Table of Contents

1. Introduction . . . 1
   1.1. Representational State Transfer . . . 1
   1.2. API Prerequisites . . . 1
2. Authentication and security . . . 2
   2.1. TLS/SSL Certification . . . 2
   2.2. Authentication . . . 5
3. Quick start example . . . 9
   3.1. Example: Access API entry point . . . 9
   3.2. Example: List data centers . . . 12
   3.3. Example: List host clusters . . . 13
   3.4. Example: List logical networks . . . 14
   3.5. Example: List hosts . . . 15
   3.6. Example: Create NFS data storage . . . 17
   3.7. Example: Create NFS ISO storage . . . 19
   3.8. Example: Attach storage domains to data center . . . 21
   3.9. Example: Create virtual machine . . . 22
   3.10. Example: Create a virtual machine NIC . . . 24
   3.11. Example: Create virtual machine disk . . . 25
   3.12. Example: Attach ISO image to virtual machine . . . 26
   3.13. Example: Start the virtual machine . . . 28
4. Requests . . . 29
5. Services . . . 53
   5.1. AffinityGroup . . . 53
   5.2. AffinityGroupVm . . . 54
   5.3. AffinityGroupVms . . . 54
   5.4. AffinityGroups . . . 55
   5.5. AffinityLabel . . . 57
   5.6. AffinityLabelHost . . . 57
   5.7. AffinityLabelHosts . . . 58
   5.8. AffinityLabelVm . . . 58
   5.9. AffinityLabelVms . . . 59
   5.10. AffinityLabels . . . 60
   5.11. AssignedAffinityLabel . . . 60
   5.12. AssignedAffinityLabels . . . 61
   5.13. AssignedCpuProfile . . . 61
   5.14. AssignedCpuProfiles . . . 62
   5.15. AssignedDiskProfile . . . 63
   5.16. AssignedDiskProfiles . . . 63

5.17. AssignedNetwork . . . 64
5.18. AssignedNetworks . . . 64
5.19. AssignedPermissions . . . 65
5.20. AssignedRoles . . . 66
5.21. AssignedTag . . . 66
5.22. AssignedTags . . . 67
5.23. AssignedVnicProfile . . . 67
5.24. AssignedVnicProfiles . . . 68
5.25. AttachedStorageDomain . . . 68
5.26. AttachedStorageDomains . . . 70
5.27. Balance . . . 70
5.28. Balances . . . 71
5.29. Bookmark . . . 72
5.30. Bookmarks . . . 72
5.31. Cluster . . . 73
5.32. ClusterLevel . . . 74
5.33. ClusterLevels . . . 75
5.34. Clusters . . . 76
5.35. Copyable . . . 77
5.36. CpuProfile . . . 77
5.37. CpuProfiles . . . 78
5.38. DataCenter . . . 79
5.39. DataCenters . . . 80
5.40. Disk . . . 83
5.41. DiskAttachment . . . 85
5.42. DiskAttachments . . . 86
5.43. DiskProfile . . . 88
5.44. DiskProfiles . . . 88
5.45. DiskSnapshot . . . 89
5.46. DiskSnapshots . . . 90
5.47. Disks . . . 90
5.48. Domain . . . 91
5.49. DomainGroup . . . 92
5.50. DomainGroups . . . 92
5.51. DomainUser . . . 93
5.52. DomainUsers . . . 93
5.53. Domains . . . 94
5.54. EngineKatelloErrata . . . 94
5.55. Event . . . 94
5.56. Events . . . 95
5.57. ExternalComputeResource . . . 100

5.58. ExternalComputeResources . . . 100
5.59. ExternalDiscoveredHost . . . 100
5.60. ExternalDiscoveredHosts . . . 101
5.61. ExternalHost . . . 101
5.62. ExternalHostGroup . . . 102
5.63. ExternalHostGroups . . . 102
5.64. ExternalHostProvider . . . 102
5.65. ExternalHostProviders . . . 103
5.66. ExternalHosts . . . 104
5.67. ExternalProvider . . . 105
5.68. ExternalProviderCertificate . . . 105
5.69. ExternalProviderCertificates . . . 105
5.70. ExternalVmImports . . . 106
5.71. FenceAgent . . . 107
5.72. FenceAgents . . . 108
5.73. File . . . 108
5.74. Files . . . 109
5.75. Filter . . . 109
5.76. Filters . . . 110
5.77. GlusterBrick . . . 111
5.78. GlusterBricks . . . 111
5.79. GlusterHook . . . 113
5.80. GlusterHooks . . . 115
5.81. GlusterVolume . . . 115
5.82. GlusterVolumes . . . 118
5.83. GraphicsConsole . . . 119
5.84. GraphicsConsoles . . . 120
5.85. Group . . . 121
5.86. Groups . . . 121
5.87. Host . . . 122
5.88. HostDevice . . . 134
5.89. HostDevices . . . 134
5.90. HostHook . . . 135
5.91. HostHooks . . . 135
5.92. HostNic . . . 136
5.93. HostNics . . . 136
5.94. HostNumaNode . . . 137
5.95. HostNumaNodes . . . 137
5.96. HostStorage . . . 137
5.97. Hosts . . . 139
5.98. Icon . . . 141

5.99. Icons . . . 141
5.100. Image . . . 142
5.101. ImageTransfer . . . 142
5.102. ImageTransfers . . . 146
5.103. Images . . . 147
5.104. InstanceType . . . 147
5.105. InstanceTypeNic . . . 148
5.106. InstanceTypeNics . . . 148
5.107. InstanceTypeWatchdog . . . 149
5.108. InstanceTypeWatchdogs . . . 150
5.109. InstanceTypes . . . 150
5.110. IscsiBond . . . 151
5.111. IscsiBonds . . . 152
5.112. Job . . . 153
5.113. Jobs . . . 154
5.114. KatelloErrata . . . 155
5.115. KatelloErratum . . . 155
5.116. MacPool . . . 155
5.117. MacPools . . . 157
5.118. Measurable . . . 158
5.119. Moveable . . . 158
5.120. Network . . . 159
5.121. NetworkAttachment . . . 160
5.122. NetworkAttachments . . . 161
5.123. NetworkFilter . . . 162
5.124. NetworkFilters . . . 162
5.125. NetworkLabel . . . 164
5.126. NetworkLabels . . . 165
5.127. Networks . . . 166
5.128. OpenstackImage . . . 168
5.129. OpenstackImageProvider . . . 170
5.130. OpenstackImageProviders . . . 171
5.131. OpenstackImages . . . 171
5.132. OpenstackNetwork . . . 172
5.133. OpenstackNetworkProvider . . . 173
5.134. OpenstackNetworkProviders . . . 174
5.135. OpenstackNetworks . . . 175
5.136. OpenstackSubnet . . . 175
5.137. OpenstackSubnets . . . 176
5.138. OpenstackVolumeAuthenticationKey . . . 176
5.139. OpenstackVolumeAuthenticationKeys . . . 177

5.140. OpenstackVolumeProvider . . . 178
5.141. OpenstackVolumeProviders . . . 179
5.142. OpenstackVolumeType . . . 180
5.143. OpenstackVolumeTypes . . . 180
5.144. OperatingSystem . . . 181
5.145. OperatingSystems . . . 181
5.146. Permission . . . 182
5.147. Permit . . . 182
5.148. Permits . . . 183
5.149. Qos . . . 183
5.150. Qoss . . . 184
5.151. Quota . . . 185
5.152. QuotaClusterLimit . . . 186
5.153. QuotaClusterLimits . . . 187
5.154. QuotaStorageLimit . . . 187
5.155. QuotaStorageLimits . . . 188
5.156. Quotas . . . 189
5.157. Role . . . 189
5.158. Roles . . . 191
5.159. SchedulingPolicies . . . 192
5.160. SchedulingPolicy . . . 192
5.161. SchedulingPolicyUnit . . . 193
5.162. SchedulingPolicyUnits . . . 194
5.163. Snapshot . . . 194
5.164. SnapshotCdrom . . . 195
5.165. SnapshotCdroms . . . 196
5.166. SnapshotDisk . . . 196
5.167. SnapshotDisks . . . 197
5.168. SnapshotNic . . . 197
5.169. SnapshotNics . . . 197
5.170. Snapshots . . . 198
5.171. SshPublicKey . . . 199
5.172. SshPublicKeys . . . 199
5.173. Statistic . . . 200
5.174. Statistics . . . 200
5.175. Step . . . 201
5.176. Steps . . . 201
5.177. Storage . . . 202
5.178. StorageDomain . . . 203
5.179. StorageDomainContentDisk . . . 207
5.180. StorageDomainContentDisks . . . 207

5.181. StorageDomainServerConnection . . . 208
5.182. StorageDomainServerConnections . . . 208
5.183. StorageDomainTemplate . . . 209
5.184. StorageDomainTemplates . . . 210
5.185. StorageDomainVm . . . 211
5.186. StorageDomainVmDiskAttachment . . . 214
5.187. StorageDomainVmDiskAttachments . . . 214
5.188. StorageDomainVms . . . 215
5.189. StorageDomains . . . 216
5.190. StorageServerConnection . . . 218
5.191. StorageServerConnectionExtension . . . 219
5.192. StorageServerConnectionExtensions . . . 220
5.193. StorageServerConnections . . . 221
5.194. System . . . 223
5.195. SystemPermissions . . . 225
5.196. Tag . . . 225
5.197. Tags . . . 226
5.198. Template . . . 226
5.199. TemplateCdrom . . . 229
5.200. TemplateCdroms . . . 229
5.201. TemplateDisk . . . 229
5.202. TemplateDiskAttachment . . . 230
5.203. TemplateDiskAttachments . . . 231
5.204. TemplateDisks . . . 232
5.205. TemplateNic . . . 232
5.206. TemplateNics . . . 233
5.207. TemplateWatchdog . . . 233
5.208. TemplateWatchdogs . . . 234
5.209. Templates . . . 235
5.210. UnmanagedNetwork . . . 237
5.211. UnmanagedNetworks . . . 238
5.212. User . . . 238
5.213. Users . . . 239
5.214. VirtualFunctionAllowedNetwork . . . 240
5.215. VirtualFunctionAllowedNetworks . . . 241
5.216. Vm . . . 241
5.217. VmApplication . . . 253
5.218. VmApplications . . . 254
5.219. VmCdrom . . . 254
5.220. VmCdroms . . . 256
5.221. VmDisk . . . 257

5.222. VmDisks . . . 259
5.223. VmGraphicsConsole . . . 259
5.224. VmHostDevice . . . 260
5.225. VmHostDevices . . . 261
5.226. VmNic . . . 261
5.227. VmNics . . . 263
5.228. VmNumaNode . . . 265
5.229. VmNumaNodes . . . 266
5.230. VmPool . . . 268
5.231. VmPools . . . 269
5.232. VmReportedDevice . . . 271
5.233. VmReportedDevices . . . 271
5.234. VmSession . . . 271
5.235. VmSessions . . . 272
5.236. VmWatchdog . . . 273
5.237. VmWatchdogs . . . 273
5.238. Vms . . . 274
5.239. VnicProfile . . . 280
5.240. VnicProfiles . . . 282
5.241. Weight . . . 283
5.242. Weights . . . 283
6. Types . . . 284
6.1. AccessProtocol enum . . . 284
6.2. Action struct . . . 284
6.3. AffinityGroup struct . . . 286
6.4. AffinityLabel struct . . . 287
6.5. Agent struct . . . 288
6.6. AgentConfiguration struct . . . 288
6.7. Api struct . . . 289
6.8. ApiSummary struct . . . 289
6.9. ApiSummaryItem struct . . . 289
6.10. Application struct . . . 290
6.11. Architecture enum . . . 290
6.12. AuthorizedKey struct . . . 290
6.13. AutoNumaStatus enum . . . 291
6.14. Balance struct . . . 291
6.15. Bios struct . . . 291
6.16. BlockStatistic struct . . . 291
6.17. Bonding struct . . . 291
6.18. Bookmark struct . . . 292
6.19. Boot struct . . . 292

6.20. BootDevice enum . . . 292
6.21. BootMenu struct . . . 293
6.22. BootProtocol enum . . . 293
6.23. BrickProfileDetail struct . . . 293
6.24. Cdrom struct . . . 294
6.25. Certificate struct . . . 294
6.26. CloudInit struct . . . 295
6.27. Cluster struct . . . 295
6.28. ClusterLevel struct . . . 298
6.29. Configuration struct . . . 298
6.30. ConfigurationType enum . . . 300
6.31. Console struct . . . 300
6.32. Core struct . . . 301
6.33. Cpu struct . . . 301
6.34. CpuMode enum . . . 301
6.35. CpuProfile struct . . . 301
6.36. CpuTopology struct . . . 302
6.37. CpuTune struct . . . 302
6.38. CpuType struct . . . 302
6.39. CreationStatus enum . . . 302
6.40. CustomProperty struct . . . 303
6.41. DataCenter struct . . . 303
6.42. DataCenterStatus enum . . . 305
6.43. Device struct . . . 305
6.44. Disk struct . . . 305
6.45. DiskAttachment struct . . . 309
6.46. DiskFormat enum . . . 310
6.47. DiskInterface enum . . . 310
6.48. DiskProfile struct . . . 310
6.49. DiskSnapshot struct . . . 311
6.50. DiskStatus enum . . . 314
6.51. DiskStorageType enum . . . 314
6.52. DiskType enum . . . 314
6.53. Display struct . . . 314
6.54. DisplayType enum . . . 315
6.55. Dns struct . . . 315
6.56. Domain struct . . . 315
6.57. EntityExternalStatus enum . . . 316
6.58. EntityProfileDetail struct . . . 316
6.59. ErrorHandling struct . . . 316
6.60. Event struct . . . 316

6.61. ExternalComputeResource struct . . . 317
6.62. ExternalDiscoveredHost struct . . . 318
6.63. ExternalHost struct . . . 318
6.64. ExternalHostGroup struct . . . 318
6.65. ExternalHostProvider struct . . . 319
6.66. ExternalProvider struct . . . 320
6.67. ExternalStatus enum . . . 320
6.68. ExternalSystemType enum . . . 321
6.69. ExternalVmImport struct . . . 321
6.70. ExternalVmProviderType enum . . . 322
6.71. Fault struct . . . 322
6.72. FenceType enum . . . 323
6.73. FencingPolicy struct . . . 323
6.74. File struct . . . 323
6.75. Filter struct . . . 323
6.76. Floppy struct . . . 324
6.77. FopStatistic struct . . . 325
6.78. GlusterBrick struct . . . 325
6.79. GlusterBrickAdvancedDetails struct . . . 326
6.80. GlusterBrickMemoryInfo struct . . . 327
6.81. GlusterBrickStatus enum . . . 327
6.82. GlusterClient struct . . . 327
6.83. GlusterHook struct . . . 327
6.84. GlusterHookStatus enum . . . 328
6.85. GlusterMemoryPool struct . . . 328
6.86. GlusterServerHook struct . . . 329
6.87. GlusterState enum . . . 329
6.88. GlusterVolume struct . . . 329
6.89. GlusterVolumeProfileDetails struct . . . 330
6.90. GlusterVolumeStatus enum . . . 330
6.91. GlusterVolumeType enum . . . 331
6.92. GracePeriod struct . . . 331
6.93. GraphicsConsole struct . . . 331
6.94. GraphicsType enum . . . 332
6.95. Group struct . . . 332
6.96. GuestOperatingSystem struct . . . 333
6.97. HardwareInformation struct . . . 333
6.98. HighAvailability struct . . . 333
6.99. Hook struct . . . 334
6.100. HookContentType enum . . . 334
6.101. HookStage enum . . . 334

6.102. HookStatus enum . . . 334
6.103. Host struct . . . 335
6.104. HostDevice struct . . . 337
6.105. HostDevicePassthrough struct . . . 338
6.106. HostNic struct . . . 338
6.107. HostNicVirtualFunctionsConfiguration struct . . . 341
6.108. HostProtocol enum . . . 341
6.109. HostStatus enum . . . 341
6.110. HostStorage struct . . . 342
6.111. HostType enum . . . 343
6.112. HostedEngine struct . . . 344
6.113. Icon struct . . . 344
6.114. Identified struct . . . 344
6.115. Image struct . . . 344
6.116. ImageTransfer struct . . . 345
6.117. ImageTransferPhase enum . . . 346
6.118. InheritableBoolean enum . . . 347
6.119. Initialization struct . . . 347
6.120. InstanceType struct . . . 348
6.121. Io struct . . . 351
6.122. Ip struct . . . 352
6.123. IpAddressAssignment struct . . . 353
6.124. IpVersion enum . . . 353
6.125. IscsiBond struct . . . 353
6.126. IscsiDetails struct . . . 353
6.127. Job struct . . . 354
6.128. JobStatus enum . . . 355
6.129. KatelloErratum struct . . . 355
6.130. KdumpStatus enum . . . 356
6.131. Kernel struct . . . 356
6.132. Ksm struct . . . 356
6.133. LogSeverity enum . . . 356
6.134. LogicalUnit struct . . . 356
6.135. LunStatus enum . . . 357
6.136. Mac struct . . . 357
6.137. MacPool struct . . . 357
6.138. MemoryOverCommit struct . . . 358
6.139. MemoryPolicy struct . . . 359
6.140. MessageBrokerType enum . . . 359
6.141. Method struct . . . 359
6.142. MigrateOnError enum . . . 359

6.143. MigrationBandwidth struct . . . 359
6.144. MigrationBandwidthAssignmentMethod enum . . . 360
6.145. MigrationOptions struct . . . 360
6.146. MigrationPolicy struct . . . 361
6.147. Network struct . . . 361
6.148. NetworkAttachment struct . . . 363
6.149. NetworkConfiguration struct . . . 363
6.150. NetworkFilter struct . . . 364
6.151. NetworkLabel struct . . . 364
6.152. NetworkPluginType enum . . . 365
6.153. NetworkStatus enum . . . 365
6.154. NetworkUsage enum . . . 365
6.155. NfsProfileDetail struct . . . 366
6.156. NfsVersion enum . . . 366
6.157. Nic struct . . . 366
6.158. NicConfiguration struct . . . 368
6.159. NicInterface enum . . . 368
6.160. NicStatus enum . . . 368
6.161. NumaNode struct . . . 368
6.162. NumaNodePin struct . . . 369
6.163. NumaTuneMode enum . . . 370
6.164. OpenStackImage struct . . . 370
6.165. OpenStackImageProvider struct . . . 370
6.166. OpenStackNetwork struct . . . 371
6.167. OpenStackNetworkProvider struct . . . 371
6.168. OpenStackNetworkProviderType enum . . . 372
6.169. OpenStackProvider struct . . . 373
6.170. OpenStackSubnet struct . . . 373
6.171. OpenStackVolumeProvider struct . . . 374
6.172. OpenStackVolumeType struct . . . 375
6.173. OpenstackVolumeAuthenticationKey struct . . . 375
6.174. OpenstackVolumeAuthenticationKeyUsageType enum . . . 376
6.175. OperatingSystem struct . . . 376
6.176. OperatingSystemInfo struct . . . 377
6.177. Option struct . . . 377
6.178. OsType enum . . . 377
6.179. Package struct . . . 378
6.180. Payload struct . . . 378
6.181. PayloadEncoding enum . . . 378
6.182. Permission struct . . . 379
6.183. Permit struct . . . 379

6.184. PmProxy struct . . . 380
6.185. PmProxyType enum . . . 380
6.186. PolicyUnitType enum . . . 380
6.187. PortMirroring struct . . . 380
6.188. PowerManagement struct . . . 380
6.189. PowerManagementStatus enum . . . 381
6.190. Product struct . . . 382
6.191. ProductInfo struct . . . 382
6.192. ProfileDetail struct . . . 382
6.193. Property struct . . . 383
6.194. ProxyTicket struct . . . 383
6.195. Qos struct . . . 383
6.196. QosType enum . . . 384
6.197. Quota struct . . . 384
6.198. QuotaClusterLimit struct . . . 385
6.199. QuotaModeType enum . . . 386
6.200. QuotaStorageLimit struct . . . 386
6.201. Range struct . . . 386
6.202. Rate struct . . . 387
6.203. ReportedConfiguration struct . . . 387
6.204. ReportedDevice struct . . . 387
6.205. ReportedDeviceType enum . . . 388
6.206. ResolutionType enum . . . 388
6.207. RngDevice struct . . . 388
6.208. RngSource enum . . . 388
6.209. Role struct . . . 388
6.210. RoleType enum . . . 389
6.211. SchedulingPolicy struct . . . 389
6.212. SchedulingPolicyUnit struct . . . 389
6.213. ScsiGenericIO enum . . . 390
6.214. SeLinux struct . . . 390
6.215. SeLinuxMode enum . . . 390
6.216. SerialNumber struct . . . 390
6.217. SerialNumberPolicy enum . . . 391
6.218. Session struct . . . 391
6.219. SkipIfConnectivityBroken struct . . . 392
6.220. SkipIfSdActive struct . . . 392
6.221. Snapshot struct . . . 392
6.222. SnapshotStatus enum . . . 397
6.223. SnapshotType enum . . . 398
6.224. SpecialObjects struct . . . 398

6.225. Spm struct . . . 398
6.226. SpmStatus enum . . . 398
6.227. Ssh struct . . . 399
6.228. SshAuthenticationMethod enum . . . 399
6.229. SshPublicKey struct . . . 399
6.230. Sso struct . . . 400
6.231. SsoMethod enum . . . 400
6.232. Statistic struct . . . 400
6.233. StatisticKind enum . . . 401
6.234. StatisticUnit enum . . . 401
6.235. Step struct . . . 401
6.236. StepEnum enum . . . 402
6.237. StepStatus enum . . . 402
6.238. StorageConnection struct . . . 402
6.239. StorageConnectionExtension struct . . . 403
6.240. StorageDomain struct . . . 404
6.241. StorageDomainStatus enum . . . 405
6.242. StorageDomainType enum . . . 406
6.243. StorageFormat enum . . . 406
6.244. StorageType enum . . . 406
6.245. SwitchType enum . . . 407
6.246. Tag struct . . . 407
6.247. Template struct . . . 407
6.248. TemplateStatus enum . . . 410
6.249. TemplateVersion struct . . . 411
6.250. Ticket struct . . . 411
6.251. TimeZone struct . . . 411
6.252. TransparentHugePages struct . . . 411
6.253. TransportType enum . . . 412
6.254. UnmanagedNetwork struct . . . 412
6.255. Usb struct . . . 412
6.256. UsbType enum . . . 412
6.257. User struct . . . 412
6.258. Value struct . . . 414
6.259. ValueType enum . . . 414
6.260. VcpuPin struct . . . 414
6.261. Vendor struct . . . 414
6.262. Version struct . . . 415
6.263. VirtioScsi struct . . . 415
6.264. VirtualNumaNode struct . . . 415
6.265. Vlan struct . . . 417

6.266. Vm struct . . . 417
6.267. VmAffinity enum . . . 421
6.268. VmBase struct . . . 422
6.269. VmDeviceType enum . . . 424
6.270. VmPlacementPolicy struct . . . 425
6.271. VmPool struct . . . 425
6.272. VmPoolType enum . . . 426
6.273. VmStatus enum . . . 426
6.274. VmSummary struct . . . 427
6.275. VmType enum . . . 427
6.276. VnicPassThrough struct . . . 427
6.277. VnicPassThroughMode enum . . . 427
6.278. VnicProfile struct . . . 428
6.279. VolumeGroup struct . . . 428
6.280. Watchdog struct . . . 428
6.281. WatchdogAction enum . . . 429
6.282. WatchdogModel enum . . . 429
6.283. Weight struct . . . 429
Appendix A: Primitive types . . . 430
A.1. String primitive . . . 430
A.2. Boolean primitive . . . 430
A.3. Integer primitive . . . 430
A.4. Decimal primitive . . . 431
A.5. Date primitive . . . 431

1. Introduction

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

The benefits of the API are:

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

This provides developers and administrators with the ability to:

- Integrate with enterprise IT systems.
- Integrate with third-party virtualization software.
- Perform automated maintenance or error checking tasks.
- Automate repetitive tasks in an oVirt environment with scripts.

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

1.1. Representational State Transfer

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

1.2. API Prerequisites

Prerequisites for using the oVirt API:

- A networked installation of oVirt Engine, which includes the API.
- A client or programming library that initiates and receives HTTP requests from the API server. For example:

  - The oVirt Python SDK.
  - The oVirt Ruby SDK.
  - The oVirt Java SDK.
  - The curl command line tool.
  - RESTClient, a debugger for RESTful web services.

- Knowledge of Hypertext Transfer Protocol (HTTP), which is the protocol used for REST API interactions. The Internet Engineering Task Force provides a Request for Comments (RFC) explaining the Hypertext Transfer Protocol at http://www.ietf.org/rfc/rfc2616.txt.
- Knowledge of Extensible Markup Language (XML) or JavaScript Object Notation (JSON), which the API uses to construct resource representations. The W3C provides a full specification on XML at http://www.w3.org/TR/xml. ECMA International provide a free publication on JSON at http://www.ecma-international.org.

2. Authentication and security

2.1. TLS/SSL Certification

The oVirt API requires Hypertext Transfer Protocol Secure (HTTPS) [1: HTTPS is described in RFC 2818 HTTP Over TLS.] for secure interaction with client software, such as the SDK and CLI components. This involves a process of obtaining the certificate of the CA used by the server, and importing it into the certificate store of your client.

2.1.1. Obtaining the CA certificate

You can obtain the CA certificate from the oVirt Engine and transfer it to the client machine using one of these methods:

Method 1

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

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

That will connect to the server and will display an output similar to this:

```
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
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW5lNDEuZXhhbXBs
SVlJe7e5FTEtHJGTAeWWM6dGbsFhip5VXM0gfqg=
-----END CERTIFICATE-----
 1 s:/C=US/O=Example Inc./CN=myengine.example.com.23416
   i:/C=US/O=Example Inc./CN=myengine.example.com.23416
-----BEGIN CERTIFICATE-----
MIIDxjCCAq6gAwIBAgICEAAwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMCVVMx
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW5lNDEuZXhhbXBs
Pkyg1rQHR6ebGQ==
-----END CERTIFICATE-----
```

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

```
-----BEGIN CERTIFICATE-----
MIIDxjCCAq6gAwIBAgICEAAwDQYJKoZIhvcNAQEFBQAwSTELMAkGA1UEBhMCVVMx
FTATBgNVBAoTDEV4YW1wbGUgSW5jLjEjMCEGA1UEAxMaZW5naW5lNDEuZXhhbXBs
Pkyg1rQHR6ebGQ==
-----END CERTIFICATE-----
```

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

Method 2

If you can't use the `openssl s_client` method described above, you can instead use a command line tool to download the CA certificate from the engine.

Examples of command line tools include `curl` and `wget`, both of which are available on multiple platforms.

If using `curl`:

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

If using `wget`:

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

Method 3

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

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

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

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

Method 4

Log in to the engine, export the certificate from the truststore and copy it to your client machine.

1. Log in to the engine machine as the `root` user.
2. Export the certificate from the truststore using the Java `keytool` management utility:

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

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

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

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

Each of these methods results in a certificate file named `ca.crt` on your client machine. An API user imports this file into the certificate store of the client.

2.1.2. Importing a certificate to a client

Importing a certificate to a client relies on how the client itself stores and interprets certificates. This guide contains some examples on importing certificates. For clients not using Network Security Services (NSS) or Java KeyStore (JKS), see your client documentation for more information on importing a certificate.

2.2. Authentication

Any user with an oVirt Engine account has access to the API. All requests need to be authenticated using either *OAuth* or basic authentication, as described below.

2.2.1. OAuth authentication

Since version 4.0 of oVirt the preferred authentication mechanism is *OAuth 2.0*, as described in RFC 6749.

OAuth is a sophisticated protocol, with several mechanism to obtain authorization and access tokens. But for usage with the oVirt API the only supported one is the *resource owner password credentials grant*, as described in section 4.3 of that RFC.

The first step in the *OAuth* authentication process is to obtain a *token*, sending the user name and password to the single sign-on service of the engine:

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

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

Table 1. OAuth token request parameters

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

These parameters must be URL-encoded. For example the `@` character in the user name needs to be

encoded as `%40`. So the resulting request body will be something like this:

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

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

If the user name and password are valid the single sign-on service of the engine will respond with a JSON document similar to this one:

```
{
  "access_token": "fqbR1ftzh8wBCviLxJcYuV5oSDI=",
  "token_type": "bearer",
  "scope": "...",
  ...
}
```

For API authentication purposes the only relevant name/value pair is the `access_token`. Don't manipulate this in any way, use it exactly as provided by the SSO service.

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

```
GET /ovirt-engine/api/vms HTTP/1.1
Host: myengine.example.com
Accept: application/xml
Authorization: Bearer fqbR1ftzh8wBCviLxJcYuV5oSDI=
```

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

```
HTTP/1.1 401 Unauthorized
```

When this happens a new token is needed, and can be requested using the same procedure described above, as the single sign-on service of the engine doesn't currently support refresh tokens.

2.2.2. Basic authentication

IMPORTANT	Basic authentication is still supported only for backwards compatibility, but it is deprecated since version 4.0 of oVirt, and will be removed in the future.

Each request uses HTTP Basic Authentication [2: Basic Authentication is described in RFC 2617

HTTP Authentication: Basic and Digest Access Authentication.] to encode the credentials. If a request does not include an appropriate `Authorization` header, the server sends a `401 Authorization Required` as a result:

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

HTTP/1.1 401 Authorization Required
```

Request are issued with an `Authorization` header for the specified realm. An API user encodes an appropriate oVirt Engine domain and user in the supplied credentials with the `username@domain:password` convention.

The following table shows the process for encoding credentials in Base64.

Table 2. Encoding credentials for API access

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

An API user provides the Base64 encoded credentials as shown:

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

HTTP/1.1 200 OK
```

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

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

2.2.3. Authentication sessions

The API also provides the ability for authentication session support. An API user sends an initial

request with authentication details, then sends all subsequent requests using a session cookie to authenticate. The following procedure demonstrates how to use an authenticated session.

Requesting an authenticated session

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

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

HTTP/1.1 200 OK
...
```

   This returns a response with the following header:

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

   Note the `JSESSIONID=` value. In this example the value is `5dQja5ubr4yvI2MM2z+LZxrK`.

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

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

HTTP/1.1 200 OK
...
```

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

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

HTTP/1.1 200 OK
...
```

3. Quick start example

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

- A networked and configured oVirt installation;
- An ISO file containing a desired virtual machine operating system to install. This chapter uses CentOS 7 for our installation ISO example; and
- oVirt's `engine-iso-uploader` tool to upload your chosen operating system ISO file.

This example uses `curl` to demonstrate API requests with a client application. Note that any application capable of HTTP requests can substitute for `curl`.

IMPORTANT For simplicity, the HTTP request headers in this example omit the `Host` and `Authorization` headers. However, these fields are mandatory and require data specific to your installation of oVirt.

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

NOTE oVirt generates an unique identifier for the `id` attribute for each resource. Identifier codes in this example might appear different to the identifier codes in your oVirt environment.

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

3.1. Example: Access API entry point

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

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

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

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

Same request, using the `curl` command:

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

The result will be an object of type Api:

```
<api>
  <link href="/ovirt-engine/api/clusters" rel="clusters"/>
  <link href="/ovirt-engine/api/datacenters" rel="datacenters"/>
  ...
  <product_info>
    <name>oVirt Engine</name>
    <vendor>ovirt.org</vendor>
    <version>
      <build>0</build>
      <full_version>4.0.0-0.0.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>
```

IMPORTANT

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

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

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

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

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

3.2. Example: List data centers

oVirt creates a `Default` data center on installation. This example uses the `Default` data center as the basis for our virtual environment.

The following request retrieves a representation of the data centers:

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

Same request, using the `curl` command:

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

The result will be a list of objects of type DataCenter:

```
<data_centers>
  <data_center href="/ovirt-engine/api/datacenters/001" id="001">
    <name>Default</name>
    <description>The default Data Center</description>
    <link href="/ovirt-engine/api/datacenters/001/clusters" rel="clusters"/>
    <link href="/ovirt-engine/api/datacenters/001/storagedomains" rel=
"storagedomains"/>
    ...
    <local>false</local>
    <quota_mode>disabled</quota_mode>
    <status>up</status>
    <supported_versions>
      <version>
        <major>4</major>
        <minor>0</minor>
      </version>
    </supported_versions>
    <version>
      <major>4</major>
      <minor>0</minor>
    </version>
  </data_center>
  ...
</data_centers>
```

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

The data center also contains a link to the service that manages the storage domains attached to the data center:

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

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

3.3. Example: List host clusters

oVirt creates a `Default` hosts cluster on installation. This example uses the `Default` cluster to group resources in your oVirt environment.

The following request retrieves a representation of the cluster collection:

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

Same request, using the `curl` command:

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

The result will be a list of objects of type Cluster:

```
<clusters>
  <cluster href="/ovirt-engine/api/clusters/002" id="002">
    <name>Default</name>
    <description>The default server cluster</description>
    <link href="/ovirt-engine/api/clusters/002/networks" rel="networks"/>
    <link href="/ovirt-engine/api/clusters/002" rel="permissions"/>
    ...
    <cpu>
      <architecture>x86_64</architecture>
      <type>Intel Conroe Family</type>
    </cpu>
    <version>
      <major>4</major>
      <minor>0</minor>
    </version>
    <data_center href="/ovirt-engine/api/datacenters/001" id="001"/>
  </cluster>
  ...
</clusters>
```

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

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

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

The `networks` link is a reference to the service that manages the networks associated to this cluster. The next section examines the networks collection in more detail.

3.4. Example: List logical networks

oVirt creates a default `ovirtmgmt` network on installation. This network acts as the management network for oVirt Engine to access hosts.

This network is associated with our `Default` cluster and is a member of the `Default` data center. This

example uses the `ovirtmgmt` network to connect our virtual machines.

The following request retrieves the list of logical networks:

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

Same request, using the `curl` command:

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

The result will be a list of objects of type Network:

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

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

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

3.5. Example: List hosts

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

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

Same request, using the `curl` command:

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

The result will be a list of objects of type Host:

```
<hosts>
  <host href="/ovirt-engine/api/hosts/004" id="004">
    <name>myhost</name>
    <link href="/ovirt-engine/api/hosts/004/nics" rel="nics"/>
    ...
    <address>node40.example.com</address>
    <cpu>
      <name>Intel Core Processor (Haswell, no TSX)</name>
      <speed>3600</speed>
      <topology>
        <cores>1</cores>
        <sockets>2</sockets>
        <threads>1</threads>
      </topology>
    </cpu>
    <memory>8371830784</memory>
    <os>
      <type>RHEL</type>
      <version>
        <full_version>7 - 2.1511.el7.centos.2.10</full_version>
        <major>7</major>
      </version>
    </os>
    <port>54321</port>
    <status>up</status>
    <cluster href="/ovirt-engine/api/clusters/002" id="002"/>
  </host>
  ...
</hosts>
```

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

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

3.6. Example: Create NFS data storage

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

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

The request should be like this:

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

And the request body should be like this:

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

The same request, using the `curl` command:

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

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

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

3.7. Example: Create NFS ISO storage

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

The request should be like this:

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

And the request body should be like this:

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

The same request, using the `curl` command:

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

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

```
<storage_domain href="/ovirt-engine/api/storagedomains/006" id="006">
  <name>myiso</name>
  <description>My ISOs</description>
  <available>42949672960</available>
  <committed>0</committed>
  <master>false</master>
  <status>unattached</status>
  <storage>
    <address>mynfs.example.com</address>
    <path>/exports/myisos</path>
    <type>nfs</type>
  </storage>
  <storage_format>v1</storage_format>
  <type>iso</type>
  <used>9663676416</used>
</storage_domain>
```

3.8. Example: Attach storage domains to data center

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

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

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

With a request body like this:

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

Same request, using the `curl` command:

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

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

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

With a request body like this:

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

Same request, using the `curl` command:

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

3.9. Example: Create virtual machine

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

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

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

And the request body should be like this:

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

Same request, using the `curl` command:

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

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

```
<vm href="/ovirt-engine/api/vms/007" id="007">
  <name>myvm</name>
  <link href="/ovirt-engine/api/vms/007/diskattachments" rel="diskattachments"/>
  <link href="/ovirt-engine/api/vms/007/nics" rel="nics"/>
  ...
  <cpu>
    <architecture>x86_64</architecture>
    <topology>
      <cores>1</cores>
      <sockets>1</sockets>
      <threads>1</threads>
    </topology>
  </cpu>
  <memory>1073741824</memory>
  <os>
    <boot>
      <devices>
        <device>hd</device>
      </devices>
    </boot>
    <type>other</type>
  </os>
  <type>desktop</type>
  <cluster href="/ovirt-engine/api/clusters/002" id="002"/>
  <status>down</status>
  <original_template href="/ovirt-engine/api/templates/000" id="00"/>
  <template href="/ovirt-engine/api/templates/000" id="000"/>
</vm>
```

3.10. Example: Create a virtual machine NIC

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

The request should be like this:

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

The request body should contain an object of type Nic describing the NIC to be created:

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

Same request, using the `curl` command:

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

3.11. Example: Create virtual machine disk

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

The request should be like this:

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

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

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

Same request, using the `curl` command:

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

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

3.12. Example: Attach ISO image to virtual machine

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

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

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

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

Same request, using the `curl` command:

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

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

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

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

The request should be like this:

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

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

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

Same request, using the `curl` command:

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

For more details see the documentation of the service that manages virtual machine CD-ROMS.

3.13. Example: Start the virtual machine

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

The request should be like this:

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

The request body should be like this:

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

Same request, using the `curl` command:

```
# curl \
--cacert '/etc/pki/ovirt-engine/ca.pem' \
--user 'admin@internal:mypassword' \
--request POST \
--header 'Version: 4' \
--header 'Content-Type: application/xml' \
--header 'Accept: application/xml' \
--data '
<action>
  <vm>
    <os>
      <boot>
        <devices>
          <device>cdrom</device>
        </devices>
      </boot>
    </os>
  </vm>
</action>
' \
https://myengine.example.com/ovirt-engine/api/vms/007/start
```

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

4. Requests

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

- POST /affinitylabels
- GET /affinitylabels
- GET /affinitylabels/{label:id}
- PUT /affinitylabels/{label:id}
- DELETE /affinitylabels/{label:id}
- POST /affinitylabels/{label:id}/hosts
- GET /affinitylabels/{label:id}/hosts
- DELETE /affinitylabels/{label:id}/hosts/{host:id}
- GET /affinitylabels/{label:id}/hosts/{host:id}
- POST /affinitylabels/{label:id}/vms
- GET /affinitylabels/{label:id}/vms
- DELETE /affinitylabels/{label:id}/vms/{vm:id}
- GET /affinitylabels/{label:id}/vms/{vm:id}

- POST /bookmarks
- GET /bookmarks
- GET /bookmarks/{bookmark:id}
- PUT /bookmarks/{bookmark:id}
- DELETE /bookmarks/{bookmark:id}
- GET /clusterlevels
- GET /clusterlevels/{level:id}
- POST /clusters
- GET /clusters
- GET /clusters/{cluster:id}
- PUT /clusters/{cluster:id}
- DELETE /clusters/{cluster:id}
- POST /clusters/{cluster:id}/affinitygroups
- GET /clusters/{cluster:id}/affinitygroups
- GET /clusters/{cluster:id}/affinitygroups/{group:id}
- PUT /clusters/{cluster:id}/affinitygroups/{group:id}
- DELETE /clusters/{cluster:id}/affinitygroups/{group:id}
- POST /clusters/{cluster:id}/affinitygroups/{group:id}/vms
- GET /clusters/{cluster:id}/affinitygroups/{group:id}/vms
- DELETE /clusters/{cluster:id}/affinitygroups/{group:id}/vms/{vm:id}
- POST /clusters/{cluster:id}/cpuprofiles
- GET /clusters/{cluster:id}/cpuprofiles
- GET /clusters/{cluster:id}/cpuprofiles/{profile:id}
- DELETE /clusters/{cluster:id}/cpuprofiles/{profile:id}
- GET /clusters/{cluster:id}/glusterhooks
- GET /clusters/{cluster:id}/glusterhooks/{hook:id}
- DELETE /clusters/{cluster:id}/glusterhooks/{hook:id}
- POST /clusters/{cluster:id}/glusterhooks/{hook:id}/disable
- POST /clusters/{cluster:id}/glusterhooks/{hook:id}/enable
- POST /clusters/{cluster:id}/glusterhooks/{hook:id}/resolve
- POST /clusters/{cluster:id}/glustervolumes
- GET /clusters/{cluster:id}/glustervolumes
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}
- DELETE /clusters/{cluster:id}/glustervolumes/{volume:id}
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/getprofilestatistics

- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- DELETE /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/activate
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/migrate
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/stopmigrate
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}
- DELETE /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/replace
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/statistics
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/statistics/{statistic:id}
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/rebalance
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/resetalloptions
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/resetoption
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/setoption
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/start
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/startprofile
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}/statistics
- GET /clusters/{cluster:id}/glustervolumes/{volume:id}/statistics/{statistic:id}
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/stop
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/stopprofile
- POST /clusters/{cluster:id}/glustervolumes/{volume:id}/stoprebalance
- GET /clusters/{cluster:id}/networkfilters
- GET /clusters/{cluster:id}/networkfilters/{networkfilter:id}
- POST /clusters/{cluster:id}/networks
- GET /clusters/{cluster:id}/networks
- GET /clusters/{cluster:id}/networks/{network:id}
- DELETE /clusters/{cluster:id}/networks/{network:id}
- PUT /clusters/{cluster:id}/networks/{network:id}
- POST /clusters/{cluster:id}/permissions
- GET /clusters/{cluster:id}/permissions
- GET /clusters/{cluster:id}/permissions/{permission:id}
- DELETE /clusters/{cluster:id}/permissions/{permission:id}
- POST /clusters/{cluster:id}/resetemulatedmachine

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

- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glusterhooks/{hook:id}/enable
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glusterhooks/{hook:id}/resolve
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/getprofilestatistics
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/activate
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/migrate
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/stopmigrate
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}
- DELETE /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}
- POST /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/replace
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/statistics
- GET /datacenters/{datacenter:id}/clusters/{cluster:id}/glustervolumes/{volume:id}/glusterbricks/{brick:id}/statistics/{statistic: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:id}

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

- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/permissions/{permission:id}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/permissions/{permission:id}
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}/permissions
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}/permissions
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}/permissions/{permission:id}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/networks/{network:id}/vnicprofiles/{profile:id}/permissions/{permission:id}
- POST /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections
- GET /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections/{storageconnection:id}
- PUT /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections/{storageconnection:id}
- DELETE /datacenters/{datacenter:id}/iscsibonds/{iscsibond:id}/storageserverconnections/{storageconnection:id}
- POST /datacenters/{datacenter:id}/networks
- GET /datacenters/{datacenter:id}/networks
- GET /datacenters/{datacenter:id}/networks/{network:id}
- PUT /datacenters/{datacenter:id}/networks/{network:id}
- DELETE /datacenters/{datacenter:id}/networks/{network:id}
- POST /datacenters/{datacenter:id}/networks/{network:id}/networklabels
- GET /datacenters/{datacenter:id}/networks/{network:id}/networklabels
- GET /datacenters/{datacenter:id}/networks/{network:id}/networklabels/{label:id}
- DELETE /datacenters/{datacenter:id}/networks/{network:id}/networklabels/{label:id}
- POST /datacenters/{datacenter:id}/networks/{network:id}/permissions
- GET /datacenters/{datacenter:id}/networks/{network:id}/permissions
- GET /datacenters/{datacenter:id}/networks/{network:id}/permissions/{permission:id}

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

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

- GET /disks
- GET /disks/{disk:id}
- DELETE /disks/{disk:id}
- POST /disks/{disk:id}/copy
- POST /disks/{disk:id}/export
- POST /disks/{disk:id}/move
- POST /disks/{disk:id}/permissions
- GET /disks/{disk:id}/permissions
- GET /disks/{disk:id}/permissions/{permission:id}
- DELETE /disks/{disk:id}/permissions/{permission:id}
- GET /disks/{disk:id}/statistics
- GET /disks/{disk:id}/statistics/{statistic:id}
- GET /domains
- GET /domains/{domain:id}
- GET /domains/{domain:id}/groups
- GET /domains/{domain:id}/groups/{group:id}
- GET /domains/{domain:id}/users
- GET /domains/{domain:id}/users/{user:id}
- POST /events
- GET /events
- POST /events/undelete
- GET /events/{event:id}
- DELETE /events/{event:id}
- POST /externalhostproviders
- GET /externalhostproviders
- GET /externalhostproviders/{provider:id}
- PUT /externalhostproviders/{provider:id}
- DELETE /externalhostproviders/{provider:id}
- GET /externalhostproviders/{provider:id}/certificates
- GET /externalhostproviders/{provider:id}/certificates/{certificate:id}
- GET /externalhostproviders/{provider:id}/computeresources
- GET /externalhostproviders/{provider:id}/computeresources/{resource:id}
- GET /externalhostproviders/{provider:id}/discoveredhosts
- GET /externalhostproviders/{provider:id}/discoveredhosts/{host:id}
- GET /externalhostproviders/{provider:id}/hostgroups

- GET /externalhostproviders/{provider:id}/hostgroups/{group:id}
- GET /externalhostproviders/{provider:id}/hosts
- GET /externalhostproviders/{provider:id}/hosts/{host:id}
- POST /externalhostproviders/{provider:id}/importcertificates
- POST /externalhostproviders/{provider:id}/testconnectivity
- POST /externalvmimports
- POST /groups
- GET /groups
- GET /groups/{group:id}
- DELETE /groups/{group:id}
- POST /groups/{group:id}/permissions
- GET /groups/{group:id}/permissions
- GET /groups/{group:id}/permissions/{permission:id}
- DELETE /groups/{group:id}/permissions/{permission:id}
- GET /groups/{group:id}/roles
- GET /groups/{group:id}/roles/{role:id}
- DELETE /groups/{group:id}/roles/{role:id}
- PUT /groups/{group:id}/roles/{role:id}
- POST /groups/{group:id}/roles/{role:id}/permits
- GET /groups/{group:id}/roles/{role:id}/permits
- GET /groups/{group:id}/roles/{role:id}/permits/{permit:id}
- DELETE /groups/{group:id}/roles/{role:id}/permits/{permit:id}
- POST /groups/{group:id}/tags
- GET /groups/{group:id}/tags
- GET /groups/{group:id}/tags/{tag:id}
- DELETE /groups/{group:id}/tags/{tag:id}
- POST /hosts
- GET /hosts
- GET /hosts/{host:id}
- PUT /hosts/{host:id}
- DELETE /hosts/{host:id}
- POST /hosts/{host:id}/activate
- POST /hosts/{host:id}/affinitylabels
- GET /hosts/{host:id}/affinitylabels
- GET /hosts/{host:id}/affinitylabels/{label:id}

- DELETE /hosts/{host:id}/affinitylabels/{label:id}
- POST /hosts/{host:id}/approve
- POST /hosts/{host:id}/commitnetconfig
- POST /hosts/{host:id}/deactivate
- GET /hosts/{host:id}/devices
- GET /hosts/{host:id}/devices/{device:id}
- POST /hosts/{host:id}/enrollcertificate
- POST /hosts/{host:id}/fence
- POST /hosts/{host:id}/fenceagents
- GET /hosts/{host:id}/fenceagents
- GET /hosts/{host:id}/fenceagents/{agent:id}
- PUT /hosts/{host:id}/fenceagents/{agent:id}
- DELETE /hosts/{host:id}/fenceagents/{agent:id}
- POST /hosts/{host:id}/forceselectspm
- GET /hosts/{host:id}/hooks
- GET /hosts/{host:id}/hooks/{hook:id}
- POST /hosts/{host:id}/install
- POST /hosts/{host:id}/iscsidiscover
- POST /hosts/{host:id}/iscsilogin
- GET /hosts/{host:id}/katelloerrata
- GET /hosts/{host:id}/katelloerrata/{katelloerratum:id}
- POST /hosts/{host:id}/networkattachments
- GET /hosts/{host:id}/networkattachments
- GET /hosts/{host:id}/networkattachments/{attachment:id}
- PUT /hosts/{host:id}/networkattachments/{attachment:id}
- DELETE /hosts/{host:id}/networkattachments/{attachment:id}
- GET /hosts/{host:id}/nics
- GET /hosts/{host:id}/nics/{nic:id}
- POST /hosts/{host:id}/nics/{nic:id}/networkattachments
- GET /hosts/{host:id}/nics/{nic:id}/networkattachments
- GET /hosts/{host:id}/nics/{nic:id}/networkattachments/{attachment:id}
- PUT /hosts/{host:id}/nics/{nic:id}/networkattachments/{attachment:id}
- DELETE /hosts/{host:id}/nics/{nic:id}/networkattachments/{attachment:id}
- POST /hosts/{host:id}/nics/{nic:id}/networklabels
- GET /hosts/{host:id}/nics/{nic:id}/networklabels

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

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

- GET /jobs
- GET /jobs/{job:id}
- POST /jobs/{job:id}/clear
- POST /jobs/{job:id}/end
- POST /jobs/{job:id}/steps
- GET /jobs/{job:id}/steps
- GET /jobs/{job:id}/steps/{step:id}
- POST /jobs/{job:id}/steps/{step:id}/end
- GET /jobs/{job:id}/steps/{step:id}/statistics
- GET /jobs/{job:id}/steps/{step:id}/statistics/{statistic:id}
- GET /katelloerrata
- GET /katelloerrata/{katelloerratum:id}
- POST /macpools
- GET /macpools
- GET /macpools/{macpool:id}
- PUT /macpools/{macpool:id}
- DELETE /macpools/{macpool:id}
- GET /networkfilters
- GET /networkfilters/{networkfilter:id}
- POST /networks
- GET /networks
- GET /networks/{network:id}
- PUT /networks/{network:id}
- DELETE /networks/{network:id}
- POST /networks/{network:id}/networklabels
- GET /networks/{network:id}/networklabels
- GET /networks/{network:id}/networklabels/{label:id}
- DELETE /networks/{network:id}/networklabels/{label:id}
- POST /networks/{network:id}/permissions
- GET /networks/{network:id}/permissions
- GET /networks/{network:id}/permissions/{permission:id}
- DELETE /networks/{network:id}/permissions/{permission:id}
- POST /networks/{network:id}/vnicprofiles
- GET /networks/{network:id}/vnicprofiles
- GET /networks/{network:id}/vnicprofiles/{profile:id}

- DELETE /networks/{network:id}/vnicprofiles/{profile:id}
- POST /networks/{network:id}/vnicprofiles/{profile:id}/permissions
- GET /networks/{network:id}/vnicprofiles/{profile:id}/permissions
- GET /networks/{network:id}/vnicprofiles/{profile:id}/permissions/{permission:id}
- DELETE /networks/{network:id}/vnicprofiles/{profile:id}/permissions/{permission:id}
- POST /openstackimageproviders
- GET /openstackimageproviders
- GET /openstackimageproviders/{provider:id}
- PUT /openstackimageproviders/{provider:id}
- DELETE /openstackimageproviders/{provider:id}
- GET /openstackimageproviders/{provider:id}/certificates
- GET /openstackimageproviders/{provider:id}/certificates/{certificate:id}
- GET /openstackimageproviders/{provider:id}/images
- GET /openstackimageproviders/{provider:id}/images/{image:id}
- POST /openstackimageproviders/{provider:id}/images/{image:id}/import
- POST /openstackimageproviders/{provider:id}/importcertificates
- POST /openstackimageproviders/{provider:id}/testconnectivity
- POST /openstacknetworkproviders
- GET /openstacknetworkproviders
- GET /openstacknetworkproviders/{provider:id}
- PUT /openstacknetworkproviders/{provider:id}
- DELETE /openstacknetworkproviders/{provider:id}
- GET /openstacknetworkproviders/{provider:id}/certificates
- GET /openstacknetworkproviders/{provider:id}/certificates/{certificate:id}
- POST /openstacknetworkproviders/{provider:id}/importcertificates
- GET /openstacknetworkproviders/{provider:id}/networks
- GET /openstacknetworkproviders/{provider:id}/networks/{network:id}
- POST /openstacknetworkproviders/{provider:id}/networks/{network:id}/import
- POST /openstacknetworkproviders/{provider:id}/networks/{network:id}/subnets
- GET /openstacknetworkproviders/{provider:id}/networks/{network:id}/subnets
- GET /openstacknetworkproviders/{provider:id}/networks/{network:id}/subnets/{subnet:id}
- DELETE /openstacknetworkproviders/{provider:id}/networks/{network:id}/subnets/{subnet:id}
- POST /openstacknetworkproviders/{provider:id}/testconnectivity
- POST /openstackvolumeproviders
- GET /openstackvolumeproviders

- GET /openstackvolumeproviders/{provider:id}
- PUT /openstackvolumeproviders/{provider:id}
- DELETE /openstackvolumeproviders/{provider:id}
- POST /openstackvolumeproviders/{provider:id}/authenticationkeys
- GET /openstackvolumeproviders/{provider:id}/authenticationkeys
- GET /openstackvolumeproviders/{provider:id}/authenticationkeys/{key:id}
- PUT /openstackvolumeproviders/{provider:id}/authenticationkeys/{key:id}
- DELETE /openstackvolumeproviders/{provider:id}/authenticationkeys/{key:id}
- GET /openstackvolumeproviders/{provider:id}/certificates
- GET /openstackvolumeproviders/{provider:id}/certificates/{certificate:id}
- POST /openstackvolumeproviders/{provider:id}/importcertificates
- POST /openstackvolumeproviders/{provider:id}/testconnectivity
- GET /openstackvolumeproviders/{provider:id}/volumetypes
- GET /openstackvolumeproviders/{provider:id}/volumetypes/{type:id}
- GET /operatingsystems
- GET /operatingsystems/{operatingsystem:id}
- POST /permissions
- GET /permissions
- GET /permissions/{permission:id}
- DELETE /permissions/{permission:id}
- POST /roles
- GET /roles
- GET /roles/{role:id}
- DELETE /roles/{role:id}
- PUT /roles/{role:id}
- POST /roles/{role:id}/permits
- GET /roles/{role:id}/permits
- GET /roles/{role:id}/permits/{permit:id}
- DELETE /roles/{role:id}/permits/{permit:id}
- POST /schedulingpolicies
- GET /schedulingpolicies
- GET /schedulingpolicies/{policy:id}
- PUT /schedulingpolicies/{policy:id}
- DELETE /schedulingpolicies/{policy:id}
- POST /schedulingpolicies/{policy:id}/balances

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

- POST /storagedomains/{storagedomain:id}/disks/{disk:id}/permissions
- GET /storagedomains/{storagedomain:id}/disks/{disk:id}/permissions
- GET /storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permission:id}
- DELETE /storagedomains/{storagedomain:id}/disks/{disk:id}/permissions/{permission:id}
- 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}/refreshluns
- POST /storagedomains/{storagedomain:id}/storageconnections
- GET /storagedomains/{storagedomain:id}/storageconnections
- GET /storagedomains/{storagedomain:id}/storageconnections/{connection:id}
- DELETE /storagedomains/{storagedomain:id}/storageconnections/{connection:id}
- GET /storagedomains/{storagedomain:id}/templates
- GET /storagedomains/{storagedomain:id}/templates/{template:id}
- DELETE /storagedomains/{storagedomain:id}/templates/{template:id}
- GET /storagedomains/{storagedomain:id}/templates/{template:id}/disks
- GET /storagedomains/{storagedomain:id}/templates/{template:id}/disks/{disk:id}
- POST /storagedomains/{storagedomain:id}/templates/{template:id}/import
- POST /storagedomains/{storagedomain:id}/templates/{template:id}/register
- POST /storagedomains/{storagedomain:id}/updateovfstore
- GET /storagedomains/{storagedomain:id}/vms
- GET /storagedomains/{storagedomain:id}/vms/{vm:id}
- DELETE /storagedomains/{storagedomain:id}/vms/{vm:id}

- GET /storagedomains/{storagedomain:id}/vms/{vm:id}/diskattachments
- GET /storagedomains/{storagedomain:id}/vms/{vm:id}/diskattachments/{attachment:id}
- GET /storagedomains/{storagedomain:id}/vms/{vm:id}/disks
- GET /storagedomains/{storagedomain:id}/vms/{vm:id}/disks/{disk:id}
- POST /storagedomains/{storagedomain:id}/vms/{vm:id}/import
- POST /storagedomains/{storagedomain:id}/vms/{vm:id}/register
- POST /tags
- GET /tags
- GET /tags/{tag:id}
- PUT /tags/{tag:id}
- DELETE /tags/{tag:id}
- POST /templates
- GET /templates
- GET /templates/{template:id}
- PUT /templates/{template:id}
- DELETE /templates/{template:id}
- GET /templates/{template:id}/cdroms
- GET /templates/{template:id}/cdroms/{cdrom:id}
- GET /templates/{template:id}/diskattachments
- GET /templates/{template:id}/diskattachments/{attachment:id}
- DELETE /templates/{template:id}/diskattachments/{attachment:id}
- POST /templates/{template:id}/export
- POST /templates/{template:id}/graphicsconsoles
- GET /templates/{template:id}/graphicsconsoles
- GET /templates/{template:id}/graphicsconsoles/{console:id}
- DELETE /templates/{template:id}/graphicsconsoles/{console:id}
- POST /templates/{template:id}/nics
- GET /templates/{template:id}/nics
- GET /templates/{template:id}/nics/{nic:id}
- PUT /templates/{template:id}/nics/{nic:id}
- DELETE /templates/{template:id}/nics/{nic:id}
- POST /templates/{template:id}/permissions
- GET /templates/{template:id}/permissions
- GET /templates/{template:id}/permissions/{permission:id}
- DELETE /templates/{template:id}/permissions/{permission:id}

- POST /templates/{template:id}/tags
- GET /templates/{template:id}/tags
- GET /templates/{template:id}/tags/{tag:id}
- DELETE /templates/{template:id}/tags/{tag:id}
- POST /templates/{template:id}/watchdogs
- GET /templates/{template:id}/watchdogs
- GET /templates/{template:id}/watchdogs/{watchdog:id}
- PUT /templates/{template:id}/watchdogs/{watchdog:id}
- DELETE /templates/{template:id}/watchdogs/{watchdog:id}
- POST /users
- GET /users
- GET /users/{user:id}
- DELETE /users/{user:id}
- POST /users/{user:id}/permissions
- GET /users/{user:id}/permissions
- GET /users/{user:id}/permissions/{permission:id}
- DELETE /users/{user:id}/permissions/{permission:id}
- GET /users/{user:id}/roles
- GET /users/{user:id}/roles/{role:id}
- DELETE /users/{user:id}/roles/{role:id}
- PUT /users/{user:id}/roles/{role:id}
- POST /users/{user:id}/roles/{role:id}/permits
- GET /users/{user:id}/roles/{role:id}/permits
- GET /users/{user:id}/roles/{role:id}/permits/{permit:id}
- DELETE /users/{user:id}/roles/{role:id}/permits/{permit:id}
- POST /users/{user:id}/sshpublickeys
- GET /users/{user:id}/sshpublickeys
- GET /users/{user:id}/sshpublickeys/{key:id}
- PUT /users/{user:id}/sshpublickeys/{key:id}
- DELETE /users/{user:id}/sshpublickeys/{key:id}
- POST /users/{user:id}/tags
- GET /users/{user:id}/tags
- GET /users/{user:id}/tags/{tag:id}
- DELETE /users/{user:id}/tags/{tag:id}
- POST /vmpools

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

- GET /vms/{vm:id}/graphicsconsoles
- GET /vms/{vm:id}/graphicsconsoles/{console:id}
- DELETE /vms/{vm:id}/graphicsconsoles/{console:id}
- POST /vms/{vm:id}/hostdevices
- GET /vms/{vm:id}/hostdevices
- GET /vms/{vm:id}/hostdevices/{device:id}
- DELETE /vms/{vm:id}/hostdevices/{device:id}
- GET /vms/{vm:id}/katelloerrata
- GET /vms/{vm:id}/katelloerrata/{katelloerratum:id}
- POST /vms/{vm:id}/logon
- POST /vms/{vm:id}/maintenance
- POST /vms/{vm:id}/migrate
- POST /vms/{vm:id}/nics
- GET /vms/{vm:id}/nics
- GET /vms/{vm:id}/nics/{nic:id}
- PUT /vms/{vm:id}/nics/{nic:id}
- DELETE /vms/{vm:id}/nics/{nic:id}
- POST /vms/{vm:id}/nics/{nic:id}/activate
- POST /vms/{vm:id}/nics/{nic:id}/deactivate
- GET /vms/{vm:id}/nics/{nic:id}/reporteddevices
- GET /vms/{vm:id}/nics/{nic:id}/reporteddevices/{reporteddevice:id}
- GET /vms/{vm:id}/nics/{nic:id}/statistics
- GET /vms/{vm:id}/nics/{nic:id}/statistics/{statistic:id}
- POST /vms/{vm:id}/numanodes
- GET /vms/{vm:id}/numanodes
- GET /vms/{vm:id}/numanodes/{node:id}
- PUT /vms/{vm:id}/numanodes/{node:id}
- DELETE /vms/{vm:id}/numanodes/{node:id}
- POST /vms/{vm:id}/permissions
- GET /vms/{vm:id}/permissions
- GET /vms/{vm:id}/permissions/{permission:id}
- DELETE /vms/{vm:id}/permissions/{permission:id}
- POST /vms/{vm:id}/previewsnapshot
- POST /vms/{vm:id}/reboot
- POST /vms/{vm:id}/reordermacaddresses

- GET /vms/{vm:id}/reporteddevices
- GET /vms/{vm:id}/reporteddevices/{reporteddevice:id}
- GET /vms/{vm:id}/sessions
- GET /vms/{vm:id}/sessions/{session:id}
- POST /vms/{vm:id}/shutdown
- POST /vms/{vm:id}/snapshots
- GET /vms/{vm:id}/snapshots
- GET /vms/{vm:id}/snapshots/{snapshot:id}
- DELETE /vms/{vm:id}/snapshots/{snapshot:id}
- GET /vms/{vm:id}/snapshots/{snapshot:id}/cdroms
- GET /vms/{vm:id}/snapshots/{snapshot:id}/cdroms/{cdrom:id}
- GET /vms/{vm:id}/snapshots/{snapshot:id}/disks
- GET /vms/{vm:id}/snapshots/{snapshot:id}/disks/{disk:id}
- GET /vms/{vm:id}/snapshots/{snapshot:id}/nics
- GET /vms/{vm:id}/snapshots/{snapshot:id}/nics/{nic:id}
- POST /vms/{vm:id}/snapshots/{snapshot:id}/restore
- POST /vms/{vm:id}/start
- GET /vms/{vm:id}/statistics
- GET /vms/{vm:id}/statistics/{statistic:id}
- POST /vms/{vm:id}/stop
- POST /vms/{vm:id}/suspend
- POST /vms/{vm:id}/tags
- GET /vms/{vm:id}/tags
- GET /vms/{vm:id}/tags/{tag:id}
- DELETE /vms/{vm:id}/tags/{tag:id}
- POST /vms/{vm:id}/thawfilesystems
- POST /vms/{vm:id}/ticket
- POST /vms/{vm:id}/undosnapshot
- POST /vms/{vm:id}/watchdogs
- GET /vms/{vm:id}/watchdogs
- GET /vms/{vm:id}/watchdogs/{watchdog:id}
- PUT /vms/{vm:id}/watchdogs/{watchdog:id}
- DELETE /vms/{vm:id}/watchdogs/{watchdog:id}
- POST /vnicprofiles
- GET /vnicprofiles

- GET /vnicprofiles/{profile:id}
- PUT /vnicprofiles/{profile:id}
- DELETE /vnicprofiles/{profile:id}
- POST /vnicprofiles/{profile:id}/permissions
- GET /vnicprofiles/{profile:id}/permissions
- GET /vnicprofiles/{profile:id}/permissions/{permission:id}
- DELETE /vnicprofiles/{profile:id}/permissions/{permission:id}

5. Services

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

5.1. AffinityGroup

This service manages a single affinity group.

Table 3. Methods summary

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

5.1.1. get GET

Retrieve the affinity group details.

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

Table 4. Parameters summary

Name	Type	Direction	Summary
`group`	AffinityGroup	Out	The affinity group.

5.1.2. remove DELETE

Remove the affinity group.

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

Table 5. Parameters summary

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

5.1.3. update PUT

Update the affinity group.

Table 6. Parameters summary

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

5.2. AffinityGroupVm

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

Table 7. Methods summary

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

5.2.1. remove DELETE

Remove this virtual machine from the affinity group.

Table 8. Parameters summary

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

5.3. AffinityGroupVms

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

Table 9. Methods summary

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

5.3.1. add POST

Add a virtual machine to the affinity group.

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

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

With the following body:

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

Table 10. Parameters summary

Name	Type	Direction	Summary
`vm`	Vm	In/Out	

5.3.2. list GET

List all virtual machines assigned to this affinity group.

Table 11. Parameters summary

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

max

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

5.4. AffinityGroups

Affinity groups service manages virtual machine relationships and dependencies.

Table 12. Methods summary

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

5.4.1. add POST

Create a new affinity group.

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

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

And use the following example in its body:

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

Table 13. Parameters summary

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

5.4.2. list GET

List existing affinity groups.

Table 14. Parameters summary

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

max

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

5.5. AffinityLabel

Single affinity label details.

Table 15. Methods summary

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

5.5.1. get GET

Retrieves details about a label.

Table 16. Parameters summary

Name	Type	Direction	Summary
`label`	AffinityLabel	Out	

5.5.2. remove DELETE

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

5.5.3. update PUT

Updates a label.

This call will update all metadata like name or description.

Table 17. Parameters summary

Name	Type	Direction	Summary
`label`	AffinityLabel	In/Out	

5.6. AffinityLabelHost

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

Table 18. Methods summary

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

5.6.1. get GET

Retrieves details about a host that has this label assigned.

Table 19. Parameters summary

Name	Type	Direction	Summary
`host`	Host	Out	

5.6.2. remove DELETE

Remove a label from a host.

5.7. AffinityLabelHosts

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

Table 20. Methods summary

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

5.7.1. add POST

Add a label to a host.

Table 21. Parameters summary

Name	Type	Direction	Summary
`host`	Host	In/Out	

5.7.2. list GET

List all hosts with the label.

Table 22. Parameters summary

Name	Type	Direction	Summary
`hosts`	Host[]	Out	

5.8. AffinityLabelVm

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

Table 23. Methods summary

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

5.8.1. get GET

Retrieves details about a vm that has this label assigned.

Table 24. Parameters summary

Name	Type	Direction	Summary
`vm`	Vm	Out	

5.8.2. remove DELETE

Remove a label from a vm.

5.9. AffinityLabelVms

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

Table 25. Methods summary

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

5.9.1. add POST

Add a label to a vm.

Table 26. Parameters summary

Name	Type	Direction	Summary
`vm`	Vm	In/Out	

5.9.2. list GET

List all vms with the label.

Table 27. Parameters summary

Name	Type	Direction	Summary
`vms`	Vm[]	Out	

5.10. AffinityLabels

Manages the affinity labels available in the system.

Table 28. Methods summary

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

5.10.1. add POST

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

Table 29. Parameters summary

Name	Type	Direction	Summary
`label`	AffinityLabel	In/Out	

5.10.2. list GET

Lists all labels present in the system.

Table 30. Parameters summary

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

max

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

5.11. AssignedAffinityLabel

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

Table 31. Methods summary

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

5.11.1. get GET

Retrieves details about the attached label.

Table 32. Parameters summary

Name	Type	Direction	Summary
`label`	AffinityLabel	Out	

5.11.2. remove DELETE

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

5.12. AssignedAffinityLabels

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

Table 33. Methods summary

Name	Summary
`add`	Attaches a label to an entity.
`list`	Lists all labels that are attached to an entity.

5.12.1. add POST

Attaches a label to an entity.

Table 34. Parameters summary

Name	Type	Direction	Summary
`label`	AffinityLabel	In/Out	

5.12.2. list GET

Lists all labels that are attached to an entity.

Table 35. Parameters summary

Name	Type	Direction	Summary
`label`	AffinityLabel[]	Out	

5.13. AssignedCpuProfile

Table 36. Methods summary

Name	Summary
`get`	
`remove`	

5.13.1. get GET

Table 37. Parameters summary

Name	Type	Direction	Summary
`profile`	CpuProfile	Out	

5.13.2. remove DELETE

Table 38. Parameters summary

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

5.14. AssignedCpuProfiles

Table 39. Methods summary

Name	Summary
`add`	
`list`	

5.14.1. add POST

Table 40. Parameters summary

Name	Type	Direction	Summary
`profile`	CpuProfile	In/Out	

5.14.2. list GET

Table 41. Parameters summary

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

max

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

5.15. AssignedDiskProfile

Table 42. Methods summary

Name	Summary
`get`	
`remove`	

5.15.1. get GET

Table 43. Parameters summary

Name	Type	Direction	Summary
`disk_profile`	DiskProfile	Out	

5.15.2. remove DELETE

Table 44. Parameters summary

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

5.16. AssignedDiskProfiles

Table 45. Methods summary

Name	Summary
`add`	
`list`	

5.16.1. add POST

Table 46. Parameters summary

Name	Type	Direction	Summary
`profile`	DiskProfile	In/Out	

5.16.2. list GET

Table 47. Parameters summary

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

max

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

5.17. AssignedNetwork

Table 48. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.17.1. get GET

Table 49. Parameters summary

Name	Type	Direction	Summary
`network`	Network	Out	

5.17.2. remove DELETE

Table 50. Parameters summary

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

5.17.3. update PUT

Table 51. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`network`	Network	In/Out	

5.18. AssignedNetworks

Table 52. Methods summary

Name	Summary
`add`	
`list`	

5.18.1. add POST

Table 53. Parameters summary

Name	Type	Direction	Summary
`network`	Network	In/Out	

5.18.2. list GET

Table 54. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of networks to return.
`networks`	Network[]	Out	

max

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

5.19. AssignedPermissions

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

Table 55. Methods summary

Name	Summary
`add`	
`list`	

5.19.1. add POST

Table 56. Parameters summary

Name	Type	Direction	Summary
`permission`	Permission	In/Out	

5.19.2. list GET

Table 57. Parameters summary

Name	Type	Direction	Summary
`permissions`	Permission[]	Out	

5.20. AssignedRoles

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

Table 58. Methods summary

Name	Summary
`list`	

5.20.1. list GET

Table 59. Parameters summary

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

max

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

5.21. AssignedTag

Table 60. Methods summary

Name	Summary
`get`	
`remove`	

5.21.1. get GET

Table 61. Parameters summary

Name	Type	Direction	Summary
`tag`	Tag	Out	

5.21.2. remove DELETE

Table 62. Parameters summary

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

5.22. AssignedTags

Table 63. Methods summary

Name	Summary
`add`	
`list`	

5.22.1. add POST

Table 64. Parameters summary

Name	Type	Direction	Summary
`tag`	Tag	In/Out	

5.22.2. list GET

Table 65. Parameters summary

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

max

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

5.23. AssignedVnicProfile

Table 66. Methods summary

Name	Summary
`get`	
`remove`	

5.23.1. get GET

Table 67. Parameters summary

Name	Type	Direction	Summary
`profile`	VnicProfile	Out	

5.23.2. remove DELETE

Table 68. Parameters summary

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

5.24. AssignedVnicProfiles

Table 69. Methods summary

Name	Summary
`add`	
`list`	

5.24.1. add POST

Table 70. Parameters summary

Name	Type	Direction	Summary
`profile`	VnicProfile	In/Out	

5.24.2. list GET

Table 71. Parameters summary

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

max

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

5.25. AttachedStorageDomain

Table 72. Methods summary

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

5.25.1. activate POST

This operation activates an attached storage domain. Once the storage domain is activated it is

ready for use with the data center.

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

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

```
<action/>
```

Table 73. Parameters summary

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

5.25.2. deactivate POST

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

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

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

```
<action/>
```

Table 74. Parameters summary

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

5.25.3. get GET

Table 75. Parameters summary

Name	Type	Direction	Summary
`storage_domain`	StorageDomain	Out	

5.25.4. remove DELETE

Table 76. Parameters summary

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

5.26. AttachedStorageDomains

Table 77. Methods summary

Name	Summary
`add`	
`list`	

5.26.1. add POST

Table 78. Parameters summary

Name	Type	Direction	Summary
`storage_domain`	StorageDomain	In/Out	

5.26.2. list GET

Table 79. Parameters summary

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

max

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

5.27. Balance

Table 80. Methods summary

Name	Summary
`get`	
`remove`	

5.27.1. get GET

Table 81. Parameters summary

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

5.27.2. remove DELETE

Table 82. Parameters summary

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

5.28. Balances

Table 83. Methods summary

Name	Summary
`add`	
`list`	

5.28.1. add POST

Table 84. Parameters summary

Name	Type	Direction	Summary
`balance`	Balance	In/Out	

5.28.2. list GET

Table 85. Parameters summary

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

max

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

5.29. Bookmark

Table 86. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.29.1. get GET

Table 87. Parameters summary

Name	Type	Direction	Summary
`bookmark`	Bookmark	Out	

5.29.2. remove DELETE

Table 88. Parameters summary

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

5.29.3. update PUT

Table 89. Parameters summary

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

5.30. Bookmarks

Table 90. Methods summary

Name	Summary
`add`	
`list`	

5.30.1. add POST

Table 91. Parameters summary

Name	Type	Direction	Summary
`bookmark`	Bookmark	In/Out	

5.30.2. list GET

Table 92. Parameters summary

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

max

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

5.31. Cluster

Table 93. Methods summary

Name	Summary
`get`	
`remove`	Removes cluster from the system.
`resetemulatedmachine`	
`update`	Updates information about the cluster.

5.31.1. get GET

Table 94. Parameters summary

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

5.31.2. remove DELETE

Removes cluster from the system.

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

Table 95. Parameters summary

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

5.31.3. resetemulatedmachine POST

Table 96. Parameters summary

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

5.31.4. update PUT

Updates information about the cluster.

Only specified fields are updated, others remain unchanged.

E.g. update cluster's CPU:

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

With request body like:

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

Table 97. Parameters summary

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

5.32. ClusterLevel

Provides information about a specific cluster level. See the ClusterLevels service for more information.

Table 98. Methods summary

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

5.32.1. get GET

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

Table 99. Parameters summary

Name	Type	Direction	Summary
`level`	ClusterLevel	Out	

5.33. ClusterLevels

Provides information about the capabilities of different cluster levels supported by the engine. Version 4.0 of the engine supports levels 4.0 and 3.6. Each of these levels support different sets of CPU types, for example. This service provides that information. For example, to find what CPU types are supported by level 3.6 the you can send a request like this:

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

That will return a ClusterLevel object containing the supported CPU types, and other information describing the cluster level:

```
<cluster_level id="3.6" href="/ovirt-engine/api/clusterlevel/3.6">
  <cpu_types>
    <cpu_type>
      <name>Intel Conroe Family</name>
      <level>3</level>
      <architecture>x86_64</architecture>
    </cpu_type>
    ...
  </cpu_types>
  ...
</cluster_level>
```

Table 100. Methods summary

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

5.33.1. list GET

Lists the cluster levels supported by the system.

Table 101. Parameters summary

Name	Type	Direction	Summary
`levels`	ClusterLevel[]	Out	

5.34. Clusters

Table 102. Methods summary

Name	Summary
`add`	Creates a new cluster.
`list`	

5.34.1. add POST

Creates a new cluster.

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

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

With a request body like this:

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

Table 103. Parameters summary

Name	Type	Direction	Summary
`cluster`	Cluster	In/Out	

5.34.2. list GET

Table 104. Parameters summary

Name	Type	Direction	Summary
`case_sensitive`	Boolean	In	Indicates if the search performed using the `search` parameter should be performed taking case into account.

Name	Type	Direction	Summary
`clusters`	Cluster[]	Out	
`filter`	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
`max`	Integer	In	Sets the maximum number of clusters to return.
`search`	String	In	A query string used to restrict the returned clusters.

case_sensitive

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

max

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

5.35. Copyable

Table 105. Methods summary

Name	Summary
`copy`	

5.35.1. copy POST

Table 106. Parameters summary

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

5.36. CpuProfile

Table 107. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.36.1. get GET

Table 108. Parameters summary

Name	Type	Direction	Summary
`profile`	CpuProfile	Out	

5.36.2. remove DELETE

Table 109. Parameters summary

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

5.36.3. update PUT

Table 110. Parameters summary

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

5.37. CpuProfiles

Table 111. Methods summary

Name	Summary
`add`	
`list`	

5.37.1. add POST

Table 112. Parameters summary

Name	Type	Direction	Summary
`profile`	CpuProfile	In/Out	

5.37.2. list GET

Table 113. Parameters summary

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

max

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

5.38. DataCenter

Table 114. Methods summary

Name	Summary
`get`	
`remove`	Removes the data center.
`update`	Updates the data center.

5.38.1. get GET

Table 115. Parameters summary

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

5.38.2. remove DELETE

Removes the data center.

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

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

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

Table 116. Parameters summary

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

force

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

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

5.38.3. update PUT

Updates the data center.

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

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

With a request body like this:

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

Table 117. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`data_center`	DataCenter	In/Out	The data center that is being updated.

5.39. DataCenters

Table 118. Methods summary

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

5.39.1. add POST

Creates a new data center.

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

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

With a request body like this:

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

Table 119. Parameters summary

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

5.39.2. list GET

Lists the data centers.

The following request retrieves a representation of the data centers:

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

The above request performed with `curl`:

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

This is what an example response could look like:

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

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

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

Table 120. Parameters summary

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

case_sensitive

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

max

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

5.40. Disk

Manages a single disk.

Table 121. Methods summary

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

5.40.1. copy POST

This operation copies a disk to the specified storage domain.

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

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

With a request body like this:

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

Table 122. Parameters summary

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

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

5.40.2. export POST

Table 123. Parameters summary

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

5.40.3. get GET

Table 124. Parameters summary

Name	Type	Direction	Summary
`disk`	Disk	Out	

5.40.4. move POST

Moves a disk to another storage domain.

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

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

With the following request body:

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

Table 125. Parameters summary

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

5.40.5. remove DELETE

Table 126. Parameters summary

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

5.41. DiskAttachment

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

Table 127. Methods summary

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

5.41.1. get GET

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

Table 128. Parameters summary

Name	Type	Direction	Summary
`attachment`	DiskAttachment	Out	

5.41.2. remove DELETE

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

Table 129. Parameters summary

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

detach_only

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

5.41.3. update PUT

Update the disk attachment and the disk properties within it.

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

Table 130. Parameters summary

Name	Type	Direction	Summary
`disk_attachment`	DiskAttachment	In/Out	

5.42. DiskAttachments

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

Table 131. Methods summary

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

5.42.1. add POST

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

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

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

```
<disk_attachment>
  <bootable>true</bootable>
  <interface>ide</interface>
  <active>true</active>
  <disk>
    <name>mydisk</name>
    <provisioned_size>1024</provisioned_size>
    ...
  </disk>
</disk_attachment>
```

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

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

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

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

Table 132. Parameters summary

Name	Type	Direction	Summary
`attachment`	DiskAttachment	In/Out	

5.42.2. list GET

List the disk that are attached to the virtual machine.

Table 133. Parameters summary

Name	Type	Direction	Summary
`attachments`	DiskAttachment[]	Out	

5.43. DiskProfile

Table 134. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.43.1. get GET

Table 135. Parameters summary

Name	Type	Direction	Summary
`profile`	DiskProfile	Out	

5.43.2. remove DELETE

Table 136. Parameters summary

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

5.43.3. update PUT

Table 137. Parameters summary

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

5.44. DiskProfiles

Table 138. Methods summary

Name	Summary
`add`	

Name	Summary
`list`	

5.44.1. add POST

Table 139. Parameters summary

Name	Type	Direction	Summary
`profile`	DiskProfile	In/Out	

5.44.2. list GET

Table 140. Parameters summary

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

max

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

5.45. DiskSnapshot

Table 141. Methods summary

Name	Summary
`get`	
`remove`	

5.45.1. get GET

Table 142. Parameters summary

Name	Type	Direction	Summary
`snapshot`	DiskSnapshot	Out	

5.45.2. remove DELETE

Table 143. Parameters summary

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

5.46. DiskSnapshots

Table 144. Methods summary

Name	Summary
`list`	

5.46.1. list GET

Table 145. Parameters summary

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

max

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

5.47. Disks

Manages the collection of disks available in the system.

Table 146. Methods summary

Name	Summary
`add`	Adds a new floating disk.
`list`	

5.47.1. add POST

Adds a new floating disk.

When creating a new floating Disk, the API requires the `storage_domain`, `provisioned_size` and `format` attributes.

To create a new floating disk with specified `provisioned_size`, `format` and `name` on a storage domain with an id `e9fedf39-5edc-4e0a-8628-253f1b9c5693`, send a request as follows:

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

With a request body as follows:

```
<disk>
  <storage_domains>
    <storage_domain id="e9fedf39-5edc-4e0a-8628-253f1b9c5693"/>
  </storage_domains>
  <name>disk1</name>
  <provisioned_size>1048576</provisioned_size>
  <format>cow</format>
</disk>
```

Table 147. Parameters summary

Name	Type	Direction	Summary
`disk`	Disk	In/Out	

5.47.2. list GET

Table 148. Parameters summary

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

case_sensitive

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

max

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

5.48. Domain

Table 149. Methods summary

Name	Summary
`get`	

5.48.1. get GET

Table 150. Parameters summary

Name	Type	Direction	Summary
`domain`	Domain	Out	

5.49. DomainGroup

Table 151. Methods summary

Name	Summary
`get`	

5.49.1. get GET

Table 152. Parameters summary

Name	Type	Direction	Summary
`get`	Group	Out	

5.50. DomainGroups

Table 153. Methods summary

Name	Summary
`list`	

5.50.1. list GET

Table 154. Parameters summary

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

case_sensitive

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

max

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

5.51. DomainUser

Table 155. Methods summary

Name	Summary
`get`	

5.51.1. get GET

Table 156. Parameters summary

Name	Type	Direction	Summary
`user`	User	Out	

5.52. DomainUsers

Table 157. Methods summary

Name	Summary
`list`	

5.52.1. list GET

Table 158. Parameters summary

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

case_sensitive

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

max

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

5.53. Domains

Table 159. Methods summary

Name	Summary
`list`	

5.53.1. list GET

Table 160. Parameters summary

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

max

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

5.54. EngineKatelloErrata

This service doesn't add any new methods, it is just a placeholder for the annotation that specifies the path of the resource that manages the Katello errata assigned to the engine.

Table 161. Methods summary

Name	Summary
`list`	

5.54.1. list GET

Table 162. Parameters summary

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

max

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

5.55. Event

Table 163. Methods summary

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

5.55.1. get GET

Table 164. Parameters summary

Name	Type	Direction	Summary
`event`	Event	Out	

5.55.2. remove DELETE

Removes an event from internal audit log.

An event can be removed by sending following request

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

Table 165. Parameters summary

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

5.56. Events

Table 166. Methods summary

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

5.56.1. add POST

Adds an external event to the internal audit log.

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

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

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

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

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

Table 167. Parameters summary

Name	Type	Direction	Summary
`event`	Event	In/Out	

5.56.2. list GET

Get list of events.

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

To the above request we get following reponse:

```
<events>
  <event href="/ovirt-engine/api/events/2" id="2">
    <description>User admin@internal-authz logged out.</description>
    <code>31</code>
    <correlation_id>1e892ea9</correlation_id>
    <custom_id>-1</custom_id>
    <flood_rate>30</flood_rate>
    <origin>oVirt</origin>
    <severity>normal</severity>
    <time>2016-09-14T12:14:34.541+02:00</time>
    <user href="/ovirt-engine/api/users/57d91d48-00da-0137-0138-000000000244"
id="57d91d48-00da-0137-0138-000000000244"/>
  </event>
  <event href="/ovirt-engine/api/events/1" id="1">
    <description>User admin logged in.</description>
    <code>30</code>
    <correlation_id>1fbd81f4</correlation_id>
    <custom_id>-1</custom_id>
    <flood_rate>30</flood_rate>
    <origin>oVirt</origin>
    <severity>normal</severity>
    <time>2016-09-14T11:54:35.229+02:00</time>
    <user href="/ovirt-engine/api/users/57d91d48-00da-0137-0138-000000000244"
id="57d91d48-00da-0137-0138-000000000244"/>
  </event>
</events>
```

The following events occur:

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

Table 168. Parameters summary

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

case_sensitive

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

from

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

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

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

max

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

search

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

We can search by providing specific severity.

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

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

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

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

```
sortby time asc page 1
```

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

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

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

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

5.56.3. undelete POST

Table 169. Parameters summary

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

5.57. ExternalComputeResource

Table 170. Methods summary

Name	Summary
`get`	

5.57.1. get GET

Table 171. Parameters summary

Name	Type	Direction	Summary
`resource`	ExternalComputeResource	Out	

5.58. ExternalComputeResources

Table 172. Methods summary

Name	Summary
`list`	

5.58.1. list GET

Table 173. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of resources to return.
`resources`	ExternalComputeResource[]	Out	

max

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

5.59. ExternalDiscoveredHost

Table 174. Methods summary

Name	Summary
`get`	

5.59.1. get GET

Table 175. Parameters summary

Name	Type	Direction	Summary
`host`	ExternalDiscoveredHost	Out	

5.60. ExternalDiscoveredHosts

Table 176. Methods summary

Name	Summary
`list`	

5.60.1. list GET

Table 177. Parameters summary

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

max

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

5.61. ExternalHost

Table 178. Methods summary

Name	Summary
`get`	

5.61.1. get GET

Table 179. Parameters summary

Name	Type	Direction	Summary
`host`	ExternalHost	Out	

5.62. ExternalHostGroup

Table 180. Methods summary

Name	Summary
`get`	

5.62.1. get GET

Table 181. Parameters summary

Name	Type	Direction	Summary
`group`	ExternalHost Group	Out	

5.63. ExternalHostGroups

Table 182. Methods summary

Name	Summary
`list`	

5.63.1. list GET

Table 183. Parameters summary

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

max

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

5.64. ExternalHostProvider

Table 184. Methods summary

Name	Summary
`get`	
`importcertificates`	
`remove`	
`testconnectivity`	
`update`	

5.64.1. get GET

Table 185. Parameters summary

Name	Type	Direction	Summary
`provider`	ExternalHost Provider	Out	

5.64.2. importcertificates POST

Table 186. Parameters summary

Name	Type	Direction	Summary
`certificates`	Certificate[]	In	

5.64.3. remove DELETE

Table 187. Parameters summary

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

5.64.4. testconnectivity POST

Table 188. Parameters summary

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

5.64.5. update PUT

Table 189. Parameters summary

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

5.65. ExternalHostProviders

Table 190. Methods summary

Name	Summary
`add`	

Name	Summary
`list`	

5.65.1. add POST

Table 191. Parameters summary

Name	Type	Direction	Summary
`provider`	ExternalHost Provider	In/Out	

5.65.2. list GET

Table 192. Parameters summary

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

max

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

5.66. ExternalHosts

Table 193. Methods summary

Name	Summary
`list`	

5.66.1. list GET

Table 194. Parameters summary

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

max

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

5.67. ExternalProvider

Table 195. Methods summary

Name	Summary
`importcertificates`	
`testconnectivity`	

5.67.1. importcertificates POST

Table 196. Parameters summary

Name	Type	Direction	Summary
`certificates`	Certificate[]	In	

5.67.2. testconnectivity POST

Table 197. Parameters summary

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

5.68. ExternalProviderCertificate

Table 198. Methods summary

Name	Summary
`get`	

5.68.1. get GET

Table 199. Parameters summary

Name	Type	Direction	Summary
`certificate`	Certificate	Out	

5.69. ExternalProviderCertificates

Table 200. Methods summary

Name	Summary
`list`	

5.69.1. list GET

Table 201. Parameters summary

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

max

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

5.70. ExternalVmImports

Provides capability to import external virtual machines.

Table 202. Methods summary

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

5.70.1. add POST

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

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

```
POST /externalvmimports
```

With request body of type ExternalVmImport, for example:

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

Table 203. Parameters summary

Name	Type	Direction	Summary
import	ExternalVmImport	In/Out	

5.71. FenceAgent

Table 204. Methods summary

Name	Summary
get	
remove	
update	

5.71.1. get GET

Table 205. Parameters summary

Name	Type	Direction	Summary
agent	Agent	Out	

5.71.2. remove DELETE

Table 206. Parameters summary

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

5.71.3. update PUT

Table 207. Parameters summary

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

5.72. FenceAgents

Table 208. Methods summary

Name	Summary
`add`	
`list`	

5.72.1. add POST

Table 209. Parameters summary

Name	Type	Direction	Summary
`agent`	Agent	In/Out	

5.72.2. list GET

Table 210. Parameters summary

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

max

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

5.73. File

Table 211. Methods summary

Name	Summary
`get`	

5.73.1. get GET

Table 212. Parameters summary

Name	Type	Direction	Summary
`file`	File	Out	

5.74. Files

Provides a way for clients to list available files.

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

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

Table 213. Methods summary

Name	Summary
`list`	

5.74.1. list GET

Table 214. Parameters summary

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

case_sensitive

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

max

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

5.75. Filter

Table 215. Methods summary

Name	Summary
`get`	

Name	Summary
`remove`	

5.75.1. get GET

Table 216. Parameters summary

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

5.75.2. remove DELETE

Table 217. Parameters summary

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

5.76. Filters

Table 218. Methods summary

Name	Summary
`add`	
`list`	

5.76.1. add POST

Table 219. Parameters summary

Name	Type	Direction	Summary
`filter`	Filter	In/Out	

5.76.2. list GET

Table 220. Parameters summary

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

max

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

5.77. GlusterBrick

Table 221. Methods summary

Name	Summary
`get`	
`remove`	Removes this brick from the volume and deletes it from the database.
`replace`	Replaces this brick with a new one.

5.77.1. get GET

Table 222. Parameters summary

Name	Type	Direction	Summary
`brick`	GlusterBrick	Out	

5.77.2. remove DELETE

Removes this brick from the volume and deletes it from the database.

Table 223. Parameters summary

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

5.77.3. replace POST

Replaces this brick with a new one. The property `brick` is required.

Table 224. Parameters summary

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

5.78. GlusterBricks

Table 225. Methods summary

Name	Summary
`activate`	
`add`	Adds given list of bricks to the volume, and updates the database accordingly.
`list`	
`migrate`	
`remove`	Removes the given list of bricks brick from the volume and deletes them from the database.
`stopmigrate`	

5.78.1. activate POST

Table 226. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the activation should be performed asynchronously.
`bricks`	GlusterBrick[]	In	

5.78.2. add POST

Adds given list of bricks to the volume, and updates the database accordingly. The properties `serverId` and `brickDir`are required.

Table 227. Parameters summary

Name	Type	Direction	Summary
`bricks`	GlusterBrick[]	In/Out	The list of bricks to be added to the volume

5.78.3. list GET

Table 228. Parameters summary

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

max

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

5.78.4. migrate POST

Table 229. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the migration should be performed asynchronously.
`bricks`	GlusterBrick[]	In	

5.78.5. remove DELETE

Removes the given list of bricks brick from the volume and deletes them from the database.

Table 230. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the remove should be performed asynchronously.
`bricks`	GlusterBrick[]	In	The list of bricks to be removed

5.78.6. stopmigrate POST

Table 231. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the action should be performed asynchronously.
`bricks`	GlusterBrick[]	In	

5.79. GlusterHook

Table 232. Methods summary

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

5.79.1. disable POST

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

Table 233. Parameters summary

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

5.79.2. enable POST

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

Table 234. Parameters summary

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

5.79.3. get GET

Table 235. Parameters summary

Name	Type	Direction	Summary
`hook`	GlusterHook	Out	

5.79.4. remove DELETE

Removes the this Gluster hook from all servers in cluster and deletes it from the database.

Table 236. Parameters summary

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

5.79.5. resolve POST

Resolves missing hook conflict depending on the resolution type.

For `ADD` resolves by copying hook stored in engine database to all servers where the hook is missing. The engine maintains a list of all servers where hook is missing.

For `COPY` resolves conflict in hook content by copying hook stored in engine database to all servers where the hook is missing. The engine maintains a list of all servers where the content is conflicting. If a host id is passed as parameter, the hook content from the server is used as the

master to copy to other servers in cluster.

Table 237. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the action should be performed asynchronously.
`host`	Host	In	
`resolution_type`	String	In	

5.80. GlusterHooks

Table 238. Methods summary

Name	Summary
`list`	

5.80.1. list GET

Table 239. Parameters summary

Name	Type	Direction	Summary
`hooks`	GlusterHook[]	Out	
`max`	Integer	In	Sets the maximum number of hooks to return.

max

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

5.81. GlusterVolume

Table 240. Methods summary

Name	Summary
`get`	
`getprofilestatistics`	
`rebalance`	
`remove`	
`resetalloptions`	
`resetoption`	
`setoption`	
`start`	

Name	Summary
`startprofile`	
`stop`	
`stopprofile`	
`stoprebalance`	

5.81.1. get GET

Table 241. Parameters summary

Name	Type	Direction	Summary
`volume`	GlusterVolume	Out	

5.81.2. getprofilestatistics POST

Table 242. Parameters summary

Name	Type	Direction	Summary
`details`	GlusterVolumeProfileDetails	Out	

5.81.3. rebalance POST

Table 243. Parameters summary

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

5.81.4. remove DELETE

Table 244. Parameters summary

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

5.81.5. resetalloptions POST

Table 245. Parameters summary

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

5.81.6. resetoption POST

Table 246. Parameters summary

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

5.81.7. setoption POST

Table 247. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the action should be performed asynchronously.
`option`	Option	In	

5.81.8. start POST

Table 248. Parameters summary

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

5.81.9. startprofile POST

Table 249. Parameters summary

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

5.81.10. stop POST

Table 250. Parameters summary

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

5.81.11. stopprofile POST

Table 251. Parameters summary

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

5.81.12. stoprebalance POST

Table 252. Parameters summary

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

5.82. GlusterVolumes

Table 253. Methods summary

Name	Summary
`add`	Creates a new Gluster volume and adds it to the database.
`list`	

5.82.1. add POST

Creates a new Gluster volume and adds it to the database. The volume is created based on properties of the `volume` parameter. The properties `name`, `volumeType` and `bricks` are required.

Table 254. Parameters summary

Name	Type	Direction	Summary
`volume`	GlusterVolume	In/Out	The Gluster volume definition from which to create the volume is passed as input and the newly created volume is returned.

5.82.2. list GET

Table 255. Parameters summary

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

case_sensitive

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

max

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

5.83. GraphicsConsole

Table 256. Methods summary

Name	Summary
`get`	Gets the configuration of the graphics console.
`remove`	

5.83.1. get GET

Gets the configuration of the graphics console.

Table 257. Parameters summary

Name	Type	Direction	Summary
`console`	GraphicsConsole	Out	
`current`	Boolean	In	Use the following query to obtain the current run-time configuration of the graphics console.

current

Use the following query to obtain the current run-time configuration of the graphics console.

```
GET /ovit-engine/api/vms/{vm:id}/graphicsconsoles/{console:id}?current=true
```

The default value is `false`.

5.83.2. remove DELETE

Table 258. Parameters summary

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

5.84. GraphicsConsoles

Table 259. Methods summary

Name	Summary
`add`	
`list`	Lists all the configured graphics consoles of the virtual machine.

5.84.1. add POST

Table 260. Parameters summary

Name	Type	Direction	Summary
`console`	GraphicsConsole	In/Out	

5.84.2. list GET

Lists all the configured graphics consoles of the virtual machine.

Table 261. Parameters summary

Name	Type	Direction	Summary
`consoles`	GraphicsConsole[]	Out	
`current`	Boolean	In	Use the following query to obtain the current run-time configuration of the graphics consoles.
`max`	Integer	In	Sets the maximum number of consoles to return.

current

Use the following query to obtain the current run-time configuration of the graphics consoles.

```
GET /ovit-engine/api/vms/{vm:id}/graphicsconsoles?current=true
```

The default value is `false`.

max

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

5.85. Group

Table 262. Methods summary

Name	Summary
`get`	
`remove`	

5.85.1. get GET

Table 263. Parameters summary

Name	Type	Direction	Summary
`get`	Group	Out	

5.85.2. remove DELETE

Table 264. Parameters summary

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

5.86. Groups

Table 265. Methods summary

Name	Summary
`add`	Add group from a directory service.
`list`	

5.86.1. add POST

Add group from a directory service. Please note that domain name is name of the authorization provider.

For example, to add the `Developers` group from the `internal-authz` authorization provider send a request like this:

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

With a request body like this:

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

Table 266. Parameters summary

Name	Type	Direction	Summary
group	Group	In/Out	

5.86.2. list GET

Table 267. Parameters summary

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

case_sensitive

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

max

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

5.87. Host

Table 268. Methods summary

Name	Summary
activate	Activate the host for use, such as running virtual machines.
approve	Approve a pre-installed Hypervisor host for usage in the virtualization environment.
commitnetconfig	Marks the network configuration as good and persists it inside the host.

Name	Summary
`deactivate`	Deactivate the host to perform maintenance tasks.
`enrollcertificate`	Enroll certificate of the host.
`fence`	Controls host's power management device.
`forceselectspm`	Manually set a host as the storage pool manager (SPM).
`get`	Get the host details.
`install`	Install VDSM and related software on the host.
`iscsidiscover`	
`iscsilogin`	
`refresh`	
`remove`	Remove the host from the system.
`setupnetworks`	This method is used to change the configuration of the network interfaces of a host.
`unregisteredstoragedomainsdiscover`	
`update`	Update the host properties.
`upgrade`	Upgrade VDSM and selected software on the host.

5.87.1. activate POST

Activate the host for use, such as running virtual machines.

Table 269. Parameters summary

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

5.87.2. approve POST

Approve a pre-installed Hypervisor host for usage in the virtualization environment.

This action also accepts an optional cluster element to define the target cluster for this host.

Table 270. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the approval should be performed asynchronously.
`cluster`	Cluster	In	

5.87.3. commitnetconfig POST

Marks the network configuration as good and persists it inside the host.

An API user commits the network configuration to persist a host network interface attachment or detachment, or persist the creation and deletion of a bonded interface.

IMPORTANT	Networking configuration is only committed after the engine has established that host connectivity is not lost as a result of the configuration changes. If host connectivity is lost, the host requires a reboot and automatically reverts to the previous networking configuration.

For example, to commit the network configuration of host with id 123 send a request like this:

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

With a request body like this:

```
<action/>
```

Table 271. Parameters summary

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

5.87.4. deactivate POST

Deactivate the host to perform maintenance tasks.

Table 272. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the deactivation should be performed asynchronously.
`reason`	String	In	
`stop_gluster_service`	Boolean	In	Indicates if the gluster service should be stopped as part of deactivating the host.

stop_gluster_service

Indicates if the gluster service should be stopped as part of deactivating the host. It can be used while performing maintenance operations on the gluster host. Default value for this variable is `false`.

5.87.5. enrollcertificate POST

Enroll certificate of the host. Useful in case you get a warning that it is about to, or already expired.

Table 273. Parameters summary

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

5.87.6. fence POST

Controls host's power management device.

For example, let's assume you want to start the host. This can be done via:

```
#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."

curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--user "${user}:${password}" \
--request POST \
--header "Version: 4" \
--header "Content-Type: application/xml" \
--header "Accept: application/xml" \
--data '
<action>
  <fence_type>start</fence_type>
</action>
' \
"${url}/hosts/123/fence"
```

Table 274. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the fencing should be performed asynchronously.
`fence_type`	String	In	
`power_management`	PowerManagement	Out	

5.87.7. forceselectspm POST

Manually set a host as the storage pool manager (SPM).

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

With a request body like this:

```
<action/>
```

Table 275. Parameters summary

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

5.87.8. get GET

Get the host details.

Table 276. Parameters summary

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

5.87.9. install POST

Install VDSM and related software on the host. The host type defines additional parameters for the action.

Example of installing a host, using `curl` and JSON, plain:

```
curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--request PUT \
--header "Content-Type: application/json" \
--header "Accept: application/json" \
--header "Version: 4" \
--user "admin@internal:..." \
--data '
{
  "root_password": "myrootpassword"
}
' \
"https://engine.example.com/ovirt-engine/api/hosts/123"
```

Example of installing a host, using `curl` and JSON, with hosted engine components:

```
curl \
curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--request PUT \
--header "Content-Type: application/json" \
--header "Accept: application/json" \
--header "Version: 4" \
--user "admin@internal:..." \
--data '
{
  "root_password": "myrootpassword"
}
' \
"https://engine.example.com/ovirt-engine/api/hosts/123?deploy_hosted_engine=true"
```

Table 277. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the installation should be performed asynchronously.
`deploy_hosted_engine`	Boolean	In	When set to `true` it means this host should deploy also hosted engine components.
`host`	Host	In	This `override_iptables` property is used to indicate if the firewall configuration should be replaced by the default one.
`image`	String	In	When installing an oVirt node a image ISO file is needed.

Name	Type	Direction	Summary
`root_password`	String	In	The password of of the `root` user, used to connect to the host via SSH.
`ssh`	Ssh	In	The SSH details used to connect to the host.
`undeploy_hosted_engine`	Boolean	In	When set to `true` it means this host should un-deploy hosted engine components and this host will not function as part of the High Availability cluster.

deploy_hosted_engine

When set to `true` it means this host should deploy also hosted engine components. Missing value is treated as `true` i.e deploy. Omitting this parameter means `false` and will perform no operation in hosted engine area.

undeploy_hosted_engine

When set to `true` it means this host should un-deploy hosted engine components and this host will not function as part of the High Availability cluster. Missing value is treated as `true` i.e un-deploy Omitting this parameter means `false` and will perform no operation in hosted engine area.

5.87.10. iscsidiscover POST

Table 278. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the discovery should be performed asynchronously.
`iscsi`	IscsiDetails	In	
`iscsi_targets`	String[]	Out	

5.87.11. iscsilogin POST

Table 279. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the login should be performed asynchronously.
`iscsi`	IscsiDetails	In	

5.87.12. refresh POST

Table 280. Parameters summary

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

5.87.13. remove DELETE

Remove the host from the system.

```
#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."

curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--user "${user}:${password}" \
--request DELETE \
--header "Version: 4" \
"${url}/hosts/1ff7a191-2f3b-4eff-812b-9f91a30c3acc"
```

Table 281. Parameters summary

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

5.87.14. setupnetworks POST

This method is used to change the configuration of the network interfaces of a host.

For example, lets assume that you have a host with three network interfaces `eth0`, `eth1` and `eth2` and that you want to configure a new bond using `eth0` and `eth1`, and put a VLAN on top of it. Using a simple shell script and the `curl` command line HTTP client that can be done as follows:

```
#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."

curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--user "${user}:${password}" \
--request POST \
```

```
--header "Version: 4" \
--header "Content-Type: application/xml" \
--header "Accept: application/xml" \
--data '
<action>
  <modified_bonds>
    <host_nic>
      <name>bond0</name>
      <bonding>
        <options>
          <option>
            <name>mode</name>
            <value>4</value>
          </option>
          <option>
            <name>miimon</name>
            <value>100</value>
          </option>
        </options>
        <slaves>
          <host_nic>
            <name>eth1</name>
          </host_nic>
          <host_nic>
            <name>eth2</name>
          </host_nic>
        </slaves>
      </bonding>
    </host_nic>
  </modified_bonds>
  <modified_network_attachments>
    <network_attachment>
      <network>
        <name>myvlan</name>
      </network>
      <host_nic>
        <name>bond0</name>
      </host_nic>
      <ip_address_assignments>
        <assignment_method>static</assignment_method>
        <ip_address_assignment>
          <ip>
            <address>192.168.122.10</address>
            <netmask>255.255.255.0</netmask>
          </ip>
        </ip_address_assignment>
      </ip_address_assignments>
    </network_attachment>
  </modified_network_attachments>
 </action>
' \
```

```
"${url}/hosts/1ff7a191-2f3b-4eff-812b-9f91a30c3acc/setupnetworks"
```

Note that this is valid for version 4 of the API. In previous versions some elements were represented as XML attributes instead of XML elements. In particular the `options` and `ip` elements were represented as follows:

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

Using the Python SDK the same can be done with the following code:

```
host.setupnetworks(
  params.Action(
    modified_bonds=params.HostNics(
      host_nic=[
        params.HostNIC(
          name="bond0",
          bonding=params.Bonding(
            options=params.Options(
              option=[
                params.Option(name="mode", value="4"),
                params.Option(name="miimon", value="100"),
              ],
            ),
            slaves=params.Slaves(
              host_nic=[
                params.HostNIC(name="eth1"),
                params.HostNIC(name="eth2"),
              ],
            ),
          ),
        ),
      ],
    ),
    modified_network_attachments=params.NetworkAttachments(
      network_attachment=[
        params.NetworkAttachment(
          network=params.Network(name="myvlan"),
          host_nic=params.HostNIC(name="bond0"),
          ip_address_assignments=params.IpAddressAssignments(
            ip_address_assignment=[
              params.IpAddressAssignment(
                assignment_method="static",
                ip=params.IP(
                  address="192.168.122.10",
                  netmask="255.255.255.0",
                ),
              ),
            ],
          ),
        ),
      ],
    ),
  ),
)
```

IMPORTANT	To make sure that the network configuration has been saved in the host, and that it will be applied when the host is rebooted, remember to call commitnetconfig.

Table 282. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the action should be performed asynchronously.
`check_connectivity`	Boolean	In	
`connectivity_timeout`	Integer	In	
`modified_bonds`	HostNic[]	In	
`modified_labels`	NetworkLabel[]	In	
`modified_network_attachments`	NetworkAttachment[]	In	
`removed_bonds`	HostNic[]	In	
`removed_labels`	NetworkLabel[]	In	
`removed_network_attachments`	NetworkAttachment[]	In	
`synchronized_network_attachments`	NetworkAttachment[]	In	

5.87.15. unregisteredstoragedomainsdiscover POST

Table 283. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the discovery should be performed asynchronously.
`iscsi`	IscsiDetails	In	
`storage_domains`	StorageDomain[]	Out	

5.87.16. update PUT

Update the host properties.

For example, to update a the kernel command line of a host send a request like this:

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

With request body like this:

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

Table 284. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`host`	Host	In/Out	

5.87.17. upgrade POST

Upgrade VDSM and selected software on the host.

Table 285. Parameters summary

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

5.88. HostDevice

Table 286. Methods summary

Name	Summary
`get`	

5.88.1. get GET

Table 287. Parameters summary

Name	Type	Direction	Summary
`device`	HostDevice	Out	

5.89. HostDevices

Table 288. Methods summary

Name	Summary
`list`	

5.89.1. list GET

Table 289. Parameters summary

Name	Type	Direction	Summary
`devices`	HostDevice[]	Out	
`max`	Integer	In	Sets the maximum number of devices to return.

max

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

5.90. HostHook

Table 290. Methods summary

Name	Summary
`get`	

5.90.1. get GET

Table 291. Parameters summary

Name	Type	Direction	Summary
`hook`	Hook	Out	

5.91. HostHooks

Table 292. Methods summary

Name	Summary
`list`	

5.91.1. list GET

Table 293. Parameters summary

Name	Type	Direction	Summary
`hooks`	Hook[]	Out	
`max`	Integer	In	Sets the maximum number of hooks to return.

max

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

5.92. HostNic

Table 294. Methods summary

Name	Summary
`get`	
`updatevirtualfunctionsconfiguration`	The action updates virtual function configuration in case the current resource represents an SR-IOV enabled NIC.

5.92.1. get GET

Table 295. Parameters summary

Name	Type	Direction	Summary
`nic`	HostNic	Out	

5.92.2. updatevirtualfunctionsconfiguration POST

The action updates virtual function configuration in case the current resource represents an SR-IOV enabled NIC. The input should be consisted of at least one of the following properties:

- `allNetworksAllowed`
- `numberOfVirtualFunctions`

Please see the `HostNicVirtualFunctionsConfiguration` type for the meaning of the properties.

Table 296. Parameters summary

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

5.93. HostNics

Table 297. Methods summary

Name	Summary
`list`	

5.93.1. list GET

Table 298. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of NICs to return.
`nics`	HostNic[]	Out	

max

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

5.94. HostNumaNode

Table 299. Methods summary

Name	Summary
`get`	

5.94.1. get GET

Table 300. Parameters summary

Name	Type	Direction	Summary
`node`	NumaNode	Out	

5.95. HostNumaNodes

Table 301. Methods summary

Name	Summary
`list`	

5.95.1. list GET

Table 302. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of nodes to return.
`nodes`	NumaNode[]	Out	

max

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

5.96. HostStorage

Table 303. Methods summary

Name	Summary
list	

5.96.1. list GET

Table 304. Parameters summary

Name	Type	Direction	Summary
report_status	Boolean	In	Indicates if the status of the LUNs in the storage should be checked.
storages	HostStorage[]	Out	

report_status

Indicates if the status of the LUNs in the storage should be checked. Checking the status of the LUN is an heavy weight operation and this data is not always needed by the user. This parameter will give the option to not perform the status check of the LUNs.

The default is `true` for backward compatibility.

Here an example with the LUN status :

```
<host_storage id="360014051136c20574f743bdbd28177fd">
  <logical_units>
    <logical_unit id="360014051136c20574f743bdbd28177fd">
      <lun_mapping>0</lun_mapping>
      <paths>1</paths>
      <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>
```

Here an example without the LUN status :

```
<host_storage id="360014051136c20574f743bdbd28177fd">
  <logical_units>
    <logical_unit id="360014051136c20574f743bdbd28177fd">
      <lun_mapping>0</lun_mapping>
      <paths>1</paths>
      <product_id>lun0</product_id>
      <serial>SLIO-ORG_lun0_1136c205-74f7-43bd-bd28-177fd5ce6993</serial>
      <size>10737418240</size>
      <vendor_id>LIO-ORG</vendor_id>
      <volume_group_id>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>
```

5.97. Hosts

Table 305. Methods summary

Name	Summary
`add`	Creates a new host.
`list`	

5.97.1. add POST

Creates a new host.

The host is created based on the attributes of the `host` parameter. The `name`, `address` and `root_password` properties are required.

For example, to add a host send the following request:

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

With the following request body:

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

NOTE	The `root_password` element is only included in the client-provided initial representation and is not exposed in the representations returned from subsequent requests.

To add a hosted engine host, use the optional `deploy_hosted_engine` parameter:

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

Table 306. Parameters summary

Name	Type	Direction	Summary
`deploy_hosted_engine`	Boolean	In	When set to `true` it means this host should deploy also hosted engine components.
`host`	Host	In/Out	The host definition from which to create the new host is passed as parameter, and the newly created host is returned.
`undeploy_hosted_engine`	Boolean	In	When set to `true` it means this host should un-deploy hosted engine components and this host will not function as part of the High Availability cluster.

deploy_hosted_engine

When set to `true` it means this host should deploy also hosted engine components. Missing value is treated as `true` i.e deploy. Omitting this parameter means `false` and will perform no operation in hosted engine area.

undeploy_hosted_engine

When set to `true` it means this host should un-deploy hosted engine components and this host will not function as part of the High Availability cluster. Missing value is treated as `true` i.e un-deploy. Omitting this parameter means `false` and will perform no operation in hosted engine area.

5.97.2. list GET

Table 307. Parameters summary

Name	Type	Direction	Summary
`case_sensitive`	Boolean	In	Indicates if the search performed using the `search` parameter should be performed taking case into account.
`filter`	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
`hosts`	Host[]	Out	
`max`	Integer	In	Sets the maximum number of hosts to return.

Name	Type	Direction	Summary
`search`	String	In	A query string used to restrict the returned hosts.

case_sensitive

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

max

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

5.98. Icon

Table 308. Methods summary

Name	Summary
`get`	

5.98.1. get GET

Table 309. Parameters summary

Name	Type	Direction	Summary
`icon`	Icon	Out	

5.99. Icons

Table 310. Methods summary

Name	Summary
`list`	

5.99.1. list GET

Table 311. Parameters summary

Name	Type	Direction	Summary
`icons`	Icon[]	Out	
`max`	Integer	In	Sets the maximum number of icons to return.

max

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

5.100. Image

Table 312. Methods summary

Name	Summary
`get`	
`import`	

5.100.1. get GET

Table 313. Parameters summary

Name	Type	Direction	Summary
`image`	Image	Out	

5.100.2. import POST

Table 314. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the import should be performed asynchronously.
`cluster`	Cluster	In	Cluster where the image should be imported.
`disk`	Disk	In	The disk which should be imported.
`import_as_template`	Boolean	In	Specify if template should be created from the imported disk.
`storage_domain`	StorageDomain	In	Storage domain where disk should be imported.
`template`	Template	In	Name of the template, which should be created.

cluster

Cluster where the image should be imported. Has effect only in case `import_as_template` parameter is set to `true`.

template

Name of the template, which should be created. Has effect only in case `import_as_template` parameter is set to `true`.

5.101. ImageTransfer

This service provides a mechanism to control an image transfer. The client will have to create a transfer by using add of the ImageTransfers service, stating the image to transfer data to/from.

After doing that, the transfer is managed by this service.

E.g., for uploading to the disk image with id `52cb593f-837c-4633-a444-35a0a0383706`, the client can use oVirt's Python's SDK as follows:

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

Transfers have phases, which govern the flow of the upload/download. A client implementing such a flow should poll/check the transfer's phase and act accordingly. All the possible phases can be found in ImageTransferPhase.

After adding a new transfer, its phase will be initializing. The client will have to poll on the transfer's phase until it changes. When the phase becomes transferring, the session is ready to start the transfer.

For example:

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

At that stage, if the transfer's phase is paused_system, then the session was not successfully established. One possible reason for that is that the ovirt-imageio-daemon is not running in the host that was selected for transfer. The transfer can be resumed by calling resume of the service that manages it.

If the session was successfully established - the returned transfer entity will contain the proxy_url and signed_ticket attributes, which the client needs to use in order to transfer the required data. The client can choose whatever technique and tool for sending the HTTPS request with the image's data.

- `proxy_url` is the address of a proxy server to the image, to do I/O to.
- `signed_ticket` is the content that needs to be added to the `Authentication` header in the HTTPS request, in order to perform a trusted communication.

For example, Python's HTTPSConnection can be used in order to perform an upload, so an `upload_headers` dict is set for the upcoming upload:

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

Using Python's `HTTPSConnection`, a new connection is established:

```
# Extract the URI, port, and path from the transfer's proxy_url.
url = urlparse(transfer.proxy_url)

# Create a new instance of the connection.
proxy_connection = HTTPSConnection(
   url.hostname,
   url.port,
   context=ssl.SSLContext(ssl.PROTOCOL_SSLv23)
)
```

The specific content range being sent must be noted in the `Content-Range` HTTPS header. This can be used in order to split the transfer into several requests for a more flexible process.

For doing that, the client will have to repeatedly extend the transfer session to keep the channel open. Otherwise, the session will terminate and the transfer will get into `paused_system` phase, and HTTPS requests to the server will be rejected.

E.g., the client can iterate on chunks of the file, and send them to the proxy server while asking the service to extend the session:

```
path = "/path/to/image"
MB_per_request = 32
with open(path, "rb") as disk:
   size = os.path.getsize(path)
   chunk_size = 1024*1024*MB_per_request
   pos = 0
   while (pos < size):
      transfer_service.extend()
      upload_headers['Content-Range'] = "bytes %d-%d/%d" % (pos, min(pos + chunk_size,
size)-1, size)
      proxy_connection.request(
         'PUT',
         url.path,
         disk.read(chunk_size),
         headers=upload_headers
      )
      r = proxy_connection.getresponse()
      print r.status, r.reason, "Completed", "{:.0%}".format(pos/ float(size))
      pos += chunk_size
```

When finishing the transfer, the user should call finalize. This will make the final adjustments and

verifications for finishing the transfer process.

For example:

```
transfer_service.finalize()
```

In case of an error, the transfer's phase will be changed to finished_failure, and the disk's status will be changed to `Illegal`. Otherwise it will be changed to finished_success, and the disk will be ready to be used. In both cases, the transfer entity will be removed shortly after.

Table 315. Methods summary

Name	Summary
`extend`	Extend the image transfer session.
`finalize`	After finishing to transfer the data, finalize the transfer.
`get`	Get the image transfer entity.
`pause`	Pause the image transfer session.
`resume`	Resume the image transfer session.

5.101.1. extend POST

Extend the image transfer session.

5.101.2. finalize POST

After finishing to transfer the data, finalize the transfer.

This will make sure that the data being transferred is valid and fits the image entity that was targeted in the transfer. Specifically, will verify that if the image entity is a QCOW disk, the data uploaded is indeed a QCOW file, and that the image doesn't have a backing file.

5.101.3. get GET

Get the image transfer entity.

Table 316. Parameters summary

Name	Type	Direction	Summary
`image_transfer`	ImageTransfer	Out	

5.101.4. pause POST

Pause the image transfer session.

5.101.5. resume POST

Resume the image transfer session. The client will need to poll the transfer's phase until it is different than `resuming`. For example:

```
transfer_service = transfers_service.image_transfer_service(transfer.id)
transfer_service.resume()
transfer = transfer_service.get()

while transfer.phase == types.ImageTransferPhase.RESUMING:
   time.sleep(1)
   transfer = transfer_service.get()
```

5.102. ImageTransfers

This service manages image transfers, for performing Image I/O API in oVirt. Please refer to image transfer for further documentation.

Table 317. Methods summary

Name	Summary
`add`	Add a new image transfer.
`list`	Retrieves the list of image transfers that are currently being performed.

5.102.1. add POST

Add a new image transfer. An image needs to be specified in order to make a new transfer.

Table 318. Parameters summary

Name	Type	Direction	Summary
`image_transfer`	ImageTransfer	In/Out	

5.102.2. list GET

Retrieves the list of image transfers that are currently being performed.

Table 319. Parameters summary

Name	Type	Direction	Summary
`image_transfer`	ImageTransfer[]	Out	

5.103. Images

Table 320. Methods summary

Name	Summary
`list`	

5.103.1. list GET

Table 321. Parameters summary

Name	Type	Direction	Summary
`images`	Image[]	Out	
`max`	Integer	In	Sets the maximum number of images to return.

max

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

5.104. InstanceType

Table 322. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.104.1. get GET

Table 323. Parameters summary

Name	Type	Direction	Summary
`instance_type`	InstanceType	Out	

5.104.2. remove DELETE

Table 324. Parameters summary

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

5.104.3. update PUT

Table 325. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`instance_type`	InstanceType	In/Out	

5.105. InstanceTypeNic

Table 326. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.105.1. get GET

Table 327. Parameters summary

Name	Type	Direction	Summary
`nic`	Nic	Out	

5.105.2. remove DELETE

Table 328. Parameters summary

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

5.105.3. update PUT

Table 329. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`nic`	Nic	In/Out	

5.106. InstanceTypeNics

Table 330. Methods summary

Name	Summary
`add`	

Name	Summary
list	

5.106.1. add POST

Table 331. Parameters summary

Name	Type	Direction	Summary
nic	Nic	In/Out	

5.106.2. list GET

Table 332. Parameters summary

Name	Type	Direction	Summary
max	Integer	In	Sets the maximum number of NICs to return.
nics	Nic[]	Out	

max

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

5.107. InstanceTypeWatchdog

Table 333. Methods summary

Name	Summary
get	
remove	
update	

5.107.1. get GET

Table 334. Parameters summary

Name	Type	Direction	Summary
watchdog	Watchdog	Out	

5.107.2. remove DELETE

Table 335. Parameters summary

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

5.107.3. update PUT

Table 336. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`watchdog`	Watchdog	In/Out	

5.108. InstanceTypeWatchdogs

Table 337. Methods summary

Name	Summary
`add`	
`list`	

5.108.1. add POST

Table 338. Parameters summary

Name	Type	Direction	Summary
`watchdog`	Watchdog	In/Out	

5.108.2. list GET

Table 339. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of watchdogs to return.
`watchdogs`	Watchdog[]	Out	

max

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

5.109. InstanceTypes

Table 340. Methods summary

Name	Summary
`add`	
`list`	

5.109.1. add POST

Table 341. Parameters summary

Name	Type	Direction	Summary
`instance_type`	InstanceType	In/Out	

5.109.2. list GET

Table 342. Parameters summary

Name	Type	Direction	Summary
`instance_type`	InstanceType[]	Out	
`max`	Integer	In	Sets the maximum number of instance types to return.

max

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

5.110. IscsiBond

Table 343. Methods summary

Name	Summary
`get`	
`remove`	Removes of an exisitng iSCSI bond.
`update`	Updates an iSCSI bond.

5.110.1. get GET

Table 344. Parameters summary

Name	Type	Direction	Summary
`bond`	IscsiBond	Out	

5.110.2. remove DELETE

Removes of an exisitng iSCSI bond.

For example, to remove the iSCSI bond `456` send a request like this:

```
DELETE /ovirt-engine/api/datacenters/123/iscsibonds/456
```

Table 345. Parameters summary

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

5.110.3. update PUT

Updates an iSCSI bond.

Updating of an iSCSI bond can be done on the `name` and the `description` attributes only. For example, to update the iSCSI bond `456` of data center `123`, send a request like this:

```
PUT /ovirt-engine/api/datacenters/123/iscsibonds/1234
```

The request body should look like this:

```
<iscsi_bond>
   <name>mybond</name>
   <description>My iSCSI bond</description>
</iscsi_bond>
```

Table 346. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`bond`	IscsiBond	In/Out	

5.111. IscsiBonds

Table 347. Methods summary

Name	Summary
`add`	Create a new iSCSI bond on a data center.
`list`	

5.111.1. add POST

Create a new iSCSI bond on a data center.

For example, to create a new iSCSI bond on data center `123` using storage connections `456` and `789`, send a request like this:

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

The request body should look like this:

```
<iscsi_bond>
  <name>mybond</name>
  <storage_connections>
    <storage_connection id="456"/>
    <storage_connection id="789"/>
  </storage_connections>
  <networks>
    <network id="abc"/>
  </networks>
</iscsi_bond>
```

Table 348. Parameters summary

Name	Type	Direction	Summary
bond	IscsiBond	In/Out	

5.111.2. list GET

Table 349. Parameters summary

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

max

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

5.112. Job

Table 350. Methods summary

Name	Summary
clear	
end	
get	

5.112.1. clear POST

Table 351. Parameters summary

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

5.112.2. end POST

Table 352. Parameters summary

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

5.112.3. get GET

Table 353. Parameters summary

Name	Type	Direction	Summary
`job`	Job	Out	

5.113. Jobs

Table 354. Methods summary

Name	Summary
`add`	
`list`	

5.113.1. add POST

Table 355. Parameters summary

Name	Type	Direction	Summary
`job`	Job	In/Out	

5.113.2. list GET

Table 356. Parameters summary

Name	Type	Direction	Summary
`jobs`	Job[]	Out	
`max`	Integer	In	Sets the maximum number of jobs to return.

max

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

5.114. KatelloErrata

Table 357. Methods summary

Name	Summary
`list`	

5.114.1. list GET

Table 358. Parameters summary

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

max

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

5.115. KatelloErratum

Table 359. Methods summary

Name	Summary
`get`	

5.115.1. get GET

Table 360. Parameters summary

Name	Type	Direction	Summary
`erratum`	KatelloErratum	Out	

5.116. MacPool

Table 361. Methods summary

Name	Summary
`get`	
`remove`	Removes a MAC address pool.

Name	Summary
`update`	Updates a MAC address pool.

5.116.1. get GET

Table 362. Parameters summary

Name	Type	Direction	Summary
`pool`	MacPool	Out	

5.116.2. remove DELETE

Removes a MAC address pool.

For example, to remove the MAC address pool having id `123` send a request like this:

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

Table 363. Parameters summary

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

5.116.3. update PUT

Updates a MAC address pool.

The `name`, `description`, `allow_duplicates`, and `ranges` attributes can be updated.

For example, to update the MAC address pool of id `123` send a request like this:

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

With a request body like this:

```
<mac_pool>
  <name>UpdatedMACPool</name>
  <description>An updated MAC address pool</description>
  <allow_duplicates>false</allow_duplicates>
  <ranges>
    <range>
      <from>00:1A:4A:16:01:51</from>
      <to>00:1A:4A:16:01:e6</to>
    </range>
    <range>
      <from>02:1A:4A:01:00:00</from>
      <to>02:1A:4A:FF:FF:FF</to>
    </range>
  </ranges>
</mac_pool>
```

Table 364. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`pool`	MacPool	In/Out	

5.117. MacPools

Table 365. Methods summary

Name	Summary
`add`	Creates a new MAC address pool.
`list`	

5.117.1. add POST

Creates a new MAC address pool.

Creation of a MAC address pool requires values for the `name` and `ranges` attributes.

For example, to create MAC address pool send a request like this:

```
POST /ovirt-engine/api/macpools
```

With a request body like this:

```
<mac_pool>
  <name>MACPool</name>
  <description>A MAC address pool</description>
  <allow_duplicates>true</allow_duplicates>
  <default_pool>false</default_pool>
  <ranges>
    <range>
      <from>00:1A:4A:16:01:51</from>
      <to>00:1A:4A:16:01:e6</to>
    </range>
  </ranges>
</mac_pool>
```

Table 366. Parameters summary

Name	Type	Direction	Summary
pool	MacPool	In/Out	

5.117.2. list GET

Table 367. Parameters summary

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

max

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

5.118. Measurable

5.119. Moveable

Table 368. Methods summary

Name	Summary
move	

5.119.1. move POST

Table 369. Parameters summary

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

5.120. Network

Table 370. Methods summary

Name	Summary
`get`	
`remove`	Removes a logical network, or the association of a logical network to a data center.
`update`	Updates a logical network.

5.120.1. get GET

Table 371. Parameters summary

Name	Type	Direction	Summary
`network`	Network	Out	

5.120.2. remove DELETE

Removes a logical network, or the association of a logical network to a data center.

For example, to remove the logical network `123` send a request like this:

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

Each network is bound exactly to one data center. So if we disassociate network with data center it has the same result as if we would just remove that network. However it might be more specific to say we're removing network `456` of data center `123`.

For example, to remove the association of network `456` to data center `123` send a request like this:

```
DELETE /ovirt-engine/api/datacenters/123/networks/456
```

Table 372. Parameters summary

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

5.120.3. update PUT

Updates a logical network.

The `name`, `description`, `ip`, `vlan`, `stp` and `display` attributes can be updated.

For example, to update the description of the logcial network `123` send a request like this:

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

With a request body like this:

```
<network>
  <description>My updated description</description>
</network>
```

The maximum transmission unit of a network is set using a PUT request to specify the integer value of the `mtu` attribute.

For example, to set the maximum transmission unit send a request like this:

```
PUT /ovirt-engine/api/datacenters/123/networks/456
```

With a request body like this:

```
<network>
  <mtu>1500</mtu>
</network>
```

Table 373. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`network`	Network	In/Out	

5.121. NetworkAttachment

Table 374. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.121.1. get GET

Table 375. Parameters summary

Name	Type	Direction	Summary
`attachment`	NetworkAttachment	Out	

5.121.2. remove DELETE

Table 376. Parameters summary

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

5.121.3. update PUT

Table 377. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`attachment`	NetworkAttachment	In/Out	

5.122. NetworkAttachments

Table 378. Methods summary

Name	Summary
`add`	
`list`	

5.122.1. add POST

Table 379. Parameters summary

Name	Type	Direction	Summary
`attachment`	NetworkAttachment	In/Out	

5.122.2. list GET

Table 380. Parameters summary

Name	Type	Direction	Summary
`attachments`	NetworkAttachment[]	Out	

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

max

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

5.123. NetworkFilter

Manages a network filter.

```
<network_filter id="00000019-0019-0019-0019-00000000026b">
  <name>example-network-filter-b</name>
  <version>
    <major>4</major>
    <minor>0</minor>
    <build>-1</build>
    <revision>-1</revision>
  </version>
</network_filter>
```

Please note that version is referring to the minimal support version for the specific filter.

Table 381. Methods summary

Name	Summary
get	Retrieves a representation of the network filter.

5.123.1. get GET

Retrieves a representation of the network filter.

Table 382. Parameters summary

Name	Type	Direction	Summary
network_filter	NetworkFilter	Out	

5.124. NetworkFilters

Represents a readonly network filters sub-collection.

The network filter enables to filter packets send to/from the VM's nic according to defined rules. For more information please refer to NetworkFilter service documentation

Network filters are supported in different versions, starting from version 3.0.

A network filter is defined for each vnic profile.

A vnic profile is defined for a specific network.

A network can be assigned to several different clusters. In the future, each network will be defined in cluster level.

Currently, each network is being defined at data center level. Potential network filters for each network are determined by the network's data center compatibility version V. V must be >= the network filter version in order to configure this network filter for a specific network. Please note, that if a network is assigned to cluster with a version supporting a network filter, the filter may not be available due to the data center version being smaller then the network filter's version.

Example of listing all of the supported network filters for a specific cluster:

```
GET http://localhost:8080/ovirt-engine/api/clusters/{cluster:id}/networkfilters
```

Output:

```xml
<network_filters>
  <network_filter id="00000019-0019-0019-0019-00000000026c">
    <name>example-network-filter-a</name>
    <version>
      <major>4</major>
      <minor>0</minor>
      <build>-1</build>
      <revision>-1</revision>
    </version>
  </network_filter>
  <network_filter id="00000019-0019-0019-0019-00000000026b">
    <name>example-network-filter-b</name>
    <version>
      <major>4</major>
      <minor>0</minor>
      <build>-1</build>
      <revision>-1</revision>
    </version>
  </network_filter>
  <network_filter id="00000019-0019-0019-0019-00000000026a">
    <name>example-network-filter-a</name>
    <version>
      <major>3</major>
      <minor>0</minor>
      <build>-1</build>
      <revision>-1</revision>
    </version>
  </network_filter>
</network_filters>
```

Table 383. Methods summary

Name	Summary
`list`	Retrieves the representations of the network filters.

5.124.1. list GET

Retrieves the representations of the network filters.

Table 384. Parameters summary

Name	Type	Direction	Summary
`filters`	NetworkFilter[]	Out	

5.125. NetworkLabel

Table 385. Methods summary

Name	Summary
`get`	
`remove`	Removes a label from a logical network.

5.125.1. get GET

Table 386. Parameters summary

Name	Type	Direction	Summary
`label`	NetworkLabel	Out	

5.125.2. remove DELETE

Removes a label from a logical network.

For example, to remove the label `exemplary` from a logical network having id `123` send the following request:

```
DELETE /ovirt-engine/api/networks/123/labels/exemplary
```

Table 387. Parameters summary

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

5.126. NetworkLabels

Table 388. Methods summary

Name	Summary
`add`	Attaches label to logical network.
`list`	

5.126.1. add POST

Attaches label to logical network.

You can attach labels to a logical network to automate the association of that logical network with physical host network interfaces to which the same label has been attached.

For example, to attach the label `mylabel` to a logical network having id `123` send a request like this:

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

With a request body like this:

```
<label id="mylabel"/>
```

Table 389. Parameters summary

Name	Type	Direction	Summary
label	NetworkLabel	In/Out	

5.126.2. list GET

Table 390. Parameters summary

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

max

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

5.127. Networks

Manages logical networks.

The engine creates a default `ovirtmgmt` network on installation. This network acts as the management network for access to hypervisor hosts. This network is associated with the `Default` cluster and is a member of the `Default` data center.

Table 391. Methods summary

Name	Summary
add	Creates a new logical network, or associates an existing network with a data center.
list	List logical networks.

5.127.1. add POST

Creates a new logical network, or associates an existing network with a data center.

Creation of a new network requires the `name` and `data_center` elements.

For example, to create a network named `mynetwork` for data center `123` send a request like this:

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

With a request body like this:

```
<network>
  <name>mynetwork</name>
  <data_center id="123"/>
</network>
```

To associate the existing network `456` with the data center `123` send a request like this:

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

With a request body like this:

```
<network>
  <name>ovirtmgmt</name>
</network>
```

Table 392. Parameters summary

Name	Type	Direction	Summary
network	Network	In/Out	

5.127.2. list GET

List logical networks.

For example:

```
GET /ovirt-engine/api/networks
```

Will respond:

```
<networks>
  <network href="/ovirt-engine/api/networks/123" id="123">
    <name>ovirtmgmt</name>
    <description>Default Management Network</description>
    <link href="/ovirt-engine/api/networks/123/permissions" rel="permissions"/>
    <link href="/ovirt-engine/api/networks/123/vnicprofiles" rel="vnicprofiles"/>
    <link href="/ovirt-engine/api/networks/123/networklabels" rel="networklabels"/>
    <mtu>0</mtu>
    <stp>false</stp>
    <usages>
      <usage>vm</usage>
    </usages>
    <data_center href="/ovirt-engine/api/datacenters/456" id="456"/>
  </network>
  ...
</networks>
```

Table 393. Parameters summary

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

case_sensitive

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

max

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

5.128. OpenstackImage

Table 394. Methods summary

Name	Summary
`get`	
`import`	Imports a virtual machine from a Glance image storage domain.

5.128.1. get GET

Table 395. Parameters summary

Name	Type	Direction	Summary
image	OpenStackImage	Out	

5.128.2. import POST

Imports a virtual machine from a Glance image storage domain.

For example, to import the image with identifier 456 from the storage domain with identifier 123 send a request like this:

```
POST /ovirt-engine/api/openstackimageproviders/123/images/456/import
```

With a request body like this:

```
<action>
  <storage_domain>
    <name>images0</name>
  </storage_domain>
  <cluster>
    <name>images0</name>
  </cluster>
</action>
```

Table 396. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the import should be performed asynchronously.
cluster	Cluster	In	This parameter is mandatory in case of using import_as_template and indicates which cluster should be used for import glance image as template.
disk	Disk	In	
import_as_template	Boolean	In	Indicates whether the image should be imported as a template.
storage_domain	StorageDomain	In	
template	Template	In	

5.129. OpenstackImageProvider

Table 397. Methods summary

Name	Summary
`get`	
`importcertificates`	
`remove`	
`testconnectivity`	
`update`	

5.129.1. get GET

Table 398. Parameters summary

Name	Type	Direction	Summary
`provider`	OpenStackImageProvider	Out	

5.129.2. importcertificates POST

Table 399. Parameters summary

Name	Type	Direction	Summary
`certificates`	Certificate[]	In	

5.129.3. remove DELETE

Table 400. Parameters summary

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

5.129.4. testconnectivity POST

Table 401. Parameters summary

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

5.129.5. update PUT

Table 402. Parameters summary

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

5.130. OpenstackImageProviders

Table 403. Methods summary

Name	Summary
`add`	
`list`	

5.130.1. add POST

Table 404. Parameters summary

Name	Type	Direction	Summary
`provider`	OpenStackImageProvider	In/Out	

5.130.2. list GET

Table 405. Parameters summary

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

max

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

5.131. OpenstackImages

Table 406. Methods summary

Name	Summary
`list`	Lists the images of a Glance image storage domain.

5.131.1. list GET

Lists the images of a Glance image storage domain.

Table 407. Parameters summary

Name	Type	Direction	Summary
`images`	OpenStackImage[]	Out	
`max`	Integer	In	Sets the maximum number of images to return.

max

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

5.132. OpenstackNetwork

Table 408. Methods summary

Name	Summary
`get`	
`import`	This operation imports an external network into oVirt.

5.132.1. get GET

Table 409. Parameters summary

Name	Type	Direction	Summary
`network`	OpenStackNetwork	Out	

5.132.2. import POST

This operation imports an external network into oVirt. The network will be added to the data center specified.

Table 410. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the import should be performed asynchronously.
`data_center`	DataCenter	In	The data center into which the network is to be imported.

data_center

The data center into which the network is to be imported. Data center is mandatory, and can be

specified using the `id` or `name` attributes, the rest of the attributes will be ignored.

5.133. OpenstackNetworkProvider

Table 411. Methods summary

Name	Summary
`get`	
`importcertificates`	
`remove`	
`testconnectivity`	
`update`	

5.133.1. get GET

Table 412. Parameters summary

Name	Type	Direction	Summary
`provider`	OpenStackNetworkProvider	Out	

5.133.2. importcertificates POST

Table 413. Parameters summary

Name	Type	Direction	Summary
`certificates`	Certificate[]	In	

5.133.3. remove DELETE

Table 414. Parameters summary

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

5.133.4. testconnectivity POST

Table 415. Parameters summary

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

5.133.5. update PUT

Table 416. Parameters summary

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

5.134. OpenstackNetworkProviders

Table 417. Methods summary

Name	Summary
`add`	The operation adds a new network provider to the system.
`list`	

5.134.1. add POST

The operation adds a new network provider to the system. If the `type` property is not present, a default value of `NEUTRON` will be used.

Table 418. Parameters summary

Name	Type	Direction	Summary
`provider`	OpenStackNetworkProvider	In/Out	

5.134.2. list GET

Table 419. Parameters summary

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

max

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

5.135. OpenstackNetworks

Table 420. Methods summary

Name	Summary
`list`	

5.135.1. list GET

Table 421. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of networks to return.
`networks`	OpenStackNetwork[]	Out	

max

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

5.136. OpenstackSubnet

Table 422. Methods summary

Name	Summary
`get`	
`remove`	

5.136.1. get GET

Table 423. Parameters summary

Name	Type	Direction	Summary
`subnet`	OpenStackSubnet	Out	

5.136.2. remove DELETE

Table 424. Parameters summary

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

5.137. OpenstackSubnets

Table 425. Methods summary

Name	Summary
`add`	
`list`	

5.137.1. add POST

Table 426. Parameters summary

Name	Type	Direction	Summary
`subnet`	OpenStackSubnet	In/Out	

5.137.2. list GET

Table 427. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of sub-networks to return.
`subnets`	OpenStackSubnet[]	Out	

max

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

5.138. OpenstackVolumeAuthenticationKey

Table 428. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.138.1. get GET

Table 429. Parameters summary

Name	Type	Direction	Summary
`key`	OpenstackVolumeAuthenticationKey	Out	

5.138.2. remove DELETE

Table 430. Parameters summary

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

5.138.3. update PUT

Table 431. Parameters summary

Name	Type	Direction	Summary
`key`	OpenstackVolumeAuthenticationKey	In/Out	

5.139. OpenstackVolumeAuthenticationKeys

Table 432. Methods summary

Name	Summary
`add`	
`list`	

5.139.1. add POST

Table 433. Parameters summary

Name	Type	Direction	Summary
`key`	OpenstackVolumeAuthenticationKey	In/Out	

5.139.2. list GET

Table 434. Parameters summary

Name	Type	Direction	Summary
`keys`	OpenstackVolumeAuthenticationKey[]	Out	

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

max

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

5.140. OpenstackVolumeProvider

Table 435. Methods summary

Name	Summary
`get`	
`importcertificates`	
`remove`	
`testconnectivity`	
`update`	

5.140.1. get GET

Table 436. Parameters summary

Name	Type	Direction	Summary
`provider`	OpenStackVolumeProvider	Out	

5.140.2. importcertificates POST

Table 437. Parameters summary

Name	Type	Direction	Summary
`certificates`	Certificate[]	In	

5.140.3. remove DELETE

Table 438. Parameters summary

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

5.140.4. testconnectivity POST

Table 439. Parameters summary

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

5.140.5. update PUT

Table 440. Parameters summary

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

5.141. OpenstackVolumeProviders

Table 441. Methods summary

Name	Summary
`add`	Adds a new volume provider.
`list`	Retrieves the list of volume providers.

5.141.1. add POST

Adds a new volume provider.

For example:

```
POST /ovirt-engine/api/openstackvolumeproviders
```

With a request body like this:

```
<openstack_volume_provider>
  <name>mycinder</name>
  <url>https://mycinder.example.com:8776</url>
  <data_center>
    <name>mydc</name>
  </data_center>
  <requires_authentication>true</requires_authentication>
  <username>admin</username>
  <password>mypassword</password>
  <tenant_name>mytenant</tenant_name>
</openstack_volume_provider>
```

Table 442. Parameters summary

Name	Type	Direction	Summary
provider	OpenStackVolumeProvider	In/Out	

5.141.2. list GET

Retrieves the list of volume providers.

Table 443. Parameters summary

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

max

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

5.142. OpenstackVolumeType

Table 444. Methods summary

Name	Summary
get	

5.142.1. get GET

Table 445. Parameters summary

Name	Type	Direction	Summary
type	OpenStackVolumeType	Out	

5.143. OpenstackVolumeTypes

Table 446. Methods summary

Name	Summary
list	

5.143.1. list GET

Table 447. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of volume types to return.
`types`	OpenStackVolumeType[]	Out	

max

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

5.144. OperatingSystem

Table 448. Methods summary

Name	Summary
`get`	

5.144.1. get GET

Table 449. Parameters summary

Name	Type	Direction	Summary
`operating_system`	OperatingSystemInfo	Out	

5.145. OperatingSystems

Table 450. Methods summary

Name	Summary
`list`	

5.145.1. list GET

Table 451. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of networks to return.
`operating_system`	OperatingSystemInfo[]	Out	

max

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

5.146. Permission

Table 452. Methods summary

Name	Summary
`get`	
`remove`	

5.146.1. get GET

Table 453. Parameters summary

Name	Type	Direction	Summary
`permission`	Permission	Out	

5.146.2. remove DELETE

Table 454. Parameters summary

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

5.147. Permit

Table 455. Methods summary

Name	Summary
`get`	
`remove`	

5.147.1. get GET

Table 456. Parameters summary

Name	Type	Direction	Summary
`permit`	Permit	Out	

5.147.2. remove DELETE

Table 457. Parameters summary

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

5.148. Permits

Table 458. Methods summary

Name	Summary
`add`	Adds a permit to the set aggregated by parent role.
`list`	

5.148.1. add POST

Adds a permit to the set aggregated by parent role. The permit must be one retrieved from the capabilities resource.

Table 459. Parameters summary

Name	Type	Direction	Summary
`permit`	Permit	In/Out	The permit to add.

5.148.2. list GET

Table 460. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of permits to return.
`permits`	Permit[]	Out	

max

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

5.149. Qos

Table 461. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.149.1. get GET

Table 462. Parameters summary

Name	Type	Direction	Summary
`qos`	Qos	Out	

5.149.2. remove DELETE

Table 463. Parameters summary

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

5.149.3. update PUT

Table 464. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`qos`	Qos	In/Out	

5.150. Qoss

Table 465. Methods summary

Name	Summary
`add`	
`list`	

5.150.1. add POST

Table 466. Parameters summary

Name	Type	Direction	Summary
`qos`	Qos	In/Out	

5.150.2. list GET

Table 467. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of QoS descriptors to return.
`qoss`	Qos[]	Out	

max

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

5.151. Quota

Table 468. Methods summary

Name	Summary
`get`	Retrieves a quota.
`remove`	Delete a quota.
`update`	Updates a quota.

5.151.1. get GET

Retrieves a quota.

An example of retrieving a quota:

```
GET /ovirt-engine/api/datacenters/123/quotas/456
```

```
<quota id="456">
  <name>myquota</name>
  <description>My new quota for virtual machines</description>
  <cluster_hard_limit_pct>20</cluster_hard_limit_pct>
  <cluster_soft_limit_pct>80</cluster_soft_limit_pct>
  <storage_hard_limit_pct>20</storage_hard_limit_pct>
  <storage_soft_limit_pct>80</storage_soft_limit_pct>
</quota>
```

Table 469. Parameters summary

Name	Type	Direction	Summary
`quota`	Quota	Out	

5.151.2. remove DELETE

Delete a quota.

An example of deleting a quota:

```
DELETE /ovirt-engine/api/datacenters/123-456/quotas/654-321
-0472718ab224 HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

Table 470. Parameters summary

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

5.151.3. update PUT

Updates a quota.

An example of updating a quota:

```
PUT /ovirt-engine/api/datacenters/123/quotas/456
```

```
<quota>
  <cluster_hard_limit_pct>30</cluster_hard_limit_pct>
  <cluster_soft_limit_pct>70</cluster_soft_limit_pct>
  <storage_hard_limit_pct>20</storage_hard_limit_pct>
  <storage_soft_limit_pct>80</storage_soft_limit_pct>
</quota>
```

Table 471. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`quota`	Quota	In/Out	

5.152. QuotaClusterLimit

Table 472. Methods summary

Name	Summary
`get`	
`remove`	

5.152.1. get GET

Table 473. Parameters summary

Name	Type	Direction	Summary
`limit`	QuotaClusterLimit	Out	

5.152.2. remove DELETE

Table 474. Parameters summary

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

5.153. QuotaClusterLimits

Table 475. Methods summary

Name	Summary
`add`	
`list`	

5.153.1. add POST

Table 476. Parameters summary

Name	Type	Direction	Summary
`limit`	QuotaClusterLimit	In/Out	

5.153.2. list GET

Table 477. Parameters summary

Name	Type	Direction	Summary
`limits`	QuotaClusterLimit[]	Out	
`max`	Integer	In	Sets the maximum number of limits to return.

max

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

5.154. QuotaStorageLimit

Table 478. Methods summary

Name	Summary
`get`	
`remove`	

5.154.1. get GET

Table 479. Parameters summary

Name	Type	Direction	Summary
`limit`	QuotaStorageLimit	Out	

5.154.2. remove DELETE

Table 480. Parameters summary

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

5.155. QuotaStorageLimits

Table 481. Methods summary

Name	Summary
`add`	
`list`	

5.155.1. add POST

Table 482. Parameters summary

Name	Type	Direction	Summary
`limit`	QuotaStorageLimit	In/Out	

5.155.2. list GET

Table 483. Parameters summary

Name	Type	Direction	Summary
`limits`	QuotaStorageLimit[]	Out	
`max`	Integer	In	Sets the maximum number of limits to return.

max

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

5.156. Quotas

Table 484. Methods summary

Name	Summary
`add`	Creates a new quota.
`list`	Lists quotas of a data center

5.156.1. add POST

Creates a new quota.

An example of creating a new quota:

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

```
<quota>
  <name>myquota</name>
  <description>My new quota for virtual machines</description>
</quota>
```

Table 485. Parameters summary

Name	Type	Direction	Summary
`quota`	Quota	In/Out	

5.156.2. list GET

Lists quotas of a data center

Table 486. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of quota descriptors to return.
`quotas`	Quota[]	Out	

max

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

5.157. Role

Table 487. Methods summary

Name	Summary
`get`	
`remove`	Removes the role.
`update`	Updates a role.

5.157.1. get GET

Table 488. Parameters summary

Name	Type	Direction	Summary
`role`	Role	Out	

5.157.2. remove DELETE

Removes the role.

To remove the role you need to know its id, then send request like this:

```
DELETE /ovirt-engine/api/roles/{role_id}
```

Table 489. Parameters summary

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

5.157.3. update PUT

Updates a role. You are allowed to update `name`, `description` and `administrative` attributes after role is created. Within this endpoint you can't add or remove roles permits you need to use service that manages permits of role.

For example to update role's `name`, `description` and `administrative` attributes send a request like this:

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

With a request body like this:

```
<role>
  <name>MyNewRoleName</name>
  <description>My new description of the role</description>
  <administrative>true</administrative>
</group>
```

Table 490. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`role`	Role	In/Out	

5.158. Roles

Provides read-only access to the global set of roles

Table 491. Methods summary

Name	Summary
`add`	Create a new role.
`list`	

5.158.1. add POST

Create a new role. The role can be administrative or non-administrative and can have different permits.

For example, to add the `MyRole` non-administrative role with permits to login and create virtual machines send a request like this (note that you have to pass permit id):

```
POST /ovirt-engine/api/roles
```

With a request body like this:

```
<role>
  <name>MyRole</name>
  <description>My custom role to create virtual machines</description>
  <administrative>false</administrative>
  <permits>
    <permit id="1"/>
    <permit id="1300"/>
  </permits>
</group>
```

Table 492. Parameters summary

Name	Type	Direction	Summary
`role`	Role	In/Out	

5.158.2. list GET

Table 493. Parameters summary

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

max

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

5.159. SchedulingPolicies

Table 494. Methods summary

Name	Summary
`add`	
`list`	

5.159.1. add POST

Table 495. Parameters summary

Name	Type	Direction	Summary
`policy`	SchedulingPolicy	In/Out	

5.159.2. list GET

Table 496. Parameters summary

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

max

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

5.160. SchedulingPolicy

Table 497. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.160.1. get GET

Table 498. Parameters summary

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

5.160.2. remove DELETE

Table 499. Parameters summary

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

5.160.3. update PUT

Table 500. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`policy`	SchedulingPolicy	In/Out	

5.161. SchedulingPolicyUnit

Table 501. Methods summary

Name	Summary
`get`	
`remove`	

5.161.1. get GET

Table 502. Parameters summary

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

5.161.2. remove DELETE

Table 503. Parameters summary

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

5.162. SchedulingPolicyUnits

Table 504. Methods summary

Name	Summary
`list`	

5.162.1. list GET

Table 505. Parameters summary

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

max

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

5.163. Snapshot

Table 506. Methods summary

Name	Summary
`get`	
`remove`	

Name	Summary
`restore`	Restores a virtual machine snapshot.

5.163.1. get GET

Table 507. Parameters summary

Name	Type	Direction	Summary
`snapshot`	Snapshot	Out	

5.163.2. remove DELETE

Table 508. Parameters summary

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

5.163.3. restore POST

Restores a virtual machine snapshot.

For example, to restore the snapshot with identifier `456` of virtual machine with identifier `123` send a request like this:

```
POST /ovirt-engine/api/vms/123/snapshots/456/restore
```

With an empty `action` in the body:

```
<action/>
```

Table 509. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the restore should be performed asynchronously.
`disks`	Disk[]	In	
`restore_memory`	Boolean	In	

5.164. SnapshotCdrom

Table 510. Methods summary

Name	Summary
`get`	

5.164.1. get GET

Table 511. Parameters summary

Name	Type	Direction	Summary
`cdrom`	Cdrom	Out	

5.165. SnapshotCdroms

Table 512. Methods summary

Name	Summary
`list`	

5.165.1. list GET

Table 513. Parameters summary

Name	Type	Direction	Summary
`cdroms`	Cdrom[]	Out	
`max`	Integer	In	Sets the maximum number of CDROMS to return.

max

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

5.166. SnapshotDisk

Table 514. Methods summary

Name	Summary
`get`	

5.166.1. get GET

Table 515. Parameters summary

Name	Type	Direction	Summary
`disk`	Disk	Out	

5.167. SnapshotDisks

Table 516. Methods summary

Name	Summary
`list`	

5.167.1. list GET

Table 517. Parameters summary

Name	Type	Direction	Summary
`disks`	Disk[]	Out	
`max`	Integer	In	Sets the maximum number of disks to return.

max

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

5.168. SnapshotNic

Table 518. Methods summary

Name	Summary
`get`	

5.168.1. get GET

Table 519. Parameters summary

Name	Type	Direction	Summary
`nic`	Nic	Out	

5.169. SnapshotNics

Table 520. Methods summary

Name	Summary
`list`	

5.169.1. list GET

Table 521. Parameters summary

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

Name	Type	Direction	Summary
nics	Nic[]	Out	

max

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

5.170. Snapshots

Table 522. Methods summary

Name	Summary
add	Creates a virtual machine snapshot.
list	

5.170.1. add POST

Creates a virtual machine snapshot.

For example, to create a new snapshot for virtual machine 123 send a request like this:

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

With a request body like this:

```
<snapshot>
  <description>My snapshot</description>
</snapshot>
```

Table 523. Parameters summary

Name	Type	Direction	Summary
snapshot	Snapshot	In/Out	

5.170.2. list GET

Table 524. Parameters summary

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

max

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

5.171. SshPublicKey

Table 525. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.171.1. get GET

Table 526. Parameters summary

Name	Type	Direction	Summary
`key`	SshPublicKey	Out	

5.171.2. remove DELETE

Table 527. Parameters summary

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

5.171.3. update PUT

Table 528. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`key`	SshPublicKey	In/Out	

5.172. SshPublicKeys

Table 529. Methods summary

Name	Summary
`add`	
`list`	

5.172.1. add POST

Table 530. Parameters summary

Name	Type	Direction	Summary
`key`	SshPublicKey	In/Out	

5.172.2. list GET

Table 531. Parameters summary

Name	Type	Direction	Summary
`keys`	SshPublicKey[]	Out	
`max`	Integer	In	Sets the maximum number of keys to return.

max

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

5.173. Statistic

Table 532. Methods summary

Name	Summary
`get`	

5.173.1. get GET

Table 533. Parameters summary

Name	Type	Direction	Summary
`statistic`	Statistic	In/Out	

5.174. Statistics

Table 534. Methods summary

Name	Summary
`list`	

5.174.1. list GET

Table 535. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of statistics to return.
`statistics`	Statistic[]	Out	

max

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

5.175. Step

Table 536. Methods summary

Name	Summary
`end`	
`get`	

5.175.1. end POST

Table 537. Parameters summary

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

5.175.2. get GET

Table 538. Parameters summary

Name	Type	Direction	Summary
`step`	Step	Out	

5.176. Steps

Table 539. Methods summary

Name	Summary
`add`	
`list`	

5.176.1. add POST

Table 540. Parameters summary

Name	Type	Direction	Summary
`step`	Step	In/Out	

5.176.2. list GET

Table 541. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of steps to return.
`steps`	Step[]	Out	

max

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

5.177. Storage

Table 542. Methods summary

Name	Summary
`get`	

5.177.1. get GET

Table 543. Parameters summary

Name	Type	Direction	Summary
`report_status`	Boolean	In	Indicates if the status of the LUNs in the storage should be checked.
`storage`	HostStorage	Out	

report_status

Indicates if the status of the LUNs in the storage should be checked. Checking the status of the LUN is an heavy weight operation and this data is not always needed by the user. This parameter will give the option to not perform the status check of the LUNs.

The default is `true` for backward compatibility.

Here an example with the LUN status :

```
<host_storage id="360014051136c20574f743bdbd28177fd">
  <logical_units>
    <logical_unit id="360014051136c20574f743bdbd28177fd">
      <lun_mapping>0</lun_mapping>
      <paths>1</paths>
      <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>
```

Here an example without the LUN status :

```
<host_storage id="360014051136c20574f743bdbd28177fd">
  <logical_units>
    <logical_unit id="360014051136c20574f743bdbd28177fd">
      <lun_mapping>0</lun_mapping>
      <paths>1</paths>
      <product_id>lun0</product_id>
      <serial>SLIO-ORG_lun0_1136c205-74f7-43bd-bd28-177fd5ce6993</serial>
      <size>10737418240</size>
      <vendor_id>LIO-ORG</vendor_id>
      <volume_group_id>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>
```

5.178. StorageDomain

Table 544. Methods summary

Name	Summary
`get`	
`isattached`	
`refreshluns`	This operation refreshes the LUN size.
`remove`	Removes the storage domain.
`update`	Updates a storage domain.

Name	Summary
updateovfstore	This operation forces the update of the OVF_STORE of this storage domain.

5.178.1. get GET

Table 545. Parameters summary

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

5.178.2. isattached POST

Table 546. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the action should be performed asynchronously.
host	Host	In	
is_attached	Boolean	Out	

5.178.3. refreshluns POST

This operation refreshes the LUN size.

After increasing the size of the underlying LUN on the storage server, the user can refresh the LUN size. This action forces a rescan of the provided LUNs and updates the database with the new size if required.

For example, in order to refresh the size of two LUNs send a request like this:

```
POST /ovirt-engine/api/storagedomains/262b056b-aede-40f1-9666-b883eff59d40/refreshluns
```

With a request body like this:

```
<action>
  <logical_units>
    <logical_unit id="1IET_00010001"/>
    <logical_unit id="1IET_00010002"/>
  </logical_units>
</action>
```

Table 547. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the refresh should be performed asynchronously.
`logical_units`	LogicalUnit[]	In	The LUNs that need to be refreshed.

5.178.4. remove DELETE

Removes the storage domain.

Without any special parameters, the storage domain is detached from the system and removed from the database. The storage domain can then be imported to the same or different setup, with all the data on it. If the storage isn't accessible the operation will fail.

If the `destroy` parameter is `true` then the operation will always succeed, even if the storage isn't accessible, the failure is just ignored and the storage domain is removed from the database anyway.

If the `format` parameter is `true` then the actual storage is formatted, and the metadata is removed from the LUN or directory, so it can no longer be imported to the same or a different setup.

Table 548. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the remove should be performed asynchronously.
`destroy`	Boolean	In	Indicates if the operation should succeed, and the storage domain removed from the database, even if the storage isn't accessible.
`format`	Boolean	In	Indicates if the actual storage should be formatted, removing all the metadata from the underlying LUN or directory: [source] ---- DELETE /ovirt-engine/api/storagedomains/123?format=true ---- This parameter is optional, and the default value is `false`.
`host`	String	In	Indicates what host should be used to remove the storage domain.

destroy

Indicates if the operation should succeed, and the storage domain removed from the database, even if the storage isn't accessible.

```
DELETE /ovirt-engine/api/storagedomains/123?destroy=true
```

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

host

Indicates what host should be used to remove the storage domain.

This parameter is mandatory, and it can contain the name or the identifier of the host. For example, to use the host named `myhost` to remove the storage domain with identifier `123` send a request like this:

```
DELETE /ovirt-engine/api/storagedomains/123?host=myhost
```

5.178.5. update PUT

Updates a storage domain.

Not all of the StorageDomain's attributes are updatable post-creation. Those that can be updated are: `name`, `description`, `comment`, `warning_low_space_indicator`, `critical_space_action_blocker` and `wipe_after_delete` (note that changing the `wipe_after_delete` attribute will not change the wipe after delete property of disks that already exist).

To update the `name` and `wipe_after_delete` attributes of a storage domain with an identifier `123`, send a request as follows:

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

With a request body as follows:

```
<storage_domain>
  <name>data2</name>
  <wipe_after_delete>true</wipe_after_delete>
</storage_domain>
```

Table 549. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`storage_domain`	StorageDomain	In/Out	

5.178.6. updateovfstore POST

This operation forces the update of the `OVF_STORE` of this storage domain.

The `OVF_STORE` is a disk image that contains the meta-data of virtual machines and disks that reside in the storage domain. This meta-data is used in case the domain is imported or exported to or from a different data center or a different installation.

By default the OVF_STORE is updated periodically (set by default to 60 minutes) but users might want to force an update after an important change, or when the they believe the OVF_STORE is corrupt.

When initiated by the user, OVF_STORE update will be performed whether an update is needed or not.

Table 550. Parameters summary

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

5.179. StorageDomainContentDisk

Table 551. Methods summary

Name	Summary
get	

5.179.1. get GET

Table 552. Parameters summary

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

5.180. StorageDomainContentDisks

Table 553. Methods summary

Name	Summary
list	

5.180.1. list GET

Table 554. Parameters summary

Name	Type	Direction	Summary
case_sensitive	Boolean	In	Indicates if the search performed using the search parameter should be performed taking case into account.
disks	Disk[]	Out	
max	Integer	In	Sets the maximum number of disks to return.

Name	Type	Direction	Summary
search	String	In	A query string used to restrict the returned disks.

case_sensitive

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

max

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

5.181. StorageDomainServerConnection

Table 555. Methods summary

Name	Summary
get	
remove	Detaches a storage connection from storage.

5.181.1. get GET

Table 556. Parameters summary

Name	Type	Direction	Summary
connection	StorageConnection	Out	

5.181.2. remove DELETE

Detaches a storage connection from storage.

Table 557. Parameters summary

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

5.182. StorageDomainServerConnections

Table 558. Methods summary

Name	Summary
add	

Name	Summary
`list`	

5.182.1. add POST

Table 559. Parameters summary

Name	Type	Direction	Summary
`connection`	StorageConnection	In/Out	

5.182.2. list GET

Table 560. Parameters summary

Name	Type	Direction	Summary
`connections`	StorageConnection[]	Out	
`max`	Integer	In	Sets the maximum number of connections to return.

max

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

5.183. StorageDomainTemplate

Table 561. Methods summary

Name	Summary
`get`	
`import`	
`register`	
`remove`	

5.183.1. get GET

Table 562. Parameters summary

Name	Type	Direction	Summary
`template`	Template	Out	

5.183.2. import POST

Table 563. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the import should be performed asynchronously.
`clone`	Boolean	In	
`cluster`	Cluster	In	
`exclusive`	Boolean	In	
`storage_domain`	StorageDomain	In	
`template`	Template	In	
`vm`	Vm	In	

5.183.3. register POST

Table 564. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the registration should be performed asynchronously.
`clone`	Boolean	In	
`cluster`	Cluster	In	
`exclusive`	Boolean	In	
`template`	Template	In	

5.183.4. remove DELETE

Table 565. Parameters summary

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

5.184. StorageDomainTemplates

Table 566. Methods summary

Name	Summary
`list`	

5.184.1. list GET

Table 567. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of templates to return.
`templates`	Template[]	Out	

max

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

5.185. StorageDomainVm

Table 568. Methods summary

Name	Summary
`get`	
`import`	Imports a virtual machine from an export storage domain.
`register`	
`remove`	Deletes a virtual machine from an export storage domain.

5.185.1. get GET

Table 569. Parameters summary

Name	Type	Direction	Summary
`vm`	Vm	Out	

5.185.2. import POST

Imports a virtual machine from an export storage domain.

For example, send a request like this:

```
POST /ovirt-engine/api/storagedomains/123/vms/456/import
```

With a request body like this:

```
<action>
  <storage_domain>
    <name>mydata</name>
  </storage_domain>
  <cluster>
    <name>mycluster</name>
  </cluster>
</action>
```

To import a virtual machine as a new entity add the `clone` parameter:

```
<action>
  <storage_domain>
    <name>mydata</name>
  </storage_domain>
  <cluster>
    <name>mycluster</name>
  </cluster>
  <clone>true</clone>
  <vm>
    <name>myvm</name>
  </vm>
</action>
```

Include an optional `disks` parameter to choose which disks to import. For example, to import the disks of the template that have the identifiers `123` and `456` send the following request body:

```
<action>
  <cluster>
    <name>mycluster</name>
  </cluster>
  <vm>
    <name>myvm</name>
  </vm>
  <disks>
    <disk id="123"/>
    <disk id="456"/>
  </disks>
</action>
```

Table 570. Parameters summary

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

Name	Type	Direction	Summary
`clone`	Boolean	In	Indicates if the identifiers of the imported virtual machine should be regenerated.
`cluster`	Cluster	In	
`collapse_snapshots`	Boolean	In	Indicates of the snapshots of the virtual machine that is imported should be collapsed, so that the result will be a virtual machine without snapshots.
`storage_domain`	StorageDomain	In	
`vm`	Vm	In	

clone

Indicates if the identifiers of the imported virtual machine should be regenerated.

By default when a virtual machine is imported the identifiers are preserved. This means that the same virtual machine can't be imported multiple times, as that identifiers needs to be unique. To allow importing the same machine multiple times set this parameter to `true`, as the default is `false`.

collapse_snapshots

Indicates of the snapshots of the virtual machine that is imported should be collapsed, so that the result will be a virtual machine without snapshots.

This parameter is optional, and if it isn't explicitly specified the default value is `false`.

5.185.3. register POST

Table 571. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the registration should be performed asynchronously.
`clone`	Boolean	In	
`cluster`	Cluster	In	
`vm`	Vm	In	

5.185.4. remove DELETE

Deletes a virtual machine from an export storage domain.

For example, to delete the virtual machine `456` from the storage domain `123`, send a request like this:

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

Table 572. Parameters summary

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

5.186. StorageDomainVmDiskAttachment

Returns the details of the disks attached to a virtual machine in the export domain.

Table 573. Methods summary

Name	Summary
`get`	Returns the details of the attachment with all its properties and a link to the disk.

5.186.1. get GET

Returns the details of the attachment with all its properties and a link to the disk.

Table 574. Parameters summary

Name	Type	Direction	Summary
`attachment`	DiskAttachment	Out	The disk attachment.

5.187. StorageDomainVmDiskAttachments

Returns the details of a disk attached to a virtual machine in the export domain.

Table 575. Methods summary

Name	Summary
`list`	List the disks that are attached to the virtual machine.

5.187.1. list GET

List the disks that are attached to the virtual machine.

Table 576. Parameters summary

Name	Type	Direction	Summary
`attachments`	DiskAttachment[]	Out	

5.188. StorageDomainVms

Lists the virtual machines of an export storage domain.

For example, to retrieve the virtual machines that are available in the storage domain with identifier 123 send the following request:

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

This will return the following response body:

```
<vms>
  <vm id="456" href="/api/storagedomains/123/vms/456">
    <name>vm1</name>
    ...
    <storage_domain id="123" href="/api/storagedomains/123"/>
    <actions>
      <link rel="import" href="/api/storagedomains/123/vms/456/import"/>
    </actions>
  </vm>
</vms>
```

Virtual machines and templates in these collections have a similar representation to their counterparts in the top-level Vm and Template collections, except they also contain a StorageDomain reference and an import action.

Table 577. Methods summary

Name	Summary
list	

5.188.1. list GET

Table 578. Parameters summary

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

max

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

5.189. StorageDomains

Table 579. Methods summary

Name	Summary
add	Adds a new storage domain.
list	

5.189.1. add POST

Adds a new storage domain.

Creation of a new StorageDomain requires the `name`, `type`, `host` and `storage` attributes. Identify the `host` attribute with the `id` or `name` attributes. In oVirt 3.6 and later you can enable the wipe after delete option by default on the storage domain. To configure this, specify `wipe_after_delete` in the POST request. This option can be edited after the domain is created, but doing so will not change the wipe after delete property of disks that already exist.

To add a new storage domain with specified `name`, `type`, `storage.type`, `storage.address` and `storage.path` and by using a host with an id `123`, send a request as follows:

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

With a request body as follows:

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

To create a new NFS ISO storage domain send a request like this:

```
<storage_domain>
  <name>myisos</name>
  <type>iso</type>
  <storage>
    <type>nfs</type>
    <address>mynfs.example.com</address>
    <path>/export/myisos</path>
  </storage>
  <host>
    <name>myhost</name>
  </host>
</storage_domain>
```

Table 580. Parameters summary

Name	Type	Direction	Summary
`storage_domain`	StorageDomain	In/Out	

5.189.2. list GET

Table 581. Parameters summary

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

case_sensitive

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

max

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

5.190. StorageServerConnection

Table 582. Methods summary

Name	Summary
`get`	
`remove`	Removes a storage connection.
`update`	Updates the storage connection.

5.190.1. get GET

Table 583. Parameters summary

Name	Type	Direction	Summary
`conection`	StorageConnection	Out	

5.190.2. remove DELETE

Removes a storage connection.

A storage connection can only be deleted if neither storage domain nor LUN disks reference it. The host name or id is optional; providing it disconnects (unmounts) the connection from that host.

Table 584. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the remove should be performed asynchronously.
`host`	String	In	The name or identifier of the host from which the connection would be unmounted (disconnected).

host

The name or identifier of the host from which the connection would be unmounted (disconnected). If not provided, no host will be disconnected.

For example, to use the host with identifier `456` to delete the storage connection with identifier `123` send a request like this:

```
DELETE /ovirt-engine/api/storageconnections/123?host=456
```

5.190.3. update PUT

Updates the storage connection.

For example, to change the address of the storage server send a request like this:

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

With a request body like this:

```
<storage_connection>
  <address>mynewnfs.example.com</address>
  <host>
    <name>myhost</name>
  </host>
</storage_connection>
```

Table 585. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`connection`	StorageConnection	In/Out	
`force`	Boolean	In	Indicates if the operation should succeed regardless to the relevant storage domain's status (i.

force

Indicates if the operation should succeed regardless to the relevant storage domain's status (i.e. updating is also applicable when storage domain's status is not maintenance).

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

5.191. StorageServerConnectionExtension

Table 586. Methods summary

Name	Summary
`get`	
`remove`	
`update`	Update a storage server connection extension for the given host.

5.191.1. get GET

Table 587. Parameters summary

Name	Type	Direction	Summary
extension	StorageConnectionExtension	Out	

5.191.2. remove DELETE

Table 588. Parameters summary

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

5.191.3. update PUT

Update a storage server connection extension for the given host.

To update the storage connection `456` of host `123` send a request like this:

```
PUT /ovirt-engine/api/hosts/123/storageconnectionextensions/456
```

With a request body like this:

```
<storage_connection_extension>
  <target>iqn.2016-01.com.example:mytarget</target>
  <username>myuser</username>
  <password>mypassword</password>
</storage_connection_extension>
```

Table 589. Parameters summary

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

5.192. StorageServerConnectionExtensions

Table 590. Methods summary

Name	Summary
add	Creates a new storage server connection extension for the given host.
list	

5.192.1. add POST

Creates a new storage server connection extension for the given host.

The extension lets the user define credentials for an iSCSI target for a specific host. For example to use `myuser` and `mypassword` as the credentials when connecting to the iSCSI target from host `123` send a request like this:

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

With a request body like this:

```
<storage_connection_extension>
  <target>iqn.2016-01.com.example:mytarget</target>
  <username>myuser</username>
  <password>mypassword</password>
</storage_connection_extension>
```

Table 591. Parameters summary

Name	Type	Direction	Summary
`extension`	StorageConnectionExtension	In/Out	

5.192.2. list GET

Table 592. Parameters summary

Name	Type	Direction	Summary
`extensions`	StorageConnectionExtension[]	Out	
`max`	Integer	In	Sets the maximum number of extensions to return.

max

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

5.193. StorageServerConnections

Table 593. Methods summary

Name	Summary
`add`	Creates a new storage connection.

Name	Summary
list	

5.193.1. add POST

Creates a new storage connection.

For example, to create a new storage connection for the NFS server `mynfs.example.com` and NFS share `/export/mydata` send a request like this:

```
POST /ovirt-engine/api/storageconnections
```

With a request body like this:

```
<storage_connection>
  <type>nfs</type>
  <address>mynfs.example.com</address>
  <path>/export/mydata</path>
  <host>
    <name>myhost</name>
  </host>
</storage_connection>
```

Table 594. Parameters summary

Name	Type	Direction	Summary
connection	StorageConnection	In/Out	

5.193.2. list GET

Table 595. Parameters summary

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

max

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

5.194. System

Table 596. Methods summary

Name	Summary
`get`	Returns basic information describing the API, like the product name, the version number and a summary of the number of relevant objects.
`reloadconfigurations`	

5.194.1. get GET

Returns basic information describing the API, like the product name, the version number and a summary of the number of relevant objects.

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

We get following response:

```
<api>
  <link rel="capabilities" href="/api/capabilities"/>
  <link rel="clusters" href="/api/clusters"/>
  <link rel="clusters/search" href="/api/clusters?search={query}"/>
  <link rel="datacenters" href="/api/datacenters"/>
  <link rel="datacenters/search" href="/api/datacenters?search={query}"/>
  <link rel="events" href="/api/events"/>
  <link rel="events/search" href="/api/events?search={query}"/>
  <link rel="hosts" href="/api/hosts"/>
  <link rel="hosts/search" href="/api/hosts?search={query}"/>
  <link rel="networks" href="/api/networks"/>
  <link rel="roles" href="/api/roles"/>
  <link rel="storagedomains" href="/api/storagedomains"/>
  <link rel="storagedomains/search" href="/api/storagedomains?search={query}"/>
  <link rel="tags" href="/api/tags"/>
  <link rel="templates" href="/api/templates"/>
  <link rel="templates/search" href="/api/templates?search={query}"/>
  <link rel="users" href="/api/users"/>
  <link rel="groups" href="/api/groups"/>
  <link rel="domains" href="/api/domains"/>
  <link rel="vmpools" href="/api/vmpools"/>
  <link rel="vmpools/search" href="/api/vmpools?search={query}"/>
  <link rel="vms" href="/api/vms"/>
  <link rel="vms/search" href="/api/vms?search={query}"/>
  <product_info>
    <name>oVirt Engine</name>
    <vendor>ovirt.org</vendor>
    <version>
      <build>4</build>
```

```
      <full_version>4.0.4</full_version>
      <major>4</major>
      <minor>0</minor>
      <revision>0</revision>
    </version>
  </product_info>
  <special_objects>
    <blank_template href="/ovirt-engine/api/templates/00000000-0000-0000-0000-
000000000000" id="00000000-0000-0000-0000-000000000000"/>
    <root_tag href="/ovirt-engine/api/tags/00000000-0000-0000-0000-000000000000"
id="00000000-0000-0000-0000-000000000000"/>
  </special_objects>
  <summary>
    <hosts>
      <active>0</active>
      <total>0</total>
    </hosts>
    <storage_domains>
      <active>0</active>
      <total>1</total>
    </storage_domains>
    <users>
      <active>1</active>
      <total>1</total>
    </users>
    <vms>
      <active>0</active>
      <total>0</total>
    </vms>
  </summary>
  <time>2016-09-14T12:00:48.132+02:00</time>
</api>
```


The entry point provides a user with links to the collections in a virtualization environment. The `rel` attribute of each collection link provides a reference point for each link.

The entry point also contains other data such as `product_info`, `special_objects` and `summary`.

Table 597. Parameters summary

Name	Type	Direction	Summary
`api`	Api	Out	

5.194.2. reloadconfigurations POST

Table 598. Parameters summary

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

5.195. SystemPermissions

This service doesn't add any new methods, it is just a placeholder for the annotation that specifies the path of the resource that manages the permissions assigned to the system object.

Table 599. Methods summary

Name	Summary
`add`	
`list`	

5.195.1. add POST

Table 600. Parameters summary

Name	Type	Direction	Summary
`permission`	Permission	In/Out	

5.195.2. list GET

Table 601. Parameters summary

Name	Type	Direction	Summary
`permissions`	Permission[]	Out	

5.196. Tag

Table 602. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.196.1. get GET

Table 603. Parameters summary

Name	Type	Direction	Summary
`tag`	Tag	Out	

5.196.2. remove DELETE

Table 604. Parameters summary

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

5.196.3. update PUT

Table 605. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`tag`	Tag	In/Out	

5.197. Tags

Table 606. Methods summary

Name	Summary
`add`	
`list`	

5.197.1. add POST

Table 607. Parameters summary

Name	Type	Direction	Summary
`tag`	Tag	In/Out	

5.197.2. list GET

Table 608. Parameters summary

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

max

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

5.198. Template

Table 609. Methods summary

Name	Summary
`export`	Exports a template to the data center export domain.
`get`	
`remove`	Removas a virtual machine template.
`update`	Updates the template.

5.198.1. export POST

Exports a template to the data center export domain.

For example, the operation can be facilitated using the following request:

```
POST /ovirt-engine/api/templates/123/export
```

With a request body like this:

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

Table 610. Parameters summary

Name	Type	Direction	Summary
`exclusive`	Boolean	In	Indicates if the existing templates with the same name should be overwritten.
`storage_domain`	StorageDomain	In	Specifies the destination export storage domain.

exclusive

Indicates if the existing templates with the same name should be overwritten.

The export action reports a failed action if a template of the same name exists in the destination domain. Set this parameter to `true` to change this behavior and overwrite any existing template.

5.198.2. get GET

Table 611. Parameters summary

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

5.198.3. remove DELETE

Removas a virtual machine template.

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

Table 612. Parameters summary

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

5.198.4. update PUT

Updates the template.

The `name`, `description`, `type`, `memory`, `cpu`, `topology`, `os`, `high_availability`, `display`, `stateless`, `usb` and `timezone` elements can be updated after a template has been created.

For example, to update a template to so that it has 1 GiB of memory send a request like this:

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

With the following request body:

```
<template>
  <memory>1073741824</memory>
</template>
```

The `version_name` name attribute is the only one that can be updated within the `version` attribute used for template versions:

```
<template>
  <version>
    <version_name>mytemplate_2</version_name>
  </version>
</template>
```

Table 613. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`template`	Template	In/Out	

5.199. TemplateCdrom

Table 614. Methods summary

Name	Summary
`get`	

5.199.1. get GET

Table 615. Parameters summary

Name	Type	Direction	Summary
`cdrom`	Cdrom	Out	

5.200. TemplateCdroms

Table 616. Methods summary

Name	Summary
`list`	

5.200.1. list GET

Table 617. Parameters summary

Name	Type	Direction	Summary
`cdroms`	Cdrom[]	Out	
`max`	Integer	In	Sets the maximum number of CDROMs to return.

max

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

5.201. TemplateDisk

Table 618. Methods summary

Name	Summary
`copy`	
`export`	
`get`	
`remove`	

5.201.1. copy POST

Table 619. Parameters summary

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

5.201.2. export POST

Table 620. Parameters summary

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

5.201.3. get GET

Table 621. Parameters summary

Name	Type	Direction	Summary
`disk`	Disk	Out	

5.201.4. remove DELETE

Table 622. Parameters summary

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

5.202. TemplateDiskAttachment

This service manages the attachment of a disk to a template.

Table 623. Methods summary

Name	Summary
`get`	Returns the details of the attachment.
`remove`	Removes the disk from the template.

5.202.1. get GET

Returns the details of the attachment.

Table 624. Parameters summary

Name	Type	Direction	Summary
`attachment`	DiskAttachment	Out	

5.202.2. remove DELETE

Removes the disk from the template. The disk will only be removed if there are other existing copies of the disk on other storage domains.

A storage domain has to be specified to determine which of the copies should be removed (template disks can have copies on multiple storage domains).

```
DELETE /ovirt-engine/api/templates/{template:id}/diskattachments/{attachment:id}?storage_domain=072fbaa1-08f3-4a40-9f34-a5ca22dd1d74
```

Table 625. Parameters summary

Name	Type	Direction	Summary
`force`	Boolean	In	
`storage_domain`	String	In	Specifies the identifier of the storage domain the image to be removed resides on.

5.203. TemplateDiskAttachments

This service manages the set of disks attached to a template. Each attached disk is represented by a DiskAttachment.

Table 626. Methods summary

Name	Summary
`list`	List the disks that are attached to the template.

5.203.1. list GET

List the disks that are attached to the template.

Table 627. Parameters summary

Name	Type	Direction	Summary
attachments	DiskAttachment[]	Out	

5.204. TemplateDisks

Table 628. Methods summary

Name	Summary
list	

5.204.1. list GET

Table 629. Parameters summary

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

max

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

5.205. TemplateNic

Table 630. Methods summary

Name	Summary
get	
remove	
update	

5.205.1. get GET

Table 631. Parameters summary

Name	Type	Direction	Summary
nic	Nic	Out	

5.205.2. remove DELETE

Table 632. Parameters summary

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

5.205.3. update PUT

Table 633. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`nic`	Nic	In/Out	

5.206. TemplateNics

Table 634. Methods summary

Name	Summary
`add`	
`list`	

5.206.1. add POST

Table 635. Parameters summary

Name	Type	Direction	Summary
`nic`	Nic	In/Out	

5.206.2. list GET

Table 636. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of NICs to return.
`nics`	Nic[]	Out	

max

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

5.207. TemplateWatchdog

Table 637. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.207.1. get GET

Table 638. Parameters summary

Name	Type	Direction	Summary
`watchdog`	Watchdog	Out	

5.207.2. remove DELETE

Table 639. Parameters summary

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

5.207.3. update PUT

Table 640. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`watchdog`	Watchdog	In/Out	

5.208. TemplateWatchdogs

Table 641. Methods summary

Name	Summary
`add`	
`list`	

5.208.1. add POST

Table 642. Parameters summary

Name	Type	Direction	Summary
`watchdog`	Watchdog	In/Out	

5.208.2. list GET

Table 643. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of watchdogs to return.
`watchdogs`	Watchdog[]	Out	

max

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

5.209. Templates

Table 644. Methods summary

Name	Summary
`add`	Creates a new template.
`list`	

5.209.1. add POST

Creates a new template.

This requires the `name` and `vm` elements. Identify the virtual machine with the `id` `name` attributes.

```
POST /ovirt-engine/api/templates
```

With a request body like this:

```
<template>
  <name>mytemplate</name>
  <vm id="123"/>
</template>
```

The template can be created as a sub version of an existing template.This requires the `name` and `vm` attributes for the new template, and the `base_template` and `version_name` attributes for the new template version. The `base_template` and `version_name` attributes must be specified within a `version` section enclosed in the `template` section. Identify the virtual machine with the `id` or `name` attributes.

```
<template>
  <name>mytemplate</name>
  <vm id="123"/>
  <version>
    <base_template id="456"/>
    <version_name>mytemplate_001</version_name>
  </version>
</template>
```

Table 645. Parameters summary

Name	Type	Direction	Summary
`clone_permissions`	Boolean	In	Specifies if the permissions of the virtual machine should be copied to the template.
`template`	Template	In/Out	

clone_permissions

Specifies if the permissions of the virtual machine should be copied to the template.

If this optional parameter is provided, and its values is `true` then the permissions of the virtual machine (only the direct ones, not the inherited ones) will be copied to the created template. For example, to create a template from the `myvm` virtual machine copying its permissions, send a request like this:

```
POST /ovirt-engine/api/templates?clone_permissions=true
```

With a request body like this:

```
<template>
  <name>mytemplate<name>
  <vm>
    <name>myvm<name>
  </vm>
</template>
```

5.209.2. list GET

Table 646. Parameters summary

Name	Type	Direction	Summary
`case_sensitive`	Boolean	In	Indicates if the search performed using the `search` parameter should be performed taking case into account.

Name	Type	Direction	Summary
`filter`	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
`max`	Integer	In	Sets the maximum number of templates to return.
`search`	String	In	A query string used to restrict the returned templates.
`templates`	Template[]	Out	

case_sensitive

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

max

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

5.210. UnmanagedNetwork

Table 647. Methods summary

Name	Summary
`get`	
`remove`	

5.210.1. get GET

Table 648. Parameters summary

Name	Type	Direction	Summary
`network`	UnmanagedNetwork	Out	

5.210.2. remove DELETE

Table 649. Parameters summary

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

5.211. UnmanagedNetworks

Table 650. Methods summary

Name	Summary
`list`	

5.211.1. list GET

Table 651. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of networks to return.
`networks`	UnmanagedNetwork[]	Out	

max

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

5.212. User

Table 652. Methods summary

Name	Summary
`get`	
`remove`	

5.212.1. get GET

Table 653. Parameters summary

Name	Type	Direction	Summary
`user`	User	Out	

5.212.2. remove DELETE

Table 654. Parameters summary

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

5.213. Users

Table 655. Methods summary

Name	Summary
add	Add user from a directory service.
list	

5.213.1. add POST

Add user from a directory service.

For example, to add the `myuser` user from the `myextension-authz` authorization provider send a request like this:

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

With a request body like this:

```
<user>
  <user_name>myuser@myextension-authz</user_name>
  <domain>
    <name>myextension-authz</name>
  </domain>
</user>
```

In case you are working with Active Directory you have to pass user principal name (UPN) as `username`, followed by authorization provider name. Due to bug 1147900 you need to provide also `principal` parameter set to UPN of the user.

For example, to add the user with UPN `myuser@mysubdomain.mydomain.com` from the `myextension-authz` authorization provider send a request body like this:

```
<user>
  <principal>myuser@mysubdomain.mydomain.com</principal>
  <user_name>myuser@mysubdomain.mydomain.com@myextension-authz</user_name>
  <domain>
    <name>myextension-authz</name>
  </domain>
</user>
```

Table 656. Parameters summary

Name	Type	Direction	Summary
user	User	In/Out	

5.213.2. list GET

Table 657. Parameters summary

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

case_sensitive

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

max

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

5.214. VirtualFunctionAllowedNetwork

Table 658. Methods summary

Name	Summary
`get`	
`remove`	

5.214.1. get GET

Table 659. Parameters summary

Name	Type	Direction	Summary
`network`	Network	Out	

5.214.2. remove DELETE

Table 660. Parameters summary

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

5.215. VirtualFunctionAllowedNetworks

Table 661. Methods summary

Name	Summary
`add`	
`list`	

5.215.1. add POST

Table 662. Parameters summary

Name	Type	Direction	Summary
`network`	Network	In/Out	

5.215.2. list GET

Table 663. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of networks to return.
`networks`	Network[]	Out	

max

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

5.216. Vm

Table 664. Methods summary

Name	Summary
`cancelmigration`	This operation stops any migration of a virtual machine to another physical host.
`clone`	
`commitsnapshot`	
`detach`	Detaches a virtual machine from a pool.
`export`	Export a virtual machine to an export domain.
`freezefilesystems`	Freeze virtual machine file systems.
`get`	Retrieves the description of the virtual machine.
`logon`	Initiates the automatic user logon to access a virtual machine from an external console.

Name	Summary
maintenance	
migrate	This operation migrates a virtual machine to another physical host.
previewsnapshot	
reboot	This operation sends a reboot request to a virtual machine.
remove	Removes the virtual machine, including the virtual disks attached to it.
reordermacaddresses	
shutdown	This operation sends a shutdown request to a virtual machine.
start	Starts the virtual machine.
stop	This operation forces a virtual machine to power-off.
suspend	This operation saves the virtual machine state to disk and stops it.
thawfilesystems	Thaw virtual machine file systems.
ticket	Generates a time-sensitive authentication token for accessing a virtual machine's display.
undosnapshot	
update	

5.216.1. cancelmigration POST

This operation stops any migration of a virtual machine to another physical host.

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

The cancel migration action does not take any action specific parameters, so the request body should contain an empty `action`:

```
<action/>
```

Table 665. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the migration should cancelled asynchronously.

5.216.2. clone POST

Table 666. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the clone should be performed asynchronously.
`vm`	Vm	In	

5.216.3. commitsnapshot POST

Table 667. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the snapshots should be committed asynchronously.

5.216.4. detach POST

Detaches a virtual machine from a pool.

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

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

```
<action/>
```

Table 668. Parameters summary

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

5.216.5. export POST

Export a virtual machine to an export domain.

For example to export virtual machine `123` to the export domain `myexport`, send a request like this:

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

With a request body like this:

```
<action>
  <storage_domain>
    <name>myexport</name>
  </storage_domain>
  <exclusive>true</exclusive>
  <discard_snapshots>true</discard_snapshots>
</action>
```

Table 669. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the export should be performed asynchronously.
`discard_snapshots`	Boolean	In	The `discard_snapshots` parameter is to be used when the virtual machine should be exported with all its snapshots collapsed.
`exclusive`	Boolean	In	The `exclusive` parameter is to be used when the virtual machine should be exported even if another copy of it already exists in the export domain (override).
`storage_domain`	StorageDomain	In	

5.216.6. freezefilesystems POST

Freeze virtual machine file systems.

This operation freezes a virtual machine's file systems using the QEMU guest agent when taking a live snapshot of a running virtual machine. Normally, this is done automatically by the manager, but this must be executed manually with the API for virtual machines using OpenStack Volume (Cinder) disks.

Example:

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

```
<action/>
```

Table 670. Parameters summary

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

5.216.7. get GET

Retrieves the description of the virtual machine.

Table 671. Parameters summary

Name	Type	Direction	Summary
`all_content`	Boolean	In	Indicates if all the attributes of the virtual machine should be included in the response.
`filter`	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
`next_run`	Boolean	In	Indicates if the returned result describes the virtual machine as it is currently running, or if describes it with the modifications that have already been performed but that will have effect only when it is restarted.
`vm`	Vm	Out	Description of the virtual machine.

all_content

Indicates if all the attributes of the virtual machine should be included in the response.

By default the following attributes are excluded:

- `console`
- `initialization.configuration.data` - The OVF document describing the virtual machine.
- `rng_source`
- `soundcard`
- `virtio_scsi`

For example, to retrieve the complete representation of the virtual machine '123' send a request like this:

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

NOTE	The reason for not including these attributes is performance: they are seldom used and they require additional queries to the database. So try to use the this parameter only when it is really needed.

next_run

Indicates if the returned result describes the virtual machine as it is currently running, or if describes it with the modifications that have already been performed but that will have effect only when it is restarted. By default the values is `false`.

If the parameter is included in the request, but without a value, it is assumed that the value is `true`, so the following request:

```
GET /vms/{vm:id};next_run
```

Is equivalent to using the value `true`:

```
GET /vms/{vm:id};next_run=true
```

5.216.8. logon POST

Initiates the automatic user logon to access a virtual machine from an external console.

This action requires the `ovirt-guest-agent-gdm-plugin` and the `ovirt-guest-agent-pam-module` packages to be installed and the `ovirt-guest-agent` service to be running on the virtual machine.

Users require the appropriate user permissions for the virtual machine in order to access the virtual machine from an external console.

This is how an example request would look like:

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

Request body:

```
<action/>
```

Table 672. Parameters summary

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

5.216.9. maintenance POST

Table 673. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the action should be performed asynchronously.
`maintenance_enabled`	Boolean	In	

5.216.10. migrate POST

This operation migrates a virtual machine to another physical host.

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

One can specify a specific host to migrate the virtual machine to:

```
<action>
  <host id="2ab5e1da-b726-4274-bbf7-0a42b16a0fc3"/>
</action>
```

Table 674. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the migration should be performed asynchronously.
`cluster`	Cluster	In	Specifies the cluster the virtual machine should migrate to.
`force`	Boolean	In	Specifies the virtual machine should migrate although it might be defined as non migratable.
`host`	Host	In	Specifies a specific host the virtual machine should migrate to.

cluster

Specifies the cluster the virtual machine should migrate to. This is an optional parameter. By default, the virtual machine is migrated to another host within the same cluster.

force

Specifies the virtual machine should migrate although it might be defined as non migratable. This is an optional parameter. By default, it is set to `false`.

host

Specifies a specific host the virtual machine should migrate to. This is an optional parameters. By default, the oVirt Engine automatically selects a default host for migration within the same cluster. If an API user requires a specific host, the user can specify the host with either an `id` or `name` parameter.

5.216.11. previewsnapshot POST

Table 675. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the preview should be performed asynchronously.
`disks`	Disk[]	In	

Name	Type	Direction	Summary
restore_memory	Boolean	In	
snapshot	Snapshot	In	
vm	Vm	In	

5.216.12. reboot POST

This operation sends a reboot request to a virtual machine.

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

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

```
<action/>
```

Table 676. Parameters summary

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

5.216.13. remove DELETE

Removes the virtual machine, including the virtual disks attached to it.

For example, to remove the virtual machine with identifier `123` send a request like this:

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

Table 677. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the remove should be performed asynchronously.
detach_only	Boolean	In	Indicates if the attached virtual disks should be detached first and preserved instead of being removed.
force	Boolean	In	Indicates if the virtual machine should be forcibly removed.

force

Indicates if the virtual machine should be forcibly removed.

Locked virtual machines and virtual machines with locked disk images cannot be removed without this flag set to true.

5.216.14. reordermacaddresses POST

Table 678. Parameters summary

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

5.216.15. shutdown POST

This operation sends a shutdown request to a virtual machine.

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

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

```
<action/>
```

Table 679. Parameters summary

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

5.216.16. start POST

Starts the virtual machine.

If the virtual environment is complete and the virtual machine contains all necessary components to function, it can be started.

This example starts the virtual machine:

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

With a request body:

```
<action/>
```

Table 680. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the action should be performed asynchronously.
`filter`	Boolean	In	Indicates if the results should be filtered according to the permissions of the user.
`pause`	Boolean	In	If set to `true`, start the virtual machine in paused mode.
`use_cloud_init`	Boolean	In	If set to `true`, the initialization type is set to *cloud-init*.
`use_sysprep`	Boolean	In	If set to `true`, the initialization type is set to *Sysprep*.
`vm`	Vm	In	The definition of the virtual machine for this specific run.

pause

If set to `true`, start the virtual machine in paused mode. Default is `false`.

use_cloud_init

If set to `true`, the initialization type is set to *cloud-init*. The default value is `false`. See this for details.

use_sysprep

If set to `true`, the initialization type is set to *Sysprep*. The default value is `false`. See this for details.

vm

The definition of the virtual machine for this specific run.

For example:

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

This will set the boot device to the CDROM only for this specific start. After the virtual machine will be powered off, this definition will be reverted.

5.216.17. stop POST

This operation forces a virtual machine to power-off.

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

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

```
<action/>
```

Table 681. Parameters summary

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

5.216.18. suspend POST

This operation saves the virtual machine state to disk and stops it. Start a suspended virtual machine and restore the virtual machine state with the start action.

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

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

```
<action/>
```

Table 682. Parameters summary

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

5.216.19. thawfilesystems POST

Thaw virtual machine file systems.

This operation thaws a virtual machine's file systems using the QEMU guest agent when taking a live snapshot of a running virtual machine. Normally, this is done automatically by the manager, but this must be executed manually with the API for virtual machines using OpenStack Volume (Cinder) disks.

Example:

```
POST /api/vms/123/thawfilesystems
```

```
<action/>
```

Table 683. Parameters summary

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

5.216.20. ticket POST

Generates a time-sensitive authentication token for accessing a virtual machine's display.

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

The client-provided action optionally includes a desired ticket value and/or an expiry time in seconds.

In any case, the response specifies the actual ticket value and expiry used.

```
<action>
  <ticket>
    <value>abcd12345</value>
    <expiry>120</expiry>
  </ticket>
</action>
```

Table 684. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the generation of the ticket should be performed asynchronously.
`ticket`	Ticket	In/Out	

5.216.21. undosnapshot POST

Table 685. Parameters summary

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

5.216.22. update PUT

Table 686. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`next_run`	Boolean	In	Indicates if the update should be applied to the virtual machine immediately, or if it should be applied only when the virtual machine is restarted.
`vm`	Vm	In/Out	

next_run

Indicates if the update should be applied to the virtual machine immediately, or if it should be applied only when the virtual machine is restarted. The default value is `false`, so by default changes are applied immediately.

5.217. VmApplication

Table 687. Methods summary

Name	Summary
`get`	

5.217.1. get GET

Table 688. Parameters summary

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

5.218. VmApplications

Table 689. Methods summary

Name	Summary
list	

5.218.1. list GET

Table 690. Parameters summary

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

max

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

5.219. VmCdrom

Manages a CDROM device of a virtual machine.

If there is a disk inserted then the `file` attribute will contain a reference to the ISO image:

```
<cdrom>
  <file id="mycd.iso"/>
</cdrom>
```

If there is no disk inserted then the `file` attribute won't be reported:

```
<cdrom>
</cdrom>
```

Changing and ejecting the disk is done using always the `update` method, to change the value of the

`file` attribute. For example, to insert or change the disk send a request like this:

```
PUT /ovirt-engine/api/vms/123/cdroms/00000000-0000-0000-0000-000000000000
```

The body should contain the new value for the `file` attribute:

```
<cdrom>
  <file id="mycd.iso"/>
</cdrom>
```

The value of the `id` attribute, `mycd.iso` in this example, should correspond to a file available in an attached ISO storage domain.

To eject the disk use a `file` with an empty `id`:

```
<cdrom>
  <file id=""/>
</cdrom>
```

By default the above operations change permanently the disk that will be visible to the virtual machine after the next boot, but they don't have any effect on the currently running virtual machine. If you want to change the disk that is visible to the current running virtual machine, add the `current=true` parameter. For example, to eject the current disk send a request like this:

```
PUT /ovirt-engine/api/vms/123/cdroms/00000000-0000-0000-0000-000000000000?current=true
```

With a request body like this:

```
<cdrom>
  <file id=""/>
</cdrom>
```

IMPORTANT	The changes made with the `current=true` parameter are never persisted, so they won't have any effect after the virtual machine is rebooted.

Table 691. Methods summary

Name	Summary
`get`	Returns the information about this CDROM device.
`update`	Updates the information about this CDROM device.

5.219.1. get GET

Returns the information about this CDROM device.

Table 692. Parameters summary

Name	Type	Direction	Summary
`cdrom`	Cdrom	Out	The information about the CDROM device.
`current`	Boolean	In	Indicates if the operation should return the information for the currently running virtual machine.

current

Indicates if the operation should return the information for the currently running virtual machine. This parameter is optional, and the default value is `false`.

5.219.2. update PUT

Updates the information about this CDROM device.

Table 693. Parameters summary

Name	Type	Direction	Summary
`cdrom`	Cdrom	In/Out	The information about the CDROM device.
`current`	Boolean	In	Indicates if the update should apply to the currently running virtual machine, or to the virtual machine after the next boot.

current

Indicates if the update should apply to the currently running virtual machine, or to the virtual machine after the next boot. This parameter is optional, and the default value is `false`, which means that by default the update will have effect only after the next boot.

5.220. VmCdroms

Manages the CDROM devices of a virtual machine.

Currently virtual machines have exactly one CDROM device. No new devices can be added, and the existing one can't be removed, thus there are no `add` or `remove` methods. Changing and ejecting CDROM disks is done with the update method of the service that manages the CDROM device.

Table 694. Methods summary

Name	Summary
`list`	Returns the list of CDROM devices of the virtual machine.

5.220.1. list GET

Returns the list of CDROM devices of the virtual machine.

Table 695. Parameters summary

Name	Type	Direction	Summary
`cdroms`	Cdrom[]	Out	The list of CDROM devices of the virtual machine.
`max`	Integer	In	Sets the maximum number of CDROMs to return.

max

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

5.221. VmDisk

Table 696. Methods summary

Name	Summary
`activate`	
`deactivate`	
`export`	
`get`	
`move`	
`remove`	Detach the disk from the virtual machine.
`update`	

5.221.1. activate POST

Table 697. Parameters summary

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

5.221.2. deactivate POST

Table 698. Parameters summary

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

5.221.3. export POST

Table 699. Parameters summary

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

5.221.4. get GET

Table 700. Parameters summary

Name	Type	Direction	Summary
`disk`	Disk	Out	

5.221.5. move POST

Table 701. Parameters summary

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

5.221.6. remove DELETE

Detach the disk from the virtual machine.

NOTE In version 3 of the API this used to also remove the disk completely from the system, but starting with version 4 it doesn't. If you need to remove it completely use the remove method of the top level disk service.

Table 702. Parameters summary

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

5.221.7. update PUT

Table 703. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`disk`	Disk	In/Out	

5.222. VmDisks

Table 704. Methods summary

Name	Summary
`add`	
`list`	

5.222.1. add POST

Table 705. Parameters summary

Name	Type	Direction	Summary
`disk`	Disk	In/Out	

5.222.2. list GET

Table 706. Parameters summary

Name	Type	Direction	Summary
`disks`	Disk[]	Out	
`max`	Integer	In	Sets the maximum number of disks to return.

max

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

5.223. VmGraphicsConsole

Table 707. Methods summary

Name	Summary
`get`	Gets the configuration of the graphics console.
`proxyticket`	
`remove`	

5.223.1. get GET

Gets the configuration of the graphics console.

Table 708. Parameters summary

Name	Type	Direction	Summary
console	GraphicsConsole	Out	
current	Boolean	In	Use the following query to obtain the current run-time configuration of the graphics console.

current

Use the following query to obtain the current run-time configuration of the graphics console.

```
GET /ovit-engine/api/vms/{vm:id}/graphicsconsoles/{console:id}?current=true
```

The default value is `false`.

5.223.2. proxyticket POST

Table 709. Parameters summary

Name	Type	Direction	Summary
async	Boolean	In	Indicates if the generation of the ticket should be performed asynchronously.
proxy_ticket	ProxyTicket	Out	

5.223.3. remove DELETE

Table 710. Parameters summary

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

5.224. VmHostDevice

Table 711. Methods summary

Name	Summary
get	
remove	

5.224.1. get GET

Table 712. Parameters summary

Name	Type	Direction	Summary
device	HostDevice	Out	

5.224.2. remove DELETE

Table 713. Parameters summary

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

5.225. VmHostDevices

Table 714. Methods summary

Name	Summary
add	
list	

5.225.1. add POST

Table 715. Parameters summary

Name	Type	Direction	Summary
device	HostDevice	In/Out	

5.225.2. list GET

Table 716. Parameters summary

Name	Type	Direction	Summary
device	HostDevice[]	Out	
max	Integer	In	Sets the maximum number of devices to return.

max

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

5.226. VmNic

Table 717. Methods summary

Name	Summary
activate	
deactivate	

Name	Summary
`get`	
`remove`	Removes the NIC.
`update`	Updates the NIC.

5.226.1. activate POST

Table 718. Parameters summary

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

5.226.2. deactivate POST

Table 719. Parameters summary

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

5.226.3. get GET

Table 720. Parameters summary

Name	Type	Direction	Summary
`nic`	Nic	Out	

5.226.4. remove DELETE

Removes the NIC.

For example, to remove the NIC with id `456` from the virtual machine with id `123` send a request like this:

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

IMPORTANT

The hotplugging feature only supports virtual machine operating systems with hotplugging operations. Example operating systems include:

- Red Hat Enterprise Linux 6
- Red Hat Enterprise Linux 5
- Windows Server 2008 and
- Windows Server 2003

Table 721. Parameters summary

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

5.226.5. update PUT

Updates the NIC.

For example, to update the NIC having with `456` belonging to virtual the machine with id `123` send a request like this:

```
PUT /ovirt-engine/api/vms/123/nics/456
```

With a request body like this:

```
<nic>
  <name>mynic</name>
  <interface>e1000</interface>
</nic>
```

IMPORTANT

The hotplugging feature only supports virtual machine operating systems with hotplugging operations. Example operating systems include:

- Red Hat Enterprise Linux 6
- Red Hat Enterprise Linux 5
- Windows Server 2008 and
- Windows Server 2003

Table 722. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`nic`	Nic	In/Out	

5.227. VmNics

Table 723. Methods summary

Name	Summary
`add`	Adds a NIC to the virtual machine.

Name	Summary
list	

5.227.1. add POST

Adds a NIC to the virtual machine.

The following example adds a network interface named `mynic` using `virtio` and the `ovirtmgmt` network to the virtual machine.

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

```
<nic>
  <interface>virtio</interface>
  <name>mynic</name>
  <network>
    <name>ovirtmgmt</name>
  </network>
</nic>
```

The following example sends that request using `curl`:

```
curl \
--request POST \
--header "Version: 4" \
--header "Content-Type: application/xml" \
--header "Accept: application/xml" \
--user "admin@internal:mypassword" \
--cacert /etc/pki/ovirt-engine/ca.pem \
--data '
<nic>
  <name>mynic</name>
  <network>
    <name>ovirtmgmt</name>
  </network>
</nic>
' \
https://myengine.example.com/ovirt-engine/api/vms/123/nics
```

IMPORTANT	The hotplugging feature only supports virtual machine operating systems with hotplugging operations. Example operating systems include: • Red Hat Enterprise Linux 6 • Red Hat Enterprise Linux 5 • Windows Server 2008 and • Windows Server 2003

Table 724. Parameters summary

Name	Type	Direction	Summary
`nic`	Nic	In/Out	

5.227.2. list GET

Table 725. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of NICs to return.
`nics`	Nic[]	Out	

max

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

5.228. VmNumaNode

Table 726. Methods summary

Name	Summary
`get`	
`remove`	Removes a virtual NUMA node.
`update`	Updates a virtual NUMA node.

5.228.1. get GET

Table 727. Parameters summary

Name	Type	Direction	Summary
`node`	VirtualNumaNode	Out	

5.228.2. remove DELETE

Removes a virtual NUMA node.

An example of removing a virtual NUMA node:

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

Table 728. Parameters summary

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

5.228.3. update PUT

Updates a virtual NUMA node.

An example of pinning a virtual NUMA node to a physical NUMA node on the host:

```
PUT /ovirt-engine/api/vms/123/numanodes/456
```

The request body should contain the following:

```
<vm_numa_node>
  <numa_node_pins>
    <numa_node_pin>
      <host_numa_node id="789"/>
      <index>0</index>
      <pinned>true</pinned>
    </numa_node_pin>
  </numa_node_pins>
</vm_numa_node>
```

Table 729. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`node`	VirtualNumaNode	In/Out	

5.229. VmNumaNodes

Table 730. Methods summary

Name	Summary
`add`	Creates a new virtual NUMA node for the virtual machine.

Name	Summary
list	Lists virtual NUMA nodes of a virtual machine.

5.229.1. add POST

Creates a new virtual NUMA node for the virtual machine.

An example of creating a NUMA node:

```
POST /ovirt-engine/api/vms/c7ecd2dc/numanodes
Accept: application/xml
Content-type: application/xml
```

The request body can contain the following:

```
<vm_numa_node>
  <cpu>
    <cores>
      <core>
        <index>0</index>
      </core>
    </cores>
  </cpu>
  <index>0</index>
  <memory>1024</memory>
</vm_numa_node>
```

Table 731. Parameters summary

Name	Type	Direction	Summary
node	VirtualNumaNode	In/Out	

5.229.2. list GET

Lists virtual NUMA nodes of a virtual machine.

Table 732. Parameters summary

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

max

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

5.230. VmPool

Table 733. Methods summary

Name	Summary
`allocatevm`	This operation allocates a virtual machine in the virtual machine pool.
`get`	
`remove`	Removes a virtual machine pool.
`update`	Update the virtual machine pool.

5.230.1. allocatevm POST

This operation allocates a virtual machine in the virtual machine pool.

```
POST /ovirt-engine/api/vmpools/123/allocatevm
```

The allocate virtual machine action does not take any action specific parameters, so the request body should contain an empty `action`:

```
<action/>
```

Table 734. Parameters summary

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

5.230.2. get GET

Table 735. Parameters summary

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

5.230.3. remove DELETE

Removes a virtual machine pool.

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

Table 736. Parameters summary

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

5.230.4. update PUT

Update the virtual machine pool.

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

The `name`, `description`, `size`, `prestarted_vms` and `max_user_vms` attributes can be updated after the virtual machine pool has been created.

```
<vmpool>
  <name>VM_Pool_B</name>
  <description>Virtual Machine Pool B</description>
  <size>3</size>
  <prestarted_vms>1</size>
  <max_user_vms>2</size>
</vmpool>
```

Table 737. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`pool`	VmPool	In/Out	The virtual machine pool that is being updated.

5.231. VmPools

Table 738. Methods summary

Name	Summary
`add`	Creates a new virtual machine pool.
`list`	

5.231.1. add POST

Creates a new virtual machine pool.

A new pool requires the `name`, `cluster` and `template` attributes. Identify the cluster and template with the `id` or `name` nested attributes:

```
POST /ovirt-engine/api/vmpools
```

With the following body:

```
<vmpool>
  <name>mypool</name>
  <cluster id="123"/>
  <template id="456"/>
</vmpool>
```

Table 739. Parameters summary

Name	Type	Direction	Summary
`pool`	VmPool	In/Out	

5.231.2. list GET

Table 740. Parameters summary

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

case_sensitive

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

max

Sets the maximum number of pools to return. If this value is not specified, all of the pools are returned.

5.232. VmReportedDevice

Table 741. Methods summary

Name	Summary
`get`	

5.232.1. get GET

Table 742. Parameters summary

Name	Type	Direction	Summary
`reported_device`	ReportedDevice	Out	

5.233. VmReportedDevices

Table 743. Methods summary

Name	Summary
`list`	

5.233.1. list GET

Table 744. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of devices to return.
`reported_device`	ReportedDevice[]	Out	

max

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

5.234. VmSession

Table 745. Methods summary

Name	Summary
`get`	

5.234.1. get GET

Table 746. Parameters summary

Name	Type	Direction	Summary
session	Session	Out	

5.235. VmSessions

Provides information about virtual machine user sessions.

Table 747. Methods summary

Name	Summary
list	Lists all user sessions for this virtual machine.

5.235.1. list GET

Lists all user sessions for this virtual machine.

For example, to retrieve the session information for virtual machine 123 send a request like this:

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

The response body will contain something like this:

```
<sessions>
  <session href="/ovirt-engine/api/vms/123/sessions/456" id="456">
    <console_user>true</console_user>
    <ip>
      <address>192.168.122.1</address>
    </ip>
    <user href="/ovirt-engine/api/users/789" id="789"/>
    <vm href="/ovirt-engine/api/vms/123" id="123"/>
  </session>
  ...
</sessions>
```

Table 748. Parameters summary

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

max

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

5.236. VmWatchdog

Table 749. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.236.1. get GET

Table 750. Parameters summary

Name	Type	Direction	Summary
`watchdog`	Watchdog	Out	

5.236.2. remove DELETE

Table 751. Parameters summary

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

5.236.3. update PUT

Table 752. Parameters summary

Name	Type	Direction	Summary
`async`	Boolean	In	Indicates if the update should be performed asynchronously.
`watchdog`	Watchdog	In/Out	

5.237. VmWatchdogs

Table 753. Methods summary

Name	Summary
`add`	
`list`	

5.237.1. add POST

Table 754. Parameters summary

Name	Type	Direction	Summary
`watchdog`	Watchdog	In/Out	

5.237.2. list GET

Table 755. Parameters summary

Name	Type	Direction	Summary
`max`	Integer	In	Sets the maximum number of watchdogs to return.
`watchdogs`	Watchdog[]	Out	

max

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

5.238. Vms

Table 756. Methods summary

Name	Summary
`add`	Creates a new virtual machine.
`list`	

5.238.1. add POST

Creates a new virtual machine.

The virtual machine can be created in different ways:

- From a template. In this case the identifier or name of the template must be provided. For example, using a plain shell script and XML:

```
#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."
curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--user "${user}:${password}" \
--request POST \
--header "Version: 4" \
--header "Content-Type: application/xml" \
--header "Accept: application/xml" \
--data '
<vm>
  <name>myvm</name>
  <template>
    <name>Blank</name>
  </template>
  <cluster>
    <name>mycluster</name>
  </cluster>
</vm>
' \
"${url}/vms"
```

- From a snapshot. In this case the identifier of the snapshot has to be provided. For example, using a plain shel script and XML:

```
#!/bin/sh -ex

url="https://engine.example.com/ovirt-engine/api"
user="admin@internal"
password="..."
curl \
--verbose \
--cacert /etc/pki/ovirt-engine/ca.pem \
--user "${user}:${password}" \
--request POST \
--header "Content-Type: application/xml" \
--header "Accept: application/xml" \
--data '
<vm>
  <name>myvm</name>
  <snapshots>
    <snapshot id="266742a5-6a65-483c-816d-d2ce49746680"/>
  </snapshots>
  <cluster>
    <name>mycluster</name>
  </cluster>
</vm>
' \
"${url}/vms"
```

When creating a virtual machine from a template or from a snapshot it is usually useful to explicitly indicate in what storage domain to create the disks for the virtual machine. If the virtual machine is created from a template then this is achieved passing a set of `disk_attachment` elements that indicate the mapping:

```
<vm>
  ...
  <disk_attachments>
    <disk_attachment>
      <disk id="8d4bd566-6c86-4592-a4a7-912dbf93c298">
        <storage_domains>
          <storage_domain id="9cb6cb0a-cf1d-41c2-92ca-5a6d665649c9"/>
        </storage_domains>
      </disk>
    <disk_attachment>
  </disk_attachments>
</vm>
```

When the virtual machine is created from a snapshot this set of disks is slightly different, it uses the `image_id` attribute instead of `id`.

```
<vm>
  ...
  <disk_attachments>
    <disk_attachment>
      <disk>
        <image_id>8d4bd566-6c86-4592-a4a7-912dbf93c298</image_id>
        <storage_domains>
          <storage_domain id="9cb6cb0a-cf1d-41c2-92ca-5a6d665649c9"/>
        </storage_domains>
      </disk>
    <disk_attachment>
  </disk_attachments>
</vm>
```

It is possible to specify additional virtual machine parameters in the XML description, e.g. a virtual machine of `desktop` type, with 2 GiB of RAM and additional description can be added sending a request body like the following:

```
<vm>
  <name>myvm</name>
  <description>My Desktop Virtual Machine</description>
  <type>desktop</type>
  <memory>2147483648</memory>
  ...
</vm>
```

A bootable CDROM device can be set like this:

```
<vm>
  ...
  <os>
    <boot dev="cdrom"/>
  </os>
</vm>
```

In order to boot from CDROM, you first need to insert a disk, as described in the CDROM service. Then booting from that CDROM can be specified using the `os.boot.devices` attribute:

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

In all cases the name or identifier of the cluster where the virtual machine will be created is mandatory.

Table 757. Parameters summary

Name	Type	Direction	Summary
`clone`	Boolean	In	Specifies if the virtual machine should be independent of the template.
`clone_permissions`	Boolean	In	Specifies if the permissions of the template should be copied to the virtual machine.
`vm`	Vm	In/Out	

clone

Specifies if the virtual machine should be independent of the template.

When a virtual machine is created from a template by default the disks of the virtual machine depend on the disks of the template, they are using the *copy on write* mechanism so that only the differences from the template take up real storage space. If this parameter is specified and the value is `true` then the disks of the created virtual machine will be *cloned*, and independent of the template. For example, to create an independent virtual machine, send a request like this:

```
POST /ovirt-engine/vms?clone=true
```

With a request body like this:

```
<vm>
  <name>myvm<name>
  <template>
    <name>mytemplate<name>
  </template>
  <cluster>
    <name>mycluster<name>
  </cluster>
</vm>
```

NOTE When this parameter is `true` the permissions of the template will also be copied, as when using `clone_permissions=true`.

clone_permissions

Specifies if the permissions of the template should be copied to the virtual machine.

If this optional parameter is provided, and its values is `true` then the permissions of the template (only the direct ones, not the inherited ones) will be copied to the created virtual machine. For example, to create a virtual machine from the `mytemplate` template copying its permissions, send a request like this:

```
POST /ovirt-engine/api/vms?clone_permissions=true
```

With a request body like this:

```
<vm>
  <name>myvm<name>
  <template>
    <name>mytemplate<name>
  </template>
  <cluster>
    <name>mycluster<name>
  </cluster>
</vm>
```

5.238.2. list GET

Table 758. Parameters summary

Name	Type	Direction	Summary
`all_content`	Boolean	In	Indicates if all the attributes of the virtual machines should be included in the response.

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

all_content

Indicates if all the attributes of the virtual machines should be included in the response.

By default the following attributes are excluded:

- `console`
- `initialization.configuration.data` - The OVF document describing the virtual machine.
- `rng_source`
- `soundcard`
- `virtio_scsi`

For example, to retrieve the complete representation of the virtual machines send a request like this:

```
GET /ovirt-engine/api/vms?all_content=true
```

NOTE	The reason for not including these attributes is performance: they are seldom used and they require additional queries to the database. So try to use the this parameter only when it is really needed.

case_sensitive

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

5.239. VnicProfile

Since 4.0 it is possible to have a customized network filter to each VNIC profile. Please note that there is a default network filter to each VNIC profile. For more details of how the default network filter is calculated please refer to the documentation in NetworkFilters.

The basic POST command of adding a new VNIC profile is as follows:

```
http://{engine_ip_address}:8080/ovirt-engine/api/networks/{network_id}/vnicprofiles
```

The output of creating a new VNIC profile depends in the body arguments that were given. In case no network filter was given, the default network filter will be configured. For example:

```
<vnic_profile>
  <name>use_default_network_filter</name>
  <network id="00000000-0000-0000-0000-000000000009"/>
</vnic_profile>
```

In case an empty network filter was given, no network filter will be configured for the specific VNIC profile regardless of the VNIC profile's default network filter. For example:

```
<vnic_profile>
  <name>no_network_filter</name>
  <network id="00000000-0000-0000-0000-000000000009"/>
  <network_filter/>
</vnic_profile>
```

In case that a specific valid network filter id was given, the VNIC profile will be configured with the given network filter regardless of the VNIC profiles's default network filter. For example:

```
<vnic_profile>
  <name>user_choice_network_filter</name>
  <network id="00000000-0000-0000-0000-000000000009"/>
  <network_filter id= "0000001b-001b-001b-001b-0000000001d5"/>
</vnic_profile>
```

Table 759. Methods summary

Name	Summary
`get`	
`remove`	
`update`	

5.239.1. get GET

Table 760. Parameters summary

Name	Type	Direction	Summary
`profile`	VnicProfile	Out	

5.239.2. remove DELETE

Table 761. Parameters summary

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

5.239.3. update PUT

Table 762. Parameters summary

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

5.240. VnicProfiles

Table 763. Methods summary

Name	Summary
`add`	
`list`	

5.240.1. add POST

Table 764. Parameters summary

Name	Type	Direction	Summary
`profile`	VnicProfile	In/Out	

5.240.2. list GET

Table 765. Parameters summary

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

max

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

5.241. Weight

Table 766. Methods summary

Name	Summary
`get`	
`remove`	

5.241.1. get GET

Table 767. Parameters summary

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

5.241.2. remove DELETE

Table 768. Parameters summary

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

5.242. Weights

Table 769. Methods summary

Name	Summary
`add`	
`list`	

5.242.1. add POST

Table 770. Parameters summary

Name	Type	Direction	Summary
`weight`	Weight	In/Out	

5.242.2. list GET

Table 771. Parameters summary

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

max

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

6. Types

This section enumerates all the data types that are available in the API.

6.1. AccessProtocol enum

Table 772. Values summary

Name	Summary
cifs	
gluster	
nfs	

6.2. Action struct

Table 773. Attributes summary

Name	Type	Summary
async	Boolean	
bricks	GlusterBrick[]	
certificates	Certificate[]	
check_connectivity	Boolean	
clone	Boolean	
cluster	Cluster	
collapse_snapshots	Boolean	
comment	String	Free text containing comments about this object.
connectivity_timeout	Integer	
data_center	DataCenter	

Name	Type	Summary
deploy_hosted_engine	Boolean	
description	String	A human readable description in plain text.
details	GlusterVolumeProfileDetails	
discard_snapshots	Boolean	
disk	Disk	
disks	Disk[]	
exclusive	Boolean	
fault	Fault	
fence_type	String	
filter	Boolean	
fix_layout	Boolean	
force	Boolean	
grace_period	GracePeriod	
host	Host	
id	String	
image	String	
import_as_template	Boolean	
is_attached	Boolean	
iscsi	IscsiDetails	
iscsi_targets	String[]	
job	Job	
logical_units	LogicalUnit[]	
maintenance_enabled	Boolean	
modified_bonds	HostNic[]	
modified_labels	NetworkLabel[]	
modified_network_attachments	NetworkAttachment[]	
name	String	A human readable name in plain text.
option	Option	
pause	Boolean	

Name	Type	Summary
power_management	PowerManagement	
proxy_ticket	ProxyTicket	
reason	String	
removed_bonds	HostNic[]	
removed_labels	NetworkLabel[]	
removed_network_attachments	NetworkAttachment[]	
resolution_type	String	
restore_memory	Boolean	
root_password	String	
snapshot	Snapshot	
ssh	Ssh	
status	String	
stop_gluster_service	Boolean	
storage_domain	StorageDomain	
storage_domains	StorageDomain[]	
succeeded	Boolean	
synchronized_network_attachments	NetworkAttachment[]	
template	Template	
ticket	Ticket	
undeploy_hosted_engine	Boolean	
use_cloud_init	Boolean	
use_sysprep	Boolean	
virtual_functions_configuration	HostNicVirtualFunctionsConfiguration	
vm	Vm	

6.3. AffinityGroup struct

An affinity group represents a group of virtual machines with a defined relationship.

Table 774. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`enforcing`	Boolean	Specifies whether the affinity group uses hard or soft enforcement of the affinity applied to virtual machines that are members of that affinity group.
`id`	String	
`name`	String	A human readable name in plain text.
`positive`	Boolean	Specifies whether the affinity group applies positive affinity or negative affinity to virtual machines that are members of that affinity group.

Table 775. Links summary

Name	Type	Summary
`cluster`	Cluster	A reference to the cluster to which the affinity group applies.
`vms`	Vm[]	List of all virtual machines assigned to this affinity group.

6.4. AffinityLabel struct

Affinity label is a label that can influence the VM scheduling. It is most often used to create a sub-cluster from available hosts.

Table 776. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.
`read_only`	Boolean	The `readOnly` property marks a label that can't be modified.

6.4.1. read_only

The `readOnly` property marks a label that can't be modified. This is mostly the case when listing internally generated labels.

Table 777. Links summary

Name	Type	Summary
hosts	Host[]	List of hosts that were labeled using this scheduling label.
vms	Vm[]	List of vms that were labeled using this scheduling label.

6.5. Agent struct

Table 778. Attributes summary

Name	Type	Summary
address	String	
comment	String	Free text containing comments about this object.
concurrent	Boolean	
description	String	A human readable description in plain text.
encrypt_options	Boolean	
id	String	
name	String	A human readable name in plain text.
options	Option[]	
order	Integer	
password	String	
port	Integer	
type	String	
username	String	

Table 779. Links summary

Name	Type	Summary
host	Host	

6.6. AgentConfiguration struct

Table 780. Attributes summary

Name	Type	Summary
address	String	
broker_type	MessageBrokerType	
network_mappings	String	
password	String	

Name	Type	Summary
port	Integer	
username	String	

6.7. Api struct

This type contains the information returned by root service of the API.

Table 781. Attributes summary

Name	Type	Summary
product_info	ProductInfo	Information about the product, like its name, the name of the vendor and the version.
special_objects	SpecialObjects	References to special objects, like the blank template and the root of the hierarchy of tags.
summary	ApiSummary	A summary containing the total number of relevant objects, like virtual machines, hosts and storage domains.
time	Date	The date and time when this information was generated.

6.8. ApiSummary struct

A summary containing the total number of relevant objects, like virtual machines, hosts and storage domains.

Table 782. Attributes summary

Name	Type	Summary
hosts	ApiSummaryItem	The summary of hosts.
storage_domains	ApiSummaryItem	The summary of storage domains.
users	ApiSummaryItem	The summary of users.
vms	ApiSummaryItem	The summary of virtual machines.

6.9. ApiSummaryItem struct

This type contains an item of the API summary. Each item contains the total and active number of some kind of object.

Table 783. Attributes summary

Name	Type	Summary
active	Integer	The total number of active objects.
total	Integer	The total number of objects.

6.10. Application struct

Table 784. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 785. Links summary

Name	Type	Summary
`vm`	Vm	

6.11. Architecture enum

Table 786. Values summary

Name	Summary
`ppc64`	
`undefined`	
`x86_64`	

6.12. AuthorizedKey struct

Table 787. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`key`	String	
`name`	String	A human readable name in plain text.

Table 788. Links summary

Name	Type	Summary
`user`	User	

6.13. AutoNumaStatus enum

Table 789. Values summary

Name	Summary
`disable`	
`enable`	
`unknown`	

6.14. Balance struct

Table 790. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 791. Links summary

Name	Type	Summary
`scheduling_policy`	SchedulingPolicy	
`scheduling_policy_unit`	SchedulingPolicyUnit	

6.15. Bios struct

Table 792. Attributes summary

Name	Type	Summary
`boot_menu`	BootMenu	

6.16. BlockStatistic struct

Table 793. Attributes summary

Name	Type	Summary
`statistics`	Statistic[]	

6.17. Bonding struct

Table 794. Attributes summary

Name	Type	Summary
ad_partner_mac	Mac	The ad_partner_mac property of the partner bond in mode 4.
options	Option[]	
slaves	HostNic[]	

6.17.1. ad_partner_mac

The ad_partner_mac property of the partner bond in mode 4. Bond mode 4 is the 802.3ad standard, also called dynamic link aggregation - Wikipedia, Presentation. ad_partner_mac is the MAC address of the system (switch) at the other end of a bond. This parameter is read only. Setting it will have no effect on the bond. It is retrieved from /sys/class/net/bondX/bonding/ad_partner_mac file on the system where the bond is located.

6.18. Bookmark struct

Table 795. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
value	String	

6.19. Boot struct

Table 796. Attributes summary

Name	Type	Summary
devices	BootDevice[]	

6.20. BootDevice enum

Table 797. Values summary

Name	Summary
cdrom	
hd	
network	

6.21. BootMenu struct

Table 798. Attributes summary

Name	Type	Summary
`enabled`	Boolean	

6.22. BootProtocol enum

The enum defines the options of the IP address assignment method to a NIC.

Table 799. Values summary

Name	Summary
`autoconf`	Stateless Address Auto-Configuration.
`dhcp`	Dynamic Host Configuration Protocol.
`none`	No address configuration.
`static`	Statically (by user) defined address, mask and gateway.

6.22.1. autoconf

Stateless Address Auto-Configuration. The mechanism is defined by RFC 4862. Please refer to this wikipedia article.

NOTE	The value is valid for IPv6 addresses only.

6.22.2. dhcp

Dynamic Host Configuration Protocol. Please refer to this wikipedia article.

6.23. BrickProfileDetail struct

Table 800. Attributes summary

Name	Type	Summary
`profile_details`	ProfileDetail[]	

Table 801. Links summary

Name	Type	Summary
`brick`	GlusterBrick	

6.24. Cdrom struct

Table 802. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`file`	File	
`id`	String	
`name`	String	A human readable name in plain text.

Table 803. Links summary

Name	Type	Summary
`instance_type`	InstanceType	
`template`	Template	
`vm`	Vm	Don't use this element, use `vms` instead.
`vms`	Vm[]	References to the virtual machines that are using this device.

6.24.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk my be used simultaneously by two or more virtual machines.

6.25. Certificate struct

Table 804. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`content`	String	
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.
`organization`	String	
`subject`	String	

6.26. CloudInit struct

Table 805. Attributes summary

Name	Type	Summary
`authorized_keys`	AuthorizedKey[]	
`files`	File[]	
`host`	Host	
`network_configuration`	NetworkConfiguration	
`regenerate_ssh_keys`	Boolean	
`timezone`	String	
`users`	User[]	

6.27. Cluster struct

Table 806. Attributes summary

Name	Type	Summary
`ballooning_enabled`	Boolean	
`comment`	String	Free text containing comments about this object.
`cpu`	Cpu	
`custom_scheduling_policy_properties`	Property[]	Custom scheduling policy properties of the cluster.
`description`	String	A human readable description in plain text.
`display`	Display	
`error_handling`	ErrorHandling	
`fencing_policy`	FencingPolicy	
`gluster_service`	Boolean	
`ha_reservation`	Boolean	
`id`	String	
`ksm`	Ksm	
`maintenance_reason_required`	Boolean	
`memory_policy`	MemoryPolicy	
`migration`	MigrationOptions	
`name`	String	A human readable name in plain text.

Name	Type	Summary
optional_reason	Boolean	
required_rng_sources	RngSource[]	
serial_number	SerialNumber	
supported_versions	Version[]	
switch_type	SwitchType	Type of switch to be used by all networks in given cluster.
threads_as_cores	Boolean	
trusted_service	Boolean	
tunnel_migration	Boolean	
version	Version	The compatibility version of the cluster.
virt_service	Boolean	

6.27.1. custom_scheduling_policy_properties

Custom scheduling policy properties of the cluster. These optional properties override the properties of the scheduling policy specified by the `scheduling_policy` link, and apply only for this specific cluster.

For example, to update the custom properties of the cluster, send a request:

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

With a request body:

```
<cluster>
  <custom_scheduling_policy_properties>
    <property>
      <name>HighUtilization</name>
      <value>70</value>
    </property>
  </custom_scheduling_policy_properties>
</cluster>
```

Update operations using the `custom_scheduling_policy_properties` attribute will not update the the properties of the scheduling policy specified by the `scheduling_policy` link, they will only be reflected on this specific cluster.

6.27.2. version

The compatibility version of the cluster.

All hosts in this cluster must support at least this compatibility version.

For example:

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

Will respond:

```
<cluster>
  ...
  <version>
    <major>4</major>
    <minor>0</minor>
  </version>
  ...
</cluster>
```

To update the compatibility version, use:

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

With a request body:

```
<cluster>
  <version>
    <major>4</major>
    <minor>1</minor>
  </version>
</cluster>
```

In order to update the cluster compatibility version, all hosts in the cluster must support the new compatibility version.

Table 807. Links summary

Name	Type	Summary
`affinity_groups`	AffinityGroup[]	
`cpu_profiles`	CpuProfile[]	
`data_center`	DataCenter	
`gluster_hooks`	GlusterHook[]	
`gluster_volumes`	GlusterVolume[]	
`management_network`	Network	

Name	Type	Summary
network_filters	NetworkFilter[]	
networks	Network[]	
permissions	Permission[]	
scheduling_policy	SchedulingPolicy	Reference to the default scheduling policy used by this cluster.

6.27.3. scheduling_policy

Reference to the default scheduling policy used by this cluster.

NOTE The scheduling policy properties are taken by default from the referenced scheduling policy, but they are overridden by the properties specified in the custom_scheduling_policy_properties attribute for this cluster.

6.28. ClusterLevel struct

Describes the capabilities supported by a specific cluster level.

Table 808. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
cpu_types	CpuType[]	The CPU types supported by this cluster level.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
permits	Permit[]	The permits supported by this cluster level.

6.29. Configuration struct

Table 809. Attributes summary

Name	Type	Summary
data	String	The document describing the virtual machine.
type	ConfigurationType	

6.29.1. data

The document describing the virtual machine.

Example of the OVF document:

```xml
<?xml version='1.0' encoding='UTF-8'?>
<ovf:Envelope xmlns:ovf="http://schemas.dmtf.org/ovf/envelope/1/"
  xmlns:rasd="http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/CIM_ResourceAllocationSettingData"
  xmlns:vssd="http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/CIM_VirtualSystemSettingData"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  ovf:version="3.5.0.0">
  <References/>
  <Section xsi:type="ovf:NetworkSection_Type">
    <Info>List of networks</Info>
    <Network ovf:name="Network 1"/>
  </Section>
  <Section xsi:type="ovf:DiskSection_Type">
    <Info>List of Virtual Disks</Info>
  </Section>
  <Content ovf:id="out" xsi:type="ovf:VirtualSystem_Type">
    <CreationDate>2014/12/03 04:25:45</CreationDate>
    <ExportDate>2015/02/09 14:12:24</ExportDate>
    <DeleteProtected>false</DeleteProtected>
    <SsoMethod>guest_agent</SsoMethod>
    <IsSmartcardEnabled>false</IsSmartcardEnabled>
    <TimeZone>Etc/GMT</TimeZone>
    <default_boot_sequence>0</default_boot_sequence>
    <Generation>1</Generation>
    <VmType>1</VmType>
    <MinAllocatedMem>1024</MinAllocatedMem>
    <IsStateless>false</IsStateless>
    <IsRunAndPause>false</IsRunAndPause>
    <AutoStartup>false</AutoStartup>
    <Priority>1</Priority>
    <CreatedByUserId>fdfc627c-d875-11e0-90f0-83df133b58cc</CreatedByUserId>
    <IsBootMenuEnabled>false</IsBootMenuEnabled>
    <IsSpiceFileTransferEnabled>true</IsSpiceFileTransferEnabled>
    <IsSpiceCopyPasteEnabled>true</IsSpiceCopyPasteEnabled>
    <Name>VM_export</Name>
    <TemplateId>00000000-0000-0000-0000-000000000000</TemplateId>
    <TemplateName>Blank</TemplateName>
    <IsInitilized>false</IsInitilized>
    <Origin>3</Origin>
    <DefaultDisplayType>1</DefaultDisplayType>
    <TrustedService>false</TrustedService>
    <OriginalTemplateId>00000000-0000-0000-0000-000000000000</OriginalTemplateId>
    <OriginalTemplateName>Blank</OriginalTemplateName>
    <UseLatestVersion>false</UseLatestVersion>
    <Section ovf:id="70b4d9a7-4f73-4def-89ca-24fc5f60e01a"
      ovf:required="false"
      xsi:type="ovf:OperatingSystemSection_Type">
      <Info>Guest Operating System</Info>
      <Description>other</Description>
```

```
    </Section>
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Info>1 CPU, 1024 Memeory</Info>
      <System>
        <vssd:VirtualSystemType>ENGINE 3.5.0.0</vssd:VirtualSystemType>
      </System>
      <Item>
        <rasd:Caption>1 virtual cpu</rasd:Caption>
        <rasd:Description>Number of virtual CPU</rasd:Description>
        <rasd:InstanceId>1</rasd:InstanceId>
        <rasd:ResourceType>3</rasd:ResourceType>
        <rasd:num_of_sockets>1</rasd:num_of_sockets>
        <rasd:cpu_per_socket>1</rasd:cpu_per_socket>
      </Item>
      <Item>
        <rasd:Caption>1024 MB of memory</rasd:Caption>
        <rasd:Description>Memory Size</rasd:Description>
        <rasd:InstanceId>2</rasd:InstanceId>
        <rasd:ResourceType>4</rasd:ResourceType>
        <rasd:AllocationUnits>MegaBytes</rasd:AllocationUnits>
        <rasd:VirtualQuantity>1024</rasd:VirtualQuantity>
      </Item>
      <Item>
        <rasd:Caption>USB Controller</rasd:Caption>
        <rasd:InstanceId>3</rasd:InstanceId>
        <rasd:ResourceType>23</rasd:ResourceType>
        <rasd:UsbPolicy>DISABLED</rasd:UsbPolicy>
      </Item>
    </Section>
  </Content>
</ovf:Envelope>
```

6.30. ConfigurationType enum

Table 810. Values summary

Name	Summary
ovf	

6.31. Console struct

Table 811. Attributes summary

Name	Type	Summary
enabled	Boolean	

6.32. Core struct

Table 812. Attributes summary

Name	Type	Summary
`index`	Integer	
`socket`	Integer	

6.33. Cpu struct

Table 813. Attributes summary

Name	Type	Summary
`architecture`	Architecture	
`cores`	Core[]	
`cpu_tune`	CpuTune	
`level`	Integer	
`mode`	CpuMode	
`name`	String	
`speed`	Decimal	
`topology`	CpuTopology	
`type`	String	

6.34. CpuMode enum

Table 814. Values summary

Name	Summary
`custom`	
`host_model`	
`host_passthrough`	

6.35. CpuProfile struct

Table 815. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	

Name	Type	Summary
`name`	String	A human readable name in plain text.

Table 816. Links summary

Name	Type	Summary
`cluster`	Cluster	
`permissions`	Permission[]	
`qos`	Qos	

6.36. CpuTopology struct

Table 817. Attributes summary

Name	Type	Summary
`cores`	Integer	
`sockets`	Integer	
`threads`	Integer	

6.37. CpuTune struct

Table 818. Attributes summary

Name	Type	Summary
`vcpu_pins`	VcpuPin[]	

6.38. CpuType struct

Describes a supported CPU type.

Table 819. Attributes summary

Name	Type	Summary
`architecture`	Architecture	The architecture of the CPU.
`level`	Integer	The level of the CPU type.
`name`	String	The name of the CPU type, for example `Intel Conroe Family`.

6.39. CreationStatus enum

Table 820. Values summary

Name	Summary
complete	
failed	
in_progress	
pending	

6.40. CustomProperty struct

Table 821. Attributes summary

Name	Type	Summary
name	String	
regexp	String	
value	String	

6.41. DataCenter struct

Table 822. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
local	Boolean	
name	String	A human readable name in plain text.
quota_mode	QuotaModeType	
status	DataCenterStatus	
storage_format	StorageFormat	
supported_versions	Version[]	
version	Version	The compatibility version of the data center.

6.41.1. version

The compatibility version of the data center.

All clusters in this data center must already be set to at least this compatibility version.

For example:

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

Will respond:

```
<data_center>
  ...
  <version>
    <major>4</major>
    <minor>0</minor>
  </version>
  ...
</data_center>
```

To update the compatibility version, use:

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

With a request body:

```
<data_center>
  <version>
    <major>4</major>
    <minor>1</minor>
  </version>
</data_center>
```

Table 823. Links summary

Name	Type	Summary
clusters	Cluster[]	Reference to clusters inside this data center.
iscsi_bonds	IscsiBond[]	Reference to ISCSI bonds used by this data center.
mac_pool	MacPool	Reference to the MAC pool used by this data center.
networks	Network[]	Reference to networks attached to this data center.
permissions	Permission[]	Reference to permissions assigned to this data center.
qoss	Qos[]	Reference to quality of service used by this data center.
quotas	Quota[]	Reference to quotas assigned to this data center.
storage_domains	StorageDomain[]	Reference to storage domains attached to this data center.

6.42. DataCenterStatus enum

Table 824. Values summary

Name	Summary
`contend`	
`maintenance`	
`not_operational`	
`problematic`	
`uninitialized`	
`up`	

6.43. Device struct

Table 825. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 826. Links summary

Name	Type	Summary
`instance_type`	InstanceType	
`template`	Template	
`vm`	Vm	Don't use this element, use `vms` instead.
`vms`	Vm[]	References to the virtual machines that are using this device.

6.43.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk my be used simultaneously by two or more virtual machines.

6.44. Disk struct

Represents a virtual disk device.

Table 827. Attributes summary

Name	Type	Summary
active	Boolean	Indicates if the disk is visible to the virtual machine.
actual_size	Integer	The actual size of the disk, in bytes.
alias	String	
bootable	Boolean	Indicates if the disk is marked as bootable.
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
format	DiskFormat	The underlying storage format.
id	String	
image_id	String	
interface	DiskInterface	The type of interface driver used to connect the disk device to the virtual machine.
logical_name	String	
lun_storage	HostStorage	
name	String	A human readable name in plain text.
propagate_errors	Boolean	Indicates if disk errors should not cause virtual machine to be paused and, instead, disk errors should be propagated to the the guest operating system.
provisioned_size	Integer	The virtual size of the disk, in bytes.
read_only	Boolean	Indicates if the disk is in read-only mode.
sgio	ScsiGenericIO	
shareable	Boolean	Indicates if the disk can be attached to multiple virtual machines.
sparse	Boolean	Indicates if the physical storage for the disk should not be preallocated.
status	DiskStatus	The status of the disk device.
storage_type	DiskStorageType	
uses_scsi_reservation	Boolean	
wipe_after_delete	Boolean	Indicates if the disk's blocks will be read back as zeros after it is deleted: - On block storage, the disk will be zeroed and only then deleted.

6.44.1. active

Indicates if the disk is visible to the virtual machine.

IMPORTANT	When adding a disk attachment to a virtual machine, the server accepts requests that don't contain this attribute, but the effect is then undefined. In some cases the disk will be automatically activated and in other cases it won't. To avoid issues it is strongly recommended to always include the this attribute with the desired value.

6.44.2. actual_size

The actual size of the disk, in bytes.

The actual size is the number of bytes actually used by the disk, and it will be smaller than the provisioned size for disks that use the `cow` format.

6.44.3. bootable

Indicates if the disk is marked as bootable.

IMPORTANT	This attribute only makes sense for disks that are actually connected to virtual machines, and in version 4 of the API it has been moved to the DiskAttachment type. It is preserved here only for backwards compatibility, and it will be removed in the future.

6.44.4. interface

The type of interface driver used to connect the disk device to the virtual machine.

IMPORTANT	This attribute only makes sense for disks that are actually connected to virtual machines, and in version 4 of the API it has been moved to the DiskAttachment type. It is preserved here only for backwards compatibility, and it will be removed in the future.

6.44.5. provisioned_size

The virtual size of the disk, in bytes.

This attribute is mandatory when creating a new disk.

6.44.6. shareable

Indicates if the disk can be attached to multiple virtual machines.

IMPORTANT	When a disk is attached to multiple virtual machines it is the responsibility of the guest operating systems of those virtual machines to coordinate access to it, to avoid corruption of the data, for example using a shared file system like GlusterFS or GFS.

6.44.7. wipe_after_delete

Indicates if the disk's blocks will be read back as zeros after it is deleted:

- On block storage, the disk will be zeroed and only then deleted.
- On file storage, since the file system already guarantees that previously removed blocks are read back as zeros, the disk will be deleted immediately.

Table 828. Links summary

Name	Type	Summary
`disk_profile`	DiskProfile	
`instance_type`	InstanceType	
`openstack_volume_type`	OpenStackVolumeType	
`permissions`	Permission[]	
`quota`	Quota	
`snapshot`	Snapshot	
`statistics`	Statistic[]	
`storage_domain`	StorageDomain	
`storage_domains`	StorageDomain[]	The storage domains associated with this disk.
`template`	Template	
`vm`	Vm	Don't use this element, use `vms` instead.
`vms`	Vm[]	References to the virtual machines that are using this device.

6.44.8. storage_domains

The storage domains associated with this disk.

NOTE	Only required when the first disk is being added to a virtual machine that was not itself created from a template.

6.44.9. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk my be used simultaneously by two or more virtual machines.

6.45. DiskAttachment struct

Describes how a disk is attached to a virtual machine.

Table 829. Attributes summary

Name	Type	Summary
`active`	Boolean	This flag indicates if the disk is active in the virtual machine it's attached to.
`bootable`	Boolean	Defines whether the disk is bootable.
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`interface`	DiskInterface	The type of interface driver used to connect the disk device to the virtual machine.
`logical_name`	String	The logical name of the virtual machine's disk, as seen from inside the virtual machine.
`name`	String	A human readable name in plain text.

6.45.1. active

This flag indicates if the disk is active in the virtual machine it's attached to.

A disk attached to a virtual machine in an active status is connected to the virtual machine at run time and can be used.

6.45.2. logical_name

The logical name of the virtual machine's disk, as seen from inside the virtual machine.

The logical name of a disk is reported only when the guest agent is installed and running inside the virtual machine.

For example, if the guest operating system is Linux and the disk is connected via a VirtIO interface, the logical name will be reported as `/dev/vda`:

```
<disk_attachment>
  ...
  <logical_name>/dev/vda</logical_name>
</disk_attachment>
```

If the guest operating system is Windows, the logical name will be reported as `\\.\PHYSICALDRIVE0`.

Table 830. Links summary

Name	Type	Summary
`disk`	Disk	The reference to the disk.
`template`	Template	The reference to the template.
`vm`	Vm	The reference to the virtual machine.

6.46. DiskFormat enum

The underlying storage format of disks.

Table 831. Values summary

Name	Summary
`cow`	The *Copy On Write* format allows snapshots, with a small performance overhead.
`raw`	The raw format does not allow snapshots, but offers improved performance.

6.47. DiskInterface enum

Table 832. Values summary

Name	Summary
`ide`	
`spapr_vscsi`	
`virtio`	
`virtio_scsi`	

6.48. DiskProfile struct

Table 833. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 834. Links summary

Name	Type	Summary
`permissions`	Permission[]	
`qos`	Qos	

Name	Type	Summary
storage_domain	StorageDomain	

6.49. DiskSnapshot struct

Table 835. Attributes summary

Name	Type	Summary
active	Boolean	Indicates if the disk is visible to the virtual machine.
actual_size	Integer	The actual size of the disk, in bytes.
alias	String	
bootable	Boolean	Indicates if the disk is marked as bootable.
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
format	DiskFormat	The underlying storage format.
id	String	
image_id	String	
interface	DiskInterface	The type of interface driver used to connect the disk device to the virtual machine.
logical_name	String	
lun_storage	HostStorage	
name	String	A human readable name in plain text.
propagate_errors	Boolean	Indicates if disk errors should not cause virtual machine to be paused and, instead, disk errors should be propagated to the the guest operating system.
provisioned_size	Integer	The virtual size of the disk, in bytes.
read_only	Boolean	Indicates if the disk is in read-only mode.
sgio	ScsiGenericIO	
shareable	Boolean	Indicates if the disk can be attached to multiple virtual machines.
sparse	Boolean	Indicates if the physical storage for the disk should not be preallocated.
status	DiskStatus	The status of the disk device.
storage_type	DiskStorageType	
uses_scsi_reservation	Boolean	

Name	Type	Summary
`wipe_after_delete`	Boolean	Indicates if the disk's blocks will be read back as zeros after it is deleted: - On block storage, the disk will be zeroed and only then deleted.

6.49.1. active

Indicates if the disk is visible to the virtual machine.

> **IMPORTANT**
> When adding a disk attachment to a virtual machine, the server accepts requests that don't contain this attribute, but the effect is then undefined. In some cases the disk will be automatically activated and in other cases it won't. To avoid issues it is strongly recommended to always include the this attribute with the desired value.

6.49.2. actual_size

The actual size of the disk, in bytes.

The actual size is the number of bytes actually used by the disk, and it will be smaller than the provisioned size for disks that use the `cow` format.

6.49.3. bootable

Indicates if the disk is marked as bootable.

> **IMPORTANT**
> This attribute only makes sense for disks that are actually connected to virtual machines, and in version 4 of the API it has been moved to the DiskAttachment type. It is preserved here only for backwards compatibility, and it will be removed in the future.

6.49.4. interface

The type of interface driver used to connect the disk device to the virtual machine.

> **IMPORTANT**
> This attribute only makes sense for disks that are actually connected to virtual machines, and in version 4 of the API it has been moved to the DiskAttachment type. It is preserved here only for backwards compatibility, and it will be removed in the future.

6.49.5. provisioned_size

The virtual size of the disk, in bytes.

This attribute is mandatory when creating a new disk.

6.49.6. shareable

Indicates if the disk can be attached to multiple virtual machines.

IMPORTANT	When a disk is attached to multiple virtual machines it is the responsibility of the guest operating systems of those virtual machines to coordinate access to it, to avoid corruption of the data, for example using a shared file system like GlusterFS or GFS.

6.49.7. wipe_after_delete

Indicates if the disk's blocks will be read back as zeros after it is deleted:

- On block storage, the disk will be zeroed and only then deleted.
- On file storage, since the file system already guarantees that previously removed blocks are read back as zeros, the disk will be deleted immediately.

Table 836. Links summary

Name	Type	Summary
`disk`	Disk	
`disk_profile`	DiskProfile	
`instance_type`	InstanceType	
`openstack_volume_type`	OpenStackVolumeType	
`permissions`	Permission[]	
`quota`	Quota	
`snapshot`	Snapshot	
`statistics`	Statistic[]	
`storage_domain`	StorageDomain	
`storage_domains`	StorageDomain[]	The storage domains associated with this disk.
`template`	Template	
`vm`	Vm	Don't use this element, use `vms` instead.
`vms`	Vm[]	References to the virtual machines that are using this device.

6.49.8. storage_domains

The storage domains associated with this disk.

NOTE	Only required when the first disk is being added to a virtual machine that was not itself created from a template.

6.49.9. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk my be used simultaneously by two or more virtual machines.

6.50. DiskStatus enum

Table 837. Values summary

Name	Summary
`illegal`	
`locked`	
`ok`	

6.51. DiskStorageType enum

Table 838. Values summary

Name	Summary
`cinder`	
`image`	
`lun`	

6.52. DiskType enum

Table 839. Values summary

Name	Summary
`data`	
`system`	

6.53. Display struct

Table 840. Attributes summary

Name	Type	Summary
`address`	String	
`allow_override`	Boolean	
`certificate`	Certificate	
`copy_paste_enabled`	Boolean	
`disconnect_action`	String	

Name	Type	Summary
file_transfer_enabled	Boolean	
keyboard_layout	String	
monitors	Integer	
port	Integer	
proxy	String	
secure_port	Integer	
single_qxl_pci	Boolean	
smartcard_enabled	Boolean	
type	DisplayType	

6.54. DisplayType enum

Table 841. Values summary

Name	Summary
spice	
vnc	

6.55. Dns struct

Table 842. Attributes summary

Name	Type	Summary
search_domains	Host[]	
servers	Host[]	

6.56. Domain struct

Table 843. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
user	User	

Table 844. Links summary

Name	Type	Summary
`groups`	Group[]	
`users`	User[]	

6.57. EntityExternalStatus enum

Table 845. Values summary

Name	Summary
`error`	
`failure`	
`info`	
`ok`	
`warning`	

6.58. EntityProfileDetail struct

Table 846. Attributes summary

Name	Type	Summary
`profile_details`	ProfileDetail[]	

6.59. ErrorHandling struct

Table 847. Attributes summary

Name	Type	Summary
`on_error`	MigrateOnError	

6.60. Event struct

Table 848. Attributes summary

Name	Type	Summary
`code`	Integer	
`comment`	String	Free text containing comments about this object.
`correlation_id`	String	
`custom_data`	String	
`custom_id`	Integer	

Name	Type	Summary
description	String	A human readable description in plain text.
flood_rate	Integer	
id	String	
name	String	A human readable name in plain text.
origin	String	
severity	LogSeverity	
time	Date	

Table 849. Links summary

Name	Type	Summary
cluster	Cluster	
data_center	DataCenter	
host	Host	
storage_domain	StorageDomain	
template	Template	
user	User	
vm	Vm	

6.61. ExternalComputeResource struct

Table 850. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
provider	String	
url	String	
user	String	

Table 851. Links summary

Name	Type	Summary
external_host_provider	ExternalHostProvider	

6.62. ExternalDiscoveredHost struct

Table 852. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`ip`	String	
`last_report`	String	
`mac`	String	
`name`	String	A human readable name in plain text.
`subnet_name`	String	

Table 853. Links summary

Name	Type	Summary
`external_host_provider`	ExternalHostProvider	

6.63. ExternalHost struct

Table 854. Attributes summary

Name	Type	Summary
`address`	String	
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 855. Links summary

Name	Type	Summary
`external_host_provider`	ExternalHostProvider	

6.64. ExternalHostGroup struct

Table 856. Attributes summary

Name	Type	Summary
architecture_name	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
domain_name	String	
id	String	
name	String	A human readable name in plain text.
operating_system_name	String	
subnet_name	String	

Table 857. Links summary

Name	Type	Summary
external_host_provider	ExternalHostProvider	

6.65. ExternalHostProvider struct

Table 858. Attributes summary

Name	Type	Summary
authentication_url	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
password	String	
properties	Property[]	
requires_authentication	Boolean	
url	String	
username	String	

Table 859. Links summary

Name	Type	Summary
certificates	Certificate[]	

Name	Type	Summary
compute_resources	ExternalComputeResource[]	
discovered_hosts	ExternalDiscoveredHost[]	
host_groups	ExternalHostGroup[]	
hosts	Host[]	

6.66. ExternalProvider struct

Table 860. Attributes summary

Name	Type	Summary
authentication_url	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
password	String	
properties	Property[]	
requires_authentication	Boolean	
url	String	
username	String	

6.67. ExternalStatus enum

Table 861. Values summary

Name	Summary
error	
failure	
info	
ok	
warning	

6.68. ExternalSystemType enum

Table 862. Values summary

Name	Summary
`gluster`	
`vdsm`	

6.69. ExternalVmImport struct

Describes parameters of virtual machine import operation from external system.

Table 863. Attributes summary

Name	Type	Summary
`name`	String	Name of the virtual machine to be imported as is defined within the external system.
`password`	String	Password to authenticate against external hypervisor system.
`provider`	ExternalVmProviderType	Type of external virtual machine provider.
`sparse`	Boolean	Specifies the disk allocation policy of resulting virtual machine: `true` for sparse, `false` for preallocated.
`url`	String	URL to be passed to the `virt-v2v` tool for conversion.
`username`	String	Username to authenticate against external hypervisor system.

6.69.1. url

URL to be passed to the `virt-v2v` tool for conversion.

Example:

```
vpx://wmware_user@vcenter-host/DataCenter/Cluster/esxi-host?no_verify=1
```

More examples can be found at http://libguestfs.org/virt-v2v.1.html.

Table 864. Links summary

Name	Type	Summary
`cluster`	Cluster	Specifies the target cluster of the resulting virtual machine.
`cpu_profile`	CpuProfile	Optionally specifies the cpu profile of the resulting virtual machine.

Name	Type	Summary
`drivers_iso`	File	Optional name of ISO carrying drivers that can be used during the virt-v2v conversion process.
`host`	Host	Optional specification of host (using host's ID) to be used for the conversion process.
`quota`	Quota	Optionally specifies the quota that will be applied to the resulting virtual machine.
`storage_domain`	StorageDomain	Specifies the target storage domain for converted disks.
`vm`	Vm	Virtual machine entity used to specify the name of the newly created virtual machine.

6.69.2. host

Optional specification of host (using host's ID) to be used for the conversion process. If not specified, one is selected automatically.

6.69.3. vm

Virtual machine entity used to specify the name of the newly created virtual machine.

If name is not specified, the source virtual machine name will be used.

6.70. ExternalVmProviderType enum

Describes type of external hypervisor system.

Table 865. Values summary

Name	Summary
`kvm`	
`vmware`	
`xen`	

6.71. Fault struct

Table 866. Attributes summary

Name	Type	Summary
`detail`	String	
`reason`	String	

6.72. FenceType enum

Table 867. Values summary

Name	Summary
`manual`	
`restart`	
`start`	
`status`	
`stop`	

6.73. FencingPolicy struct

Table 868. Attributes summary

Name	Type	Summary
`enabled`	Boolean	
`skip_if_connectivity_broken`	SkipIfConnectivityBroken	
`skip_if_sd_active`	SkipIfSdActive	

6.74. File struct

Table 869. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`content`	String	
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.
`type`	String	

Table 870. Links summary

Name	Type	Summary
`storage_domain`	StorageDomain	

6.75. Filter struct

Table 871. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
position	Integer	

Table 872. Links summary

Name	Type	Summary
scheduling_policy_unit	SchedulingPolicyUnit	

6.76. Floppy struct

Table 873. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
file	File	
id	String	
name	String	A human readable name in plain text.

Table 874. Links summary

Name	Type	Summary
instance_type	InstanceType	
template	Template	
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

6.76.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk my be used simultaneously by two or more virtual machines.

6.77. FopStatistic struct

Table 875. Attributes summary

Name	Type	Summary
`name`	String	
`statistics`	Statistic[]	

6.78. GlusterBrick struct

Table 876. Attributes summary

Name	Type	Summary
`brick_dir`	String	
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`device`	String	
`fs_name`	String	
`gluster_clients`	GlusterClient[]	
`id`	String	
`memory_pools`	GlusterMemoryPool[]	
`mnt_options`	String	
`name`	String	A human readable name in plain text.
`pid`	Integer	
`port`	Integer	
`server_id`	String	
`status`	GlusterBrickStatus	

Table 877. Links summary

Name	Type	Summary
`gluster_volume`	GlusterVolume	
`instance_type`	InstanceType	
`statistics`	Statistic[]	
`template`	Template	
`vm`	Vm	Don't use this element, use `vms` instead.

Name	Type	Summary
vms	Vm[]	References to the virtual machines that are using this device.

6.78.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk my be used simultaneously by two or more virtual machines.

6.79. GlusterBrickAdvancedDetails struct

Table 878. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
device	String	
fs_name	String	
gluster_clients	GlusterClient[]	
id	String	
memory_pools	GlusterMemoryPool[]	
mnt_options	String	
name	String	A human readable name in plain text.
pid	Integer	
port	Integer	

Table 879. Links summary

Name	Type	Summary
instance_type	InstanceType	
template	Template	
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

6.79.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk my be used simultaneously by two or more virtual

machines.

6.80. GlusterBrickMemoryInfo struct

Table 880. Attributes summary

Name	Type	Summary
`memory_pools`	GlusterMemoryPool[]	

6.81. GlusterBrickStatus enum

Table 881. Values summary

Name	Summary
`down`	Brick is in `down` state, the data cannot be stored or retrieved from it.
`unknown`	When the status cannot be determined due to host being non-responsive.
`up`	Brick is in `up` state, the data can be stored or retrieved from it.

6.82. GlusterClient struct

Table 882. Attributes summary

Name	Type	Summary
`bytes_read`	Integer	
`bytes_written`	Integer	
`client_port`	Integer	
`host_name`	String	

6.83. GlusterHook struct

Table 883. Attributes summary

Name	Type	Summary
`checksum`	String	
`comment`	String	Free text containing comments about this object.
`conflict_status`	Integer	
`conflicts`	String	
`content`	String	
`content_type`	HookContentType	

Name	Type	Summary
description	String	A human readable description in plain text.
gluster_command	String	
id	String	
name	String	A human readable name in plain text.
stage	HookStage	
status	GlusterHookStatus	

Table 884. Links summary

Name	Type	Summary
cluster	Cluster	
server_hooks	GlusterServerHook[]	

6.84. GlusterHookStatus enum

Table 885. Values summary

Name	Summary
disabled	Hook is disabled in the cluster.
enabled	Hook is enabled in the cluster.
missing	Unknown/missing hook status.

6.85. GlusterMemoryPool struct

Table 886. Attributes summary

Name	Type	Summary
alloc_count	Integer	
cold_count	Integer	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
hot_count	Integer	
id	String	
max_alloc	Integer	
max_stdalloc	Integer	
name	String	A human readable name in plain text.

Name	Type	Summary
padded_size	Integer	
pool_misses	Integer	
type	String	

6.86. GlusterServerHook struct

Table 887. Attributes summary

Name	Type	Summary
checksum	String	
comment	String	Free text containing comments about this object.
content_type	HookContentType	
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
status	GlusterHookStatus	

Table 888. Links summary

Name	Type	Summary
host	Host	

6.87. GlusterState enum

Table 889. Values summary

Name	Summary
down	
unknown	
up	

6.88. GlusterVolume struct

Table 890. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
disperse_count	Integer	

Name	Type	Summary
id	String	
name	String	A human readable name in plain text.
options	Option[]	
redundancy_count	Integer	
replica_count	Integer	
status	GlusterVolumeStatus	
stripe_count	Integer	
transport_types	TransportType[]	
volume_type	GlusterVolumeType	

Table 891. Links summary

Name	Type	Summary
bricks	GlusterBrick[]	
cluster	Cluster	
statistics	Statistic[]	

6.89. GlusterVolumeProfileDetails struct

Table 892. Attributes summary

Name	Type	Summary
brick_profile_details	BrickProfileDetail[]	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
nfs_profile_details	NfsProfileDetail[]	

6.90. GlusterVolumeStatus enum

Table 893. Values summary

Name	Summary
down	Volume needs to be started, for clients to be able to mount and use it.
unknown	When the status cannot be determined due to host being non-responsive.
up	Volume is started, and can be mounted and used by clients.

6.91. GlusterVolumeType enum

Table 894. Values summary

Name	Summary
disperse	
distribute	
distributed_disperse	
distributed_replicate	
distributed_stripe	
distributed_striped_replicate	
replicate	
stripe	
striped_replicate	

6.92. GracePeriod struct

Table 895. Attributes summary

Name	Type	Summary
expiry	Integer	

6.93. GraphicsConsole struct

Table 896. Attributes summary

Name	Type	Summary
address	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.

Name	Type	Summary
port	Integer	
protocol	GraphicsType	
tls_port	Integer	

Table 897. Links summary

Name	Type	Summary
instance_type	InstanceType	
template	Template	
vm	Vm	

6.94. GraphicsType enum

Table 898. Values summary

Name	Summary
spice	
vnc	

6.95. Group struct

Table 899. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
domain_entry_id	String	
id	String	
name	String	A human readable name in plain text.
namespace	String	

Table 900. Links summary

Name	Type	Summary
domain	Domain	
permissions	Permission[]	
roles	Role[]	Used only to represent the initial role assignments for a new group, thereafter modification of role assignments are only supported via the `roles` sub-collection.

Name	Type	Summary
tags	Tag[]	

6.96. GuestOperatingSystem struct

Table 901. Attributes summary

Name	Type	Summary
architecture	String	
codename	String	
distribution	String	
family	String	
kernel	Kernel	
version	Version	

6.97. HardwareInformation struct

Table 902. Attributes summary

Name	Type	Summary
family	String	
manufacturer	String	
product_name	String	
serial_number	String	
supported_rng_sources	RngSource[]	
uuid	String	
version	String	

6.98. HighAvailability struct

Table 903. Attributes summary

Name	Type	Summary
enabled	Boolean	
priority	Integer	

6.99. Hook struct

Table 904. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`event_name`	String	
`id`	String	
`md5`	String	
`name`	String	A human readable name in plain text.

Table 905. Links summary

Name	Type	Summary
`host`	Host	

6.100. HookContentType enum

Table 906. Values summary

Name	Summary
`binary`	
`text`	

6.101. HookStage enum

Table 907. Values summary

Name	Summary
`post`	
`pre`	

6.102. HookStatus enum

Table 908. Values summary

Name	Summary
`disabled`	
`enabled`	
`missing`	

6.103. Host struct

Table 909. Attributes summary

Name	Type	Summary
address	String	
auto_numa_status	AutoNumaStatus	
certificate	Certificate	
comment	String	Free text containing comments about this object.
cpu	Cpu	
description	String	A human readable description in plain text.
device_passthrough	HostDevicePassthrough	
display	Display	Optionally specify the display address of this host explicitly.
external_status	ExternalStatus	
hardware_information	HardwareInformation	
hosted_engine	HostedEngine	
id	String	
iscsi	IscsiDetails	
kdump_status	KdumpStatus	
ksm	Ksm	
libvirt_version	Version	
max_scheduling_memory	Integer	
memory	Integer	
name	String	A human readable name in plain text.
numa_supported	Boolean	
os	OperatingSystem	
override_iptables	Boolean	
port	Integer	
power_management	PowerManagement	
protocol	HostProtocol	

Name	Type	Summary
root_password	String	When creating a new host, a root password is required if the password authentication method is chosen, but this is not subsequently included in the representation.
se_linux	SeLinux	
spm	Spm	
ssh	Ssh	
status	HostStatus	
status_detail	String	
summary	VmSummary	
transparent_huge_pages	TransparentHugePages	
type	HostType	Indicates if the host contains a full installation of the operating system or a scaled-down version intended only to host virtual machines.
update_available	Boolean	
version	Version	The version of VDSM.

6.103.1. version

The version of VDSM.

For example:

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

This GET request will return the following output:

```
<host>
  ...
  <version>
    <build>999</build>
    <full_version>vdsm-4.18.999-419.gitcf06367.el7</full_version>
    <major>4</major>
    <minor>18</minor>
    <revision>0</revision>
  </version>
  ...
</host>
```

Table 910. Links summary

Name	Type	Summary
affinity_labels	AffinityLabel[]	
agents	Agent[]	
cluster	Cluster	
devices	Device[]	
external_host_provider	ExternalHostProvider	
hooks	Hook[]	
katello_errata	KatelloErratum[]	
network_attachments	NetworkAttachment[]	
nics	Nic[]	
numa_nodes	NumaNode[]	
permissions	Permission[]	
statistics	Statistic[]	
storage_connection_extensions	StorageConnectionExtension[]	
storages	HostStorage[]	
tags	Tag[]	
unmanaged_networks	UnmanagedNetwork[]	

6.104. HostDevice struct

Table 911. Attributes summary

Name	Type	Summary
capability	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
iommu_group	Integer	
name	String	A human readable name in plain text.
physical_function	HostDevice	
placeholder	Boolean	
product	Product	

Name	Type	Summary
`vendor`	Vendor	
`virtual_functions`	Integer	

Table 912. Links summary

Name	Type	Summary
`host`	Host	
`parent_device`	HostDevice	
`vm`	Vm	

6.105. HostDevicePassthrough struct

Table 913. Attributes summary

Name	Type	Summary
`enabled`	Boolean	

6.106. HostNic struct

Represents a host NIC.

For example, the XML representation of a host NIC looks like this:

```
<host_nic href="/ovirt-engine/api/hosts/123/nics/456" id="456">
  <name>eth0</name>
  <boot_protocol>static</boot_protocol>
  <bridged>true</bridged>
  <custom_configuration>true</custom_configuration>
  <ip>
    <address>192.168.122.39</address>
    <gateway>192.168.122.1</gateway>
    <netmask>255.255.255.0</netmask>
    <version>v4</version>
  </ip>
  <ipv6>
    <gateway>::</gateway>
    <version>v6</version>
  </ipv6>
  <ipv6_boot_protocol>none</ipv6_boot_protocol>
  <mac>
    <address>52:54:00:0c:79:1d</address>
  </mac>
  <mtu>1500</mtu>
  <status>up</status>
</host_nic>
```

Table 914. Attributes summary

Name	Type	Summary
`ad_aggregator_id`	Integer	The `ad_aggregator_id` property of a bond or bond slave, for bonds in mode 4.
`base_interface`	String	
`bonding`	Bonding	
`boot_protocol`	BootProtocol	
`bridged`	Boolean	
`check_connectivity`	Boolean	
`comment`	String	Free text containing comments about this object.
`custom_configuration`	Boolean	
`description`	String	A human readable description in plain text.
`id`	String	
`ip`	Ip	
`ipv6`	Ip	
`ipv6_boot_protocol`	BootProtocol	
`mac`	Mac	The MAC address of the NIC.

Name	Type	Summary
mtu	Integer	
name	String	A human readable name in plain text.
network_labels	NetworkLabel[]	
override_configuration	Boolean	
properties	Property[]	
speed	Integer	
statistics	Statistic[]	A link to the statistics of the NIC.
status	NicStatus	
virtual_functions_configuration	HostNicVirtualFunctionsConfiguration	For a SR-IOV physical function NIC describes its virtual functions configuration.
vlan	Vlan	

6.106.1. ad_aggregator_id

The `ad_aggregator_id` property of a bond or bond slave, for bonds in mode 4. Bond mode 4 is the 802.3ad standard, also called dynamic link aggregation - Wikipedia Presentation. This is only valid for bonds in mode 4, or NICs (NIC - network interface card) which are part of a bond. It is not present for bonds in other modes, or NICs which are not part in a bond in mode 4. The `ad_aggregator_id` property indicates which of the bond slaves are active. The value of the `ad_aggregator_id` of an active slave is the same the value of the `ad_aggregator_id` property of the bond. This parameter is read only. Setting it will have no effect on the bond/NIC. It is retrieved from `/sys/class/net/bondX/bonding/ad_aggregator` file for a bond, and the `/sys/class/net/ensX/bonding_slave/ad_aggregator_id` file for a NIC.

Table 915. Links summary

Name	Type	Summary
host	Host	
network	Network	A reference to the network which the interface should be connected.
physical_function	HostNic	For a SR-IOV virtual function NIC references to its physical function NIC.
qos	Qos	

6.106.2. network

A reference to the network which the interface should be connected. A blank network id is allowed.

6.107. HostNicVirtualFunctionsConfiguration struct

Describes virtual functions configuration for an SR-IOV enabled physical function NIC.

Table 916. Attributes summary

Name	Type	Summary
`all_networks_allowed`	Boolean	Defines whether all networks are allowed to be defined on the related virtual functions or specified ones only.
`max_number_of_virtual_functions`	Integer	Maximum number of virtual functions the NIC supports.
`number_of_virtual_functions`	Integer	Number of curently defined virtual functions.

6.107.1. max_number_of_virtual_functions

Maximum number of virtual functions the NIC supports. Read-only property.

6.107.2. number_of_virtual_functions

Number of curently defined virtual functions. User-defined value between 0 and `maxNumberOfVirtualFunctions`.

6.108. HostProtocol enum

Table 917. Values summary

Name	Summary
`stomp`	
`xml`	

6.109. HostStatus enum

Table 918. Values summary

Name	Summary
`connecting`	
`down`	
`error`	
`initializing`	
`install_failed`	
`installing`	
`installing_os`	
`kdumping`	

Name	Summary
maintenance	
non_operational	
non_responsive	
pending_approval	
preparing_for_maintenance	
reboot	
unassigned	
up	

6.110. HostStorage struct

Table 919. Attributes summary

Name	Type	Summary
address	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
logical_units	LogicalUnit[]	
mount_options	String	
name	String	A human readable name in plain text.
nfs_retrans	Integer	The number of times to retry a request before attempting further recovery actions.
nfs_timeo	Integer	The time in tenths of a second to wait for a response before retrying NFS requests.
nfs_version	NfsVersion	
override_luns	Boolean	
password	String	
path	String	
port	Integer	
portal	String	
target	String	
type	StorageType	
username	String	

Name	Type	Summary
vfs_type	String	
volume_group	VolumeGroup	

6.110.1. nfs_retrans

The number of times to retry a request before attempting further recovery actions. The value must be in the range of 0 to 65535. For more details see the description of the `retrans` mount option in the `nfs` man page.

6.110.2. nfs_timeo

The time in tenths of a second to wait for a response before retrying NFS requests. The value must be in the range of 0 to 65535. For more details see the description of the `timeo` mount option in the `nfs` man page.

Table 920. Links summary

Name	Type	Summary
host	Host	

6.111. HostType enum

This enumerated type is used to what type of operating system is used by the host.

Table 921. Values summary

Name	Summary
ovirt_node	The host is NGN (Next Generation Node) - a new implementation of RHEV_H which is like RHEL, CentOS or Fedora installation.
rhel	The host contains a full RHEL, CentOS or Fedora installation.
rhev_h	The host contains a small scaled version of RHEL, CentOS or Fedora, used solely to host virtual machines.

6.111.1. ovirt_node

The host is NGN (Next Generation Node) - a new implementation of RHEV_H which is like RHEL, CentOS or Fedora installation. The main difference between NGN and legacy RHEV-H is that NGN has a writeable file system and will handle its installation instead of pushing RPMs to it by the engine in legacy RHEV-H.

6.112. HostedEngine struct

Table 922. Attributes summary

Name	Type	Summary
`active`	Boolean	
`configured`	Boolean	
`global_maintenance`	Boolean	
`local_maintenance`	Boolean	
`score`	Integer	

6.113. Icon struct

Table 923. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`data`	String	
`description`	String	A human readable description in plain text.
`id`	String	
`media_type`	String	
`name`	String	A human readable name in plain text.

6.114. Identified struct

This interface is intended to be the base for all types of the model that represents objects with an identifier.

Table 924. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

6.115. Image struct

Table 925. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.

Table 926. Links summary

Name	Type	Summary
storage_domain	StorageDomain	

6.116. ImageTransfer struct

This type contains information regarding an image transfer being performed.

Table 927. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
phase	ImageTransferPhase	The current phase of the image transfer being made.
proxy_url	String	The URL of the proxy server which the user should do I/O to.
signed_ticket	String	The signed ticket that should be attached as an `Authentication` header in the HTTPS request for the proxy server to do I/O to (See `proxy_url` attribute).

6.116.1. phase

The current phase of the image transfer being made. Each transfer needs a managed session, which must be opened in order for the user to be able to do I/O to an image. Please refer to image transfer for further documentation.

6.116.2. proxy_url

The URL of the proxy server which the user should do I/O to. This attribute is available only if the image transfer entity is in transferring phase. See `phase` for details.

Table 928. Links summary

Name	Type	Summary
host	Host	The host which will be used to write to the image which is targeted for I/O.
image	Image	The image which is targeted for I/O.

6.117. ImageTransferPhase enum

A list of possible phases for an image transfer entity. Each of these values defines a specific point in a transfer flow.

Please refer to image transfer for further documentation.

Table 929. Values summary

Name	Summary
cancelled	This phase will be set as a result of the user actively cancelling the transfer, which can only be performed via the webadmin.
finalizing_failure	This phase can only be set by oVirt webadmin, and states that there was some error during the transfer and it is being finalized with failure.
finalizing_success	This phase will be set when the user calls finalize.
finished_failure	States that the targeted image failed the verification, and cannot be used.
finished_success	States that the transfer session was successfully closed, and the targeted image was verified and ready to be used.
initializing	The initial phase of an added image transfer.
paused_system	This phase stands for a situation in which the session was timed out, or if some other error occurred with this transfer, e.
paused_user	This phase will be set as a result of an active pause call by the user, using pause.
resuming	The phase where the transfer had been resumed by the client calling resume.
transferring	The phase where the transfer session is open, and the client can do I/O to the desired image using its preferred tools.
unknown	An unknown phase, will only be set in cases of unpredictable errors.

6.117.1. finalizing_success

This phase will be set when the user calls finalize. Calling finalize is essential in order to finish the transfer session and using the targeted image. After finalizing, the phase will be changed to finished_success or finished_failure.

Please refer to image transfer for further documentation.

6.117.2. finished_failure

States that the targeted image failed the verification, and cannot be used. After reaching this phase, the image transfer entity will soon be deleted, and the targeted image will be set to illegal.

6.117.3. finished_success

States that the transfer session was successfully closed, and the targeted image was verified and ready to be used. After reaching this phase, the image transfer entity will soon be deleted.

6.117.4. initializing

The initial phase of an added image transfer. It is set as long as the transfer session is establishing. Once the session is established, the phase will be changed to `transferring`

6.117.5. paused_system

This phase stands for a situation in which the session was timed out, or if some other error occurred with this transfer, e.g, ovirt-imageio-daemon is not running in the selected host for transferring. For trying to resume the session, the client should call resume. After resuming, the phase will change to `resuming`

6.117.6. resuming

The phase where the transfer had been resumed by the client calling resume. Resuming will start a new session, and after calling it, the phase should be changed to `transferring`, or `paused_system` in case of a failure.

6.118. InheritableBoolean enum

Enum representing boolean value that can be either set or inherited from higher level. Usual inheritance order is VM → Cluster → engine-config.

Table 930. Values summary

Name	Summary
`false`	Set value to false on this level.
`inherit`	Inherit value from higher level.
`true`	Set value to true on this level.

6.119. Initialization struct

Table 931. Attributes summary

Name	Type	Summary
`active_directory_ou`	String	

Name	Type	Summary
authorized_ssh_keys	String	
cloud_init	CloudInit	
configuration	Configuration	
custom_script	String	
dns_search	String	
dns_servers	String	
domain	String	
host_name	String	
input_locale	String	
nic_configurations	NicConfiguration[]	
org_name	String	
regenerate_ids	Boolean	
regenerate_ssh_keys	Boolean	
root_password	String	
system_locale	String	
timezone	String	
ui_language	String	
user_locale	String	
user_name	String	
windows_license_key	String	

6.120. InstanceType struct

Table 932. Attributes summary

Name	Type	Summary
bios	Bios	
comment	String	Free text containing comments about this object.
console	Console	
cpu	Cpu	The configuration of the CPU of the virtual machine.
cpu_shares	Integer	
creation_time	Date	

Name	Type	Summary
custom_compatibility_version	Version	Virtual machine custom compatibility version.
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	
delete_protected	Boolean	
description	String	A human readable description in plain text.
display	Display	
domain	Domain	
high_availability	HighAvailability	
id	String	
initialization	Initialization	
io	Io	
large_icon	Icon	
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	
migration	MigrationOptions	
migration_downtime	Integer	
name	String	A human readable name in plain text.
origin	String	
os	OperatingSystem	
rng_device	RngDevice	
serial_number	SerialNumber	
small_icon	Icon	
soundcard_enabled	Boolean	
sso	Sso	
start_paused	Boolean	
stateless	Boolean	
status	TemplateStatus	
time_zone	TimeZone	
tunnel_migration	Boolean	

Name	Type	Summary
type	VmType	
usb	Usb	
version	TemplateVersion	
virtio_scsi	VirtioScsi	
vm	Vm	

6.120.1. cpu

The configuration of the CPU of the virtual machine.

The sockets configuration can be updated without rebooting the virtual machine. The cores and the threads require a reboot in order to take place.

For example, to change the number of sockets to 4 immediately and the number of cores and threads to 2 after reboot send a request:

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

With a request body:

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

6.120.2. custom_compatibility_version

Virtual machine custom compatibility version.

This field allows to customize a virtual machine to its own compatibility version. If `custom_compatibility_version` is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in and is checked against capabilities of the host the virtual machine is planned to run on.

6.120.3. memory

The virtual machine's memory, in bytes.

For example, in order to update a virtual machine to contain 1 GiB of memory send a request like this:

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

With a request body like this:

```
<vm>
  <memory>1073741824</memory>
</vm>
```

NOTE Memory in the previous example is converted to bytes using the following formula: 1 GiB = 2^{30} bytes = 1073741824 bytes.

NOTE Memory hot plug is supported from oVirt 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

Table 933. Links summary

Name	Type	Summary
`cdroms`	Cdrom[]	
`cluster`	Cluster	
`cpu_profile`	CpuProfile	
`disk_attachments`	DiskAttachment[]	References to the disks attached to the template.
`graphics_consoles`	GraphicsConsole[]	
`nics`	Nic[]	
`permissions`	Permission[]	
`quota`	Quota	
`storage_domain`	StorageDomain	
`tags`	Tag[]	
`watchdogs`	Watchdog[]	

6.121. Io struct

Table 934. Attributes summary

Name	Type	Summary
threads	Integer	

6.122. Ip struct

Represents the IP configuration of a network interface.

Table 935. Attributes summary

Name	Type	Summary
address	String	The text representation of the IP address.
gateway	String	The address of the default gateway.
netmask	String	The network mask.
version	IpVersion	The version of the IP protocol.

6.122.1. address

The text representation of the IP address.

For example, an IPv4 address will be represented as follows:

```
<ip>
  <address>192.168.0.1</address>
  ...
</ip>
```

An IPv6 address will be represented as follows:

```
<ip>
  <address>2620:52:0:20f0:4216:7eff:feaa:1b50</address>
  ...
</ip>
```

6.122.2. version

The version of the IP protocol.

This attribute is optional, and when it isn't given the value will be inferred from the value of the address attribute. If that value is a valid IPv4 address, then it will be assumed that the value of this attribute is v4. If it is a valid IPv6 address, then it will be assumed that the value of this attribute is v6.

NOTE This inference of the IP version from the value of the address is implemented since version 4.1 of the engine, before that this attribute was mandatory.

6.123. IpAddressAssignment struct

Table 936. Attributes summary

Name	Type	Summary
assignment_method	BootProtocol	
ip	Ip	

6.124. IpVersion enum

The enum defines valid values for the IP protocol version.

Table 937. Values summary

Name	Summary
v4	IPv4.
v6	IPv6.

6.125. IscsiBond struct

Table 938. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.

Table 939. Links summary

Name	Type	Summary
data_center	DataCenter	
networks	Network[]	
storage_connections	StorageConnection[]	

6.126. IscsiDetails struct

Table 940. Attributes summary

Name	Type	Summary
address	String	

Name	Type	Summary
disk_id	String	
initiator	String	
lun_mapping	Integer	
password	String	
paths	Integer	
port	Integer	
portal	String	
product_id	String	
serial	String	
size	Integer	
status	String	
storage_domain_id	String	
target	String	
username	String	
vendor_id	String	
volume_group_id	String	

6.127. Job struct

Table 941. Attributes summary

Name	Type	Summary
auto_cleared	Boolean	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
end_time	Date	
external	Boolean	
id	String	
last_updated	Date	
name	String	A human readable name in plain text.
start_time	Date	
status	JobStatus	

Table 942. Links summary

Name	Type	Summary
owner	User	
steps	Step[]	

6.128. JobStatus enum

Table 943. Values summary

Name	Summary
aborted	
failed	
finished	
started	
unknown	

6.129. KatelloErratum struct

Table 944. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
issued	Date	
name	String	A human readable name in plain text.
packages	Package[]	
severity	String	
solution	String	
summary	String	
title	String	
type	String	

Table 945. Links summary

Name	Type	Summary
host	Host	
vm	Vm	

6.130. KdumpStatus enum

Table 946. Values summary

Name	Summary
`disabled`	
`enabled`	
`unknown`	

6.131. Kernel struct

Table 947. Attributes summary

Name	Type	Summary
`version`	Version	

6.132. Ksm struct

Table 948. Attributes summary

Name	Type	Summary
`enabled`	Boolean	
`merge_across_nodes`	Boolean	

6.133. LogSeverity enum

Table 949. Values summary

Name	Summary
`alert`	
`error`	
`normal`	
`warning`	

6.134. LogicalUnit struct

Table 950. Attributes summary

Name	Type	Summary
`address`	String	
`disk_id`	String	
`id`	String	

Name	Type	Summary
lun_mapping	Integer	
password	String	
paths	Integer	
port	Integer	
portal	String	
product_id	String	
serial	String	
size	Integer	
status	LunStatus	
storage_domain_id	String	
target	String	
username	String	
vendor_id	String	
volume_group_id	String	

6.135. LunStatus enum

Table 951. Values summary

Name	Summary
free	
unusable	
used	

6.136. Mac struct

Table 952. Attributes summary

Name	Type	Summary
address	String	

6.137. MacPool struct

Represents a MAC address pool.

Example of a XML representatian of a MAC address pool:

```
<mac_pool href="/ovirt-engine/api/macpools/123" id="123">
  <name>Default</name>
  <description>Default MAC pool</description>
  <allow_duplicates>false</allow_duplicates>
  <default_pool>true</default_pool>
  <ranges>
    <range>
      <from>00:1A:4A:16:01:51</from>
      <to>00:1A:4A:16:01:E6</to>
    </range>
  </ranges>
</mac_pool>
```

Table 953. Attributes summary

Name	Type	Summary
`allow_duplicates`	Boolean	Defines whether duplicate MAC addresses are permitted in the pool.
`comment`	String	Free text containing comments about this object.
`default_pool`	Boolean	Defines whether this is the default pool.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.
`ranges`	Range[]	Defines the range of MAC addresses for the pool.

6.137.1. allow_duplicates

Defines whether duplicate MAC addresses are permitted in the pool. If not specified defaults to `false`.

6.137.2. default_pool

Defines whether this is the default pool. If not specified defaults to `false`.

6.137.3. ranges

Defines the range of MAC addresses for the pool. Multiple ranges can be defined.

6.138. MemoryOverCommit struct

Table 954. Attributes summary

Name	Type	Summary
`percent`	Integer	

6.139. MemoryPolicy struct

Table 955. Attributes summary

Name	Type	Summary
`ballooning`	Boolean	
`guaranteed`	Integer	
`over_commit`	MemoryOverCommit	
`transparent_huge_pages`	TransparentHugePages	

6.140. MessageBrokerType enum

Table 956. Values summary

Name	Summary
`qpid`	
`rabbit_mq`	

6.141. Method struct

Table 957. Attributes summary

Name	Type	Summary
`id`	SsoMethod	

6.142. MigrateOnError enum

Table 958. Values summary

Name	Summary
`do_not_migrate`	
`migrate`	
`migrate_highly_available`	

6.143. MigrationBandwidth struct

Defines the bandwidth used by migration.

Table 959. Attributes summary

Name	Type	Summary
assignment_method	MigrationBandwidthAssignmentMethod	The way how the bandwidth is assigned.
custom_value	Integer	Custom bandwidth in Mbit/s.

6.143.1. custom_value

Custom bandwidth in Mbit/s. Will be applied only if the assignmentMethod attribute is custom.

6.144. MigrationBandwidthAssignmentMethod enum

Defines the method how the migration bandwidth is assigned.

Table 960. Values summary

Name	Summary
auto	Takes the bandwidth from QoS if QoS defined.
custom	Custom defined bandwidth in Mbit/s.
hypervisor_default	Takes the value as configured on the hypervisor.

6.144.1. auto

Takes the bandwidth from QoS if QoS defined. If not, taken from detected link speed being used. If nothing detected, falls back to hypervisor_default value.

6.145. MigrationOptions struct

Table 961. Attributes summary

Name	Type	Summary
auto_converge	InheritableBoolean	
bandwidth	MigrationBandwidth	The bandwidth which is allowed to be used by the migrations.
compressed	InheritableBoolean	

Table 962. Links summary

Name	Type	Summary
policy	MigrationPolicy	Reference to the migration policy as defined using engine-config.

6.146. MigrationPolicy struct

A policy describing how the migration is going to be treated (convergence, how many parallel migrations allowed).

Table 963. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.

6.147. Network struct

Logical network.

An example of the JSON representation of a logical network:

```
{
  "network" : [ {
    "data_center" : {
      "href" : "/ovirt-engine/api/datacenters/123",
      "id" : "123"
    },
    "stp" : "false",
    "mtu" : "0",
    "usages" : {
      "usage" : [ "vm" ]
    },
    "name" : "ovirtmgmt",
    "description" : "Management Network",
    "href" : "/ovirt-engine/api/networks/456",
    "id" : "456",
    "link" : [ {
      "href" : "/ovirt-engine/api/networks/456/permissions",
      "rel" : "permissions"
    }, {
      "href" : "/ovirt-engine/api/networks/456/vnicprofiles",
      "rel" : "vnicprofiles"
    }, {
      "href" : "/ovirt-engine/api/networks/456/labels",
      "rel" : "labels"
    } ]
  } ]
}
```

An example of the XML representation of the same logical network:

```
<network href="/ovirt-engine/api/networks/456" id="456">
  <name>ovirtmgmt</name>
  <description>Management Network</description>
  <link href="/ovirt-engine/api/networks/456/permissions" rel="permissions"/>
  <link href="/ovirt-engine/api/networks/456/vnicprofiles" rel="vnicprofiles"/>
  <link href="/ovirt-engine/api/networks/456/labels" rel="labels"/>
  <data_center href="/ovirt-engine/api/datacenters/123" id="123"/>
  <stp>false</stp>
  <mtu>0</mtu>
  <usages>
    <usage>vm</usage>
  </usages>
</network>
```

Table 964. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`display`	Boolean	
`id`	String	
`ip`	Ip	
`mtu`	Integer	Specifies the maximum transmission unit for the network.
`name`	String	A human readable name in plain text.
`profile_required`	Boolean	
`required`	Boolean	
`status`	NetworkStatus	
`stp`	Boolean	Specifies whether spanning tree protocol is enabled for the network.
`usages`	NetworkUsage[]	Defines a set of usage elements for the network.
`vlan`	Vlan	

6.147.1. usages

Defines a set of usage elements for the network.

Users can, for example, specify that the network is to be used for virtual machine traffic and also for display traffic with the `vm` and `display` values.

Table 965. Links summary

Name	Type	Summary
cluster	Cluster	
data_center	DataCenter	A reference to the data center of which the network is a member.
network_labels	NetworkLabel[]	A reference to the labels assigned to the network.
permissions	Permission[]	A reference to the permissions of the network.
qos	Qos	
vnic_profiles	VnicProfile[]	A reference to the profiles of the network.

6.148. NetworkAttachment struct

Table 966. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
in_sync	Boolean	
ip_address_assignments	IpAddressAssignment[]	
name	String	A human readable name in plain text.
properties	Property[]	
reported_configurations	ReportedConfiguration[]	

Table 967. Links summary

Name	Type	Summary
host	Host	
host_nic	HostNic	
network	Network	
qos	Qos	

6.149. NetworkConfiguration struct

Table 968. Attributes summary

Name	Type	Summary
dns	Dns	

Name	Type	Summary
nics	Nic[]	

6.150. NetworkFilter struct

Network filter enables to filter packets send to/from the VM's nic according to defined rules.

There are several types of network filters supported based on libvirt. More details about the different network filters can be found here.

In addition to libvirt's network filters, there are two additional network filters: The first called vdsm-no-mac-spoofing, composed of no-mac-spoofing and no-arp-mac-spoofing. The second called ovirt-no-filter is used when no network filter is to be defined for the VM's nic. ovirt-no-filter network filter is only used for internal implementation, and doesn't exist on the nics.

This is a example of the XML representation:

```
<network_filter id="00000019-0019-0019-0019-00000000026c">
  <name>example-filter</name>
  <version>
    <major>4</major>
    <minor>0</minor>
    <build>-1</build>
    <revision>-1</revision>
  </version>
</network_filter>
```

If any part of the version is not present, it is represented by -1.

Table 969. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
version	Version	Represent the minimal supported version of the specific NetworkFilter for which it was first introduced.

6.151. NetworkLabel struct

Table 970. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.

Table 971. Links summary

Name	Type	Summary
host_nic	HostNic	
network	Network	

6.152. NetworkPluginType enum

Table 972. Values summary

Name	Summary
open_vswitch	

6.153. NetworkStatus enum

Table 973. Values summary

Name	Summary
non_operational	
operational	

6.154. NetworkUsage enum

Table 974. Values summary

Name	Summary
display	
gluster	The network will be used for Gluster(bricks) data traffic.
management	
migration	
vm	

6.155. NfsProfileDetail struct

Table 975. Attributes summary

Name	Type	Summary
nfs_server_ip	String	
profile_details	ProfileDetail[]	

6.156. NfsVersion enum

Table 976. Values summary

Name	Summary
auto	
v3	
v4	
v4_1	

6.157. Nic struct

Represents a NIC of a virtual machine.

For example, the XML representation of a NIC will look like this:

```
<nic href="/ovirt-engine/api/vms/123/nics/456" id="456">
  <name>nic1</name>
  <vm href="/ovirt-engine/api/vms/123" id="123"/>
  <interface>virtio</interface>
  <linked>true</linked>
  <mac>
    <address>02:00:00:00:00:00</address>
  </mac>
  <plugged>true</plugged>
  <vnic_profile href="/ovirt-engine/api/vnicprofiles/789" id="789"/>
</nic>
```

Table 977. Attributes summary

Name	Type	Summary
boot_protocol	BootProtocol	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	

Name	Type	Summary
`interface`	NicInterface	The type of driver used for the NIC.
`linked`	Boolean	Defines if the NIC is linked to the virtual machine.
`mac`	Mac	The MAC address of the interface.
`name`	String	A human readable name in plain text.
`on_boot`	Boolean	
`plugged`	Boolean	Defines if the NIC is plugged in to the virtual machine.

Table 978. Links summary

Name	Type	Summary
`instance_type`	InstanceType	
`network`	Network	A reference to the network which the interface should be connected to.
`network_attachments`	NetworkAttachment[]	
`network_labels`	NetworkLabel[]	
`reported_devices`	ReportedDevice[]	
`statistics`	Statistic[]	A link to the statistics for the NIC.
`template`	Template	
`virtual_function_allowed_labels`	NetworkLabel[]	
`virtual_function_allowed_networks`	Network[]	
`vm`	Vm	Don't use this element, use `vms` instead.
`vms`	Vm[]	References to the virtual machines that are using this device.
`vnic_profile`	VnicProfile	

6.157.1. network

A reference to the network which the interface should be connected to. A blank network id is allowed.

Usage of this element for creating or updating a NIC is deprecated, use `vnic_profile` instead. It is preserved because it is still in use by the `initialization` element, as a holder for IP addresses and other network details.

6.157.2. vms

References to the virtual machines that are using this device. A device may be used by several

virtual machines; for example, a shared disk my be used simultaneously by two or more virtual machines.

6.158. NicConfiguration struct

Table 979. Attributes summary

Name	Type	Summary
`boot_protocol`	BootProtocol	
`ip`	Ip	
`name`	String	
`on_boot`	Boolean	

6.159. NicInterface enum

Table 980. Values summary

Name	Summary
`e1000`	
`pci_passthrough`	
`rtl8139`	
`rtl8139_virtio`	
`spapr_vlan`	
`virtio`	

6.160. NicStatus enum

Table 981. Values summary

Name	Summary
`down`	
`up`	

6.161. NumaNode struct

Represents a physical NUMA node.

Example XML representation:

```
<host_numa_node href="/ovirt-engine/api/hosts/0923f1ea/numanodes/007cf1ab"
id="007cf1ab">
  <cpu>
    <cores>
      <core>
        <index>0</index>
      </core>
    </cores>
  </cpu>
  <index>0</index>
  <memory>65536</memory>
  <node_distance>40 20 40 10</node_distance>
  <host href="/ovirt-engine/api/hosts/0923f1ea" id="0923f1ea"/>
</host_numa_node>
```

Table 982. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
cpu	Cpu	
description	String	A human readable description in plain text.
id	String	
index	Integer	
memory	Integer	Memory of the NUMA node in MB.
name	String	A human readable name in plain text.
node_distance	String	

Table 983. Links summary

Name	Type	Summary
host	Host	
statistics	Statistic[]	

6.162. NumaNodePin struct

Represents pinning of a virtual NUMA node to a physical NUMA node.

Table 984. Attributes summary

Name	Type	Summary
host_numa_node	NumaNode	Deprecated - has no function.
index	Integer	Index of a physical NUMA node to which the virtual NUMA node is pinned.

Name	Type	Summary
pinned	Boolean	Deprecated - should always be true.

6.163. NumaTuneMode enum

Table 985. Values summary

Name	Summary
interleave	
preferred	
strict	

6.164. OpenStackImage struct

Table 986. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.

Table 987. Links summary

Name	Type	Summary
openstack_image_provider	OpenStackImageProvider	

6.165. OpenStackImageProvider struct

Table 988. Attributes summary

Name	Type	Summary
authentication_url	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
password	String	
properties	Property[]	

Name	Type	Summary
requires_authentication	Boolean	
tenant_name	String	
url	String	
username	String	

Table 989. Links summary

Name	Type	Summary
certificates	Certificate[]	
images	OpenStackImage[]	

6.166. OpenStackNetwork struct

Table 990. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.

Table 991. Links summary

Name	Type	Summary
openstack_network_provider	OpenStackNetworkProvider	

6.167. OpenStackNetworkProvider struct

Table 992. Attributes summary

Name	Type	Summary
agent_configuration	AgentConfiguration	
authentication_url	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.

Name	Type	Summary
`password`	String	
`plugin_type`	NetworkPluginType	
`properties`	Property[]	
`read_only`	Boolean	Indicates whether the provider is read only.
`requires_authentication`	Boolean	
`tenant_name`	String	
`type`	OpenStackNetworkProviderType	The type of provider.
`url`	String	
`username`	String	

6.167.1. read_only

Indicates whether the provider is read only. A read-only provider does not allow adding, modifying or deleting of networks or subnets. Port-related operations are allowed, as they are required for the provisioning of virtual nics.

Table 993. Links summary

Name	Type	Summary
`certificates`	Certificate[]	
`networks`	OpenStackNetwork[]	
`subnets`	OpenStackSubnet[]	

6.168. OpenStackNetworkProviderType enum

The OpenStack network provider can either be implemented by OpenStack Neutron, in which case the Neutron agent is automatically installed on the hosts, or it can be an external provider implementing the OpenStack API, in which case the virtual interface driver will be a custom solution installed manually.

Table 994. Values summary

Name	Summary
`external`	Indicates that the provider is an external one, implementing the OpenStack Neutron API.
`neutron`	Indicates that the provider is OpenStack Neutron.

6.168.1. external

Indicates that the provider is an external one, implementing the OpenStack Neutron API. The virtual interface driver in this case is implemented by the external provider.

6.168.2. neutron

Indicates that the provider is OpenStack Neutron. The standard OpenStack Neutron agent will be used as the virtual interface driver.

6.169. OpenStackProvider struct

Table 995. Attributes summary

Name	Type	Summary
`authentication_url`	String	
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.
`password`	String	
`properties`	Property[]	
`requires_authentication`	Boolean	
`tenant_name`	String	
`url`	String	
`username`	String	

6.170. OpenStackSubnet struct

Table 996. Attributes summary

Name	Type	Summary
`cidr`	String	
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`dns_servers`	String[]	
`gateway`	String	
`id`	String	
`ip_version`	String	

Name	Type	Summary
name	String	A human readable name in plain text.

Table 997. Links summary

Name	Type	Summary
openstack_network	OpenStackNetwork	

6.171. OpenStackVolumeProvider struct

Table 998. Attributes summary

Name	Type	Summary
authentication_url	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
password	String	
properties	Property[]	
requires_authentication	Boolean	
tenant_name	String	
url	String	
username	String	

Table 999. Links summary

Name	Type	Summary
authentication_keys	OpenstackVolumeAuthenticationKey[]	
certificates	Certificate[]	
data_center	DataCenter	
volume_types	OpenStackVolumeType[]	

6.172. OpenStackVolumeType struct

Table 1000. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.
`properties`	Property[]	

Table 1001. Links summary

Name	Type	Summary
`openstack_volume_provider`	OpenStackVolumeProvider	

6.173. OpenstackVolumeAuthenticationKey struct

Table 1002. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`creation_date`	Date	
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.
`usage_type`	OpenstackVolumeAuthenticationKeyUsageType	
`uuid`	String	
`value`	String	

Table 1003. Links summary

Name	Type	Summary
`openstack_volume_provider`	OpenStackVolumeProvider	

6.174. OpenstackVolumeAuthenticationKeyUsageType enum

Table 1004. Values summary

Name	Summary
`ceph`	

6.175. OperatingSystem struct

Information describing the operating system. Used for virtual machines and hosts.

Table 1005. Attributes summary

Name	Type	Summary
`boot`	Boot	
`cmdline`	String	
`custom_kernel_cmdline`	String	A custom part of the host kernel command line.
`initrd`	String	
`kernel`	String	
`reported_kernel_cmdline`	String	Host kernel command line as reported by a running host.
`type`	String	
`version`	Version	

6.175.1. custom_kernel_cmdline

A custom part of the host kernel command line. This will be merged with the existing kernel command line.

You must re-install and then reboot the host to apply the changes implemented by this attribute.

Parameters merging: During each host deploy procedure, kernel parameters that were added in the previous host deploy procedure are removed using `grubby --update-kernel DEFAULT --remove-args <previous_custom_params>` and the current kernel command line customization is applied using `grubby --update-kernel DEFAULT --args <custom_params>`. The Engine internally keeps track of the last applied kernel parameters customization.

NOTE This attribute is currently only used for hosts.

6.175.2. reported_kernel_cmdline

Host kernel command line as reported by a running host.

Read-only attribute. Attempts to change this attribute are silently ignored.

NOTE	This attribute is currently only used for hosts.

6.176. OperatingSystemInfo struct

Table 1006. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`large_icon`	Icon	
`name`	String	A human readable name in plain text.
`small_icon`	Icon	

6.177. Option struct

Table 1007. Attributes summary

Name	Type	Summary
`name`	String	
`type`	String	
`value`	String	

6.178. OsType enum

Table 1008. Values summary

Name	Summary
`other`	
`other_linux`	
`rhel_3`	
`rhel_3x64`	
`rhel_4`	
`rhel_4x64`	
`rhel_5`	
`rhel_5x64`	
`rhel_6`	
`rhel_6x64`	

Name	Summary
unassigned	
windows_2003	
windows_2003x64	
windows_2008	
windows_2008r2x64	
windows_2008x64	
windows_2012x64	
windows_7	
windows_7x64	
windows_8	
windows_8x64	
windows_xp	

6.179. Package struct

Table 1009. Attributes summary

Name	Type	Summary
name	String	

6.180. Payload struct

Table 1010. Attributes summary

Name	Type	Summary
files	File[]	
type	VmDeviceType	
volume_id	String	

6.181. PayloadEncoding enum

Table 1011. Values summary

Name	Summary
base64	
plaintext	

6.182. Permission struct

Table 1012. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 1013. Links summary

Name	Type	Summary
`cluster`	Cluster	
`data_center`	DataCenter	
`disk`	Disk	
`group`	Group	
`host`	Host	
`role`	Role	
`storage_domain`	StorageDomain	
`template`	Template	
`user`	User	
`vm`	Vm	
`vm_pool`	VmPool	

6.183. Permit struct

Table 1014. Attributes summary

Name	Type	Summary
`administrative`	Boolean	
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 1015. Links summary

Name	Type	Summary
role	Role	

6.184. PmProxy struct

Table 1016. Attributes summary

Name	Type	Summary
type	PmProxyType	

6.185. PmProxyType enum

Table 1017. Values summary

Name	Summary
cluster	Fence proxy is selected from the same cluster as fenced host.
dc	Fence proxy is selected from the same data center as fenced host.
other_dc	Fence proxy is selected from a different data center than fenced host.

6.186. PolicyUnitType enum

This enum holds the types of all internal policy units types

Table 1018. Values summary

Name	Summary
filter	
load_balancing	
weight	

6.187. PortMirroring struct

6.188. PowerManagement struct

Table 1019. Attributes summary

Name	Type	Summary
address	String	The host name or IP address of the host.
agents	Agent[]	Specifies fence agent options when multiple fences are used.
automatic_pm_enabled	Boolean	Toggles the automated power control of the host in order to save energy.

Name	Type	Summary
`enabled`	Boolean	Indicates whether power management configuration is enabled or disabled.
`kdump_detection`	Boolean	Toggles whether to determine if kdump is running on the host before it is shut down.
`options`	Option[]	Fencing options for the selected type= specified with the option name="" and value="" strings.
`password`	String	A valid, robust password for power management.
`pm_proxies`	PmProxy[]	Determines the power management proxy.
`status`	PowerManagementStatus	Determines the power status of the host.
`type`	String	Fencing device code.
`username`	String	A valid user name for power management.

6.188.1. agents

Specifies fence agent options when multiple fences are used.

Use the order sub-element to prioritize the fence agents. Agents are run sequentially according to their order until the fence action succeeds. When two or more fence agents have the same order, they are run concurrently. Other sub-elements include type, ip, user, password, and options.

6.188.2. automatic_pm_enabled

Toggles the automated power control of the host in order to save energy. When set to true, the host will be automatically powered down if the cluster's load is low, and powered on again when required. This is set to true when a host is created, unless disabled by the user.

6.188.3. kdump_detection

Toggles whether to determine if kdump is running on the host before it is shut down. When set to `true`, the host will not shut down during a kdump process. This is set to `true` when a host has power management enabled, unless disabled by the user.

6.188.4. type

Fencing device code.

A list of valid fencing device codes are available in the `capabilities` collection.

6.189. PowerManagementStatus enum

Table 1020. Values summary

Name	Summary
`off`	Host is OFF.
`on`	Host is ON.
`unknown`	Unknown status.

6.190. Product struct

Table 1021. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

6.191. ProductInfo struct

This type contains information about the product, including its name, the name of the vendor, and the version.

Table 1022. Attributes summary

Name	Type	Summary
`name`	String	The name of the product, for example `oVirt Engine`.
`vendor`	String	The name of the vendor, for example `ovirt.
`version`	Version	The version number of the product.

6.191.1. vendor

The name of the vendor, for example `ovirt.org`.

6.192. ProfileDetail struct

Table 1023. Attributes summary

Name	Type	Summary
`block_statistics`	BlockStatistic[]	
`duration`	Integer	
`fop_statistics`	FopStatistic[]	
`profile_type`	String	

Name	Type	Summary
statistics	Statistic[]	

6.193. Property struct

Table 1024. Attributes summary

Name	Type	Summary
name	String	
value	String	

6.194. ProxyTicket struct

Table 1025. Attributes summary

Name	Type	Summary
value	String	

6.195. Qos struct

Table 1026. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
cpu_limit	Integer	
description	String	A human readable description in plain text.
id	String	
inbound_average	Integer	
inbound_burst	Integer	
inbound_peak	Integer	
max_iops	Integer	
max_read_iops	Integer	
max_read_throughput	Integer	
max_throughput	Integer	
max_write_iops	Integer	
max_write_throughput	Integer	
name	String	A human readable name in plain text.

Name	Type	Summary
outbound_average	Integer	
outbound_average_linkshare	Integer	
outbound_average_realtime	Integer	
outbound_average_upperlimit	Integer	
outbound_burst	Integer	
outbound_peak	Integer	
type	QosType	

Table 1027. Links summary

Name	Type	Summary
data_center	DataCenter	

6.196. QosType enum

Table 1028. Values summary

Name	Summary
cpu	
hostnetwork	
network	
storage	

6.197. Quota struct

Represents a quota object.

An example XML representation of a quota:

```
<quota href="/ovirt-engine/api/datacenters/7044934e/quotas/dcad5ddc" id="dcad5ddc">
  <name>My Quota</name>
  <description>A quota for my oVirt environment</description>
  <cluster_hard_limit_pct>0</cluster_hard_limit_pct>
  <cluster_soft_limit_pct>0</cluster_soft_limit_pct>
  <data_center href="/ovirt-engine/api/datacenters/7044934e" id="7044934e"/>
  <storage_hard_limit_pct>0</storage_hard_limit_pct>
  <storage_soft_limit_pct>0</storage_soft_limit_pct>
</quota>
```

Table 1029. Attributes summary

Name	Type	Summary
cluster_hard_limit_pct	Integer	
cluster_soft_limit_pct	Integer	
comment	String	Free text containing comments about this object.
data_center	DataCenter	
description	String	A human readable description in plain text.
disks	Disk[]	
id	String	
name	String	A human readable name in plain text.
storage_hard_limit_pct	Integer	
storage_soft_limit_pct	Integer	
users	User[]	
vms	Vm[]	

Table 1030. Links summary

Name	Type	Summary
permissions	Permission[]	
quota_cluster_limits	QuotaClusterLimit[]	
quota_storage_limits	QuotaStorageLimit[]	

6.198. QuotaClusterLimit struct

Table 1031. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
memory_limit	Decimal	
memory_usage	Decimal	
name	String	A human readable name in plain text.
vcpu_limit	Integer	

Name	Type	Summary
vcpu_usage	Integer	

Table 1032. Links summary

Name	Type	Summary
cluster	Cluster	
quota	Quota	

6.199. QuotaModeType enum

Table 1033. Values summary

Name	Summary
audit	
disabled	
enabled	

6.200. QuotaStorageLimit struct

Table 1034. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
limit	Integer	
name	String	A human readable name in plain text.
usage	Decimal	

Table 1035. Links summary

Name	Type	Summary
quota	Quota	
storage_domain	StorageDomain	

6.201. Range struct

Table 1036. Attributes summary

Name	Type	Summary
`from`	String	
`to`	String	

6.202. Rate struct

Table 1037. Attributes summary

Name	Type	Summary
`bytes`	Integer	
`period`	Integer	

6.203. ReportedConfiguration struct

Table 1038. Attributes summary

Name	Type	Summary
`actual_value`	String	
`expected_value`	String	
`in_sync`	Boolean	
`name`	String	

6.204. ReportedDevice struct

Table 1039. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`ips`	Ip[]	
`mac`	Mac	
`name`	String	A human readable name in plain text.
`type`	ReportedDeviceType	

Table 1040. Links summary

Name	Type	Summary
`vm`	Vm	

6.205. ReportedDeviceType enum

Table 1041. Values summary

Name	Summary
network	

6.206. ResolutionType enum

Table 1042. Values summary

Name	Summary
add	
copy	

6.207. RngDevice struct

Table 1043. Attributes summary

Name	Type	Summary
rate	Rate	
source	RngSource	

6.208. RngSource enum

Table 1044. Values summary

Name	Summary
hwrng	
random	

6.209. Role struct

Table 1045. Attributes summary

Name	Type	Summary
administrative	Boolean	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
mutable	Boolean	
name	String	A human readable name in plain text.

Table 1046. Links summary

Name	Type	Summary
`permits`	Permit[]	
`user`	User	

6.210. RoleType enum

Table 1047. Values summary

Name	Summary
`admin`	
`user`	

6.211. SchedulingPolicy struct

Table 1048. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`default_policy`	Boolean	
`description`	String	A human readable description in plain text.
`id`	String	
`locked`	Boolean	
`name`	String	A human readable name in plain text.
`properties`	Property[]	

Table 1049. Links summary

Name	Type	Summary
`balances`	Balance[]	
`filters`	Filter[]	
`weight`	Weight[]	

6.212. SchedulingPolicyUnit struct

Table 1050. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.

Name	Type	Summary
enabled	Boolean	
id	String	
internal	Boolean	
name	String	A human readable name in plain text.
properties	Property[]	
type	PolicyUnitType	

6.213. ScsiGenericIO enum

Table 1051. Values summary

Name	Summary
filtered	
unfiltered	

6.214. SeLinux struct

Table 1052. Attributes summary

Name	Type	Summary
mode	SeLinuxMode	

6.215. SeLinuxMode enum

Table 1053. Values summary

Name	Summary
disabled	
enforcing	
permissive	

6.216. SerialNumber struct

Table 1054. Attributes summary

Name	Type	Summary
policy	SerialNumberPolicy	
value	String	

6.217. SerialNumberPolicy enum

Table 1055. Values summary

Name	Summary
`custom`	
`host`	
`vm`	

6.218. Session struct

Describes user session to a virtual machine.

Table 1056. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`console_user`	Boolean	Indicates if this is a console session.
`description`	String	A human readable description in plain text.
`id`	String	
`ip`	Ip	IP address user is connected from.
`name`	String	A human readable name in plain text.
`protocol`	String	Protocol used by the session.

6.218.1. console_user

Indicates if this is a console session.

The value will be `true` for console users: SPICE or VNC, `false` for others: e.g. RDP, SSH.

6.218.2. ip

IP address user is connected from.

Currently only available for console users.

6.218.3. protocol

Protocol used by the session.

Currently not used, intended for info about how is user connected: SPICE, VNC, SSH, RDP.

Table 1057. Links summary

Name	Type	Summary
user	User	User related to this session.
vm	Vm	Link to virtual machine related to this session.

6.218.4. user

User related to this session.

If user is a console user, it is a link to real oVirt user. Otherwise only username is provided.

6.219. SkipIfConnectivityBroken struct

Table 1058. Attributes summary

Name	Type	Summary
enabled	Boolean	
threshold	Integer	

6.220. SkipIfSdActive struct

Table 1059. Attributes summary

Name	Type	Summary
enabled	Boolean	

6.221. Snapshot struct

Represents a snapshot object.

```
<snapshot id="456" href="/ovirt-engine/api/vms/123/snapshots/456">
  <actions>
    <link rel="restore" href="/ovirt-engine/api/vms/123/snapshots/456/restore"/>
  </actions>
  <vm id="123" href="/ovirt-engine/api/vms/123"/>
  <description>Virtual Machine 1 - Snapshot A</description>
  <type>active</type>
  <date>2010-08-16T14:24:29</date>
  <persist_memorystate>false</persist_memorystate>
</snapshot>
```

Table 1060. Attributes summary

Name	Type	Summary
bios	Bios	

Name	Type	Summary
comment	String	Free text containing comments about this object.
console	Console	
cpu	Cpu	The configuration of the CPU of the virtual machine.
cpu_shares	Integer	
creation_time	Date	
custom_compatibility_version	Version	Virtual machine custom compatibility version.
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	
date	Date	
delete_protected	Boolean	
description	String	A human readable description in plain text.
display	Display	
domain	Domain	
fqdn	String	
guest_operating_system	GuestOperatingSystem	
guest_time_zone	TimeZone	
high_availability	HighAvailability	
id	String	
initialization	Initialization	
io	Io	
large_icon	Icon	
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	
migration	MigrationOptions	
migration_downtime	Integer	
name	String	A human readable name in plain text.
next_run_configuration_exists	Boolean	
numa_tune_mode	NumaTuneMode	

Name	Type	Summary
origin	String	
os	OperatingSystem	
payloads	Payload[]	
persist_memorystate	Boolean	
placement_policy	VmPlacementPolicy	The configuration of the virtual machine's placement policy.
rng_device	RngDevice	
run_once	Boolean	
serial_number	SerialNumber	
small_icon	Icon	
snapshot_status	SnapshotStatus	
snapshot_type	SnapshotType	
soundcard_enabled	Boolean	
sso	Sso	
start_paused	Boolean	
start_time	Date	
stateless	Boolean	
status	VmStatus	
status_detail	String	
stop_reason	String	
stop_time	Date	
time_zone	TimeZone	
tunnel_migration	Boolean	
type	VmType	
usb	Usb	
use_latest_template_version	Boolean	
virtio_scsi	VirtioScsi	

6.221.1. cpu

The configuration of the CPU of the virtual machine.

The sockets configuration can be updated without rebooting the virtual machine. The cores and the

threads require a reboot in order to take place.

For example, to change the number of sockets to 4 immediately and the number of cores and threads to 2 after reboot send a request:

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

With a request body:

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

6.221.2. custom_compatibility_version

Virtual machine custom compatibility version.

This field allows to customize a virtual machine to its own compatibility version. If `custom_compatibility_version` is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in and is checked against capabilities of the host the virtual machine is planned to run on.

6.221.3. memory

The virtual machine's memory, in bytes.

For example, in order to update a virtual machine to contain 1 GiB of memory send a request like this:

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

With a request body like this:

```
<vm>
  <memory>1073741824</memory>
</vm>
```

NOTE	Memory in the previous example is converted to bytes using the following formula: 1 GiB = 2^{30} bytes = 1073741824 bytes.

NOTE	Memory hot plug is supported from oVirt 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

6.221.4. placement_policy

The configuration of the virtual machine's placement policy.

This configuration can be updated to pin a virtual machine to one or more hosts.

NOTE	Virtual machines that are pinned to multiple hosts cannot be live migrated, but in the event of a host failure, any virtual machine configured to be highly available is automatically restarted on one of the other hosts to which the virtual machine is pinned.

For example, to pin a virtual machine to two hosts, you would send a request like the following:

```
PUT /api/vms/123
```

With a request body like this:

```
<vm>
  <high_availability>
    <enabled>true</enabled>
    <priority>1</priority>
  </high_availability>
  <placement_policy>
    <hosts>
      <host>
        <name>Host1</name>
      </host>
      <host>
        <name>Host2</name>
      </host>
    </hosts>
    <affinity>pinned</affinity>
  </placement_policy>
</vm>
```

Table 1061. Links summary

Name	Type	Summary
`affinity_labels`	AffinityLabel[]	
`applications`	Application[]	

Name	Type	Summary
cdroms	Cdrom[]	
cluster	Cluster	
cpu_profile	CpuProfile	
disk_attachments	DiskAttachment[]	References to the disks attached to the virtual machine.
external_host_provider	ExternalHostProvider	
floppies	Floppy[]	
graphics_consoles	GraphicsConsole[]	
host	Host	
host_devices	HostDevice[]	
instance_type	InstanceType	
katello_errata	KatelloErratum[]	
nics	Nic[]	Link to the the list of network interface devices on the virtual machine.
numa_nodes	NumaNode[]	
permissions	Permission[]	
quota	Quota	
reported_devices	ReportedDevice[]	
sessions	Session[]	
snapshots	Snapshot[]	
statistics	Statistic[]	
storage_domain	StorageDomain	
tags	Tag[]	
template	Template	
vm	Vm	
vm_pool	VmPool	
watchdogs	Watchdog[]	

6.222. SnapshotStatus enum

Table 1062. Values summary

Name	Summary
in_preview	

Name	Summary
`locked`	
`ok`	

6.223. SnapshotType enum

Table 1063. Values summary

Name	Summary
`active`	
`preview`	
`regular`	
`stateless`	

6.224. SpecialObjects struct

This type contains references to special objects, like the blank template and the root of the hierarchy of tags.

Table 1064. Links summary

Name	Type	Summary
`blank_template`	Template	Reference to the blank template.
`root_tag`	Tag	Reference to the root of the hierarchy of tags.

6.225. Spm struct

Table 1065. Attributes summary

Name	Type	Summary
`priority`	Integer	
`status`	SpmStatus	

6.226. SpmStatus enum

Table 1066. Values summary

Name	Summary
`contending`	
`none`	
`spm`	

6.227. Ssh struct

Table 1067. Attributes summary

Name	Type	Summary
`authentication_method`	SshAuthenticationMethod	
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`fingerprint`	String	
`id`	String	
`name`	String	A human readable name in plain text.
`port`	Integer	
`user`	User	

6.228. SshAuthenticationMethod enum

Table 1068. Values summary

Name	Summary
`password`	
`publickey`	

6.229. SshPublicKey struct

Table 1069. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`content`	String	
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 1070. Links summary

Name	Type	Summary
`user`	User	

6.230. Sso struct

Table 1071. Attributes summary

Name	Type	Summary
methods	Method[]	

6.231. SsoMethod enum

Table 1072. Values summary

Name	Summary
guest_agent	

6.232. Statistic struct

Table 1073. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
kind	StatisticKind	
name	String	A human readable name in plain text.
type	ValueType	
unit	StatisticUnit	
values	Value[]	

Table 1074. Links summary

Name	Type	Summary
brick	GlusterBrick	
disk	Disk	
gluster_volume	GlusterVolume	
host	Host	
host_nic	HostNic	
host_numa_node	NumaNode	
nic	Nic	
step	Step	

Name	Type	Summary
vm	Vm	

6.233. StatisticKind enum

Table 1075. Values summary

Name	Summary
counter	
gauge	

6.234. StatisticUnit enum

Table 1076. Values summary

Name	Summary
bits_per_second	
bytes	
bytes_per_second	
count_per_second	
none	
percent	
seconds	

6.235. Step struct

Table 1077. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
end_time	Date	
external	Boolean	
external_type	ExternalSystemType	
id	String	
name	String	A human readable name in plain text.
number	Integer	
start_time	Date	

Name	Type	Summary
status	StepStatus	
type	StepEnum	

Table 1078. Links summary

Name	Type	Summary
job	Job	
parent_step	Step	
statistics	Statistic[]	

6.236. StepEnum enum

Table 1079. Values summary

Name	Summary
executing	
finalizing	
rebalancing_volume	
removing_bricks	
unknown	
validating	

6.237. StepStatus enum

Table 1080. Values summary

Name	Summary
aborted	
failed	
finished	
started	
unknown	

6.238. StorageConnection struct

Represents a storage server connection.

Example:

```
<storage_connection id="123">
  <address>mynfs.example.com</address>
  <type>nfs</type>
  <path>/exports/mydata</path>
</storage_connection>
```

Table 1081. Attributes summary

Name	Type	Summary
address	String	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	
mount_options	String	
name	String	A human readable name in plain text.
nfs_retrans	Integer	
nfs_timeo	Integer	
nfs_version	NfsVersion	
password	String	
path	String	
port	Integer	
portal	String	
target	String	
type	StorageType	
username	String	
vfs_type	String	

Table 1082. Links summary

Name	Type	Summary
host	Host	

6.239. StorageConnectionExtension struct

Table 1083. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.

Name	Type	Summary
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
password	String	
target	String	
username	String	

Table 1084. Links summary

Name	Type	Summary
host	Host	

6.240. StorageDomain struct

Table 1085. Attributes summary

Name	Type	Summary
available	Integer	
comment	String	Free text containing comments about this object.
committed	Integer	
critical_space_action_blocker	Integer	
description	String	A human readable description in plain text.
external_status	ExternalStatus	
id	String	
import	Boolean	
master	Boolean	
name	String	A human readable name in plain text.
status	StorageDomainStatus	
storage	HostStorage	
storage_format	StorageFormat	
type	StorageDomainType	
used	Integer	
warning_low_space_indicator	Integer	

Name	Type	Summary
wipe_after_delete	Boolean	

Table 1086. Links summary

Name	Type	Summary
data_center	DataCenter	This is used to link to the data center that the storage domain is attached to.
data_centers	DataCenter[]	This is a set of links to the data centers that the storage domain is attached to.
disk_profiles	DiskProfile[]	
disk_snapshots	DiskSnapshot[]	
disks	Disk[]	
files	File[]	
host	Host	Host is only relevant at creation time.
images	Image[]	
permissions	Permission[]	
storage_connections	StorageConnection[]	
templates	Template[]	
vms	Vm[]	

6.240.1. data_center

This is used to link to the data center that the storage domain is attached to. It is preserved for backwards compatibility, as the storage domain may be attached to multiple data centers (if it is an ISO domain). Use the `dataCenters` element instead.

6.241. StorageDomainStatus enum

Table 1087. Values summary

Name	Summary
activating	
active	
detaching	
inactive	
locked	
maintenance	
mixed	

Name	Summary
preparing_for_maintenance	
unattached	
unknown	

6.242. StorageDomainType enum

Table 1088. Values summary

Name	Summary
data	
export	
image	
iso	
volume	

6.243. StorageFormat enum

Table 1089. Values summary

Name	Summary
v1	
v2	
v3	

6.244. StorageType enum

Table 1090. Values summary

Name	Summary
cinder	
fcp	
glance	
glusterfs	
iscsi	
localfs	
nfs	
posixfs	

6.245. SwitchType enum

Enumerated type describing all switch types currently supported by engine.

Table 1091. Values summary

Name	Summary
`legacy`	Type representing native switch.
`ovs`	Type representing Open vSwitch.

6.246. Tag struct

Table 1092. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 1093. Links summary

Name	Type	Summary
`group`	Group	
`host`	Host	
`parent`	Tag	
`template`	Template	
`user`	User	
`vm`	Vm	

6.247. Template struct

Table 1094. Attributes summary

Name	Type	Summary
`bios`	Bios	
`comment`	String	Free text containing comments about this object.
`console`	Console	
`cpu`	Cpu	The configuration of the CPU of the virtual machine.
`cpu_shares`	Integer	

Name	Type	Summary
creation_time	Date	
custom_compatibility_version	Version	Virtual machine custom compatibility version.
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	
delete_protected	Boolean	
description	String	A human readable description in plain text.
display	Display	
domain	Domain	
high_availability	HighAvailability	
id	String	
initialization	Initialization	
io	Io	
large_icon	Icon	
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	
migration	MigrationOptions	
migration_downtime	Integer	
name	String	A human readable name in plain text.
origin	String	
os	OperatingSystem	
rng_device	RngDevice	
serial_number	SerialNumber	
small_icon	Icon	
soundcard_enabled	Boolean	
sso	Sso	
start_paused	Boolean	
stateless	Boolean	
status	TemplateStatus	
time_zone	TimeZone	

Name	Type	Summary
tunnel_migration	Boolean	
type	VmType	
usb	Usb	
version	TemplateVersion	
virtio_scsi	VirtioScsi	
vm	Vm	

6.247.1. cpu

The configuration of the CPU of the virtual machine.

The sockets configuration can be updated without rebooting the virtual machine. The cores and the threads require a reboot in order to take place.

For example, to change the number of sockets to 4 immediately and the number of cores and threads to 2 after reboot send a request:

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

With a request body:

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

6.247.2. custom_compatibility_version

Virtual machine custom compatibility version.

This field allows to customize a virtual machine to its own compatibility version. If `custom_compatibility_version` is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in and is checked against capabilities of the host the virtual machine is planned to run on.

6.247.3. memory

The virtual machine's memory, in bytes.

For example, in order to update a virtual machine to contain 1 GiB of memory send a request like this:

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

With a request body like this:

```
<vm>
  <memory>1073741824</memory>
</vm>
```

NOTE Memory in the previous example is converted to bytes using the following formula: 1 GiB = 2^{30} bytes = 1073741824 bytes.

NOTE Memory hot plug is supported from oVirt 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

Table 1095. Links summary

Name	Type	Summary
`cdroms`	Cdrom[]	
`cluster`	Cluster	
`cpu_profile`	CpuProfile	
`disk_attachments`	DiskAttachment[]	References to the disks attached to the template.
`graphics_consoles`	GraphicsConsole[]	
`nics`	Nic[]	
`permissions`	Permission[]	
`quota`	Quota	
`storage_domain`	StorageDomain	
`tags`	Tag[]	
`watchdogs`	Watchdog[]	

6.248. TemplateStatus enum

Table 1096. Values summary

Name	Summary
`illegal`	
`locked`	
`ok`	

6.249. TemplateVersion struct

Table 1097. Attributes summary

Name	Type	Summary
`version_name`	String	
`version_number`	Integer	

Table 1098. Links summary

Name	Type	Summary
`base_template`	Template	

6.250. Ticket struct

Table 1099. Attributes summary

Name	Type	Summary
`expiry`	Integer	
`value`	String	

6.251. TimeZone struct

Table 1100. Attributes summary

Name	Type	Summary
`name`	String	
`utc_offset`	String	

6.252. TransparentHugePages struct

Table 1101. Attributes summary

Name	Type	Summary
`enabled`	Boolean	

6.253. TransportType enum

Table 1102. Values summary

Name	Summary
`rdma`	
`tcp`	

6.254. UnmanagedNetwork struct

Table 1103. Attributes summary

Name	Type	Summary
`comment`	String	Free text containing comments about this object.
`description`	String	A human readable description in plain text.
`id`	String	
`name`	String	A human readable name in plain text.

Table 1104. Links summary

Name	Type	Summary
`host`	Host	
`host_nic`	HostNic	

6.255. Usb struct

Table 1105. Attributes summary

Name	Type	Summary
`enabled`	Boolean	
`type`	UsbType	

6.256. UsbType enum

Table 1106. Values summary

Name	Summary
`legacy`	
`native`	

6.257. User struct

Table 1107. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
department	String	
description	String	A human readable description in plain text.
domain_entry_id	String	
email	String	
id	String	
last_name	String	
logged_in	Boolean	
name	String	A human readable name in plain text.
namespace	String	Name space where user resides.
password	String	
principal	String	Same as user_name principal has different formats based on LDAP provider.
user_name	String	Username of the user.

6.257.1. namespace

Name space where user resides. When using the authorization provider that stores users in the LDAP (see here for details) this attribute equals to naming context of the LDAP. When using the built-in authorization provider that stores users in the database (see here for details) this attribute is ignored.

6.257.2. principal

Same as user_name principal has different formats based on LDAP provider. In case of most LDAP providers it is value of the uid LDAP attribute. In case of Active Directory it is the user principal name (UPN).

6.257.3. user_name

Username of the user. The format depends on authorization provider type. In case of most LDAP providers it is value of the uid LDAP attribute. In case of Active Directory it is the user principal name (UPN). UPN or uid must be followed by authorization provider name. For example in case of LDAP using uid attribute it is: myuser@myextension-authz. In case of Active Directory using UPN it is: myuser@mysubdomain.mydomain.com@myextension-authz. This attribute is required parameter when adding new user.

Table 1108. Links summary

Name	Type	Summary
domain	Domain	
groups	Group[]	
permissions	Permission[]	
roles	Role[]	
ssh_public_keys	SshPublicKey[]	
tags	Tag[]	

6.258. Value struct

Table 1109. Attributes summary

Name	Type	Summary
datum	Decimal	
detail	String	

6.259. ValueType enum

Table 1110. Values summary

Name	Summary
decimal	
integer	
string	

6.260. VcpuPin struct

Table 1111. Attributes summary

Name	Type	Summary
cpu_set	String	
vcpu	Integer	

6.261. Vendor struct

Table 1112. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.

Name	Type	Summary
id	String	
name	String	A human readable name in plain text.

6.262. Version struct

Table 1113. Attributes summary

Name	Type	Summary
build	Integer	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
full_version	String	
id	String	
major	Integer	
minor	Integer	
name	String	A human readable name in plain text.
revision	Integer	

6.263. VirtioScsi struct

Table 1114. Attributes summary

Name	Type	Summary
enabled	Boolean	

6.264. VirtualNumaNode struct

Represents virtual NUMA node.

An example XML representation:

```
<vm_numa_node href="/ovirt-engine/api/vms/f1aa8209/numanodes/706cb6b0" id="706cb6b0">
  <cpu>
    <cores>
      <core>
        <index>0</index>
      </core>
    </cores>
  </cpu>
  <index>0</index>
  <memory>1024</memory>
  <numa_node_pins>
    <numa_node_pin>
      <host_numa_node id="007cf1ab"/>
      <index>0</index>
      <pinned>true</pinned>
    </numa_node_pin>
  </numa_node_pins>
  <vm href="/ovirt-engine/api/vms/f1aa8209" id="f1aa8209"/>
</vm_numa_node>
```

Table 1115. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
cpu	Cpu	
description	String	A human readable description in plain text.
id	String	
index	Integer	
memory	Integer	Memory of the NUMA node in MB.
name	String	A human readable name in plain text.
node_distance	String	
numa_node_pins	NumaNodePin[]	

Table 1116. Links summary

Name	Type	Summary
host	Host	
statistics	Statistic[]	
vm	Vm	

6.265. Vlan struct

Table 1117. Attributes summary

Name	Type	Summary
id	Integer	

6.266. Vm struct

Table 1118. Attributes summary

Name	Type	Summary
bios	Bios	
comment	String	Free text containing comments about this object.
console	Console	
cpu	Cpu	The configuration of the CPU of the virtual machine.
cpu_shares	Integer	
creation_time	Date	
custom_compatibility_version	Version	Virtual machine custom compatibility version.
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	
delete_protected	Boolean	
description	String	A human readable description in plain text.
display	Display	
domain	Domain	
fqdn	String	
guest_operating_system	GuestOperatingSystem	
guest_time_zone	TimeZone	
high_availability	HighAvailability	
id	String	
initialization	Initialization	
io	Io	
large_icon	Icon	

Name	Type	Summary
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	
migration	MigrationOptions	
migration_downtime	Integer	
name	String	A human readable name in plain text.
next_run_configuration_exists	Boolean	
numa_tune_mode	NumaTuneMode	
origin	String	
os	OperatingSystem	
payloads	Payload[]	
placement_policy	VmPlacementPolicy	The configuration of the virtual machine's placement policy.
rng_device	RngDevice	
run_once	Boolean	
serial_number	SerialNumber	
small_icon	Icon	
soundcard_enabled	Boolean	
sso	Sso	
start_paused	Boolean	
start_time	Date	
stateless	Boolean	
status	VmStatus	
status_detail	String	
stop_reason	String	
stop_time	Date	
time_zone	TimeZone	
tunnel_migration	Boolean	
type	VmType	
usb	Usb	
use_latest_template_version	Boolean	
virtio_scsi	VirtioScsi	

6.266.1. cpu

The configuration of the CPU of the virtual machine.

The sockets configuration can be updated without rebooting the virtual machine. The cores and the threads require a reboot in order to take place.

For example, to change the number of sockets to 4 immediately and the number of cores and threads to 2 after reboot send a request:

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

With a request body:

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

6.266.2. custom_compatibility_version

Virtual machine custom compatibility version.

This field allows to customize a virtual machine to its own compatibility version. If `custom_compatibility_version` is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in and is checked against capabilities of the host the virtual machine is planned to run on.

6.266.3. memory

The virtual machine's memory, in bytes.

For example, in order to update a virtual machine to contain 1 GiB of memory send a request like this:

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

With a request body like this:

```
<vm>
  <memory>1073741824</memory>
</vm>
```

NOTE Memory in the previous example is converted to bytes using the following formula: 1 GiB = 2^{30} bytes = 1073741824 bytes.

NOTE Memory hot plug is supported from oVirt 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

6.266.4. placement_policy

The configuration of the virtual machine's placement policy.

This configuration can be updated to pin a virtual machine to one or more hosts.

NOTE Virtual machines that are pinned to multiple hosts cannot be live migrated, but in the event of a host failure, any virtual machine configured to be highly available is automatically restarted on one of the other hosts to which the virtual machine is pinned.

For example, to pin a virtual machine to two hosts, you would send a request like the following:

```
PUT /api/vms/123
```

With a request body like this:

```
<vm>
  <high_availability>
    <enabled>true</enabled>
    <priority>1</priority>
  </high_availability>
  <placement_policy>
    <hosts>
      <host>
        <name>Host1</name>
      </host>
      <host>
        <name>Host2</name>
      </host>
    </hosts>
    <affinity>pinned</affinity>
  </placement_policy>
</vm>
```

Table 1119. Links summary

Name	Type	Summary
affinity_labels	AffinityLabel[]	
applications	Application[]	
cdroms	Cdrom[]	
cluster	Cluster	
cpu_profile	CpuProfile	
disk_attachments	DiskAttachment[]	References to the disks attached to the virtual machine.
external_host_provider	ExternalHostProvider	
floppies	Floppy[]	
graphics_consoles	GraphicsConsole[]	
host	Host	
host_devices	HostDevice[]	
instance_type	InstanceType	
katello_errata	KatelloErratum[]	
nics	Nic[]	Link to the the list of network interface devices on the virtual machine.
numa_nodes	NumaNode[]	
permissions	Permission[]	
quota	Quota	
reported_devices	ReportedDevice[]	
sessions	Session[]	
snapshots	Snapshot[]	
statistics	Statistic[]	
storage_domain	StorageDomain	
tags	Tag[]	
template	Template	
vm_pool	VmPool	
watchdogs	Watchdog[]	

6.267. VmAffinity enum

Table 1120. Values summary

Name	Summary
migratable	
pinned	
user_migratable	

6.268. VmBase struct

Table 1121. Attributes summary

Name	Type	Summary
bios	Bios	
comment	String	Free text containing comments about this object.
console	Console	
cpu	Cpu	The configuration of the CPU of the virtual machine.
cpu_shares	Integer	
creation_time	Date	
custom_compatibility_version	Version	Virtual machine custom compatibility version.
custom_cpu_model	String	
custom_emulated_machine	String	
custom_properties	CustomProperty[]	
delete_protected	Boolean	
description	String	A human readable description in plain text.
display	Display	
domain	Domain	
high_availability	HighAvailability	
id	String	
initialization	Initialization	
io	Io	
large_icon	Icon	
memory	Integer	The virtual machine's memory, in bytes.
memory_policy	MemoryPolicy	
migration	MigrationOptions	
migration_downtime	Integer	
name	String	A human readable name in plain text.

Name	Type	Summary
origin	String	
os	OperatingSystem	
rng_device	RngDevice	
serial_number	SerialNumber	
small_icon	Icon	
soundcard_enabled	Boolean	
sso	Sso	
start_paused	Boolean	
stateless	Boolean	
time_zone	TimeZone	
tunnel_migration	Boolean	
type	VmType	
usb	Usb	
virtio_scsi	VirtioScsi	

6.268.1. cpu

The configuration of the CPU of the virtual machine.

The sockets configuration can be updated without rebooting the virtual machine. The cores and the threads require a reboot in order to take place.

For example, to change the number of sockets to 4 immediately and the number of cores and threads to 2 after reboot send a request:

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

With a request body:

```
<vm>
  <cpu>
    <topology>
      <sockets>4</sockets>
      <cores>2</cores>
      <threads>2</threads>
    </topology>
  </cpu>
</vm>
```

6.268.2. custom_compatibility_version

Virtual machine custom compatibility version.

This field allows to customize a virtual machine to its own compatibility version. If `custom_compatibility_version` is set, it overrides the cluster's compatibility version for this particular virtual machine.

The compatibility version of a virtual machine is limited by the data center the virtual machine resides in and is checked against capabilities of the host the virtual machine is planned to run on.

6.268.3. memory

The virtual machine's memory, in bytes.

For example, in order to update a virtual machine to contain 1 GiB of memory send a request like this:

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

With a request body like this:

```
<vm>
  <memory>1073741824</memory>
</vm>
```

NOTE	Memory in the previous example is converted to bytes using the following formula: 1 GiB = 2^{30} bytes = 1073741824 bytes.
NOTE	Memory hot plug is supported from oVirt 3.6 onwards. You can use the example above to increase memory while the virtual machine is running.

Table 1122. Links summary

Name	Type	Summary
`cluster`	Cluster	
`cpu_profile`	CpuProfile	
`quota`	Quota	
`storage_domain`	StorageDomain	

6.269. VmDeviceType enum

Table 1123. Values summary

Name	Summary
cdrom	
floppy	

6.270. VmPlacementPolicy struct

Table 1124. Attributes summary

Name	Type	Summary
affinity	VmAffinity	

Table 1125. Links summary

Name	Type	Summary
hosts	Host[]	

6.271. VmPool struct

Table 1126. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
display	Display	
id	String	
max_user_vms	Integer	
name	String	A human readable name in plain text.
prestarted_vms	Integer	
rng_device	RngDevice	
size	Integer	
soundcard_enabled	Boolean	
stateful	Boolean	Virtual machine pool's stateful flag.
type	VmPoolType	
use_latest_template_version	Boolean	

6.271.1. stateful

Virtual machine pool's stateful flag.

Virtual machines from a stateful virtual machine pool are always started in stateful mode (stateless

snapshot is not created). The state of the virtual machine is preserved even when the virtual machine is passed to a different user.

Table 1127. Links summary

Name	Type	Summary
`cluster`	Cluster	
`instance_type`	InstanceType	Reference to the instance type on which this pool is based.
`permissions`	Permission[]	
`template`	Template	
`vm`	Vm	

6.271.2. instance_type

Reference to the instance type on which this pool is based. It can be set only on pool creation and cannot be edited.

6.272. VmPoolType enum

Table 1128. Values summary

Name	Summary
`automatic`	
`manual`	

6.273. VmStatus enum

Table 1129. Values summary

Name	Summary
`down`	
`image_locked`	
`migrating`	
`not_responding`	
`paused`	
`powering_down`	
`powering_up`	
`reboot_in_progress`	
`restoring_state`	
`saving_state`	
`suspended`	
`unassigned`	

Name	Summary
`unknown`	
`up`	
`wait_for_launch`	

6.274. VmSummary struct

Table 1130. Attributes summary

Name	Type	Summary
`active`	Integer	
`migrating`	Integer	
`total`	Integer	

6.275. VmType enum

Table 1131. Values summary

Name	Summary
`desktop`	
`server`	

6.276. VnicPassThrough struct

Table 1132. Attributes summary

Name	Type	Summary
`mode`	VnicPassThroughMode	Defines if the vNIC to be implemented as a virtual device or as a pass-through to a host device.

6.277. VnicPassThroughMode enum

The enum describes whether vNIC to be implemented as a pass-through device or a virtual one. Currently it supports only 2 option, but there is a plan to add more in the future.

Table 1133. Values summary

Name	Summary
`disabled`	To be implemented as a virtual device
`enabled`	To be implemented as a pass-through device

6.278. VnicProfile struct

Table 1134. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.
custom_properties	CustomProperty[]	
description	String	A human readable description in plain text.
id	String	
name	String	A human readable name in plain text.
pass_through	VnicPassThrough	
port_mirroring	Boolean	

Table 1135. Links summary

Name	Type	Summary
network	Network	
network_filter	NetworkFilter	Network filter will enhance the admin ability to manage the network packets traffic from/to the participated VMs.
permissions	Permission[]	
qos	Qos	

6.279. VolumeGroup struct

Table 1136. Attributes summary

Name	Type	Summary
id	String	
logical_units	LogicalUnit[]	
name	String	

6.280. Watchdog struct

Table 1137. Attributes summary

Name	Type	Summary
action	WatchdogAction	
comment	String	Free text containing comments about this object.
description	String	A human readable description in plain text.
id	String	

Name	Type	Summary
model	WatchdogModel	
name	String	A human readable name in plain text.

Table 1138. Links summary

Name	Type	Summary
instance_type	InstanceType	
template	Template	
vm	Vm	Don't use this element, use vms instead.
vms	Vm[]	References to the virtual machines that are using this device.

6.280.1. vms

References to the virtual machines that are using this device. A device may be used by several virtual machines; for example, a shared disk my be used simultaneously by two or more virtual machines.

6.281. WatchdogAction enum

Table 1139. Values summary

Name	Summary
dump	
none	
pause	
poweroff	
reset	

6.282. WatchdogModel enum

Table 1140. Values summary

Name	Summary
i6300esb	

6.283. Weight struct

Table 1141. Attributes summary

Name	Type	Summary
comment	String	Free text containing comments about this object.

Name	Type	Summary
description	String	A human readable description in plain text.
factor	Integer	
id	String	
name	String	A human readable name in plain text.

Table 1142. Links summary

Name	Type	Summary
scheduling_policy	SchedulingPolicy	
scheduling_policy_unit	SchedulingPolicyUnit	

Appendix A: Primitive types

This section describes the primitive data types supported by the API.

A.1. String primitive

A finite sequence of Unicode characters.

A.2. Boolean primitive

Represents the *false* and *true* concepts used in mathematical logic.

The valid values are the strings `false` and `true`.

Case is ignored by the engine, so for example `False` and `FALSE` also valid values. However the server will always return lower case values.

For backwards compatibility with older versions of the engine, the values `0` and `1` are also accepted. The value `0` has the same meaning than `false`, and `1` has the same meaning than `true`. Try to avoid using these values, as support for them may be removed in the future.

A.3. Integer primitive

Represents the mathematical concept of integer number.

The valid values are finite sequences of decimal digits.

Currently the engine implements this type using a signed 32 bit integer, so the minimum value is -2^{31} (-2147483648) and the maximum value is 2^{31}-1 (2147483647).

However, there are some attributes in the system where the range of values possible with 32 bit isn't enough. In those exceptional cases the engine uses 64 bit integers, in particular for the

following attributes:

- `Disk.actual_size`
- `Disk.provisioned_size`
- `GlusterClient.bytes_read`
- `GlusterClient.bytes_written`
- `Host.max_scheduling_memory`
- `Host.memory`
- `HostNic.speed`
- `LogicalUnit.size`
- `MemoryPolicy.guaranteed`
- `NumaNode.memory`
- `QuotaStorageLimit.limit`
- `StorageDomain.available`
- `StorageDomain.used`
- `StorageDomain.committed`
- `VmBase.memory`

For these exception cases the minimum value is -2^{63} (-9223372036854775808) and the maximum value is $2^{63}-1$ (9223372036854775807).

NOTE	In the future the integer type will be implemented using unlimited precission integers, so the above limitations and exceptions will eventually disappear.

A.4. Decimal primitive

Represents the mathematical concept of real number.

Currently the engine implements this type using 32 bit IEEE 754 single precission floating point numbers.

For some attributes this isn't enough precission. In those exceptional cases the engine uses 64 bit double precission floating point numbers, in particular for the following attributes:

- `QuotaStorageLimit.usage`
- `QuotaStorageLimit.memory_limit`
- `QuotaStorageLimit.memory_usage`

NOTE	In the future the decimal type will be implemented using unlimited precission decimal numbers, so the above limitations and exceptions will eventually disappear.

A.5. Date primitive

Represents a date and time.

The format returned by the engine is the one described in the XML Schema specification when requesting XML. For example, if you send a request like this to retrieve the XML representation of a virtual machine:

```
GET /ovirt-engine/api/vms/123
Accept: application/xml
```

The response body will contain the following XML document:

```
<vm id="123" href="/ovirt-engine/api/vms/123">
  ...
  <creation_time>2016-09-08T09:53:35.138+02:00</creation_time>
  ...
</vm>
```

When requesting the JSON representation the engine uses a different, format: an integer containg the number of seconds since Jan 1st 1970, also know as *epoch time*. For example, if you send a request like this to retrieve the JSON representation of a virtual machine:

```
GET /ovirt-engine/api/vms/123
Accept: application/json
```

The response body will contain the following JSON document:

```
{
  "id": "123",
  "href="/ovirt-engine/api/vms/123",
  ...
  "creation_time": 1472564909990,
  ...
}
```

NOTE In both cases, the dates returned by the engine use the time zone configured in the server where it is running, in the above examples it is UTC+2.