.112.5. 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。

7.112.6. 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。

7.112.7. 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。

7.112.8. 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。

7.112.9. 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ドニングの導入により、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。

7.113. GRACEPERIOD 構造体

表7.153 属性の概要

名前	タイプ	概要
expiry	Integer	

7.114. GRAPHICSCONSOLE 構造体

表7.154 属性の概要

名前	タイプ	概要
address	文字列	
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	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7.155 リンクの概要

名前	タイプ	概要
instance_type	InstanceType	
template	Template	

名前	タイプ	概要
vm	Vm	

7.115. GRAPHICSTYPE ENUM

グラフィックコンソールへの接続に使用されるグラフィックプロトコル。

表7.156 値の概要

名前	概要
spice	SPICE タイプのグラフィックスプロトコル。
vnc	VNC タイプのグラフィックスプロトコル。

7.115.1. spice

SPICE タイプのグラフィックスプロトコル。詳細は、[SPICE のドキュメント](#) を参照してください。

7.115.2. vnc

VNC タイプのグラフィックスプロトコル。VNC は Virtual Network Computing の略で、RFB (Remote Frame Buffer) プロトコルを使用して別のマシンをリモートで制御するグラフィカルデスクトップ共有システムです。

7.116. 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7.157 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
domain_entry_id	文字列	含まれるディレクトリーサービスドメイン ID。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
namespace	文字列	グループが存在する名前空間。

表7.158 リンクの概要

名前	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。
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。

7.116.1. 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。

7.117. GUESTOPERATINGSYSTEMSTRUCT

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を表します。

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/vms/123
```

結果は以下のようになります。

```
<vm href="/ovirt-engine/api/vms/123" id="123">
...
<guest_operating_system>
  <architecture>x86_64</architecture>
  <codename>Maipo</codename>
  <distribution>Red Hat Enterprise Linux Server</distribution>
  <family>Linux</family>
  <kernel>
    <version>
      <build>0</build>
      <full_version>3.10.0-514.10.2.el7.x86_64</full_version>
      <major>3</major>
      <minor>10</minor>
      <revision>514</revision>
    </version>
  </kernel>
  <version>
    <full_version>7.3</full_version>
    <major>7</major>
    <minor>3</minor>
  </version>
</guest_operating_system>
</vm>
```

```

</version>
</guest_operating_system>
</vm>

```

表7.159 属性の概要

名前	タイプ	概要
architecture	文字列	x86_64 などのオペレーティングシステムのアーキテクチャー。
codename	文字列	Maipo などのオペレーティングシステムのコードネーム。
distribution	文字列	オペレーティングシステムディストリビューションの完全な名前。
family	文字列	Linux などのオペレーティングシステムのファミリー。
kernel	カーネル	オペレーティングシステムのカーネルバージョン。
version	バージョン	インストールされているオペレーティングシステムのバージョン。

7.118. HARDWAREINFORMATION 構造体

ホストのハードウェア情報を表します。

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/hosts/123
```

結果は以下のようになります。

```

<host href="/ovirt-engine/api/hosts/123" id="123">
...
<hardware_information>
  <family>Red Hat Enterprise Linux</family>
  <manufacturer>Red Hat</manufacturer>
  <product_name>RHEV Hypervisor</product_name>
  <serial_number>01234567-89AB-CDEF-0123-456789ABCDEF</serial_number>
  <supported_rng_sources>
    <supported_rng_source>random</supported_rng_source>
  </supported_rng_sources>
  <uuid>12345678-9ABC-DEF0-1234-56789ABCDEF0</uuid>
  <version>1.2-34.5.el7ev</version>
</hardware_information>
...
</application>

```

表7.160 属性の概要

名前	タイプ	概要
family	文字列	ホストの CPU のタイプ。
manufacturer	文字列	ホストのマシンの製造元とハードウェアベンダー。
product.name	文字列	ホストの製品名 (例: RHEV Hypervisor)。
serial_number	文字列	ホストのシャーシの一意の ID。
supported_rng_sources	RngSource[]	サポートされている乱数ジェネレーターのソース。
uuid	文字列	各ホストの一意の ID。
version	文字列	各製造元の固有の名前。

7.119. HIGHAVAILABILITY 構造体

仮想マシンの高可用性を表すタイプ。

表7.161 属性の概要

名前	タイプ	概要
enabled	Boolean	仮想マシンが高可用性と見なされているかどうかを定義します。
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。

7.119.1. enabled

仮想マシンが高可用性と見なされているかどうかを定義します。スプリットブレインのシナリオを防ぐには、仮想マシンのリースを設定することを強く推奨します (このセクションを参照)。ブートディスクのストレージドメインまたはその他のアクティブなストレージドメインを使用します。

7.119.2. 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	Medium
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
Medium	50
高	100

7.120. HOOK 構造体

フックを表します。

表7.162 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
event_name	文字列	フックを実行するイベントの名前。
id	文字列	一意の ID
md5	文字列	フックのチェックサム。
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.163 リンクの概要

名前	タイプ	概要
ホスト	ホスト	フックが属するホストへの参照。

7.121. 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7.164 値の概要

名前	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。

7.122. HOOKSTAGE ENUM

タイプは、フックが実行されるボリュームイベントのステージを表します。

表7.165 値の概要

名前	概要
post	ボリューム開始後のステージ。
pre	ボリューム開始前のステージ。

7.123. HOOKSTATUS ENUM

タイプはフックのステータスを表します。

表7.166 値の概要

名前	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。

7.124. HOST 構造体

ホストを表すタイプ。

表7.167 属性の概要

名前	タイプ	概要
address	文字列	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。

名前	タイプ	概要
証明書 (certificate)	証明書	ホスト証明書。
comment	文字列	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	文字列	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。
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	このホストの自己ホスト型エンジンステータス。
id	文字列	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	文字列	人間が判読できるプレーンテキストでの名前。
network_operation_in_progress	Boolean	'setup networks'、'sync networks'、または 'refresh capabilities'などのネットワーク関連の操作が現在このホストで実行されているかどうかを指定します。
numa_supported	Boolean	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。

名前	タイプ	概要
os	OperatingSystem	このホストのオペレーティングシステム。
override_iptables	Boolean	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。
ovn_configured	Boolean	ホストが OVN を正しく設定したかどうかを示します。
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。
reinstallation_required	Boolean	ホストを再インストールする必要があるかどうかを指定します。
root_password	文字列	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	文字列	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	Boolean	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。

名前	タイプ	概要
vgpu_placement	VgpuPlacement	vGPU 配置ストラテジーを指定します。

7.124.1. 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。

7.124.2. hosted_engine

このホストの自己ホスト型エンジンステータス。



重要

ホストまたはホストのコレクションが取得される時、操作の **all_content** パラメーターが明示的に **true** に設定されていない限り、この属性は含まれません。詳細は、ホスト [1つ](#) または [複数](#) を取得する操作のドキュメントを参照してください。

7.124.3. 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。

7.124.4. 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>
```

7.124.5. 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。

7.124.6. network_operation_in_progress

'setup networks'、'sync networks'、または 'refresh capabilities'などのネットワーク関連の操作が現在このホストで実行されているかどうかを指定します。



注記

この属性を応答に含めるには、ヘッダー **All-Content:true** を要求に追加する必要があります。

7.124.7. 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。

7.124.8. 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。

7.124.9. 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。

7.124.10. 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。

7.124.11. status_detail

ホストステータスの詳細。Gluster ホストに関連します。

7.124.12. 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>
```

7.124.13. version

VDSM のバージョン。

以下はその例です。

```
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>vdsmd-4.18.999-419.gitcf06367.el7</full_version>
  <major>4</major>
  <minor>18</minor>
  <revision>0</revision>
</version>
...
</host>
```

表7.168 リンクの概要

名前	タイプ	概要
affinity_labels	AffinityLabel[]	
agents	Agent[]	
cluster	クラスター	
cpu_units	HostCpuUnit[]	トポロジー(ソケット、コア)に関する詳細情報と、現在の CPU ピン留めに関する情報を含むすべてのホストの CPU のリスト。
devices	HostDevice[]	
external_host_provider	ExternalHostProvider	
external_network_provider_configurations	ExternalNetworkProviderConfiguration[]	ホスト上でプロビジョニングされた外部ネットワークプロバイダー。
hooks	Hook[]	

名前	タイプ	概要
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。
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[]	

7.124.14. cpu_units

トポロジー (ソケット、コア) に関する詳細情報と、現在の CPU ピン留めに関する情報を含むすべてのホストの CPU のリスト。

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

以下のような XML で応答を受け取ります。

```
<host_cpu_units>
  <host_cpu_unit>
    <core_id>0</core_id>
    <cpu_id>0</cpu_id>
    <socket_id>0</socket_id>
    <vms>
      <vm href="/ovirt-engine/api/vms/def" id="def" />
    </vms>
  </host_cpu_unit>
  <host_cpu_unit>
    <core_id>0</core_id>
    <cpu_id>1</cpu_id>
    <socket_id>1</socket_id>
    <runs_vdsm>true</runs_vdsm>
```

```

</host_cpu_unit>
<host_cpu_unit>
  <core_id>0</core_id>
  <cpu_id>2</cpu_id>
  <socket_id>2</socket_id>
</host_cpu_unit>
</host_cpu_units>

```

7.124.15. external_network_provider_configurations

ホスト上でプロビジョニングされた外部ネットワークプロバイダー。

この属性は読み取り専用です。これを設定しても、ホストには影響しません。このパラメーターの値は、クラスタの [Default Network Provider](#) を反映しています。

7.124.16. 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>

```

7.124.17. 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>

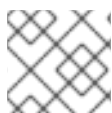
```



```

<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。

名前	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)。
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。

名前	説明
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。

7.125. HOSTCPUUNIT STRUCT

現在のピン留めステータスを示すホストの物理 CPU を表すタイプ。

表7.169 属性の概要

名前	タイプ	概要
comment	文字列	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。
core_id	Integer	CPU が属するコアの ID。
cpu_id	Integer	CPU の ID。
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
runs_vdsm	Boolean	CPU が VDSM を実行していることを示すフラグ
socket_id	Integer	CPU が属するソケットの ID。

表7.170 リンクの概要

名前	タイプ	概要
vms	Vm[]	仮想 CPU がこの物理 CPU に固定されている VM のリスト。

7.126. HOSTDEVICE 構造体

表7.171 属性の概要

名前	タイプ	概要
capability	文字列	
comment	文字列	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。

名前	タイプ	概要
description	文字列	プレーンテキストでの人間が判読できる説明。
driver	文字列	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	文字列	一意の ID
iommu_group	Integer	
m_dev_types	MDevType[]	物理デバイス上でサポートされているすべての mdev タイプのリスト。
name	文字列	人間が判読できるプレーンテキストでの名前。
physical_function	HostDevice	
placeholder	Boolean	
product	Product	
vendor	Vendor	
virtual_functions	Integer	

7.126.1. 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7.172 リンクの概要

名前	タイプ	概要
ホスト	ホスト	
parent_device	HostDevice	
vm	Vm	

7.127. HOSTDEVICEPASSTHROUGH 構造体

表7.173 属性の概要

名前	タイプ	概要
enabled	Boolean	

7.128. 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7.174 属性の概要

名前	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。
base_interface	文字列	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	Boolean	ブリッジネットワークのステータスを定義します。
check_connectivity	Boolean	
comment	文字列	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	Boolean	
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	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。

名前	タイプ	概要
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	文字列	人間が判読できるプレーンテキストでの名前。
override_configuration	Boolean	
properties	Property[]	
speed	Integer	
status	NicStatus	
virtual_functions_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	

7.128.1. 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、[ウィキペディア](#) や [プレゼンテーション](#) を参照)。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 にのみ有効です。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。このパラメーター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。

7.128.2. 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。

表7.175 リンクの概要

名前	タイプ	概要
host	Host	
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。

名前	タイプ	概要
network_labels	NetworkLabel[]	この NIC に適用されるラベル。
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。
statistics	Statistic[]	NIC の統計へのリンク。

7.128.3. 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。

7.128.4. 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.current.rx.bps` - 受信したデータ速度 (ビット毎秒、バージョン 4.2 以降)。
- `data.current.tx.bps` - 送信されたデータ速度 (ビット毎秒、バージョン 4.2 以降)。
- `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。

7.129. 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7.176 属性の概要

名前	タイプ	概要
all_networks_allowed	Boolean	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。

名前	タイプ	概要
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。

7.129.1. 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。

7.129.2. 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。

7.130. 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7.177 値の概要

名前	概要
stomp	STOMP 上の JSON-RPC プロトコル。
xml	XML-RPC プロトコル。

7.131. HOSTSTATUS ENUM

ホストステータスを表すタイプ。

表7.178 値の概要

名前	概要
----	----

名前	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。
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。

7.131.1. 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。

7.131.2. 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。

7.131.3. 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。

7.131.4. 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。

7.131.5. 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。

7.131.6. 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。

7.131.7. 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。

7.131.8. 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。Vintage Node はサポートされなくなり、非推奨となったため、このプロパティーは関連しなくなりました。

7.131.9. preparing_for_maintenance

ホストはメンテナ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。

7.132. HOSTSTORAGE 構造体

表7.179 属性の概要

名前	タイプ	概要
address	文字列	
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
driver_options	Property[]	cinder ドライバーを使用してストレージドメインを作成するときに渡されるオプション。

名前	タイプ	概要
driver_sensitive_options	Property[]	cinder ドライバーを使用してストレージドメインを作成するときに渡される機密情報を含むパラメーター。
id	文字列	一意の ID
logical_units	LogicalUnit[]	
mount_options	文字列	
name	文字列	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	Boolean	
password	文字列	
path	文字列	
port	Integer	
portal	文字列	
target	文字列	
type	StorageType	
username	文字列	
vfs_type	文字列	
volume_group	VolumeGroup	

7.132.1. driver_options

cinder ドライバーを使用してストレージドメインを作成するときに渡されるオプション。

以下に例を示します (Kaminario バックエンド):

```
POST /ovirt-engine/api/storagedomains/
```

```

<storage_domain>
  <name>kamniraio-cinder</name>
  <type>managed_block_storage</type>
  <storage>
    <type>managed_block_storage</type>
    <driver_options>
      <property>
        <name>san_ip</name>
        <value>192.168.1.1</value>
      </property>
      <property>
        <name>san_login</name>
        <value>username</value>
      </property>
      <property>
        <name>san_password</name>
        <value>password</value>
      </property>
      <property>
        <name>use_multipath_for_image_xfer</name>
        <value>>true</value>
      </property>
      <property>
        <name>volume_driver</name>
        <value>cinder.volume.drivers.kaminario.kaminario_iscsi.KaminarioISCSIDriver</value>
      </property>
    </driver_options>
  </storage>
  <host>
    <name>host</name>
  </host>
</storage_domain>

```

7.132.2. driver_sensitive_options

cinder ドライバーを使用してストレージドメインを作成するときに渡される機密情報を含むパラメーター。これらのパラメーターは、保存時に暗号化されます。

たとえば、次の XML は、ユーザー名、パスワード、および SAN IP アドレスを暗号化して保存します。

```
POST /ovirt-engine/api/storagedomains/
```

```

<storage_domain>
  <name>kamniraio-cinder</name>
  <type>managed_block_storage</type>
  <storage>
    <type>managed_block_storage</type>
    <driver_options>
      <property>
        <name>san_ip</name>
        <value>192.168.1.1</value>
      </property>
      <property>
        <name>san_login</name>

```

```

    <value>username</value>
  </property>
</property>
  <name>san_password</name>
  <value>password</value>
</property>
</property>
  <name>use_multipath_for_image_xfer</name>
  <value>>true</value>
</property>
</property>
  <name>volume_driver</name>
  <value>cinder.volume.drivers.kaminario.kaminario_iscsi.KaminarioISCSIDriver</value>
</property>
</driver_options>
<driver_sensitive_options>
  <property>
    <name>username</name>
    <value>admin</value>
  </property>
  <property>
    <name>password</name>
    <value>123</value>
  </property>
  <property>
    <name>san_ip</name>
    <value>192.168.1.1</value>
  </property>
</driver_sensitive_options>
</storage>
<host>
  <name>host</name>
</host>
</storage_domain>

```

7.132.3. 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。

7.132.4. 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7.180 リンクの概要

名前	タイプ	概要
ホスト	ホスト	

7.133. 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7.181 値の概要

名前	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。

7.133.1. 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。

7.133.2. 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。

Vintage Node はサポートされなくなり、非推奨となったため、このプロパティは関連しなくなりました。

7.134. HOSTEDENGINE 構造体

表7.182 属性の概要

名前	タイプ	概要
active	Boolean	
configured	Boolean	
global_maintenance	Boolean	
local_maintenance	Boolean	

名前	タイプ	概要
score	Integer	

7.135. ICON 構造体

仮想マシンまたはテンプレートのアイコン。

表7.183 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	アイコンファイルの Base64 エンコードコンテンツ。
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
media_type	文字列	アイコンファイルのフォーマット。
name	文字列	人間が判読できるプレーンテキストでの名前。

7.135.1. 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**

7.136. 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7.184 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。

名前	タイプ	概要
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

7.137. IMAGE 構造体

イメージエンティティを表します。

表7.185 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
size	Integer	イメージファイルのサイズ。
type	ImageFileType	イメージファイルのタイプ。

表7.186 リンクの概要

名前	タイプ	概要
storage_domain	StorageDomain	このイメージに関連付けられたストレージドメイン。

7.138. IMAGEFILETYPE ENUM

イメージのファイルタイプを表します。

表7.187 値の概要

名前	概要
disk	イメージは、仮想マシンのディスクとして使用できるディスクフォーマットです。
floppy	イメージは、たとえば Windows に VirtIO ドライバーをインストールするために、仮想マシンに接続できるフロッピーディスクです。
iso	イメージは ` です。

7.138.1. iso

イメージは、仮想マシンを起動およびインストールするための CD-ROM として使用できる **.iso** ファイルです。

7.139. 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7.188 属性の概要

名前	タイプ	概要
active	Boolean	この転送に少なくとも1つのアクティブなセッションがあるかどうかを示します。つまり、クライアントとデーモンの間に少なくとも1つのライブ転送セッションがあるかどうかを示します。
comment	文字列	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	文字列	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。
format	DiskFormat	アップロード中に送信された、またはダウンロード中に受信されたデータの形式。
id	文字列	一意の ID
inactivity_timeout	Integer	クライアントが非アクティブの場合のタイムアウト (秒単位) で、これを過ぎると Red Hat Virtualization Manager によって転送が中止されます。
name	文字列	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	文字列	ユーザーが入力または出力するプロキシサーバーの URL。
shallow	Boolean	イメージチェーン全体ではなく、指定されたイメージのみをダウンロードします。
timeout_policy	ImageTransferTimeoutPolicy	タイムアウトポリシーは、クライアントが inactivityTimeout を超えてアイドル状態になっている場合に、システムが転送を処理する方法を決定します。
transfer_url	文字列	ユーザーが直接入力または出力できるデーモンサーバーの URL。

名前	タイプ	概要
transferred	Integer	転送されたバイト数を示します。

7.139.1. 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。

7.139.2. format

アップロード中に送信された、またはダウンロード中に受信されたデータの形式。指定しない場合、デフォルトでディスクの形式になります。

7.139.3. inactivity_timeout

クライアントが非アクティブの場合のタイムアウト (秒単位) で、これを過ぎると Red Hat Virtualization Manager によって転送が中止されます。非アクティブタイムアウトを無効にするには、'0' を指定します。指定されていない場合は、値はデフォルトで **engine-config** 値 (value: TransferImageClientInactivityTimeoutInSeconds) になります。

7.139.4. 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。

7.139.5. 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、イメージ転送が [転送](#) フェーズにある場合のみ使用できます。詳細は、**phase** を参照してください。

7.139.6. shallow

イメージチェーン全体ではなく、指定されたイメージのみをダウンロードします。

true の場合、format="raw" および direction="download" を使用すると、転送には指定されたディスクスナップショットからのデータのみが含まれ、未割り当て領域はホールとして報告されます。デフォルトでは、転送にはすべてのディスクスナップショットからのデータが含まれます。

ディスクスナップショットを指定すると、指定したディスクスナップショットのデータのみが転送に含まれます。ディスクを指定すると、アクティブなディスクスナップショットからのデータのみが転送に含まれます。

このパラメーターは、format="raw" を使用していない場合、または direction="upload" の場合は効果がありません。

例: 単一のスナップショットをダウンロードする場合:

```
<image_transfer>
```

```
<snapshot id="2fb24fa2-a5db-446b-b733-4654661cd56d"/>
<direction>download</direction>
<format>raw</format>
<shallow>true</shallow>
</image_transfer>
```

アクティブなスナップショットディスクイメージ (ディスクスナップショットとしてアクセスできない) をダウンロードするには、ディスクを指定します。

```
<image_transfer>
  <disk id="ff6be46d-ef5d-41d6-835c-4a68e8956b00"/>
  <direction>download</direction>
  <format>raw</format>
  <shallow>true</shallow>
</image_transfer>
```

どちらの場合も、imageio クライアントを使用して qcow2 イメージをダウンロードできるようになりました。

```
from ovirt_imageio import client

client.download(
    transfer.transfer_url,
    "51275e7d-42e9-491f-9d65-b9211c897eac",
    backing_file="07c0ccac-0845-4665-9097-d0a3b16cf43b",
    backing_format="qcow2")
```

7.139.7. transfer_url

ユーザーが直接入力または出力できるデーモンサーバーの URL。

これは、**proxy_url** の代わりになります。つまり、クライアントがホストマシンにアクセスできる場合、プロキシをバイパスしてホストに直接転送することができ、スループットパフォーマンスが向上する可能性があります。この属性は、イメージ転送が **転送** フェーズにある場合にのみ使用できます。詳細は、**phase** を参照してください。

表7.189 リンクの概要

名前	タイプ	概要
backup	バックアップ	イメージ転送に関連付けられたバックアップ。
disk	ディスク	入力または出力の対象となるディスク。
ホスト	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	イメージ	入力または出力の対象となるイメージ。
snapshot	DiskSnapshot	入力または出力の対象となるディスクスナップショット。

7.139.8. backup

イメージ転送に関連付けられたバックアップ。バックアップの一部であるディスクのイメージ転送を開始するときに指定します。

7.139.9. 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。指定されていない場合は、アクティブなホストがデータセンターからランダムに選択されます。

7.139.10. image

入力または出力の対象となるイメージ。



重要

この属性は、エンジンのバージョン 4.2 以降で非推奨になりました。代わりに、**disk** または **snapshot** の属性を使用してください。

7.140. 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7.190 値の概要

名前	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。

7.141. 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7.191 値の概要

名前	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。

名前	概要
cancelled_system	このフェーズは、システムが転送をキャンセルした結果として設定されます。
cancelled_user	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_cleanup	このフェーズは、ユーザーが転送をキャンセルし、必要なクリーンアップが行われていることを示します。
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_cleanup	このフェーズは、ユーザーが転送をキャンセルし、必要なクリーンアップが行われたことを示します。
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 (例: 選択したホストで <code>ovirt-imageio</code> が実行されていない) が発生したことを意味します。
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。
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。

7.141.1. 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。

7.141.2. 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。

7.141.3. 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。システムが転送をキャンセルした場合も同様です。

7.141.4. 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。

7.141.5. 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。

7.141.6. 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 (例: 選択したホストで `ovirt-imageio` が実行されていない) が発生したことを意味します。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。

7.141.7. 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。

7.141.8. 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。

7.142. IMAGETRANSFERTIMEOUTPOLICY ENUM

[イメージ転送](#) のタイムアウトポリシー。

クライアントが `inactivityTimeout` (秒) の間非アクティブな場合に、システムが転送を処理する方法を定義します。

詳細は、[イメージ転送](#) を参照してください。

表7.192 値の概要

名前	概要
<code>cancel</code>	転送をキャンセルし、ディスクのロックを解除します。
<code>legacy</code>	LEGACY ポリシーは、デフォルトであるレガシー機能を保持します。
<code>pause</code>	転送を一時停止します。

7.142.1. cancel

転送をキャンセルし、ディスクのロックを解除します。アップロード方向を使用したイメージ転送の場合、ディスクは削除されます。

7.142.2. legacy

LEGACY ポリシーは、デフォルトであるレガシー機能を保持します。デフォルトの動作では、方向がダウンロードの場合は転送がキャンセルされ、アップロードの場合は一時停止されます。

7.142.3. pause

転送を一時停止します。ユーザーは、転送を再開またはキャンセルできます。転送が一時停止されている間、ディスクはロックされたままになります。

7.143. 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 → クラスター → engine-config です。

表7.193 値の概要

名前	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。

7.144. INITIALIZATION 構造体

表7.194 属性の概要

名前	タイプ	概要
active_directory_ou	文字列	
authorized_ssh_keys	文字列	
cloud_init	CloudInit	cloud-init 設定を指定する非推奨の属性。
cloud_init_network_protocol	CloudInitNetworkProtocol	cloud-init ネットワークパラメーターのフォーマットに使用する cloud-init プロトコルを指定する属性。
configuration	設定	

名前	タイプ	概要
Custom Script	文字列	
dns_search	文字列	
dns_servers	文字列	
domain	文字列	
host_name	文字列	
input_locale	文字列	
nic_configurations	NicConfiguration[]	
org_name	文字列	
regenerate_ids	Boolean	
regenerate_ssh_keys	Boolean	
root_password	文字列	
system_locale	文字列	
timezone	文字列	
ui_language	文字列	
user_locale	文字列	
user_name	文字列	
windows_license_key	文字列	

7.144.1. cloud_init

cloud-init 設定を指定する非推奨の属性。

この属性と [CloudInit](#) タイプは非推奨となっており、今後削除される予定です。cloud-init 設定を指定するには、[Initialization](#) タイプ内の属性を使用します。これら2つのタイプの属性間のマッピングは次のとおりです。

CloudInit	初期化
<code>authorized_keys</code>	<code>authorized_ssh_keys</code>
<code>dns.search_domains</code>	<code>dns_search</code>
<code>dns.servers</code>	<code>dns_servers</code>
<code>files</code>	Custom Script
ホスト	<code>host_name</code>
<code>network_configuration.nics</code>	<code>nic_configurations</code>
<code>regenerate_ssh_keys</code>	<code>regenerate_ssh_keys</code>
<code>timezone</code>	<code>timezone</code>
<code>users</code>	<code>user_name & root_password</code>

cloud-init の使い方の詳細については、[Python](#) と [Java](#) の例を参照してください。

7.144.2. cloud_init_network_protocol

cloud-init ネットワークパラメーターのフォーマットに使用する cloud-init プロトコルを指定する属性。省略した場合、[CloudInitNetworkProtocol](#) で説明されているように、デフォルト値が使用されます。

7.145. 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す)。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7.195 属性の概要

名前	タイプ	概要
<code>auto_pinning_policy</code>	AutoPinningPolicy	自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。
<code>bios</code>	Bios	仮想マシンの BIOS 設定への参照。
<code>comment</code>	文字列	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>console</code>	Console	この仮想マシン用に設定されたコンソール。
<code>cpu</code>	Cpu	仮想マシン CPU の設定。
<code>cpu_pinning_policy</code>	CpuPinningPolicy	CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。
CPU Shares	Integer	
<code>creation_time</code>	Date	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。
<code>custom_cpu_model</code>	文字列	

名前	タイプ	概要
Custom Emulated Machine	文字列	
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	Boolean	true の場合、仮想マシンは削除できません。
description	文字列	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	文字列	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	Icon	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)。
multi_queues_enabled	Boolean	true の場合、各仮想インターフェイスは、利用可能な仮想 CPU に応じて最適な数のキューを取得します。
name	文字列	人間が判読できるプレーンテキストでの名前。

名前	タイプ	概要
origin	文字列	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。
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。
serial_number	SerialNumber	クラスター内の仮想マシンのシリアル番号。
small_icon	Icon	仮想マシンの小さなアイコン。
Soundcard Enabled	Boolean	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	Boolean	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	Boolean	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。
storage_error_resume_behaviour	VmStorageErrorResumeBehaviour	ストレージエラー後に仮想マシンを再開する方法を決定します。
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。
tpm_enabled	Boolean	true の場合、TPM デバイスが仮想マシンに追加されます。
tunnel_migration	Boolean	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であるかサブバージョンであることを示します。

名前	タイプ	概要
virtio-scsi	VirtioScsi	VirtIO SCSI 設定への参照。
virtio_scsi_multi_queues	Integer	このフィールドの Virtio-SCSI controller のキュー数には virtioScsiMultiQueuesEnabled が true である必要があります。詳細は virtioScsiMultiQueuesEnabled を参照してください。
virtio_scsi_multi_queues_enabled	Boolean	true の場合、Virtio-SCSI デバイスは、使用可能な仮想 CPU とディスク、または指定された virtioScsiMultiQueues に応じて、いくつかの複数キューを取得します。
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。

7.145.1. auto_pinning_policy

自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。



重要

エンジンのバージョン 4.5 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後、削除される可能性があります。代わりに CpuPinningPolicy を使用してください。

7.145.2. 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>
```

7.145.3. cpu_pinning_policy

CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。指定されていない場合、CPU ピニング文字列の以前の動作により、CpuPinningPolicy が None または Manual に決定されます。

7.145.4. 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。

7.145.5. 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。

7.145.6. initialization

仮想マシンの初期化設定への参照。



注記

Red Hat Virtualization 4.1.8 以降、このプロパティは空のタグを送信することでクリアできます。

たとえば、**initialization** 属性をクリアするには、次のようなリクエストを送ります。

```
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>
  <initialization/>
</vm>
```

このようなリクエストへのレスポンス、およびヘッダー **All-Content: true** を持つリクエストには、引き続きこの属性が含まれます。

7.145.7. 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。

7.145.8. 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。

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。

7.145.9. 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>
```

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。上記の例を使用して、仮想マシンが `up` 状態のときにメモリーを増やすことができます。サイズの増分は、**HotPlugMemoryBlockSizeMb** 設定値 (デフォルトでは 256 MiB) の値で割り切れる必要があります。メモリーサイズの増分がこの値で割り切れない場合、メモリーサイズの変更は次の実行設定にのみ保存されます。メモリーのホットプラグ操作が成功するたびに、1つまたは2つの新しいメモリーデバイスが作成されます。

メモリーのホットアンプラグは、Red Hat Virtualization 4.2 以降でサポートされています。メモリーのホットアンプラグは、仮想マシンの状態が `up` の場合にのみ実行できます。ホットアンプラグ操作で取り外すことができるのは、以前にホットプラグされたメモリーデバイスのみです。要求されたメモリーの減少分は、以前にホットプラグされたメモリーデバイスの組み合わせのサイズに一致するように切り捨てられます。要求されたメモリー値は、丸められずに次の実行設定に格納されます。



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

Red Hat Virtualization Manager は内部的に値を切り捨てて整数の MiB (1MiB = 2^{20} バイト) にします。

7.145.10. migration

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。



注記

このメソッドが返す ID で移行ポリシーを照会する API はまだ実装されていません。**/ovirt-engine/api/options/MigrationPolicies** を使用して、すべての移行ポリシーとその ID のリストを取得します。

7.145.11. 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。

7.145.12. 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**

7.145.13. 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>

```

7.145.14. 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。

7.145.15. 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。

7.145.16. tpm_enabled

true の場合、TPM デバイスが仮想マシンに追加されます。デフォルト値は **false** です。このプロパティは、"All-Content=true" ヘッダーが設定されている場合に、フェッチするときのみ表示されません。

表7.196 リンクの概要

名前	タイプ	概要
cdroms	Cdrom[]	テンプレートに接続された CD-ROM デバイスへの参照。
cluster	クラスター	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。
mediated_devices	VmMediatedDevice[]	仲介デバイスの設定。
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。

名前	タイプ	概要
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。

7.146. IO 構造体

表7.197 属性の概要

名前	タイプ	概要
threads	Integer	

7.147. 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7.198 属性の概要

名前	タイプ	概要
address	文字列	IP アドレスのテキスト表現。
gateway	文字列	デフォルトゲートウェイのアドレス。
netmask	文字列	ネットワークマスク。
version	IpVersion	IP プロトコルのバージョン。

7.147.1. 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>
```

7.147.2. 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。

7.147.3. 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。

7.148. 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7.199 属性の概要

名前	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。

7.149. IPVERSION ENUM

IP プロトコルバージョンの値を定義します。

表7.200 値の概要

名前	概要
v4	IPv4。

名前	概要
v6	IPv6。

7.150. ISCSIBOND STRUCT

表7.201 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.202 リンクの概要

名前	タイプ	概要
data_center	DataCenter	
networks	Network[]	
storage_connections	StorageConnection[]	

7.151. ISCSIDetails STRUCT

表7.203 属性の概要

名前	タイプ	概要
address	文字列	
disk_id	文字列	
initiator	文字列	
lun_mapping	Integer	
password	文字列	

名前	タイプ	概要
paths	Integer	
port	Integer	
portal	文字列	
product_id	文字列	
serial	文字列	
size	Integer	
status	文字列	
storage_domain_id	文字列	
target	文字列	
username	文字列	
vendor_id	文字列	
volume_group_id	文字列	

7.152. 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7.204 属性の概要

名前	タイプ	概要
auto_cleared	Boolean	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	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
end_time	Date	ジョブの終了時刻。

名前	タイプ	概要
external	Boolean	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	文字列	一意の ID
last_updated	Date	ジョブの最終更新日。
name	文字列	人間が判読できるプレーンテキストでの名前。
start_time	Date	ジョブの開始時刻。
status	JobStatus	ジョブのステータス。

7.152.1. 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7.205 リンクの概要

名前	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。

7.153. JOBSTATUS ENUM

ステップのステータスを表します。

表7.206 値の概要

名前	概要
aborted	中止されたジョブのステータス。
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。

7.153.1. 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。

7.153.2. 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。

7.153.3. 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。

7.153.4. 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。

7.154. KATELLOERRATUM STRUCT

Katello erratum を表すタイプ。

表7.207 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
issued	Date	Katello エラータが発行された日付。
name	文字列	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。
severity	文字列	Katello エラータの重大度。
solution	文字列	Katello エラータで記述されている問題のソリューション。
summary	文字列	Katello エラータの要約。
title	文字列	Katello エラータのタイトル。
type	文字列	Katello エラータのタイプ。

7.154.1. 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。

7.154.2. 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7.208 リンクの概要

名前	タイプ	概要
ホスト	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。

7.155. KDUMPSTATUS ENUM

表7.209 値の概要

名前	概要
disabled	
enabled	
unknown	

7.156. KERNEL 構造体

表7.210 属性の概要

名前	タイプ	概要
version	バージョン	

7.157. KSM 構造体

表7.211 属性の概要

名前	タイプ	概要
enabled	Boolean	
merge_across_nodes	Boolean	

7.158. LINKLAYERDISCOVERYPROTOCOLELEMENTSTRUCT

Link Layer Discovery Protocol (LLDP) によって受信された情報要素を表します。IEEE 802.1AB は、タイプ、長さ、値 (TLV) を「情報要素の短い可変長エンコーディング」と定義しています。このタイプは、そのような情報要素を表します。

属性 **name** は、値の内容を説明するために使用される人間が判読できる文字列であり、一意ではない場合があります。**type** とオプションの **oui** および **subtype** から作成できるため、この名前は冗長です。**name** の目的は、情報要素の読み取りを簡略化することです。プロパティの **name** は、IEEE 802.1AB の第 8 章で使用されている文字列とまったく同じです。

組織固有の情報要素には、**127** の **type** と、**oui** および **subtype** の属性があります。

たとえば、情報要素の XML 表現は次のようになります。

```
<link_layer_discovery_protocol_element>
  <name>Port VLAN Id</name>
  <oui>32962</oui>
  <properties>
    <property>
      <name>vlan id</name>
      <value>488</value>
    </property>
    <property>
      <name>vlan name</name>
      <value>v2-0488-03-0505</value>
    </property>
  </properties>
  <subtype>3</subtype>
  <type>127</type>
</link_layer_discovery_protocol_element>
```

表7.212 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
oui	Integer	整数としてエンコードされた組織固有識別子 (OUI)。
properties	Property[]	情報要素によって転送される構造化データを、名前と値のペアのリストとして表します。
subtype	Integer	整数としてエンコードされた組織的に定義されたサブタイプ。

名前	タイプ	概要
type	Integer	整数としてエンコードされた LinkLayerDiscoveryProtocolElement のタイプ。

7.158.1. oui

整数としてエンコードされた組織固有識別子 (OUI)。 **type** が **127** の場合のみ有効です。

7.158.2. subtype

整数としてエンコードされた組織的に定義されたサブタイプ。 **type** が **127** の場合のみ有効です。

7.159. LOGMAXMEMORYUSEDTHRESHOLDTYPE ENUM

システムでサポートされているすべての最大メモリしきい値タイプについて説明します。

表7.213 値の概要

名前	概要
absolute_value_in_mb	絶対値でのしきい値タイプ。
percentage	パーセンテージでのしきい値タイプ。

7.159.1. absolute_value_in_mb

絶対値でのしきい値タイプ。

絶対値を指定すると、MB 単位の空きメモリーが **LogMaxMemoryUsedThreshold** で指定された値を下回ると、監査ログイベントがログに記録されます。

7.159.2. percentage

パーセンテージでのしきい値タイプ。

パーセンテージを指定すると、使用されているメモリーが **LogMaxMemoryUsedThreshold** で指定された値を超えると、監査ログイベントがログに記録されます。

7.160. LOGSEVERITY ENUM

イベントの重大度を表す列挙型。

表7.214 値の概要

名前	概要
alert	アラートの重大度。

名前	概要
error	エラーの重大度。
normal	通常の重大度。
warning	警告の重大度。

7.160.1. 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。

7.160.2. 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。

7.160.3. normal

通常の重大度。情報イベントに使用されます。

7.160.4. 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。

7.161. LOGICALUNIT 構造体

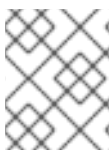
表7.215 属性の概要

名前	タイプ	概要
address	文字列	
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	Boolean	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	文字列	
id	文字列	
lun_mapping	Integer	
password	文字列	
paths	Integer	

名前	タイプ	概要
port	Integer	
portal	文字列	
product_id	文字列	
serial	文字列	
size	Integer	
status	LunStatus	
storage_domain_id	文字列	
target	文字列	
username	文字列	
vendor_id	文字列	
volume_group_id	文字列	

7.161.1. 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 ドキュメント](#) 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。

7.161.2. 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 documentation](#) ドキュメントを参照してください。



重要

システムのバージョン 4.2.1 以降、カーネルの sysfs ファイル **discard_zeroes_data** が非推奨になったため、この属性のサポートが削除されました。後方互換性のために保持されていますが、値は常に **false** になります。

7.162. LUNSTATUS ENUM

表7.216 値の概要

名前	概要
free	
unusable	
used	

7.163. MDEVTYPE STRUCT

仲介デバイスは、物理デバイスのリソースを分割できるようにするソフトウェアデバイスです。

詳細は [Libvirt-MDEV](#) を参照してください。

表7.217 属性の概要

名前	タイプ	概要
available_instances	Integer	MDev タイプの使用可能なインスタンス数。
description	文字列	MDev タイプの説明。
human_readable_name	文字列	MDev タイプの人間が読める名前。
name	文字列	MDev タイプ名。

7.164. MAC 構造体

仮想ネットワークインターフェイスの MAC アドレスを表します。

表7.218 属性の概要

名前	タイプ	概要
address	文字列	MAC アドレス

7.165. 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>

```

表7.219 属性の概要

名前	タイプ	概要
allow_duplicates	Boolean	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	文字列	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	Boolean	これがデフォルトプールかどうかを定義します。
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	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。

7.165.1. 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。

7.165.2. 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。

7.165.3. 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。

表7.220 リンクの概要

名前	タイプ	概要
----	-----	----

名前	タイプ	概要
permissions	Permission[]	MacPool に関連付けられているパーミッションへの参照を返します。

7.166. MEMORYOVERCOMMIT STRUCT

表7.221 属性の概要

名前	タイプ	概要
percent	Integer	

7.167. 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プロパティの論理グループ。

表7.222 属性の概要

名前	タイプ	概要
ballooning	Boolean	
guaranteed	Integer	バルーンメカニズムによって排出されないことが保証されているメモリーの量 (バイト単位)。
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	

7.167.1. guaranteed

バルーンメカニズムによって排出されないことが保証されているメモリーの量 (バイト単位)。

Red Hat Virtualization Manager は、この値を内部的に切り捨てて 整数の MiB ($1\text{MiB} = 2^{20}$ バイト) にします。



注記

Red Hat Virtualization 4.2 以降では、同じ要求で **メモリー** も更新され、仮想マシンの状態が **up** であれば、仮想マシンの実行中に更新できます。

7.167.2. 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、Red Hat Virtualization Manager は値を切り捨てて最も近い下位の MiB 値にします。

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。

7.168. MESSAGEBROKERTYPE ENUM

非推奨のメッセージブローカータイプ。

Red Hat Virtualization 4.4.0 以降、OpenStack Neutron エージェントのデプロイメントが削除されたため、無視されます。

表7.223 値の概要

名前	概要
qpid	
rabbit_mq	

7.169. METHOD 構造体

表7.224 属性の概要

名前	タイプ	概要
id	SsoMethod	

7.170. MIGRATEONERROR ENUM

表7.225 値の概要

名前	概要
do_not_migrate	
migrate	
migrate_highly_available	

7.171. MIGRATIONBANDWIDTH 構造体

移行で使用される帯域幅を定義します。

表7.226 属性の概要

名前	タイプ	概要
assignment_method	MigrationBandwidthAssignmentMethod	帯域幅を割り当てるのに使用するメソッド。
custom_value	Integer	Mbps 単位のカスタム帯域幅。

7.171.1. 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。

7.172. MIGRATIONBANDWIDTHHASSIGNMENTMETHOD ENUM

移行帯域幅の割り当て方法を定義します。

表7.227 値の概要

名前	概要
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。
custom	カスタム定義の帯域幅 (Mbit/s)。
hypervisor_default	ハイパーバイザーで設定された値を取得します。

7.172.1. 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。

7.173. MIGRATIONOPTIONS 構造体

移行オプションのタイプ。

表7.228 属性の概要

名前	タイプ	概要
auto_converge	InheritableBoolean	

名前	タイプ	概要
bandwidth	MigrationBandwidth	移行で使用できる帯域幅。
compressed	InheritableBoolean	
custom_parallel_migrations	Integer	使用する並行移行接続の数を指定します。
encrypted	InheritableBoolean	移行を暗号化するかどうかを指定します。
parallel_migrations_policy	ParallelMigrationsPolicy	並列移行接続を使用するかどうか、および使用方法を指定します。

7.173.1. custom_parallel_migrations

使用する並行移行接続の数を指定します。ParallelMigrationsPolicy が CUSTOM の場合のみ指定できます。有効な値の範囲は 2-255 です。推奨される値の範囲は 2-16 です。

表7.229 リンクの概要

名前	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。

7.174. 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7.230 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

7.175. 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7.231 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。

名前	タイプ	概要
display	Boolean	非推奨となりました。ネットワークをディスプレイネットワークとして定義するには、usages を使用する必要があります。
dns_resolver_configuration	DnsResolverConfiguration	GET を使用してネットワークを取得すると、DNS リゾルバーの設定が報告されます。
id	文字列	一意の ID
ip	Ip	非推奨 (使用されていません)。
mtu	Integer	ネットワークの最大伝送単位を指定します。
name	文字列	人間が判読できるプレーンテキストでの名前。
port_isolation	Boolean	同じホストで実行されている VM 間の通信をこのネットワークでブロックするかどうかを定義します。
profile_required	Boolean	ネットワークの作成時に仮想ネットワークインターフェイスプロファイルを自動的に作成するかどうかを指定します。
必須	Boolean	クラスター内のすべてのホストに対して、ネットワークが必須であるかどうかを定義します。
status	NetworkStatus	ネットワークのステータス。
stp	Boolean	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。
vds_name	文字列	ホストで使用されているネットワークの名前。
vlan	Vlan	VLAN タグ。

7.175.1. dns_resolver_configuration

GET を使用してネットワークを取得すると、DNS リゾルバーの設定が報告されます。新規にネットワークを作成する場合も、既存のネットワークを更新する場合も、任意で設定することができます。

7.175.2. port_isolation

同じホストで実行されている VM 間の通信をこのネットワークでブロックするかどうかを定義します。VM ネットワークにのみ適用されます。複数のホスト間の通信がブロックされていることを確認するのは、ネットワーク管理者の責任です。この属性は、ネットワークの作成時にのみ設定でき、編集することはできません。値が設定されていない場合、同じホストで実行されている VM 間の通信が許可されます。

7.175.3. 必須

クラスター内のすべてのホストに対して、ネットワークが必須であるかどうかを定義します。'required' の **operational** ネットワークがホストから省略された場合、ホストは **non_operational** とマークされます。

7.175.4. status

ネットワークのステータス。ネットワークが 'required' と定義され、アクティブなクラスターホストから省略されている場合は **non_operational** になります。それ以外の場合は **operational** になります。

7.175.5. 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。

7.175.6. vdsname

ホストで使用されているネットワークの名前。この代替名は、ネットワーク名がホストのブリッジ名として機能するのに不適切であることが判明した場合に、VDSM によって自動的に生成されます。不適切な名前にはスペースまたは特殊文字が含まれているか、15 文字を超えており、ホスト上の UUID に置き換えられます。このパラメーターは読み取り専用です。設定しても効果はありません。

表7.232 リンクの概要

名前	タイプ	概要
cluster	クラスター	このネットワークがアタッチされているクラスターへの参照。
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。
external_provider	OpenStackNetworkProvider	ネットワークが作成される OpenStack ネットワークプロバイダーへのオプションの参照。
external_provider_physical_network	Network	物理ネットワークアクセスに使用する必要があるネットワークへのオプションの参照。
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	Quality of Service への参照。
vnic_profiles	VnicProfile[]	ネットワークのプロファイルへの参照。

7.175.7. cluster

このネットワークがアタッチされているクラスターへの参照。クラスターレベルからネットワークにアクセスする場合にのみ入力されます。

7.175.8. external_provider

ネットワークが作成される OpenStack ネットワークプロバイダーへのオプションの参照。

ネットワークの作成時に指定すると、一致する OpenStack ネットワークも作成されます。

7.175.9. external_provider_physical_network

物理ネットワークアクセスに使用する必要があるネットワークへのオプションの参照。**external_provider** が指定されている場合にのみ有効です。

7.176. 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のようになります。

```
<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7.233 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	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	文字列	一意の ID
in_sync	Boolean	
ip_address_assignments	IpAddressAssignment[]	ネットワークの IP 設定。
name	文字列	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。

7.176.1. 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。

7.176.2. 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。

名前	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
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7.234 リンクの概要

名前	タイプ	概要
ホスト	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	

7.177. NETWORKCONFIGURATION 構造体

表7.235 属性の概要

名前	タイプ	概要
dns	Dns	
nics	Nic[]	

7.178. 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。

デフォルトのネットワークフィルターは、ネットワークの種類と設定に基づいています。 **EnableMACAntiSpoofingFilterRules** が True の場合、仮想マシンネットワークのデフォルトフィルターは **vds-no-mac-spoof** です。それ以外の場合、フィルターは設定されません。 **OVN** ネットワークの場合、フィルターは設定されません。

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 **vds-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7.236 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	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。

7.178.1. 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。

7.179. NETWORKFILTERPARAMETER 構造体

ネットワークフィルターのパラメーター。

詳細については、[Libvirt-Filters](#) を参照してください。これは XML 表現の例です。

```
<network_filter_parameter id="123">
  <name>IP</name>
  <value>10.0.1.2</value>
</network_filter_parameter>
```

表7.237 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
value	文字列	パラメーターの値を表します。

表7.238 リンクの概要

名前	タイプ	概要
nic	Nic	パラメーターが関連付けられている仮想マシンの NIC。

7.180. NETWORKLABEL 構造体

ホストネットワークインターフェイスとネットワークに追加できるラベルを表します。ラベルは、ラベル **id** によってネットワークとホストネットワークインターフェイスをバインドします。

表7.239 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.240 リンクの概要

名前	タイプ	概要
host_nic	HostNic	このラベルを含むホストネットワークインターフェイスへの参照。
network	Network	このラベルを含むネットワークへの参照。

7.181. NETWORKPLUGINTYPE ENUM

ネットワークプラグインタイプ。

ホスト上のプロバイダードライバーの実装を指定します。

Red Hat Virtualization Manager のバージョン 4.2 以降、このタイプは非推奨となり、**OpenStackNetworkProvider** タイプの **external_plugin_type** 属性が使用されるようになりました。

表7.241 値の概要

名前	概要
open_vswitch	Open vSwitch

7.181.1. open_vswitch

Open vSwitch

このプロバイダーに [Open vSwitch](#) ベースのドライバー実装を使用する必要があることを指定します。

Red Hat Virtualization Manager のバージョン 4.2 以降、この値は非推奨になりました。代わりに、**OpenStackNetworkProvider.external_plugin_type** 属性で文字列 **open_vswitch** を使用してください。

7.182. NETWORKSTATUS ENUM

表7.242 値の概要

名前	概要
non_operational	
operational	

7.183. 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7.243 値の概要

名前	概要
default_route	ホストのデフォルトゲートウェイと DNS リゾルバー設定は、このネットワークから取得されます。
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	

7.183.1. default_route

ホストのデフォルトゲートウェイと DNS リゾルバー設定は、このネットワークから取得されます。

このネットワークがホストにアタッチされている場合、DNS リゾルバー設定は、ネットワークアタッチメントの **dns_resolver_configuration** 属性から取得されます。このネットワークアタッチメントに **dns_resolver_configuration** 属性がない場合は、ネットワーク自体の **dns_resolver_configuration** から取得されます。そこにも **dns_resolver_configuration** 属性が存在しない場合、DNS リゾルバーは設定されません。

このフラグをネットワークに設定すると、ホストのデフォルトゲートウェイは、ネットワークアタッチメントの `ip_address_assignment` の `gateway` 属性から取得されます。

7.183.2. 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。

7.184. NFSPROFILEDETAIL STRUCT

表7.244 属性の概要

名前	タイプ	概要
<code>nfs_server_ip</code>	文字列	
<code>profile_details</code>	<code>ProfileDetail[]</code>	

7.185. NFSVERSION ENUM

表7.245 値の概要

名前	概要
<code>auto</code>	
<code>v3</code>	
<code>v4</code>	
<code>v4_0</code>	NFS 4.
<code>v4_1</code>	
<code>v4_2</code>	NFS 4.

7.185.1. v4_0

NFS 4.0.

7.185.2. v4_2

NFS 4.2.

7.186. 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7.246 属性の概要

名前	タイプ	概要
boot_protocol	BootProtocol	NIC に IP アドレスを割り当てる方法を定義します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
interface	NicInterface	NIC に使用されるドライバーのタイプ。
linked	Boolean	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	文字列	人間が判読できるプレーンテキストでの名前。
on_boot	Boolean	オペレーティングシステムの起動時にネットワークインターフェイスをアクティブにするかどうかを定義します。
plugged	Boolean	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。
synced	Boolean	仮想マシンの NIC 設定がエンジンによって表される設定と同期されるかどうかを定義します。

表7.247 リンクの概要

名前	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。

名前	タイプ	概要
network_attachments	NetworkAttachment[]	ホスト NIC に関連付けられているネットワークアタッチメントのコレクションへのリンク。
network_filter_parameters	NetworkFilterParameter[]	ネットワークフィルターパラメーターへのリンク。
network_labels	NetworkLabel[]	ホスト NIC に関連付けられているネットワークラベルのコレクションへのリンク。
reported_devices	ReportedDevice[]	仮想ネットワークインターフェイスに関連付けられている報告されたデバイスのコレクションへのリンク。
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[]	SR-IOV NIC の仮想機能にアタッチできるネットワークラベルのコレクションへのリンク。
virtual_function_allowed_networks	Network[]	SR-IOV NIC の仮想機能にアタッチできるネットワークのコレクションへのリンク。
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	関連する仮想ネットワークインターフェイスプロファイルへのリンク。

7.186.1. 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。

7.186.2. 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。

7.187. NICCONFIGURATION 構造体

タイプは、仮想ネットワークインターフェイスの設定を表します。

表7.248 属性の概要

名前	タイプ	概要
boot_protocol	BootProtocol	IPv4 ブートプロトコル。
ip	Ip	IPv4 アドレスの詳細。
ipv6	Ip	IPv6 アドレスの詳細。
ipv6_boot_protocol	BootProtocol	IPv6 ブートプロトコル。
name	文字列	ネットワークインターフェイス名。
on_boot	Boolean	仮想マシンのゲストオペレーティングシステムの起動時にネットワークインターフェイスをアクティブにするかどうかを指定します。

7.188. NICINTERFACE ENUM

エミュレートされた仮想ネットワークインターフェイスデバイスモデルのオプションを定義します。

表7.249 値の概要

名前	概要
e1000	e1000。
e1000e	e1000e。
pci_passthrough	PCI パススルー。
rtl8139	rtl8139。
rtl8139_virtio	デュアルモード rtl8139、VirtIO。
spapr_vlan	sPAPR VLAN。
virtio	VirtIO。

7.189. NICSTATUS ENUM

ネットワークインターフェイスカードのステータス。

表7.250 値の概要

名前	概要
down	NIC がダウンしており、アクセスできません。
up	NIC が起動しており、アクセスできます。

7.190. NOTIFIABLEEVENT ENUM

Red Hat Virtualization サーバーのイベントのサブセットを表すタイプ。ユーザーは、これをサブスクライブして通知を受け取ることができます。

表7.251 値の概要

名前	概要
cluster_alert_ha_reservation	HA 予約チェックに失敗しました
cluster_alert_ha_reservation_down	HA 予約チェックに合格しました
dwh_error	ETL サービスエラー
dwh_stopped	ETL サービスが停止しました
engine_backup_completed	エンジンのバックアップが正常に完了しました
engine_backup_failed	エンジンのバックアップに失敗しました
engine_backup_started	エンジンのバックアップが開始しました
engine_ca_certification_has_expired	エンジン CA の認証の有効期限が切れています
engine_ca_certification_is_about_to_expire	エンジンの CA 証明書の有効期限がまもなく切れます
engine_certification_has_expired	エンジンの認証の有効期限が切れています

名前	概要
engine_certification_is_about_to_expire	エンジンの 証明書の有効期限がまもなく切れます
engine_stop	エンジンが停止しました
faulty_multipaths_on_host	ホスト上にある、障害のあるマルチパスのパス
gluster_brick_status_changed	ブリックの状態変化を検出しました
gluster_hook_added_failed	競合するサーバーに Gluster Hook を追加できませんでした
gluster_hook_added	Gluster Hook を追加しました
gluster_hook_conflict_detected	Gluster Hook で競合が検出されました
gluster_hook_detected_delete	Gluster Hook の削除が検出されました
gluster_hook_detected_new	新しい Gluster Hook を検出しました
gluster_hook_disable	Gluster フックが無効化されました
gluster_hook_disable_failed	Gluster Hook を無効化できませんでした
gluster_hook_enable	Gluster Hook が有効化されました
gluster_hook_enable_failed	Gluster フックを有効化できませんでした
gluster_hook_remove_failed	クラスターから Gluster Hook を削除できませんでした
gluster_hook_removed	Gluster Hook を削除しました
gluster_server_add_failed	Gluster サーバーを追加できませんでした

名前	概要
gluster_server_remove	Gluster サーバーが削除されました
gluster_server_remove_failed	Gluster サーバーの削除に失敗しました
gluster_service_restart_failed	Gluster サービスの再起動に失敗しました
gluster_service_restarted	Gluster サービスが再起動されました
gluster_service_start_failed	Gluster サービスの開始に失敗しました
gluster_service_started	Gluster サービスが開始されました
gluster_service_stop_failed	Gluster サービスを停止できませんでした
gluster_service_stopped	Gluster サービスが停止しました
gluster_volume_add_brick	Gluster ボリュームブリックが追加されました
gluster_volume_add_brick_failed	Gluster ボリュームにブリックを追加できませんでした
gluster_volume_all_snapshots_delete_failed	ボリュームのスナップショットを削除できませんでした
gluster_volume_all_snapshots_deleted	ボリュームですべてのスナップショットが削除されました
gluster_volume_brick_replaced	グラスターボリュームブリックが置き換えられました
gluster_volume_confirmed_space_low	ボリュームの空き容量が少ないことが確認されました
gluster_volume_create	Gluster ボリュームが作成されました

名前	概要
gluster_volume_create_failed	Gluster ボリュームを作成できませんでした
gluster_volume_delete	Gluster ボリュームが削除されました
gluster_volume_delete_failed	Gluster ボリュームを削除できませんでした
gluster_volume_migrate_brick_data_finished	ブリックを削除するための Gluster ボリュームのデータ移行が完了しました
gluster_volume_option_added	Gluster ボリュームオプションが追加されました
gluster_volume_option_modified	Gluster ボリュームオプションが変更されました
gluster_volume_option_set_failed	Gluster ボリュームオプションを設定できませんでした
gluster_volume_options_reset	Gluster ボリュームオプションがリセットされました
gluster_volume_options_reset_all	すべての Gluster ボリュームオプションがリセットされました
gluster_volume_options_reset_failed	Gluster ボリュームオプションをリセットできませんでした
gluster_volume_profile_start	Gluster ボリュームプロファイルを開始しました
gluster_volume_profile_start_failed	Gluster ボリュームプロファイルの開始に失敗しました

名前	概要
gluster_volume_profile_stop	Gluster ボリュームプロファイルが停止しました
gluster_volume_profile_stop_failed	Gluster ボリュームプロファイルの停止に失敗しました
gluster_volume_rebalance_finished	Gluster ボリュームのリバランスが終了しました
gluster_volume_rebalance_not_found_from_cli	CLI からボリューム上のリバランスの情報が見つかりませんでした。
gluster_volume_rebalance_start	Gluster ボリュームリバランスが開始されました
gluster_volume_rebalance_start_detected_from_cli	CLI から gluster ボリュームのリバランスの開始が検出されました
gluster_volume_rebalance_start_failed	Gluster ボリュームリバランスを開始できませんでした
gluster_volume_rebalance_stop	Gluster ボリュームのリバランスが停止しました
gluster_volume_rebalance_stop_failed	Gluster ボリュームのリバランスを停止できませんでした
gluster_volume_remove_bricks	Gluster ボリュームブリックが削除されました
gluster_volume_remove_bricks_failed	Gluster ボリュームブリックを削除できませんでした
gluster_volume_remove_bricks_stop	Gluster ボリュームからのブリックの削除を停止しました

名前	概要
gluster_volume_remove_bricks_stop_failed	Gluster ボリュームからのブリックの削除を停止できませんでした
gluster_volume_replace_brick_failed	Gluster ボリュームのブリック置き換えに失敗しました
gluster_volume_replace_brick_start	Gluster ボリュームのブリック置き換えが開始されました
gluster_volume_replace_brick_start_failed	Gluster ボリュームのブリック置き換えを開始できませんでした
gluster_volume_snapshot_activate_failed	ボリュームでスナップショットをアクティブ化できませんでした
gluster_volume_snapshot_activated	ボリュームでスナップショットがアクティブ化されました
gluster_volume_snapshot_create_failed	<code>\${clusterName}</code> クラスタで <code>\${glusterVolumeName}</code> のボリュームのスナップショットが作成できませんでした。
gluster_volume_snapshot_created	<code>\${clusterName}</code> クラスタで <code>\${glusterVolumeName}</code> ボリュームの <code>\${snapname}</code> スナップショットが作成されました。
gluster_volume_snapshot_deactivate_failed	ボリュームのスナップショットを非アクティブ化できませんでした
gluster_volume_snapshot_deactivated	ボリュームのスナップショットが非アクティブ化されました
gluster_volume_snapshot_delete_failed	ボリュームのスナップショットを削除できませんでした

名前	概要
gluster_volume_snapshot_deleted	ボリュームのスナップショットが削除されました
gluster_volume_snapshot_restore_failed	ボリュームのスナップショットを復元できませんでした
gluster_volume_snapshot_restored	ボリュームのスナップショットが復元されました
gluster_volume_start	Gluster ボリュームが開始されました
gluster_volume_start_failed	Gluster ボリュームを開始できませんでした
gluster_volume_stop	Gluster ボリュームが停止しました
gluster_volume_stop_failed	Gluster ボリュームを停止できませんでした
ha_vm_failed	高可用性 仮想マシンが失敗しました
ha_vm_restart_failed	高可用性 VM の再起動に失敗しました
host_activate_failed	ホストのアクティブ化に失敗しました
host_activate_manual_ha	ホストはアクティブ化されましたが、ホスト型エンジン HA サービスは引き続きメンテナンスモードである可能性があります
host_approve_failed	ホストの承認に失敗しました
host_bond_slave_state_down	ホストのボンディングのスレーブが状態をダウンに変更しました
host_certificate_has_invalid_san	ホストの証明書に無効なサブジェクト代替名 (SAN) が含まれています
host_certificate_has_expired	ホストの認定の有効期限が切れています

名前	概要
host_certification_is_about_to_expire	ホストの証明書の有効期限がまもなく切れます
host_failure	ホストが応答しません
host_high_cpu_use	ホストの CPU 使用量が定義されたしきい値を超えました
host_high_mem_use	ホストのメモリー使用量が定義されたしきい値を超えました
host_high_swap_use	ホストスワップのメモリー使用量が定義されたしきい値を超えました
host_initiated_run_vm_failed	別のホストで VM を再起動できませんでした
host_install_failed	ホストのインストールに失敗しました
host_interface_high_network_use	ホストネットワークインターフェイスの使用量が定義されたしきい値を超えました
host_interface_state_down	ホストのインターフェイスの状態がダウンに変更されました
host_low_mem	ホストの空きメモリーが定義されたしきい値を下回っています
host_low_swap	ホストの空きスワップメモリーが定義されたしきい値を下回っています
host_recover_failed	ホストの復旧に失敗しました
host_set_nonoperational	ホストの状態が non-operational に設定されました
host_set_nonoperational_domain	ストレージドメインにアクセスできないため、ホストの状態が操作不能に設定されました
host_set_nonoperational_iface_down	インターフェイスがないため、ホストの状態が non-operational に設定されました

名前	概要
host_slow_storage_response_time	ストレージの応答時間が遅いです
host_time_drift_alert	ホストには時間のずれがあります
host_untrusted	ホストの状態が non-operational に設定されました。
host_updates_are_available	ホストには利用可能な更新があります
host_updates_are_available_with_packages	ホストには更新可能なパッケージがあります
importexport_import_template_from_trusted_to_untrusted	テンプレートが信頼できるクラスターから信頼できないクラスターにインポートされました
importexport_import_template_from_untrusted_to_trusted	テンプレートが信頼できないクラスターから信頼できるクラスターにインポートされました
importexport_import_vm_from_trusted_to_untrusted	仮想マシンを信頼できるクラスターから信頼できないクラスターにインポートします
importexport_import_vm_from_untrusted_to_trusted	仮想マシンを信頼できないクラスターから信頼できるクラスターにインポートします
irs_confirmed_disk_space_low	ディスク容量が少ないことが確認されました
irs_disk_space_low	ディスク領域不足
irs_disk_space_low_error	ディスク容量が極端に少ない
IRS_FAILURE	ストレージへのアクセスに失敗しました

名前	概要
<code>mac_address_is_external</code>	外部 MAC アドレスを持つ VM
<code>multipath_devices_without_valid_paths_on_host</code>	ホストに有効なパスがないマルチパスデバイス
<code>network_update_display_for_cluster_with_active_vm</code>	アクティブな VM を持つクラスターでディスプレイネットワークが更新されました
<code>network_update_display_for_host_with_active_vm</code>	アクティブな VM を持つホストでディスプレイネットワークが更新されました
<code>no_faulty_multipaths_on_host</code>	ホスト上に障害のあるマルチパスのパスはありません
<code>number_of_lvs_on_storage_domain_exceeded_threshold</code>	ストレージドメインの LV の数がしきい値を超えました
<code>remove_gluster_volume_bricks_not_found_from_cli</code>	CLI からボリューム上のブリックを削除するための情報が見つかりませんでした。
<code>start_removing_gluster_volume_bricks</code>	ボリュームからブリックを削除し始めました
<code>start_removing_gluster_volume_bricks_detected_from_cli</code>	CLI からのボリューム上のブリックのブリック除去の開始が検出されました
<code>start_removing_gluster_volume_bricks_failed</code>	ボリュームブリックを削除できませんでした
<code>system_change_storage_pool_status_no_host_for_spm</code>	データセンターの SPM の選択に失敗しました

名前	概要
system_deactivated_storage_domain	ストレージドメインの状態が非アクティブに設定されました
user_add_vm_from_trusted_to_untrusted	信頼できない VM が信頼できるテンプレートから作成されました
user_add_vm_from_untrusted_to_trusted	信頼できる VM は、信頼できないテンプレートから作成されました
user_add_vm_template_from_trusted_to_untrusted	信頼できないテンプレートが信頼できる VM から作成されました
user_add_vm_template_from_untrusted_to_trusted	信頼できるテンプレートが信頼できない VM から作成されました
user_host_maintenance	ホストが Maintenance Mode に切り替えられました
user_host_maintenance_manual_ha	ホストは、メンテナンスモードに切り替わりましたが、ホストエンジンの HA メンテナンスモードを有効化できませんでした
user_host_maintenance_migration_failed	ホストを Maintenance Mode に切り替えることができませんでした
user_update_vm_from_trusted_to_untrusted	VM が信頼できるクラスターから信頼できないクラスターに移動しました
user_update_vm_from_untrusted_to_trusted	VM が信頼できないクラスターから信頼できるクラスターに移動しました
user_update_vm_template_from_trusted_to_untrusted	テンプレートが信頼できるクラスターから信頼できないクラスターに移動しました

名前	概要
user_update_vm_template_from_untrusted_to_trusted	テンプレートが信頼できないクラスターから信頼できるクラスターに移動しました
vm_console_connected	接続された VM コンソール
vm_console_disconnected	VM コンソールが切断されました
vm_down_error	VM がエラーを表示してダウンしています
vm_failure	ホストで VM が見つかりません
vm_migration_failed	移行に失敗しました
vm_migration_start	VM の移行を開始します
vm_migration_to_server_failed	移行先ホストへの VM の移行に失敗しました
vm_not_responding	VM が応答していません
vm_paused	VM が一時停止されました
vm_paused_eio	ストレージ I/O エラーのために VM が一時停止されました
vm_paused_enospc	ストレージスペースが不足しているため、VM が一時停止されました
vm_paused_eperm	ストレージの読み取り/書き込み権限の問題により、VM が一時停止されました
vm_paused_error	仮想マシンは不明なストレージエラーが原因で一時停止されました
vm_recovered_from_pause_error	仮想マシンが一時停止から稼働中に復元されました
vm_set_ticket	VM コンソールセッションが開始されました

名前	概要
vm_status_restored	VM ステータスが復元されました

7.190.1. gluster_volume_rebalance_not_found_from_cli

CLI からボリューム上のリバランスの情報が見つかりませんでした。不明とマークします。

7.190.2. host_untrusted

ホストの状態が non-operational に設定されました。ホストは認証サービスによって信頼されていません

7.190.3. remove_gluster_volume_bricks_not_found_from_cli

CLI からボリューム上のブリックを削除するための情報が見つかりませんでした。不明とマークします。

7.191. NOTIFICATIONMETHOD ENUM

イベントサブスクリプションの通知方法を表すタイプ。現在 API でサポートされているのは SMTP のみです。将来的には、SNMP 通知のサポートが追加される可能性があります。

表7.252 値の概要

名前	概要
smtp	電子メールによる通知。
snmp	SNMP による通知。

7.191.1. smtp

電子メールによる通知。

SMTP 通知メソッドを使用するイベントサブスクリプションには、アドレスフィールドに電子メールアドレスが含まれます。

7.191.2. snmp

SNMP による通知。

SNMP 通知メソッドを使用するイベントサブスクリプションには、アドレスフィールドに SNMP アドレスが含まれます。

7.192. 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7.253 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
index	Integer	
memory	Integer	NUMA ノードのメモリー (MB)。
name	文字列	人間が判読できるプレーンテキストでの名前。
node_distance	文字列	

表7.254 リンクの概要

名前	タイプ	概要
ホスト	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。

7.192.1. 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。

名前	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。

7.193. 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7.255 属性の概要

名前	タイプ	概要
host_numa_node	NumaNode	非推奨。

名前	タイプ	概要
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。
pinned	Boolean	非推奨。

7.193.1. host_numa_node

非推奨。機能はありません。

7.193.2. pinned

非推奨。常に **true** である必要があります。

7.194. NUMATUNEMODE ENUM

表7.256 値の概要

名前	概要
interleave	
preferred	
strict	

7.195. OPENSTACKIMAGE 構造体

表7.257 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.258 リンクの概要

名前	タイプ	概要
openstack_image_provider	OpenStackImageProvider	

7.196. OPENSTACKIMAGEPROVIDERSTRUCT

表7.259 属性の概要

名前	タイプ	概要
authentication_url	文字列	外部プロバイダー認証 URL アドレスを定義します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
password	文字列	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	Boolean	プロバイダー認証が必要かどうかを定義します。
tenant-name	文字列	OpenStack Identity API v2 のテナント名を定義します。
url	文字列	外部プロバイダーの URL アドレスを定義します。
username	文字列	認証プロセス中に使用されるユーザー名を定義します。

7.196.1. 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。

7.196.2. tenant-name

OpenStack Identity API v2.0 のテナント名を定義します。

表7.260 リンクの概要

名前	タイプ	概要
certificates	Certificate[]	
images	OpenStackImage[]	

7.197. OPENSTACKNETWORK 構造体

表7.261 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.262 リンクの概要

名前	タイプ	概要
openstack_net work_provider	OpenStackNetworkProvider	

7.198. OPENSTACKNETWORKPROVIDER 構造体

表7.263 属性の概要

名前	タイプ	概要
agent_configuration	AgentConfiguration	非推奨エージェントの設定オプション。
authentication_url	文字列	外部プロバイダー認証 URL アドレスを定義します。
auto_sync	Boolean	このプロバイダーの ネットワーク が自動的に同期されるかどうかを示します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。

名前	タイプ	概要
external_plugin_type	文字列	ネットワークプラグインタイプ。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
password	文字列	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タイプ。
project_domain_name	文字列	OpenStack Identity API v3 のプロジェクトのドメイン名を定義します。
project_name	文字列	OpenStack Identity API v3 のプロジェクト名を定義します。
properties	Property[]	プロバイダー名/値のプロパティの配列。
read_only	Boolean	プロバイダーが読み取り専用かどうかを示します。
requires_authentication	Boolean	プロバイダー認証が必要かどうかを定義します。
tenant-name	文字列	OpenStack Identity API v2 のテナント名を定義します。
type	OpenStackNetworkProviderType	プロバイダーのタイプ。
unmanaged	Boolean	プロバイダーが Red Hat Virtualization によって管理されていないかどうかを示します。
url	文字列	外部プロバイダーの URL アドレスを定義します。
user_domain_name	文字列	OpenStack Identity API v3 の ExternalProvider で username のドメイン名を定義します。
username	文字列	認証プロセス中に使用されるユーザー名を定義します。

7.198.1. agent_configuration

非推奨エージェントの設定オプション。

Red Hat Virtualization 4.4.0 以降、OpenStack Neutron エージェントのデプロイメントが削除されたため、無視されます。

7.198.2. auto_sync

このプロバイダーの **ネットワーク** が自動的に同期されるかどうかを示します。

true の場合、このプロバイダーのネットワークはバックグラウンドで Red Hat Virtualization に自動的にかつ周期的に同期されます。これは、このプロバイダーのすべての新しいネットワークがインポートされ、すべての破棄されたネットワークが、この外部プロバイダーをデフォルトプロバイダーとして持つすべての **クラスター** から削除されることを意味します。プロバイダーでネットワークの名前が変更された場合、その変更は Red Hat Virtualization のネットワークエンティティに同期されます。さらに、プロバイダーをデフォルトのプロバイダーとして持つ新しいクラスターが追加された場合、既にインポートされたネットワークは、同期中にこの新しいクラスターにアタッチされます。

自動的に開始されたインポートにより、以下のステップがトリガーされます。

- 外部プロバイダーのネットワークは、その外部プロバイダーをデフォルトプロバイダーとして持つクラスターのデータセンター内のすべての **データセンター** にインポートされます。
- 関連するデータセンターとネットワークごとに **vNIC プロファイル** が作成されます。
- ネットワークは、デフォルトプロバイダーとしてその外部プロバイダーを持つ各クラスターに割り当てられます。

すべてのユーザーは、新しい vNIC プロファイルを使用できます。

後方互換性向けのデフォルトは **false** です。

7.198.3. external_plugin_type

ネットワークプラグインタイプ。

この属性を使用すると、外部 NIC が追加または変更されたときに、ホストで正しいプロバイダードライバーを選択できます。ドライバーの自動インストールがサポートされている場合 (**ovirt-provider-ovn** などの一部の定義済み実装でのみ使用可能)、この属性により、新しく追加されたホストにインストールするドライバー実装をシステムが決定できるようになります。

7.198.4. plugin_type

ネットワークプラグインタイプ。

Red Hat Virtualization Manager のバージョン 4.2 以降、この属性は非推奨になり、**external_plugin_type** が使用されるようになりました。この属性はタイプが **open_vswitch** のプロバイダーに対してのみ有効であり、**external_plugin_type** 属性の値が **open_vswitch** と等しい場合にのみ返されます。

更新中に **plugin_type** と **external_plugin_type** の両方が指定された場合、**plugin_type** の値は無視されます。

外部プロバイダーの場合、この値は表示されず、更新リクエスト中に無視されます。

7.198.5. 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。

7.198.6. 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。

7.198.7. tenant-name

OpenStack Identity API v2.0 のテナント名を定義します。

7.198.8. unmanaged

プロバイダーが Red Hat Virtualization によって管理されていないかどうかを示します。

true の場合、認証とサブネット制御は完全に外部プロバイダーに委ねられ、Red Hat Virtualization によって管理されません。

後方互換性向けのデフォルトは **false** です。

表7.264 リンクの概要

名前	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。

7.199. 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7.265 値の概要

名前	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。

7.199.1. 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。

7.199.2. 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。

7.200. OPENSTACKPROVIDER 構造体

表7.266 属性の概要

名前	タイプ	概要
authentication_url	文字列	外部プロバイダー認証 URL アドレスを定義します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
password	文字列	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	Boolean	プロバイダー認証が必要かどうかを定義します。
tenant-name	文字列	OpenStack Identity API v2 のテナント名を定義します。
url	文字列	外部プロバイダーの URL アドレスを定義します。
username	文字列	認証プロセス中に使用されるユーザー名を定義します。

7.200.1. 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。

7.200.2. tenant-name

OpenStack Identity API v2.0 のテナント名を定義します。

7.201. OPENSTACKSUBNET 構造体

表7.267 属性の概要

名前	タイプ	概要
cidr	文字列	ネットワーク CIDR を定義します。
comment	文字列	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	文字列	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	文字列	IP ゲートウェイを定義します。
id	文字列	一意の ID
ip_version	文字列	IP バージョンを定義します。
name	文字列	人間が判読できるプレーンテキストでの名前。

7.201.1. 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7.268 リンクの概要

名前	タイプ	概要
openstack_network	OpenStackNetwork	OpenStack ネットワークを管理するサービスへの参照。

7.202. OPENSTACKVOLUMEPROVIDER 構造体

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表7.269 属性の概要

名前	タイプ	概要
authentication_url	文字列	外部プロバイダー認証 URL アドレスを定義します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID

名前	タイプ	概要
name	文字列	人間が判読できるプレーンテキストでの名前。
password	文字列	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	Boolean	プロバイダー認証が必要かどうかを定義します。
tenant-name	文字列	OpenStack Identity API v2 のテナント名を定義します。
url	文字列	外部プロバイダーの URL アドレスを定義します。
username	文字列	認証プロセス中に使用されるユーザー名を定義します。

7.202.1. 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。

7.202.2. tenant-name

OpenStack Identity API v2.0 のテナント名を定義します。

表7.270 リンクの概要

名前	タイプ	概要
authentication_keys	OpenstackVolumeAuthenticationKey[]	
certificates	Certificate[]	
data_center	DataCenter	
volume_types	OpenStackVolumeType[]	

7.203. OPENSTACKVOLUMETYPE STRUCT

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表7.271 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
properties	Property[]	

表7.272 リンクの概要

名前	タイプ	概要
openstack_volume_provider	OpenStackVolume Provider	

7.204. OPENSTACKVOLUMEAUTHENTICATIONKEY STRUCT

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表7.273 属性の概要

名前	タイプ	概要
comment	文字列	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	Date	
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
usage_type	OpenstackVolume AuthenticationKey UsageType	
uuid	文字列	
value	文字列	

表7.274 リンクの概要

名前	タイプ	概要
<code>openstack_volume_provider</code>	OpenStackVolume Provider	

7.205. OPENSTACKVOLUMEAUTHENTICATIONKEYUSAGETYPE ENUM

Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられました。

表7.275 値の概要

名前	概要
<code>ceph</code>	

7.206. OPERATINGSYSTEM STRUCT

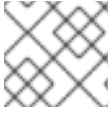
オペレーティングシステムを説明する情報。これは、仮想マシンとホストの両方に使用されます。

表7.276 属性の概要

名前	タイプ	概要
<code>boot</code>	Boot	ブートシーケンスの設定。
<code>cmdline</code>	文字列	Linux オペレーティングシステムを使用している場合に、仮想マシンを起動するためのカスタムカーネルパラメーター。
<code>custom_kernel_cmdline</code>	文字列	ホストカーネルコマンドラインのカスタム部分。
<code>initrd</code>	文字列	Linux オペレーティングシステムが使用されている場合は、ISO ストレージドメインのカスタム初期 RAM ディスクへのパス。
<code>kernel</code>	文字列	Linux オペレーティングシステムが使用されている場合は、ISO ストレージドメイン上のカスタムカーネルへのパス。
<code>reported_kernel_cmdline</code>	文字列	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。
<code>type</code>	文字列	人間が判読できる形式のオペレーティングシステム名。
<code>version</code>	バージョン	

7.206.1. boot

ブートシーケンスの設定。

**注記**

ホストには使用されません。

7.206.2. cmdline

Linux オペレーティングシステムを使用している場合に、仮想マシンを起動するためのカスタムカーネルパラメーター。

**注記**

ホストには使用されません。

7.206.3. 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。

7.206.4. initrd

Linux オペレーティングシステムが使用されている場合は、ISO ストレージドメインのカスタム初期 RAM ディスクへのパス。

例: **iso://initramfs-3.10.0-514.6.1.el7.x86_64.img**

**注記**

ホストには使用されません。

7.206.5. kernel

Linux オペレーティングシステムが使用されている場合は、ISO ストレージドメイン上のカスタムカーネルへのパス。

例: **iso://vmlinuz-3.10.0-514.6.1.el7.x86_64.**

**注記**

ホストには使用されません。

7.206.6. 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。

7.206.7. type

人間が判読できる形式のオペレーティングシステム名。

たとえば、**Fedora** または **RHEL** などです。通常、[オペレーティングシステム](#) サービスによって返される名前の1つになります。



注記

ホストの読み取り専用。

7.207. OPERATINGSYSTEMINFO STRUCT

ゲストオペレーティングシステムを表します。

表7.277 属性の概要

名前	タイプ	概要
architecture	アーキテクチャー	オペレーティングシステムのアーキテクチャー
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
large_icon	Icon	ゲストオペレーティングシステムの大きなアイコン。
name	文字列	人間が判読できるプレーンテキストでの名前。
small_icon	Icon	ゲストオペレーティングシステムの小さなアイコン。
tpm_support	TpmSupport	TPM サポートステータス。

7.207.1. large_icon

ゲストオペレーティングシステムの大きなアイコン。最大サイズ: 幅 150px、高さ 120px。

7.207.2. small_icon

ゲストオペレーティングシステムの小さなアイコン。最大サイズ: 幅 43px、高さ 43px。

7.208. OPTION 構造体

表7.278 属性の概要

名前	タイプ	概要
name	文字列	
type	文字列	
value	文字列	

7.209. OSTYPE ENUM

オペレーティングシステムの種類を表すタイプ。



警告

このタイプは、[OperatingSystemInfo](#) タイプの導入により非推奨になりました。オペレーティングシステムは、API の最上位コレクションとして利用できます: [operating_systems](#)。

エンドユーザーは、これらの値のいずれかを選択して、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 (ゲストオペレーティングシステム) のタイプを宣言します。この宣言により、システムは仮想マシンの設定を調整して、ユーザーエクスペリエンスを向上させることができます。たとえば、システムはオペレーティングシステムに最適なデバイスを選択します。システムはユーザーの選択に依存しており、インストールされた実際のゲストオペレーティングシステムを確認して検証しないことに注意してください。

表7.279 値の概要

名前	概要
other	他の値で指定されていない、他のタイプのオペレーティングシステム。
other_linux	他の値で指定されたもの以外の Linux ディストリビューション。
rhel_3	Red Hat Enterprise Linux 3 32 ビット
rhel_3x64	Red Hat Enterprise Linux 3 64 ビット
rhel_4	Red Hat Enterprise Linux 4 32 ビット

名前	概要
rhel_4x64	Red Hat Enterprise Linux 4 64 ビット
rhel_5	Red Hat Enterprise Linux 5 32 ビット
rhel_5x64	Red Hat Enterprise Linux 5 64 ビット
rhel_6	Red Hat Enterprise Linux 6 32 ビット
rhel_6x64	Red Hat Enterprise Linux 6 64 ビット
unassigned	この値は、 other にマッピングされる。
windows_2003	Windows 2003 32 ビット
windows_2003x64	Windows 2003 64 ビット
windows_2008	Windows 2008 32 ビット
windows_2008r2x64	Windows 2008 R2 64 ビット
windows_2008x64	Windows 2008 64 ビット
windows_2012x64	Windows 2012 64 ビット
windows_7	Windows 7 32 ビット
windows_7x64	Windows 7 64 ビット
windows_8	Windows 8 32 ビット
windows_8x64	Windows 8 64 ビット
windows_xp	Windows XP.

7.210. 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7.280 属性の概要

名前	タイプ	概要
name	文字列	パッケージの名前

7.211. PARALLELMIGRATIONS POLICY ENUM

並行移行の接続ポリシーを表すタイプ。

表7.281 値の概要

名前	概要
auto	並列接続と非並列接続を自動的に選択します。
auto_parallel	並列接続を使用し、それらの数を自動的に選択します。
custom	手動で指定した数の並列接続を使用します。
disabled	非並列接続を使用します。
inherit	クラスター値を使用します (仮想マシンにのみ適用可能)。

7.211.1. auto

並列接続と非並列接続を自動的に選択します。並列接続が使用されている場合、それらの数が自動的に選択されます。

7.211.2. custom

手動で指定した数の並列接続を使用します。並列接続の数は、`MigrationOptions.customParallelMigrations` で設定する必要があります。

7.212. PAYLOAD STRUCT

表7.282 属性の概要

名前	タイプ	概要
files	File[]	
type	VmDeviceType	

名前	タイプ	概要
volume_id	文字列	

7.213. PAYLOADENCODING ENUM

表7.283 値の概要

名前	概要
base64	
plaintext	

7.214. PERMISSION STRUCT

タイプはパーミッションを表します。

表7.284 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.285 リンクの概要

名前	タイプ	概要
cluster	クラスター	クラスターへの参照。
data_center	DataCenter	データセンターへの参照。
disk	ディスク	ディスクへの参照。
group	グループ	グループへの参照。
ホスト	ホスト	ホストへの参照。
role	ロール	ロールへの参照。

名前	タイプ	概要
storage_domain	StorageDomain	ストレージドメインへの参照。
template	Template	テンプレートへの参照。
user	User	ユーザーへの参照。
vm	Vm	仮想マシンへの参照。
vm_pool	VmPool	仮想マシンプールへの参照。

7.215. PERMIT 構造体

タイプは permit を表します。

表7.286 属性の概要

名前	タイプ	概要
administrative	Boolean	permit が管理上のものかどうかを指定します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.287 リンクの概要

名前	タイプ	概要
role	ロール	permit が属するロールへの参照。

7.216. PMPROXY 構造体

表7.288 属性の概要

名前	タイプ	概要
type	PmProxyType	

7.217. PMPROXYTYPE ENUM

表7.289 値の概要

名前	概要
cluster	フェンスプロキシーは、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	フェンスプロキシーは、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 す。
other_dc	フェンスプロキシーは、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。

7.218. 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7.290 値の概要

名前	概要
filter	
load_balancing	
weight	

7.219. PORTMIRRORING STRUCT

7.220. POWERMANAGEMENT STRUCT

表7.291 属性の概要

名前	タイプ	概要
address	文字列	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	Boolean	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	Boolean	電源管理設定が有効か無効かを示します。
kdump_detectio n	Boolean	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。

名前	タイプ	概要
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	文字列	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	文字列	フェンシングデバイスのコード。
username	文字列	電源管理用の有効なユーザー名。

7.220.1. 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。

7.220.2. 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。

7.220.3. 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。

7.220.4. 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。

7.221. POWERMANAGEMENTSTATUS ENUM

表7.292 値の概要

名前	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。

7.222. PRODUCT 構造体

表7.293 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

7.223. 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7.294 属性の概要

名前	タイプ	概要
instance_id	文字列	製品のこの特定のインストールの ID。
name	文字列	oVirt Engine などの製品名です。
vendor	文字列	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。

7.223.1. vendor

ベンダーの名前 (例: **ovirt.org**)。

7.224. PROFILEDETAIL STRUCT

表7.295 属性の概要

名前	タイプ	概要
block_statistics	BlockStatistic[]	
duration	Integer	
fop_statistics	FopStatistic[]	
profile_type	文字列	
statistics	Statistic[]	

7.225. PROPERTY STRUCT

表7.296 属性の概要

名前	タイプ	概要
name	文字列	
value	文字列	

7.226. PROXYTICKET STRUCT

表7.297 属性の概要

名前	タイプ	概要
value	文字列	

7.227. 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7.298 値の概要

名前	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。

7.227.1. 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。

7.228. 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7.299 属性の概要

名前	タイプ	概要
comment	文字列	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	Integer	最大処理能力 (単位: %)。
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
inbound_average	Integer	Mbps (メガビット/秒) 単位の望ましい平均インバウンドビットレート。
inbound_burst	Integer	1回のバーストで配信できるデータの量 (MB 単位)。
inbound_peak	Integer	Mbps (メガビット/秒) 単位の最大インバウンドレート。
max_iops	Integer	1秒あたりの最大許容入出力操作数。
max_read_iops	Integer	1秒あたりの最大許容入力操作数。
max_read_throughput	Integer	読み取り操作の最大許容スループット。
max_throughput	Integer	最大許容総スループット。
max_write_iops	Integer	1秒あたりの最大許容出力操作数。
max_write_throughput	Integer	書き込み操作の最大許容スループット。
name	文字列	人間が判読できるプレーンテキストでの名前。
outbound_average	Integer	Mbps (メガビット/秒) 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 (メガビット/秒) 単位のコミットされたレート。
outbound_average_upperlimit	Integer	Mbps (メガビット/秒) 単位のネットワークで使用される最大帯域幅。
outbound_burst	Integer	MB 単位の1回のバーストで送信できるデータの量。

名前	タイプ	概要
outbound_peak	Integer	Mbps (メガビット/秒) 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。

7.228.1. 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。

7.228.2. inbound_average

Mbps (メガビット/秒) 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。

7.228.3. inbound_burst

1回のバーストで配信できるデータの量 (MB 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。

7.228.4. inbound_peak

Mbps (メガビット/秒) 単位の最大インバウンドレート。

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、**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。

7.228.5. 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。

7.228.6. 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。

7.228.7. 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。

7.228.8. 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。

7.228.9. 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。

7.228.10. 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。

7.228.11. outbound_average

Mbps (メガビット/秒) 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。

7.228.12. 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。

7.228.13. outbound_average_realtime

Mbps (メガビット/秒) 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。

7.228.14. outbound_average_upperlimit

Mbps (メガビット/秒) 単位のネットワークで使用される最大帯域幅。

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。

7.228.15. outbound_burst

MB 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は、**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。

7.228.16. outbound_peak

Mbps (メガビット/秒) 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7.300 リンクの概要

名前	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。

7.229. 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7.301 値の概要

名前	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。

7.230. 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7.302 属性の概要

名前	タイプ	概要
cluster_hard_limit_pct	Integer	
cluster_soft_limit_pct	Integer	
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
disks	Disk[]	
id	文字列	一意の ID
name	文字列	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7.303 リンクの概要

名前	タイプ	概要
permissions	Permission[]	
quota_cluster_limits	QuotaClusterLimit[]	
quota_storage_limits	QuotaStorageLimit[]	

7.231. QUOTA_CLUSTER_LIMIT_STRUCT

表7.304 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
memory_limit	10 進数	
memory_usage	10 進数	
name	文字列	人間が判読できるプレーンテキストでの名前。
vcpu_limit	Integer	
vcpu_usage	Integer	

表7.305 リンクの概要

名前	タイプ	概要
cluster	クラスター	
quota	クォータ	

7.232. QUOTA_MODE_TYPE_ENUM

表7.306 値の概要

名前	概要
audit	
disabled	
enabled	

7.233. QUOTASTORAGELIMIT STRUCT

表7.307 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
limit	Integer	
name	文字列	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7.308 リンクの概要

名前	タイプ	概要
quota	クォータ	
storage_domain	StorageDomain	

7.234. RANGE 構造体

表7.309 属性の概要

名前	タイプ	概要
from	文字列	
to	文字列	

7.235. 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7.310 属性の概要

名前	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)。

7.236. REGISTRATIONAFFINITYGROUPMAPPING 構造体

このタイプは、オブジェクト登録の一部としてアフィニティーグループをマッピングする方法を説明します。オブジェクトは、仮想マシン、テンプレートなどにすることができます。

このマッピングを使用した XML 表現の例:

```
<action>
  <registration_configuration>
    <affinity_group_mappings>
      <registration_affinity_group_mapping>
        <from>
          <name>affinity</name>
        </from>
        <to>
          <name>affinity2</name>
        </to>
      </registration_affinity_group_mapping>
    </affinity_group_mappings>
  </registration_configuration>
</action>
```

表7.311 リンクの概要

名前	タイプ	概要
from	AffinityGroup	元のアフィニティーグループへの参照。
to	AffinityGroup	宛先アフィニティーグループへの参照。

7.236.1. from

元のアフィニティーグループへの参照。 **name** を使用して指定できます。

7.237. REGISTRATIONAFFINITYLABELMAPPING 構造体

このタイプは、オブジェクト登録の一部としてアフィニティーラベルをマッピングする方法を説明します。オブジェクトは、仮想マシン、テンプレートなどにすることができます。

マッピングを使用した XML 表現の例:

```
<action>
  <registration_configuration>
    <affinity_label_mappings>
      <registration_affinity_label_mapping>
        <from>
          <name>affinity_label</name>
        </from>
        <to>
          <name>affinity_label2</name>
        </to>
      </registration_affinity_label_mapping>
    </affinity_label_mappings>
  </registration_configuration>
</action>
```

表7.312 リンクの概要

名前	タイプ	概要
from	AffinityLabel	元のアフィニティーラベルへの参照。
to	AffinityLabel	宛先アフィニティーラベルへの参照。

7.237.1. from

元のアフィニティーラベルへの参照。**name** を使用して指定できます。

7.238. REGISTRATIONCLUSTERMAPPING 構造体

このタイプは、オブジェクト登録の一部としてクラスターをマッピングする方法を説明します。オブジェクトは、仮想マシン、テンプレートなどにすることができます。

このマッピングを使用した XML 表現の例:

```
<action>
  <registration_configuration>
    <cluster_mappings>
      <registration_cluster_mapping>
        <from>
          <name>myoriginalcluster</name>
        </from>
        <to>
          <name>mynewcluster</name>
        </to>
      </registration_cluster_mapping>
    </cluster_mappings>
  </registration_configuration>
</action>
```

表7.313 リンクの概要

名前	タイプ	概要
from	クラスター	元のクラスターへの参照。
to	クラスター	宛先クラスターへの参照。

7.238.1. from

元のクラスターへの参照。**id** または **name** を使用して指定できます。

7.238.2. to

宛先クラスターへの参照。**id** または **name** を使用して指定できます。

7.239. REGISTRATIONCONFIGURATION 構造体

このタイプは、オブジェクト (仮想マシン、テンプレートなど) の登録方法を記述し、障害復旧ソリューションの実装に使用されます。

このタイプに含まれる各マッピングを使用して、元のシステムのオブジェクトを、仮想マシンまたはテンプレートが登録されているシステムの対応するオブジェクトにマップできます。たとえば、クラスター A で設定された仮想マシンを使用したプライマリーセットアップと、クラスター B を使用したアクティブなセカンダリーセットアップが存在するとします。クラスター B はその仮想マシンと互換性があり、障害復旧シナリオにおいてストレージドメインをセカンダリーセットアップにインポートし、ユーザーが仮想マシンをクラスター B に登録できます。

その場合、クラスターマッピングを定義することで、回復プロセスを自動化できます。エンティティーが登録されると、その OVF はそれがクラスター A に属していることを示しますが、マッピングはクラスター A がクラスター B に置き換えられることを示します。Red Hat Virtualization Manager は切り替えを行い、仮想マシンをセカンダリーサイトのクラスター B に登録する必要があります。

クラスターマッピングは一例に過ぎず、さまざまな種類のマッピングがあります。

- クラスターマッピング。
- LUN マッピング。
- ロールマッピング
- ドメインマッピング
- パーMISSIONのマッピング。
- アフィニティーグループのマッピング。
- アフィニティーラベルのマッピング。
- 仮想 NIC プロファイルのマッピング。

Red Hat Virtualization Manager で登録操作が行われると、各マッピングは特定の OVF のデータに使用されます。

マッピングを使用した XML 表現の例:

■

```
<action>
  <registration_configuration>
    <cluster_mappings>
      <registration_cluster_mapping>
        <from>
          <name>myoriginalcluster</name>
        </from>
        <to>
          <name>mynewcluster</name>
        </to>
      </registration_cluster_mapping>
    </cluster_mappings>
    <role_mappings>
      <registration_role_mapping>
        <from>
          <name>SuperUser</name>
        </from>
        <to>
          <name>UserVmRunTimeManager</name>
        </to>
      </registration_role_mapping>
    </role_mappings>
    <domain_mappings>
      <registration_domain_mapping>
        <from>
          <name>redhat</name>
        </from>
        <to>
          <name>internal</name>
        </to>
      </registration_domain_mapping>
    </domain_mappings>
    <lun_mappings>
      <registration_lun_mapping>
        <from id="111">
        </from>
        <to id="222">
          <alias>weTestLun</alias>
          <lun_storage>
            <type>iscsi</type>
            <logical_units>
              <logical_unit id="36001405fb1ddb4b91e44078f1ffcfef">
                <address>44.33.11.22</address>
                <port>3260</port>
                <portal>1</portal>
                <target>iqn.2017-11.com.name.redhat:444</target>
              </logical_unit>
            </logical_units>
          </lun_storage>
        </to>
      </registration_lun_mapping>
    </lun_mappings>
    <affinity_group_mappings>
      <registration_affinity_group_mapping>
        <from>
          <name>affinity</name>
```

```

</from>
<to>
  <name>affinity2</name>
</to>
</registration_affinity_group_mapping>
</affinity_group_mappings>
<affinity_label_mappings>
<registration_affinity_label_mapping>
  <from>
    <name>affinity_label</name>
  </from>
  <to>
    <name>affinity_label2</name>
  </to>
</registration_affinity_label_mapping>
</affinity_label_mappings>
<vnic_profile_mappings>
<registration_vnic_profile_mapping>
  <from>
    <name>gold</name>
    <network>
      <name>red</name>
    </network>
  </from>
  <to id="738dd914-8ec8-4a8b-8628-34672a5d449b"/>
</registration_vnic_profile_mapping>
<registration_vnic_profile_mapping>
  <from>
    <name>silver</name>
    <network>
      <name>blue</name>
    </network>
  </from>
  <to>
    <name>copper</name>
    <network>
      <name>orange</name>
    </network>
  </to>
</registration_vnic_profile_mapping>
</vnic_profile_mappings>
</registration_configuration>
</action>

```

表7.314 属性の概要

名前	タイプ	概要
affinity_group_mappings	RegistrationAffinityGroupMapping[]	アフィニティーグループのマッピング方法について説明します。
affinity_label_mappings	RegistrationAffinityLabelMapping[]	アフィニティーラベルのマッピング方法について説明します。

名前	タイプ	概要
cluster_mappings	RegistrationClusterMapping[]	オブジェクトが参照するクラスターのマッピング方法について説明します。
domain_mappings	RegistrationDomainMapping[]	ユーザーのドメインのマッピング方法について説明します。
lun_mappings	RegistrationLunMapping[]	LUN のマッピング方法について説明します。
role_mappings	RegistrationRoleMapping[]	ロールのマッピング方法について説明します。
vnic_profile_mappings	RegistrationVnicProfileMapping[]	登録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。

7.240. REGISTRATIONDOMAINMAPPING 構造体

このタイプは、オブジェクト登録の一部としてユーザーのドメインをマップする方法を説明します。オブジェクトは、仮想マシン、テンプレートなどにすることができます。注: これは、ユーザー名が同じであり、ドメイン名のみが変更されるという前提に基づいています。

このマッピングを使用した XML 表現の例:

```
<action>
  <registration_configuration>
    <domain_mappings>
      <registration_domain_mapping>
        <from>
          <name>redhat</name>
        </from>
        <to>
          <name>internal</name>
        </to>
      </registration_domain_mapping>
    </domain_mappings>
  </registration_configuration>
</action>
```

表7.315 リンクの概要

名前	タイプ	概要
from	Domain	元のドメインへの参照。
to	Domain	宛先ドメインへの参照。

7.240.1. from

元のドメインへの参照。**name** を使用して指定できます。

7.241. REGISTRATIONLUNMAPPING 構造体

このタイプは、オブジェクト登録の一部として LUN をマップする方法を示します。オブジェクトは、仮想マシン、テンプレートなどにすることができます。

外部 LUN ディスクは、ストレージドメインに存在しないエンティティです。オブジェクトが登録されている環境に存在する必要がないため、指定する必要があります。このマッピングを使用した XML 表現の例:

```
<action>
  <registration_configuration>
    <lun_mappings>
      <registration_lun_mapping>
    <lun_mappings>
      <registration_lun_mapping>
        <from id="111">
        </from>
        <to id="222">
          <alias>weTestLun</alias>
          <lun_storage>
            <type>iscsi</type>
            <logical_units>
              <logical_unit id="36001405fb1ddb4b91e44078f1ffcfef">
                <address>44.33.11.22</address>
                <port>3260</port>
                <portal>1</portal>
                <target>iqn.2017-11.com.name.redhat:444</target>
              </logical_unit>
            </logical_units>
          </lun_storage>
        </to>
      </registration_lun_mapping>
    </lun_mappings>
  </registration_configuration>
</action>
```

表7.316 リンクの概要

名前	タイプ	概要
from	ディスク	元の LUN への参照。
to	ディスク	仮想マシンに追加される LUN への参照。

7.241.1. from

元の LUN への参照。これは、**id** 属性で指定する必要があります。

7.242. REGISTRATIONROLEMAPPING 構造体

このタイプは、オブジェクト登録の一部としてロールをマップする方法を記述します。オブジェクトは、仮想マシン、テンプレートなどにすることができます。

ロールマッピングは、プライマリーサイトとセカンダリーサイトの間の相関するロールをマップすることを目的としています。たとえば、登録されている仮想マシンに対して、ロール **UserVmRunTimeManager** を持つパーミッションがある場合があります。したがって、**UserVmRunTimeManager** の代わりに **SuperUser** ロールを使用して、仮想マシンをセカンダリーセットアップに登録するマッピングを送信することができます。このマッピングを使用した XML 表現の例:

```
<action>
  <registration_configuration>
    <role_mappings>
      <registration_eole_mapping>
        <from>
          <name>SuperUser</name>
        </from>
        <to>
          <name>UserVmRunTimeManager</name>
        </to>
      </registration_role_mapping>
    </role_mappings>
  </registration_configuration>
</action>
```

表7.317 リンクの概要

名前	タイプ	概要
from	ロール	元のロールへの参照。
to	ロール	宛先ロールへの参照。

7.242.1. from

元のロールへの参照。 **name** を使用して指定できます。

7.243. REGISTRATIONVNICPROFILEMAPPING STRUCT

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。ターゲットは、プロファイル ID、またはプロファイル名とネットワーク名のペアとして指定できます。

たとえば、目的の仮想 NIC プロファイルマッピングに次の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\名
red	gold	738dd914-8ec8-4a8b-8628-34672a5d449b

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\名
<empty> (ネットワーク名なし)	<empty> (ネットワークプロファイル名なし)	892a12ec-2028-4451-80aa-ff3bf55d6bac
blue	silver	orange\copper
yellow	platinum	<empty> (プロファイルなし)
green	bronze	

次に、次のスニペットを `RegistrationConfiguration` に追加する必要があります

```
<vnic_profile_mappings>
  <registration_vnic_profile_mapping>
    <from>
      <name>gold</name>
      <network>
        <name>red</name>
      </network>
    </from>
    <to id="738dd914-8ec8-4a8b-8628-34672a5d449b"/>
  </registration_vnic_profile_mapping>
  <registration_vnic_profile_mapping>
    <from>
      <name></name>
      <network>
        <name></name>
      </network>
    </from>
    <to id="892a12ec-2028-4451-80aa-ff3bf55d6bac"/>
  </registration_vnic_profile_mapping>
  <registration_vnic_profile_mapping>
    <from>
      <name>silver</name>
      <network>
        <name>blue</name>
      </network>
    </from>
    <to>
      <name>copper</name>
      <network>
        <name>orange</name>
      </network>
    </to>
  </registration_vnic_profile_mapping>
  <registration_vnic_profile_mapping>
    <from>
      <name>platinum</name>
      <network>
        <name>yellow</name>
      </network>
    </from>
  </registration_vnic_profile_mapping>
</vnic_profile_mappings>
```

```

    </network>
  </from>
  <to>
    <name></name>
    <network>
      <name></name>
    </network>
  </to>
</registration_vnic_profile_mapping>
<registration_vnic_profile_mapping>
  <from>
    <name>bronze</name>
    <network>
      <name>green</name>
    </network>
  </from>
</registration_vnic_profile_mapping>
</vnic_profile_mappings>

```

表7.318 リンクの概要

名前	タイプ	概要
from	VnicProfile	外部ネットワークおよび外部ネットワークプロファイルへの参照。
to	VnicProfile	既存の仮想 NIC プロファイルへの参照。

7.243.1. from

外部ネットワークおよび外部ネットワークプロファイルへの参照。どちらも **name** を使用して指定する必要があります。

7.243.2. to

既存の仮想 NIC プロファイルへの参照。**name** または **id** で指定する必要があります。**name** または **id** の両方ではなく、どちらか一方を指定します。

7.244. REPORTEDCONFIGURATION 構造体

表7.319 属性の概要

名前	タイプ	概要
actual_value	文字列	
expected_value	文字列	
in_sync	Boolean	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 。

名前	タイプ	概要
name	文字列	

7.245. REPORTEDDEVICE 構造体

表7.320 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
ips	Ip[]	
mac	Mac	
name	文字列	人間が判読できるプレーンテキストでの名前。
type	ReportedDeviceType	

表7.321 リンクの概要

名前	タイプ	概要
vm	Vm	

7.246. REPORTEDDEVICETYPE ENUM

表7.322 値の概要

名前	概要
network	

7.247. RESOLUTIONTYPE ENUM

表7.323 値の概要

名前	概要
add	
copy	

7.248. RNGDEVICE 構造体

乱数ジェネレーター (RNG) デバイスモデル。

表7.324 属性の概要

名前	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。

7.249. 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7.325 値の概要

名前	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。

7.249.1. 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 **rbdm** 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。

7.250. ROLE 構造体

システムロールを表します。

表7.326 属性の概要

名前	タイプ	概要
administrative	Boolean	ロールを管理専用にするかどうかを定義します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
mutable	Boolean	ロールを更新または削除する機能を定義します。
name	文字列	人間が判読できるプレーンテキストでの名前。

7.250.1. 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7.327 リンクの概要

名前	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	

7.251. 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7.328 値の概要

名前	概要
admin	管理ロール。
user	ユーザーロール。

7.252. SCHEDULINGPOLICY STRUCT

表7.329 属性の概要

名前	タイプ	概要
comment	文字列	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	Boolean	
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
locked	Boolean	
name	文字列	人間が判読できるプレーンテキストでの名前。
properties	Property[]	

表7.330 リンクの概要

名前	タイプ	概要
balances	Balance[]	
filters	Filter[]	
weight	Weight[]	

7.253. SCHEDULINGPOLICYUNIT STRUCT

表7.331 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
enabled	Boolean	
id	文字列	一意の ID
internal	Boolean	
name	文字列	人間が判読できるプレーンテキストでの名前。
properties	Property[]	

名前	タイプ	概要
type	PolicyUnitType	

7.254. SCSIGENERICIO ENUM

ダイレクト LUN ディスクが SCSI パススルーを使用している場合、権限 I/O ポリシーはこの enum によって決定されます。

表7.332 値の概要

名前	概要
disabled	SCSI パススルーを無効にします。
filtered	特権 SCSI I/O を拒否します。
unfiltered	特権 SCSI I/O を許可します。

7.255. SELINUX 構造体

システム内の SELinux を表します。

表7.333 属性の概要

名前	タイプ	概要
mode	SeLinuxMode	SELinux の現在のモード。

7.256. SELINUXMODE ENUM

SELinux 強制モードを表します。

表7.334 値の概要

名前	概要
disabled	SELinux はカーネルで無効になっています。
enforcing	SELinux が実行され、パーミッションを強制しています。
permissive	SELinux が実行され、ログを記録していますが、パーミッションが強制されていません。

7.257. SERIALNUMBER 構造体

表7.335 属性の概要

名前	タイプ	概要
policy	SerialNumberPolicy	
value	文字列	

7.258. SERIALNUMBERPOLICY ENUM

シリアル番号のポリシーを表すタイプ。

表7.336 値の概要

名前	概要
custom	このポリシーにより、ユーザーはシリアル番号として任意の文字列を指定できます。
ホスト	このポリシーはレガシーポリシーです。
none	このポリシーは、シリアル番号ポリシーを削除し、デフォルトの null に移動するために使用されます。
vm	このポリシーでは、仮想マシン ID をシリアル番号として使用します。

7.258.1. ホスト

このポリシーはレガシーポリシーです。シリアル番号としてホスト ID を使用します。

7.259. 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7.337 属性の概要

名前	タイプ	概要
comment	文字列	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	Boolean	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
ip	Ip	ユーザーの接続元 IP アドレス。
name	文字列	人間が判読できるプレーンテキストでの名前。

名前	タイプ	概要
protocol	文字列	セッションで使用されるプロトコル。

7.259.1. 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。

7.259.2. 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。

7.259.3. 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7.338 リンクの概要

名前	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。

7.259.4. 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。

7.260. SKIPIFCONNECTIVITYBROKEN STRUCT

表7.339 属性の概要

名前	タイプ	概要
enabled	Boolean	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。

名前	タイプ	概要
threshold	Integer	接続テストのしきい値。

7.260.1. 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。

7.260.2. 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。

7.261. 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7.340 属性の概要

名前	タイプ	概要
enabled	Boolean	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。

7.261.1. 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。

7.262. 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7.341 属性の概要

名前	タイプ	概要
auto_pinning_policy	AutoPinningPolicy	自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。
bios	Bios	仮想マシンの BIOS 設定への参照。
comment	文字列	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	Console	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_pinning_policy	CpuPinningPolicy	CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。
CPU Shares	Integer	
creation_time	Date	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	文字列	
Custom Emulated Machine	文字列	
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	Date	このスナップショットが作成された日付。
delete_protected	Boolean	true の場合、仮想マシンは削除できません。
description	文字列	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	文字列	仮想マシンの完全修飾ドメイン名。

名前	タイプ	概要
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)。
has_illegal_images	Boolean	仮想マシンに、ディスクが ILLEGAL 状態のスナップショットがあるかどうかを示します。
high_availability	高可用性	仮想マシンの高可用性設定。
id	文字列	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	Icon	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)。
multi_queues_enabled	Boolean	true の場合、各仮想インターフェイスは、利用可能な仮想 CPU に応じて最適な数のキューを取得します。
name	文字列	人間が判読できるプレーンテキストでの名前。
next_run_configuration_exists	Boolean	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_mode	NumaTuneMode	NUMA トポロジーの適用方法。
origin	文字列	この仮想マシンのオリジン。

名前	タイプ	概要
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	Boolean	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	Boolean	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	Icon	仮想マシンの小さなアイコン。
snapshot_statuses	SnapshotStatus	スナップショットのステータス。
snapshot_type	SnapshotType	スナップショットのタイプ。
Soundcard Enabled	Boolean	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	Boolean	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	Date	仮想マシンが起動された日付。
stateless	Boolean	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	文字列	人間が読める現在のステータスの詳細。
stop_reason	文字列	仮想マシンが停止した理由。

名前	タイプ	概要
stop_time	Date	仮想マシンが停止された日付。
storage_error_resume_behaviour	VmStorageErrorResumeBehaviour	ストレージエラー後に仮想マシンを再開する方法を決定します。
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。
tpm_enabled	Boolean	true の場合、TPM デバイスが仮想マシンに追加されます。
tunnel_migration	Boolean	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	Boolean	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。
virtio_scsi_multi_queues	Integer	このフィールドの Virtio-SCSI controller のキュー数には <code>virtioScsiMultiQueuesEnabled</code> が true である必要があります。詳細は <code>virtioScsiMultiQueuesEnabled</code> を参照してください。
virtio_scsi_multi_queues_enabled	Boolean	true の場合、Virtio-SCSI デバイスは、使用可能な仮想 CPU とディスク、または指定された <code>virtioScsiMultiQueues</code> に応じて、いくつかの複数キューを取得します。

7.262.1. auto_pinning_policy

自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。



重要

エンジンのバージョン 4.5 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後、削除される可能性があります。代わりに `CpuPinningPolicy` を使用してください。

7.262.2. 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>
```

7.262.3. cpu_pinning_policy

CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。指定されていない場合、CPU ピニング文字列の以前の動作により、CpuPinningPolicy が None または Manual に決定されます。

7.262.4. 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。

7.262.5. 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。

7.262.6. initialization

仮想マシンの初期化設定への参照。



注記

Red Hat Virtualization 4.1.8 以降、このプロパティは空のタグを送信することでクリアできます。

たとえば、**initialization** 属性をクリアするには、次のようなリクエストを送ります。

```
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>
  <initialization/>
</vm>
```

このようなリクエストへのレスポンス、およびヘッダー **All-Content: true** を持つリクエストには、引き続きこの属性が含まれます。

7.262.7. 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。

7.262.8. 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。

7.262.9. 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>
```

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。上記の例を使用して、仮想マシンが **up** 状態のときにメモリーを増やすことができます。サイズの増分は、**HotPlugMemoryBlockSizeMb** 設定値 (デフォルトでは 256 MiB) の値で割り切れる必要があります。メモリーサイズの増分がこの値で割り切れない場合、メモリーサイズの変更は次の実行設定にのみ保存されます。メモリーのホットプラグ操作が成功するたびに、1つまたは2つの新しいメモリーデバイスが作成されます。

メモリーのホットアンプラグは、Red Hat Virtualization 4.2 以降でサポートされています。メモリーのホットアンプラグは、仮想マシンの状態が **up** の場合にのみ実行できます。ホットアンプラグ操作で取り外すことができるのは、以前にホットプラグされたメモリーデバイスのみです。要求されたメモリーの減少分は、以前にホットプラグされたメモリーデバイスの組み合わせのサイズに一致するように切り捨てられます。要求されたメモリー値は、丸められずに次の実行設定に格納されます。

**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**

Red Hat Virtualization Manager は内部的に値を切り捨てて整数の MiB ($1 \text{ MiB} = 2^{20} \text{ バイト}$) にします。

7.262.10. migration

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。

**注記**

このメソッドが返す ID で移行ポリシーを照会する API はまだ実装されていません。`/ovirt-engine/api/options/MigrationPolicies` を使用して、すべての移行ポリシーとその ID のリストを取得します。

7.262.11. 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。

7.262.12. 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。

7.262.13. numa_tune_mode

NUMA トポロジーの適用方法。非推奨となりました。vNUMA ノードごとの NUMA 調整が使用されません。

7.262.14. 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`

7.262.15. `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。

7.262.16. `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>
```

7.262.17. `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。

7.262.18. 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。

7.262.19. 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。

7.262.20. tpm_enabled

true の場合、TPM デバイスが仮想マシンに追加されます。デフォルト値は **false** です。このプロパティは、"All-Content=true" ヘッダーが設定されている場合に、フェッチするときのみ表示されません。

表7.342 リンクの概要

名前	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	クラスター	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。
disks	Disk[]	スナップショットにリンクされているディスクのリスト。
dynamic_cpu	DynamicCpu	仮想マシン CPU の動的設定。
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。
ホスト	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。

名前	タイプ	概要
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。
mediated_devices	VmMediatedDevice[]	仲介デバイスの設定。
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[]	
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	このスナップショットが取得された仮想マシン。
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。

7.262.21. 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。

7.262.22. 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>
```

7.262.23. 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。

7.262.24. statistics

この仮想マシンから収集された統計データ。

一部の統計、特に **memory.buffered** および **memory.cached** は、[Red Hat Virtualization](#) **ゲストエージェント** が仮想マシンにインストールされている場合にのみ使用できることに注意してください。

7.263. SNAPSHOTSTATUS ENUM

スナップショットの現在のステータスを表します。

表7.343 値の概要

名前	概要
in_preview	スナップショットをプレビューしています。
locked	スナップショットはロックされています。
ok	スナップショットに問題はありません。

7.263.1. locked

スナップショットはロックされています。

スナップショットは、作成、削除、復元、またはプレビュー中にロックされます。

7.264. SNAPSHOTTYPE ENUM

スナップショットのタイプを表します。

表7.344 値の概要

名前	概要
active	仮想マシンの現在の設定への参照。
preview	プレビュー中のスナップショットがある場合、 active スナップショットは preview になります。
regular	ユーザーが作成したスナップショット。
stateless	ステートレス仮想マシン用に内部で作成されたスナップショット。

7.264.1. preview

プレビュー中のスナップショットがある場合、**active** スナップショットは **preview** になります。

つまり、これは、プレビュー前の **active** スナップショットです。

7.264.2. stateless

ステートレス仮想マシン用に内部で作成されたスナップショット。

このスナップショットは、仮想マシンの起動時に作成され、仮想マシンのシャットダウン時に復元されます。

7.265. 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7.345 リンクの概要

名前	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。

7.266. SPM 構造体

表7.346 属性の概要

名前	タイプ	概要
priority	Integer	
status	SpmStatus	

7.267. SPMSTATUS ENUM

表7.347 値の概要

名前	概要
contending	
none	
spm	

7.268. SSH 構造体

表7.348 属性の概要

名前	タイプ	概要
authentication_method	SshAuthentication Method	
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
フィンガープリント (fingerprint)	文字列	ホストの SSH 公開鍵のフィンガープリント。

名前	タイプ	概要
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
port	Integer	
public_key	文字列	link:https://tools で定義されている SSH 公開鍵フォーマットを使用したホストの SSH 公開鍵。
user	User	

7.268.1. フィンガープリント (fingerprint)

ホストの SSH 公開鍵のフィンガープリント。このフィールドは 4.4.5 以降で非推奨となり、今後削除される予定です。

代わりに `publicKey` を使用してください。

7.268.2. public_key

[RFC4253](#) で定義されている SSH 公開鍵フォーマットを使用した、ホストの SSH 公開鍵。

7.269. SSHAUTHENTICATIONMETHOD ENUM

表7.349 値の概要

名前	概要
password	
publickey	

7.270. SSHPUBLICKEY STRUCT

表7.350 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	保存された SSH キーが含まれています。
description	文字列	プレーンテキストでの人間が判読できる説明。

名前	タイプ	概要
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.351 リンクの概要

名前	タイプ	概要
user	User	

7.271. SSO 構造体

表7.352 属性の概要

名前	タイプ	概要
メソッド	Method[]	

7.272. SSOMETHOD ENUM

表7.353 値の概要

名前	概要
guest_agent	

7.273. 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。

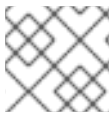
統計には、さまざまなエンティティの統計値が含まれています。次のオブジェクトには統計が含まれています:

- 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>
    <host_nic 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7.354 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
kind	StatisticKind	統計測定値のタイプ。
name	文字列	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7.355 リンクの概要

名前	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	
ホスト	ホスト	
host_nic	HostNic	ホスト NIC への参照。
host_numa_node	NumaNode	
nic	Nic	
step	Step	
vm	Vm	

7.274. STATISTICKIND ENUM

表7.356 値の概要

名前	概要
カウンター	
ゲージ	

7.275. STATISTICUNIT ENUM

表7.357 値の概要

名前	概要
bits_per_second	
bytes	
bytes_per_second	

名前	概要
<code>count_per_second</code>	
<code>none</code>	
<code>percent</code>	
秒	

7.276. 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7.358 属性の概要

名前	タイプ	概要
<code>comment</code>	文字列	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>	文字列	プレーンテキストでの人間が判読できる説明。
<code>end_time</code>	Date	ステップの終了時刻。
<code>external</code>	Boolean	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。
<code>external_type</code>	ExternalSystemType	ステップが参照する外部システム。
<code>id</code>	文字列	一意の ID
<code>name</code>	文字列	人間が判読できるプレーンテキストでの名前。
<code>number</code>	Integer	現在の階層レベルでのステップの順序。
<code>progress</code>	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。
<code>start_time</code>	Date	ステップの開始時間。
<code>status</code>	StepStatus	ステップのステータス。
<code>type</code>	StepEnum	ステップのタイプ。

7.276.1. 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7.359 リンクの概要

名前	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[]	

7.277. STEPENUM ENUM

ステップタイプを表すタイプ。

表7.360 値の概要

名前	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。

7.277.1. 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。

7.277.2. 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。

7.277.3. 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。

7.277.4. 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。

7.277.5. 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。

7.277.6. 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。

7.278. STEPSTATUS ENUM

ステップのステータスを表します。

表7.361 値の概要

名前	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。

7.278.1. 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。

7.278.2. 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。

7.278.3. 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。

7.278.4. 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。

7.279. 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7.362 属性の概要

名前	タイプ	概要
address	文字列	ストレージサーバー接続のアドレス。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
mount_options	文字列	NFS ストレージサーバー接続のマウントオプション。
name	文字列	人間が判読できるプレーンテキストでの名前。
nfs_retrans	Integer	NFS ストレージサーバー接続の NFS retrans 値。
nfs_timeo	Integer	NFS ストレージサーバー接続の NFS timeo 値。
nfs_version	NfsVersion	NFS ストレージサーバー接続の NFS バージョン。
password	文字列	iSCSI ストレージサーバー接続のパスワード。
path	文字列	NFS ストレージサーバー接続のパス。
port	Integer	iSCSI ストレージサーバー接続のポート。
portal	文字列	iSCSI ストレージサーバー接続のポータル。
target	文字列	iSCSI ストレージサーバー接続のターゲット。
type	StorageType	ストレージサーバー接続のタイプ。

名前	タイプ	概要
username	文字列	iSCSI ストレージサーバー接続のユーザー名。
vfs_type	文字列	NFS ストレージサーバー接続の VFS タイプ。

表7.363 リンクの概要

名前	タイプ	概要
gluster_volume	GlusterVolume	そのストレージドメインで使用される gluster ボリュームへのリンク。
ホスト	ホスト	

7.280. STORAGECONNECTIONEXTENSION 構造体

表7.364 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
password	文字列	
target	文字列	
username	文字列	

表7.365 リンクの概要

名前	タイプ	概要
ホスト	ホスト	

7.281. 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7.366 属性の概要

名前	タイプ	概要
available	Integer	
backup	Boolean	この属性は、データストレージドメインがバックアップドメインとして使用されているかどうかを示します。
block_size	Integer	ストレージドメインのブロックサイズをバイト単位で指定します。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
discard_after_delete	Boolean	ブロック <code>storage domain</code> 上の <code>disk</code> 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	文字列	一意の ID

名前	タイプ	概要
import	Boolean	
マスター	Boolean	
name	文字列	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	Boolean	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	Boolean	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	
warning_low_space_indicator	Integer	
wipe_after_delete	Boolean	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。

7.281.1. backup

この属性は、データストレージドメインがバックアップドメインとして使用されているかどうかを示します。ドメインがバックアップに設定されている場合、エクスポートストレージドメインを使用するのと同じ方法で、障害復旧の目的で仮想マシンとテンプレートを保存するために使用されます。この属性は、データストレージドメインでのみ使用でき、ISO ドメインまたはエクスポートストレージドメインでは使用できません。ユーザーは、データストレージドメインの作成中またはデータストレージドメインのインポート中に、この機能を使用できます。

7.281.2. block_size

ストレージドメインのブロックサイズをバイト単位で指定します。省略可能で、その場合はデフォルトで 512 バイトになります。すべてのストレージドメインが、使用可能なすべてのサイズをサポートしているわけではありません。

7.281.3. 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。

7.281.4. 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。

7.281.5. 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。



重要

システムのバージョン 4.2.1 以降、カーネルの `sysfs` ファイル `discard_zeroes_data` が非推奨になったため、この属性のサポートが削除されました。後方互換性のために保持されていますが、値は常に **false** になります。

7.281.6. 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7.367 リンクの概要

名前	タイプ	概要
<code>data_center</code>	<code>DataCenter</code>	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。

名前	タイプ	概要
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[]	
ホスト	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[]	

7.281.7. 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。

7.282. 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7.368 リンクの概要

名前	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。

7.283. STORAGEDOMAINSTATUS ENUM

表7.369 値の概要

名前	概要
activating	
active	
detaching	
inactive	
locked	
maintenance	
mixed	
preparing_for_maintenance	
unattached	
unknown	

7.284. STORAGEDOMAINTYPE ENUM

[ストレージドメイン](#) が管理するデータの種別を示します。

表7.370 値の概要

名前	概要
data	データドメインは、システム内の仮想マシンとテンプレートのディスクとスナップショットを格納するために使用されます。
export	エクスポートドメインは、データセンターと Red Hat Virtualization 環境の間で、仮想マシンとテンプレートをコピーおよび移動するために使用される一時的なストレージリポジトリです。
image	外部システムからインポートできるイメージドメインストアイメージ。
iso	ISO ドメインは、仮想マシンのオペレーティングシステムとアプリケーションのインストールと起動に使用される ISO ファイル (または論理 CD) を格納します。
managed_block_storage	管理対象ブロックストレージドメインは、ブロックストレージデバイス上に作成されます。

名前	概要
volume	ボリュームドメインは、仮想マシンのディスクとして使用できる論理ボリュームを格納します。

7.284.1. data

データドメインは、システム内の仮想マシンとテンプレートのディスクとスナップショットを格納するために使用されます。さらに、ディスクのスナップショットもデータドメインに保存されます。データドメインは、データセンター間で共有することはできません。

7.284.2. export

エクスポートドメインは、データセンターと Red Hat Virtualization 環境の間で、仮想マシンとテンプレートをコピーおよび移動するために使用される一時的なストレージリポジトリです。エクスポートドメインは、仮想マシンのバックアップにも使用できます。エクスポートドメインは、データセンター間で移動することができますが、同時に1つのデータセンターでしか有効にすることができません。

7.284.3. image

外部システムからインポートできるイメージドメインストアイメージ。たとえば、OpenStack Glance イメージリポジトリからのイメージ。

7.284.4. iso

ISO ドメインは、仮想マシンのオペレーティングシステムとアプリケーションのインストールと起動に使用される ISO ファイル (または論理 CD) を格納します。ISO ドメインにより、データセンターの物理メディアの必要性がなくなります。ISO ドメインは、異なるデータセンターで共有することができます。

7.284.5. managed_block_storage

管理対象ブロックストレージドメインは、ブロックストレージデバイス上に作成されます。これらのドメインは、cinder によってアクセスおよび管理されます。

7.284.6. volume

ボリュームドメインは、仮想マシンのディスクとして使用できる論理ボリュームを格納します。たとえば、OpenStack Cinder ブロックストレージサービスのボリュームです。

7.285. STORAGEFORMAT ENUM

[ストレージドメイン](#) のフォーマットを表すタイプ。

表7.371 値の概要

名前	概要
----	----

名前	概要
v1	ストレージドメインフォーマットのバージョン1は、NFS、iSCSI、およびFCストレージドメインに適用できます。
v2	ストレージドメインフォーマットのバージョン2は、iSCSI および FC ストレージドメインに適用できます。
v3	ストレージドメインフォーマットのバージョン3は、NFS、POSIX、iSCSI、およびFCストレージドメインに適用できます。
v4	ストレージドメインフォーマットのバージョン4。
v5	ストレージドメインフォーマットのバージョン5は、NFS、POSIX、およびGlusterストレージドメインに適用できます。

7.285.1. v1

ストレージドメインフォーマットのバージョン1は、NFS、iSCSI、およびFCストレージドメインに適用できます。

各ストレージドメインには、独自の構造を説明するメタデータと、仮想マシンのバックアップに使用されるすべての物理ボリュームの名前が含まれています。マスタートメインには、ストレージプール内のすべてのドメインと物理ボリューム名のメタデータが追加で含まれています。このメタデータの合計サイズは2 KiB に制限されており、プールに含めることができるストレージドメインの数が制限されます。テンプレートと仮想マシンのベースイメージは読み取り専用です。

7.285.2. v2

ストレージドメインフォーマットのバージョン2は、iSCSI および FC ストレージドメインに適用できません。

すべてのストレージドメインとプールのメタデータは、論理ボリュームに書き込まれるのではなく、論理ボリュームタグとして保存されます。仮想マシンボリュームに関するメタデータは、引き続きドメインの論理ボリュームに保存されます。物理ボリューム名はメタデータに含まれなくなりました。テンプレートと仮想マシンのベースイメージは読み取り専用です。

7.285.3. v3

ストレージドメインフォーマットのバージョン3は、NFS、POSIX、iSCSI、およびFCストレージドメインに適用できます。

すべてのストレージドメインとプールのメタデータは、論理ボリュームに書き込まれるのではなく、論理ボリュームタグとして保存されます。仮想マシンボリュームに関するメタデータは、引き続きドメインの論理ボリュームに保存されます。仮想マシンとテンプレートのベースイメージは読み取り専用ではなくなりました。この変更により、ライブスナップショット、ライブストレージの移行、およびスナップショットからのクローン作成が可能になります。英語以外のボリューム名に対して、Unicode メタデータのサポートが追加されました。

7.285.4. v5

ストレージドメインフォーマットのバージョン 5 は、NFS、POSIX、および Gluster ストレージドメインに適用できます。

4096 バイトのブロックサイズと可変 sanlock アライメントのサポートが追加されました。

7.286. STORAGETYPE ENUM

ストレージドメインのタイプを表すタイプ。

表7.372 値の概要

名前	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。
managed_block_storage	管理対象ブロック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。

7.286.1. 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。

7.286.2. 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。

7.286.3. 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。

7.286.4. managed_block_storage

管理対象ブロックストレージドメイン。cinderlib を使用して管理されるストレージドメイン。サポート対象のストレージドライバーについては、[Available Drivers](#) を参照してください。

7.287. 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7.373 値の概要

名前	概要
legacy	ネイティブスイッチタイプ。
ovs	Open vSwitch タイプ。

7.288. SYSTEMOPTION STRUCT

システムの設定オプションを表すタイプ。

表7.374 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
値	SystemOptionValue[]	さまざまなシステムバージョンのオプションの値。

7.289. SYSTEMOPTIONVALUE STRUCT

設定オプションの値とバージョンのペアを表すタイプ。

表7.375 属性の概要

名前	タイプ	概要
value	文字列	特定のバージョンの設定オプションの値。
version	文字列	設定オプションのバージョン。

7.290. TAG 構造体

システム内のタグを表します。

表7.376 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.377 リンクの概要

名前	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。

7.291. TEMPLATE 構造体

仮想マシンテンプレートを表すタイプ。テンプレートを使用すると、共通の設定とディスク状態で仮想マシンを迅速にインスタンス化できます。

表7.378 属性の概要

名前	タイプ	概要
auto_pinning_policy	AutoPinningPolicy	自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。
bios	Bios	仮想マシンの BIOS 設定への参照。
comment	文字列	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	Console	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。

名前	タイプ	概要
cpu_pinning_policy	CpuPinningPolicy	CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。
CPU Shares	Integer	
creation_time	Date	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	文字列	
Custom Emulated Machine	文字列	
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	Boolean	true の場合、仮想マシンは削除できません。
description	文字列	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	文字列	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	Icon	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)。

名前	タイプ	概要
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)。
multi_queues_enabled	Boolean	true の場合、各仮想インターフェイスは、利用可能な仮想 CPU に応じて最適な数のキューを取得します。
name	文字列	人間が判読できるプレーンテキストでの名前。
origin	文字列	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。
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。
serial_number	SerialNumber	クラスター内の仮想マシンのシリアル番号。
small_icon	Icon	仮想マシンの小さなアイコン。
Soundcard Enabled	Boolean	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	Boolean	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	Boolean	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。
storage_error_resume_behaviour	VmStorageErrorResumeBehaviour	ストレージエラー後に仮想マシンを再開する方法を決定します。
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。

名前	タイプ	概要
tpm_enabled	Boolean	true の場合、TPM デバイスが仮想マシンに追加されます。
tunnel_migration	Boolean	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である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。
virtio_scsi_multi_queues	Integer	このフィールドの Virtio-SCSI controller のキュー数には <code>virtioScsiMultiQueuesEnabled</code> が true である必要があります。詳細は <code>virtioScsiMultiQueuesEnabled</code> を参照してください。
virtio_scsi_multi_queues_enabled	Boolean	true の場合、Virtio-SCSI デバイスは、使用可能な仮想 CPU とディスク、または指定された <code>virtioScsiMultiQueues</code> に応じて、いくつかの複数キューを取得します。
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。

7.291.1. auto_pinning_policy

自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。



重要

エンジンのバージョン 4.5 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後、削除される可能性があります。代わりに `CpuPinningPolicy` を使用してください。

7.291.2. 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>
```

7.291.3. cpu_pinning_policy

CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。指定されていない場合、CPU ピニング文字列の以前の動作により、CpuPinningPolicy が None または Manual に決定されます。

7.291.4. 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。

7.291.5. 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。

7.291.6. initialization

仮想マシンの初期化設定への参照。



注記

Red Hat Virtualization 4.1.8 以降、このプロパティは空のタグを送信することでクリアできます。

たとえば、**initialization** 属性をクリアするには、次のようなリクエストを送ります。

```
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>
  <initialization/>
</vm>
```

このようなリクエストへのレスポンス、およびヘッダー **All-Content: true** を持つリクエストには、引き続きこの属性が含まれます。

7.291.7. 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。

7.291.8. 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。

7.291.9. 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>
```

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。上記の例を使用して、仮想マシンが **up** 状態のときにメモリーを増やすことができます。サイズの増分は、**HotPlugMemoryBlockSizeMb** 設定値 (デフォルトでは 256 MiB) の値で割り切れる必要があります。メモリーサイズの増分がこの値で割り切れない場合、メモリーサイズの変更は次の実行設定にのみ保存されます。メモリーのホットプラグ操作が成功するたびに、1つまたは2つの新しいメモリーデバイスが作成されます。

メモリーのホットアンプラグは、Red Hat Virtualization 4.2 以降でサポートされています。メモリーのホットアンプラグは、仮想マシンの状態が **up** の場合にのみ実行できます。ホットアンプラグ操作で取り外すことができるのは、以前にホットプラグされたメモリーデバイスのみです。要求されたメモリーの減少分は、以前にホットプラグされたメモリーデバイスの組み合わせのサイズに一致するように切り捨てられます。要求されたメモリー値は、丸められずに次の実行設定に格納されます。



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

Red Hat Virtualization Manager は内部的に値を切り捨てて整数の MiB ($1 \text{ MiB} = 2^{20} \text{ バイト}$) にします。

7.291.10. migration

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。



注記

このメソッドが返す ID で移行ポリシーを照会する API はまだ実装されていません。/ovirt-engine/api/options/MigrationPolicies を使用して、すべての移行ポリシーとその ID のリストを取得します。

7.291.11. 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。

7.291.12. 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**

7.291.13. 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>
```

7.291.14. 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。

7.291.15. 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。

7.291.16. tpm_enabled

true の場合、TPM デバイスが仮想マシンに追加されます。デフォルト値は **false** です。このプロパティは、"All-Content=true" ヘッダーが設定されている場合に、フェッチするときのみ表示されません。

表7.379 リンクの概要

名前	タイプ	概要
cdroms	Cdrrom[]	テンプレートに接続された CD-ROM デバイスへの参照。
cluster	クラスター	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。

名前	タイプ	概要
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。
mediated_devices	VmMediatedDevice[]	仲介デバイスの設定。
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。

7.292. 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7.380 値の概要

名前	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。

7.293. 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7.381 属性の概要

名前	タイプ	概要
<code>version_name</code>	文字列	このバージョンの名前。
<code>version_number</code>	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。

7.293.1. 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7.382 リンクの概要

名前	タイプ	概要
<code>base_template</code>	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。

7.294. 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7.383 属性の概要

名前	タイプ	概要
<code>expiry</code>	Integer	チケットの有効期間 (秒単位)。
<code>value</code>	文字列	仮想マシンのアクセスチケット。

7.295. TIMEZONE 構造体

タイムゾーンの表現。

表7.384 属性の概要

名前	タイプ	概要
<code>name</code>	文字列	タイムゾーンの名前。
<code>utc_offset</code>	文字列	UTC オフセット。

7.295.1. utc_offset

UTC オフセット。

UTC からのオフセット。

7.296. TPMSUPPORT ENUM

表7.385 値の概要

名前	概要
必須	TPM はオペレーティングシステムで必要です
supported	TPM はサポートされていますが、オプションです
unsupported	

7.297. TRANSPARENTHUGEPAGES 構造体

Transparent huge pages (THP) のサポートを表すタイプ。

表7.386 属性の概要

名前	タイプ	概要
enabled	Boolean	THP サポートを有効にします。

7.298. 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7.387 値の概要

名前	概要
rdma	Remote direct memory access。
tcp	TCP。

7.299. UNMANAGEDNETWORK 構造体

表7.388 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.389 リンクの概要

名前	タイプ	概要
ホスト	ホスト	
host_nic	HostNic	

7.300. USB 構造体

仮想マシンの USB デバイスの設定。

表7.390 属性の概要

名前	タイプ	概要
enabled	Boolean	USB デバイスを含めるかどうかを決定します。
type	UsbType	USB タイプ、現在は native のみがサポートされます。

7.301. USBTYPE ENUM

USB デバイスリダイレクトのタイプ。

表7.391 値の概要

名前	概要
legacy	レガシー USB リダイレクト。
native	ネイティブ USB リダイレクト。

7.301.1. legacy

レガシー USB リダイレクト。

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。

7.301.2. native

ネイティブ USB リダイレクト。

ネイティブ USB リダイレクトでは、Linux および Windows 仮想マシンの KVM/SPICE USB リダイレクトが許可されます。仮想 (ゲスト) マシンには、ネイティブ USB 用のゲストがインストールされたエージェントやドライバーは必要ありません。Linux クライアントでは、USB リダイレクトに必要なすべてのパッケージは **virt-viewer** パッケージで提供されます。Windows クライアントで **usbdk** パッケージもインストールする必要があります。

7.302. USER 構造体

システム内のユーザーを表します。

表7.392 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	
description	文字列	プレーンテキストでの人間が判読できる説明。
domain_entry_id	文字列	
email	文字列	
id	文字列	一意の ID
last_name	文字列	
logged_in	Boolean	
name	文字列	人間が判読できるプレーンテキストでの名前。
namespace	文字列	ユーザーが存在する名前空間。
password	文字列	
principal	文字列	user_name と同様です。
user_name	文字列	ユーザーのユーザー名。
user_options	Property[]	ユーザーオプションを使用すると、個々のユーザーごとに設定をカスタマイズするために使用されるキー/値のプロパティを保存できます。

7.302.1. 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 [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 [oVirt Engine extension - AAA - JDBC](#) を参照してください。

7.302.2. 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。

7.302.3. 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。

7.302.4. user_options

ユーザーオプションを使用すると、個々のユーザーごとに設定をカスタマイズするために使用されるキー/値のプロパティを保存できます。バージョン 4.4.5 以降、このプロパティは非推奨となり、後方互換性のためにのみ保持されていることに注意してください。これは今後削除されます。代わりに [options](#) エンドポイントを使用してください。

表7.393 リンクの概要

名前	タイプ	概要
domain	Domain	
groups	Group[]	
options	UserOption[]	
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。

7.303. USEROPTION 構造体

ユーザーオプションを使用すると、個々のユーザーごとに設定をカスタマイズするために使用されるキー/値のプロパティを保存できます。

表7.394 属性の概要

名前	タイプ	概要
comment	文字列	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。

名前	タイプ	概要
content	文字列	文字列としてエンコードされた JSON コンテンツ。
description	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

7.303.1. content

文字列としてエンコードされた JSON コンテンツ。有効な JSON がサポートされています。

表7.395 リンクの概要

名前	タイプ	概要
user	User	

7.304. VALUE 構造体

表7.396 属性の概要

名前	タイプ	概要
datum	10 進数	
detail	文字列	

7.305. VALUETYPE ENUM

表7.397 値の概要

名前	概要
decimal	
整数	
string	

7.306. VCPUPIN 構造体

表7.398 属性の概要

名前	タイプ	概要
<code>cpu_set</code>	文字列	
<code>vcpu</code>	Integer	

7.307. VENDOR 構造体

表7.399 属性の概要

名前	タイプ	概要
<code>comment</code>	文字列	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>	文字列	プレーンテキストでの人間が判読できる説明。
<code>id</code>	文字列	一意の ID
<code>name</code>	文字列	人間が判読できるプレーンテキストでの名前。

7.308. VERSION 構造体

表7.400 属性の概要

名前	タイプ	概要
<code>build</code>	Integer	
<code>comment</code>	文字列	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>	文字列	プレーンテキストでの人間が判読できる説明。
<code>full_version</code>	文字列	
<code>id</code>	文字列	一意の ID
<code>major</code>	Integer	
<code>minor</code>	Integer	
<code>name</code>	文字列	人間が判読できるプレーンテキストでの名前。
<code>revision</code>	Integer	

7.309. VGPUPLACEMENT ENUM

vGPU 配置ストラテジー。

最初の使用可能な物理カードに vGPU を配置するか、複数の物理カードに分散することができます。

表7.401 値の概要

名前	概要
consolidated	統合配置を使用します。
separated	分離配置を使用します。

7.309.1. consolidated

統合配置を使用します。各 vGPU は、空き容量のある最初の物理カードに配置されます。

これはデフォルトの配置であり、物理カードの利用可能なスペースをすべて利用します。

7.309.2. separated

分離配置を使用します。可能であれば、各 vGPU は個別の物理カードに配置されます。

これは、vGPU のパフォーマンスを向上させるのに役立ちます。

7.310. 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7.402 属性の概要

名前	タイプ	概要
enabled	Boolean	Virtio SCSI サポートを有効にします。

7.311. 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7.403 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
index	Integer	
memory	Integer	NUMA ノードのメモリー (MB)。
name	文字列	人間が判読できるプレーンテキストでの名前。
node_distance	文字列	
numa_node_pins	NumaNodePin[]	
numa_tune_mode	NumaTuneMode	NUMA トポロジーの適用方法。

表7.404 リンクの概要

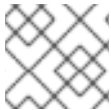
名前	タイプ	概要
host	host	
statistics	Statistic[]	各 host NUMA ノードリソースは、host NUMA ノード固有の統計の統計サブコレクションを公開します。
vm	Vm	

7.311.1. statistics

各 host NUMA ノードリソースは、host 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。

名前	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。

7.312. VLAN 構造体

仮想 LAN (VLAN) タイプを表すタイプ。

表7.405 属性の概要

名前	タイプ	概要
id	Integer	仮想 LAN ID。

7.313. VM 構造体

仮想マシンを表します。

表7.406 属性の概要

名前	タイプ	概要
auto_pinning_policy	AutoPinningPolicy	自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。
bios	Bios	仮想マシンの BIOS 設定への参照。
comment	文字列	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	Console	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_pinning_policy	CpuPinningPolicy	CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。
CPU Shares	Integer	
creation_time	Date	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	文字列	
Custom Emulated Machine	文字列	
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	Boolean	true の場合、仮想マシンは削除できません。
description	文字列	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。

名前	タイプ	概要
fqdn	文字列	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)。
has_illegal_images	Boolean	仮想マシンに、ディスクが ILLEGAL 状態のスナップショットがあるかどうかを示します。
high_availability	高可用性	仮想マシンの高可用性設定。
id	文字列	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	Icon	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)。
multi_queues_enabled	Boolean	true の場合、各仮想インターフェイスは、利用可能な仮想 CPU に応じて最適な数のキューを取得します。
name	文字列	人間が判読できるプレーンテキストでの名前。
next_run_configuration_exists	Boolean	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_mode	NumaTuneMode	NUMA トポロジーの適用方法。
origin	文字列	この仮想マシンのオリジン。

名前	タイプ	概要
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_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	Boolean	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	Icon	仮想マシンの小さなアイコン。
Soundcard Enabled	Boolean	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	Boolean	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	Date	仮想マシンが起動された日付。
stateless	Boolean	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	文字列	人間が読める現在のステータスの詳細。
stop_reason	文字列	仮想マシンが停止した理由。
stop_time	Date	仮想マシンが停止された日付。
storage_error_resume_behaviour	VmStorageErrorResumeBehaviour	ストレージエラー後に仮想マシンを再開する方法を決定します。

名前	タイプ	概要
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。
tpm_enabled	Boolean	true の場合、TPM デバイスが仮想マシンに追加されます。
tunnel_migration	Boolean	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	Boolean	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。
virtio_scsi_multi_queues	Integer	このフィールドの Virtio-SCSI controller のキュー数には <code>virtioScsiMultiQueuesEnabled</code> が <code>true</code> である必要があります。詳細は <code>virtioScsiMultiQueuesEnabled</code> を参照してください。
virtio_scsi_multi_queues_enabled	Boolean	true の場合、Virtio-SCSI デバイスは、使用可能な仮想 CPU とディスク、または指定された <code>virtioScsiMultiQueues</code> に応じて、いくつかの複数キューを取得します。

7.313.1. auto_pinning_policy

自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。



重要

エンジンのバージョン 4.5 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後、削除される可能性があります。代わりに `CpuPinningPolicy` を使用してください。

7.313.2. 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>
```

7.313.3. cpu_pinning_policy

CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。指定されていない場合、CPU ピニング文字列の以前の動作により、CpuPinningPolicy が None または Manual に決定されます。

7.313.4. 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。

7.313.5. 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。

7.313.6. initialization

仮想マシンの初期化設定への参照。



注記

Red Hat Virtualization 4.1.8 以降、このプロパティは空のタグを送信することでクリアできます。

たとえば、**initialization** 属性をクリアするには、次のようなリクエストを送ります。

```
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>
  <initialization/>
</vm>
```

このようなリクエストへのレスポンス、およびヘッダー **All-Content: true** を持つリクエストには、引き続きこの属性が含まれます。

7.313.7. 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。

7.313.8. 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。

7.313.9. 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。

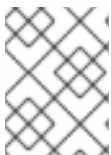
```
PUT /ovirt-engine/api/vms/123
```

リクエスト本文は、以下のようになります。

```
<vm>
  <memory>1073741824</memory>
</vm>
```

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。上記の例を使用して、仮想マシンが `up` 状態のときにメモリーを増やすことができます。サイズの増分は、**HotPlugMemoryBlockSizeMb** 設定値 (デフォルトでは 256 MiB) の値で割り切れる必要があります。メモリーサイズの増分がこの値で割り切れない場合、メモリーサイズの変更は次の実行設定にのみ保存されます。メモリーのホットプラグ操作が成功するたびに、1つまたは2つの新しいメモリーデバイスが作成されます。

メモリーのホットアンプラグは、Red Hat Virtualization 4.2 以降でサポートされています。メモリーのホットアンプラグは、仮想マシンの状態が `up` の場合にのみ実行できます。ホットアンプラグ操作で取り外すことができるのは、以前にホットプラグされたメモリーデバイスのみです。要求されたメモリーの減少分は、以前にホットプラグされたメモリーデバイスの組み合わせのサイズに一致するように切り捨てられます。要求されたメモリー値は、丸められずに次の実行設定に格納されます。



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

Red Hat Virtualization Manager は内部的に値を切り捨てて整数の MiB ($1 \text{ MiB} = 2^{20} \text{ バイト}$) にします。

7.313.10. migration

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。



注記

このメソッドが返す ID で移行ポリシーを照会する API はまだ実装されていません。/ovirt-engine/api/options/MigrationPolicies を使用して、すべての移行ポリシーとその ID のリストを取得します。

7.313.11. 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。

7.313.12. 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。

7.313.13. numa_tune_mode

NUMA トポロジーの適用方法。非推奨となりました。vNUMA ノードごとの NUMA 調整が使用されません。

7.313.14. 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**

7.313.15. 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>
```

7.313.16. 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。

7.313.17. 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。

7.313.18. 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。

7.313.19. tpm_enabled

true の場合、TPM デバイスが仮想マシンに追加されます。デフォルト値は **false** です。このプロパティは、"All-Content=true" ヘッダーが設定されている場合に、フェッチするときのみ表示されます。

表7.407 リンクの概要

名前	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	クラスター	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。
dynamic_cpu	DynamicCpu	仮想マシン CPU の動的設定。
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。
ホスト	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。
mediated_devices	VmMediatedDevice[]	仲介デバイスの設定。
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。

名前	タイプ	概要
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[]	
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。

7.313.20. 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。

7.313.21. 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>

```

7.313.22. 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。

7.313.23. statistics

この仮想マシンから収集された統計データ。

一部の統計、特に **memory.buffered** および **memory.cached** は、[Red Hat Virtualization ゲストエージェント](#) が仮想マシンにインストールされている場合にのみ使用できることに注意してください。

7.314. VMAFFINITY ENUM

表7.408 値の概要

名前	概要
migratable	
pinned	
user_migratable	

7.315. 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7.409 属性の概要

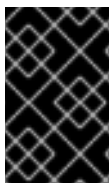
名前	タイプ	概要
auto_pinning_policy	AutoPinningPolicy	自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。
bios	Bios	仮想マシンの BIOS 設定への参照。
comment	文字列	このオブジェクトに関するコメントを含むフリーテキスト。
console	Console	この仮想マシン用に設定されたコンソール。
cpu	Cpu	仮想マシン CPU の設定。
cpu_pinning_policy	CpuPinningPolicy	CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。
CPU Shares	Integer	
creation_time	Date	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	文字列	
Custom Emulated Machine	文字列	
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	Boolean	true の場合、仮想マシンは削除できません。
description	文字列	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	文字列	一意の ID

名前	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	Icon	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)。
multi_queues_e nabled	Boolean	true の場合、各仮想インターフェイスは、利用可能な仮想 CPU に応じて最適な数のキューを取得します。
name	文字列	人間が判読できるプレーンテキストでの名前。
origin	文字列	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。
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。
serial_number	SerialNumber	クラスター内の仮想マシンのシリアル番号。
small_icon	Icon	仮想マシンの小さなアイコン。
Soundcard Enabled	Boolean	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	Boolean	true の場合、仮想マシンは起動後、最初は 'paused' 状態になります。

名前	タイプ	概要
stateless	Boolean	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。
storage_error_resume_behavior	VmStorageErrorResumeBehaviour	ストレージエラー後に仮想マシンを再開する方法を決定します。
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。
tpm_enabled	Boolean	true の場合、TPM デバイスが仮想マシンに追加されます。
tunnel_migration	Boolean	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。
virtio_scsi_multi_queues	Integer	このフィールドの Virtio-SCSI controller のキュー数には <code>virtioScsiMultiQueuesEnabled</code> が <code>true</code> である必要があります。詳細は <code>virtioScsiMultiQueuesEnabled</code> を参照してください。
virtio_scsi_multi_queues_enabled	Boolean	true の場合、Virtio-SCSI デバイスは、使用可能な仮想 CPU とディスク、または指定された <code>virtioScsiMultiQueues</code> に応じて、いくつかの複数キューを取得します。

7.315.1. auto_pinning_policy

自動 CPU および NUMA 設定を適用するかどうか、およびその適用方法を指定します。



重要

エンジンのバージョン 4.5 以降、この操作は非推奨になり、後方互換性のためにのみ保持されます。これは今後、削除される可能性があります。代わりに `CpuPinningPolicy` を使用してください。

7.315.2. 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>
```

7.315.3. cpu_pinning_policy

CPU および NUMA 設定を適用するかどうか、またどのように適用するかを指定します。指定されていない場合、CPU ピニング文字列の以前の動作により、CpuPinningPolicy が None または Manual に決定されます。

7.315.4. 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。

7.315.5. 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。

7.315.6. initialization

仮想マシンの初期化設定への参照。



注記

Red Hat Virtualization 4.1.8 以降、このプロパティは空のタグを送信することでクリアできます。

たとえば、**initialization** 属性をクリアするには、次のようなリクエストを送ります。

```
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>
  <initialization/>
</vm>
```

このようなリクエストへのレスポンス、およびヘッダー **All-Content: true** を持つリクエストには、引き続きこの属性が含まれます。

7.315.7. 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。

7.315.8. 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。

7.315.9. 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>
```

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。上記の例を使用して、仮想マシンが **up** 状態のときにメモリーを増やすことができます。サイズの増分は、**HotPlugMemoryBlockSizeMb** 設定値 (デフォルトでは 256 MiB) の値で割り切れる必要があります。メモリーサイズの増分がこの値で割り切れない場合、メモリーサイズの変更は次の実行設定にのみ保存されます。メモリーのホットプラグ操作が成功するたびに、1つまたは2つの新しいメモリーデバイスが作成されます。

メモリーのホットアンプラグは、Red Hat Virtualization 4.2 以降でサポートされています。メモリーのホットアンプラグは、仮想マシンの状態が **up** の場合にのみ実行できます。ホットアンプラグ操作で取り外すことができるのは、以前にホットプラグされたメモリーデバイスのみです。要求されたメモリーの減少分は、以前にホットプラグされたメモリーデバイスの組み合わせのサイズに一致するように切り捨てられます。要求されたメモリー値は、丸められずに次の実行設定に格納されます。



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

Red Hat Virtualization Manager は内部的に値を切り捨てて整数の MiB ($1 \text{ MiB} = 2^{20} \text{ バイト}$) にします。

7.315.10. migration

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。



注記

このメソッドが返す ID で移行ポリシーを照会する API はまだ実装されていません。`/ovirt-engine/api/options/MigrationPolicies` を使用して、すべての移行ポリシーとその ID のリストを取得します。

7.315.11. 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。

7.315.12. origin

この仮想マシンのオリジン。

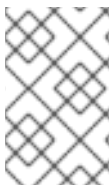
値:

- `ovirt`
- `rhev`
- `vmware`
- `xen`
- `external`
- `hosted_engine`
- `managed_hosted_engine`
- `kvm`
- `physical_machine`
- `hyperv`

7.315.13. 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>
```

7.315.14. 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。

7.315.15. 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。

7.315.16. tpm_enabled

true の場合、TPM デバイスが仮想マシンに追加されます。デフォルト値は **false** です。このプロパティは、"All-Content=true" ヘッダーが設定されている場合に、フェッチするときのみ表示されません。

表7.410 リンクの概要

名前	タイプ	概要
cluster	クラスター	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。

7.316. VMDEVICETYPE ENUM

表7.411 値の概要

名前	概要
cdrom	
floppy	

7.317. VMMEDATEDDEVICE STRUCT

仮想マシン仲介デバイスは、vGPU 仲介デバイスのプロパティを指定するフェイクデバイスです。これは実際のデバイスではなく、ホストデバイスの一部を設定する方法の仕様として機能します。

表7.412 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。
spec_params	Property[]	デバイスのプロパティ。

表7.413 リンクの概要

名前	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。

名前	タイプ	概要
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。

7.317.1. 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。

7.318. VMPLACEMENTPOLICY 構造体

表7.414 属性の概要

名前	タイプ	概要
affinity	VmAffinity	

表7.415 リンクの概要

名前	タイプ	概要
hosts	Host[]	

7.319. VMPOOL 構造体

仮想マシンプールを表すタイプ。

表7.416 属性の概要

名前	タイプ	概要
auto_storage_select	Boolean	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。
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
display	表示	プール内の仮想マシンに対して設定された表示設定。
id	文字列	一意の ID

名前	タイプ	概要
max_user_vms	Integer	特定のユーザーに割り当てることができるプール内の仮想マシンの最大数。
name	文字列	人間が判読できるプレーンテキストでの名前。
prestarted_vms	Integer	システムは、指定された数の仮想マシンをプールから事前起動しようとしています。
rng_device	RngDevice	プール内の仮想マシン用に設定された乱数ジェネレーターデバイス。
size	Integer	プール内の仮想マシンの数。
Soundcard Enabled	Boolean	プール内の仮想マシンに対してサウンドカードを設定する必要があるかどうかを示します。
stateful	Boolean	仮想マシンプールのステートフルフラグ。
tpm_enabled	Boolean	true の場合、TPM デバイスが仮想マシンに追加されます。
type	VmPoolType	プール内の仮想マシンの割り当て解除ポリシー。
use_latest_template_version	Boolean	プール内の仮想マシンが、プールのベースになっているテンプレートの新しいバージョンに更新されているかどうかを示します。

7.319.1. 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。

7.319.2. display

プール内の仮想マシンに対して設定された表示設定。

**警告**

この属性は機能しておらず、現在は非推奨となっていることに注意してください。代わりに **Vm.display** を使用してください。

7.319.3. prestarted_vms

システムは、指定された数の仮想マシンをプールから事前起動しようとします。

これらの仮想マシンは、どのユーザーにもアタッチされずに起動されます。こうすることで、ユーザーはプールからより速く仮想マシンを取得できます。

7.319.4. 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。

7.319.5. tpm_enabled

true の場合、TPM デバイスが仮想マシンに追加されます。デフォルト値は **false** です。このプロパティは、"All-Content=true" ヘッダーが設定されている場合に、フェッチするときのみ表示されます。

表7.417 リンクの概要

名前	タイプ	概要
cluster	クラスター	プールが存在するクラスターへの参照。
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[]	この仮想マシンプールに設定されたパーミッション。
template	Template	プールのベースとなるテンプレートへの参照。
vm	Vm	プールの一部である任意の仮想マシンへの参照。

7.319.6. 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。

7.319.7. vm

プールの一部である任意の仮想マシンへの参照。

この仮想マシンは、プールのテンプレートの最新バージョンをベースにしていない可能性があることに注意してください。

7.320. VMPOOLTYPE ENUM

仮想マシンプール内の仮想マシンの割り当て解除ポリシーを表すタイプ。

表7.418 値の概要

名前	概要
automatic	このポリシーは、プール内の仮想マシンがシステムによって自動的に割り当て解除されることを示します。
manual	このポリシーは、プール内の仮想マシンが管理者によって手動で割り当て解除されることを示します。

7.320.1. automatic

このポリシーは、プール内の仮想マシンがシステムによって自動的に割り当て解除されることを示します。

このポリシーでは、プールの一部でユーザーに割り当てられている仮想マシンがシャットダウンされると、その仮想マシンはユーザーから切り離され、プールのデフォルト状態に復元され、プールに戻されます (つまり、その仮想マシンを別のユーザーに割り当てることができます)。

7.320.2. manual

このポリシーは、プール内の仮想マシンが管理者によって手動で割り当て解除されることを示します。

このポリシーを使用すると、プールの一部である仮想マシンはそのユーザーに割り当てられたままになり、シャットダウン時にその状態が保持されます。仮想マシンをプールに戻すには、管理者がその仮想マシンに対するユーザーのパーミッションを削除して、明示的に割り当てを解除する必要があります。

7.321. VMSTATUS ENUM

仮想マシンの状態を表すタイプ。

表7.419 値の概要

名前	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。

名前	概要
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。
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。

7.321.1. 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。

7.321.2. 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。

7.321.3. 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。

7.321.4. 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。

7.321.5. 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。

7.321.6. 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。

7.321.7. 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。

7.321.8. 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。

7.322. VMSTORAGEERRORRESUMEBEHAVIOUR ENUM

この仮想マシンにいくつかのディスクがあるストレージが応答しなくなると、仮想マシンは一時停止します。

これは可能なオプションであり、ストレージが再び利用可能になった瞬間に仮想マシンで何が起こるべきかを示しています。

表7.420 値の概要

名前	概要
auto_resume	ストレージが再び使用可能になると、仮想マシンは自動的に再開されます。
kill	仮想マシンはタイムアウト後に強制終了されます (ハイパーバイザーで設定可能)。
leave_paused	仮想マシンに何もしません。

7.322.1. auto_resume

ストレージが再び使用可能になると、仮想マシンは自動的に再開されます。

これは、4.2 より前で利用できる唯一の動作です。

7.322.2. kill

仮想マシンはタイムアウト後に強制終了されます (ハイパーバイザーで設定可能)。

これは、リースを使用する高可用性仮想マシンでサポートされる唯一のオプションです。高可用性の仮想マシンがこのインフラストラクチャーを使用して再起動され、なんらかの形で再開されると、スプリットブレインが発生するリスクがあります。

7.322.3. leave_paused

仮想マシンに何もしません。

カスタムフェイルオーバーが実装されていて、ユーザーが仮想マシンを再開しない場合に便利です。

7.323. VMSUMMARY 構造体

特定のホスト上の仮想マシンに関連する情報を含むタイプ。

表7.421 属性の概要

名前	タイプ	概要
active	Integer	ホスト上でアクティブな仮想マシンの数。
migrating	Integer	ホストとの間で移行中の仮想マシンの数。
total	Integer	ホストに存在する仮想マシンの数。

7.324. VMATYPE ENUM

仮想マシンの最適化対象を表す型。

表7.422 値の概要

名前	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。
high_performance	仮想マシンは、高性能仮想マシン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。

7.324.1. 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ることを意味しています。

7.324.2. high_performance

仮想マシンは、高性能仮想マシンとして使用することを目的としています。

現在のところ、仮想マシンの設定は可能な限り最高のパフォーマンスで、可能な限りベアメタルに近いパフォーマンスメトリックで実行するように自動的に設定されます。

可能な限り最高のパフォーマンスを得るために推奨される設定設定の一部は、自動的に設定できません。仮想マシンを実行する前に手動で設定することをお勧めします。

次の設定変更は自動的に設定されます。

- ヘッドレスモードを有効にする。
- シリアルコンソールを有効にする。
- パススルーホスト CPU を有効にする。
- I/O スレッドを有効にする。
- I/O スレッドのピンングを有効にし、ピンングトポロジを設定する。
- 準仮想化乱数ジェネレーター PCI (virtio-rng) デバイスを有効にする。
- すべての USB デバイスを無効にする。
- サウンドカードデバイスを無効にする。
- スマートカードデバイスを無効にする。
- メモリーバルーンデバイスを無効にします。
- ウォッチドッグデバイスを無効にする。
- 移行を無効にする。
- 高可用性を無効にする。

次の推奨される設定変更は、ユーザーが手動で設定する必要があります。

- CPU ピンングトポロジを有効にします。
- Non-Uniform Memory Access (NUMA) ピンングトポロジを有効にします。
- huge page 設定を有効にして設定します。
- カーネルの同一ページマージ (KSM) を無効にします。

7.324.3. 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ています。

7.325. VNICPASSTHROUGH 構造体

表7.423 属性の概要

名前	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。

7.326. 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7.424 値の概要

名前	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。

7.327. 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7.425 属性の概要

名前	タイプ	概要
comment	文字列	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.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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
migratable	Boolean	pass_through NIC が移行可能かどうかをマークします。
name	文字列	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	Boolean	ポートミラーリングを有効にします。

7.327.1. migratable

`pass_through` NIC が移行可能かどうかをマークします。

`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。リクエスト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。

仮想マシンを移行する場合、この仮想マシンは、すべての `pass_through` NIC が **migratable** としてフラグ付けされている場合にのみ移行されます。

7.327.2. 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、パススルーを有効にできません。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。

7.327.3. 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** と **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。ホスト外部のネットワークのトラフィックは増加しませんが、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 と 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 プロファイルを使用した NIC のホットリンク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7.426 リンクの概要

名前	タイプ	概要
failover	VnicProfile	ダウンタイムのない SR-IOV 移行向けのフェイルオーバー vNIC プロファイル

名前	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。

7.327.4. 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、仮想マシンとの間のネットワークパケットトラフィックを管理する機能を強化します。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リングのルール、または両方の組み合わせが含まれる場合があります。

7.327.5. 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。

7.328. 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する非推奨のタイプ。

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

次の形式は 4.2.1 以降非推奨となり、今後削除される予定です。

```
<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_mappings>
```



```

<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7.427 属性の概要

名前	タイプ	概要
<code>source_network_name</code>	文字列	外部ネットワークの名前を記述する非推奨の属性。
<code>source_network_profile_name</code>	文字列	外部ネットワークプロファイルの名前を記述する非推奨の属性。

7.328.1. source_network_name

外部ネットワークの名前を記述する非推奨の属性。



警告

この属性は、エンジンのバージョン 4.2.1 以降は非推奨となっており、後方互換性のためにのみ保持されていることに注意してください。これは今後削除されます。

7.328.2. source_network_profile_name

外部ネットワークプロファイルの名前を記述する非推奨の属性。



警告

この属性は、エンジンのバージョン 4.2.1 以降は非推奨となっており、後方互換性のためにのみ保持されていることに注意してください。これは今後削除されます。

表7.428 リンクの概要

名前	タイプ	概要
<code>target_vnic_profile</code>	VnicProfile	既存の仮想 NIC プロファイルを記述する非推奨の属性。

7.328.3. target_vnic_profile

既存の仮想 NIC プロファイルを記述する非推奨の属性。



警告

この属性は、エンジンのバージョン 4.2.1 以降は非推奨となっており、後方互換性のためにのみ保持されていることに注意してください。これは今後削除されます。

7.329. VOLUMEGROUP STRUCT

表7.429 属性の概要

名前	タイプ	概要
id	文字列	
logical_units	LogicalUnit[]	
name	文字列	

7.330. 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7.430 属性の概要

名前	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	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
id	文字列	一意の ID
model	WatchdogModel	ウォッチドッグデバイスのモデル。
name	文字列	人間が判読できるプレーンテキストでの名前。

7.330.1. 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7.431 リンクの概要

名前	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。

7.330.2. 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。

7.331. 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7.432 値の概要

名前	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。

7.331.1. 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。

7.332. WATCHDOGMODEL ENUM

このタイプはウォッチドッグモデルを表します。

表7.433 値の概要

名前	概要
diag288	S390X マシンのウォッチドッグモデル。
i6300esb	PCI ベースのウォッチドッグモデル。

7.332.1. diag288

S390X マシンのウォッチドッグモデル。

S390X には、DIAG288 命令を介して制御される統合ウォッチドッグファシリティがあります。このモデルは、S390X 仮想マシンに使用します。

7.332.2. i6300esb

PCI ベースのウォッチドッグモデル。

x86_64 および PPC64 仮想マシンには I6300ESB ウォッチドッグを使用します。

7.333. WEIGHT 構造体

表7.434 属性の概要

名前	タイプ	概要
comment	文字列	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	文字列	プレーンテキストでの人間が判読できる説明。
factor	Integer	
id	文字列	一意の ID
name	文字列	人間が判読できるプレーンテキストでの名前。

表7.435 リンクの概要

名前	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](#) 単精度浮動小数点数を使用してこの型を実装しています。

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。

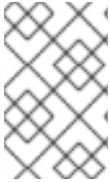
```
<vm id="123" href="/ovirt-engine/api/vms/123">
...
<creation_time>2016-09-08T09:53:35.138+02:00</creation_time>
...
</vm>
```

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。つまり、1970 年 1 月 1 日からのミリ秒数を含む整数 (**エポック時間** と呼ばれます)。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

どちらの場合も、エンジンによって返される日付は、エンジンが実行されているサーバーで設定されたタイムゾーンを使用します。これらの例では、タイムゾーンは UTC+2 です。

付録B API バージョン 4 の変更点

このセクションでは、API のバージョン 4 以降に導入された下位互換性を確保できない変更を列挙します。

B.1. API バージョン 4.0 での変更点 (バージョン 3.6 の後継)

B.1.1. 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.1.2. 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
CPUs	Cpus
CdRom	Cdrom
Cdroms	Cdroms
DNS	Dns
GuestNicConfiguration	NicConfiguration
GuestNicsConfiguration	NicConfigurations
HostNICStates	HostNICStates
HostNIC	HostNic
HostStorage[]	HostStorages
IO	Io
IP	Ip
IPs	Ips
KSM	Ksm

Version 3	Version 4
MAC	Mac
NIC	Nic
PreviewVMs	PreviewVms
QoS	Qos
QoSs	Qoss
RSDL	RsdI
SELinux	SELINUX=
SPM	Spm
SSHPublicKey	SshPublicKey
SSHPublicKeys	SshPublicKeys
SSH	Ssh
SkipIfSDActive	SkipIfSdActive
Slaves	HostNics
Storage	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.1.3. 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.1.4. 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.1.5. 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.1.6. 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.1.7. 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.1.8. 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.1.9. 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.1.10. 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.1.11. 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.1.12. 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.1.13. 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.1.14. 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 Nehalem 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 Nehalem Family</type>
  <architecture>X86_64</architecture>
</cpu>
```

B.1.15. 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.1.16. 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.1.17. 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ー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.1.18. 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ー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.1.19. 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ー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.1.20. 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ー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.1.21. 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.1.22. 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.1.23. 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.1.24. 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.1.25. 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.1.26. 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.1.27. 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.1.28. 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.1.29. 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.1.30. 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、仮想マシンの削除時に、ディスクを削除する必要はなく、仮想マシンから切り離すだけであることを指定するために使用されていました。

```
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.1.31. 要素 **vm_pool** 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 **vm_pool** 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.1.32. 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.1.33. 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.1.34. 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.1.35. 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.1.36. 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.1.37. 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.1.38. 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、スケジュール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.1.39. 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 **strip_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.1.40. 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.1.41. 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_pin>...</vcpu_pin>
        ...
      </vcpu_pins>
    </cpu_tune>
  </cpu>
</vm>

```

B.1.42. `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.1.43. **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.1.44. 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.1.45. 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.1.46. 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.1.47. 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.1.48. 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.1.49. 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.1.50. 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.1.51. 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.1.52. 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.1.53. 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.1.54. 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 Nehalem 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 Nehalem Family</name>
      <level>2</level>
      <architecture>x86_64</architecture>
    </cpu_type>
    ...
  </cpu_types>
  ...
</cluster_level>

```

B.1.55. 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.1.56. **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>
```

B.2. エンジンバージョン 4.5 の変更点

B.2.1. Openstack Volume (Cinder) 統合は Managed Block Storage に置き換えられる

付録C 法的通知

Copyright © 2022 Red Hat, Inc.

Licensed under the ([Creative Commons Attribution–ShareAlike 4.0 International License](#)). Derived from documentation for the ([oVirt Project](#)). If you distribute this document or an adaptation of it, you must provide the URL for the original version.

Modified versions must remove all Red Hat trademarks.

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.