



Red Hat Satellite 5.6

API Overview

A reference guide to the Red Hat Satellite API

Edition 1

Last Updated: 2017-09-26

Red Hat Satellite 5.6 API Overview

A reference guide to the Red Hat Satellite API
Edition 1

Daniel Macpherson
Red Hat Engineering Content Services
dmacpher@redhat.com

Legal Notice

Copyright © 2013 Red Hat, Inc.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

Abstract

This book acts as a reference for the XML-RPC API for Red Hat Satellite. It contains a short introduction to XML-RPC, examples, and a full references all Red Hat Satellite API methods.

Table of Contents

PREFACE	14
1. ABOUT THIS GUIDE	14
2. AUDIENCE	14
PART I. INTRODUCTION	15
CHAPTER 1. INTRODUCTION TO THE RED HAT SATELLITE API	16
CHAPTER 2. USING XML-RPC WITH THE RED HAT SATELLITE API	17
PART II. EXAMPLES	19
CHAPTER 3. EXAMPLES	20
3.1. PERL EXAMPLE	20
3.2. PYTHON EXAMPLE	20
3.3. RUBY EXAMPLE	21
PART III. REFERENCE	22
CHAPTER 4. NAMESPACE: ACTIVATIONKEY	23
4.1. METHOD: ADDCHILDCHANNELS	23
4.2. METHOD: ADDCONFIGCHANNELS	23
4.3. METHOD: ADDENTITLEMENTS	24
4.4. METHOD: ADDPACKAGENAMES	24
4.5. METHOD: ADDPACKAGES	25
4.6. METHOD: ADDSERVERGROUPS	25
4.7. METHOD: CHECKCONFIGDEPLOYMENT	26
4.8. METHOD: CREATE	26
4.9. METHOD: DELETE	27
4.10. METHOD: DISABLECONFIGDEPLOYMENT	27
4.11. METHOD: ENABLECONFIGDEPLOYMENT	27
4.12. METHOD: GETDETAILS	28
4.13. METHOD: LISTACTIVATEDSYSTEMS	29
4.14. METHOD: LISTACTIVATIONKEYS	29
4.15. METHOD: LISTCONFIGCHANNELS	30
4.16. METHOD: REMOVECHILDCHANNELS	31
4.17. METHOD: REMOVECONFIGCHANNELS	31
4.18. METHOD: REMOVEENTITLEMENTS	32
4.19. METHOD: REMOVEPACKAGENAMES	32
4.20. METHOD: REMOVEPACKAGES	33
4.21. METHOD: REMOVESERVERGROUPS	33
4.22. METHOD: SETCONFIGCHANNELS	34
4.23. METHOD: SETDETAILS	34
CHAPTER 5. NAMESPACE: API	36
5.1. METHOD: GETAPICALLLIST	36
5.2. METHOD: GETAPINAMESPACECALLLIST	36
5.3. METHOD: GETAPINAMESPACES	37
5.4. METHOD: GETVERSION	37
5.5. METHOD: SYSTEMVERSION	37
CHAPTER 6. NAMESPACE: AUTH	38
6.1. METHOD: LOGIN	38
6.2. METHOD: LOGOUT	38

CHAPTER 7. NAMESPACE: CHANNEL	39
7.1. METHOD: LISTALLCHANNELS	39
7.2. METHOD: LISTMYCHANNELS	39
7.3. METHOD: LISTPOPULARCHANNELS	40
7.4. METHOD: LISTREDHATCHANNELS	40
7.5. METHOD: LISTRETIREDCHANNELS	41
7.6. METHOD: LISTSHAREDCHANNELS	42
7.7. METHOD: LISTSOFTWARECHANNELS	42
7.8. METHOD: LISTVENDORCHANNELS	43
CHAPTER 8. NAMESPACE: CHANNEL.ACCESS	44
8.1. METHOD: DISABLEUSERRESTRICTIONS	44
8.2. METHOD: ENABLEUSERRESTRICTIONS	44
8.3. METHOD: GETORGSHARING	44
8.4. METHOD: SETORGSHARING	45
CHAPTER 9. NAMESPACE: CHANNEL.ORG	46
9.1. METHOD: DISABLEACCESS	46
9.2. METHOD: ENABLEACCESS	46
9.3. METHOD: LIST	46
CHAPTER 10. NAMESPACE: CHANNEL.SOFTWARE	48
10.1. METHOD: ADDPACKAGES	48
10.2. METHOD: ADDREPOFILTER	48
10.3. METHOD: ASSOCIATEREPO	48
10.4. METHOD: AVAILABLEENTITLEMENTS	50
10.5. METHOD: CLEARREPOFILTERS	50
10.6. METHOD: CLONE	50
10.7. METHOD: CREATE	51
10.8. METHOD: CREATE	52
10.9. METHOD: CREATE	53
10.10. METHOD: CREATEREPO	54
10.11. METHOD: DELETE	55
10.12. METHOD: DISASSOCIATEREPO	55
10.13. METHOD: GETCHANNELLASTBUILDID	57
10.14. METHOD: GETDETAILS	57
10.15. METHOD: GETDETAILS	58
10.16. METHOD: GETREPODETAILS	59
10.17. METHOD: GETREPODETAILS	60
10.18. METHOD: GETREPOSYNCCRONEXPRESSION	60
10.19. METHOD: ISGLOBALLYSUBSCRIBABLE	61
10.20. METHOD: ISUSERMANAGEABLE	61
10.21. METHOD: ISUSERSUBSCRIBABLE	61
10.22. METHOD: LISTALLPACKAGES	62
10.23. METHOD: LISTALLPACKAGES	62
10.24. METHOD: LISTALLPACKAGES	63
10.25. METHOD: LISTALLPACKAGES	64
10.26. METHOD: LISTALLPACKAGES	65
10.27. METHOD: LISTALLPACKAGESBYDATE	66
10.28. METHOD: LISTALLPACKAGESBYDATE	67
10.29. METHOD: LISTALLPACKAGESBYDATE	67
10.30. METHOD: LISTARCHES	68
10.31. METHOD: LISTCHANNELREPOS	68
10.32. METHOD: LISTCHILDREN	69

10.33. METHOD: LISTERRATA	70
10.34. METHOD: LISTERRATA	71
10.35. METHOD: LISTERRATA	71
10.36. METHOD: LISTERRATA	72
10.37. METHOD: LISTERRATA	73
10.38. METHOD: LISTERRATABYTYPE	74
10.39. METHOD: LISTLATESTPACKAGES	74
10.40. METHOD: LISTPACKAGESWITHOUTCHANNEL	75
10.41. METHOD: LISTREPOFILTERS	76
10.42. METHOD: LISTSUBSCRIBEDSYSTEMS	76
10.43. METHOD: LISTSYSTEMCHANNELS	76
10.44. METHOD: LISTUSERREPOS	77
10.45. METHOD: MERGEERRATA	77
10.46. METHOD: MERGEERRATA	78
10.47. METHOD: MERGEERRATA	79
10.48. METHOD: MERGEPACKAGES	79
10.49. METHOD: REGENERATENEEEDEDCACHE	80
10.50. METHOD: REGENERATENEEEDEDCACHE	80
10.51. METHOD: REGENERATEYUMCACHE	81
10.52. METHOD: REMOVEERRATA	81
10.53. METHOD: REMOVEPACKAGES	81
10.54. METHOD: REMOVEREPO	82
10.55. METHOD: REMOVEREPO	82
10.56. METHOD: REMOVEREPOFILTER	82
10.57. METHOD: SETCONTACTDETAILS	83
10.58. METHOD: SETDETAILS	83
10.59. METHOD: SETGLOBALLYSUBSCRIBABLE	84
10.60. METHOD: SETREPOFILTERS	84
10.61. METHOD: SETSYSTEMCHANNELS	85
10.62. METHOD: SETUSERMANAGEABLE	85
10.63. METHOD: SETUSERSUBSCRIBABLE	86
10.64. METHOD: SUBSCRIBESYSTEM	86
10.65. METHOD: SYNCREPO	87
10.66. METHOD: SYNCREPO	87
10.67. METHOD: UPDATEREPO	88
10.68. METHOD: UPDATEREPOLABEL	88
10.69. METHOD: UPDATEREPOURL	89
10.70. METHOD: UPDATEREPOURL	89
CHAPTER 11. NAMESPACE: CONFIGCHANNEL	91
11.1. METHOD: CHANNELEXISTS	91
11.2. METHOD: CREATE	91
11.3. METHOD: CREATEORUPDATEPATH	92
11.4. METHOD: CREATEORUPDATESYMLINK	94
11.5. METHOD: DELETECHANNELS	95
11.6. METHOD: DELETEDFILEREVISIONS	95
11.7. METHOD: DELETEDFILES	96
11.8. METHOD: DEPLOYALLSYSTEMS	96
11.9. METHOD: DEPLOYALLSYSTEMS	96
11.10. METHOD: GETDETAILS	97
11.11. METHOD: GETDETAILS	98
11.12. METHOD: GETENCODEDFILEREVISION	98
11.13. METHOD: GETFILEREVISION	99

11.14. METHOD: GETFILEREVISIONS	101
11.15. METHOD: LISTFILES	102
11.16. METHOD: LISTGLOBALS	102
11.17. METHOD: LISTSUBSCRIBEDSYSTEMS	103
11.18. METHOD: LOOKUPCHANNELINFO	104
11.19. METHOD: LOOKUPFILEINFO	104
11.20. METHOD: LOOKUPFILEINFO	106
11.21. METHOD: SCHEDULEFILECOMPARISONS	107
11.22. METHOD: UPDATE	107
CHAPTER 12. NAMESPACE: DISTCHANNEL	109
12.1. METHOD: LISTDEFAULTMAPS	109
12.2. METHOD: LISTMAPSFORORG	109
12.3. METHOD: LISTMAPSFORORG	110
12.4. METHOD: SETMAPFORORG	110
CHAPTER 13. NAMESPACE: ERRATA	112
13.1. METHOD: ADDPACKAGES	112
13.2. METHOD: APPLICABLETOCHANNELS	112
13.3. METHOD: BUGZILLAFIXES	113
13.4. METHOD: CLONE	113
13.5. METHOD: CLONEASORIGINAL	114
13.6. METHOD: CLONEASORIGINALASYNC	114
13.7. METHOD: CLONEASYNC	115
13.8. METHOD: CREATE	115
13.9. METHOD: DELETE	116
13.10. METHOD: FINDBYCVE	117
13.11. METHOD: GETDETAILS	117
13.12. METHOD: LISTAFFECTEDSYSTEMS	118
13.13. METHOD: LISTBYDATE	119
13.14. METHOD: LISTCVES	119
13.15. METHOD: LISTKEYWORDS	120
13.16. METHOD: LISTPACKAGES	120
13.17. METHOD: LISTUNPUBLISHEDERRATA	121
13.18. METHOD: PUBLISH	122
13.19. METHOD: PUBLISHASORIGINAL	122
13.20. METHOD: REMOVEPACKAGES	123
13.21. METHOD: SETDETAILS	123
CHAPTER 14. NAMESPACE: KICKSTART	126
14.1. METHOD: CLONEPROFILE	126
14.2. METHOD: CREATEPROFILE	126
14.3. METHOD: CREATEPROFILE	127
14.4. METHOD: CREATEPROFILEWITHCUSTOMURL	127
14.5. METHOD: CREATEPROFILEWITHCUSTOMURL	128
14.6. METHOD: DELETERPROFILE	128
14.7. METHOD: DISABLEPROFILE	128
14.8. METHOD: FINDKICKSTARTFORIP	129
14.9. METHOD: IMPORTFILE	129
14.10. METHOD: IMPORTFILE	130
14.11. METHOD: IMPORTFILE	130
14.12. METHOD: IMPORTRAWFILE	131
14.13. METHOD: IMPORTRAWFILE	131
14.14. METHOD: ISPROFILEDISABLED	132

14.15. METHOD: LISTALLIPRANGES	132
14.16. METHOD: LISTKICKSTARTABLECHANNELS	132
14.17. METHOD: LISTKICKSTARTABLETREES	134
14.18. METHOD: LISTKICKSTARTS	134
14.19. METHOD: RENAMEPROFILE	135
CHAPTER 15. NAMESPACE: KICKSTART.FILEPRESERVATION	136
15.1. METHOD: CREATE	136
15.2. METHOD: DELETE	136
15.3. METHOD: GETDETAILS	136
15.4. METHOD: LISTALLFILEPRESERVATIONS	137
CHAPTER 16. NAMESPACE: KICKSTART.KEYS	138
16.1. METHOD: CREATE	138
16.2. METHOD: DELETE	138
16.3. METHOD: GETDETAILS	138
16.4. METHOD: LISTALLKEYS	139
16.5. METHOD: UPDATE	139
CHAPTER 17. NAMESPACE: KICKSTART.PROFILE	141
17.1. METHOD: ADDIPRANGE	141
17.2. METHOD: ADDSCRIPT	141
17.3. METHOD: ADDSCRIPT	142
17.4. METHOD: ADDSCRIPT	142
17.5. METHOD: COMPAREACTIVATIONKEYS	143
17.6. METHOD: COMPAREADVANCEDOPTIONS	145
17.7. METHOD: COMPAREPACKAGES	145
17.8. METHOD: DOWNLOADKICKSTART	146
17.9. METHOD: DOWNLOADRENDEREDKICKSTART	147
17.10. METHOD: GETADVANCEDOPTIONS	147
17.11. METHOD: GETCFGPRESERVATION	147
17.12. METHOD: GETCHILDCHANNELS	148
17.13. METHOD: GETCUSTOMOPTIONS	148
17.14. METHOD: GETKICKSTARTTREE	149
17.15. METHOD: GETUPDATETYPE	149
17.16. METHOD: GETVARIABLES	149
17.17. METHOD: LISTIPRANGES	150
17.18. METHOD: LISTSCRIPTS	150
17.19. METHOD: REMOVEIPRANGE	151
17.20. METHOD: REMOVESCRIPT	151
17.21. METHOD: SETADVANCEDOPTIONS	152
17.22. METHOD: SETCFGPRESERVATION	152
17.23. METHOD: SETCHILDCHANNELS	153
17.24. METHOD: SETCUSTOMOPTIONS	153
17.25. METHOD: SETKICKSTARTTREE	153
17.26. METHOD: SETLOGGING	154
17.27. METHOD: SETUPDATETYPE	154
17.28. METHOD: SETVARIABLES	154
CHAPTER 18. NAMESPACE: KICKSTART.PROFILE.KEYS	156
18.1. METHOD: ADDACTIVATIONKEY	156
18.2. METHOD: GETACTIVATIONKEYS	156
18.3. METHOD: REMOVEACTIVATIONKEY	157

CHAPTER 19. NAMESPACE: KICKSTART.PROFILE.SOFTWARE	158
19.1. METHOD: APPENDTOSOFTWARELIST	158
19.2. METHOD: GETSOFTWARELIST	158
19.3. METHOD: SETSOFTWARELIST	158
CHAPTER 20. NAMESPACE: KICKSTART.PROFILE.SYSTEM	160
20.1. METHOD: ADDFILEPRESERVATIONS	160
20.2. METHOD: ADDKEYS	160
20.3. METHOD: CHECKCONFIGMANAGEMENT	160
20.4. METHOD: CHECKREMOTECOMMANDS	161
20.5. METHOD: DISABLECONFIGMANAGEMENT	161
20.6. METHOD: DISABLEREMOTECOMMANDS	161
20.7. METHOD: ENABLECONFIGMANAGEMENT	162
20.8. METHOD: ENABLEREMOTECOMMANDS	162
20.9. METHOD: GETLOCALE	163
20.10. METHOD: GETPARTITIONINGScheme	163
20.11. METHOD: GETREGISTRATIONTYPE	163
20.12. METHOD: GETSELINUX	164
20.13. METHOD: LISTFILEPRESERVATIONS	164
20.14. METHOD: LISTKEYS	165
20.15. METHOD: REMOVEFILEPRESERVATIONS	165
20.16. METHOD: REMOVEKEYS	166
20.17. METHOD: SETLOCALE	166
20.18. METHOD: SETPARTITIONINGScheme	167
20.19. METHOD: SETREGISTRATIONTYPE	167
20.20. METHOD: SETSELINUX	167
CHAPTER 21. NAMESPACE: KICKSTART.SNIPPET	169
21.1. METHOD: CREATEORUPDATE	169
21.2. METHOD: DELETE	169
21.3. METHOD: LISTALL	170
21.4. METHOD: LISTCUSTOM	170
21.5. METHOD: LISTDEFAULT	170
CHAPTER 22. NAMESPACE: KICKSTART.TREE	172
22.1. METHOD: CREATE	172
22.2. METHOD: DELETE	172
22.3. METHOD: DELETETREEANDPROFILES	172
22.4. METHOD: GETDETAILS	173
22.5. METHOD: LIST	174
22.6. METHOD: LISTINSTALLTYPES	174
22.7. METHOD: RENAME	174
22.8. METHOD: UPDATE	175
CHAPTER 23. NAMESPACE: ORG	176
23.1. METHOD: CREATE	176
23.2. METHOD: DELETE	177
23.3. METHOD: GETCRASHFILESIZELIMIT	177
23.4. METHOD: GETDETAILS	177
23.5. METHOD: GETDETAILS	178
23.6. METHOD: GETPOLICYFORSCAPFILEUPLOAD	179
23.7. METHOD: GETPOLICYFORSCAPRESULTDELETION	179
23.8. METHOD: ISCRASHREPORTINGENABLED	179
23.9. METHOD: ISCRASHFILEUPLOADENABLED	180

23.10. METHOD: LISTORGS	180
23.11. METHOD: LISTSOFTWAREENTITLEMENTS	181
23.12. METHOD: LISTSOFTWAREENTITLEMENTS	182
23.13. METHOD: LISTSOFTWAREENTITLEMENTS	182
23.14. METHOD: LISTSOFTWAREENTITLEMENTSFORORG	183
23.15. METHOD: LISTSYSTEMENTITLEMENTS	184
23.16. METHOD: LISTSYSTEMENTITLEMENTS	184
23.17. METHOD: LISTSYSTEMENTITLEMENTS	185
23.18. METHOD: LISTSYSTEMENTITLEMENTSFORORG	186
23.19. METHOD: LISTUSERS	186
23.20. METHOD: MIGRATESYSTEMS	187
23.21. METHOD: SETCRASHFILESIZELIMIT	187
23.22. METHOD: SETCRASHREPORTING	188
23.23. METHOD: SETCRASHFILEUPLOAD	188
23.24. METHOD: SETPOLICYFORSCAPFILEUPLOAD	189
23.25. METHOD: SETPOLICYFORSCAPRESULTDELETION	189
23.26. METHOD: SETSOFTWAREENTITLEMENTS	189
23.27. METHOD: SETSOFTWAREFLEXENTITLEMENTS	190
23.28. METHOD: SETSYSTEMENTITLEMENTS	190
23.29. METHOD: UPDATENAME	191
CHAPTER 24. NAMESPACE: ORG.TRUSTS	193
24.1. METHOD: ADDTRUST	193
24.2. METHOD: GETDETAILS	193
24.3. METHOD: LISTCHANNELSCONSUMED	194
24.4. METHOD: LISTCHANNELSPROVIDED	194
24.5. METHOD: LISTORGS	195
24.6. METHOD: LISTSYSTEMSAFFECTED	195
24.7. METHOD: LISTTRUSTS	196
24.8. METHOD: REMOVETRUST	196
CHAPTER 25. NAMESPACE: PACKAGES	197
25.1. METHOD: FINDBYNVREA	197
25.2. METHOD: GETDETAILS	198
25.3. METHOD: GETPACKAGE	199
25.4. METHOD: GETPACKAGEURL	199
25.5. METHOD: LISTCHANGELOG	199
25.6. METHOD: LISTDEPENDENCIES	200
25.7. METHOD: LISTFILES	200
25.8. METHOD: LISTPROVIDINGCHANNELS	201
25.9. METHOD: LISTPROVIDINGERRATA	201
25.10. METHOD: REMOVEPACKAGE	202
CHAPTER 26. NAMESPACE: PACKAGES.PROVIDER	203
26.1. METHOD: ASSOCIATEKEY	203
26.2. METHOD: LIST	203
26.3. METHOD: LISTKEYS	204
CHAPTER 27. NAMESPACE: PACKAGES.SEARCH	205
27.1. METHOD: ADVANCED	205
27.2. METHOD: ADVANCEDWITHACTKEY	205
27.3. METHOD: ADVANCEDWITHCHANNEL	206
27.4. METHOD: NAME	207
27.5. METHOD: NAMEANDDESCRIPTION	208

27.6. METHOD: NAMEANDSUMMARY	209
CHAPTER 28. NAMESPACE: PREFERENCES.LOCALE	210
28.1. METHOD: LISTLOCALES	210
28.2. METHOD: LISTTIMEZONES	210
28.3. METHOD: SETLOCALE	210
28.4. METHOD: SETTIMEZONE	211
CHAPTER 29. NAMESPACE: PROXY	212
29.1. METHOD: ACTIVATEPROXY	212
29.2. METHOD: CREATEMONITORINGSCOUT	212
29.3. METHOD: DEACTIVATEPROXY	212
29.4. METHOD: ISPROXY	213
29.5. METHOD: LISTAVAILABLEPROXYCHANNELS	213
CHAPTER 30. NAMESPACE: SATELLITE	214
30.1. METHOD: GETCERTIFICATEEXPIRATIONDATE	214
30.2. METHOD: ISMONITORINGENABLED	214
30.3. METHOD: ISMONITORINGENABLEDBYSYSTEMID	214
30.4. METHOD: LISTENTITLEMENTS	215
30.5. METHOD: LISTPROXIES	215
CHAPTER 31. NAMESPACE: SCHEDULE	217
31.1. METHOD: ARCHIVEACTIONS	217
31.2. METHOD: CANCELACTIONS	217
31.3. METHOD: DELETEACTIONS	217
31.4. METHOD: LISTALLACTIONS	218
31.5. METHOD: LISTARCHIVEDACTIONS	218
31.6. METHOD: LISTCOMPLETEDACTIONS	219
31.7. METHOD: LISTCOMPLETEDSYSTEMS	220
31.8. METHOD: LISTFAILED ACTIONS	220
31.9. METHOD: LISTFAILED SYSTEMS	221
31.10. METHOD: LISTINPROGRESSACTIONS	221
31.11. METHOD: LISTINPROGRESSSYSTEMS	222
31.12. METHOD: RESCHEDULEACTIONS	223
CHAPTER 32. NAMESPACE: SYNC.MASTER	224
32.1. METHOD: ADDTOMASTER	224
32.2. METHOD: CREATE	224
32.3. METHOD: DELETE	225
32.4. METHOD: GETDEFAULTMASTER	225
32.5. METHOD: GETMASTER	225
32.6. METHOD: GETMASTERBYLABEL	226
32.7. METHOD: GETMASTERORGS	226
32.8. METHOD: GETMASTERS	227
32.9. METHOD: MAKEDEFAULT	227
32.10. METHOD: MAPTOLOCAL	227
32.11. METHOD: SETCACERT	228
32.12. METHOD: SETMASTERORGS	228
32.13. METHOD: UNSETDEFAULTMASTER	229
32.14. METHOD: UPDATE	229
CHAPTER 33. NAMESPACE: SYNC.SLAVE	230
33.1. METHOD: CREATE	230
33.2. METHOD: DELETE	230

33.3. METHOD: GETALLOWEDORGS	231
33.4. METHOD: GETSLAVE	231
33.5. METHOD: GETSLAVEBYNAME	231
33.6. METHOD: GETSLAVES	232
33.7. METHOD: SETALLOWEDORGS	232
33.8. METHOD: UPDATE	233
CHAPTER 34. NAMESPACE: SYSTEM	234
34.1. METHOD: ADDENTITLEMENTS	234
34.2. METHOD: ADDNOTE	234
34.3. METHOD: APPLYERRATA	234
34.4. METHOD: COMPAREPACKAGEPROFILE	235
34.5. METHOD: COMPAREPACKAGES	236
34.6. METHOD: CONVERTTOFLEXENTITLEMENT	237
34.7. METHOD: CREATEPACKAGEPROFILE	237
34.8. METHOD: CREATESYSTEMRECORD	237
34.9. METHOD: DELETECUSTOMVALUES	238
34.10. METHOD: DELETEGUESTPROFILES	238
34.11. METHOD: DELETENOTE	239
34.12. METHOD: DELETENOTES	239
34.13. METHOD: DELETEPACKAGEPROFILE	239
34.14. METHOD: DELETESYSTEM	240
34.15. METHOD: DELETESYSTEM	240
34.16. METHOD: DELETESYSTEMS	240
34.17. METHOD: DELETETAGFROMSNAPSHOT	241
34.18. METHOD: DOWNLOADSYSTEMID	241
34.19. METHOD: GETCONNECTIONPATH	241
34.20. METHOD: GETCPU	242
34.21. METHOD: GETCUSTOMVALUES	243
34.22. METHOD: GETDETAILS	243
34.23. METHOD: GETDEVICES	244
34.24. METHOD: GETDMI	245
34.25. METHOD: GETENTITLEMENTS	245
34.26. METHOD: GETEVENTHISTORY	246
34.27. METHOD: GETID	246
34.28. METHOD: GETMEMORY	247
34.29. METHOD: GETNAME	247
34.30. METHOD: GETNETWORK	248
34.31. METHOD: GETNETWORKDEVICES	248
34.32. METHOD: GETREGISTRATIONDATE	249
34.33. METHOD: GETRELEVANTERRATA	249
34.34. METHOD: GETRELEVANTERRATABYTYPE	250
34.35. METHOD: GETRUNNINGKERNEL	250
34.36. METHOD: GETSCRIPTACTIONDETAILS	251
34.37. METHOD: GETSCRIPTRESULTS	251
34.38. METHOD: GETSUBSCRIBEDBASECHANNEL	252
34.39. METHOD: GETSYSTEMCURRENCYMULTIPLIERS	253
34.40. METHOD: GETSYSTEMCURRENCYSCORES	254
34.41. METHOD: GETUNSCHEDULEDERRATA	254
34.42. METHOD: GETUUID	255
34.43. METHOD: GETVARIABLES	255
34.44. METHOD: ISNVREINSTALLED	256
34.45. METHOD: ISNVREINSTALLED	256

34.46. METHOD: LISTACTIVATIONKEYS	257
34.47. METHOD: LISTACTIVESYSTEMS	257
34.48. METHOD: LISTACTIVESYSTEMSDETAILS	257
34.49. METHOD: LISTADMINISTRATORS	260
34.50. METHOD: LISTALLINSTALLABLEPACKAGES	260
34.51. METHOD: LISTBASECHANNELS	261
34.52. METHOD: LISTCHILDCHANNELS	261
34.53. METHOD: LISTDUPLICATESBYHOSTNAME	262
34.54. METHOD: LISTDUPLICATESBYIP	263
34.55. METHOD: LISTDUPLICATESBYMAC	263
34.56. METHOD: LISTELIGIBLEFLEXGUESTS	264
34.57. METHOD: LISTEXTRAPACKAGES	264
34.58. METHOD: LISTFLEXGUESTS	265
34.59. METHOD: LISTGROUPS	265
34.60. METHOD: LISTINACTIVESYSTEMS	266
34.61. METHOD: LISTINACTIVESYSTEMS	266
34.62. METHOD: LISTLATESTAVAILABLEPACKAGE	267
34.63. METHOD: LISTLATESTINSTALLABLEPACKAGES	267
34.64. METHOD: LISTLATESTUPGRADABLEPACKAGES	268
34.65. METHOD: LISTNEWERINSTALLEDPACKAGES	269
34.66. METHOD: LISTNOTES	269
34.67. METHOD: LISTOLDERINSTALLEDPACKAGES	270
34.68. METHOD: LISTOUTOFDATESYSTEMS	271
34.69. METHOD: LISTPACKAGEPROFILES	271
34.70. METHOD: LISTPACKAGES	271
34.71. METHOD: LISTPACKAGESFROMCHANNEL	272
34.72. METHOD: LISTPHYSICALSYSTEMS	273
34.73. METHOD: LISTSUBSCRIBABLEBASECHANNELS	273
34.74. METHOD: LISTSUBSCRIBABLECHILDCHANNELS	274
34.75. METHOD: LISTSUBSCRIBEDCHILDCHANNELS	274
34.76. METHOD: LISTSYSTEMEVENTS	276
34.77. METHOD: LISTSYSTEMS	277
34.78. METHOD: LISTSYSTEMSWITHEXTRAPACKAGES	278
34.79. METHOD: LISTSYSTEMSWITHPACKAGE	278
34.80. METHOD: LISTSYSTEMSWITHPACKAGE	278
34.81. METHOD: LISTUNGROUPEDSYSTEMS	279
34.82. METHOD: LISTUSERSYSTEMS	279
34.83. METHOD: LISTUSERSYSTEMS	280
34.84. METHOD: LISTVIRTUALGUESTS	280
34.85. METHOD: LISTVIRTUALHOSTS	281
34.86. METHOD: OBTAINREACTIVATIONKEY	281
34.87. METHOD: OBTAINREACTIVATIONKEY	282
34.88. METHOD: PROVISIONSYSTEM	282
34.89. METHOD: PROVISIONSYSTEM	282
34.90. METHOD: PROVISIONVIRTUALGUEST	283
34.91. METHOD: PROVISIONVIRTUALGUEST	283
34.92. METHOD: PROVISIONVIRTUALGUEST	284
34.93. METHOD: REMOVEENTITLEMENTS	284
34.94. METHOD: SCHEDULEAPPLYERRATA	285
34.95. METHOD: SCHEDULEAPPLYERRATA	285
34.96. METHOD: SCHEDULEAPPLYERRATA	286
34.97. METHOD: SCHEDULEAPPLYERRATA	286
34.98. METHOD: SCHEDULEGUESTACTION	287

34.99. METHOD: SCHEDULEGUESTACTION	287
34.100. METHOD: SCHEDULEHARDWAREREFRESH	287
34.101. METHOD: SCHEDULEPACKAGEINSTALL	288
34.102. METHOD: SCHEDULEPACKAGEREFRESH	288
34.103. METHOD: SCHEDULEPACKAGEREMOVE	289
34.104. METHOD: SCHEDULEREBOOT	289
34.105. METHOD: SCHEDULESCRIPTRUN	289
34.106. METHOD: SCHEDULESCRIPTRUN	290
34.107. METHOD: SCHEDULESYNCPACKAGESWITHSYSTEM	291
34.108. METHOD: SEARCHBYNAME	291
34.109. METHOD: SETBASECHANNEL	291
34.110. METHOD: SETBASECHANNEL	292
34.111. METHOD: SETCHILDCHANNELS	292
34.112. METHOD: SETCUSTOMVALUES	293
34.113. METHOD: SETDETAILS	293
34.114. METHOD: SETGROUPMEMBERSHIP	294
34.115. METHOD: SETGUESTCPUS	294
34.116. METHOD: SETGUESTMEMORY	295
34.117. METHOD: SETLOCKSTATUS	295
34.118. METHOD: SETPRIMARYINTERFACE	296
34.119. METHOD: SETPROFILENAME	296
34.120. METHOD: SETVARIABLES	296
34.121. METHOD: TAGLATESTSNAPSHOT	297
34.122. METHOD: UPGRADEENTITLEMENT	297
34.123. METHOD: WHOREGISTERED	298
CHAPTER 35. NAMESPACE: SYSTEM.CONFIG	299
35.1. METHOD: ADDCHANNELS	299
35.2. METHOD: CREATEORUPDATEPATH	299
35.3. METHOD: CREATEORUPDATESYMLINK	301
35.4. METHOD: DELETEFILES	303
35.5. METHOD: DEPLOYALL	303
35.6. METHOD: LISTCHANNELS	304
35.7. METHOD: LISTFILES	304
35.8. METHOD: LOOKUPFILEINFO	305
35.9. METHOD: REMOVECHANNELS	307
35.10. METHOD: SETCHANNELS	307
CHAPTER 36. NAMESPACE: SYSTEM.CRASH	309
36.1. METHOD: CREATECRASHNOTE	309
36.2. METHOD: DELETECRASH	309
36.3. METHOD: DELETECRASHNOTE	309
36.4. METHOD: GETCRASHCOUNTINFO	310
36.5. METHOD: GETCRASHFILE	310
36.6. METHOD: GETCRASHFILEURL	311
36.7. METHOD: GETCRASHNOTESFORCRASH	311
36.8. METHOD: GETCRASHOVERVIEW	311
36.9. METHOD: GETCRASHESBYUUID	312
36.10. METHOD: LISTSYSTEMCRASHFILES	312
36.11. METHOD: LISTSYSTEMCRASHES	313
CHAPTER 37. NAMESPACE: SYSTEM.CUSTOMINFO	315
37.1. METHOD: CREATEKEY	315
37.2. METHOD: DELETEKEY	315

37.3. METHOD: LISTALLKEYS	315
37.4. METHOD: UPDATEKEY	316
CHAPTER 38. NAMESPACE: SYSTEM.PROVISIONING.SNAPSHOT	317
38.1. METHOD: ADDTAGTOSNAPSHOT	317
38.2. METHOD: DELETESNAPSHOT	317
38.3. METHOD: DELETESNAPSHOTS	317
38.4. METHOD: DELETESNAPSHOTS	318
38.5. METHOD: LISTSNAPSHOTCONFIGFILES	319
38.6. METHOD: LISTSNAPSHOTPACKAGES	320
38.7. METHOD: LISTSNAPSHOTS	320
CHAPTER 39. NAMESPACE: SYSTEM.SCAP	323
39.1. METHOD: DELETEXCCDFSCAN	323
39.2. METHOD: GETXCCDFSCANDETAILS	323
39.3. METHOD: GETXCCDFSCANRULERESULTS	324
39.4. METHOD: LISTXCCDFSCANS	324
39.5. METHOD: SCHEDULEXCCDFSCAN	325
39.6. METHOD: SCHEDULEXCCDFSCAN	325
39.7. METHOD: SCHEDULEXCCDFSCAN	326
39.8. METHOD: SCHEDULEXCCDFSCAN	326
CHAPTER 40. NAMESPACE: SYSTEM.SEARCH	327
40.1. METHOD: DEVICEDESCRIPTION	327
40.2. METHOD: DEVICEDRIVER	327
40.3. METHOD: DEVICEID	328
40.4. METHOD: DEVICEVENDORID	329
40.5. METHOD: HOSTNAME	329
40.6. METHOD: IP	330
40.7. METHOD: NAMEANDDESCRIPTION	331
40.8. METHOD: UUID	331
CHAPTER 41. NAMESPACE: SYSTEMGROUP	333
41.1. METHOD: ADDORREMOVEADMINS	333
41.2. METHOD: ADDORREMOVESYSTEMS	333
41.3. METHOD: CREATE	334
41.4. METHOD: DELETE	334
41.5. METHOD: GETDETAILS	334
41.6. METHOD: GETDETAILS	335
41.7. METHOD: LISTACTIVESYSTEMSINGROUP	335
41.8. METHOD: LISTADMINISTRATORS	336
41.9. METHOD: LISTALLGROUPS	336
41.10. METHOD: LISTGROUPSWITHNOASSOCIATEDADMINS	337
41.11. METHOD: LISTINACTIVESYSTEMSINGROUP	337
41.12. METHOD: LISTINACTIVESYSTEMSINGROUP	338
41.13. METHOD: LISTSYSTEMS	338
41.14. METHOD: SCHEDULEAPPLYERRATATOACTIVE	339
41.15. METHOD: SCHEDULEAPPLYERRATATOACTIVE	340
41.16. METHOD: UPDATE	340
CHAPTER 42. NAMESPACE: USER	342
42.1. METHOD: ADDASSIGNEDSYSTEMGROUP	342
42.2. METHOD: ADDASSIGNEDSYSTEMGROUPS	342
42.3. METHOD: ADDDEFAULTSYSTEMGROUP	343

42.4. METHOD: ADDDEFAULTSYSTEMGROUPS	343
42.5. METHOD: ADDROLE	343
42.6. METHOD: CREATE	344
42.7. METHOD: CREATE	344
42.8. METHOD: DELETE	345
42.9. METHOD: DISABLE	345
42.10. METHOD: ENABLE	345
42.11. METHOD: GETDETAILS	346
42.12. METHOD: GETLOGGEDINTIME	346
42.13. METHOD: LISTASSIGNABLEROLES	347
42.14. METHOD: LISTASSIGNEDSYSTEMGROUPS	347
42.15. METHOD: LISTDEFAULTSYSTEMGROUPS	348
42.16. METHOD: LISTROLES	348
42.17. METHOD: LISTUSERS	349
42.18. METHOD: REMOVEASSIGNEDSYSTEMGROUP	349
42.19. METHOD: REMOVEASSIGNEDSYSTEMGROUPS	349
42.20. METHOD: REMOVEDEFAULTSYSTEMGROUP	350
42.21. METHOD: REMOVEDEFAULTSYSTEMGROUPS	350
42.22. METHOD: REMOVEROLE	351
42.23. METHOD: SETDETAILS	351
42.24. METHOD: USEPAMAUTHENTICATION	352
APPENDIX A. REVISION HISTORY	353

PREFACE

Red Hat Network (<https://access.redhat.com/home>) provides system-level support and management of Red Hat systems and networks. It brings together the tools, services, and information repositories needed to maximize the reliability, security, and performance of Red Hat systems. To use Red Hat Network, system administrators register software and hardware profiles, known as *System Profiles*, of their client systems with Red Hat Network. When a client system requests package updates, only the applicable packages for the client are returned.

Red Hat Satellite allows organizations to use the benefits of Red Hat Network without having to provide public Internet access to their servers or other client systems. System profiles are stored locally on the Satellite server. The Satellite website is served from a local web server and is only accessible to systems that can reach the Satellite server. All package management tasks, including errata updates, are performed through the Satellite server.

Red Hat Satellite provides a solution for organizations that require absolute control over and privacy of the maintenance and package deployment of their servers. It allows Red Hat Network customers the greatest flexibility and power in keeping systems secure and updated. Modules can be added to the Satellite server to provide extra functionality.

1. ABOUT THIS GUIDE

This guide explains how to use the Red Hat Satellite XML-RPC API.

2. AUDIENCE

The target audience for this guide includes **developers** and **system administrators** who aim to integrate and manage Red Hat Satellite with external clients and products.

PART I. INTRODUCTION

CHAPTER 1. INTRODUCTION TO THE RED HAT SATELLITE API

Red Hat Satellite provides organizations with the benefits of Red Hat Network without the need for public Internet access for servers or client systems. In addition, users of Red Hat Satellite can:

- Maintain complete control and privacy over package management and server maintenance within their own networks;
- Store System Profiles on a Satellite server, which connect to the Red Hat Network website via a local web server; and,
- Perform package management tasks, including errata updates, through the local area network.

This gives Red Hat Network customers the greatest flexibility and power to keep servers secure and up-to-date.

Red Hat Satellite includes an application programming interface (API), which offers software developers and system administrators control over their Satellite servers outside of the standard web and command line interfaces. The API is useful for developers and administrators who aim to integrate the functionality of a Red Hat Satellite with custom scripts or external applications that access the API via XML remote procedure call (RPC) protocol, or XML-RPC.

This documentation acts as a reference to the Red Hat Satellite XML-RPC API. It aims to provide developers and administrators with instructions and examples to help harness the functionality of their Red Hat Satellite through the XML-RPM protocol.

CHAPTER 2. USING XML-RPC WITH THE RED HAT SATELLITE API

XML-RPC uses HTTP to send an XML-encoded request to a server. The request contains the following:

Namespace

A namespace is a grouping of methods based upon a particular function, object or resource. For example, the `auth` namespace groups authentication function, or the `errata` namespace groups function that control errata.

Method

A method represents a certain action. Each method controls a specific function of the Red Hat Satellite. For example, the `login` method in the `auth` namespace logs a user into Red Hat Satellite and returns a session key.

Parameter

A parameter is a piece of input to control specific aspects of a method. For example, the `login` method in the `auth` namespace requires `username` and `password` parameters to specify the login details of a certain user. The `login` method also accepts an optional `duration` parameter to specify the length of time until the user session expires..

The XML-encoded request usually appears in the following structure:

```
<methodCall>
  <methodName>namespace.method</methodName>
  <params>
    <param>
      <value><name>parameter</name></value>
    </param>
    <param>
      <value><name>parameter</name></value>
    </param>
    ...
  </params>
</methodCall>
```

For example, use the following XML-RPC request to retrieve a list of all Red Hat Satellite API namespaces:

```
<methodCall>
  <methodName>auth.login</methodName>
  <params>
    <param>
      <value><username>admin</username></value>
    </param>
    <param>
      <value><password>p@55w0rd!</password></value>
    </param>
  </params>
</methodCall>
```

In this example, the request sends the **username** and **password** as a parameters to the **login** method, which is a part of the **auth** namespace. The Satellite server returns the following response:

```
<methodResponse>
  <params>
    <param>
      <value><sessionKey>83d8b35f</sessionKey></value>
    </param>
  </params>
</methodResponse>
```

The response contains a returned parameter for **sessionKey**, which is a key used to authenticate most of the API methods.

Most programming languages include XML-RPC modules and libraries that automatically parse the desired method and parameters into the XML-encoded request and parse the resulting response as returned variable. For examples, see [Part II, “Examples”](#).

For more information about XML-RPC, see <http://www.xmlrpc.com/>.

PART II. EXAMPLES

CHAPTER 3. EXAMPLES

The following sections demonstrate Red Hat Satellite API usage using different programming languages and their respective XML-RPC requests.

3.1. PERL EXAMPLE

This Perl example shows the `system.listUserSystems` call being used to get a list of systems a user has access. The example prints the name of each system. The `Frontier::Client` Perl module is found in the `perl-Frontier-RPC` RPM.

```
#!/usr/bin/perl
use Frontier::Client;

my $HOST = 'satellite.example.com';
my $user = 'username';
my $pass = 'password';

my $client = new Frontier::Client(url => "http://$HOST/rpc/api");
my $session = $client->call('auth.login',$user, $pass);

my $systems = $client->call('system.listUserSystems', $session);
foreach my $system (@$systems) {
    print $system->{'name'}."\n";
}
$client->call('auth.logout', $session);
```

3.2. PYTHON EXAMPLE

The following example demonstrates the `user.listUsers` call. The example prints the name of each group.

```
#!/usr/bin/python
import xmlrpclib

SATELLITE_URL = "http://satellite.example.com/rpc/api"
SATELLITE_LOGIN = "username"
SATELLITE_PASSWORD = "password"

client = xmlrpclib.Server(SATELLITE_URL, verbose=0)

key = client.auth.login(SATELLITE_LOGIN, SATELLITE_PASSWORD)
list = client.user.list_users(key)
for user in list:
    print user.get('login')

client.auth.logout(key)
```

The following example shows how to use date-time parameters. This code schedules the installation of package `rhnlb-2.5.22.9.el6.noarch` to system with id `1000000001`.


```
#!/usr/bin/python
from datetime import datetime
import time
import xmlrpclib

SATELLITE_URL = "http://satellite.example.com/rpc/api"
SATELLITE_LOGIN = "username"
SATELLITE_PASSWORD = "password"

client = xmlrpclib.Server(SATELLITE_URL, verbose=0)

key = client.auth.login(SATELLITE_LOGIN, SATELLITE_PASSWORD)
package_list = client.packages.findByNvrea(key, 'rhnlb', '2.5.22',
'9.el6', '', 'noarch')
today = datetime.today()
earliest_occurrence = xmlrpclib.DateTime(today)
client.system.schedulePackageInstall(key, 1000000001, package_list[0]
['id'], earliest_occurrence)

client.auth.logout(key)
```

3.3. RUBY EXAMPLE

The following example demonstrates the `channel.listAllChannels` API call. The example prints a list of channel labels.

```
#!/usr/bin/env ruby
require "xmlrpc/client"

@SATELLITE_URL = "http://satellite.example.com/rpc/api"
@SATELLITE_LOGIN = "username"
@SATELLITE_PASSWORD = "password"

@client = XMLRPC::Client.new2(@SATELLITE_URL)

@key = @client.call('auth.login', @SATELLITE_LOGIN, @SATELLITE_PASSWORD)
channels = @client.call('channel.listAllChannels', @key)
for channel in channels do
  p channel["label"]
end

@client.call('auth.logout', @key)
```

PART III. REFERENCE

CHAPTER 4. NAMESPACE: ACTIVATIONKEY

Contains methods to access common activation key functions available from the web interface.

Namespace: `activationkey`

4.1. METHOD: ADDCHILDCHANNELS

Description

Add child channels to an activation key.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string key`
- `array:`
 - `string - childChannelLabel`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

4.2. METHOD: ADDCONFIGCHANNELS

Description

When given a list of activation keys and configuration channels, this method adds given configuration channels to either the top or the bottom (whichever you specify) of an activation key's configuration channels list. The ordering of the configuration channels provided in the add list remains maintained while adding. If one of the configuration channels in the 'add' list already exists in an activation key, the configuration channel's ranking moves to the appropriate place.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `string - activationKey`
- `array:`
 - `string - List of configuration channel labels in the ranked order.`
- `boolean addToTop`
 - `true` - To prepend the given channels to the beginning of the activation key's config channel list

- `false` - To append the given channels to the end of the activation key's config channel list

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

4.3. METHOD: ADDENTITLEMENTS

Description

Add entitlements to an activation key. Currently only add-on entitlements are permitted e.g. `monitoring_entitled`, `provisioning_entitled`, `virtualization_host`, `virtualization_host_platform`.

Parameters

The following parameters are available for this method:

- `string` `sessionKey`
- `string` `key`
- `array`:
 - `string` - `entitlement` label
 - `monitoring_entitled`
 - `provisioning_entitled`
 - `virtualization_host`
 - `virtualization_host_platform`

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

4.4. METHOD: ADDPACKAGENAMES

Description

Add packages to an activation key using package name only.



NOTE

This method is deprecated. Use the `addPackages` (`string` `sessionKey`, `string` `key`, `array[packages]`) method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`
- array:
 - string - `packageName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

Available since: 10.2

4.5. METHOD: ADDPACKAGES

Description

Add packages to an activation key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`
- array:
 - struct - `packages`
 - string `name` - Package name
 - string `arch` - Architecture label (Optional)

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.6. METHOD: ADDSERVERGROUPS

Description

Add server groups to an activation key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`
- array:

- `int - serverGroupId`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

4.7. METHOD: CHECKCONFIGDEPLOYMENT

Description

Check configuration file deployment status for the activation key specified.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string key`

Returns

The following return values are available for this method:

- `1` if enabled, `0` if disabled, exception thrown otherwise.

4.8. METHOD: CREATE

Description

Create a new activation key. The passed activation key parameter is prefixed with the organization ID. This value is returned from the `create` call. For example, if the caller passes in the key `foo` and adds to an organization with the ID `100`, the actual activation key will be `100-foo`. This allows setting a usage limit on this activation key. If unlimited usage is desired, use a `create` method that does not include this parameter.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string key` - Leave empty to have new key generated.
- `string description`
- `string baseChannelLabel` - Leave empty to accept default.
- `int usageLimit` - If unlimited usage is desired, use a `create` method that does not include this parameter.
- `array`:
 - `string - Add-on entitlement label to associate with the key.`
 - `monitoring_entitled`

- provisioning_entitled
- virtualization_host
- virtualization_host_platform
- boolean `universalDefault`

Returns

The following return values are available for this method:

- string - The new activation key.

4.9. METHOD: DELETE

Description

Delete an activation key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.10. METHOD: DISABLECONFIGDEPLOYMENT

Description

Disable configuration file deployment for the specified activation key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.11. METHOD: ENABLECONFIGDEPLOYMENT

Description

Enable configuration file deployment for the specified activation key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.12. METHOD: GETDETAILS

Description

View an activation key's details.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`

Returns

The following return values are available for this method:

- struct - activation key
 - string `key`
 - string `description`
 - int `usage_limit`
 - string `base_channel_label`
 - array `child_channel_labels`
 - string `childChannelLabel`
 - array `entitlements`
 - string `entitlementLabel`
 - array `server_group_ids`
 - string `serverGroupId`
 - array `package_names`
 - string `packageName` - deprecated by `packages`

- array `packages`
 - `struct` - package
 - `string name` - package name
 - `string arch` - architecture label (optional)
- boolean `universal_default`
- boolean `disabled`

Available since:10.2

4.13. METHOD: LISTACTIVATEDSYSTEMS

Description

List the systems activated with the key provided.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string key`

Returns

The following return values are available for this method:

- `array:`
 - `struct` - system structure
 - `int id` - System id
 - `string hostname`
 - `dateTime.iso8601 last_checkin` - Last time server successfully checked in

4.14. METHOD: LISTACTIVATIONKEYS

Description

List activation keys that are visible to the user.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`

- struct - activation key
 - string key
 - string description
 - int usage_limit
 - string base_channel_label
 - array child_channel_labels
 - string childChannelLabel
 - array entitlements
 - string entitlementLabel
 - array server_group_ids
 - string serverGroupId
 - array package_names
 - string packageName - deprecated by packages
 - array packages
 - struct - package
 - string name - package name
 - string arch - architecture label (optional)
 - boolean universal_default
 - boolean disabled

Available since:10.2

4.15. METHOD: LISTCONFIGCHANNELS

Description

List configuration channels associated to an activation key.

Parameters

The following parameters are available for this method:

- string sessionKey
- string key

Returns

The following return values are available for this method:

- array:
 - struct - Configuration Channel information
 - int `id`
 - int `orgId`
 - string `label`
 - string `name`
 - string `description`
 - struct `configChannelType`
 - struct - Configuration Channel Type information
 - int `id`
 - string `label`
 - string `name`
 - int `priority`

4.16. METHOD: REMOVECHILDCHANNELS

Description

Remove child channels from an activation key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`
- array:
 - string - `childChannelLabel`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.17. METHOD: REMOVECONFIGCHANNELS

Description

Remove configuration channels from the given activation keys.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - string - `activationKey`
- array:
 - string - `configChannelLabel`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.18. METHOD: REMOVEENTITLEMENTS

Description

Remove entitlements (by label) from an activation key. Currently only add-on entitlements are permitted e.g. `monitoring_entitled`, `provisioning_entitled`, `virtualization_host`, `virtualization_host_platform`.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`
- array:
 - string - `entitlementlabel`
 - `monitoring_entitled`
 - `provisioning_entitled`
 - `virtualization_host`
 - `virtualization_host_platform`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.19. METHOD: REMOVEPACKAGENAMES

Description

Remove package names from an activation key.

**NOTE**

This method is deprecated. Use the `removePackages(string sessionKey, string key, array[packages])`

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`
- array:
 - string - `packageName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

Available since: 10.2

4.20. METHOD: REMOVEPACKAGES

Description

Remove package names from an activation key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`
- array:
 - struct - `packages`
 - string `name` - Package name
 - string `arch` - Architecture label (optional)

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.21. METHOD: REMOVESERVERGROUPS

Description

Remove server groups from an activation key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`
- array:
 - int - `serverGroupId`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.22. METHOD: SETCONFIGCHANNELS

Description

Replace the existing set of configuration channels on the given activation keys. Channels are ranked by their order in the array.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - string - `activationKey`
- array:
 - string - `configChannelLabel`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

4.23. METHOD: SETDETAILS

Description

Update the details of an activation key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `key`

- **struct** - activation key
 - **string description** (optional)
 - **string base_channel_label** (optional)
 - **int usage_limit** (optional)
 - **boolean unlimited_usage_limit** - Set **true** for unlimited usage and to override **usage_limit**
 - **boolean universal_default** (optional)
 - **boolean disabled** (optional)

Returns

The following return values are available for this method:

- **int** - **1** on success, exception thrown otherwise.

CHAPTER 5. NAMESPACE: API

Methods providing information about the API.

Namespace: `api`

5.1. METHOD: GETAPICALLLIST

Description

Lists all available API calls grouped by namespace.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `struct - method_info`
 - `string name` - method name
 - `string parameters` - method parameters
 - `string exceptions` - method exceptions
 - `string return` - method return type

5.2. METHOD: GETAPINAMESPACECALLLIST

Description

Lists all available API calls for the specified namespace.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string namespace`

Returns

The following return values are available for this method:

- `struct - method_info`
 - `string name` - method name
 - `string parameters` - method parameters
 - `string exceptions` - method exceptions

- `string return` - method return type

5.3. METHOD: GETAPINAMESPACES

Description

Lists available API namespaces.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `struct` - namespace
 - `string namespace` - API namespace
 - `string handler` - API Handler

5.4. METHOD: GETVERSION

Description

Returns the version of the API. Since Satellite 5.3 it is no longer related to server version.

Returns

The following return values are available for this method:

- `string` - the API version

5.5. METHOD: SYSTEMVERSION

Description

Returns the server version.

Returns

The following return values are available for this method:

- `string` - the server version

CHAPTER 6. NAMESPACE: AUTH

This namespace provides methods to authenticate with the system's management server.

Namespace: `auth`

6.1. METHOD: LOGIN

Description

Log in with a username and password. Returns the session key most other API methods use.

Parameters

The following parameters are available for this method:

- `string username`
- `string password`
- `int duration` - length of session (optional)

Returns

The following return values are available for this method:

- `string sessionKey`

6.2. METHOD: LOGOUT

Description

Logout the user with the given session key.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

CHAPTER 7. NAMESPACE: CHANNEL

Provides method to get back a list of Software Channels.

Namespace: `channel`

7.1. METHOD: LISTALLCHANNELS

Description

List all software channels the user's organization is entitled.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - channel info`
 - `int id`
 - `string label`
 - `string name`
 - `string provider_name`
 - `int packages`
 - `int systems`
 - `string arch_name`

7.2. METHOD: LISTMYCHANNELS

Description

List all software channels that belong to the user's organization.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`

- struct - channel info
 - int `id`
 - string `label`
 - string `name`
 - string `provider_name`
 - int `packages`
 - int `systems`
 - string `arch_name`

7.3. METHOD: LISTPOPULARCHANNELS

Description

List the most popular software channels. This returns the channels with the number of systems subscribed as specified by the popularity count.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `popularityCount`

Returns

The following return values are available for this method:

- array:
 - struct - channel info
 - int `id`
 - string `label`
 - string `name`
 - string `provider_name`
 - int `packages`
 - int `systems`
 - string `arch_name`

7.4. METHOD: LISTREDHATCHANNELS

Description

List all Red Hat software channels the user's organization is entitled.



NOTE

This method is deprecated. Use the `listVendorChannels(String sessionKey)` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - channel info
 - int `id`
 - string `label`
 - string `name`
 - string `provider_name`
 - int `packages`
 - int `systems`
 - string `arch_name`

7.5. METHOD: LISTRETIREDCHANNELS

Description

List all retired software channels. These are channels the user's organization is entitled but are no longer supported because they have reached their end-of-life date.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - channel info
 - int `id`

- `string label`
- `string name`
- `string provider_name`
- `int packages`
- `int systems`
- `string arch_name`

7.6. METHOD: LISTSHAREDCHANNELS

Description

List all software channels the user's organization may share.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - channel info`
 - `int id`
 - `string label`
 - `string name`
 - `string provider_name`
 - `int packages`
 - `int systems`
 - `string arch_name`

7.7. METHOD: LISTSOFTWARECHANNELS

Description

List all visible software channels.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - channel
 - string `label`
 - string `name`
 - string `parent_label`
 - string `end_of_life`
 - string `arch`

7.8. METHOD: LISTVENDORCHANNELS

Description

Lists all the vendor software channels the user's organization is entitled.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - channel info
 - int `id`
 - string `label`
 - string `name`
 - string `provider_name`
 - int `packages`
 - int `systems`
 - string `arch_name`

CHAPTER 8. NAMESPACE: CHANNEL.ACCESS

Provides methods to retrieve and alter channel access restrictions.

Namespace: `channel.access`

8.1. METHOD: DISABLEUSERRESTRICTIONS

Description

Disable user restrictions for the given channel. If disabled, all users within the organization may subscribe to the channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - label of the channel

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

8.2. METHOD: ENABLEUSERRESTRICTIONS

Description

Enable user restrictions for the given channel. If enabled, only selected users within the organization may subscribe to the channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - label of the channel

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

8.3. METHOD: GETORGSHARING

Description

Get organization sharing access control.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - label of the channel

Returns

The following return values are available for this method:

- `string` - The access value (one of the following: `public`, `private`, or `protected`).

8.4. METHOD: SETORGSHARING

Description

Set organization sharing access control.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - label of the channel
- `string access` - Access one of the following: `public`, `private`, or `protected`.

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

CHAPTER 9. NAMESPACE: CHANNEL.ORG

Provides methods to retrieve and alter organization trust relationships for a channel.

Namespace: `channel.org`

9.1. METHOD: DISABLEACCESS

Description

Disable access to the channel for the given organization.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - label of the channel
- int `orgId` - ID of organization with disabled access

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

9.2. METHOD: ENABLEACCESS

Description

Enable access to the channel for the given organization.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - label of the channel
- int `orgId` - ID of organization granted access

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

9.3. METHOD: LIST

Description

List the organizations associated with the given channel that may be trusted.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - label of the channel

Returns

The following return values are available for this method:

- array:
 - struct - org
 - int `org_id`
 - string `org_name`
 - boolean `access_enabled`

CHAPTER 10. NAMESPACE: CHANNEL.SOFTWARE

Provides methods to access and modify many aspects of a channel.

Namespace: `channel.software`

10.1. METHOD: ADDPACKAGES

Description

Adds a given list of packages to the given channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - target channel
- array:
 - int - `packageId` - ID of a package to add to the channel

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.2. METHOD: ADDREPOFILTER

Description

Adds a filter for a given repo.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - repository label
- struct - `filter_map`
 - string `filter` - string to filter on
 - string `flag` - plus (+) for include, minus (-) for exclude

Returns

The following return values are available for this method:

- int `sort` - order for new filter

10.3. METHOD: ASSOCIATEREPO

Description

Associates a repository with a channel

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel label
- string `repoLabel` - repository label

Returns

The following return values are available for this method:

- struct - channel
- int `id`
- string `name`
- string `label`
- string `arch_name`
- string `summary`
- string `description`
- string `checksum_label`
- `dateTime.iso8601` `last_modified`
- string `maintainer_name`
- string `maintainer_email`
- string `maintainer_phone`
- string `support_policy`
- string `gpg_key_url`
- string `gpg_key_id`
- string `gpg_key_fp`
- `dateTime.iso8601` `yumrepo_last_sync` (optional)
- string `end_of_life`
- string `parent_channel_label`
- string `clone_original`

- array:
 - struct - contentSources
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.4. METHOD: AVAILABLEENTITLEMENTS

Description

Returns the number of available subscriptions for the given channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query

Returns

The following return values are available for this method:

- int `number` of available subscriptions for the given channel

10.5. METHOD: CLEARREPOFILTERS

Description

Removes the filters for a repo.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - repository label

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.6. METHOD: CLONE

Description

Clone a channel. If `arch_label` is omitted, the architecture label of the original channel is used. If `parent_label` is omitted, the clone becomes a base channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `original_label`
- struct - channel details
 - string `name`
 - string `label`
 - string `summary`
 - string `parent_label` (optional)
 - string `arch_label` (optional)
 - string `gpg_key_url` (optional) - `gpg_url` can be used too
 - string `gpg_key_id` - (optional), `gpg_id` can be used too
 - string `gpg_key_fp` - (optional), `gpg_fingerprint` can be used too
 - string `description` - (optional)
- boolean `original_state`

Returns

The following return values are available for this method:

- int - the cloned channel ID

10.7. METHOD: CREATE

Description

Creates a software channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - label of the new channel
- string `name` - name of the new channel
- string `summary` - summary of the channel
- string `archLabel` - the label of the architecture for the channel.
 - `channel-ia32` - For 32 bit channel architecture

- `channel-ia64` - For 64 bit channel architecture
- `channel-sparc` - For Sparc channel architecture
- `channel-alpha` - For Alpha channel architecture
- `channel-s390` - For s390 channel architecture
- `channel-s390x` - For s390x channel architecture
- `channel-iSeries` - For i-Series channel architecture
- `channel-pSeries` - For p-Series channel architecture
- `channel-x86_64` - For x86_64 channel architecture
- `channel-ppc` - For PPC channel architecture
- `channel-sparc-sun-solaris` - For Sparc Solaris channel architecture
- `channel-i386-sun-solaris` - For i386 Solaris channel architecture
- `string parentLabel` - label of the parent of this channel or an empty string if it does not have one
- `string checksumType` - checksum type for this channel, used for yum repository metadata generation
 - `sha1` - Offers widest compatibility with clients
 - `sha256` - Offers highest security, but is compatible only with newer clients e.g. Red Hat Enterprise Linux 6 and newer.
- `struct - gpgKey`
 - `string url` - GPG key URL
 - `string id` - GPG key ID
 - `string fingerprint` - GPG key Fingerprint

Returns

The following return values are available for this method:

- `int - 1` if the creation operation succeeded, 0 otherwise

Available since: 10.9

10.8. METHOD: CREATE

Description

Creates a software channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - label of the new channel
- string `name` - name of the new channel
- string `summary` - summary of the channel
- string `archLabel` - the label of the architecture for the channel.
 - `channel-ia32` - For 32 bit channel architecture
 - `channel-ia64` - For 64 bit channel architecture
 - `channel-sparc` - For Sparc channel architecture
 - `channel-alpha` - For Alpha channel architecture
 - `channel-s390` - For s390 channel architecture
 - `channel-s390x` - For s390x channel architecture
 - `channel-iSeries` - For i-Series channel architecture
 - `channel-pSeries` - For p-Series channel architecture
 - `channel-x86_64` - For x86_64 channel architecture
 - `channel-ppc` - For PPC channel architecture
 - `channel-sparc-sun-solaris` - For Sparc Solaris channel architecture
 - `channel-i386-sun-solaris` - For i386 Solaris channel architecture
- string `parentLabel` - label of the parent of this channel or an empty string if it does not have one
- string `checksumType` - checksum type for this channel, used for yum repository metadata generation
 - `sha1` - Offers widest compatibility with clients
 - `sha256` - Offers highest security, but is compatible only with newer clients e.g. Red Hat Enterprise Linux 6 and newer.

Returns

The following return values are available for this method:

- int - 1 if the creation operation succeeded, 0 otherwise

Available since: 10.9

10.9. METHOD: CREATE

Description

Creates a software channel

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - label of the new channel
- string `name` - name of the new channel
- string `summary` - summary of the channel
- string `archLabel` - the label of the architecture for the channel.
 - `channel-ia32` - For 32 bit channel architecture
 - `channel-ia64` - For 64 bit channel architecture
 - `channel-sparc` - For Sparc channel architecture
 - `channel-alpha` - For Alpha channel architecture
 - `channel-s390` - For s390 channel architecture
 - `channel-s390x` - For s390x channel architecture
 - `channel-iSeries` - For i-Series channel architecture
 - `channel-pSeries` - For p-Series channel architecture
 - `channel-x86_64` - For x86_64 channel architecture
 - `channel-ppc` - For PPC channel architecture
 - `channel-sparc-sun-solaris` - For Sparc Solaris channel architecture
 - `channel-i386-sun-solaris` - For i386 Solaris channel architecture
- string `parentLabel` - label of the parent of this channel or an empty string if it does not have one

Returns

The following return values are available for this method:

- int - 1 if the creation operation succeeded, 0 otherwise

10.10. METHOD: CREATEREPO

Description

Creates a repository

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - repository label
- string `type` - repository type (only YUM is supported)
- string `url` - repository url

Returns

The following return values are available for this method:

- struct - channel
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.11. METHOD: DELETE

Description

Deletes a custom software channel

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to delete

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.12. METHOD: DISASSOCIATEREPO

Description

Disassociates a repository from a channel

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel label

- string `repoLabel` - repository label

Returns

The following return values are available for this method:

- struct - channel
- int `id`
- string `name`
- string `label`
- string `arch_name`
- string `summary`
- string `description`
- string `checksum_label`
- `dateTime.iso8601` `last_modified`
- string `maintainer_name`
- string `maintainer_email`
- string `maintainer_phone`
- string `support_policy`
- string `gpg_key_url`
- string `gpg_key_id`
- string `gpg_key_fp`
- `dateTime.iso8601` `yumrepo_last_sync` (optional)
- string `end_of_life`
- string `parent_channel_label`
- string `clone_original`
- array:
 - struct - `contentSources`
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.13. METHOD: GETCHANNELLASTBUILDID

Description

Returns the last build date of the `repomd.xml` file for the given channel as a localised string.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - id of desired channel

Returns

The following return values are available for this method:

- the last build date of the `repomd.xml` file as a localised string

10.14. METHOD: GETDETAILS

Description

Returns details of the given channel as a map.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query

Returns

The following return values are available for this method:

- struct - channel
- int `id`
- string `name`
- string `label`
- string `arch_name`
- string `summary`
- string `description`
- string `checksum_label`
- `dateTime.iso8601` `last_modified`
- string `maintainer_name`
- string `maintainer_email`

- string `maintainer_phone`
- string `support_policy`
- string `gpg_key_url`
- string `gpg_key_id`
- string `gpg_key_fp`
- `dateTime.iso8601 yumrepo_last_sync` (optional)
- string `end_of_life`
- string `parent_channel_label`
- string `clone_original`
- array:
 - struct - `contentSources`
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.15. METHOD: GETDETAILS

Description

Returns details of the given channel as a map.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - channel to query

Returns

The following return values are available for this method:

- struct - channel
- int `id`
- string `name`
- string `label`
- string `arch_name`

- `string summary`
- `string description`
- `string checksum_label`
- `dateTime.iso8601 last_modified`
- `string maintainer_name`
- `string maintainer_email`
- `string maintainer_phone`
- `string support_policy`
- `string gpg_key_url`
- `string gpg_key_id`
- `string gpg_key_fp`
- `dateTime.iso8601 yumrepo_last_sync` (optional)
- `string end_of_life`
- `string parent_channel_label`
- `string clone_original`
- `array:`
 - `struct - contentSources`
 - `int id`
 - `string label`
 - `string sourceUrl`
 - `string type`

10.16. METHOD: GETREPODETAILS

Description

Returns details of the given repository.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string repoLabel` - repository to query

Returns

The following return values are available for this method:

- struct - channel
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.17. METHOD: GETREPODETAILS

Description

Returns details of the given repo

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `repoLabel` - repo to query

Returns

The following return values are available for this method:

- struct - channel
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.18. METHOD: GETREPOSYNCCRONEXPRESSION

Description

Returns repository synchronization cron expression.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel label

Returns

The following return values are available for this method:

- string `quartzexpression`

10.19. METHOD: ISGLOBALLYSUBSCRIBABLE

Description

Returns whether any user in the organization may subscribe to the channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query

Returns

The following return values are available for this method:

- int - 1 if true, 0 otherwise

10.20. METHOD: ISUSERMANAGEABLE

Description

Returns whether the given user may manage the channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - label of the channel
- string `login` - login of the target user

Returns

The following return values are available for this method:

- int - 1 if manageable, 0 if not

10.21. METHOD: ISUSERSUBSCRIBABLE

Description

Returns whether any user may subscribe to the channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - label of the channel
- string `login` - login of the target user

Returns

The following return values are available for this method:

- int - 1 if subscribable, 0 if not

10.22. METHOD: LISTALLPACKAGES

Description

Lists all packages in the channel, regardless of package version, between given dates.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query
- `dateTime.iso8601` `startDate`
- `dateTime.iso8601` `endDate`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - string `name`
 - string `version`
 - string `release`
 - string `epoch`
 - string `checksum`
 - string `checksum_type`
 - int `id`
 - string `arch_label`
 - string `last_modified_date`
 - string `last_modified` (deprecated)

10.23. METHOD: LISTALLPACKAGES

Description

Lists all packages in the channel, regardless of version, where the last modified date is greater than the given date.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query
- `dateTime.iso8601` `startDate`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - string `name`
 - string `version`
 - string `release`
 - string `epoch`
 - string `checksum`
 - string `checksum_type`
 - int `id`
 - string `arch_label`
 - string `last_modified_date`
 - string `last_modified` (deprecated)

10.24. METHOD: LISTALLPACKAGES

Description

Lists all packages in the channel, regardless of the package version.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query

Returns

The following return values are available for this method:

- array:
 - struct - package

- `string name`
- `string version`
- `string release`
- `string epoch`
- `string checksum`
- `string checksum_type`
- `int id`
- `string arch_label`
- `string last_modified_date`
- `string last_modified` (deprecated)

10.25. METHOD: LISTALLPACKAGES

Description

Lists all packages in the channel, regardless of package version, between given dates. Example Date: '2008-08-20 08:00:00'



NOTE

This method is deprecated. Use the `listAllPackages(string sessionKey, string channelLabel, dateTime.iso8601 startDate, dateTime.iso8601 endDate)` method.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - channel to query
- `string startDate`
- `string endDate`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - `string name`
 - `string version`

- `string release`
- `string epoch`
- `string checksum`
- `string checksum_type`
- `int id`
- `string arch_label`
- `string last_modified_date`
- `string last_modified` (deprecated)

10.26. METHOD: LISTALLPACKAGES

Description

Lists all packages in the channel, regardless of version, where the last modified date is greater than the given date. Example Date: '2008-08-20 08:00:00'



NOTE

This method is deprecated. Use the `listAllPackages(string sessionKey, string channelLabel, dateTime.iso8601 startDate)` method.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - channel to query
- `string startDate`

Returns

The following return values are available for this method:

- `array:`
 - `struct - package`
 - `string name`
 - `string version`
 - `string release`
 - `string epoch`
 - `string checksum`

- `string checksum_type`
- `int id`
- `string arch_label`
- `string last_modified_date`
- `string last_modified` (deprecated)

10.27. METHOD: LISTALLPACKAGESBYDATE

Description

Lists all packages in the channel, regardless of the package version, between the given dates. Example Date: '2008-08-20 08:00:00'



NOTE

This method is deprecated. Use the `listAllPackages(string sessionKey, string channelLabel, dateTime.iso8601 startDate, dateTime.iso8601 endDate)` method.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - channel to query
- `string startDate`
- `string endDate`

Returns

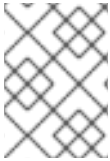
The following return values are available for this method:

- array:
 - struct - package
 - `string name`
 - `string version`
 - `string release`
 - `string epoch`
 - `string id`
 - `string arch_label`
 - `string last_modified`

10.28. METHOD: LISTALLPACKAGESBYDATE

Description

Lists all packages in the channel, regardless of the package version, where the last modified date is greater than the given date. Example Date: '2008-08-20 08:00:00'



NOTE

This method is deprecated. Use the `listAllPackages(string sessionKey, string channelLabel, dateTime.iso8601 startDate)` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query
- string `startDate`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - string `name`
 - string `version`
 - string `release`
 - string `epoch`
 - string `id`
 - string `arch_label`
 - string `last_modified`

10.29. METHOD: LISTALLPACKAGESBYDATE

Description

Lists all packages in the channel, regardless of the package version



NOTE

This method is deprecated. Use the `listAllPackages(string sessionKey, string channelLabel)` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query

Returns

The following return values are available for this method:

- array:
 - struct - package
 - string `name`
 - string `version`
 - string `release`
 - string `epoch`
 - string `id`
 - string `arch_label`
 - string `last_modified`

10.30. METHOD: LISTARCHES

Description

Lists the potential software channel architectures for creation.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - channel arch
 - string `name`
 - string `label`

10.31. METHOD: LISTCHANNELREPOS

Description

Lists associated repositories with the given channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel label

Returns

The following return values are available for this method:

- array:
 - struct - channel
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.32. METHOD: LISTCHILDREN

Description

List the children of a channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - the label of the channel

Returns

The following return values are available for this method:

- array:
 - struct - channel
 - int `id`
 - string `name`
 - string `label`
 - string `arch_name`
 - string `summary`
 - string `description`

- string `checksum_label`
- `dateTime.iso8601 last_modified`
- string `maintainer_name`
- string `maintainer_email`
- string `maintainer_phone`
- string `support_policy`
- string `gpg_key_url`
- string `gpg_key_id`
- string `gpg_key_fp`
- `dateTime.iso8601 yumrepo_last_sync` (optional)
- string `end_of_life`
- string `parent_channel_label`
- string `clone_original`
- array:
 - struct - `contentSources`
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.33. METHOD: LISTERRATA

Description

List the errata applicable to a channel after given start date.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query
- `dateTime.iso8601 startDate`

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `update_date` - Erratum update date
 - string `advisory_synopsis` - Summary of the erratum.
 - string `advisory_type` - Type label, such as "Security", "Bug Fix", or "Enhancement"
 - string `advisory_name` - Name, such as RHSA

10.34. METHOD: LISTERRATA

Description

List the errata applicable to a channel between a start date and end date.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - channel to query
- `dateTime.iso8601` `startDate`
- `dateTime.iso8601` `endDate`

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `update_date` - Erratum update date
 - string `advisory_synopsis` - Summary of the erratum.
 - string `advisory_type` - Type label, such as "Security", "Bug Fix", or "Enhancement"
 - string `advisory_name` - Name, such as RHSA

10.35. METHOD: LISTERRATA

Description

List the errata applicable to a channel.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - Channel to query

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - `int id` - Errata Id
 - `string advisory_synopsis` - Summary of the erratum.
 - `string advisory_type` - Type label, such as "Security", "Bug Fix", or "Enhancement"
 - `string advisory_name` - Name, such as RHSA
 - `string advisory` - Name of the advisory (deprecated)
 - `string issue_date` - Date format, which follows YYYY-MM-DD HH24:MI:SS (deprecated)
 - `string update_date` - date format, which follows YYYY-MM-DD HH24:MI:SS (deprecated)
 - `string synopsis` (deprecated)
 - `string last_modified_date` - date format, which follows YYYY-MM-DD HH24:MI:SS (deprecated)

10.36. METHOD: LISTERRATA

Description

List the errata applicable to a channel after given startDate



NOTE

The method is deprecated. Use the `listErrata(string sessionKey, string channelLabel, dateTime.iso8601 startDate)` method.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - Channel to query

- `string startDate`

Returns

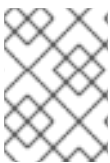
The following return values are available for this method:

- `array:`
 - `struct - errata`
 - `string advisory` - Name of the advisory
 - `string issue_date` - Date format, which follows YYYY-MM-DD HH24:MI:SS
 - `string update_date` - date format, which follows YYYY-MM-DD HH24:MI:SS
 - `string synopsis`
 - `string advisory_type`
 - `string last_modified_date` - date format, which follows YYYY-MM-DD HH24:MI:SS

10.37. METHOD: LISTERRATA

Description

List the errata applicable to a channel between a start date and end date.



NOTE

This method is deprecated. Use the `listErrata(string sessionKey, string channelLabel, dateTime.iso8601 startDate, dateTime.iso8601)` method.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - Channel to query
- `string startDate`
- `string startDate`

Returns

The following return values are available for this method:

- `array:`
 - `struct - errata`
 - `string advisory` - Name of the advisory
 - `string issue_date` - Date format, which follows YYYY-MM-DD HH24:MI:SS

- string `update_date` - Date format, which follows YYYY-MM-DD HH24:MI:SS
- string `synopsis`
- string `advisory_type`
- string `last_modified_date` - Date format, which follows YYYY-MM-DD HH24:MI:SS

10.38. METHOD: LISTERRATABYTYPE

Description

List the errata of a specific type applicable to a channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Channel to query
- string `advisoryType` - Type of advisory; one of the following: "Security Advisory", "Product Enhancement Advisory", or "Bug Fix Advisory".

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - string `advisory` - Name of the advisory
 - string `issue_date` - Date format, which follows YYYY-MM-DD HH24:MI:SS
 - string `update_date` - Date format, which follows YYYY-MM-DD HH24:MI:SS
 - string `synopsis`
 - string `advisory_type`
 - string `last_modified_date` - Date format, which follows YYYY-MM-DD HH24:MI:SS

10.39. METHOD: LISTLATESTPACKAGES

Description

Lists the packages with the latest version, including release and epoch, for the given channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Channel to query

Returns

The following return values are available for this method:

- array:
 - struct - package
 - string `name`
 - string `version`
 - string `release`
 - string `epoch`
 - int `id`
 - string `arch_label`

10.40. METHOD: LISTPACKAGESWITHOUTCHANNEL

Description

Lists all packages not associated with a channel. Typically these are custom packages.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - string `name`
 - string `version`
 - string `release`
 - string `epoch`
 - int `id`
 - string `arch_label`
 - string `path` - The path on the file system the package resides
 - string `provider` - The provider of the package, determined by its GPG key
 - `dateTime.iso8601 last_modified`

10.41. METHOD: LISTREPOFILTERS

Description

Lists the filters for a repo.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - Repository label

Returns

The following return values are available for this method:

- array:
 - struct - filter
 - int `sortOrder`
 - string `filter`
 - string `flag`

10.42. METHOD: LISTSUBSCRIBEDSYSTEMS

Description

Returns list of subscribed systems for the given channel label.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Channel to query

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`

10.43. METHOD: LISTSYSTEMCHANNELS

Description

Returns a list of channels that a system is subscribed to for the given system ID.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - channel
 - string `label`
 - string `name`

10.44. METHOD: LISTUSERREPOS

Description

Returns a list of `ContentSource` (repositories) the user can see.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - map
 - long `id` - ID of the repo
 - string `label` - Label of the repo
 - string `sourceUrl` - URL of the repo

10.45. METHOD: MERGEERRATA

Description

Merges all errata from one channel into another.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `mergeFromLabel` - The label of the channel to pull errata

- string `mergeToLabel` - The label to push errata

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `advisory_type` - Type of advisory
 - string `advisory_name` - Name of the advisory
 - string `advisory_synopsis` - Summary of the erratum

10.46. METHOD: MERGEERRATA

Description

Merges all errata from one channel into another based upon a given start and end date.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `mergeFromLabel` - The label of the channel to pull errata
- string `mergeToLabel` - The label to push the errata
- string `startDate`
- string `endDate`

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - int `id` - Errata Id
 - string `date` - Erratum creation date
 - string `advisory_type` - Type of advisory
 - string `advisory_name` - Name of advisory
 - string `advisory_synopsis` - Summary of erratum

10.47. METHOD: MERGEERRATA

Description

Merges a list of errata from one channel into another.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `mergeFromLabel` - The label of the channel to pull errata
- string `mergeToLabel` - The label to push errata
- array:
 - string - `advisory` - The advisory name of the errata to merge

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `advisory_type` - Type of advisory
 - string `advisory_name` - Name of advisory
 - string `advisory_synopsis` - Summary of erratum

10.48. METHOD: MERGEPACKAGES

Description

Merges all packages from one channel into another.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `mergeFromLabel` - The label of the channel to pull packages
- string `mergeToLabel` - The label to push the packages

Returns

The following return values are available for this method:

- array:

- struct - package
 - string name
 - string version
 - string release
 - string epoch
 - int id
 - string arch_label
 - string path - The path on the file system the package resides
 - string provider - The provider of the package, determined by its GPG key
 - dateTime.iso8601 last_modified

10.49. METHOD: REGENERATENEEEDEDCACHE

Description

Completely clear and regenerate the needed errata and package cache for all systems subscribed to the specified channel. Only use this method if you believe your cache is incorrect for all systems in a given channel.

Parameters

The following parameters are available for this method:

- string sessionKey
- string channelLabel - The label of the channel

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.50. METHOD: REGENERATENEEEDEDCACHE

Description

Completely clear and regenerate the needed errata and package cache for all systems subscribed. Only a Satellite Administrator may perform this action.

Parameters

The following parameters are available for this method:

- string sessionKey

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.51. METHOD: REGENERATEYUMCACHE

Description

Regenerate YUM cache for the specified channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - The label of the channel

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.52. METHOD: REMOVEERRATA

Description

Removes a given list of errata from the given channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Target channel
- array:
 - string - `advisoryName` - Name of an erratum to remove
- boolean `removePackages` - True to remove packages from the channel

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.53. METHOD: REMOVEPACKAGES

Description

Removes a given list of packages from the given channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`

- string `channelLabel` - Target channel.
- array:
 - int - `packageId` - ID of a package to remove from the channel.

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.54. METHOD: REMOVEREPO

Description

Removes a repository.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- long `id` - ID of repository to remove

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.55. METHOD: REMOVEREPO

Description

Removes a repository

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - Label of repo to be removed

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.56. METHOD: REMOVEREPOFILTER

Description

Removes a filter for a given repo.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - Repository label
- struct - `filter_map`
 - string `filter` - String to filter on
 - string `flag` - plus (+) for include, minus (-) for exclude

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.57. METHOD: SETCONTACTDETAILS

Description

Set contact and support information for given channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Label of the channel
- string `maintainerName` - Name of the channel maintainer
- string `maintainerEmail` - Email of the channel maintainer
- string `maintainerPhone` - Phone number of the channel maintainer
- string `supportPolicy` - Channel support policy

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.58. METHOD: SETDETAILS

Description

Allows to modify channel attributes

Parameters

The following parameters are available for this method:

- string `sessionKey`

- **int channelId** - channel id
- **struct - channel_map**
 - **string checksum_label** - New channel repository checksum label (optional)
 - **string name** - New channel name (optional)
 - **string summary** - New channel summary (optional)
 - **string description** - New channel description (optional)
 - **string maintainer_name** - New channel maintainer name (optional)
 - **string maintainer_email** - New channel email address (optional)
 - **string maintainer_phone** - New channel phone number (optional)
 - **string gpg_key_url** - New channel GPG key URL (optional)
 - **string gpg_key_id** - New channel GPG key ID (optional)
 - **string gpg_key_fp** - New channel GPG key fingerprint (optional)

Returns

The following return values are available for this method:

- **int - 1** on success, exception thrown otherwise.

10.59. METHOD: SETGLOBALLYSUBSCRIBABLE

Description

Set globally subscribable attribute for given channel.

Parameters

The following parameters are available for this method:

- **string sessionKey**
- **string channelLabel** - Label of the channel
- **boolean subscribable** - **true** if the channel is to be globally subscribable. **false** otherwise.

Returns

The following return values are available for this method:

- **int - 1** on success, exception thrown otherwise.

10.60. METHOD: SETREPOFILTERS

Description

Replaces the existing set of filters for a given repository. Filters are ranked by their order in the array.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - Repository label
- array:
 - struct - `filter_map`
 - string `filter` - String to filter on
 - string `flag` - Plus (+) for include, minus (-) for exclude

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.61. METHOD: SETSYSTEMCHANNELS

Description

Change a system's subscribed channels to the list of channels passed.



NOTE

This method is deprecated. The `system.setBaseChannel(string sessionKey, int serverId, string channelLabel)` and `system.setChildChannels(string sessionKey, int serverId, array[string channelLabel])` methods now replace it.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- array:
 - string - `channelLabel` - Labels of the channels to subscribe the system.

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.62. METHOD: SETUSERMANAGEABLE

Description

Set the manageable flag for a given channel and user. If value is set to `true`, this method gives the user 'manage' permissions for the channel. Otherwise, the privilege is revoked.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Label of the channel
- string `login` - Login of the target user
- boolean `value` - Value of the flag to set

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.63. METHOD: SETUSERSUBSCRIBABLE

Description

Set the subscribable flag for a given channel and user. If value is set to `true`, this method gives the user 'subscribe' permissions to the channel. Otherwise, the privilege is revoked.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Label of the channel
- string `login` - Login of the target user
- boolean `value` - Value of the flag to set

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.64. METHOD: SUBSCRIBESYSTEM

Description

Subscribes a system to a list of channels. If a base channel is included, it is set before setting child channels. When setting child channels, the current child channel subscriptions are cleared. To fully unsubscribe the system from all channels, provide an empty list of channel labels.

**NOTE**

This method is deprecated. The `system.setBaseChannel(string sessionKey, int serverId, string channelLabel)` and `system.setChildChannels(string sessionKey, int serverId, array[string channelLabel])` methods now replace it.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `array:`
 - `string - label` - Channel label to subscribe the system

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

10.65. METHOD: SYNCREPO**Description**

Trigger immediate repository synchronization.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - Channel label

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

10.66. METHOD: SYNCREPO**Description**

Schedule periodic repository synchronization.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - Channel label

- string `cron` expression - If empty, all periodic schedules are disabled

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

10.67. METHOD: UPDATEREPO

Description

Updates a `ContentSource` (repository).

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - Repository ID
- string `label` - New repository label
- string `url` - New repository URL

Returns

The following return values are available for this method:

- struct - channel
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.68. METHOD: UPDATEREPOLABEL

Description

Updates the repository label.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - Repository ID
- string `label` - New repository label

Returns

The following return values are available for this method:

- struct - channel
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.69. METHOD: UPDATEREPOURL

Description

Updates the repository source URL.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - Repository ID
- string `url` - Repository URL

Returns

The following return values are available for this method:

- struct - channel
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

10.70. METHOD: UPDATEREPOURL

Description

Updates repository source URL

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - Repository label
- string `url` - Repository URL

Returns

The following return values are available for this method:

- **struct - channel**
 - **int id**
 - **string label**
 - **string sourceUrl**
 - **string type**

CHAPTER 11. NAMESPACE: CONFIGCHANNEL

Provides methods to access and modify many aspects of configuration channels.

Namespace: `configchannel`

11.1. METHOD: CHANNELEXISTS

Description

Check for the existence of the config channel provided.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Channel to check for

Returns

The following return values are available for this method:

- 1 if exists, 0 otherwise.

11.2. METHOD: CREATE

Description

Create a new global config channel. Caller must be at least a config administrator or an organization administrator.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel`
- string `channelName`
- string `channelDescription`

Returns

The following return values are available for this method:

- struct - Configuration Channel information
 - int `id`
 - int `orgId`
 - string `label`
 - string `name`

- **string description**
- **struct configChannelType**
- **struct - Configuration Channel Type information**
 - **int id**
 - **string label**
 - **string name**
 - **int priority**

11.3. METHOD: CREATEORUPDATEPATH

Description

Create a new file or directory with the given path, or update an existing path.

Parameters

The following parameters are available for this method:

- **string sessionKey**
- **string configChannelLabel**
- **string path**
- **boolean isDir** - **true** if the path is a directory, **false** if it is a file.
- **struct - path info**
 - **string contents** - Contents of the file in text or base64 encoded if binary (only for non-directories)
 - **boolean contents_enc64** - Identifies base64 encoded content (disabled by default, only for non-directories)
 - **string owner** - Owner of the file or directory.
 - **string group** - Group name of the file or directory.
 - **string permissions** - Octal file or directory permissions e.g. 644
 - **string selinux_ctx** - SELinux security context (optional)
 - **string macro-start-delimiter** - Config file macro start delimiter; use null or empty string to accept the default (only for non-directories)
 - **string macro-end-delimiter** - Config file macro end delimiter; use null or empty string to accept the default (only for non-directories)
 - **int revision** - Next revision number; auto increment for null

- boolean **binary** - Mark the binary content; if **true**, base64 encoded content is expected (only for non-directories)

Returns

The following return values are available for this method:

- **struct** - Configuration Revision information
 - string **type**
 - **file**
 - **directory**
 - **symlink**
 - string **path** - File Path
 - string **target_path** - Symbolic link Target File Path; present for Symbolic links only.
 - string **channel** - Channel Name
 - string **contents** - File contents (base64 encoded according to the **contents_enc64** attribute)
 - boolean **contents_enc64** - Identifies base64 encoded content
 - int **revision** - File Revision
 - **dateTime.iso8601** **creation** - Creation date
 - **dateTime.iso8601** **modified** - Last modified date
 - string **owner** - File Owner; present for files or directories only
 - string **group** - File Group; present for files or directories only
 - int **permissions** - File Permissions; present for files or directories only (deprecated)
 - string **permissions_mode** - File Permissions; Present for files or directories only
 - string **selinux_ctx** - SELinux context (optional)
 - boolean **binary** - **true** or **false**; present for files only.
 - string **md5** - File's MD5 signature; present for files only
 - string **macro-start-delimiter** - Macro start delimiter for a config file; present for text files only.
 - string **macro-end-delimiter** - Macro end delimiter for a config file; present for text files only.

Available since:10.2

11.4. METHOD: CREATEORUPDATESYMLINK

Description

Create a new symbolic link with the given path, or update an existing path.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `configChannelLabel`
- string `path`
- struct - path info
 - string `target_path` - The target path for the symbolic link
 - string `selinux_ctx` - SELinux security context (optional)
 - int `revision` - Next revision number; skip this field for automatic revision number assignment

Returns

The following return values are available for this method:

- struct - Configuration Revision information
 - string `type`
 - file
 - directory
 - symlink
 - string `path` - File Path
 - string `target_path` - Symbolic link Target File Path; present for Symbolic links only
 - string `channel` - Channel Name
 - string `contents` - File contents; base64 encoded according to the `contents_enc64` attribute
 - boolean `contents_enc64` - Identifies base64 encoded content
 - int `revision` - File Revision
 - `dateTime.iso8601` `creation` - Creation date
 - `dateTime.iso8601` `modified` - Last modified date
 - string `owner` - File Owner; present for files or directories only

- string **group** - File Group; present for files or directories only
- int **permissions** - File Permissions; present for files or directories only (deprecated)
- string **permissions_mode** - File Permissions; present for files or directories only
- string **selinux_ctx** - SELinux context (optional)
- boolean **binary** - **true** or **false**; present for files only.
- string **md5** - File's MD5 signature; present for files only.
- string **macro-start-delimiter** - Macro start delimiter for a config file; present for text files only.
- string **macro-end-delimiter** - Macro end delimiter for a config file; present for text files only.

Available since: 10.2

11.5. METHOD: DELETECHANNELS

Description

Delete a list of global config channels. Caller must be a config admin.

Parameters

The following parameters are available for this method:

- string **sessionKey**
- array:
 - string **configuration** - channel labels to delete

Returns

The following return values are available for this method:

- int - **1** on success, exception thrown otherwise

11.6. METHOD: DELETEFILEREVISIONS

Description

Delete specified revisions of a given configuration file.

Parameters

The following parameters are available for this method:

- string **sessionKey**
- string **channelLabel** - Label of config channel to look up
- string **filePath** - Configuration file path

- array:
 - int - List of revisions to delete

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

11.7. METHOD: DELETEFILES

Description

Remove file paths from a global channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Channel to remove files
- array:
 - string `file` - Paths to remove

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

11.8. METHOD: DEPLOYALLSYSTEMS

Description

Schedule an immediate configuration deployment for all systems subscribed to a particular configuration channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - The configuration channel's label

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

11.9. METHOD: DEPLOYALLSYSTEMS

Description

Schedule a configuration deployment for all systems subscribed to a particular configuration channel.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - The configuration channel's label
- `dateTime.iso8601 date` - The date to schedule the action

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise.

11.10. METHOD: GETDETAILS

Description

Look up configuration channel details.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel`

Returns

The following return values are available for this method:

- `struct` - Configuration Channel information
 - `int id`
 - `int orgId`
 - `string label`
 - `string name`
 - `string description`
 - `struct configChannelType`
 - `struct` - Configuration Channel Type information
 - `int id`
 - `string label`
 - `string name`
 - `int priority`

11.11. METHOD: GETDETAILS

Description

Look up configuration channel details.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `channelId`

Returns

The following return values are available for this method:

- struct - Configuration Channel information
 - int `id`
 - int `orgId`
 - string `label`
 - string `name`
 - string `description`
 - struct `configChannelType`
 - struct - Configuration Channel Type information
 - int `id`
 - string `label`
 - string `name`
 - int `priority`

11.12. METHOD: GETENCODEDFILEREVISION

Description

Get revision of the specified configuration file and transmit the contents as base64 encoded.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `configChannelLabel` - Label of config channel to look up
- string `filePath` - Config file path to examine

- **int revision** - Config file revision to examine

Returns

The following return values are available for this method:

- **struct** - Configuration Revision information
 - **string type**
 - **file**
 - **directory**
 - **symlink**
 - **string path** - File Path
 - **string target_path** - Symbolic link Target File Path. Present for Symbolic links only.
 - **string channel** - Channel Name
 - **string contents** - File contents (base64 encoded according to the **contents_enc64** attribute)
 - **boolean contents_enc64** - Identifies base64 encoded content
 - **int revision** - File Revision
 - **dateTime.iso8601 creation** - Creation Date
 - **dateTime.iso8601 modified** - Last Modified Date
 - **string owner** - File Owner. Present for files or directories only.
 - **string group** - File Group. Present for files or directories only.
 - **int permissions** - File Permissions (Deprecated). Present for files or directories only.
 - **string permissions_mode** - File Permissions. Present for files or directories only.
 - **string selinux_ctx** - SELinux Context (optional).
 - **boolean binary** - **true** or **false**; present for files only
 - **string md5** - File's md5 signature. Present for files only.
 - **string macro-start-delimiter** - Macro start delimiter for a config file. Present for text files only.
 - **string macro-end-delimiter** - Macro end delimiter for a config file. Present for text files only.

11.13. METHOD: GETFILEREVISION

Description

Get revision of the specified configuration file and transmit the contents as plain text.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `configChannelLabel` - Label of config channel to look up
- string `filePath` - Config file path to examine
- int `revision` - Config file revision to examine

Returns

The following return values are available for this method:

- struct - Configuration Revision information
 - string `type`
 - file
 - directory
 - symlink
 - string `path` - File path
 - string `target_path` - Symbolic link Target File Path. Present for Symbolic links only.
 - string `channel` - Channel name
 - string `contents` - File contents (base64 encoded according to the `contents_enc64` attribute)
 - boolean `contents_enc64` - Identifies base64 encoded content
 - int `revision` - File revision
 - `dateTime.iso8601` `creation` - Creation date
 - `dateTime.iso8601` `modified` - Last modified date
 - string `owner` - File owner; present for files or directories only
 - string `group` - File Group; present for files or directories only
 - int `permissions` - File permissions; present for files or directories only. (deprecated)
 - string `permissions_mode` - File permissions; present for files or directories only
 - string `selinux_ctx` - SELinux context (optional)
 - boolean `binary` - `true` or `false`; present for files only
 - string `md5` - File's MD5 signature; present for files only

- string **macro-start-delimiter** - Macro start delimiter for a config file; present for text files only
- string **macro-end-delimiter** - Macro end delimiter for a config file; present for text files only

11.14. METHOD: GETFILEREVISIONS

Description

Get list of revisions for specified config file.

Parameters

The following parameters are available for this method:

- string **sessionKey**
- string **channelLabel** - Label of config channel to look up
- string **filePath** - Config file path to examine

Returns

The following return values are available for this method:

- array:
 - struct - Configuration Revision information
 - string **type**
 - file
 - directory
 - symlink
 - string **path** - File path
 - string **target_path** - Symbolic link target file path; present for symbolic links only.
 - string **channel** - Channel name
 - string **contents** - File contents (base64 encoded according to the `contents_enc64` attribute)
 - boolean **contents_enc64** - Identifies base64 encoded content
 - int **revision** - File revision
 - dateTime.iso8601 **creation** - Creation date
 - dateTime.iso8601 **modified** - Last modified date
 - string **owner** - File owner; present for files or directories only
 - string **group** - File Group; present for files or directories only

- **int permissions** - File permissions; present for files or directories only (deprecated)
- **string permissions_mode** - File permissions; present for files or directories only
- **string selinux_ctx** - SELinux context (optional)
- **boolean binary** - **true** or **false**; present for files only.
- **string md5** - File's MD5 signature; present for files only
- **string macro-start-delimiter** - Macro start delimiter for a config file; present for text files only.
- **string macro-end-delimiter** - Macro end delimiter for a config file; present for text files only.

11.15. METHOD: LISTFILES

Description

Return a list of files in a channel.

Parameters

The following parameters are available for this method:

- **string sessionKey**
- **string channelLabel** - label of config channel to list files on.

Returns

The following return values are available for this method:

- **array:**
 - **struct** - Configuration File information
 - **string type**
 - **file**
 - **directory**
 - **symlink**
 - **string path** - File path
 - **dateTime.iso8601 last_modified** - Last modified date

11.16. METHOD: LISTGLOBALS

Description

List all the global config channels accessible to the logged-in user.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - Configuration Channel information
 - `int id`
 - `int orgId`
 - `string label`
 - `string name`
 - `string description`
 - `string type`
 - `struct configChannelType`
 - struct - Configuration Channel Type information
 - `int id`
 - `string label`
 - `string name`
 - `int priority`

11.17. METHOD: LISTSUBSCRIBEDSYSTEMS

Description

Return a list of systems subscribed to a configuration channel.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - label of config channel to list subscribed systems.

Returns

The following return values are available for this method:

- array:
 - struct - system

- `int id`
- `string name`

11.18. METHOD: LOOKUPCHANNELINFO

Description

Lists details for channels given their labels.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `string configuration` - channel label

Returns

The following return values are available for this method:

- `array:`
 - `struct` - Configuration Channel information
 - `int id`
 - `int orgId`
 - `string label`
 - `string name`
 - `string description`
 - `struct configChannelType`
 - `struct` - Configuration Channel Type information
 - `int id`
 - `string label`
 - `string name`
 - `int priority`

11.19. METHOD: LOOKUPFILEINFO

Description

Returns details about the latest path revisions from a list of paths and a channel.

Parameters

The following parameters are available for this method:

- **string sessionKey**
- **string channelLabel** - Label of config channel to look up
- **array:**
 - **string** - List of paths to examine.

Returns

The following return values are available for this method:

- **array:**
 - **struct** - Configuration Revision information
 - **string type**
 - **file**
 - **directory**
 - **symlink**
 - **string path** - File path
 - **string target_path** - Symbolic link target file path; present for symbolic links only.
 - **string channel** - Channel name
 - **string contents** - File contents (base64 encoded according to the `contents_enc64` attribute)
 - **boolean contents_enc64** - Identifies base64 encoded content
 - **int revision** - File revision
 - **dateTime.iso8601 creation** - Creation date
 - **dateTime.iso8601 modified** - Last modified date
 - **string owner** - File owner; present for files or directories only
 - **string group** - File Group; present for files or directories only
 - **int permissions** - File permissions; present for files or directories only (deprecated)
 - **string permissions_mode** - File permissions; present for files or directories only
 - **string selinux_ctx** - SELinux context (optional)
 - **boolean binary** - `true` or `false`; present for files only.
 - **string md5** - File's MD5 signature; present for files only

- `string macro-start-delimiter` - Macro start delimiter for a config file; present for text files only.
- `string macro-end-delimiter` - Macro end delimiter for a config file; present for text files only.

Available since: 10.2

11.20. METHOD: LOOKUPFILEINFO

Description

Returns details about the latest path revisions given a path, revision number, and a channel.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel` - Label of config channel to look up
- `string path` - Path of the file or directory
- `int revision` - The revision number

Returns

The following return values are available for this method:

- `struct` - Configuration Revision information
 - `string type`
 - `file`
 - `directory`
 - `symlink`
 - `string path` - File path
 - `string target_path` - Symbolic link target file path; present for symbolic links only.
 - `string channel` - Channel name
 - `string contents` - File contents (base64 encoded according to the `contents_enc64` attribute)
 - `boolean contents_enc64` - Identifies base64 encoded content
 - `int revision` - File revision
 - `dateTime.iso8601 creation` - Creation date
 - `dateTime.iso8601 modified` - Last modified date
 - `string owner` - File owner; present for files or directories only

- string **group** - File Group; present for files or directories only
- int **permissions** - File permissions; present for files or directories only (deprecated)
- string **permissions_mode** - File permissions; present for files or directories only
- string **selinux_ctx** - SELinux context (optional)
- boolean **binary** - **true** or **false**; present for files only.
- string **md5** - File's MD5 signature; present for files only
- string **macro-start-delimiter** - Macro start delimiter for a config file; present for text files only.
- string **macro-end-delimiter** - Macro end delimiter for a config file; present for text files only.

Available since: 10.12

11.21. METHOD: SCHEDULEFILECOMPARISONS

Description

Schedule a comparison of the latest revision of a file against the version deployed on a list of systems.

Parameters

The following parameters are available for this method:

- string **sessionKey**
- string **channelLabel** - Label of config channel
- string **path** - File path
- array:
 - long - The list of server ID for the comparison performed on

Returns

The following return values are available for this method:

- int **actionId** - The action ID of the scheduled action

11.22. METHOD: UPDATE

Description

Update a global config channel. Caller is at least a config administrator, an organization administrator, or have access to a system containing this config channel.

Parameters

The following parameters are available for this method:

- string **sessionKey**

- `string channelLabel`
- `string channelName`
- `string description`

Returns

The following return values are available for this method:

- `struct - Configuration Channel information`
 - `int id`
 - `int orgId`
 - `string label`
 - `string name`
 - `string description`
 - `struct configChannelType`
 - `struct - Configuration Channel Type information`
 - `int id`
 - `string label`
 - `string name`
 - `int priority`

CHAPTER 12. NAMESPACE: DISTCHANNEL

Provides methods to access and modify distribution channel information

Namespace: `distchannel`

12.1. METHOD: LISTDEFAULTMAPS

Description

Lists the default distribution channel maps.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- struct - `distChannelMap`
 - string `os` - Operating system
 - string `release` - Operating system release
 - string `arch_name` - Channel architecture
 - string `channel_label` - Channel label
 - string `org_specific` - Y for organization-specific, N for default

12.2. METHOD: LISTMAPSFORORG

Description

Lists distribution channel maps valid for the user's organization.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- struct - `distChannelMap`
 - string `os` - Operating system
 - string `release` - Operating system release
 - string `arch_name` - Channel architecture

- string `channel_label` - Channel label
- string `org_specific` - Y for organization-specific, N for default

12.3. METHOD: LISTMAPSFORORG

Description

Lists distribution channel maps valid for an organization. Requires Satellite administration permissions.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`

Returns

The following return values are available for this method:

- struct - `distChannelMap`
 - string `os` - Operating system
 - string `release` - Operating system release
 - string `arch_name` - Channel architecture
 - string `channel_label` - Channel label
 - string `org_specific` - Y for organization-specific, N for default

12.4. METHOD: SETMAPFORORG

Description

Sets and overrides (or removes if `channelLabel` is empty) a distribution channel map within an organization.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `os`
- string `release`
- string `archName`
- string `channelLabel`

Returns

The following return values are available for this method:

- int - **1** on success, exception thrown otherwise.

CHAPTER 13. NAMESPACE: ERRATA

Provides methods to access and modify errata.

Namespace: `errata`

13.1. METHOD: ADDPACKAGES

Description

Add a set of packages to an erratum with the given advisory name. This method only allows for modification of custom errata created either through the UI or API.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string advisoryName`
- `array:`
 - `int - packageId`

Returns

The following return values are available for this method:

- `int` - represents the number of packages added, exception otherwise

13.2. METHOD: APPLICABLETOCHANNELS

Description

Returns a list of channels applicable to the erratum with the given advisory name.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string advisoryName`

Returns

The following return values are available for this method:

- `array:`
 - `struct - channel`
 - `int channel_id`
 - `string label`
 - `string name`

- `string parent_channel_label`

13.3. METHOD: BUGZILLAFIXES

Description

Get the Bugzilla fixes for an erratum matching the given `advisoryName`. The bugs are returned in a struct where the bug ID is the key.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string advisoryName`

Returns

The following return values are available for this method:

- struct - Bugzilla info
 - `string bugzilla_id` - Actual bug ID number and the key for the struct entry
 - `string bug_summary` - Summary where the key is the bug ID number

13.4. METHOD: CLONE

Description

Clone a list of errata into the specified channel.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channel_label`
- array:
 - `string advisory` - The advisory name of the errata to clone

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - `int id` - Errata ID
 - `string date` - Erratum creation date
 - `string advisory_type` - Type of advisory

- string `advisory_name` - Name of advisory
- string `advisory_synopsis` - Summary of erratum

13.5. METHOD: CLONEASORIGINAL

Description

Clones a list of errata into a specified cloned channel according to the original errata.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channel_label`
- array:
 - string - `advisory` - The advisory name of the errata to clone

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `advisory_type` - Type of advisory
 - string `advisory_name` - Name of advisory
 - string `advisory_synopsis` - Summary of erratum

13.6. METHOD: CLONEASORIGINALASYNC

Description

Asynchronously clones a list of errata into a specified cloned channel according the original errata.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channel_label`
- array:
 - string `advisory` - The advisory name of the errata to clone

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

13.7. METHOD: CLONEASYNC

Description

Asynchronously clone a list of errata into the specified channel.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channel_label`
- `array`:
 - `string advisory` - The advisory name of the errata to clone

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

13.8. METHOD: CREATE

Description

Create a custom errata. If `publish` is set to true, the errata is published too.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `struct` - errata info
 - `string synopsis`
 - `string advisory_name`
 - `int advisory_release`
 - `string advisory_type` - Type of advisory (one of the following: `Security Advisory`, `Product Enhancement Advisory`, or `Bug Fix Advisory`)
 - `string product`
 - `string errataFrom`
 - `string topic`

- **string description**
- **string references**
- **string notes**
- **string solution**
- **array:**
 - **struct - bug**
 - **int id** - Bug ID
 - **string summary**
 - **string url**
- **array:**
 - **string keyword** - List of keywords to associate with the errata
- **array:**
 - **int - packageId**
- **boolean publish** - Should the errata be published
- **array:**
 - **string - channelLabel** - List of channels to publish the errata; ignored if **publish** is set to **false**

Returns

The following return values are available for this method:

- **struct - errata**
 - **int id** - Errata ID
 - **string date** - Erratum creation date
 - **string advisory_type** - Type of advisory
 - **string advisory_name** - Name of advisory
 - **string advisory_synopsis** - Summary of erratum

13.9. METHOD: DELETE

Description

Delete an erratum. This method only allows deletion of custom errata created either through the UI or API.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `advisoryName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

13.10. METHOD: FINDBYCVE

Description

Looks up the details for errata associated with the given CVE value e.g. CVE-2008-3270

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `cveName`

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `advisory_type` - Type of advisory
 - string `advisory_name` - Name of advisory
 - string `advisory_synopsis` - Summary of erratum

13.11. METHOD: GETDETAILS

Description

Retrieves the details for the erratum matching the given advisory name.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `advisoryName`

Returns

The following return values are available for this method:

- **struct - erratum**
 - string **issue_date**
 - string **update_date**
 - string **last_modified_date** - This date is only included for published erratum and represents the erratum's late date of modification
 - string **synopsis**
 - int **release**
 - string **type**
 - string **product**
 - string **errataFrom**
 - string **topic**
 - string **description**
 - string **references**
 - string **notes**
 - string **solution**

13.12. METHOD: LISTAFFECTEDSYSTEMS

Description

Return the list of systems affected by the erratum with advisory name.

Parameters

The following parameters are available for this method:

- string **sessionKey**
- string **advisoryName**

Returns

The following return values are available for this method:

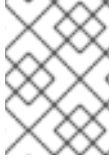
- array:
 - struct - system
 - int **id**
 - string **name**

- `dateTime.iso8601 last_checkin` - Last successful server check-in

13.13. METHOD: LISTBYDATE

Description

List errata that have been applied to a particular channel by date.



NOTE

This method is deprecated. The `channel.software.listErrata(string sessionKey, string channelLabel)` now replaces it.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string channelLabel`

Returns

The following return values are available for this method:

- `array:`
 - `struct - errata`
 - `int id` - Errata ID
 - `string date` - Erratum creation date
 - `string advisory_type` - Type of advisory
 - `string advisory_name` - Name of advisory
 - `string advisory_synopsis` - Summary of erratum

13.14. METHOD: LISTCVEs

Description

Returns a list of [CVEs](#) applicable to the erratum with the given advisory name. CVEs may be associated only with published errata.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string advisoryName`

Returns

The following return values are available for this method:

- array:
 - string `cveName`

13.15. METHOD: LISTKEYWORDS

Description

Get the keywords associated with an erratum matching the given advisory name.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `advisoryName`

Returns

The following return values are available for this method:

- array:
 - string - Keyword associated with erratum

13.16. METHOD: LISTPACKAGES

Description

Returns a list of the packages affected by the erratum with the given advisory name.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `advisoryName`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - int `id`
 - string `name`
 - string `epoch`
 - string `version`
 - string `release`
 - string `arch_label`

- `array providing_channels`
 - `string` - Channel label providing this package
- `string build_host`
- `string description`
- `string checksum`
- `string checksum_type`
- `string vendor`
- `string summary`
- `string cookie`
- `string license`
- `string path`
- `string file`
- `string build_date`
- `string last_modified_date`
- `string size`
- `string payload_size`

13.17. METHOD: LISTUNPUBLISHEDERRATA

Description

Returns a list of unpublished errata

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - erratum`
 - `int id`
 - `int published`
 - `string advisory`

- string `advisory_name`
- string `advisory_type`
- string `synopsis`
- `dateTime.iso8601` `created`
- `dateTime.iso8601` `update_date`

13.18. METHOD: PUBLISH

Description

Publish an existing unpublished errata to a set of channels.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `advisoryName`
- array:
 - string - `channelLabel` - List of channel labels to publish the errata

Returns

The following return values are available for this method:

- struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `advisory_type` - Type of advisory
 - string `advisory_name` - Name of advisory
 - string `advisory_synopsis` - Summary of erratum

13.19. METHOD: PUBLISHASORIGINAL

Description

Publishes an existing unpublished cloned errata to a set of cloned channels according to its original erratum.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `advisoryName`

- array:
 - string `channelLabel` - List of channel labels to publish errata

Returns

The following return values are available for this method:

- struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `advisory_type` - Type of advisory
 - string `advisory_name` - Name of advisory
 - string `advisory_synopsis` - Summary of erratum

13.20. METHOD: REMOVEPACKAGES

Description

Remove a set of packages from an erratum with the given advisory name. This method only allows for modification of custom errata created either through the UI or API.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `advisoryName`
- array:
 - int - `packageId`

Returns

The following return values are available for this method:

- int - Represents the number of packages removed, exception otherwise

13.21. METHOD: SETDETAILS

Description

Set erratum details. All arguments are optional and are only modified if included in the struct. This method only allows for modification of custom errata created either through the UI or API.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `advisoryName`

- **struct - errata details**
 - **string synopsis**
 - **string advisory_name**
 - **int advisory_release**
 - **string advisory_type** - Type of advisory (one of the following: **Security Advisory**, **Product Enhancement Advisory**, or **Bug Fix Advisory**)
 - **string product**
 - **dateTime.iso8601 issue_date**
 - **dateTime.iso8601 update_date**
 - **string errataFrom**
 - **string topic**
 - **string description**
 - **string references**
 - **string notes**
 - **string solution**
 - **array bugs** - **bugs** is the key into the struct
 - array:
 - **struct - bug**
 - **int id** - Bug ID
 - **string summary**
 - **string url**
 - **array keywords** - **keywords** is the key into the struct
 - array:
 - **string - keyword** - List of keywords to associate with the errata
 - **array CVEs** - **cves** is the key into the struct
 - array:
 - **string - cves** - List of CVEs to associate with the errata (valid only for published errata)

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

CHAPTER 14. NAMESPACE: KICKSTART

Provides methods to create kickstart files.

Namespace: `kickstart`

14.1. METHOD: CLONEPROFILE

Description

Clone a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabelToClone` - Label of the kickstart profile to clone
- string `newKsLabel` - Label of the cloned profile

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

14.2. METHOD: CREATEPROFILE

Description

Import a kickstart profile into Red Hat Network.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `profileLabel` - Label for the new kickstart profile
- string `virtualizationType` - `none`, `para_host`, `qemu`, `xenfv`, or `xenpv`
- string `kickstartableTreeLabel` - Label of a kickstartable tree to associate the new profile
- string `kickstartHost` - Kickstart hostname (of a Red Hat Satellite or Red Hat Satellite Proxy) used to construct the default download URL for the new kickstart profile.
- string `rootPassword` - root password
- string `updateType` - Update the profile to use the newest tree available. Possible values are: `none` (default), `red_hat` (only use kickstart trees synchronized from Red Hat), or `all` (includes custom kickstart trees)

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

14.3. METHOD: CREATEPROFILE

Description

Import a kickstart profile into Red Hat Network.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string profileLabel` - Label for the new kickstart profile
- `string virtualizationType` - `none`, `para_host`, `qemu`, `xenfv`, or `xenpv`
- `string kickstartableTreeLabel` - Label of a kickstartable tree to associate the new profile
- `string kickstartHost` - Kickstart hostname (of a Red Hat Satellite or Red Hat Satellite Proxy) used to construct the default download URL for the new kickstart profile
- `string rootPassword` - root password

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

14.4. METHOD: CREATEPROFILEWITHCUSTOMURL

Description

Import a kickstart profile into Red Hat Network.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string profileLabel` - Label for the new kickstart profile
- `string virtualizationType` - `none`, `para_host`, `qemu`, `xenfv`, or `xenpv`
- `string kickstartableTreeLabel` - Label of a kickstartable tree to associate the new profile
- `boolean downloadUrl` - Download URL, or `default` to use the kickstart tree's default URL.
- `string rootPassword` - root password

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

14.5. METHOD: CREATEPROFILEWITHCUSTOMURL

Description

Import a kickstart profile into Red Hat Network.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `profileLabel` - Label for the new kickstart profile
- string `virtualizationType` - `none`, `para_host`, `qemu`, `xenfv`, or `xenpv`
- string `kickstartableTreeLabel` - Label of a kickstartable tree to associate the new profile
- boolean `downloadUrl` - Download URL, or `default` to use the kickstart tree's default URL.
- string `rootPassword` - root password
- string `updateType` - Update the profile to use the newest tree available. Possible values are: `none` (default), `red_hat` (only use kickstart trees synchronized from Red Hat), or `all` (includes custom kickstart trees)

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

14.6. METHOD: DELETEPROFILE

Description

Delete a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The label of the kickstart profile to remove

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

14.7. METHOD: DISABLEPROFILE

Description

Enable or disable a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `profileLabel` - Label for the kickstart tree to enable or disable
- string `disabled` - `true` to disable the profile

Returns

The following return values are available for this method:

- int - `1` on success, exception thrown otherwise.

14.8. METHOD: FINDKICKSTARTFORIP

Description

Find an associated kickstart for a given IP address.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ipAddress` - The IP address to find

Returns

The following return values are available for this method:

- string - `label` of the kickstart. Returns an empty string ("") if not found.

14.9. METHOD: IMPORTFILE

Description

Import a kickstart profile into Red Hat Network.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `profileLabel` - Label for the new kickstart profile
- string `virtualizationType` - `none`, `para_host`, `qemu`, `xenfv`, or `xenpv`
- string `kickstartableTreeLabel` - Label of a kickstartable tree to associate the new profile
- string `kickstartFileContents` - Contents of the kickstart file to import

Returns

The following return values are available for this method:

- int - `1` on success, exception thrown otherwise

14.10. METHOD: IMPORTFILE

Description

Import a kickstart profile into Red Hat Network.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `profileLabel` - Label for the new kickstart profile
- string `virtualizationType` - `none`, `para_host`, `qemu`, `xenfv`, or `xenpv`
- string `kickstartableTreeLabel` - Label of a kickstartable tree to associate the new profile
- string `kickstartHost` - Kickstart hostname (of a Red Hat Satellite or Red Hat Satellite Proxy) used to construct the default download URL for the new kickstart profile; using this option signifies that this default URL will be used instead of any `url/nfs/cdrom/harddrive` commands in the kickstart file itself
- string `kickstartFileContents` - Contents of the kickstart file to import

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

14.11. METHOD: IMPORTFILE

Description

Import a kickstart profile into Red Hat Network.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `profileLabel` - Label for the new kickstart profile
- string `virtualizationType` - `none`, `para_host`, `qemu`, `xenfv`, or `xenpv`
- string `kickstartableTreeLabel` - Label of a kickstartable tree to associate the new profile
- string `kickstartHost` - Kickstart hostname (of a Red Hat Satellite or Red Hat Satellite Proxy) used to construct the default download URL for the new kickstart profile. Using this option signifies that this default URL will be used instead of any `url/nfs/cdrom/harddrive` commands in the kickstart file itself.
- string `kickstartFileContents` - Contents of the kickstart file to import

- string `updateType` - Update the profile to use the newest tree available. Possible values are: `none` (default), `red_hat` (only use kickstart trees synchronized from Red Hat), or `all` (includes custom kickstart trees)

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

14.12. METHOD: IMPORTRAWFILE

Description

Import a raw kickstart file into Red Hat Satellite.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `profileLabel` - Label for the new kickstart profile
- string `virtualizationType` - `none`, `para_host`, `qemu`, `xenfv`, or `xenpv`
- string `kickstartableTreeLabel` - Label of a kickstartable tree to associate the new profile
- string `kickstartFileContents` - Contents of the kickstart file to import

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

14.13. METHOD: IMPORTRAWFILE

Description

Import a raw kickstart file into Red Hat Satellite.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `profileLabel` - Label for the new kickstart profile
- string `virtualizationType` - `none`, `para_host`, `qemu`, `xenfv`, or `xenpv`
- string `kickstartableTreeLabel` - Label of a kickstartable tree to associate the new profile.
- string `kickstartFileContents` - Contents of the kickstart file to import.

- string `updateType` - Update the profile to use the newest tree available. Possible values are: `none` (default), `red_hat` (only use kickstart trees synchronized from Red Hat), or `all` (includes custom kickstart trees)

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

14.14. METHOD: ISPROFILEDISABLED

Description

Returns whether a kickstart profile is disabled

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `profileLabel` - Kickstart profile label

Returns

The following return values are available for this method:

- true if profile is disabled

14.15. METHOD: LISTALLIPRANGES

Description

List all IP ranges and their associated kickstarts available in the user's organization.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - Kickstart IP range
 - string `ksLabel` - The kickstart label associated with the IP range
 - string `max` - The maximum IP of the range
 - string `min` - The minimum IP of the range

14.16. METHOD: LISTKICKSTARTABLECHANNELS

Description

List kickstartable channels for the logged in user.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

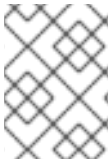
- `array:`
 - `struct - channel`
 - `int id`
 - `string name`
 - `string label`
 - `string arch_name`
 - `string summary`
 - `string description`
 - `string checksum_label`
 - `dateTime.iso8601 last_modified`
 - `string maintainer_name`
 - `string maintainer_email`
 - `string maintainer_phone`
 - `string support_policy`
 - `string gpg_key_url`
 - `string gpg_key_id`
 - `string gpg_key_fp`
 - `dateTime.iso8601 yumrepo_last_sync` (optional)
 - `string end_of_life`
 - `string parent_channel_label`
 - `string clone_original`
 - `array:`

- struct - contentSources
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

14.17. METHOD: LISTKICKSTARTABLETREES

Description

List the available kickstartable trees for the given channel.



NOTE

This method is deprecated. Use the `kickstart.tree.list(string sessionKey, string channelLabel)` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Label of channel to search

Returns

The following return values are available for this method:

- array:
 - struct - kickstartable tree
 - int `id`
 - string `label`
 - string `base_path`
 - int `channel_id`

14.18. METHOD: LISTKICKSTARTS

Description

Provides a list of kickstart profiles visible to the user's organization.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - kickstart
 - string `label`
 - string `tree_label`
 - string `name`
 - boolean `advanced_mode`
 - boolean `org_default`
 - boolean `active`
 - string `update_type`

14.19. METHOD: RENAMEPROFILE

Description

Rename a kickstart profile in Red Hat Satellite.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `originalLabel` - Label for the kickstart profile to rename
- string `newLabel` - New label for the kickstart profile

Returns

The following return values are available for this method:

- int - `1` on success, exception thrown otherwise.

CHAPTER 15. NAMESPACE: KICKSTART.FILEPRESERVATION

Provides methods to retrieve and manipulate kickstart file preservation lists.

Namespace: `kickstart.filepreservation`

15.1. METHOD: CREATE

Description

Create a new file preservation list.

Parameters

The following parameters are available for this method:

- string `session_key`
- string `name` - Name of the file list to create
- array:
 - string `name` - File names to include

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

15.2. METHOD: DELETE

Description

Delete a file preservation list.

Parameters

The following parameters are available for this method:

- string `session_key`
- string `name` - Name of the file list to delete

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

15.3. METHOD: GETDETAILS

Description

Returns all of the data associated with the given file preservation list.

Parameters

The following parameters are available for this method:

- `string session_key`
- `string name` - Name of the file list to retrieve details

Returns

The following return values are available for this method:

- `struct` - file list
 - `string name`
 - `array file_names`
 - `string name`

15.4. METHOD: LISTALLFILEPRESERVATIONS

Description

List all file preservation lists for the organization associated with the user logged into the given session.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array`:
 - `struct` - file preservation
 - `int id`
 - `string name`
 - `dateTime.iso8601 created`
 - `dateTime.iso8601 last_modified`

CHAPTER 16. NAMESPACE: KICKSTART.KEYS

Provides methods to manipulate kickstart keys.

Namespace: `kickstart.keys`

16.1. METHOD: CREATE

Description

Creates a new key with the given parameters.

Parameters

The following parameters are available for this method:

- string `session_key`
- string `description`
- string `type` - Valid values are `GPG` or `SSL`
- string `content`

Returns

The following return values are available for this method:

- int - `1` on success, exception thrown otherwise.

16.2. METHOD: DELETE

Description

deletes the key identified by the given parameters

Parameters

The following parameters are available for this method:

- string `session_key`
- string `description`

Returns

The following return values are available for this method:

- int - `1` on success, exception thrown otherwise.

16.3. METHOD: GETDETAILS

Description

Returns all of the data associated with the given key.

Parameters

The following parameters are available for this method:

- `string session_key`
- `string description`

Returns

The following return values are available for this method:

- `struct - key`
 - `string description`
 - `string type`
 - `string content`

16.4. METHOD: LISTALLKEYS

Description

List all keys for the org associated with the user logged into the given session.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - key`
 - `string description`
 - `string type`

16.5. METHOD: UPDATE

Description

Updates type and content of the key identified by the description.

Parameters

The following parameters are available for this method:

- `string session_key`
- `string description`
- `string type` - Valid values are `GPG` or `SSL`
- `string content`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise

CHAPTER 17. NAMESPACE: KICKSTART.PROFILE

Provides methods to access and modify many aspects of a kickstart profile.

Namespace: `kickstart.profile`

17.1. METHOD: ADDIPRANGE

Description

Add an IP range to a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - The label of the kickstart
- string `min` - The IP address for minimum of the range
- string `max` - The IP address for the maximum of the range

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

17.2. METHOD: ADDSCRIPT

Description

Add a pre or post script to a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart label to add the script
- string `name` - The kickstart script name
- string `contents` - The full script to add
- string `interpreter` - The path to the interpreter to use e.g. `/bin/bash`; an empty string enables the kickstart default interpreter
- string `type` - The type of script; either `pre` or `post`
- boolean `chroot` - Whether to run the script in the chrooted install location (recommended) or not.

Returns

The following return values are available for this method:

- `int id` - The ID of the added script

17.3. METHOD: ADDSCRIPT

Description

Add a pre or post script to a kickstart profile.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel` - The kickstart label to add the script
- `string name` - The kickstart script name
- `string contents` - The full script to add
- `string interpreter` - The path to the interpreter to use e.g. `/bin/bash`; an empty string enables the kickstart default interpreter
- `string type` - The type of script; either `pre` or `post`
- `boolean chroot` - Whether to run the script in the chrooted install location (recommended) or not.
- `boolean template` - Enable templating using cobbler

Returns

The following return values are available for this method:

- `int id` - The ID of the added script

17.4. METHOD: ADDSCRIPT

Description

Add a pre or post script to a kickstart profile.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel` - The kickstart label to add the script
- `string name` - The kickstart script name
- `string contents` - The full script to add
- `string interpreter` - The path to the interpreter to use e.g. `/bin/bash`; an empty string enables the kickstart default interpreter

- **string type** - The type of script; either **pre** or **post**
- **boolean chroot** - Whether to run the script in the chrooted install location (recommended) or not.
- **boolean template** - Enable templating using cobbler
- **boolean erroronfail** - Whether to throw an error if the script fails or not

Returns

The following return values are available for this method:

- **int id** - The ID of the added script

17.5. METHOD: COMPAREACTIVATIONKEYS

Description

Returns a list for each kickstart profile. Each list contains activation keys not present on the other profile.

Parameters

The following parameters are available for this method:

- **string sessionKey**
- **string kickstartLabel1**
- **string kickstartLabel2**

Returns

The following return values are available for this method:

- **struct** - Comparison Info
 - **array "kickstartLabel1"** - Actual label of the first kickstart profile is the key into the struct
 - **array:**
 - **struct** - activation key
 - **string key**
 - **string description**
 - **int usage_limit**
 - **string base_channel_label**
 - **array child_channel_labels**
 - **string childChannelLabel**
 - **array entitlements**

- `string entitlementLabel`
- `array server_group_ids`
 - `string serverGroupId`
- `array package_names`
 - `string packageName` - (deprecated by packages)
- `array packages`
 - `struct - package`
 - `string name` - `packageName`
 - `string arch` - `archLabel` - optional
- `boolean universal_default`
- `boolean disabled`
- `array "kickstartLabel2"` - Actual label of the second kickstart profile is the key into the struct
 - `array:`
 - `struct - activation key`
 - `string key`
 - `string description`
 - `int usage_limit`
 - `string base_channel_label`
 - `array child_channel_labels`
 - `string childChannelLabel`
 - `array entitlements`
 - `string entitlementLabel`
 - `array server_group_ids`
 - `string serverGroupId`
 - `array package_names`
 - `string packageName` - (deprecated by packages)
 - `array packages`
 - `struct - package`

- string `name` - `packageName`
- string `arch` - `archLabel` - optional
- boolean `universal_default`
- boolean `disabled`

17.6. METHOD: COMPAREADVANCEDOPTIONS

Description

Returns a list for each kickstart profile. Each list contains the properties that differ between the profiles and their values for that specific profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kickstartLabel1`
- string `kickstartLabel2`

Returns

The following return values are available for this method:

- struct - Comparison Info
 - array "kickstartLabel1" - Actual label of the first kickstart profile is the key into the struct
 - array:
 - struct - value
 - string `name`
 - string `value`
 - boolean `enabled`
 - array "kickstartLabel2" - Actual label of the second kickstart profile is the key into the struct
 - array:
 - struct - value
 - string `name`
 - string `value`
 - boolean `enabled`

17.7. METHOD: COMPAREPACKAGES

Description

Returns a list for each kickstart profile. Each list contains package names not present on the other profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kickstartLabel1`
- string `kickstartLabel2`

Returns

The following return values are available for this method:

- struct - Comparison Info
 - array "kickstartLabel1" - Actual label of the first kickstart profile is the key into the struct
 - array:
 - string - packagename
 - array "kickstartLabel2" - Actual label of the second kickstart profile is the key into the struct
 - array:
 - string - packagename

17.8. METHOD: DOWNLOADKICKSTART

Description

Download the full contents of a kickstart file.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel1` - The label of the kickstart to download.
- string `host` - The host to use when referring to the satellite itself; usually this is the FQDN of the Satellite, but can be the IP address or shortname of it too

Returns

The following return values are available for this method:

- string - The contents of the kickstart file

**NOTE**

If an activation key is not associated with the kickstart file, registration does not occur in the Satellite-generated `%post` section. If one is associated, it is used for registration.

17.9. METHOD: DOWNLOADRENDEREDKICKSTART

Description

Downloads the Cobbler-rendered Kickstart file.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The label of the kickstart to download

Returns

The following return values are available for this method:

- string - The contents of the kickstart file

17.10. METHOD: GETADVANCEDOPTIONS

Description

Get advanced options for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - Label of kickstart profile to be changed

Returns

The following return values are available for this method:

- array:
 - struct - option
 - string `name`
 - string `arguments`

17.11. METHOD: GETCFGPRESERVATION

Description

Get `ks.cfg` preservation option for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kslabel` - Label of kickstart profile to be changed.

Returns

The following return values are available for this method:

- boolean - The value of the option; `true` means that `ks.cfg` is copied to `/root` and `false` means it is not.

17.12. METHOD: GETCHILDCHANNELS

Description

Get the child channels for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kslabel` - Label of kickstart profile

Returns

The following return values are available for this method:

- array:
 - string - `channelLabel`

17.13. METHOD: GETCUSTOMOPTIONS

Description

Get custom options for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel`

Returns

The following return values are available for this method:

- array:
 - struct - option
 - int `id`

- string arguments

17.14. METHOD: GETKICKSTARTTREE

Description

Get the kickstart tree for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kslabel` - Label of kickstart profile to be changed

Returns

The following return values are available for this method:

- string `kstreeLabel` - Label of the kickstart tree

17.15. METHOD: GETUPDATETYPE

Description

Get the update type for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kslabel` - Label of kickstart profile

Returns

The following return values are available for this method:

- string `update_type` - Update type for this kickstart profile

17.16. METHOD: GETVARIABLES

Description

Returns a list of variables associated with the specified kickstart profile

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel`

Returns

The following return values are available for this method:

- array:
 - struct - kickstart variable
 - string key
 - string or int value

17.17. METHOD: LISTIPRANGES

Description

List all IP ranges for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - The label of the kickstart

Returns

The following return values are available for this method:

- array:
 - struct - Kickstart IP Range
 - string `ksLabel` - The kickstart label associated with the IP range
 - string `max` - The maximum IP of the range
 - string `min` - The minimum IP of the range

17.18. METHOD: LISTSCRIPTS

Description

List the pre and post scripts for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The label of the kickstart

Returns

The following return values are available for this method:

- array:
 - struct - kickstart script
 - int `id`

- `string contents`
- `string script_type` - The type of script; either `pre` or `post`
- `string interpreter` - The scripting language interpreter to use for this script. An empty string indicates the default kickstart shell.
- `boolean chroot` - `true` if the script executes within the chroot environment.
- `boolean erroronfail` - `true` if the script throws an error if it fails.
- `boolean template` - `true` if templating using cobbler is enabled

17.19. METHOD: REMOVEIPRANGE

Description

Remove an IP range from a kickstart profile.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel` - The kickstart label of the IP range to remove
- `string ip_address` - An IP Address that falls within the range to remove. The minimum or maximum of the range also works.

Returns

The following return values are available for this method:

- `int` - `1` on successful removal, `0` if range was not found for the specified kickstart, exception otherwise.

17.20. METHOD: REMOVESCRIPT

Description

Remove a script from a kickstart profile.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel` - The kickstart from which to remove the script
- `int scriptId` - The ID of the script to remove

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

17.21. METHOD: SETADVANCEDOPTIONS

Description

Set advanced options for a kickstart profile. If `md5_crypt_rootpw` is set to `True`, `root_pw` is taken as plain text and MD5 encrypted on server side, otherwise a hash encoded password (according to the `auth` option) is expected

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel`
- array:
 - struct - advanced options
 - string `name` - Name of the advanced option. Valid option names: `autostep`, `interactive`, `install`, `upgrade`, `text`, `network`, `cdrom`, `harddrive`, `nfs`, `url`, `lang`, `langsupport`, `keyboard`, `mouse`, `device`, `deviceprobe`, `zerombr`, `clearpart`, `bootloader`, `timezone`, `auth`, `rootpw`, `selinux`, `reboot`, `firewall`, `xconfig`, `skipx`, `key`, `ignoredisk`, `autopart`, `cmdline`, `firstboot`, `graphical`, `iscsi`, `iscsiname`, `logging`, `monitor`, `multipath`, `poweroff`, `halt`, `services`, `shutdown`, `user`, `vnc`, `zfcpl`, `driverdisk`, `md5_crypt_rootpw`
 - string `arguments` - Arguments of the option

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

17.22. METHOD: SETCFGPRESERVATION

Description

Set `ks.cfg` preservation option for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kslabel` - Label of kickstart profile to be changed
- boolean `preserve` - Defines whether or not `ks.cfg` and all `%include` fragments are copied to `/root`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

17.23. METHOD: SETCHILDCHANNELS

Description

Set the child channels for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kslabel` - Label of kickstart profile to be changed
- string[] `channelLabels` - List of labels of child channels

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

17.24. METHOD: SETCUSTOMOPTIONS

Description

Set custom options for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel`
- string `options`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

17.25. METHOD: SETKICKSTARTTREE

Description

Set the kickstart tree for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kslabel` - Label of kickstart profile to be changed
- string `kstreeLabel` - Label of new kickstart tree

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

17.26. METHOD: SETLOGGING

Description

Set logging options for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kslabel` - Label of kickstart profile to be changed
- boolean `pre` - Defines whether or not to log the pre section of a kickstart to `/root/ks-pre.log`
- boolean `post` - Defines whether or not to log the post section of a kickstart to `/root/ks-post.log`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

17.27. METHOD: SETUPDATETYPE

Description

Set the update type for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kslabel` - Label of kickstart profile to be changed
- string `updateType` - The new update type to set. Possible values are `red_hat`, `all`, and `none`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

17.28. METHOD: SETVARIABLES

Description

Associates list of kickstart variables with the specified kickstart profile.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel`
- `array:`
 - `struct - kickstart variable`
 - `string key`
 - `string or int value`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise

CHAPTER 18. NAMESPACE: KICKSTART.PROFILE.KEYS

Provides methods to access and modify the list of activation keys associated with a kickstart profile.

Namespace: `kickstart.profile.keys`

18.1. METHOD: ADDACTIVATIONKEY

Description

Add an activation key association to the kickstart profile.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel` - The kickstart profile label
- `string key` - The activation key

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

18.2. METHOD: GETACTIVATIONKEYS

Description

Looks up the activation keys associated with the kickstart profile.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel` - The kickstart profile label

Returns

The following return values are available for this method:

- `array`:
 - `struct` - activation key
 - `string key`
 - `string description`
 - `int usage_limit`
 - `string base_channel_label`

- array `child_channel_labels`
 - string `childChannelLabel`
- array `entitlements`
 - string `entitlementLabel`
- array `server_group_ids`
 - string `serverGroupId`
- array `package_names`
 - string `packageName` (deprecated by `packages`)
- array `packages`
 - struct - package
 - string `name` - Package name
 - string `arch` - Architecture label (optional)
- boolean `universal_default`
- boolean `disabled`

18.3. METHOD: REMOVEACTIVATIONKEY

Description

Remove an activation key association from the kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label
- string `key` - The activation key

Returns

The following return values are available for this method:

- int - **1** on success, exception thrown otherwise.

CHAPTER 19. NAMESPACE: KICKSTART.PROFILE.SOFTWARE

Provides methods to access and modify the software list associated with a kickstart profile.

Namespace: `kickstart.profile.software`

19.1. METHOD: APPENDTOSOFTWARELIST

Description

Append the list of software packages to a kickstart profile. Duplicate packages are ignored.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel` - The label of a kickstart profile
- `string[] packageList` - A list of package names to be added to the profile

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

19.2. METHOD: GETSOFTWARELIST

Description

Get a list of a kickstart profile's software packages.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel` - The label of a kickstart profile

Returns

The following return values are available for this method:

- `Array`:
 - `string` - A list of a kickstart profile's software packages

19.3. METHOD: SETSOFTWARELIST

Description

Set the list of software packages for a kickstart profile.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string ksLabel` - The label of a kickstart profile
- Array:
 - `string packageList` - A list of package names to be set on the profile

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

CHAPTER 20. NAMESPACE: KICKSTART.PROFILE.SYSTEM

Provides methods to set various properties of a kickstart profile.

Namespace: `kickstart.profile.system`

20.1. METHOD: ADDFILEPRESERVATIONS

Description

Adds the given list of file preservations to the specified kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kickstartLabel`
- array:
 - string - `filePreservations`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

20.2. METHOD: ADDKEYS

Description

Adds the given list of keys to the specified kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kickstartLabel`
- array:
 - string - `keyDescription`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

20.3. METHOD: CHECKCONFIGMANAGEMENT

Description

Check the configuration management status for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label

Returns

The following return values are available for this method:

- boolean `enabled` - `true` if configuration management is enabled; otherwise, `false`

20.4. METHOD: CHECKREMOTECOMMANDS

Description

Check the remote commands status flag for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label

Returns

The following return values are available for this method:

- boolean `enabled` - `true` if remote commands support is enabled; otherwise, `false`

20.5. METHOD: DISABLECONFIGMANAGEMENT

Description

Disables the configuration management flag in a kickstart profile so that a system created using this profile is *not* configuration capable.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label

Returns

The following return values are available for this method:

- int - `1` on success, exception thrown otherwise

20.6. METHOD: DISABLEREMOTECOMMANDS

Description

Disables the remote command flag in a kickstart profile so that a system created using this profile is capable of running remote commands.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

20.7. METHOD: ENABLECONFIGMANAGEMENT

Description

Enables the configuration management flag in a kickstart profile so that a system created using this profile is configuration capable.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

20.8. METHOD: ENBLEREMOTECOMMANDS

Description

Enables the remote command flag in a kickstart profile so that a system created using this profile is capable of running remote commands.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

20.9. METHOD: GETLOCALE

Description

Retrieves the locale for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label

Returns

The following return values are available for this method:

- struct - locale info
 - string `locale`
 - boolean `useUtc`
 - `true` - The hardware clock uses UTC
 - `false` - The hardware clock does not use UTC

20.10. METHOD: GETPARTITIONINGScheme

Description

Get the partitioning scheme for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The label of a kickstart profile.

Returns

The following return values are available for this method:

- Array:
 - string - A list of partitioning commands used to setup the partitions, logical volumes and volume groups.

20.11. METHOD: GETREGISTRATIONTYPE

Description

Returns the registration type of a given kickstart profile. Registration type is one of `reactivation`, `deletion`, or `none`. These types determine the behavior of the registration when using this profile for reprovisioning.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kickstartLabel`

Returns

The following return values are available for this method:

- string `registrationType`
 - reactivation
 - deletion
 - none

20.12. METHOD: GETSELINUX

Description

Retrieves the SELinux enforcing mode property of a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label

Returns

The following return values are available for this method:

- string `enforcingMode`
 - enforcing
 - permissive
 - disabled

20.13. METHOD: LISTFILEPRESERVATIONS

Description

Returns the set of all file preservations associated with the given kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kickstartLabel`

Returns

The following return values are available for this method:

- array:
 - struct - file list
 - string name
 - array `file_names`
 - string name

20.14. METHOD: LISTKEYS

Description

Returns the set of all keys associated with the given kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kickstartLabel`

Returns

The following return values are available for this method:

- array:
 - struct - key
 - string `description`
 - string `type`
 - string `content`

20.15. METHOD: REMOVEFILEPRESERVATIONS

Description

Removes the given list of file preservations from the specified kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kickstartLabel`
- array:
 - string - `filePreservations`

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

20.16. METHOD: REMOVEKEYS

Description

Removes the given list of keys from the specified kickstart profile.

Parameters

The following parameters are available for this method:

- `string` `sessionKey`
- `string` `kickstartLabel`
- `array`:
 - `string` - `keyDescription`

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

20.17. METHOD: SETLOCALE

Description

Sets the locale for a kickstart profile.

Parameters

The following parameters are available for this method:

- `string` `sessionKey`
- `string` `ksLabel` - The kickstart profile label
- `string` `locale` - The locale code
- `boolean` `useUtc`
 - `true` - The hardware clock uses UTC
 - `false` - The hardware clock does not use UTC

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

20.18. METHOD: SETPARTITIONINGScheme

Description

Set the partitioning scheme for a kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The label of the kickstart profile to update.
- Array:
 - string `scheme` - The partitioning scheme is a list of partitioning command strings used to setup the partitions, volume groups and logical volumes.

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

20.19. METHOD: SETREGISTRATIONTYPE

Description

Sets the registration type of a given kickstart profile. Registration type is one of `reactivation`, `deletion`, or `none`. These types determine the behaviour of re-registration when using this profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `kickstartLabel`
- string `registrationType`
 - `reactivation` - Try and generate a reactivation key and use that to register the system when reprovisioning a system
 - `deletion` - Try and delete the existing system profile and reregister the system being reprovisioned as new
 - `none` - Preserve the status quo and leave the current system as a duplicate on a reprovision.

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

20.20. METHOD: SETSELINUX

Description

Sets the SELinux enforcing mode property of a kickstart profile so that a system created using this profile has the appropriate SELinux enforcing mode.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `ksLabel` - The kickstart profile label
- string `enforcingMode` - The SELinux enforcing mode
 - `enforcing`
 - `permissive`
 - `disabled`

Returns

The following return values are available for this method:

- int - `1` on success, exception thrown otherwise

CHAPTER 21. NAMESPACE: KICKSTART.SNIPPET

Provides methods to create kickstart files.

Namespace: `kickstart.snippet`

21.1. METHOD: CREATEORUPDATE

Description

Creates a snippet with the given name and contents if it does not exist. If it does exist, the existing snippet is updated.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `name`
- string `contents`

Returns

The following return values are available for this method:

- struct - snippet
 - string `name`
 - string `contents`
 - string `fragment` - The string to include in a kickstart file that generates this snippet
 - string `file` - The local path to the file containing this snippet

21.2. METHOD: DELETE

Description

Delete the specified snippet. If the snippet is not found, `0` is returned.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `name`

Returns

The following return values are available for this method:

- int - `1` on success, exception thrown otherwise.

21.3. METHOD: LISTALL

Description

List all cobbler snippets for the logged in user

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - snippet
 - string `name`
 - string `contents`
 - string `fragment` - The string to include in a kickstart file that generates this snippet
 - string `file` - The local path to the file containing this snippet

21.4. METHOD: LISTCUSTOM

Description

List only custom snippets for the logged in user. These snippets are editable.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - snippet
 - string `name`
 - string `contents`
 - string `fragment` - The string to include in a kickstart file that generates this snippet
 - string `file` - The local path to the file containing this snippet

21.5. METHOD: LISTDEFAULT

Description

List only pre-made default snippets for the logged in user. These snippets are not editable.

Parameters

The following parameters are available for this method:

- **string `sessionKey`**

Returns

The following return values are available for this method:

- **array:**
 - **struct - snippet**
 - **string `name`**
 - **string `contents`**
 - **string `fragment`** - The string to include in a kickstart file that generates this snippet
 - **string `file`** - The local path to the file containing this snippet

CHAPTER 22. NAMESPACE: KICKSTART.TREE

Provides methods to access and modify the kickstart trees.

Namespace: `kickstart.tree`

22.1. METHOD: CREATE

Description

Create a kickstart tree (Distribution) in Red Hat Satellite.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `treeLabel` - The new kickstart tree label
- string `basePath` - Path to the base or root of the kickstart tree
- string `channelLabel` - Label of channel to associate with the kickstart tree.
- string `installType` - Label for `KickstartInstallType` (`rhel_3`, `rhel_4`, `rhel_5`, `rhel_6`, `fedora_18`).

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

22.2. METHOD: DELETE

Description

Delete a kickstart tree (Distribution) in Red Hat Satellite.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `treeLabel` - Label for the kickstart tree to delete

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

22.3. METHOD: DELETETREEANDPROFILES

Description

Delete a `kickstarttree` and any profiles associated with this kickstart tree.

**WARNING**

This method deletes all profiles associated with this kickstart tree.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `treeLabel` - Label for the kickstart tree to delete

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

22.4. METHOD: GETDETAILS

Description

The detailed information about a kickstartable tree given the tree name.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `treeLabel` - Label of kickstartable tree to search

Returns

The following return values are available for this method:

- struct - kickstartable tree
 - int `id`
 - string `label`
 - string `abs_path`
 - int `channel_id`
 - struct - kickstart install type
 - int `id`
 - string `label`
 - string `name`

22.5. METHOD: LIST

Description

List the available kickstartable trees for the given channel.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `channelLabel` - Label of channel to search

Returns

The following return values are available for this method:

- array:
 - struct - kickstartable tree
 - int `id`
 - string `label`
 - string `base_path`
 - int `channel_id`

22.6. METHOD: LISTINSTALLTYPES

Description

List the available kickstartable install types.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - kickstart install type
 - int `id`
 - string `label`
 - string `name`

22.7. METHOD: RENAME

Description

Rename a kickstart tree (Distribution) in Red Hat Satellite.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `originalLabel` - Label for the kickstart tree to rename
- string `newLabel` - The kickstart tree's new label

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

22.8. METHOD: UPDATE

Description

Edit a kickstart tree (Distribution) in Red Hat Satellite.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `treeLabel` - Label for the kickstart tree
- string `basePath` - Path to the base or root of the kickstart tree
- string `channelLabel` - Label of channel to associate with kickstart tree
- string `installType` - Label for `KickstartInstallType` (`rhel_3`, `rhel_4`, `rhel_5`, `rhel_6`, `fedora_18`).

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

CHAPTER 23. NAMESPACE: ORG

Contains methods to access common organization management functions available from the web interface

Namespace: org

23.1. METHOD: CREATE

Description

Create a new organization and associated administrator account

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `orgName` - Organization name; must meet same criteria as in the web UI
- string `adminLogin` - New administrator login name
- string `adminPassword` - New administrator password
- string `prefix` - New administrator's prefix; must match one of the values available in the web UI (i.e. Dr., Mr., Mrs., Sr., etc.)
- string `firstName` - New administrator's first name
- string `lastName` - New administrator's first name
- string `email` - New administrator's e-mail
- boolean `usePamAuth` - `true` if PAM authentication should be used for the new administrator account

Returns

The following return values are available for this method:

- struct - organization info
 - int `id`
 - string `name`
 - int `active_users` - Number of active users in the organization
 - int `systems` - Number of systems in the organization
 - int `trusts` - Number of trusted organizations
 - int `system_groups` - Number of system groups in the organization (optional)
 - int `activation_keys` - Number of activation keys in the organization (optional)

- `int kickstart_profiles` - Number of kickstart profiles in the organization (optional)
- `int configuration_channels` - Number of configuration channels in the organization (optional)

23.2. METHOD: DELETE

Description

Delete an organization. The default organization (i.e. `orgId=1`) cannot be deleted.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int orgId`

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

23.3. METHOD: GETCRASHFILESIZELIMIT

Description

Get the organization wide crash file size limit. The limit value must be a non-negative number. Zero means no limit.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int orgId`

Returns

The following return values are available for this method:

- `int` - Crash file size limit

23.4. METHOD: GETDETAILS

Description

The detailed information about an organization given the organization ID.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int orgId`

Returns

The following return values are available for this method:

- **struct** - organization info
 - **int id**
 - **string name**
 - **int active_users** - Number of active users in the organization
 - **int systems** - Number of systems in the organization
 - **int trusts** - Number of trusted organizations
 - **int system_groups** - Number of system groups in the organization (optional)
 - **int activation_keys** - Number of activation keys in the organization (optional)
 - **int kickstart_profiles** - Number of kickstart profiles in the organization (optional)
 - **int configuration_channels** - Number of configuration channels in the organization (optional)

23.5. METHOD: GETDETAILS

Description

The detailed information about an organization given the organization name.

Parameters

The following parameters are available for this method:

- **string sessionKey**
- **string name**

Returns

The following return values are available for this method:

- **struct** - organization info
 - **int id**
 - **string name**
 - **int active_users** - Number of active users in the organization
 - **int systems** - Number of systems in the organization
 - **int trusts** - Number of trusted organizations
 - **int system_groups** - Number of system groups in the organization (optional)
 - **int activation_keys** - Number of activation keys in the organization (optional)

- `int kickstart_profiles` - Number of kickstart profiles in the organization (optional)
- `int configuration_channels` - Number of configuration channels in the organization (optional)

23.6. METHOD: GETPOLICYFORSCAPFILEUPLOAD

Description

Get the status of SCAP detailed result file upload settings for the given organization.

Parameters

The following paramaters are available for this method:

- `string sessionKey`
- `int orgId`

Returns

The following return values are available for this method:

- `struct - scap_upload_info`
 - `boolean enabled` - Aggregation of detailed SCAP results is enabled
 - `int size_limit` - Limit (in Bytes) for a single SCAP file upload

23.7. METHOD: GETPOLICYFORSCAPRESULTDELETION

Description

Get the status of SCAP result deletion settings for the given organization.

Parameters

The following paramaters are available for this method:

- `string sessionKey`
- `int orgId`

Returns

The following return values are available for this method:

- `struct - scap_deletion_info`
 - `boolean enabled` - Deletion of SCAP results is enabled
 - `int retention_period` - Period (in days) after which a scan can be deleted (if enabled).

23.8. METHOD: ISCRASHREPORTINGENABLED

Description

Get the status of crash reporting settings for the given organization. Returns `true` if enabled, `false` otherwise.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`

Returns

The following return values are available for this method:

- boolean - Get the status of crash reporting settings

23.9. METHOD: ISCRASHFILEUPLOADENABLED

Description

Get the status of crash file upload settings for the given organization. Returns `true` if enabled, `false` otherwise.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`

Returns

The following return values are available for this method:

- boolean - Get the status of crash file upload settings

23.10. METHOD: LISTORGS

Description

Returns the list of organizations.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- struct - organization info
 - int `id`
 - string `name`

- int `active_users` - Number of active users in the organization
- int `systems` - Number of systems in the organization
- int `trusts` - Number of trusted organizations
- int `system_groups` - Number of system groups in the organization (optional)
- int `activation_keys` - Number of activation keys in the organization (optional)
- int `kickstart_profiles` - Number of kickstart profiles in the organization (optional)
- int `configuration_channels` - Number of configuration channels in the organization (optional)

23.11. METHOD: LISTSOFTWAREENTITLEMENTS

Description

List software entitlement allocation information across all organizations. Caller must be an administrator.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - entitlement usage
 - string `label`
 - string `name`
 - int `free`
 - int `used`
 - int `allocated`
 - int `unallocated`
 - int `free_flex`
 - int `used_flex`
 - int `allocated_flex`
 - int `unallocated_flex`

23.12. METHOD: LISTSOFTWAREENTITLEMENTS

Description

List each organization's allocation of a given software entitlement. Organizations with no allocation are not present in the list. A value of **-1** indicates unlimited entitlements.



NOTE

This method is deprecated. Use the `listSoftwareEntitlements(string sessionKey, string label, boolean includeUnentitled)` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - Software entitlement label

Returns

The following return values are available for this method:

- array:
 - struct - entitlement usage
 - int `org_id`
 - string `org_name`
 - int `allocated`
 - int `unallocated`
 - int `used`
 - int `free`

23.13. METHOD: LISTSOFTWAREENTITLEMENTS

Description

List each organization's allocation of a given software entitlement. A value of **-1** indicates unlimited entitlements.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - Software entitlement label

- boolean `includeUnentitled` - If `true`, the result includes both organizations that have the entitlement as well as those that do not; otherwise, the result will only include organizations that have the entitlement

Returns

The following return values are available for this method:

- array:
 - struct - entitlement usage
 - int `org_id`
 - string `org_name`
 - int `allocated`
 - int `unallocated`
 - int `used`
 - int `free`

Available since: 10.4

23.14. METHOD: LISTSOFTWAREENTITLEMENTSFORORG

Description

List an organization's allocation of each software entitlement. A value of `-1` indicates unlimited entitlements.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`

Returns

The following return values are available for this method:

- array:
 - struct - entitlement usage
 - string `label`
 - string `name`
 - int `allocated`
 - int `unallocated`
 - int `free`

- `int used`
- `int allocated_flex`
- `int unallocated_flex`
- `int free_flex`
- `int used_flex`

23.15. METHOD: LISTSYSTEMENTITLEMENTS

Description

Lists system entitlement allocation information across all organizations. Caller must be an administrator.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - entitlement usage`
 - `string label`
 - `string name`
 - `int free`
 - `int used`
 - `int allocated`
 - `int unallocated`
 - `int free_flex`
 - `int used_flex`
 - `int allocated_flex`
 - `int unallocated_flex`

23.16. METHOD: LISTSYSTEMENTITLEMENTS

Description

List each organization's allocation of a system entitlement. If the organization has no allocation for a particular entitlement, it does not appear in the list.

**NOTE**

This method is deprecated. Use the `listSystemEntitlements(string sessionKey, string label, boolean includeUnentitled)` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label`

Returns

The following return values are available for this method:

- array:
 - struct - entitlement usage
 - int `org_id`
 - string `org_name`
 - int `allocated`
 - int `unallocated`
 - int `used`
 - int `free`

23.17. METHOD: LISTSYSTEMENTITLEMENTS**Description**

List each organization's allocation of a system entitlement.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label`
- boolean `includeUnentitled` - If `true`, the result includes both organizations that have the entitlement as well as those that do not; otherwise, the result only includes organizations that have the entitlement

Returns

The following return values are available for this method:

- array:
 - struct - entitlement usage

- `int org_id`
- `string org_name`
- `int allocated`
- `int unallocated`
- `int used`
- `int free`

Available since: 10.4

23.18. METHOD: LISTSYSTEMENTITLEMENTSFORORG

Description

List an organization's allocation of each system entitlement.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int orgId`

Returns

The following return values are available for this method:

- `array:`
 - `struct - entitlement usage`
 - `string label`
 - `string name`
 - `int free`
 - `int used`
 - `int allocated`
 - `int unallocated`

23.19. METHOD: LISTUSERS

Description

Returns the list of users in a given organization.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`

Returns

The following return values are available for this method:

- array:
 - struct - user
 - string `login`
 - string `login_uc`
 - string `name`
 - string `email`
 - boolean `is_org_admin`

23.20. METHOD: MIGRATESYSTEMS

Description

Migrate systems from one organization to another. If a Satellite administrator executes this method, the systems migrate from their current organization to the organization specified by the `toOrgId`. If organization administrator executea this method, the systems existing in the same organization as that administrator migrate to the organization specified by the `toOrgId`. In any scenario, the origin and destination organizations are defined in a trust.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `toOrgId` - ID of the destination organization for the system(s) migration
- array:
 - int `systemId`

Returns

The following return values are available for this method:

- array:
 - int `serverIdMigrated`

23.21. METHOD: SETCRASHFILESIZELIMIT

Description

Set the organization-wide crash file size limit. The limit value must be non-negative. Zero means no limit.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`
- int `limit` - The limit to set (non-negative value)

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

23.22. METHOD: SETCRASHREPORTING

Description

Set the status of crash reporting settings for the given organization. Disabling crash reporting automatically disables crash file upload.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`
- boolean `enable` - Use `true` or `false` to enable or disable respectively

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

23.23. METHOD: SETCRASHFILEUPLOAD

Description

Set the status of crash file upload settings for the given organization. Modifying the settings is possible as long as crash reporting is enabled.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`
- boolean `enable` - Use `true` or `false` to enable or disable respectively

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

23.24. METHOD: SETPOLICYFORSCAPFILEUPLOAD

Description

Set the status of SCAP detailed result file upload settings for the given organization.

Parameters

The following paramaters are available for this method:

- string `sessionKey`
- int `orgId`
- struct - `scap_upload_info`
 - boolean `enabled` - Aggregation of detailed SCAP results is enabled
 - int `size_limit` - Limit (in Bytes) for a single SCAP file upload

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

23.25. METHOD: SETPOLICYFORSCAPRESULTDELETION

Description

Set the status of SCAP result deletion settings for the given organization.

Parameters

The following paramaters are available for this method:

- string `sessionKey`
- int `orgId`
- struct - `scap_deletion_info`
 - boolean `enabled` - Deletion of SCAP results is enabled
 - int `retention_period` - Period (in days) after which a scan can be deleted (if enabled).

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

23.26. METHOD: SETSOFTWAREENTITLEMENTS

Description

Set an organization's entitlement allocation for the given software entitlement. If increasing the entitlement allocation, the default organization (i.e. `orgId=1`) must have a sufficient number of free entitlements.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`
- string `label` - Software entitlement label
- int `allocation`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

23.27. METHOD: SETSOFTWAREFLEXENTITLEMENTS

Description

Set an organization's flex entitlement allocation for the given software entitlement. If increasing the flex entitlement allocation, the default organization (i.e. `orgId=1`) must have a sufficient number of free flex entitlements.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`
- string `label` - Software entitlement label
- int `allocation`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

23.28. METHOD: SETSYSTEMENTITLEMENTS

Description

Set an organization's entitlement allocation for the given software entitlement. If increasing the entitlement allocation, the default organization (i.e. `orgId=1`) must have a sufficient number of free entitlements.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int orgId`
- `string label` - System entitlement label. Valid values include:
 - `enterprise_entitled`
 - `monitoring_entitled`
 - `provisioning_entitled`
 - `virtualization_host`
 - `virtualization_host_platform`
- `int allocation`

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

23.29. METHOD: UPDATENAME

Description

Updates the name of an organization/

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int orgId`
- `string name` - Organization name; must meet same criteria as in the web UI

Returns

The following return values are available for this method:

- `struct` - organization info
 - `int id`
 - `string name`
 - `int active_users` - Number of active users in the organization
 - `int systems` - Number of systems in the organization
 - `int trusts` - Number of trusted organizations
 - `int system_groups` - Number of system groups in the organization (optional)

- `int activation_keys` - Number of activation keys in the organization (optional)
- `int kickstart_profiles` - Number of kickstart profiles in the organization (optional)
- `int configuration_channels` - Number of configuration channels in the organization (optional)

CHAPTER 24. NAMESPACE: ORG.TRUSTS

Contains methods to access common organization trust information available from the web interface.

Namespace: `org.trusts`

24.1. METHOD: ADDTRUST

Description

Add an organization to the list of trusted organizations.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int orgId`
- `int trustOrgId`

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

24.2. METHOD: GETDETAILS

Description

The trust details about an organization given the organization's ID.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int trustOrgId` - ID of the trusted organization

Returns

The following return values are available for this method:

- `struct` - org trust details
 - `dateTime.iso8601 created` - Date the organization was created
 - `dateTime.iso8601 trusted_since` - Date the organization was defined as trusted
 - `int channels_provided` - Number of channels provided by the organization
 - `int channels_consumed` - Number of channels consumed by the organization
 - `int systems_migrated_to` - Number of systems migrated to the organization

- `int systems_migrated_from` - Number of systems migrated from the organization

24.3. METHOD: LISTCHANNELSCONSUMED

Description

Lists all software channels that the organization given may consume from the user's organization.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int trustOrgId` - ID of the trusted organization

Returns

The following return values are available for this method:

- array:
 - struct - channel info
 - `int channel_id`
 - `string channel_name`
 - `int packages`
 - `int systems`

24.4. METHOD: LISTCHANNELSPROVIDED

Description

Lists all software channels that the defined organization is providing to the user's organization.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int trustOrgId` - ID of the trusted organization

Returns

The following return values are available for this method:

- array:
 - struct - channel info
 - `int channel_id`
 - `string channel_name`
 - `int packages`

- `int systems`

24.5. METHOD: LISTORGS

Description

List all organizations the user's organization trusts.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - trusted organizations`
 - `int org_id`
 - `string org_name`
 - `int shared_channels`

24.6. METHOD: LISTSYSTEMSAFFECTED

Description

Get a list of systems within the *trusted* organization that would be affected if the *trust* relationship was removed. This basically lists systems that are sharing at least one package.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int orgId`
- `string trustOrgId`

Returns

The following return values are available for this method:

- `array:`
 - `struct - affected systems`
 - `int systemId`
 - `string systemName`

24.7. METHOD: LISTTRUSTS

Description

Returns the list of trusted organizations.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`

Returns

The following return values are available for this method:

- array:
 - struct - trusted organizations
 - int `orgId`
 - string `orgName`
 - bool `trustEnabled`

24.8. METHOD: REMOVETRUST

Description

Remove an organization to the list of trusted organizations.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `orgId`
- int `trustOrgId`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

CHAPTER 25. NAMESPACE: PACKAGES

Methods to retrieve information about the Packages contained within this server.

Namespace: packages

25.1. METHOD: FINDBYNVREA

Description

Looks up the details for packages with the given name, version, release, architecture label, and optionally epoch.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `name`
- string `version`
- string `release`
- string `epoch` - If set to something other than empty string, strict matching will be used and the epoch string must be correct; if set to an empty string and if the epoch is null or there is only one NVRA combination, the NVRA combination is returned (Empty string is recommended)
- string `archLabel`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - string `name`
 - string `version`
 - string `release`
 - string `epoch`
 - int `id`
 - string `arch_label`
 - string `path` - The path on that file system that the package resides
 - string `provider` - The provider of the package, determined by the gpg key it was signed with
 - `dateTime.iso8601 last_modified`

25.2. METHOD: GETDETAILS

Description

Retrieve details for the package with the ID.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `packageId`

Returns

The following return values are available for this method:

- struct - package
 - int `id`
 - string `name`
 - string `epoch`
 - string `version`
 - string `release`
 - string `arch_label`
 - array `providing_channels`
 - string `ChannelLabel` providing this package
 - string `build_host`
 - string `description`
 - string `checksum`
 - string `checksum_type`
 - string `vendor`
 - string `summary`
 - string `cookie`
 - string `license`
 - string `file`
 - string `build_date`
 - string `last_modified_date`

- string `size`
- string `path` - The path on the Satellite's file system that the package resides
- string `payload_size`

25.3. METHOD: GETPACKAGE

Description

Retrieve the package file associated with a package.



IMPORTANT

Some larger package files result in an exception when attempting to download over XML-RPC. For larger package file, use the `getPackageUrl` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `package_id`

Returns

The following return values are available for this method:

- binary object - package file

25.4. METHOD: GETPACKAGEURL

Description

Retrieve the URL used to download a package. The URL expires after a certain time period.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `package_id`

Returns

The following return values are available for this method:

- string - The download url

25.5. METHOD: LISTCHANGELOG

Description

List the change log for a package.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `packageId`

Returns

The following return values are available for this method:

- string

25.6. METHOD: LISTDEPENDENCIES

Description

List the dependencies for a package.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `packageId`

Returns

The following return values are available for this method:

- array:
 - struct - dependency
 - string "dependency"
 - string "dependency_type" - One of the following:
 - requires
 - conflicts
 - obsoletes
 - provides
 - string "dependency_modifier"

25.7. METHOD: LISTFILES

Description

List the files associated with a package.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int packageId`

Returns

The following return values are available for this method:

- array:
 - struct - file info
 - `string path`
 - `string type`
 - `string last_modified_date`
 - `string checksum`
 - `string checksum_type`
 - `int size`
 - `string linkto`

25.8. METHOD: LISTPROVIDINGCHANNELS

Description

List the channels that provide the a package.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int packageId`

Returns

The following return values are available for this method:

- array:
 - struct - channel
 - `string label`
 - `string parent_label`
 - `string name`

25.9. METHOD: LISTPROVIDINGERRATA

Description

List the errata providing a package.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int packageId`

Returns

The following return values are available for this method:

- `array:`
 - `struct - errata`
 - `string advisory`
 - `string issue_date`
 - `string last_modified_date`
 - `string update_date`
 - `string synopsis`
 - `string type`

25.10. METHOD: REMOVEPACKAGE

Description

Remove a package from the satellite.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int packageId`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise

CHAPTER 26. NAMESPACE: PACKAGES.PROVIDER

Methods to retrieve information about Package Providers associated with packages.

Namespace: `packages.provider`

26.1. METHOD: ASSOCIATEKEY

Description

Associate a package security key and with the package provider. If the provider or key does not exist, it is created. User executing the request must be an administrator.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `providerName` - The provider name
- string `key` - The actual key
- string `type` - The type of the key. Only `gpg` is supported

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

26.2. METHOD: LIST

Description

List all Package Providers. User executing the request must be an administrator.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - package provider
 - string `name`
 - array `keys`
 - struct - package security key
 - string `key`

- string type

26.3. METHOD: LISTKEYS

Description

List all security keys associated with a package provider. User executing the request must be an administrator.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `providerName` - The provider name

Returns

The following return values are available for this method:

- array:
 - struct - package security key
 - string `key`
 - string `type`

CHAPTER 27. NAMESPACE: PACKAGES.SEARCH

Methods to interface to package search capabilities in search server.

Namespace: `packages . search`

27.1. METHOD: ADVANCED

Description

Advanced method to search lucene indexes with a passed in query written in Lucene Query Parser syntax. Lucene Query Parser syntax is defined [at lucene.apache.org](http://lucene.apache.org). Fields searchable for packages include: `name`, `epoch`, `version`, `release`, `arch`, `description`, `summary`.

Example 27.1. Lucene query example

```
name:kernel AND version:2.6.18 AND -description:devel
```

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `luceneQuery` - A query written in the form of Lucene Query Parser Syntax

Returns

The following return values are available for this method:

- array:
 - struct - package overview
 - int `id`
 - string `name`
 - string `summary`
 - string `description`
 - string `version`
 - string `release`
 - string `arch`
 - string `epoch`
 - string `provider`

27.2. METHOD: ADVANCEDWITHACTKEY

Description

Advanced method to search lucene indexes with a passed in query written in Lucene Query Parser syntax, additionally this method will limit results to those which are associated with a given activation key. Lucene Query Parser syntax is defined [at lucene.apache.org](http://lucene.apache.org). Fields searchable for packages include: **name**, **epoch**, **version**, **release**, **arch**, **description**, **summary**.

Example 27.2. Lucene query example

```
name:kernel AND version:2.6.18 AND -description:devel
```

Parameters

The following parameters are available for this method:

- string **sessionKey**
- string **luceneQuery** - A query written in the form of Lucene Query Parser Syntax
- string **actKey** - Activation key containing packages

Returns

The following return values are available for this method:

- array:
 - struct - package overview
 - int **id**
 - string **name**
 - string **summary**
 - string **description**
 - string **version**
 - string **release**
 - string **arch**
 - string **epoch**
 - string **provider**

27.3. METHOD: ADVANCEDWITHCHANNEL

Description

Advanced method to search lucene indexes with a passed in query written in Lucene Query Parser syntax, additionally this method will limit results to those which are in the passed in channel label. Lucene Query Parser syntax is defined [at lucene.apache.org](http://lucene.apache.org). Fields searchable for packages include: **name**, **epoch**, **version**, **release**, **arch**, **description**, **summary**.

■

Example 27.3. Lucene query example

```
name:kernel AND version:2.6.18 AND -description:devel
```

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `luceneQuery` - A query written in the form of Lucene Query Parser Syntax
- string `channelLabel` - Channel Label

Returns

The following return values are available for this method:

- array:
 - struct - package overview
 - int `id`
 - string `name`
 - string `summary`
 - string `description`
 - string `version`
 - string `release`
 - string `arch`
 - string `epoch`
 - string `provider`

27.4. METHOD: NAME**Description**

Search the Lucene package indexes for all packages that match the given name.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `name` - Package name to search

Returns

The following return values are available for this method:

- array:
 - struct - package overview
 - int `id`
 - string `name`
 - string `summary`
 - string `description`
 - string `version`
 - string `release`
 - string `arch`
 - string `epoch`
 - string `provider`

27.5. METHOD: NAMEANDDESCRIPTION

Description

Search the Lucene package indexes for all packages that match the given query in name or description.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `query` - Text to match in package name or description

Returns

The following return values are available for this method:

- array:
 - struct - package overview
 - int `id`
 - string `name`
 - string `summary`
 - string `description`
 - string `version`
 - string `release`
 - string `arch`

- `string epoch`
- `string provider`

27.6. METHOD: NAMEANDSUMMARY

Description

Search the Lucene package indexes for all packages that match the given query in name or summary.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string query` - Text to match in package name or summary

Returns

The following return values are available for this method:

- `array:`
 - `struct - package overview`
 - `int id`
 - `string name`
 - `string summary`
 - `string description`
 - `string version`
 - `string release`
 - `string arch`
 - `string epoch`
 - `string provider`

CHAPTER 28. NAMESPACE: PREFERENCES.LOCALE

Provides methods to access and modify user locale information

Namespace: `preferences.locale`

28.1. METHOD: LISTLOCALES

Description

Returns a list of all understood locales. Use results as input for `setLocale`.

Returns

The following return values are available for this method:

- array:
 - string - Locale code

28.2. METHOD: LISTTIMEZONES

Description

Returns a list of all understood timezones. Use results as input for `setTimeZone`.

Returns

The following return values are available for this method:

- array:
 - struct - timezone
 - int `time_zone_id` - Unique identifier for timezone
 - string `olson_name` - Name as identified by the Olson database

28.3. METHOD: SETLOCALE

Description

Set a user's locale.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- string `locale` - Locale to set (from `listLocales`)

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

28.4. METHOD: SETTIMEZONE

Description

Set a user's timezone.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string login` - User's login name
- `int tzid` - Timezone ID (from `listTimeZones`)

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

CHAPTER 29. NAMESPACE: PROXY

Provides methods to activate/deactivate a proxy server.

Namespace: proxy

29.1. METHOD: ACTIVATEPROXY

Description

Activates the proxy identified from the given client certificate e.g. `systemid` file.

Parameters

The following parameters are available for this method:

- string `systemid` - `systemid` file
- string `version` - Version of proxy to be registered

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

29.2. METHOD: CREATEMONITORINGSCOUT

Description

Create Monitoring Scout for proxy.

Parameters

The following parameters are available for this method:

- string `systemid` - `systemid` file

Returns

The following return values are available for this method:

- string

Available since: 10.7

29.3. METHOD: DEACTIVATEPROXY

Description

Deactivates the proxy identified from the given client certificate e.g. `systemid` file.

Parameters

The following parameters are available for this method:

- string `systemid` - `systemid` file

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

29.4. METHOD: ISPROXY

Description

Test for proxy on the system identified by the given client certificate e.g. `systemid` file.

Parameters

The following parameters are available for this method:

- string `systemid` - `systemid` file

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

29.5. METHOD: LISTAVAILABLEPROXYCHANNELS

Description

List available version of proxy channel for system identified from the given client certificate e.g. `systemid` file.

Parameters

The following parameters are available for this method:

- string `systemid` - `systemid` file

Returns

The following return values are available for this method:

- array:
 - string - `version`

Available since: 10.5

CHAPTER 30. NAMESPACE: SATELLITE

Provides methods to obtain details on the Satellite.

Namespace: `satellite`

30.1. METHOD: GETCERTIFICATEEXPIRATIONDATE

Description

Retrieves the certificate expiration date of the activated certificate.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `dateTime.iso8601`

30.2. METHOD: ISMONITORINGENABLED

Description

Indicates if monitoring is enabled on the Satellite.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `boolean true` if monitoring is enabled

30.3. METHOD: ISMONITORINGENABLEDBYSYSTEMID

Description

Indicates if monitoring is enabled on the Satellite.

Parameters

The following parameters are available for this method:

- `string systemid - systemid file`

Returns

The following return values are available for this method:

- `boolean true` if monitoring is enabled

30.4. METHOD: LISTENTITLEMENTS

Description

Lists all channel and system entitlements for the organization associated with the user executing the request.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- struct - channel or system entitlements
 - array `system`
 - struct - system entitlement
 - string `label`
 - string `name`
 - int `used_slots`
 - int `free_slots`
 - int `total_slots`
 - array `channel`
 - struct - channel entitlement
 - string `label`
 - string `name`
 - int `used_slots`
 - int `free_slots`
 - int `total_slots`
 - int `used_flex`
 - int `free_flex`
 - int `total_flex`

30.5. METHOD: LISTPROXIES

Description

List the Proxies within the user's organization.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last time the server completed a successful check-in

CHAPTER 31. NAMESPACE: SCHEDULE

Methods to retrieve information about scheduled actions

Namespace: `schedule`

31.1. METHOD: ARCHIVEACTIONS

Description

Archive all actions in the given list.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - int `action` - The action ID

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

31.2. METHOD: CANCELACTIONS

Description

Cancel all actions in a given list. If provided an invalid action, the Satellite cancels none of the provided actions.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - int `action` - The action ID

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

31.3. METHOD: DELETEACTIONS

Description

Delete all archived actions in a given list.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `int action` - The action ID

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

31.4. METHOD: LISTALLACTIONS

Description

Returns a list of all actions. This includes completed, in progress, failed and archived actions.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct` - action
 - `int id` - Action ID
 - `string name` - Action name
 - `string type` - Action type
 - `string scheduler` - The user that scheduled the action (optional)
 - `dateTime.iso8601 earliest` - The action's earliest date and time for the action's next performance
 - `int completedSystems` - Number of systems that completed the action
 - `int failedSystems` - Number of systems that failed the action
 - `int inProgressSystems` - Number of systems in progress

31.5. METHOD: LISTARCHIVEDACTIONS

Description

Returns a list of archived actions.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - action`
 - `int id` - Action ID
 - `string name` - Action name
 - `string type` - Action type
 - `string scheduler` - The user that scheduled the action (optional)
 - `dateTime.iso8601 earliest` - The action's earliest date and time for the action's next performance
 - `int completedSystems` - Number of systems that completed the action
 - `int failedSystems` - Number of systems that failed the action
 - `int inProgressSystems` - Number of systems in progress

31.6. METHOD: LISTCOMPLETEDACTIONS

Description

Returns a list of successfully completed actions.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - action`
 - `int id` - Action ID
 - `string name` - Action name
 - `string type` - Action type
 - `string scheduler` - The user that scheduled the action (optional)

- `dateTime.iso8601 earliest` - The action's earliest date and time for the action's next performance
- `int completedSystems` - Number of systems that completed the action
- `int failedSystems` - Number of systems that failed the action
- `int inProgressSystems` - Number of systems in progress

31.7. METHOD: LISTCOMPLETEDSYSTEMS

Description

Returns a list of systems that have completed a specific action.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string actionId`

Returns

The following return values are available for this method:

- `array:`
 - `struct - system`
 - `int server_id`
 - `string server_name` - Server name
 - `string base_channel` - Base channel the server uses
 - `dateTime.iso8601 timestamp` - The completion time for the action
 - `string message` - Optional message containing details on the execution of the action; for example, if the action failed, this contains the failure text

31.8. METHOD: LISTFAILED ACTIONS

Description

Returns a list of failed actions.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - action
 - int `id` - Action ID
 - string `name` - Action name
 - string `type` - Action type
 - string `scheduler` - The user that scheduled the action (optional)
 - `dateTime.iso8601 earliest` - The action's earliest date and time for the action's next performance
 - int `completedSystems` - Number of systems that completed the action
 - int `failedSystems` - Number of systems that failed the action
 - int `inProgressSystems` - Number of systems in progress

31.9. METHOD: LISTFAILEDSYSTEMS

Description

Returns a list of systems that have failed a specific action.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `actionId`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `server_id`
 - string `server_name` - Server name
 - string `base_channel` - Base channel the server uses
 - `dateTime.iso8601 timestamp` - The completion time for the action
 - string `message` - Optional message containing details on the execution of the action; for example, if the action failed, this contains the failure text

31.10. METHOD: LISTINPROGRESSACTIONS

Description

Returns a list of actions that are in progress.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - action`
 - `int id` - Action ID
 - `string name` - Action name
 - `string type` - Action type
 - `string scheduler` - The user that scheduled the action (optional)
 - `dateTime.iso8601 earliest` - The action's earliest date and time for the action's next performance
 - `int completedSystems` - Number of systems that completed the action
 - `int failedSystems` - Number of systems that failed the action
 - `int inProgressSystems` - Number of systems in progress

31.11. METHOD: LISTINPROGRESSSYSTEMS

Description

Returns a list of systems that have a specific action in progress.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string actionId`

Returns

The following return values are available for this method:

- `array:`
 - `struct - system`
 - `int server_id`
 - `string server_name` - Server name

- string `base_channel` - Base channel the server uses
- `dateTime.iso8601 timestamp` - The completion time for the action
- string `message` - Optional message containing details on the execution of the action; for example, if the action failed, this contains the failure text

31.12. METHOD: RESCHEDULEACTIONS

Description

Reschedule all actions in the given list.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - int `action` - The action ID
- boolean `onlyFailed` - `true` to only reschedule failed actions, `false` to reschedule all

Returns

The following return values are available for this method:

- int - `1` on success, exception thrown otherwise

CHAPTER 32. NAMESPACE: SYNC.MASTER

Contains methods to set up information about known Master Satellites for use on the Slave Satellite side of Inter-Satellite Synchronization.

Namespace: `sync.master`

32.1. METHOD: ADDTOMASTER

Description

Add a single organization to the list of those the specified Master has exported to this Slave.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int id` - ID of the desired Master
- `struct` - master-org details
 - `int masterOrgId`
 - `string masterOrgName`
 - `int localOrgId`

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

32.2. METHOD: CREATE

Description

Create a new Master, known to this Slave.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string label` - The Master's fully-qualified domain name

Returns

The following return values are available for this method:

- `struct` - IssMaster info
 - `int id`
 - `string label`

32.3. METHOD: DELETE

Description

Remove the specified Master.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - ID of the Master to remove

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

32.4. METHOD: GETDEFAULTMASTER

Description

Return the current default master for this slave

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- struct - `IssMaster` info
 - int `id`
 - string `label`
 - string `caCert`
 - boolean `isCurrentMaster`

32.5. METHOD: GETMASTER

Description

Get information about the specified Master.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - ID of the desired Master

Returns

The following return values are available for this method:

- struct - IssMaster info
 - int `id`
 - string `label`

32.6. METHOD: GETMASTERBYLABEL

Description

Get information about the specified Master using a label to identify.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `label` - Label of the desired Master

Returns

The following return values are available for this method:

- struct - IssMaster info
 - int `id`
 - string `label`

32.7. METHOD: GETMASTERORGS

Description

List all organizations the specified Master has exported to this Slave.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - ID of the desired Master

Returns

The following return values are available for this method:

- array:
 - struct - IssMasterOrg info
 - int `masterOrgId`
 - string `masterOrgName`

- `int localOrgId`

32.8. METHOD: GETMASTERS

Description

Get all Masters known by a Slave.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - IssMaster info`
 - `int id`
 - `string label`

32.9. METHOD: MAKEDEFAULT

Description

Make the specified Master the default for this Slave's satellite synchronization.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int id` - ID of the Master to make the default

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise.

32.10. METHOD: MAPTOLOCAL

Description

Add a single organizations to the list of those the specified Master has exported to this Slave.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int masterId` - ID of the desired Master

- `int masterOrgId` - ID of the desired Master
- `int localOrgId` - ID of the desired Master

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

32.11. METHOD: SETCACERT

Description

Set the Certificate Authority for the specified Master.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int id` - ID of the Master to affect
- `string caCertFilename` - path to specified Master's CA cert

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise.

32.12. METHOD: SETMASTERORGS

Description

List all organizations the specified Master has exported to this Slave.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int id` - ID of the desired Master
- `array`:
 - `struct` - master-org details
 - `int masterOrgId`
 - `string masterOrgName`
 - `int localOrgId`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

32.13. METHOD: UNSETDEFAULTMASTER

Description

Make this slave have no default Master for Satellite synchronization.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

32.14. METHOD: UPDATE

Description

Updates the label of the specified Master.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int id` - ID of the Master to update
- `string label` - Desired new label

Returns

The following return values are available for this method:

- `struct - IssMaster` info
 - `int id`
 - `string label`

CHAPTER 33. NAMESPACE: SYNC.SLAVE

Contains methods to set up information about allowed Red Hat Satellite Slave Servers, for use on the Red Hat Satellite Master Server side of Inter-Satellite Synchronization.

Namespace: `sync.slave`

33.1. METHOD: CREATE

Description

Create a new Slave, known to this Slave.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `slave` - Slave's fully-qualified domain name
- boolean `enabled` - Defines if the Slave is active and communicating
- boolean `allowAllOrgs` - Defines whether to export all organizations to this slave

Returns

The following return values are available for this method:

- struct - `IssSlave` info
 - int `id`
 - string `slave`
 - boolean `enabled`
 - boolean `allowAllOrgs`

33.2. METHOD: DELETE

Description

Remove the specified Slave

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - ID of the Slave to remove

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

33.3. METHOD: GETALLOWEDORGS

Description

Get all Slaves this Master knows.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - ID of the desired Slave

Returns

The following return values are available for this method:

- array:
 - int - IDs of allowed organizations

33.4. METHOD: GETSLAVE

Description

Find a Slave by specifying its ID.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `id` - ID of the desired Slave

Returns

The following return values are available for this method:

- struct - `IssSlave` info
 - int `id`
 - string `slave`
 - boolean `enabled`
 - boolean `allowAllOrgs`

33.5. METHOD: GETSLAVEBYNAME

Description

Find a Slave by specifying its fully-qualified domain name.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string fqdn` - Domain name of the desired Slave

Returns

The following return values are available for this method:

- `struct` - `IssSlave` info
 - `int id`
 - `string slave`
 - `boolean enabled`
 - `boolean allowAllOrgs`

33.6. METHOD: GETSLAVES

Description

Get all slaves this master knows.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array`:
 - `struct` - `IssSlave` info
 - `int id`
 - `string slave`
 - `boolean enabled`
 - `boolean allowAllOrgs`

33.7. METHOD: SETALLOWEDORGS

Description

Get all the Slaves this Master knows.

Parameters

The following parameters are available for this method:

- `string sessionKey`

- `int id` - ID of the desired Slave
- `array`:
 - `int` - List of revisions to delete

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

33.8. METHOD: UPDATE

Description

Updates the label of the specified Slave

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int id` - ID of the Slave to update
- `boolean enabled` - Defines if the Slave is active and communicating
- `boolean allowAllOrgs` - Defines whether to export all organizations to this slave

Returns

The following return values are available for this method:

- `struct` - `IssSlave` info
 - `int id`
 - `string slave`
 - `boolean enabled`
 - `boolean allowAllOrgs`

CHAPTER 34. NAMESPACE: SYSTEM

Provides methods to access and modify registered system.

Namespace: `system`

34.1. METHOD: ADDENTITLEMENTS

Description

Add add-on entitlements to a server. Entitlements a server already has are quietly ignored.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `array:`
 - `string - entitlementLabel`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise

34.2. METHOD: ADDNOTE

Description

Add a new note to the given server.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `string subject` - The note's subject
- `string body` - Content of the note

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise

34.3. METHOD: APPLYERRATA

Description

Schedules an action to apply errata updates to a system.



NOTE

This method is deprecated. Use the `system.scheduleApplyErrata(string sessionKey, int serverId, array[int errataId])` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- array:
 - int - `errataId`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.4. METHOD: COMPAREPACKAGEPROFILE

Description

Compare a system's packages against a package profile. In the result returned, `this_system` represents the server provided as an input and `other_system` represents the profile provided as an input.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `profileLabel`

Returns

The following return values are available for this method:

- array:
 - struct - Package Metadata
 - int `package_name_id`
 - string `package_name`
 - string `package_arch`

- string **this_system** - Version of package on this system
- string **other_system** - Version of package on the other system
- int **comparison**
 - **0** - No difference
 - **1** - Package on this system only
 - **2** - Newer package version on this system
 - **3** - Package on other system only
 - **4** - Newer package version on other system

34.5. METHOD: COMPAREPACKAGES

Description

Compares the packages installed on two systems.

Parameters

The following parameters are available for this method:

- string **sessionKey**
- int **thisServerId**
- int **otherServerId**

Returns

The following return values are available for this method:

- array:
 - struct - Package Metadata
 - int **package_name_id**
 - string **package_name**
 - string **package_arch**
 - string **this_system** - Version of package on this system
 - string **other_system** - Version of package on the other system
 - int **comparison**
 - **0** - No difference
 - **1** - Package on this system only
 - **2** - Newer package version on this system

- 3 - Package on other system only
- 4 - Newer package version on other system

34.6. METHOD: CONVERTTOFLEXENTITLEMENT

Description

Converts the given list of systems for a given channel family to use the flex entitlement.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - int `serverId`
- string `channelFamilyLabel`

Returns

The following return values are available for this method:

- int - The total the number of systems converted to use flex entitlement

34.7. METHOD: CREATEPACKAGEPROFILE

Description

Create a new stored package profile from a systems installed package list.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `profileLabel`
- string `description`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.8. METHOD: CREATESYSTEMRECORD

Description

Creates a cobbler system record with the specified kickstart label

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `ksLabel`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.9. METHOD: DELETECUSTOMVALUES

Description

Delete the custom values defined for the custom system information keys provided from the given system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- array:
 - string - `customInfoLabel`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.10. METHOD: DELETEGUESTPROFILES

Description

Delete the specified list of guest profiles for a given host.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `hostId`
- array:
 - string - `guestNames`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.11. METHOD: DELETENOTE**Description**

Deletes the given note from the server.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- int `noteId`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.12. METHOD: DELETENOTES**Description**

Deletes all notes from the server.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.13. METHOD: DELETEPACKAGEPROFILE**Description**

Delete a package profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`

- `int profileId`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise

34.14. METHOD: DELETESYSTEM

Description

Delete a system given its client certificate.

Parameters

The following parameters are available for this method:

- `string systemid` - `systemid` file

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise

Available since: 10.10

34.15. METHOD: DELETESYSTEM

Description

Delete a system given its server ID synchronously.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise

34.16. METHOD: DELETESYSTEMS

Description

Delete systems given a list of system IDs asynchronously.

Parameters

The following parameters are available for this method:

- `string sessionKey`

- array:
 - int - `serverId`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.17. METHOD: DELETETAGFROMSNAPSHOT

Description

Deletes tag from system snapshot.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `tagName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.18. METHOD: DOWNLOADSYSTEMID

Description

Get the system ID file for a given server.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- string

34.19. METHOD: GETCONNECTIONPATH

Description

Get the list of proxies the given system connects through in order to reach the server.

Parameters

The following parameters are available for this method:

- **string sessionKey**
- **int serverId**

Returns

The following return values are available for this method:

- **array:**
 - **struct** - proxy connection path details
 - **int position** - Position of proxy in chain. The proxy the system connects directly to is listed in position 1
 - **int id** - Proxy system ID
 - **string hostname** - Proxy host name

34.20. METHOD: GETCPU

Description

Gets the CPU information of a system.

Parameters

The following parameters are available for this method:

- **string sessionKey**
- **int serverId**

Returns

The following return values are available for this method:

- **struct** - CPU
 - **string cache**
 - **string family**
 - **string mhz**
 - **string flags**
 - **string model**
 - **string vendor**
 - **string arch**
 - **string stepping**
 - **string count**

- `int socket_count` (if available)

34.21. METHOD: GETCUSTOMVALUES

Description

Get the custom data values defined for the server.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `struct` - custom value
 - `string` - Custom info label

34.22. METHOD: GETDETAILS

Description

Get system details.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `struct` - server details
 - `int id` - System ID
 - `string profile_name`
 - `string base_entitlement` - System's base entitlement label; either `enterprise_entitled` or `sw_mgr_entitled`
 - `array string`
 - `add-on_entitlements` - System's add-on entitlements
 - Labels including `monitoring_entitled`, `provisioning_entitled`, `virtualization_host`, or `virtualization_host_platform`

- boolean `auto_update` - `true` if system has auto errata updates enabled
- string `release` - The Operating System release i.e. `4AS`, `5Server`, etc
- string `address1`
- string `address2`
- string `city`
- string `state`
- string `country`
- string `building`
- string `room`
- string `rack`
- string `description`
- string `hostname`
- `dateTime.iso8601` `last_boot`
- string `osa_status` - Either `unknown`, `offline`, or `online`
- boolean `lock_status` - `true` indicates the system is locked. `false` indicates that the system is unlocked
- string `virtualization` - Virtualization type for virtual guests only (optional)

34.23. METHOD: GETDEVICES

Description

Gets a list of devices for a system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - device
 - string `device` (optional)

- string `device_class` - Includes **CDROM, FIREWIRE, HD, USB, VIDEO, OTHER**, etc
- string `driver`
- string `description`
- string `bus`
- string `pcitype`

34.24. METHOD: GETDMI

Description

Gets the DMI information of a system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- struct - DMI
 - string `vendor`
 - string `system`
 - string `product`
 - string `asset`
 - string `board`
 - string `bios_release` (optional)
 - string `bios_vendor` (optional)
 - string `bios_version` (optional)

34.25. METHOD: GETENTITLEMENTS

Description

Gets the entitlements for a given server.

Parameters

The following parameters are available for this method:

- string `sessionKey`

- `int serverId`

Returns

The following return values are available for this method:

- `array:`
 - `string - entitlement_label`

34.26. METHOD: GETEVENTHISTORY

Description

Returns a list of history items associated with the system, ordered from newest to oldest. Note that the details might be empty for events scheduled against the system (as compared to instant). For more information about such events, see the `system.listSystemEvents` operation.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `array:`
 - `struct - History Event`
 - `dateTime.iso8601 completed` - Date the event occurred (optional)
 - `string summary` - Summary of the event
 - `string details` - Details of the event

34.27. METHOD: GETID

Description

Get system IDs and last check-in information for the given system name.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string systemName`

Returns

The following return values are available for this method:

- `array:`

- struct - system
 - int id
 - string name
 - dateTime.iso8601 last_checkin - Time of server's last successful check-in

34.28. METHOD: GETMEMORY

Description

Gets the memory information for a system.

Parameters

The following parameters are available for this method:

- string sessionKey
- int serverId

Returns

The following return values are available for this method:

- struct - memory
 - int ram - The amount of physical memory in MB
 - int swap - The amount of swap space in MB

34.29. METHOD: GETNAME

Description

Get system name and last check-in information for the given system ID.

Parameters

The following parameters are available for this method:

- string sessionKey
- string serverId

Returns

The following return values are available for this method:

- struct - name info
 - int id - Server ID
 - string name - Server name
 - dateTime.iso8601 last_checkin - Time of server's last successful check-in

34.30. METHOD: GETNETWORK

Description

Get the addresses and hostname for a given server.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- struct - network info
 - string `ip` - IPv4 address of server
 - string `ip6` - IPv6 address of server
 - string `hostname` - Hostname of server

34.31. METHOD: GETNETWORKDEVICES

Description

Returns the network devices for the given server.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - network device
 - string `ip` - IP address assigned to this network device
 - string `interface` - Network interface assigned to device e.g. eth0
 - string `netmask` - Network mask assigned to device
 - string `hardware_address` - Hardware Address of device.
 - string `module` - Network driver used for this device.
 - string `broadcast` - Broadcast address for device.

- array `ipv6` - List of IPv6 addresses
- array:
 - struct - ipv6 address
 - string `address` - IPv6 address of this network device
 - string `netmask` - IPv6 netmask of this network device
 - string `scope` - IPv6 address scope

34.32. METHOD: GETREGISTRATIONDATE

Description

Returns the date the system was registered.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- `dateTime.iso8601` - The date the system was registered in local time

34.33. METHOD: GETRELEVANTERRATA

Description

Returns a list of all errata that are relevant to the system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `update_date` - Erratum update date

- string `advisory_synopsis` - Summary of the erratum.
- string `advisory_type` - Type label, such as "Security", "Bug Fix", or "Enhancement"
- string `advisory_name` - Name, such as RHSA

34.34. METHOD: GETRELEVANTERRATABYTYPE

Description

Returns a list of all errata of the specified type relevant to the system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `advisoryType` - Type of advisory; one of the following: "Security Advisory", "Product Enhancement Advisory", or "Bug Fix Advisory".

Returns

The following return values are available for this method:

- array:
 - struct - errata
 - int `id` - Errata ID
 - string `date` - Erratum creation date
 - string `update_date` - Erratum update date
 - string `advisory_synopsis` - Summary of the erratum.
 - string `advisory_type` - Type label, such as "Security", "Bug Fix", or "Enhancement"
 - string `advisory_name` - Name, such as RHSA

34.35. METHOD: GETRUNNINGKERNEL

Description

Returns the running kernel of the given system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- string

34.36. METHOD: GETSCRIPTACTIONDETAILS

Description

Returns script details for script run actions

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `actionId` - ID of the script run action

Returns

The following return values are available for this method:

- struct - Script details
 - int `id` - Action ID
 - string `content` - Script content
 - string `run_as_user` - Run as user
 - string `run_as_group` - Run as group
 - int `timeout` - Timeout in seconds
 - array:
 - struct - script result
 - int `serverId` - ID of the server the script runs
 - `dateTime.iso8601` `startDate` - Time script began execution
 - `dateTime.iso8601` `stopDate` - Time script stopped execution
 - int `returnCode` - Script execution return code
 - string `output` - Output of the script (base64 encoded according to the `output_enc64` attribute)
 - boolean `output_enc64` - Identifies base64 encoded output
- result

34.37. METHOD: GETSCRIPTRESULTS

Description

Fetch results from a script execution. Returns an empty array if no results are yet available.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `actionId` - ID of the script run action

Returns

The following return values are available for this method:

- array:
 - struct - script result
 - int `serverId` - ID of the server the script runs
 - `dateTime.iso8601 startDate` - Time script began execution
 - `dateTime.iso8601 stopDate` - Time script stopped execution
 - int `returnCode` - Script execution return code
 - string `output` - Output of the script (base64 encoded according to the `output_enc64` attribute)
 - boolean `output_enc64` - Identifies base64 encoded output

34.38. METHOD: GETSUBSCRIBEDBASECHANNEL

Description

Provides the base channel of a given system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- struct - channel
 - int `id`
 - string `name`
 - string `label`
 - string `arch_name`

- string `summary`
- string `description`
- string `checksum_label`
- `dateTime.iso8601` `last_modified`
- string `maintainer_name`
- string `maintainer_email`
- string `maintainer_phone`
- string `support_policy`
- string `gpg_key_url`
- string `gpg_key_id`
- string `gpg_key_fp`
- `dateTime.iso8601` `yumrepo_last_sync` (optional)
- string `end_of_life`
- string `parent_channel_label`
- string `clone_original`
- array:
 - struct - `contentSources`
 - int `id`
 - string `label`
 - string `sourceUrl`
 - string `type`

34.39. METHOD: GETSYSTEMCURRENCYMULTIPLIERS

Description

Get the system currency score multipliers.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- Map of score multipliers

34.40. METHOD: GETSYSTEMCURRENCYSCORES

Description

Get the System Currency scores for all servers the user has access.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system currency
 - int `sid`
 - int - critical security errata count
 - int - important security errata count
 - int - moderate security errata count
 - int - low security errata count
 - int - bug fix errata count
 - int - enhancement errata count
 - int - system currency score

34.41. METHOD: GETUNSCHEDULEDERRATA

Description

Provides an array of errata applicable to a given system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - errata

- `int id` - Errata ID
- `string date` - Erratum creation date
- `string advisory_type` - Type of the advisory
- `string advisory_name` - Name of the advisory
- `string advisory_synopsis` - Summary of the erratum

34.42. METHOD: GETUUID

Description

Get the UUID from the given system ID.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

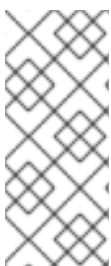
The following return values are available for this method:

- `string`

34.43. METHOD: GETVARIABLES

Description

Lists kickstart variables set in the system record for the specified server.



NOTE

This call assumes that a system record exists in cobbler for the given system and raises an XML-RPC fault if not the case. To create a system record over XML-RPC use `system.createSystemRecord`. To create a system record in the Web UI navigate to **System** → **Specified System** → **Provisioning** → **Select a Kickstart profile** → **Create Cobbler System Record**.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `struct` - System kickstart variables

- boolean `netboot` - netboot enabled
- array - kickstart variables
 - struct - kickstart variable
 - string `key`
 - string or int `value`

34.44. METHOD: ISNVREINSTALLED

Description

Check if the package with the given NVRE is installed on given system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `name` - Package name
- string `version` - Package version
- string `release` - Package release

Returns

The following return values are available for this method:

- 1 if package exists, 0 if not, and an exception is thrown if an error occurs

34.45. METHOD: ISNVREINSTALLED

Description

Is the package with the given NVRE installed on given system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `name` - Package name
- string `version` - Package version
- string `release` - Package release
- string `epoch` - Package epoch

Returns

The following return values are available for this method:

- 1 if package exists, 0 if not, and an exception is thrown if an error occurs

34.46. METHOD: LISTACTIVATIONKEYS

Description

List the activation keys registered with the system. An empty list is returned if an activation key is not used during registration.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - string - key

34.47. METHOD: LISTACTIVESYSTEMS

Description

Returns a list of active servers visible to the user.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.48. METHOD: LISTACTIVESYSTEMSDETAILS

Description

Given a list of server IDs, returns a list of active servers' details visible to the user.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `int - serverIds`

Returns

The following return values are available for this method:

- `array:`
 - `struct - server details`
 - `int id` - The server's ID
 - `string name` - The server's name
 - `dateTime.iso8601 last_checkin` - Last time server successfully checked in (in UTC)
 - `int ram` - The amount of physical memory in MB.
 - `int swap` - The amount of swap space in MB.
 - `struct network_devices` - The server's network devices
 - `struct - network device`
 - `string ip` - IP address assigned to this network device
 - `string interface` - Network interface assigned to device e.g. eth0
 - `string netmask` - Network mask assigned to device
 - `string hardware_address` - Hardware Address of device.
 - `string module` - Network driver used for this device.
 - `string broadcast` - Broadcast address for device.
 - `array ipv6` - List of IPv6 addresses
 - `array:`
 - `struct - ipv6 address`
 - `string address` - IPv6 address of this network device
 - `string netmask` - IPv6 netmask of this network device
 - `string scope` - IPv6 address scope

- **struct dmi_info** - The server's dmi info
 - **struct** - DMI
 - **string vendor**
 - **string system**
 - **string product**
 - **string asset**
 - **string board**
 - **string bios_release** - (optional)
 - **string bios_vendor** - (optional)
 - **string bios_version** - (optional)
- **struct cpu_info** - The server's cpu info
 - **struct** - CPU
 - **string cache**
 - **string family**
 - **string mhz**
 - **string flags**
 - **string model**
 - **string vendor**
 - **string arch**
 - **string stepping**
 - **string count**
 - **int socket_count**(if available)"
- **array subscribed_channels** - List of subscribed channels
 - **array:**
 - **struct** - channel
 - **int channel_id** - The channel ID
 - **string channel_label** - The channel label
- **array active_guest_system_IDs** - List of virtual guest system IDs for active guests
 - **array:**

- `int guest_id` - The guest's system ID

34.49. METHOD: LISTADMINISTRATORS

Description

Returns a list of users that can administer the system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `array:`
 - `struct - user`
 - `int id`
 - `string login`
 - `string login_uc` - Upper case version of the login
 - `boolean enabled` - `true` if user is enabled, `false` if the user is disabled

34.50. METHOD: LISTALLINSTALLABLEPACKAGES

Description

Get the list of all installable packages for a given system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `struct - package`
 - `string name`
 - `string version`
 - `string release`

- string `epoch`
- int `id`
- string `arch_label`

34.51. METHOD: LISTBASECHANNELS

Description

Returns a list of subscribable base channels.



NOTE

This method is deprecated. Use the `listSubscribableBaseChannels(string sessionId, int serverId)` method.

Parameters

The following parameters are available for this method:

- string `sessionId`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - channel
 - int `id` - Base channel ID
 - string `name` - Name of channel
 - string `label` - Label of channel
 - int `current_base` - 1 indicates it is the current base channel

34.52. METHOD: LISTCHILDCHANNELS

Description

Returns a list of subscribable child channels. This only shows channels the system is *not* currently subscribed.



NOTE

This method is deprecated. Use the `listSubscribableChildChannels(string sessionId, int serverId)` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - child channel
 - int `id`
 - string `name`
 - string `label`
 - string `summary`
 - string `has_license`
 - string `gpg_key_url`

34.53. METHOD: LISTDUPLICATESBYHOSTNAME

Description

List duplicate systems by hostname.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - Duplicate Group
 - string `hostname`
 - array `systems`
 - struct - system
 - int `systemId`
 - string `systemName`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.54. METHOD: LISTDUPLICATESBYIP

Description

List duplicate systems by IP Address.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - Duplicate Group
 - string `ip`
 - array `systems`
 - struct - system
 - int `systemId`
 - string `systemName`
 - dateTime.iso8601 `last_checkin` - Last successful server check-in

34.55. METHOD: LISTDUPLICATESBYMAC

Description

List duplicate systems by MAC Address.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - Duplicate Group
 - string `mac`
 - array `systems`
 - struct - system
 - int `systemId`

- `string systemName`
- `dateTime.iso8601 last_checkin` - Last successful server check-in

34.56. METHOD: LISTELIGIBLEFLEXGUESTS

Description

List eligible flex guests accessible to the user.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - channel family system group`
 - `int id`
 - `string label`
 - `string name`
 - `array:`
 - `int - systems`

34.57. METHOD: LISTEXTRAPACKAGES

Description

List extra packages for a system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `array:`
 - `struct - package`
 - `string name`
 - `string version`

- string `release`
- string `epoch` - Returned only if non-zero
- string `arch`
- date `installtime` - Returned only if known

34.58. METHOD: LISTFLEXGUESTS

Description

List flex guests accessible to the user

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - channel family system group
 - int `id`
 - string `label`
 - string `name`
 - array:
 - int - `systems`

34.59. METHOD: LISTGROUPS

Description

List the available groups for a given system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - system group

- `int id` - Server group ID
- `int subscribed` - `1` if the given server is subscribed to this server group, `0` otherwise
- `string system_group_name` - Name of the server group
- `string sgid` - Server group ID (deprecated)

34.60. METHOD: LISTINACTIVESYSTEMS

Description

Lists systems that have been inactive for the default period of inactivity.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - `int id`
 - `string name`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.61. METHOD: LISTINACTIVESYSTEMS

Description

Lists systems that have been inactive for the specified number of days.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int days`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - `int id`
 - `string name`

- `dateTime.iso8601 last_checkin` - Last successful server check-in

34.62. METHOD: LISTLATESTAVAILABLEPACKAGE

Description

Get the latest available version of a package for each system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `int - serverId`
- `string packageName`

Returns

The following return values are available for this method:

- `array:`
 - `struct - system`
 - `int id` - server ID
 - `string name` - server name
 - `struct package` - package structure
 - `struct - package`
 - `int id`
 - `string name`
 - `string version`
 - `string release`
 - `string epoch`
 - `string arch`

34.63. METHOD: LISTLATESTINSTALLABLEPACKAGES

Description

Get the list of latest installable packages for a given system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - `string name`
 - `string version`
 - `string release`
 - `string epoch`
 - `int id`
 - `string arch_label`

34.64. METHOD: LISTLATESTUPGRADABLEPACKAGES

Description

Get the list of latest upgradable packages for a given system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - `string name`
 - `string arch`
 - `string from_version`
 - `string from_release`
 - `string from_epoch`
 - `string to_version`

- `string to_release`
- `string to_epoch`
- `string to_package_id`

34.65. METHOD: LISTNEWERINSTALLEDPACKAGES

Description

Given a package name, version, release, and epoch, returns the list of packages installed on the system with the same name that are newer.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `string name` - Package name
- `string version` - Package version
- `string release` - Package release
- `string epoch` - Package epoch

Returns

The following return values are available for this method:

- `array:`
 - `struct` - package
 - `string name`
 - `string version`
 - `string release`
 - `string epoch`

34.66. METHOD: LISTNOTES

Description

Provides a list of notes associated with a system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- array:
 - struct - note details
 - int `id`
 - string `subject` - Subject of the note
 - string `note` - Contents of the note
 - int `system_id` - The ID of the system associated with the note
 - string `creator` - Creator of the note
 - date `updated` - Date of the last note update

34.67. METHOD: LISTOLDERINSTALLEDPACKAGES

Description

Given a package name, version, release, and epoch, returns the list of packages installed on the system with the same name that are older.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `name` - Package name
- string `version` - Package version
- string `release` - Package release
- string `epoch` - Package epoch

Returns

The following return values are available for this method:

- array:
 - struct - package
 - string `name`
 - string `version`
 - string `release`
 - string `epoch`

34.68. METHOD: LISTOUTOFDATESYSTEMS

Description

Returns list of systems needing package updates.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.69. METHOD: LISTPACKAGEPROFILES

Description

List the package profiles in this organization

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - package profile
 - int `id`
 - string `name`
 - string `channel`

34.70. METHOD: LISTPACKAGES

Description

List the installed packages for a given system. The attribute `installtime` is returned since API version 10.10.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - int `id`
 - string `name`
 - string `version`
 - string `release`
 - string `epoch`
 - string `arch`
 - date `installtime` - returned only if known

34.71. METHOD: LISTPACKAGESFROMCHANNEL

Description

Provides a list of packages installed on a system that are also contained in the given channel. The installed package list does not include architecture information before Red Hat Enterprise Linux 5, so it is architecture unaware. Red Hat Enterprise Linux 5 systems do upload the architecture information, and thus are architecture aware.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `channelLabel`

Returns

The following return values are available for this method:

- array:
 - struct - package
 - string `name`

- `string version`
- `string release`
- `string epoch`
- `int id`
- `string arch_label`
- `string path` - The path on that file system that the package resides
- `string provider` - The provider of the package, determined by its GPG key
- `dateTime.iso8601 last_modified`

34.72. METHOD: LISTPHYSICALSYSTEMS

Description

Returns a list of all physical servers visible to the user.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - system`
 - `int id`
 - `string name`
 - `dateTime.iso8601 last_checkin` - Last time server successfully checked in

34.73. METHOD: LISTSUBSCRIBABLEBASECHANNELS

Description

Returns a list of subscribable base channels.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- array:
 - struct - channel
 - int `id` - Base channel ID
 - string `name` - Name of channel
 - string `label` - Label of channel
 - int `current_base` - 1 indicates it is the current base channel

34.74. METHOD: LISTSUBSCRIBABLECHILDCHANNELS

Description

Returns a list of subscribable child channels. This only shows channels the system is *not* currently subscribed.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - child channel
 - int `id`
 - string `name`
 - string `label`
 - string `summary`
 - string `has_license`
 - string `gpg_key_url`

34.75. METHOD: LISTSUBSCRIBEDCHILDCHANNELS

Description

Returns a list of subscribed child channels.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `array:`
 - `struct - channel`
 - `int id`
 - `string name`
 - `string label`
 - `string arch_name`
 - `string summary`
 - `string description`
 - `string checksum_label`
 - `dateTime.iso8601 last_modified`
 - `string maintainer_name`
 - `string maintainer_email`
 - `string maintainer_phone`
 - `string support_policy`
 - `string gpg_key_url`
 - `string gpg_key_id`
 - `string gpg_key_fp`
 - `dateTime.iso8601 yumrepo_last_sync` (optional)
 - `string end_of_life`
 - `string parent_channel_label`
 - `string clone_original`
 - `array:`
 - `struct - contentSources`
 - `int id`
 - `string label`

- `string sourceUrl`
- `string type`

34.76. METHOD: LISTSYSTEMEVENTS

Description

List all system events for given server. This includes *all* events for the server since registration. This might require the caller to filter the results to fetch the desired events.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId` - ID of system

Returns

The following return values are available for this method:

- `array:`
 - `struct - action`
 - `int failed_count` - Number of times action failed
 - `string modified` - Date modified (deprecated by `modified_date`)
 - `dateTime.iso8601 modified_date` - Date modified
 - `string created` - Date created. (deprecated by `created_date`)
 - `dateTime.iso8601 created_date` - Date created
 - `string action_type`
 - `int successful_count` - Number of times action was successful
 - `string earliest_action` - Earliest date this action will occur
 - `int archived` - If this action is archived. (`1` or `0`)
 - `string scheduler_user` - available only if concrete user has scheduled the action
 - `string prerequisite` - Pre-requisite action (optional)
 - `string name` - Name of this action
 - `int id` - ID of this action
 - `string version` - Version of action

- string **completion_time** - The date and time the event was completed; format is YYYY-MM-dd hh:mm:ss.ms e.g. 2007-06-04 13:58:13.0 (optional) (deprecated by `completed_date`)
- `dateTime.iso8601` **completed_date** - The date and time the event was completed (optional)
- string **pickup_time** - The date and time of the action pick-up. Format is YYYY-MM-dd hh:mm:ss.ms e.g. 2007-06-04 13:58:13.0 (optional) (deprecated by `pickup_date`)
- `dateTime.iso8601` **pickup_date** - The date and time of the action pick-up. (optional)
- string **result_msg** - The result string after the action executes on the client machine (optional)
- array **additional_info** - This array contains additional information for the event, if available
 - struct - info
 - string **detail** - The detail provided depends on the specific event; for example, for a package event, this is the package name; for an errata event, this is the advisory name and synopsis; for a config file event, this is the path and optional revision information
 - string **result** - The result (if included) depends on the specific event; for example, for a package or errata event, no result is included; for a config file event, the result might include an error (if one occurred, such as the file was missing); in the case of a config file comparison, it might include the differences found

Available since: 10.8

34.77. METHOD: LISTSYSTEMS

Description

Returns a list of all servers visible to the user.

Parameters

The following parameters are available for this method:

- string **sessionKey**

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int **id**
 - string **name**
 - `dateTime.iso8601` **last_checkin** - Last successful server check-in

34.78. METHOD: LISTSYSTEMSWITHEXTRAPACKAGES

Description

List systems with extra packages.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - int `id` - System ID
 - string `name` - System profile name
 - int `extra_pkg_count` - Extra packages count

34.79. METHOD: LISTSYSTEMSWITHPACKAGE

Description

Lists the systems that have the given installed package.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `pid` - The package ID

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.80. METHOD: LISTSYSTEMSWITHPACKAGE

Description

Lists the systems that have the given installed package

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `name` - The package name
- string `version` - The package version
- string `release` - The package release

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.81. METHOD: LISTUNGROUPEDSYSTEMS

Description

List systems that are not associated with any system groups.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id` - Server ID
 - string `name`

34.82. METHOD: LISTUSERSYSTEMS

Description

List systems for a given user.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string login` - User's login name

Returns

The following return values are available for this method:

- array:
 - struct - system
 - `int id`
 - `string name`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.83. METHOD: LISTUSERSYSTEMS

Description

List systems for the logged in user.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - `int id`
 - `string name`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.84. METHOD: LISTVIRTUALGUESTS

Description

Lists the virtual guests for a given virtual host.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int sid` - The virtual host's ID

Returns

The following return values are available for this method:

- array:
 - struct - virtual system
 - int `id`
 - string `name`
 - string `guest_name` - The virtual guest name as provided by the virtual host
 - `dateTime.iso8601 last_checkin` - Last successful server check-in
 - string `uuid`

34.85. METHOD: LISTVIRTUALHOSTS

Description

Lists the virtual hosts visible to the user

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.86. METHOD: OBTAINREACTIVATIONKEY

Description

Obtains a reactivation key for this server.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- string

34.87. METHOD: OBTAINREACTIVATIONKEY

Description

Obtains a reactivation key for this server.

Parameters

The following parameters are available for this method:

- string `systemId` - `systemId` file

Returns

The following return values are available for this method:

- string

Available since: 10.10

34.88. METHOD: PROVISIONSYSTEM

Description

Provision a system using the specified kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId` - ID of the system to be provisioned
- string `profileName` - Kickstart profile to use

Returns

The following return values are available for this method:

- int - ID of the action scheduled, otherwise exception thrown on error

34.89. METHOD: PROVISIONSYSTEM

Description

Provision a system using the specified kickstart profile.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId` - ID of the system to be provisioned
- string `profileName` - Kickstart profile to use

- `dateTime.iso8601 earliestDate`

Returns

The following return values are available for this method:

- `int` - ID of the action scheduled, otherwise exception thrown on error

34.90. METHOD: PROVISIONVIRTUALGUEST

Description

Provision a guest on the host specified. Defaults to: `memory=512MB`, `vcpu=1`, `storage=3GB`, `mac_address=random`.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId` - ID of host to provision guest
- `string guestName`
- `string profileName` - Kickstart profile to use

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

34.91. METHOD: PROVISIONVIRTUALGUEST

Description

Provision a guest on the host specified. This schedules the guest for creation and begins the provisioning process when the host checks in. If OSAD is enabled, provisioning begins immediately. Defaults to `mac_address=random`.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId` - ID of host to provision guest
- `string guestName`
- `string profileName` - Kickstart profile to use
- `int memoryMb` - Memory to allocate to the guest
- `int vcpu` - Number of virtual CPUs to allocate to the guest
- `int storageGb` - Size of the guest's disk image

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.92. METHOD: PROVISIONVIRTUALGUEST

Description

Provision a guest on the host specified. This schedules the guest for creation and begins the provisioning process when the host checks in. If OSAD is enabled, provisioning begins immediately.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId` - ID of host to provision guest
- string `guestName`
- string `profileName` - Kickstart Profile to use
- int `memoryMb` - Memory to allocate to the guest
- int `vcpus` - Number of virtual CPUs to allocate to the guest
- int `storageGb` - Size of the guest's disk image
- string `macAddress` - macAddress to give the guest's virtual networking hardware

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.93. METHOD: REMOVEENTITLEMENTS

Description

Remove add-on entitlements from a server. Entitlements a server does not have are quietly ignored.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- array:
 - string - `entitlement_label`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.94. METHOD: SCHEDULEAPPLYERRATA

Description

Schedules an action to apply errata updates to multiple systems.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - int `serverId`
- array:
 - int `errataId`

Returns

The following return values are available for this method:

- array:
 - int `actionId` - The action ID of the scheduled action

Available since: 13.0

34.95. METHOD: SCHEDULEAPPLYERRATA

Description

Schedules an action to apply errata updates to multiple systems at a given date and time.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - int - `serverId`
- array:
 - int - `errataId`
- `dateTime.iso8601 earliestOccurrence`

Returns

The following return values are available for this method:

- array:
 - int - `actionId`

Available since: 13.0

34.96. METHOD: SCHEDULEAPPLYERRATA

Description

Schedules an action to apply errata updates to a system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- array:
 - int - `errataId`

Returns

The following return values are available for this method:

- array:
 - int - `actionId`

Available since: 13.0

34.97. METHOD: SCHEDULEAPPLYERRATA

Description

Schedules an action to apply errata updates to a system at a given date and time.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- array:
 - int - `errataId`
- `dateTime.iso8601 earliestOccurrence`

Returns

The following return values are available for this method:

- array:

- `int - actionId`

Available since: 13.0

34.98. METHOD: SCHEDULEGUESTACTION

Description

Schedules a guest action for the specified virtual guest for a given date and time.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int sid` - The system IG of the guest
- `string state` - One of the following actions: `start`, `suspend`, `resume`, `restart`, or `shutdown`
- `dateTime.iso8601 date` - The date and time to schedule the action

Returns

The following return values are available for this method:

- `int actionId` - The action ID of the scheduled action

34.99. METHOD: SCHEDULEGUESTACTION

Description

Schedules a guest action for the specified virtual guest for the current time.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int sid` - The system Id of the guest
- `string state` - One of the following actions: `start`, `suspend`, `resume`, `restart`, or `shutdown`

Returns

The following return values are available for this method:

- `int actionId` - The action ID of the scheduled action

34.100. METHOD: SCHEDULEHARDWAREREFRESH

Description

Schedule a hardware refresh for a system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `dateTime.iso8601 earliestOccurrence`

Returns

The following return values are available for this method:

- `int actionId` - The action ID of the scheduled action

Available since: 13.0

34.101. METHOD: SCHEDULEPACKAGEINSTALL

Description

Schedule package installation for a system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `array:`
 - `int - packageId`
- `dateTime.iso8601 earliestOccurrence`

Returns

The following return values are available for this method:

- `int actionId` - The action ID of the scheduled action

Available since: 13.0

34.102. METHOD: SCHEDULEPACKAGEREFRESH

Description

Schedule a package list refresh for a system.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `dateTime.iso8601 earliestOccurrence`

Returns

The following return values are available for this method:

- int - ID of the action scheduled, otherwise exception thrown on error

34.103. METHOD: SCHEDULEPACKAGEREMOVE

Description

Schedule package removal for a system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- array:
 - int - `packageId`
- `dateTime.iso8601 earliestOccurrence`

Returns

The following return values are available for this method:

- int - ID of the action scheduled, otherwise exception thrown on error

34.104. METHOD: SCHEDULEREBOOT

Description

Schedule a reboot for a system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- `dateTime.iso860 earliestOccurrence`

Returns

The following return values are available for this method:

- int `actionId` - The action ID of the scheduled action

Available since: 13.0

34.105. METHOD: SCHEDULESCRIPTRUN

Description

Schedule a script to run.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `int` - System IDs of the servers to run the script
- `string username` - User to run script
- `string groupname` - Group to run script
- `int timeout` - Seconds to allow the script to run before timeout
- `string script` - Contents of the script to run
- `dateTime.iso8601 earliestOccurrence` - Earliest the script can run

Returns

The following return values are available for this method:

- `int` - ID of the script run action. Used to fetch results with `system.getScriptResults`

34.106. METHOD: SCHEDULESCRIPTRUN

Description

Schedule a script to run.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId` - ID of the server to run the script
- `string username` - User to run script
- `string groupname` - Group to run script
- `int timeout` - Seconds to allow the script to run before timeout
- `string script` - Contents of the script to run
- `dateTime.iso8601 earliestOccurrence` - Earliest the script can run

Returns

The following return values are available for this method:

- `int` - ID of the script run action. Used to fetch results with `system.getScriptResults`

34.107. METHOD: SCHEDULESYNCPACKAGESWITHSYSTEM

Description

Synchronize packages from a source system to a target.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `targetServerId` - Target system to apply package changes
- int `sourceServerId` - Source system to retrieve package state
- array:
 - int - `packageId` - Package IDs to be synchronized
- `dateTime.iso8601 date` - Date to schedule action

Returns

The following return values are available for this method:

- int `actionId` - The action ID of the scheduled action

34.108. METHOD: SEARCHBYNAME

Description

Returns a list of system IDs whose name matches the supplied regular expression.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `regexp` - A regular expression

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last successful server check-in

34.109. METHOD: SETBASECHANNEL

Description

Assigns the server to a new baseChannel.

**NOTE**

This method is deprecated. Use the `system.setBaseChannel(string sessionKey, int serverId, string channelLabel)` method.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- int `channelId`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.110. METHOD: SETBASECHANNEL

Description

Assigns the server to a new base channel. If the user provides an empty string for the `channelLabel`, the current base channel and all child channels are removed from the system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `channelLabel`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.111. METHOD: SETCHILDCHANNELS

Description

Subscribe the given server to the child channels provided. This method will unsubscribe the server from any child channels currently subscribed, but that are not included in the list. The user provides either a list of channel IDs (int) or a list of channel labels (string) as input.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- array:
 - int (deprecated) or string - `channelId` (deprecated) or `channelLabel`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.112. METHOD: SETCUSTOMVALUES

Description

Set custom values for the specified server.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- struct - Map of custom labels to custom values
 - string - Custom information label
 - string `value`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.113. METHOD: SETDETAILS

Description

Set server details. All arguments are optional and are only modified if included in the struct.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId` - ID of server to look up details
- struct - server details
 - string `profile_name` - System's profile name

- string **base_entitlement** - System's base entitlement label; either **enterprise_entitled** or **sw_mgr_entitled**
- boolean **auto_errata_update** - **true** if system has auto errata updates enabled
- string **description** - System description
- string **address1** - System's address line 1
- string **address2** - System's address line 2
- string **city**
- string **state**
- string **country**
- string **building**
- string **room**
- string **rack**

Returns

The following return values are available for this method:

- int - **1** on success, exception thrown otherwise

34.114. METHOD: SETGROUPMEMBERSHIP

Description

Set a server's membership in a given group.

Parameters

The following parameters are available for this method:

- string **sessionKey**
- int **serverId**
- int **serverGroupId**
- boolean **member** - **1** to assign the given server to the given server group, **0** to remove the given server from the given server group

Returns

The following return values are available for this method:

- int - **1** on success, exception thrown otherwise

34.115. METHOD: SETGUESTCPUS

Description

Schedule an action of a guest's host to set that guest's CPU allocation.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int sid` - The guest's system ID
- `int numOfCpus` - The number of virtual CPUs to allocate to the guest

Returns

The following return values are available for this method:

- `int actionID` - The action ID for the schedule action on the host system

34.116. METHOD: SETGUESTMEMORY

Description

Schedule an action of a guest's host to set that guest's memory allocation

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int sid` - The guest's system ID
- `int memory` - The amount of memory to allocate to the guest

Returns

The following return values are available for this method:

- `int actionID` - The action ID for the schedule action on the host system

34.117. METHOD: SETLOCKSTATUS

Description

Set server lock status.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `boolean lockStatus` - `true` to lock the system, `false` to unlock the system

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.118. METHOD: SETPRIMARYINTERFACE

Description

Sets new primary network interface.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `interfaceName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.119. METHOD: SETPROFILENAME

Description

Set the profile name for the server.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `name` - Name of the profile

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.120. METHOD: SETVARIABLES

Description

Sets a list of kickstart variables in the cobbler system record for the specified server.

**NOTE**

This call assumes that a system record exists in cobbler for the given system and raises an XML-RPC fault if not the case. To create a system record over XML-RPC use `system.createSystemRecord`. To create a system record in the Web UI navigate to **System** → **Specified System** → **Provisioning** → **Select a Kickstart profile** → **Create Cobbler System Record**.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- boolean `netboot`
- array:
 - struct - kickstart variable
 - string `key`
 - string or int `value`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.121. METHOD: TAGLATESTSNAPSHOT**Description**

Tags latest system snapshot.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `tagName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.122. METHOD: UPGRADEENTITLEMENT**Description**

Adds an entitlement to a given server.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `entitlementName` - One of: `enterprise_entitled`, `provisioning_entitled`, `monitoring_entitled`, `nonlinux_entitled`, `virtualization_host`, or `virtualization_host_platform`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

34.123. METHOD: WHOREGISTERED

Description

Returns information about the user who registered the system

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `sid` - ID of the system in question

Returns

The following return values are available for this method:

- struct - user
 - int `id`
 - string `login`
 - string `login_uc` - Upper case version of the login
 - boolean `enabled` - `true` if user is enabled, `false` if the user is disabled

CHAPTER 35. NAMESPACE: SYSTEM.CONFIG

Provides methods to access and modify many aspects of configuration channels and server association.

Namespace: `system.config`

35.1. METHOD: ADDCHANNELS

Description

Given a list of servers and configuration channels, this method appends the configuration channels to either the top or the bottom, depending of the user's choice, of a system's subscribed configuration channels list. The list maintains the ordering of the configuration channels provided while adding. If the has previous subscribed to one of the configuration channels in the "add" list, the subscribed channel is ranked to the appropriate place.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `int` - IDs of the systems to add the channels
- `array:`
 - `string` - List of configuration channel labels in the ranked order
- `boolean addToTop`
 - `true` to prepend the given channels list to the top of the configuration channels list of a server
 - `false` to append the given channels list to the bottom of the configuration channels list of a server

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

35.2. METHOD: CREATEORUPDATEPATH

Description

Create a new text or binary file or directory with the given path, or update an existing path on a server.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

- **string path** - The configuration file/directory path
- **boolean isDir**
 - **true** if the path is a directory
 - **false** if the path is a file
- **struct - path info**
 - **string contents** - Contents of the file (text or base64 encoded if binary) (only for non-directories)
 - **boolean contents_enc64** - Identifies base64 encoded content (default: disabled, only for non-directories)
 - **string owner** - Owner of the file or directory
 - **string group** - Group name of the file or directory
 - **string permissions** - Octal file or directory permissions e.g. 644
 - **string macro-start-delimiter** - Config file macro end delimiter. Use null or empty string to accept the default (only for non-directories)
 - **string macro-end-delimiter** - Config file macro end delimiter. Use null or empty string to accept the default (only for non-directories)
 - **string selinux_ctx** - SELinux context (optional)
 - **int revision** - Next revision number, auto increment for null
 - **boolean binary** - Mark the binary content, if **true**, base64 encoded content is expected (only for non-directories)
- **int commitToLocal**
 - **1** to commit configuration files to the system's local override configuration channel
 - **0** to commit configuration files to the system's sandbox configuration channel

Returns

The following return values are available for this method:

- **struct - Configuration Revision information**
 - **string type**
 - file
 - directory
 - symlink
 - **string path** - File path

- string `target_path` - Symbolic link target file path; present for symbolic links only
- string `channel` - Channel name
- string `contents` - File contents (base64 encoded according to the `contents_enc64` attribute)
- boolean `contents_enc64` - Identifies base64 encoded content
- int `revision` - File revision
- `dateTime.iso8601` `creation` - Creation date
- `dateTime.iso8601` `modified` - Last modified date
- string `owner` - File Owner; present for files or directories only
- string `group` - File Group; present for files or directories only
- int `permissions` - File Permissions; present for files or directories only (deprecated)
- string `permissions_mode` - File Permissions; present for files or directories only
- string `selinux_ctx` - SELinux context (optional)
- boolean `binary` - `true` or `false`; present for files only
- string `md5` - File's MD5 signature; present for files only
- string `macro-start-delimiter` - Macro start delimiter for a config file; present for text files only
- string `macro-end-delimiter` - Macro end delimiter for a config file; present for text files only

Available since: 10.2

35.3. METHOD: CREATEORUPDATESYMLINK

Description

Create a new symbolic link with the given path, or update an existing path.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string `path` - The configuration file or directory path
- struct - path info
 - string `target_path` - The target path for the symbolic link

- string `selinux_ctx` - SELinux security context (optional)
- int `revision` - Next revision number, auto-increment for null
- int `commitToLocal`
 - 1 to commit configuration files to the system's local override configuration channel
 - 0 to commit configuration files to the system's sandbox configuration channel

Returns

The following return values are available for this method:

- struct - Configuration Revision information
 - string `type`
 - file
 - directory
 - symlink
 - string `path` - File path
 - string `target_path` - Symbolic link target file path; present for Symbolic links only.
 - string `channel` - Channel name
 - string `contents` - File contents (base64 encoded according to the `contents_enc64` attribute)
 - boolean `contents_enc64` - Identifies base64 encoded content
 - int `revision` - File revision
 - dateTime.iso8601 `creation` - Creation date
 - dateTime.iso8601 `modified` - Last modified date
 - string `owner` - File Owner; present for files or directories only
 - string `group` - File Group; present for files or directories only
 - int `permissions` - File Permissions (Deprecated); present for files or directories only
 - string `permissions_mode` - File Permissions; present for files or directories only
 - string `selinux_ctx` - SELinux context (optional)
 - boolean `binary` - `true` or `false`; present for files only.
 - string `md5` - File's MD5 signature; present for files only
 - string `macro-start-delimiter` - Macro start delimiter for a config file; present for text files only

- `string macro-end-delimiter` - Macro end delimiter for a config file; present for text files only

Available since: 10.2

35.4. METHOD: DELETEFILES

Description

Removes file paths from a local or sandbox channel of a server.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `array:`
 - `string` - paths to remove.
- `boolean deleteFromLocal`
 - `true` to delete configuration file paths from the system's local override configuration channel
 - `false` to delete configuration file paths from the system's sandbox configuration channel

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

35.5. METHOD: DEPLOYALL

Description

Schedules a deploy action for all the configuration files on the given list of systems.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `int` - id of the systems to schedule configuration files deployment
- `dateTime.iso8601 date` - Earliest date for the deploy action.

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

35.6. METHOD: LISTCHANNELS

Description

List all global configuration channels associated to a system in the order of their ranking.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`

Returns

The following return values are available for this method:

- `array`:
 - `struct` - Configuration Channel information
 - `int id`
 - `int orgId`
 - `string label`
 - `string name`
 - `string description`
 - `struct configChannelType`
 - `struct` - Configuration Channel Type information
 - `int id`
 - `string label`
 - `string name`
 - `int priority`

35.7. METHOD: LISTFILES

Description

Return the list of files in a given channel.

Parameters

The following parameters are available for this method:

- `string sessionKey`

- `int serverId`
- `int listLocal`
 - `1` to return configuration files in the system's local override configuration channel
 - `0` to return configuration files in the system's sandbox configuration channel

Returns

The following return values are available for this method:

- `array:`
 - `struct` - Configuration File information
 - `string type`
 - `file`
 - `directory`
 - `symlink`
 - `string path` - File Path
 - `string channel_label` - The label of the central configuration channel that has this file; this entry only shows if a central channel has not overwritten the file
 - `struct channel_type`
 - `struct` - Configuration channel type information
 - `int id`
 - `string label`
 - `string name`
 - `int priority`
 - `dateTime.iso8601 last_modified` - Last modified date

35.8. METHOD: LOOKUPFILEINFO

Description

Given a list of paths and a server, this method returns details about the latest revisions of the paths.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int serverId`
- `array:`

- string - Paths to query
- **int searchLocal**
 - **1** to search configuration file paths in the system's local override configuration or systems subscribed central channels
 - **0** to search configuration file paths in the system's sandbox configuration channel

Returns

The following return values are available for this method:

- array:
 - struct - Configuration Revision information
 - string **type**
 - file
 - directory
 - symlink
 - string **path** - File path
 - string **target_path** - Symbolic link target file path; present for symbolic links only.
 - string **channel** - Channel name
 - string **contents** - File contents (base64 encoded according to the **contents_enc64** attribute)
 - boolean **contents_enc64** - Identifies base64 encoded content
 - int **revision** - File revision
 - dateTime.iso8601 **creation** - Creation date
 - dateTime.iso8601 **modified** - Last modified date
 - string **owner** - File Owner; present for files or directories only
 - string **group** - File Group; present for files or directories only
 - int **permissions** - File Permissions ; present for files or directories only (deprecated)
 - string **permissions_mode** - File Permissions; present for files or directories only
 - string **selinux_ctx** - SELinux context (optional)
 - boolean **binary** - **true** or **false**; present for files only
 - string **md5** - File's MD5 signature; present for files only

- `string macro-start-delimiter` - Macro start delimiter for a config file; present for text files only
- `string macro-end-delimiter` - Macro end delimiter for a config file; present for text files only

Available since:10.2

35.9. METHOD: REMOVECHANNELS

Description

Remove config channels from the given servers.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `int` - The IDs of the systems to remove configuration channels
- `array:`
 - `string` - List of configuration channel labels to remove

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

35.10. METHOD: SETCHANNELS

Description

Replace the existing set of config channels on the given servers. Channels are ranked according to their order in the `configChannelLabels` array.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `array:`
 - `int` - IDs of the systems to set the channels
- `array:`
 - `string` - List of configuration channel labels in the ranked order

Returns

The following return values are available for this method:

- int - **1** on success, exception thrown otherwise

CHAPTER 36. NAMESPACE: SYSTEM.CRASH

Provides methods to access and modify software crash information.

Namespace: `system.crash`

36.1. METHOD: CREATECRASHNOTE

Description

Create a crash note.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int crashId`
- `string subject`
- `string details`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

36.2. METHOD: DELETECRASH

Description

Delete a crash with given crash ID.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int crashId`

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

36.3. METHOD: DELETECRASHNOTE

Description

Delete a crash note.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `crashNoteId`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

36.4. METHOD: GETCRASHCOUNTINFO

Description

Return date of last software crashes report for given system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- struct - Crash Count Information
 - int `total_count` - Total number of software crashes for a system
 - int `unique_count` - Number of unique software crashes for a system
 - `dateTime.iso8601 last_report` - Date of the last software crash report

36.5. METHOD: GETCRASHFILE

Description

Download a crash file.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `crashFileId`

Returns

The following return values are available for this method:

- base64 - base64 encoded crash file

36.6. METHOD: GETCRASHFILEURL

Description

Get a crash file download URL.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `crashFileId`

Returns

The following return values are available for this method:

- string - The crash file download URL

36.7. METHOD: GETCRASHNOTESFORCRASH

Description

List crash notes for crash.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `crashId`

Returns

The following return values are available for this method:

- array:
 - struct - `crashNote`
 - int `id`
 - string `subject`
 - string `details`
 - string `updated`

36.8. METHOD: GETCRASHOVERVIEW

Description

Get software crash overview.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - crash
 - `string uuid` - Crash UUID
 - `string component` - Package component (set if unique and non-empty)
 - `int crash_count` - Number of crashes occurred
 - `int system_count` - Number of systems affected
 - `dateTime.iso8601 last_report` - Last crash occurrence

36.9. METHOD: GETCRASHESBYUUID

Description

List software crashes with given UUID.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string uuid`

Returns

The following return values are available for this method:

- array:
 - struct - crash
 - `int server_id` - ID of the server the crash occurred
 - `int crash_id` - ID of the crash with given UUID
 - `int crash_count` - Number of times the crash with given UUID occurred
 - `string crash_component` - Crash component
 - `dateTime.iso8601 last_report` - Last crash occurrence

36.10. METHOD: LISTSYSTEMCRASHFILES

Description

Return list of crash files for given crash ID.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `crashId`

Returns

The following return values are available for this method:

- array:
 - struct - `crashFile`
 - int `id`
 - string `filename`
 - string `path`
 - int `filesize`
 - boolean `is_uploaded`
 - date `created`
 - date `modified`

36.11. METHOD: LISTSYSTEMCRASHES

Description

Return list of software crashes for a system.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`

Returns

The following return values are available for this method:

- array:
 - struct - `crash`
 - int `id`
 - string `crash`
 - string `path`
 - int `count`

- `string uuid`
- `string analyzer`
- `string architecture`
- `string cmdline`
- `string component`
- `string executable`
- `string kernel`
- `string reason`
- `string username`
- `date created`
- `date modified`

CHAPTER 37. NAMESPACE: SYSTEM.CUSTOMINFO

Provides methods to access and modify custom system information.

Namespace: `system.custominfo`

37.1. METHOD: CREATEKEY

Description

Create a new custom key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `keyLabel` - New key's label
- string `keyDescription` - New key's description

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

37.2. METHOD: DELETEKEY

Description

Delete an existing custom key and all systems' values for the key.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `keyLabel` - New key's label

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise.

37.3. METHOD: LISTALLKEYS

Description

List the custom information keys defined for the user's organization.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- `array:`
 - `struct - custom info`
 - `int id`
 - `string label`
 - `string description`
 - `int system_count`
 - `dateTime.iso8601 last_modified`

37.4. METHOD: UPDATEKEY

Description

Update description of a custom key.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string keyLabel` - Key to change
- `string keyDescription` - New key's description

Returns

The following return values are available for this method:

- `int - 1` on success, exception thrown otherwise.

CHAPTER 38. NAMESPACE: SYSTEM.PROVISIONING.SNAPSHOT

Provides methods to access and delete system snapshots.

Namespace: `system.provisioning.snapshot`

38.1. METHOD: ADDTAGTOSNAPSHOT

Description

Adds tag to snapshot.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int snapshotId` - ID of the snapshot
- `string tag` - Name of the snapshot tag

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

38.2. METHOD: DELETESNAPSHOT

Description

Deletes a snapshot with the given snapshot ID.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int snapshotId` - ID of snapshot to delete

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

Available since: 10.1

38.3. METHOD: DELETESNAPSHOTS

Description

Deletes all snapshots across multiple systems based on the given date criteria. For example:

- If the user provides `startDate` only, all snapshots created either on or after the date provided are removed
- If user provides `startDate` and `endDate`, all snapshots created on or between the dates provided are removed
- If the user doesn't provide a `startDate` and `endDate`, all snapshots are removed

Parameters

The following parameters are available for this method:

- string `sessionKey`
- struct - date details
 - `dateTime.iso8601 startDate` (Optional, unless `endDate` is provided)
 - `dateTime.iso8601 endDate` (Optional)

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

Available since: 10.1

38.4. METHOD: DELETESNAPSHOTS

Description

Deletes all snapshots for a given system based on the date criteria. For example,

- If the user provides `startDate` only, all snapshots created either on or after the date provided are removed
- If user provides `startDate` and `endDate`, all snapshots created on or between the dates provided are removed
- If the user doesn't provide a `startDate` and `endDate`, all snapshots associated with the server are removed

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `sid` - system ID of system to delete snapshots
- struct - date details
 - `dateTime.iso8601 startDate` (optional, unless `endDate` is provided)
 - `dateTime.iso8601 endDate` (optional)

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

Available since: 10.1

38.5. METHOD: LISTSNAPSHOTCONFIGFILES

Description

List the configuration files associated with a snapshot.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int snapId`

Returns

The following return values are available for this method:

- `array`:
 - `struct` - Configuration Revision information
 - `string type`
 - `file`
 - `directory`
 - `symlink`
 - `string path` - File path
 - `string target_path` - Symbolic link Target file path; present for symbolic links only
 - `string channel` - Channel name
 - `string contents` - File contents (base64 encoded according to the `contents_enc64` attribute)
 - `boolean contents_enc64` - Identifies base64 encoded content
 - `int revision` - File revision
 - `dateTime.iso8601 creation` - Creation date
 - `dateTime.iso8601 modified` - Last modified date
 - `string owner` - File owner; present for files or directories only
 - `string group` - File Group; present for files or directories only
 - `int permissions` - File Permissions; present for files or directories only (Deprecated)

- string `permissions_mode` - File Permissions; present for files or directories only
- string `selinux_ctx` - SELinux Context (optional)
- boolean `binary` - `true` or `false`; present for files only
- string `md5` - File's md5 signature. Present for files only
- string `macro-start-delimiter` - Macro start delimiter for a config file. Present for text files only
- string `macro-end-delimiter` - Macro end delimiter for a config file. Present for text files only

Available since: 10.2

38.6. METHOD: LISTSNAPSHOTPACKAGES

Description

List the packages associated with a snapshot.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `snapId`

Returns

The following return values are available for this method:

- array:
 - struct - package nvera
 - string `name`
 - string `epoch`
 - string `version`
 - string `release`
 - string `arch`

Available since: 10.1

38.7. METHOD: LISTSNAPSHOTS

Description

List snapshots for a given system. A user may optionally provide a start and end date to narrow the snapshots that are listed. For example:

- If the user provides `startDate` only, all snapshots created either on or after the date provided are returned
- If user provides `startDate` and `endDate`, all snapshots created on or between the dates provided are returned
- If the user doesn't provide a `startDate` and `endDate`, all snapshots associated with the server are returned

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- struct - date details
 - `dateTime.iso8601 startDate` (optional, unless `endDate` is provided)
 - `dateTime.iso8601 endDate` (optional)

Returns

The following return values are available for this method:

- array:
 - struct - server snapshot
 - int `id`
 - string `reason` - The reason for the snapshot's existence
 - `dateTime.iso8601 created`
 - array `channels`
 - string - Labels of channels associated with the snapshot
 - array `groups`
 - string - Names of server groups associated with the snapshot
 - array `entitlements`
 - string - Names of system entitlements associated with the snapshot
 - array `config_channels`
 - string - Labels of config channels the snapshot is associated with
 - array `tags`
 - string - Tag names associated with this snapshot
 - string `Invalid_reason` - If the snapshot is invalid, this is the reason (optional)

Available since: 10.1

CHAPTER 39. NAMESPACE: SYSTEM.SCAP

Provides methods to schedule SCAP scans and access the results.

Namespace: `system.scap`

39.1. METHOD: DELETEXCCDFSCAN

Description

Delete OpenSCAP XCCDF Scan from Satellite database. Note only SCAP scans that have passed their retention period can be deleted.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `Id` of XCCDF scan (`xid`).

Returns

The following return values are available for this method:

- boolean - indicates success of the operation.

39.2. METHOD: GETXCCDFSCANDetails

Description

Get details of given OpenSCAP XCCDF scan.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int - ID of XCCDF scan (`xid`).

Returns

The following return values are available for this method:

- struct - OpenSCAP XCCDF Scan
 - int `xid` - XCCDF TestResult ID
 - int `sid` - The server ID
 - int `action_id` - ID of the parent action
 - string `path` - Path to XCCDF document
 - string `oscap_parameters` - OSCAP command-line arguments
 - string `test_result` - Identifier of XCCDF TestResult

- string **benchmark** - Identifier of XCCDF Benchmark
- string **benchmark_version** - Version of the Benchmark
- string **profile** - Identifier of XCCDF Profile
- string **profile_title** - Title of XCCDF Profile
- dateTime.iso8601 **start_time** - Client machine time of scan start
- dateTime.iso8601 **end_time** - Client machine time of scan completion
- string **errors** - Stderr output of scan

39.3. METHOD: GETXCCDFSCANRULERESULTS

Description

Return a full list of **RuleResults** for given OpenSCAP XCCDF scan.

Parameters

The following parameters are available for this method:

- string **sessionKey**
- int - ID of XCCDF scan (xid)

Returns

The following return values are available for this method:

- array:
 - struct - OpenSCAP XCCDF RuleResult
 - string **idref** - **idref** from XCCDF document
 - string **result** - Result of evaluation
 - string **idents** - Comma separated list of XCCDF idents

39.4. METHOD: LISTXCCDFSCANS

Description

Return a list of finished OpenSCAP scans for a given system.

Parameters

The following parameters are available for this method:

- string **sessionKey**
- int **serverId**

Returns

The following return values are available for this method:

- array:
 - struct - OpenSCAP XCCDF Scan
 - int `xid` - XCCDF TestResult ID
 - string `profile` - XCCDF Profile
 - string `path` - Path to XCCDF document
 - `dateTime.iso8601` `completed` - Scan completion time

39.5. METHOD: SCHEDULEXCCDFSCAN

Description

Schedule OpenSCAP scan.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - int - `serverId`
- string - Path to xccdf content on targeted systems
- string - Additional parameters for OSCAP tool

Returns

The following return values are available for this method:

- int - ID of SCAP action created

39.6. METHOD: SCHEDULEXCCDFSCAN

Description

Schedule OpenSCAP scan.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- array:
 - int - `serverId`
- string - Path to xccdf content on targeted systems

- string - Additional parameters for OSCAP tool
- `dateTime.iso8601 date` - The date to schedule the action

Returns

The following return values are available for this method:

- int - ID of SCAP action created

39.7. METHOD: SCHEDULEXCCDFSCAN

Description

Schedule SCAP XCCDF scan.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string - Path to xccdf content on targeted system
- string - Additional parameters for OSCAP tool

Returns

The following return values are available for this method:

- int - ID of the SCAP action created

39.8. METHOD: SCHEDULEXCCDFSCAN

Description

Schedule SCAP XCCDF scan.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- int `serverId`
- string - Path to xccdf content on targeted system
- string - Additional parameters for OSCAP tool
- `dateTime.iso8601 date` - The date to schedule the action

Returns

The following return values are available for this method:

- int - ID of the scap action created

CHAPTER 40. NAMESPACE: SYSTEM.SEARCH

Provides methods to perform system search requests using the search server.

Namespace: `system.search`

40.1. METHOD: DEVICEDESCRIPTION

Description

List the systems which match the device description.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `searchTerm`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last time server attempted a successful check-in
 - string `hostname`
 - string `ip`
 - string `hw_description` - Hardware description if not null
 - string `hw_device_id` - Hardware device ID if not null
 - string `hw_vendor_id` - Hardware vendor ID if not null
 - string `hw_driver` - Hardware driver if not null

40.2. METHOD: DEVICEDRIVER

Description

List the systems that match this device driver.

Parameters

The following parameters are available for this method:

- string `sessionKey`

- string `searchTerm`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last time server attempted a successful check-in
 - string `hostname`
 - string `ip`
 - string `hw_description` - Hardware description if not null
 - string `hw_device_id` - Hardware device id if not null
 - string `hw_vendor_id` - Hardware vendor id if not null
 - string `hw_driver` - Hardware driver if not null

40.3. METHOD: DEVICEID

Description

List the systems which match this device ID.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `searchTerm`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last time server attempted a successful check-in
 - string `hostname`

- string `ip`
- string `hw_description` - Hardware description if not null
- string `hw_device_id` - Hardware device id if not null
- string `hw_vendor_id` - Hardware vendor id if not null
- string `hw_driver` - Hardware driver if not null

40.4. METHOD: DEVICEVENDORID

Description

List the systems which match this device Vendor ID.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `searchTerm`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - dateTime.iso8601 `last_checkin` - Last time server attempted a successful check-in
 - string `hostname`
 - string `ip`
 - string `hw_description` - Hardware description if not null
 - string `hw_device_id` - Hardware device id if not null
 - string `hw_vendor_id` - Hardware vendor id if not null
 - string `hw_driver` - Hardware driver if not null

40.5. METHOD: HOSTNAME

Description

List the systems which match this hostname.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `searchTerm`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - `dateTime.iso8601 last_checkin` - Last time server attempted a successful check-in
 - string `hostname`
 - string `ip`
 - string `hw_description` - Hardware description if not null
 - string `hw_device_id` - Hardware device id if not null
 - string `hw_vendor_id` - Hardware vendor id if not null
 - string `hw_driver` - Hardware driver if not null

40.6. METHOD: IP

Description

List the systems which match this IP.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `searchTerm`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`

- `dateTime.iso8601 last_checkin` - Last time server attempted a successful check-in
- `string hostname`
- `string ip`
- `string hw_description` - Hardware description if not null
- `string hw_device_id` - Hardware device id if not null
- `string hw_vendor_id` - Hardware vendor id if not null
- `string hw_driver` - Hardware driver if not null

40.7. METHOD: NAMEANDDESCRIPTION

Description

List the systems which match this name or description.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string searchTerm`

Returns

The following return values are available for this method:

- `array:`
 - `struct - system`
 - `int id`
 - `string name`
 - `dateTime.iso8601 last_checkin` - Last time server attempted a successful check-in
 - `string hostname`
 - `string ip`
 - `string hw_description` - Hardware description if not null
 - `string hw_device_id` - Hardware device id if not null
 - `string hw_vendor_id` - Hardware vendor id if not null
 - `string hw_driver` - Hardware driver if not null

40.8. METHOD: UUID

Description

List the systems which match this UUID.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `searchTerm`

Returns

The following return values are available for this method:

- array:
 - struct - system
 - int `id`
 - string `name`
 - dateTime.iso8601 `last_checkin` - Last time server attempted a successful check-in
 - string `hostname`
 - string `ip`
 - string `hw_description` - Hardware description if not null
 - string `hw_device_id` - Hardware device id if not null
 - string `hw_vendor_id` - Hardware vendor id if not null
 - string `hw_driver` - Hardware driver if not null

CHAPTER 41. NAMESPACE: SYSTEMGROUP

Provides methods to access and modify system groups.

Namespace: `systemgroup`

41.1. METHOD: ADDORREMOVEADMINS

Description

Add or remove administrators to or from the given group. Satellite and organization administrators are granted access to groups within their organization by default. Therefore, users with those roles are not included in the array provided. Caller must be an organization administrator.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string systemGroupName`
- `array:`
 - `string - loginName` - User's login name
- `int add` - `1` to add administrators, `0` to remove

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

41.2. METHOD: ADDORREMOVESYSTEMS

Description

Add or remove the given servers to a system group.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string systemGroupName`
- `array:`
 - `int - serverId`
- `boolean add` - `true` to add to the group, `false` to remove

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

41.3. METHOD: CREATE

Description

Create a new system group.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `name` - Name of the system group
- string `description` - Description of the system group

Returns

The following return values are available for this method:

- struct - system group
 - int `id`
 - string `name`
 - string `description`
 - int `org_id`
 - int `system_count`

41.4. METHOD: DELETE

Description

Delete a system group.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `systemGroupName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

41.5. METHOD: GETDETAILS

Description

Retrieve details of a system group based on its ID.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `int systemGroupId`

Returns

The following return values are available for this method:

- `struct - system group`
 - `int id`
 - `string name`
 - `string description`
 - `int org_id`
 - `int system_count`

41.6. METHOD: GETDETAILS

Description

Retrieve details of a system group based on its name.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string systemGroupName`

Returns

The following return values are available for this method:

- `struct - System Group`
 - `int id`
 - `string name`
 - `string description`
 - `int org_id`
 - `int system_count`

41.7. METHOD: LISTACTIVESYSTEMSINGROUP

Description

Lists active systems within a system group.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `systemGroupName`

Returns

The following return values are available for this method:

- array:
 - int - `server_id`

41.8. METHOD: LISTADMINISTRATORS

Description

Returns the list of users who can administer the given group. Caller must be a system group administrator or an organization administrator.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `systemGroupName`

Returns

The following return values are available for this method:

- array:
 - struct - user
 - int `id`
 - string `login`
 - string `login_uc` - Upper case version of the login
 - boolean `enabled` - `true` if user is enabled, `false` if the user is disabled

41.9. METHOD: LISTALLGROUPS

Description

Retrieve a list of system groups that are accessible by the logged in user.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system group
 - `int id`
 - `string name`
 - `string description`
 - `int org_id`
 - `int system_count`

41.10. METHOD: LISTGROUPSWITHNOASSOCIATEDADMINS

Description

Returns a list of system groups that do not have an administrator i.e. one who is not an organization administrator, as they have implicit access to system groups. Caller must be an organization administrator.

Parameters

The following parameters are available for this method:

- `string sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - system group
 - `int id`
 - `string name`
 - `string description`
 - `int org_id`
 - `int system_count`

41.11. METHOD: LISTINACTIVESYSTEMSINGROUP

Description

Lists inactive systems within a system group using a specified inactivity time.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `systemGroupName`
- int `daysInactive` - Number of days a system must not check in to be considered inactive

Returns

The following return values are available for this method:

- array:
 - int - `server_id`

41.12. METHOD: LISTINACTIVESYSTEMSINGROUP

Description

Lists inactive systems within a system group using the default 1-day threshold.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `systemGroupName`

Returns

The following return values are available for this method:

- array:
 - int - `server_id`

41.13. METHOD: LISTSYSTEMS

Description

Return a list of systems associated with this system group. User must have access to this system group.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `systemGroupName`

Returns

The following return values are available for this method:

- array:

- o struct - server details
 - int `id` - System ID
 - string `profile_name`
 - string `base_entitlement` - System's base entitlement label; either `enterprise_entitled` or `sw_mgr_entitled`
 - array string
 - `addon_entitlements` - System's addon entitlements
 - Labels including: `monitoring_entitled`, `provisioning_entitled`, `virtualization_host`, and `virtualization_host_platform`
 - boolean `auto_update` - `true` if system has auto errata updates enabled
 - string `release` - The operating system release i.e. `4AS`, `5Server`
 - string `address1`
 - string `address2`
 - string `city`
 - string `state`
 - string `country`
 - string `building`
 - string `room`
 - string `rack`
 - string `description`
 - string `hostname`
 - `dateTime.iso8601 last_boot`
 - string `osa_status` - Either `unknown`, `offline`, or `online`
 - boolean `lock_status` - `true` indicates that the system is locked. `false` indicates that the system is unlocked
 - string `virtualization` - Virtualization type for virtual guests only (optional)

41.14. METHOD: SCHEDULEAPPLYERRATATOACTIVE

Description

Schedules an action to apply errata updates to active systems from a group.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `systemGroupName`
- array:
 - int - `errataId`

Returns

The following return values are available for this method:

- array:
 - int `actionId` - The action ID of the scheduled action

41.15. METHOD: SCHEDULEAPPLYERRATATOACTIVE

Description

Schedules an action to apply errata updates to active systems from a group at a given date and time.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `systemGroupName`
- array:
 - int - `errataId`
- `dateTime.iso8601 earliestOccurrence`

Returns

The following return values are available for this method:

- array:
 - int `actionId` - The action ID of the scheduled action

41.16. METHOD: UPDATE

Description

Update an existing system group.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `systemGroupName`

- **string description**

Returns

The following return values are available for this method:

- **struct - system group**
 - **int id**
 - **string name**
 - **string description**
 - **int org_id**
 - **int system_count**

CHAPTER 42. NAMESPACE: USER

User namespace contains methods to access common user functions available from the web user interface.

Namespace: user

42.1. METHOD: ADDASSIGNEDSYSTEMGROUP

Description

Add system group to user's list of assigned system groups.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- string `serverGroupName`
- boolean `setDefault` - Defines if the system group is added to user's list of default system groups

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.2. METHOD: ADDASSIGNEDSYSTEMGROUPS

Description

Add system groups to user's list of assigned system groups.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- array:
 - string - `serverGroupName`
- boolean `setDefault` - Defines if system groups are also added to user's list of default system groups

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.3. METHOD: ADDDEFAULTSYSTEMGROUP

Description

Add system group to user's list of default system groups.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- string `serverGroupName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.4. METHOD: ADDDEFAULTSYSTEMGROUPS

Description

Add system groups to user's list of default system groups.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- array:
 - string - `serverGroupName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.5. METHOD: ADDROLE

Description

Adds a role to a user.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User login name to update

- string `role` - Role label to add. Can be any of: `satellite_admin`, `org_admin`, `channel_admin`, `config_admin`, `system_group_admin`, `activation_key_admin`, or `monitoring_admin`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.6. METHOD: CREATE

Description

Create a new user.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `desiredLogin` - Desired login name; fails if already in use
- string `desiredPassword`
- string `firstName`
- string `lastName`
- string `email` - User's e-mail address

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.7. METHOD: CREATE

Description

Create a new user.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `desiredLogin` - Desired login name; fails if already in use
- string `desiredPassword`
- string `firstName`
- string `lastName`

- `string email` - User's e-mail address
- `int usePamAuth` - `1` to use PAM authentication for this user, `0` otherwise

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

42.8. METHOD: DELETE

Description

Delete a user.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string login` - User login name to delete

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

42.9. METHOD: DISABLE

Description

Disable a user.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string login` - User login name to disable

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

42.10. METHOD: ENABLE

Description

Enable a user.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string login` - User login name to enable

Returns

The following return values are available for this method:

- `int` - 1 on success, exception thrown otherwise

42.11. METHOD: GETDETAILS

Description

Returns the details about a given user.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string login` - User's login name

Returns

The following return values are available for this method:

- `struct` - user details
 - `string first_names` (deprecated, use `first_name`)
 - `string first_name`
 - `string last_name`
 - `string email`
 - `int org_id`
 - `string prefix`
 - `string last_login_date`
 - `string created_date`
 - `boolean enabled` - `true` if user is enabled, `false` if the user is disabled
 - `boolean use_pam` - `true` if user is configured to use PAM authentication

42.12. METHOD: GETLOGGEDINTIME

Description

Returns the time user last logged in.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name

Returns

The following return values are available for this method:

- `dateTime.iso8601`

42.13. METHOD: LISTASSIGNABLEROLES

Description

Returns a list of user roles that this user can assign to others.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - string - role label

42.14. METHOD: LISTASSIGNEDSYSTEMGROUPS

Description

Returns the system groups that a user can administer.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name

Returns

The following return values are available for this method:

- array:
 - struct - system group
 - int `id`
 - string `name`
 - string `description`

- `int system_count`
- `int org_id` - Organization ID for this system group

42.15. METHOD: LISTDEFAULTSYSTEMGROUPS

Description

Returns a user's list of default system groups.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string login` - User's login name

Returns

The following return values are available for this method:

- `array:`
 - `struct` - system group
 - `int id`
 - `string name`
 - `string description`
 - `int system_count`
 - `int org_id` - Organization ID for this system group

42.16. METHOD: LISTROLES

Description

Returns a list of the user's roles.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string login` - User's login name

Returns

The following return values are available for this method:

- `array:`
 - `string` - role label

42.17. METHOD: LISTUSERS

Description

Returns a list of users in your organization.

Parameters

The following parameters are available for this method:

- string `sessionKey`

Returns

The following return values are available for this method:

- array:
 - struct - user
 - int `id`
 - string `login`
 - string `login_uc` - Upper case version of the login
 - boolean `enabled` - `true` if user is enabled, `false` if the user is disabled

42.18. METHOD: REMOVEASSIGNEDSYSTEMGROUP

Description

Remove system group from the user's list of assigned system groups.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- string `serverGroupName`
- boolean `setDefault` - Defines if the system group is removed from the user's list of default system groups

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.19. METHOD: REMOVEASSIGNEDSYSTEMGROUPS

Description

Remove system groups from a user's list of assigned system groups.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- array:
 - string - `serverGroupName`
- boolean `setDefault` - Defines if the system groups are also removed from the user's list of default system groups

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.20. METHOD: REMOVEDEFAULTSYSTEMGROUP

Description

Remove a system group from user's list of default system groups.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- string `serverGroupName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.21. METHOD: REMOVEDEFAULTSYSTEMGROUPS

Description

Remove system groups from a user's list of default system groups.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- array:

- string - `serverGroupName`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.22. METHOD: REMOVEROLE

Description

Remove a role from a user.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User login name to update
- string `role` - Role label to add. Can be any of: `satellite_admin`, `org_admin`, `channel_admin`, `config_admin`, `system_group_admin`, `activation_key_admin`, or `monitoring_admin`

Returns

The following return values are available for this method:

- int - 1 on success, exception thrown otherwise

42.23. METHOD: SETDETAILS

Description

Updates the details of a user.

Parameters

The following parameters are available for this method:

- string `sessionKey`
- string `login` - User's login name
- struct - user details
 - string `first_names` (deprecated, use `first_name`)
 - string `first_name`
 - string `last_name`
 - string `email`
 - string `prefix`

- `string password`

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

42.24. METHOD: USEPAMAUTHENTICATION

Description

Toggles whether or not a user uses PAM authentication or basic Red Hat Network authentication.

Parameters

The following parameters are available for this method:

- `string sessionKey`
- `string login` - User's login name
- `int pam_value`
 - `1` to enable PAM authentication
 - `0` to disable

Returns

The following return values are available for this method:

- `int` - `1` on success, exception thrown otherwise

APPENDIX A. REVISION HISTORY

Revision 1.0-15.401 Mass publication of all Satellite 5.6 books	Thu Aug 20 2015	Dan Macpherson
Revision 1.0-15.400 Rebuild with publican 4.0.0	2013-10-31	Rüdiger Landmann
Revision 1.0-15 Final version of documentation suite	Fri Sep 27 2013	Dan Macpherson
Revision 1.0-14 Final changes for Satellite API	Mon Sep 23 2013	Dan Macpherson
Revision 1.0-13 Revised Subtitle, Abstract and Preface for all Guides	Tue Sep 10 2013	Dan Macpherson
Revision 1.0-12 Minor change on configchannel.getEncodedFileRevision() for BZ#998951	Wed Sep 4 2013	Dan Macpherson
Revision 1.0-11 Fixing packages.getPackage() method for BZ#1003565	Wed Sep 4 2013	Dan Macpherson
Revision 1.0-10 Elaboration on configchannel.getEncodedFileRevision() for BZ#998951	Wed Sep 4 2013	Dan Macpherson
Revision 1.0-9 First implementation of QE Review feedback	Thu Aug 29 2013	Dan Macpherson
Revision 1.0-8 Second implementation of tech review feedback	Sun Jul 28 2013	Dan Macpherson
Revision 1.0-7 Corrections for BZ#987245	Wed Jul 24 2013	Dan Macpherson
Revision 1.0-6 First implementation of tech review feedback	Tue Jul 23 2013	Dan Macpherson
Revision 1.0-5 Typo correction	Wed Jul 3 2013	Dan Macpherson
Revision 1.0-4 Final beta updates	Wed Jul 3 2013	Dan Macpherson
Revision 1.0-3 Beta documentation creation	Wed Jul 3 2013	Dan Macpherson
Revision 1.0-2 Minor documentation cleaning	Wed Jul 3 2013	Dan Macpherson
Revision 1.0-1 Minor fix to examples	Wed Jul 3 2013	Dan Macpherson
Revision 1.0-0 Initial creation and generation of the API Guide	Wed Jul 3 2013	Dan Macpherson

